

UCLA

UCLA Electronic Theses and Dissertations

Title

Trajectories of Iterates of Orthogonal Projections

Permalink

<https://escholarship.org/uc/item/0p25z0gt>

ISBN

9798190935981

Author

Kong, Mark Hay-Yat

Publication Date

2026-09-11

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Trajectories of Iterates of Orthogonal Projections

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mathematics

by

Mark Hay-Yat Kong

2026

© Copyright by
Mark Hay-Yat Kong
2026

ABSTRACT OF THE DISSERTATION

Trajectories of Iterates of Orthogonal Projections

by

Mark Hay-Yat Kong

Doctor of Philosophy in Mathematics

University of California, Los Angeles, 2026

Professor Deanna Needell, Chair

We study cases and variants of the following problem: Given a collection of subspaces of a finite-dimensional Euclidean space and a unit vector, iteratively project that vector onto the subspaces in an order determined by a subspace-selection rule. After many iterates, how far away can the final point be from the subspaces? The answer depends on the subspace selection rule. In this dissertation, we study the cyclic rule, in which one cycles over the subspaces, and the greedy rule, in which one always selects a subspace in the collection maximally far from the current position, and some variants. In the cyclic case, we show that if we cycle through T subspaces K times each, then the average squared distance from the final iterate to the subspaces in the collection is at most $O(T^2/K)$ by proving a new result of independent interest: For each positive integer T , a characterization of which points in the complex plane can lie in the numerical range of the product of T (real or complex) projections. Specifically, for each fixed T , the set of possible points forms the region bounded by a sinusoidal spiral. In the greedy case, we give a proof that after k iterations, the largest possible squared distance from the final iterate to any of the subspaces is $O(k^{-1/3})$ via a geometric argument and show that, in the greedy-without-replacement case, no bound on the

final average squared distance going to 0 in terms of only k exists by constructing examples with large final average squared distance.

The dissertation of Mark Hay-Yat Kong is approved.

Mihai Cucuringu

Guido F. Montúfar Cuartas

Hayden K. Schaeffer

Deanna Needell, Committee Chair

University of California, Los Angeles

2026

Contents

Acknowledgments	xvi
1 Introduction	1
1.1 Problem Statement	2
1.2 Relations to iterative algorithms for optimization over Euclidean space . . .	5
2 Cyclic Orderings	12
2.1 Introduction	12
2.1.1 Problem setup	13
2.2 Upper bound	19
2.2.1 Approach	20
2.2.2 Notation for proof	21
2.3 Bounding the range of P	22
2.4 Application to cyclic forgetting bounds	27
2.4.1 Real projections	29
2.5 Appendix for Cyclic Forgetting	30
2.5.1 A geometric interpretation of the forgetting bound in [41]	30
2.5.2 A tighter normalization	31
2.5.3 Optimizing the quadratic form more precisely	31
2.5.4 Proof of Lemma 2.9	32

2.5.5	Any (possibly inconsistent) cyclic sequence of T tasks converges to a cycle of length T	35
2.5.6	Proof of Lemma 2.13	36
2.5.7	Reducing to $\omega = 1$ via quaternions	40
2.5.8	The range of P includes the interior	41
2.5.9	Optimizing sequences for P are coplanar	45
2.5.10	Products of projections with numerical range intersecting $\partial\Gamma_k$	46
2.5.11	Real projections	46
2.5.12	Existence of real realizations	47
2.5.13	Remark on the task dependency on forgetting	51
2.5.14	Forgetting vs. Regret	51
3	Greedy Orderings	53
3.1	Introduction	53
3.2	Setting and Background: Continual Linear Regression	56
3.3	Greedy task orderings: A formal approach and intuition	60
3.4	Benefits of greedy orderings	64
3.4.1	Motivating experiments: Greedy outperforms random ordering	65
3.4.2	Provable benefits for high-rank, “nearly determined” tasks	66
3.5	Failure modes in greedy orderings	68
3.5.1	Greedy orderings can fail where random ones do not	68
3.5.2	Single-pass vs. repetition in greedy orderings	69
3.5.3	Extension: Hybrid task orderings	71
3.6	Discussion and related work	71
3.7	Further related work	76
3.8	Appendix to section 3.4.1: Regression experiments	79
3.8.1	Isotropic data	79

3.8.2	Anisotropic data	81
3.8.3	A note on statistical significance	82
3.9	Appendix to section 3.4.1: Classification experiments	82
3.9.1	Code	82
3.9.2	Experiment details	85
3.9.3	Out-of-domain feature extractors	87
3.9.4	Regularization ablation study	89
3.9.5	Allowing task repetition	89
3.9.6	Rule calculation with partial data	90
3.10	Appendix to section 3.4.2: Proofs for “nearly determined” tasks	91
3.10.1	Optimality guarantee when $r = d - 1$	91
3.10.2	Loss bound when $r = d - 1$	93
3.11	Appendix to section 3.5.1: Lower bound’s “adversarial” constructions	94
3.11.1	General dimension construction and proof (theorem 3.12)	94
3.11.2	Adversarial 3d construction (example 3.11)	102
3.12	Appendix to section 3.5.2: Single-pass vs. repetition	109
3.12.1	Appendix to the upper bound for greedy orderings with repetition (theorem 3.13)	109
3.12.2	Regression experiments on single-pass vs. repetition	122
3.13	Appendix to section 3.5.3: Hybrid task ordering	124
3.13.1	Hybrid ordering scheme	124
3.13.2	Hybrid ordering upper bound	127
3.13.3	Hybrid ordering experiments	132
3.14	Appendix to section 3.4.1: Code snippet for regression experiments	132
3.15	Lower bound technical appendix: Delta positivity proof	140
3.15.1	Proof outline	141
3.15.2	Auxiliary: Algebraic inequalities	141

3.15.3	Proof body	143
3.15.4	Conclusion	170
3.16	Lower bound technical appendix: Properties of the recursive construction . .	172
3.16.1	Proof outline	173
3.16.2	Full proof	176
4	Conclusion	197

List of Figures

1.1	A sample sequence of iterates. Image created at geogebra.org	5
3.1	Intuition. Consider a collection of jointly-realizable linear regression tasks (e.g., A, B, C, D). Each task has an affine solution space (e.g., where $\mathbf{X}_A \mathbf{w} = \mathbf{y}_A$), and $\mathbf{w}_\star$ is an “offline” joint solution at the intersection of all tasks. Employing a projection perspective on learning in continual models [41, 42], transitions between <i>dissimilar</i> tasks ($A \rightarrow D \rightarrow B \rightarrow C$) intuitively lead to faster convergence toward the intersection compared to transitions between <i>similar</i> tasks ($A \rightarrow B \rightarrow C \rightarrow D$).	56
3.2	Projection dynamics.	60
3.3	Task ordering comparison. Transitioning between dissimilar tasks consistently outperforms random transitions, with Greedy MR and MD achieving comparable performance.	66
3.4	Learning an adversarial collection.	69
3.5	The effect of repetitions.	70
3.6	Hybrid ordering experiment.	71
3.7	Task typicality.	74

3.8	Comparing orderings for varying dimensions d and ranks r of the data matrices, for isotropic data. $T = 50$. For such isotropic data, for fixed ratio r/d , the random ordering performance changes relatively little. In contrast, greedy orderings that prioritize dissimilarity benefit from a lower dimension when r/d is fixed, and Greedy Min Dist benefits from a higher dimension (to see this from the plots, focus on single columns in the grid). This may be related to some notion of increased task “density” in lower dimensions: when r/d is fixed, increasing d increases $d - r$, expanding the set of possible task projections (see (3.2)). As a result, a fixed number of tasks T covers this space more sparsely in higher dimensions. In lower dimensions, the same T tasks yield denser coverage, which may cause greedy dissimilarity-based selection to take steps with large projections. In all strategies, for fixed ambient dimension, higher task rank consistently yields improved performance (focus on single rows). This is because the solution subspaces are of rank $d - r$, so increasing r (with fixed d) lowers the subspace rank, increasing the distances between them and resulting in larger projections.	80
3.9	Comparing orderings for varying task count T, for isotropic data. $d = 100$, $r = 10$. Dissimilarity-based greedy strategies become more effective as the number of tasks increases, while Greedy-Min-Dist becomes less effective. We can give a heuristic explanation for the improvement to greedy strategies: In an isotropic setting (so task directions are sampled uniformly), increasing the number of tasks increases the coverage of the unit sphere. This results in a higher probability of encountering task pairs with large angular separation between their solution subspaces, which greedy ordering utilizes. .	81

- 3.10 **Comparing orderings for varying dimensions d and ranks r of the data matrices, for anisotropic data.** $T = 50$. Compared to the isotropic case (fig. 3.8), we observe slower rates for all strategies. This may be related to how the pairwise distances between task solution subspaces are smaller, due to the higher correlation in the anisotropic case, though such a causation was not experimentally tested. As rank increases (focusing on a single row in the grid), the Maximum Residual (MR; definition 3.8) ordering underperforms and aligns with the random one very closely. We do not have an explanation of this alignment. The underperformance may be related to the combination of high rank and strong intra-task correlation, which may lead to ill-conditioned data matrices (for each task). In such a case, small perturbations, or steps, in the solution space may cause disproportionately large changes in residuals, which may cause MR to select tasks with large residuals that advance the iterate only marginally toward the intersection ($\mathbf{w}_*$). 83
- 3.11 **Comparing orderings for varying task count T , for anisotropic data.** $d = 100$, $r = 10$. Compared to the isotropic case (fig. 3.9), the benefit to greedy orderings from increasing the number of tasks T is much smaller. The harm to Greedy min dist is also smaller than it was in the isotropic case. We can heuristically explain this as follows: In the anisotropic case, a large number of tasks must be added to induce the substantial “angles” that greedy orderings can exploit. Thus, while more tasks does cover more directions, it covers the space of directions more slowly in the anisotropic case. 84

3.12	Showing variations across different experiments. Same as fig. 3.3, we have $T = 50$, $r = 10$, $d = 100$, with random isotropic data. Light lines represent each individual experiment. While minor variations exist across experiments, the low standard error at only 10 runs confirms the consistency of the results. We did not test whether, for fixed data, Greedy MD tends to beat Greedy MR, even though this appears true for the averages.	84
3.13	Comparison of orderings under different pretraining setups.	88
3.14	Regularization strength ablation study.	89
3.15	The effect of task repetition on CIFAR-100 continual classification.	90
3.16	Greedy MR rule calculation using partial data.	91
3.17	Numerical positivity of $\Delta(d) \triangleq \min_{\{t,k t \in \{2,\dots,d-1\}, k \in \{t,\dots,d-1\}\}} \Delta_{t,k}$	102
3.18	Example of continual learning the 3d adversarial construction with greedy ordering. Chosen tasks coincide for MD and MR greedy orderings. $K = 1000, T = 3999$	106
3.19	Final average loss after greedy ordering for the 3d adversarial construction, for different task counts. $T = 4K - 1$	106
3.20	The effect of repetitions for varying dimensions d and ranks r of the data matrices, for isotropic data. $T = 50$. Random orderings without replacement consistently outperform their with-replacement counterparts. In contrast, greedy orderings with repetition outperform the single-pass variant. As explained in section 3.5.2, one can heuristically argue that repetition in greedy orderings is beneficial because it enables larger steps (and faster convergence to the joint solution $\mathbf{w}_\star$).	123

3.21	The effect of repetitions for varying task count T , for isotropic data. $d = 100, r = 10$. As the task count increases, the differences between with and without repetition diminish. However, in all subplots we only learn the first 50 tasks. In the left subplot, the effect of repetition becomes pronounced in the latter parts of the task sequences. Repetition offers less benefit when many diverse, unexplored tasks remain.	124
3.22	The effect of repetitions for varying dimensions d and ranks r of the data matrices, for anisotropic data. $T = 50$. Previously in section 3.8 (fig. 3.10), we explained that the performance of all ordering strategies deteriorates when the pairwise distances between task solution subspaces are small. This effect is even more pronounced for low-rank tasks (left columns), where the complementary high-rank solution subspaces overlap substantially. We also observe that in those low-rank regimes, different orderings exhibit more similar performance, and consequently, repetitions become less impactful. While we cannot fully explain the small performance degradation observed when allowing repetitions in greedy ordering for low-dimensional, low-rank settings (top-left subfigure), it may be related to the slower convergence to the joint solution $\mathbf{w}_\star$, nullifying the effect of the loss upper bound induced by the distance to $\mathbf{w}_\star$ (proposition 3.6; see also fig. 3.23 below).	125

3.23	The distance to $\mathbf{w}_\star$ is a loose upper bound on the loss for high-similarity tasks. Greedy ordering with $d = 60$, $T = 50$. While repetitions do lead to a slightly faster decrease in the squared distance to $\mathbf{w}_\star$, this decrease remains slow when tasks are highly similar (as in the low-rank setting on the left). Consequently, the upper bound of proposition 3.6 becomes looser, as the loss itself decreases more rapidly. This discrepancy makes it difficult to draw firm conclusions about the convergence of the loss, including the exact impact of repetitions. A similar gap between the loss and the distance to the joint solution $\mathbf{w}_\star$ in highly similar tasks was also noted by Evron et al. [41, Section 5.1 therein].	126
3.24	Hybrid performance for varying dimensions d and ranks r of the data matrices, for isotropic data. $T = 50$. In high-rank and/or low-dimensional settings, the rank-dependent upper bound employed by the hybrid strategy in this case is lower, prompting an earlier transition from the greedy to the random phase. For $r/d = 0.1$, the random phase within the hybrid method has slightly lower slope than fully random ordering (so the loss decreases slower), possibly because the initial greedy steps remove tasks with large decrements from the remaining task set. The slopes are hard to compare for $r/d = 0.5$ and very slightly faster for $r/d = 0.9$.	133
3.25	Hybrid performance for varying task count T, for isotropic data. $d = 100$, $r = 10$. We see similar trends. The previously observed difference between the random phase following the greedy phase and fully random ordering is smaller at higher task counts, possibly because more tasks with large decrements remain available for selection.	133
3.26	Numerical proof of lemma 3.47 for $d < 100,000$. Using the recursive definition of x_k , we calculated the sequence for each value of d , $\forall k \in [d]$, and compared with $\tilde{x}_k$.	174

List of Tables

- 3.1 Loss bounds in continual linear regression over jointly realizable tasks (based on Table 1 of Evron et al. [43]). The presented bounds are “worst case”: upper bounds apply to any collection of T jointly realizable tasks, while lower bounds are achieved by specific constructions. Bounds for random orderings apply to the expected loss. We omit scaling terms ($\|\mathbf{w}_\star\|^2 R^2$) and constant multiplicative factors (which are mild). Notation: k = iterations; d = dimensions; $\bar{r}, r_{\max}$ = average/maximum data matrix ranks; $a, b \triangleq \min(a, b)$ 110

ACKNOWLEDGMENTS

Chapter 2 is a version of William Swartworth, Deanna Needell, Rachel Ward, Mark Kong, and Halyun Jeong, “Nearly Optimal Bounds for Cyclic Forgetting,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 68197–68206, 2023. doi: 10.52202/075280-2982.

Professor Deanna Needell suggested the problem of improving forgetting bounds. William Swartworth suggested the problem of improving the bound on the numerical range of a product of projections. All authors contributed to proving theorems and writing and framing the paper. Mark Kong, William Swartworth, and Halyun Jeong wrote code to help gain intuition about the problem. Professor Deanna Needell and the UCLA DTG grant financially supported me to present this at NeurIPS 2023.

Chapter 3 is a version of Matan Tsipory, Ran Levinstein, Itay Evron, Mark Kong, Deanna Needell, and Daniel Soudry, “Are Greedy Task Orderings Better Than Random in Continual Linear Regression?” *Advances in Neural Information Processing Systems*, vol. 38, pp. 172355–172456, 2025. doi:10.52202/085713-5192.

Matan Tsipory, Ran Levinstein, Itay Evron, Deanna Needell, and Daniel Soudry formulated and directed the project. Itay Evron, Mark Kong, Ran Levinstein, and Matan Tsipory proved theorems that appear in the final paper. Itay Evron, Ran Levinstein, and Matan Tsipory designed and conducted experiments using code. All authors contributed to writing the paper. Professor Deanna Needell financially supported me to present this at NeurIPS 2025 through NSF DMS-2408912.

VITA

2017	B.S. Mathematics, Harvard University
2024	M.A. Mathematics, UCLA
2021–2026	Teaching Assistant, UCLA

PUBLICATIONS

M. Tsipory, R. Levinstein, I. Evron, M. Kong, D. Needell, and D. Soudry. “Are Greedy Task Orderings Better Than Random in Continual Linear Regression?” *Advances in Neural Information Processing Systems 38 (NeurIPS 2025)*, 2025.

W. Swartworth, D. Needell, R. Ward, M. Kong, and H. Jeong. “Nearly Optimal Bounds for Cyclic Forgetting.” *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*, 2023.

Chapter 1

Introduction

This dissertation concerns a problem I find natural enough to be stated as a linear algebra problem of its own right, but also can be described using the language of certain iterative algorithms for certain optimization problems in Euclidean space. From the latter perspective, studying this problem allows us to understand structure that may appear in the context of such iterative algorithms or optimization problems.

By “optimization problem over $\mathbf{R}^d$ ” I mean a problem of the form “Give an algorithm that takes as input a function $f : \mathbf{R}^d \rightarrow \mathbf{R}$ belonging to some function class and outputs a point $p \in \mathbf{R}^d$ with the aim to make $f(p)$ as large (or small) as possible.” Thus each choice of function class and method of presenting f as input to the algorithm¹ gives an optimization problem.

By “iterative algorithm” I mean² an algorithm for solving an optimization problem over $\mathbf{R}^d$ in which one repeatedly applies a procedure $\mathbf{R}^d \rightarrow \mathbf{R}^d$ starting with some initial point $p \in \mathbf{R}^d$, chosen by some initialization algorithm, until some stopping condition, such as number of repetitions, and outputs the final point.

Example applications of such algorithms include algorithms for solving matrix equations

¹For example, as an evaluation oracle or as a formula

²It is possible to use these words in more general ways, but we will not discuss them in this dissertation.

and optimization algorithms for machine learning.

Some of these algorithms, such as the Kaczmarz and Gauss-Seidel methods for solving matrix equations and minibatch gradient descent for machine learning, are more closely related to our problem, as discussed below, while others such as the Jacobi method for solving matrix equations and Adam for machine learning are less closely related to the problem in its stated form.

We first give a statement of the problem, then describe several related algorithms and their relation to the problem in more detail.

1.1 Problem Statement

Problem Setup 1.1. A **selection rule on $\mathbf{R}^d$** is a function³ that takes in a vector in $\mathbf{R}^d$, an indexed family of subspaces⁴ of $\mathbf{R}^d$, and possibly some state and outputs the index of one of the subspaces⁵. Our notation for a selection rule is $\mathfrak{S}(\{S_t\}_{t \in \mathcal{T}}, w_0, \text{data})$ where data is an optional argument that one may use e.g., to maintain state.

A loss aggregation function is a function that takes in a collection of losses and outputs a single loss. Here, a “loss” is an output of a loss function, which in this dissertation will always be a nonnegative real. Our notation for a loss aggregation function is $\mathfrak{L}(\cdot)$.

³We use the word “function” informally: In principle, it may be random or partial and may have other caveats I haven’t imagined. By “partial”, I mean not only that it may make assumptions about how the family is indexed, but also that it may disallow certain collections of subspaces if one wants to focus analysis only on certain collections. For example, the cyclic selection rule described below assumes the collection is finite and indexed by $\{1, \dots, T\}$.

⁴In this dissertation, the word “subspace” by itself always means a subspace containing the origin, as opposed to an affine subspace. The questions considered in this dissertation apply to affine subspaces, which will become a useful object of analysis in the greedy case, but in such cases we will specify “affine subspace”.

⁵We output an index instead of the subspace itself purely for notational convenience. It will, at times, be convenient to refer to the output of this selection rule as the subspaces; In this case, we consider the input indexed family to be the set of subspaces indexed by itself.

Example 1.2.

- Examples of selection rules:
 - Cyclic: Let T denote the number of subspaces, which we assume does not change from call to call. Maintain as state a number in $\{1, 2, \dots, T\}$, initialized at 1. Each time the selection rule is called, output and then increment the state modulo T .
 - Greedy: Output the index of the subspace that is farthest away from the input vector⁶. If there is a tie, this selection rule may output any of them.

Alternatively, given any selection rule $\mathcal{S}$, we say $\mathcal{S}$ is Greedy if it always outputs the index of one of the subspaces that is farthest away from the input vector.

- Examples of loss aggregation functions:
 - Max loss: Returns the maximum of the input losses.
 - Average loss: Returns the average of the input losses.

Each choice of a selection rule $\mathfrak{S}$, loss aggregation function $\mathfrak{L}$, and list of parameters (such as number of steps I , number of subspaces T , or dimension d ; If these are taken as parameters, one removes these from the list of variables we take a supremum over below) defines a problem:

Problem 1.3. Let $S_1, \dots, S_T \subset \mathbf{R}^d$ be subspaces. Let $w_0 \in \mathbf{R}^d$ be a unit vector. Iteratively define a sequence of vectors $w_1, \dots, w_I \in \mathbf{R}^d$ as follows: For each $1 \leq i \leq I$,

1. Use $\mathfrak{S}$ to pick $t_i \in \{1, 2, \dots, T\}$.
2. Define w_i to be the projection⁷ of w_{i-1} onto S_{t_i} .

⁶This may be undefined if the indexed family of subspaces is infinite. One can define variants of this selection rule to account for this case, but the same bounds from this dissertation will apply to the natural extensions I can think of so we will not consider this in detail.

⁷In this dissertation, “projection” always means “orthogonal projection” unless otherwise indicated

For each $1 \leq j \leq T$, define the loss with respect to S_j to be $\ell_j := \text{dist}(w_I, S_j)^2$.

In terms of the chosen parameters, estimate/give bounds on

$$\sup_{\substack{T, d \in \mathbf{Z}_{>0} \\ S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ w_0 \in \mathbf{R}^d, \|w_0\|=1 \\ \text{such that [optional list of conditions]}}} \mathfrak{L}(\ell_1, \dots, \ell_T).$$

(For a randomized selection rule, one could estimate/bound the expected value or other quantities of interest on the distribution of the outputs of the loss aggregation function. However this dissertation mostly considers only deterministic selection rules⁸.) (In this problem setup, we tie the subspaces we project to with the subspaces we take the final loss over. In principle, one could consider untying them, so you project onto subspaces from one family and then calculate the losses with respect to another, as some kind of “feature learning”. We will not do this in this dissertation.)

In other words, how large can an adversary who controls the initial configuration force your aggregate loss?

We give an illustration of the problem in Figure 1.1.

Here, we initialize at an arbitrary point w_0 and iteratively project onto subspaces. The subspaces shown are one line and two planes, showing that we specifically allow subspaces of codimension greater than one. The selection rule in the figure is neither cyclic nor greedy.

It is routine to show one can write max instead of sup in all cases considered in this dissertation (that is, all such supremums are attained), though I do not have a proof of this that does not require small modifications for each formulation.

⁸There are some cases where the selection rules are not fully specified and in principle could be randomized, but in such cases our bounds apply to adversarially selected outputs

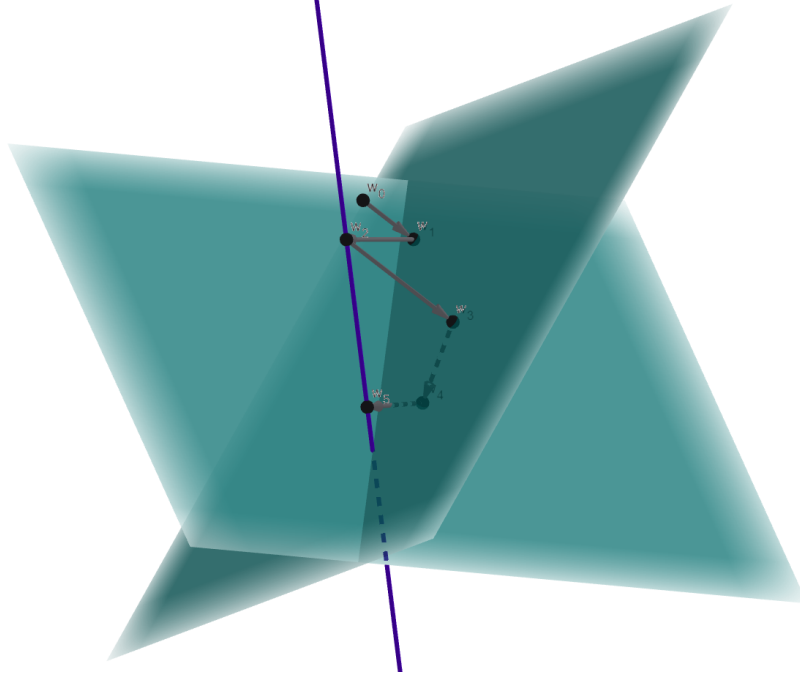


Figure 1.1: A sample sequence of iterates. Image created at [geogebra.org](https://www.geogebra.org)

1.2 Relations to iterative algorithms for optimization over Euclidean space

The iterative procedure described in the problem statement with various selection rules appears in several contexts:

- Equivalent to Kaczmarz methods: Also known as the algebraic reconstruction technique in the field of computed tomography, the basic Kaczmarz method is a row-action methods for solving matrix equations of the form $Ax = b$ where A is a (not necessarily square) matrix with given entries, b is a column vector with given entries, and we wish to find entries of a vector x close to satisfying the equation, for various notions of “close to”.

It operates by starting with a (possibly informed) guess w_0 , then repeatedly using some row-selection rule to pick a row index r and projecting each iterate w_i to the affine hyperplane $A_{r,:}x = b_r$ to get the next iterate w_{i+1} . If there exists a true solution

x_* with $Ax_* = b$, then by translating x_* to 0 and rescaling the initial guess to have norm 1, we recover the above iterative procedure exactly; If no such solution exists, we may replace “subspaces” with “affine subspaces” in the definition of the algorithm. This generalizes to Block Kaczmarz, defined in the same way as in the previous paragraph except selecting multiple row indices in each step using a multi-row-selection rule. Indeed, the solution space for a collection of r rows is an affine subspace of codimension at most r (assuming it is nonempty).

- [41, Other optimization methods] observed that our setting is, in a sense, Lagrange dual to block coordinate ascent/descent in a certain setting. (For convenience, I will write “coordinate descent” to mean coordinate ascent/descent.)

Coordinate descent is the iterative optimization algorithm that seeks to minimize a function $f : \mathbf{R}^d \rightarrow \mathbf{R}$ by, in each iterate, selecting one coordinate to modify and finding the lowest value of f obtainable by modifying only that coordinate.

In contrast, in the Kaczmarz method mentioned above, each iteration may modify all coordinates.

To see the duality in the simplest case, we consider using the standard Kaczmarz method (as opposed to the block Kaczmarz method) to solve $X\vec{w} = \vec{y}$ initializing at $\vec{w}_0 = \vec{0}$. The iterates under the Kaczmarz method consist of iteratively adding scalar multiples of rows of X to $\vec{w}$, so all iterates are in the range of X^T . Thus we may introduce “dual coordinates” α , whose relevance to duality will be explained shortly, as follows: We parametrize the range of X^T via $\alpha \in \text{dom}(X^T)$, $\alpha \mapsto X^T\alpha$. Thus each vector $w \in \text{range}(X^T)$ may have multiple choices of dual coordinates. Then the number of coordinates of α is the number of rows of X , and projecting onto the solution space of row i can be obtained by varying only the i th coordinate of α .

This is not yet coordinate descent, as we have not specified an objective function for α . To do this, recall that running the Kaczmarz method to convergence to a solution for $\vec{w}$

is equivalent to solving the optimization problem of minimizing⁹ $\frac{1}{2}\|\vec{w}\|^2$ subject to the constraint $X\vec{w} = \vec{y}$. It turns out that the corresponding updates to α are exactly those obtained by applying coordinate descent to the Lagrange dual problem of maximizing $\min_{\vec{w}} \frac{1}{2}\|\vec{w}\|^2 - \langle \alpha, X\vec{w} - \vec{y} \rangle$, where one can check that the choice of $\vec{w}$ achieving this minimum is always $\vec{w} = X^T \alpha$.

- It is a special case of incremental gradient descent and of minibatch gradient descent. In general, incremental gradient descent[50, 85] considers the case when you wish to optimize a function of the form $\sum_{j \in J} f_j$ where the f_j are smooth loss functions and J is an indexing set. Its iteration rule is defined as follows: If one starts an iteration at a vector $\vec{w}$, then to get the next iterate, you select an index $j \in J$ and a step size $\eta \in \mathbf{R}$ and update to

$$\vec{w} - \eta \nabla f_j(\vec{w}).$$

Minibatch gradient descent is a variant in which one models the loss function as coming from many data points and each data point has its own loss function, and the goal is to minimize the average loss or total loss. In the special case that the minibatches partition the full dataset, one can use the notation above and take J to be the set of minibatches and f_j to be the sum of the losses corresponding to datapoints in the j th minibatch.

To realize our setup as a special case of this, for each subspace S_j , consider the loss function $\ell_j(\cdot) := \text{dist}(\cdot, S_j)^2$ as above.

Then suppose one attempts to minimize $\sum_j \ell_j$, and we take $f_j = \ell_j$ to be the loss function of a minibatch. In Chapter 2, we will see a realization of this.

⁹There is a bit of freedom here in that it also minimizes any strictly increasing function of $\|\vec{w}\|$, so perhaps it is less surprising that one can choose a monotonic function making the duality work out, and if I wanted another optimization method I could have picked a different function. That being said, the factor of $\frac{1}{2}$ is harmless here as removing it only rescales the dual coordinate, and it is still coordinate descent.

Then if $\vec{w}'$ denotes the projection of $\vec{w}$ onto S_j , then

$$\vec{w}' = \vec{w} - \frac{1}{2} \nabla \ell_j(\vec{w}).$$

Therefore, the iterates are the same as if one applied minibatch incremental gradient descent with step size $\frac{1}{2}$. If minibatches are selected cyclically, this gives the cyclic task ordering, and if they are selected by always selecting the minibatch with the worst loss, that gives the greedy task ordering.

- Projections onto Convex Sets is an approach to solve a problem where one is given a list of convex constraints and wishes to find a point satisfying all of them. This problem is called the convex feasibility problem. It works by iteratively projecting onto the solution spaces of the convex constraints.

The Kaczmarz method mentioned above is standard special case of POCS where the convex sets are subspaces[16].

Another special case of POCS is when the convex sets are half-spaces. In this case the method is called the Agmon-Motzkin-Schoenberg method, in which case one often either selects which half-space to project to cyclically or by always selecting a most violated constraint[114], matching both our setups.

As the result of projecting a point onto a halfspace not containing the point is the same as the result of projecting it onto the supporting hyperplane, the trajectory is one obtainable by a sequence of such projections onto hyperplanes.

- Subgradient descent is an iterative method for optimizing a convex function f . Its iterations are defined as follows: If one starts at $\vec{w}$, then to get the next iterate, one selects a subgradient $\vec{v} \in \partial f$ at $\vec{w}$ and updates $\vec{w} \rightarrow \vec{w} - \eta \vec{v}$ where η is the learning rate.

The greedy ordering is a special case of subgradient descent.

To see this, say our collection of subspaces is $S_1, \dots, S_T$. Define the loss function

$$f(\vec{w}) := \max_{t \in \{1, \dots, T\}} \text{dist}(\vec{w}, S_t).$$

Then if one takes the Polyak step size $\eta = \frac{f(\vec{w}) - f^*}{\|\vec{v}\|^2}$, where $f^* = 0$ is the optimal value of the loss function, then as long as the subgradient is always chosen to come from exactly one of the subspaces S_t , the update always projects directly onto S_t (which, by the definition of f , must be one of the subspaces farthest from $\vec{w}$ to have its normal direction be a subgradient).

This was studied in the preprint [133] whose Theorem 2.3 claimed¹⁰ to give the exact largest possible loss in the general Polyak step size setting after I iterates whose asymptotics are $O(I^{-1/4})$ (so for squared distance, the asymptotics would be $O(I^{-1/2})$), and Theorem 2.6 gives a construction realizing this sharp bound which is essentially of the form we study in this dissertation.

- In the pure greedy algorithm[34, 38], also known as the matching pursuit algorithm in signal processing[80], one has a “dictionary”, which is a collection of unit vectors $\vec{a}_i \in \mathbf{R}^d$, and a target vector $\vec{v} \in \mathbf{R}^d$, and one seeks coefficients λ_i such that $\sum_i \lambda_i \vec{a}_i \approx \vec{v}$. The algorithm is initialize all coefficients to 0 and then to iteratively select the vector $\vec{a}_i$ whose inner product with the residual is biggest in absolute value, and modifies λ_i to make that inner product 0.

If one considers the orthogonal complements of the dictionary vectors as our subspaces, this is exactly the greedy selection rule.

- [19] study the method of remotest projections and alternating projections. Both settings start with a collection of closed subspaces $S_1, \dots, S_T$ of a general Hilbert space

¹⁰Actually, it is not hard to construct a counterexample to this theorem as stated in the general Polyak setting. But if one adds the assumption that all subgradients used in the iterations have the same norm, which is true in our setting, then the proof appears to work

whose intersection is $\{\vec{0}\}$ and studies iterates of a point w under iterated projections to these subspaces. The method of remotest projections always projects to a subspace that is maximally far from its current position, whereas the method of alternating projections cyclically projects onto the subspaces, just like our cyclic ordering.

These equivalent formulations have been studied with various other loss functions or performance criteria:

- Some papers study the distance to a true solution.

For example, [19] studied the convergence rate of the iterates under remotest projections and alternating projections to $\vec{0}$.

POCS studies multiple types of loss functions, such as the distance to the nearest common intersection of the convex constraints and also the distance to the most violated convex constraint[16].

- [67] bounded the size of the residual in matching pursuit in terms of the smallest possible sum of the sizes of the coefficients in a linear combination of vectors in the dictionary making up the target.

This dissertation takes the perspective where we care primarily about minimizing the losses realized by each of the subspaces, combined via some loss aggregation function like average or max.

An upper bound for the max loss aggregation function certainly gives an upper bound for the average loss aggregation function, and the methods we use in the following chapters will generally work for the max loss aggregation function. However, in the chapter on cyclic orderings, we focus on the average loss aggregation function due to the context of the paper.

In the next chapter, we give a new upper bound on the average loss for the cyclic ordering rule and write about its connections to forgetting in continual learning. As part of the proof, we also give an exact classification of which points in $\mathbf{C}$ can lie in the numerical range of the product of T (real or complex) projections.

In the following chapter, we study greedy orderings, where one selects the subspace farthest away from the current position, and variants such as disallowing the revisiting of subspaces. For the orderings mentioned in this paragraph (we defer discussion of other orderings to that chapter), we prove a nontrivial upper bound that goes to 0 in the greedy case and construct examples showing that if we always select a subspace farthest away from the current position only among unused subspaces, the final average loss can be bounded away from zero even if the number of iterations is arbitrarily large, even in 3D.

Chapter 2

Cyclic Orderings

This chapter is based on [119].

2.1 Introduction

The main mathematical problem we study in the paper has connections to “continual learning” or “lifelong learning”[26, 95, 121, 127], which refers to the problem of deciding how to make a system (say, a machine learning system) learn when the data with which it is supposed to learn is presented not all at once but one part at a time.

In the field of continual learning, one consideration is “forgetting” or “catastrophic forgetting”[52, 65, 123], which refers to how a system trained on some part X of the data may, during the course of training on other parts, get worse at part X. Based on the setting, one may wish to control how much or what aspects of each part one wants the system to forget during training¹.

Depending on the setting, these parts may or may not overlap and may or may not appear more than once during training.

¹For instance, if one wishes the system to generalize, one may want the system to forget the aspects that don’t generalize and not forget the aspects that do generalize. However, this will not be the focus of this paper.

The problem we consider in this paper relates to the case when the parts repeat cyclically; That is, one has a finite collection of “parts” which the system sees one at a time, then the system sees each part in the same order a second time, and so on for a finite number of cycles.

This setting has been studied before. For example, financial or search engine data has been reported as having a “day of the week effect”[98, 131]. For another example, one method for solving linear systems is the method of cyclic alternating projections, originated for two subspaces by Von Neumann[125] and extended to finitely many subspaces by Halperin[53]. When the linear system is expressed in matrix form (i.e., when you are given coefficients), is known as a (block) Kaczmarz method[64].

[41] asked (among several other questions) the question of what bounds can be obtained in a certain specific cyclic continual learning setting to be described in the next section and proved several bounds. In this paper, we improve these bounds.

2.1.1 Problem setup

In this section we give a mathematical statement of the problem. Then we relate the problem to continual learning.

We state the problem in a self-contained way:

Problem 2.1. Let $S_1, \dots, S_T \subseteq \mathbf{R}^d$ be subspaces, and let $w_0 \in \mathbf{R}^d$ be a unit vector.

Define a sequence of vectors $w_1, w_2, \dots, w_I \in \mathbf{R}^d$ as follows:

1. For k from 1 to $K := \lceil \frac{I}{T} \rceil$:
 - (a) For t from 1 to T :
 - i. Define $w_{(k-1)T+t}$ to be the projection of $w_{(k-1)T+t-1}$ to S_t .

Thus, the case $k = 1, t = 1$ defines w_1 , then $k = 1, t = 2$ defines w_2 , and so on until the last

iterate w_{KT} . All iterates after w_I are ignored².

For any $w \in \mathbf{R}^d$ and any subspace $S \subseteq \mathbf{R}^d$, define

$$L_S(w) := \text{dist}(w, S)^2.$$

In terms of T and I , what are the largest possible values of

$$\max(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I))$$

and

$$\text{avg}(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I))?$$

In other words, we wish to calculate or at least give bounds on

$$\begin{aligned} F_{\text{cyclic}, \max}(\text{subspaces} = T, \text{iters} = I, \text{cycles} = I/T) \\ := \sup_{\substack{d \in \mathbf{Z}_{\geq 0} \\ S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ w_0 \in \mathbf{R}^d, \|w_0\|=1}} \max(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I)) \end{aligned}$$

and

$$\begin{aligned} F_{\text{cyclic}, \text{avg}}(\text{subspaces} = T, \text{iters} = I, \text{cycles} = I/T) \\ := \sup_{\substack{d \in \mathbf{Z}_{\geq 0} \\ S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ w_0 \in \mathbf{R}^d, \|w_0\|=1}} \text{avg}(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I)). \end{aligned}$$

We omit “cyclic” when clear from context. Any two of the inputs to $F_{\max}$ or F_{avg} determine the third, so below we will only specify two of the inputs.

Throughout this paper, we will usually assume we have completed a whole number of

²We compute all the way to w_{KT} only for convenience of writing the algorithm. In practice one could stop after computing w_I .

cycles, so I is a multiple of T . In this case, we let m denote the number of cycles and write our bounds in terms of T and m instead of T and I .

For I/T large, the bounds obtained in the whole-number-of-cycles case give bounds in the general case, as the final iterate can equivalently be obtained by applying $\lfloor I/T \rfloor$ cycles starting at the $I \bmod T$ th iterate.

For the remainder of this paper, we assume I is a multiple of T and let $m = I/T$.

Notation 2.2. For any problem data S , which consists of the ordered collection of subspaces and initialization vector w_0 , $F_S(I)$ denotes the average loss after I iterations.

Thus after m cycles of T tasks, we wish to bound $\sup_S F_S(mT)$.

Relation to Continual Learning

The system we train is a linear model. Each datapoint is a pair $(\vec{x}, y) \in \mathbf{R}^d \times \mathbf{R}$.

We assume there exists at least one solution $\vec{w}_*$ that fits all datapoints (i.e., so that $\langle \vec{w}_*, \vec{x} \rangle = y$ for all datapoints $(\vec{x}, y)$).

The data are presented to our system by being grouped into “tasks”, each of which is a list of some of those datapoints.

At each training iteration, the system has a current state $\vec{w} \in \mathbf{R}^d$ which should be thought of as a linear map $\mathbf{R}^d \rightarrow \mathbf{R} : \vec{x} \mapsto \langle \vec{w}, \vec{x} \rangle$ that attempts, for each datapoint $(\vec{x}, y)$, to send $\vec{x}$ to y .

The training algorithm is “at each step, jump to the closest solution to the given task”. That is, if the current state is $\vec{w}_t$ and the task when stacked into a matrix is $(X_t, \vec{y}_t)$ (so each datapoint $(\vec{x}, y)$ is turned into a row of X_t and the corresponding entry in $\vec{y}_t$), then:

1. The set vectors $\vec{w}' \in \mathbf{R}^d$ satisfying $X_t \vec{w}' = \vec{y}_t$ forms an affine subspace of $\mathbf{R}^d$
2. The next iterate $\vec{w}_{t+1}$ is the projection of $\vec{w}_t$ onto that affine subspace.

This is equivalent to the algorithm obtained if, at each iteration, you run gradient descent to convergence on the square loss $\vec{w} \mapsto \|X_t \vec{w} - \vec{y}_t\|^2$.

Finally, to relate this to catastrophic forgetting, we give a way to quantify forgetting suggested by [41]: If we let T denote the number of tasks, $S = ((X_1, \vec{y}_1), \dots, (X_T, \vec{y}_T))$ the T -tuple of tasks, and m the number of cycles, then we quantify the “forgetting” of the final iterate to be its average square loss

$$F_S(mT) := \frac{1}{n} \sum_{i=1}^n \|X_t \vec{w}_n - \vec{y}_t\|^2.$$

The reason this may be considered “forgetting” is that during training, for each task, there was some iterate which had 0 loss on that task. Thus any contribution to the test loss comes from the final iterate performing worse on that task than some earlier iterate.

To relate this formulation to the version in the previous section, we make two modifications: First, we translate and rescale the setup so that the common solution is at $\vec{w}_* = \vec{0}$ and our initial guess is a unit vector, or equivalently set all $\vec{y}$ to $\vec{0}$. Second, as stated, this setup is slightly more general than the one introduced in the previous section. Indeed, given a task $(X, \vec{y}) = (X, \vec{0})$, the loss function for this task may not be proportional to the squared distance to the solution space if the data is not symmetrical, as the loss at $\vec{w}$ is $\|X\vec{w}\|^2 = \vec{w}^T X^T X \vec{w}$, and $X^T X$ need not be a multiple of the identity.

However, this loss is always within a factor of $\|X\|_2^2$ of this distance, so we rescale all data matrices by the largest squared operator norm and aim for a bound under the assumption $\|X\|_2 \leq 1$.

Remark 2.3. If the data is so symmetric that $X^T X = I$ for every matrix X , then this recovers our prior situation exactly. This realizes our problem as a special case of minibatch gradient descent.

Observations and Prior Work

A trivial bound is

$$\sup_S F_S(mT) \leq 1,$$

as the initial iterate is at most 1 away from the origin and future iterates only get closer.

[41] previously obtained these two bounds:

$$\sup_S F_S(mT) \leq \frac{T^2}{\sqrt{mT}},$$

$$\sup_S F_S(mT) \leq \frac{T^2(d - r_{\max})}{2mT}$$

where $r_{\max}$ is the largest rank of a subspace being projected *away from* (so $d - r_{\max}$ is the smallest rank of a subspace being projected on to).

We improve this asymptotically to:

Theorem 2.4.

$$\sup_S F_S(KT) \leq O\left(\frac{T^2}{K}\right)$$

We do this by using one of the lemmas [41] proved to get their dimension-independent bound $\frac{T^2}{\sqrt{mT}}$: They showed that if A denotes the linear map obtained by composing the T projections in each cycle, then letting $\Delta_m(\vec{u}) = \|A^m \vec{u}\| - \|A^{m+1} \vec{u}\|$,

$$F_S(mT) \leq \frac{T-1}{2} \max_{\vec{u}: \|\vec{u}\|=1} \Delta_m(\vec{u}) \leq (T-1) \|A^m(I-A)\|.$$

[41] continued by bounding

$$\|A^m(I-A)\| \leq \sqrt{\frac{T}{m}}.$$

For comparison, our proof will proceed by obtaining the bound

$$\|A^m(I-A)\| \leq O\left(\frac{T}{m}\right).$$

which is better when $T \ll m$ (which is the only case in which the resulting bound beats the trivial bound anyway).

Others have also attempted to bound $\|A^m(I - A)\|$ where A is a product of projection: [104] also gave a bound. However, that bound is not good for our purposes as we require a bound that is uniform over all A that are products of T projections, whereas [104] focused on getting bounds for large m for each fixed A .

Prior work on related iterative methods

As mentioned earlier, our setting is related to the setting of Kaczmarz methods. There has been much work studying Kaczmarz methods, but it is common to bound not the final average loss but rather the distance to the least-norm solution. Furthermore, such bounds sometimes depend on the data distribution, for instance getting a bound that depends on a condition number which may be arbitrarily large[40, 87, 115].

Prior work has also calculated forgetting bounds under assumptions on the distribution of the data, like [76] who assumed a Gaussian distribution.

In contrast, our bounds depend only on the number of tasks and on the normalizations mentioned above.

Previous bounds on the numerical range of the product of T projections

Our method to get the improved bound uses, as an intermediate step, getting a better (in fact sharp) bound on the numerical range of any linear map expressible as the product of T projections.

This yields a bound on $\|A^m(I - A)\|$ via further results in [28] to be discussed later.

We are not the first to attempt to bound the numerical range of such a linear map: Lemma 5.1 in [28] also gives a bound. However, that bound is not sharp, and the same method would indeed give a bound on $\|A^m(I - A)\|$. However, the resulting bound on forgetting obtained by applying the same method is not good enough for our purposes: The

resulting bound on forgetting is of the form $O(f(T)/m)$ where $f(T)$ is exponential.

Another attempt to bound the numerical range of the product of T projections is given in [12], which gives a better bound than [28]. However, their bound on the numerical range depends on the relative orientation of the subspaces of projection (specifically, the Friedrich's number), and as we wish to get a uniform bound over all choices of projections this includes subspaces with bad Friedrich's number (close to 1), and it turns out this does not result in a better asymptotic bound on $\|A^m(I - A)\|$.

So the weaker bounds on such numerical ranges were good enough for the purposes of those papers³, but not for ours.

2.2 Upper bound

[41] showed that, if A denotes the linear map given by one cycle, letting $\Delta_m(\vec{u}) := \|A^m \vec{u}\|^2 - \|A^{m+1} \vec{u}\|^2$,

$$\sup_S F_S(mT) \leq \frac{T-1}{2} \cdot \max_{\vec{u}: \|\vec{u}\|=1} \Delta_m(\vec{u}) \leq (T-1) \|A^m(1 - A)\|.$$

[29] showed⁴ that, for all polynomials p ,

$$\|p(A)\| \leq (1 + \sqrt{2}) \sup_{z \in W(A)} |p(z)|.$$

In this paper, we determine $\bigcup_A W(A)$ where A ranges over all products of k complex⁵ orthogonal projections in any dimension.

³as far as I can tell, plugging in our improved numerical range bound does not affect the main results of the paper

⁴There are now several claimed proofs of Crouzeix's conjecture [62, 77], so we can in fact replace $1 + \sqrt{2}$ with 2, improving our bound by a constant factor.

⁵It will turn out that the union is the same for real projections and complex projections

Then for any product A of k real projections, $W(A)$ will be contained in this union, so the final bound is obtained by maximizing $|z^m(1-z)|$ over this union.

2.2.1 Approach

To get a bound on the numerical range, we first show that the numerical range can be expressed as the set of complex numbers of the form

$$P(\vec{v}_0, \dots, \vec{v}_{k-1}) := \langle \vec{v}_0, \vec{v}_1 \rangle \langle \vec{v}_1, \vec{v}_2 \rangle \dots \langle \vec{v}_{k-2}, \vec{v}_{k-1} \rangle \langle \vec{v}_{k-1}, \vec{v}_0 \rangle, \quad (2.1)$$

for complex unit vectors $\vec{v}_i \in \mathbf{C}^d$.

We then show that the set of such numbers is

$$re^{i\theta} : \cos(\pi/k)r^{-1/k} \geq \cos\left(\frac{\theta - \pi}{k}\right).$$

It turns out that the boundary of this set is known as a sinusoidal spiral.

It has an equivalent description as the image of the the convex hull of the k th roots of unity under the map $z \mapsto z^k$. Thus the boundary can be obtained by taking any of the edges of the k -gon bounded by the k -th roots of unity and raising it to the k th power.

To show that the range of P is as we claim, we first reduce to working in $\mathbf{C}^2$. Indeed, the input to P is a sequence of vectors, and if such vector $\vec{v}$ of them does not lie in the plane Q spanned by its neighbors, then by projecting $\vec{v}$ onto Q and rescaling the projection, we get a point in $\mathbf{C}$ in the range of P with the same argument and larger magnitude.

For now, we focus on getting the points on the outer boundary of the range of P , as that will be sufficient to get the forgetting bound.

So for any input to P in any dimension, one can find an input in $\mathbf{C}^2$ whose output has the same argument and at least as large a magnitude, so for the rest of this argument we may assume all input vectors are in $\mathbf{C}^2$.

Next, the map P is smooth as a map of real manifolds $\mathbf{C}^n \supset (S^3)^n \rightarrow \mathbf{C}$. Therefore any input mapping to a point on the boundary of the range of P has Jacobian of determinant zero. Geometrically, this means that all directional derivatives of P at such a point must be parallel or $\vec{0}$.

By writing this constraint algebraically, we solve for all the critical points, and calculating their images under P gives all critical values, and taking the critical values of maximal norm for each given argument yields the boundary.

Remark 2.5. As we solve for the critical points exactly, this allows us to describe all tuples $(\vec{v}_0, \dots, \vec{v}_{k-1}) \in (\mathbf{C}^2)^k$ whose image under P maps to the boundary. It turns out that there is a description of such tuples in terms of quaternionic roots of unity.

We believe the resulting bound on $\|A^m(I - A)\| = \|A^m - A^{m+1}\|$ may be independently useful:

Lemma 2.6. Let $A: \mathbf{C}^n \rightarrow \mathbf{C}^n$ be a linear map expressible as a product of k complex projections. Then

$$\|(I - A)A^m\| \leq (C + o_{k,m \rightarrow \infty}(1)) \frac{k}{m},$$

where C is an absolute constant which may be taken to be 0.4.

Combining this with the inequality of [41] immediately gives the main result, Theorem 2.4.

2.2.2 Notation for proof

$\mathbf{C}$ and $\mathbf{H}$ denote the complex numbers and the quaternions, respectively.

All vector spaces will be over $\mathbf{C}$ throughout the proof.

We adopt the following conventions:

- Complex inner products are conjugate linear in the first argument:

$$\langle \vec{v}, \vec{w} \rangle = \vec{v}^* \vec{w}$$

where v^* denotes the conjugate transpose of $\vec{v}$.

- The argument of a complex number is in $(-\pi, \pi]$.

The numerical range of a matrix $A \in \mathbf{C}^{d \times d}$ is defined as

$$W(A) = \{\vec{z}^* A \vec{z} \mid \vec{z} \in \mathbf{C}^d, \|\vec{z}\| = 1\}.$$

2.3 Bounding the range of P

In this section, we give a complete description of the range of P as defined in (2.1). Furthermore, we give a complete description of the inputs consisting of vectors in $\mathbf{C}^2$ that map to boundary points.

We first introduce some notation:

Notation 2.7. For $(a, b) \in \mathbf{C}^2$, let

$$(a, b)^\perp := (\bar{b}, -\bar{a}).$$

Direct calculation shows:

Proposition 2.8. For $\vec{v}, \vec{w} \in \mathbf{C}^2$, $\langle \vec{v}, \vec{w}^\perp \rangle = -\overline{\langle \vec{v}^\perp, \vec{w} \rangle}$.

We first treat P as a smooth map $\mathbf{R}^{2k} \rightarrow \mathbf{R}^2$ and give a lemma algebraically characterizing its critical points. We defer the proof to the appendix.

Lemma 2.9. Let $c \in \mathbf{C}$ be a critical point of P . Then there exist

$$(a_0, b_0) = (1, 0), (a_1, b_1), \dots, (a_{k-1}, b_{k-1}) \in \mathbf{C}^2$$

such that $P((a_m, b_m)_{m=0}^{k-1}) = c$. Furthermore, for any such preimage $(a_m, b_m)_{m=0}^{k-1}$, after possibly multiplying each vector (a_m, b_m) by a complex unit⁶, there exist $\alpha, \beta \in \mathbf{C}$ such that

1. $\langle (a_m, b_m), (a_{m+1}, b_{m+1}) \rangle = \alpha$ for all m
2. $\langle (a_m, b_m)^\perp, (a_{m+1}, b_{m+1}) \rangle = \beta \omega^{m-1}$ for all m

with indices treated mod k .

We now algebraically solve for the critical points of P to get the following theorem:

Theorem 2.10. Let $d \geq 2$. Let $P(\vec{v}_0, \dots, \vec{v}_{k-1}) = \langle \vec{v}_0, \vec{v}_1 \rangle \langle \vec{v}_1, \vec{v}_2 \rangle \dots \langle \vec{v}_{k-2}, \vec{v}_{k-1} \rangle \langle \vec{v}_{k-1}, \vec{v}_0 \rangle$ and let Γ_k be the range of P as each $\vec{v}_i$ ranges over unit vectors in $\mathbf{C}^d$. Then

$$\Gamma_k = \left\{ r e^{i\theta} \mid \cos\left(\frac{\pi}{k}\right) r^{-1/k} \geq \cos\left(\frac{\theta - \pi}{k}\right), r \geq 0, \theta \in [0, 2\pi] \right\},$$

which is a filled sinusoidal spiral.

Proof. As argued above, at least for the outer boundary, it suffices to consider all input vectors to be in $\mathbf{C}^2$. We give a direct proof in the appendix for this chapter that this also yields all points in the interior⁷.

The result is trivial for $k \leq 2$, so we will assume $k \geq 3$.

As P is a continuous function on a compact domain, its image is compact, hence bounded.

Thus the range of P is contained in the region bounded by all critical values of P .

⁶This makes sense as multiplying inputs to P by complex units does not affect the output of P

⁷There is also an easy proof via the interpretation as the union of all numerical ranges: The numerical range of a nontrivial (i.e., non-identity) product of projections always contains 0, as it is singular. So by convexity of the numerical range, any product of projections whose numerical range realizes a point on the boundary also realizes the line segment from that point to 0.

Fix a critical value c of P , and choose a_i, b_i as in Lemma 2.9. For every m , the orthogonal vectors (a_m, b_m) and $(a_m, b_m)^\perp$ form a basis of $\mathbf{C}^2$. Setting $\alpha = a_0$ and $\beta = b_0$, the relations in the lemma give the recurrence relation

$$\begin{aligned}
(a_1, b_1) &= \alpha(1, 0) + \beta(1, 0)^\perp \\
(a_2, b_2) &= \alpha(a_1, b_1) + \beta\omega(a_1, b_1)^\perp \\
(a_3, b_3) &= \alpha(a_2, b_2) + \beta\omega^2(a_2, b_2)^\perp \\
&\vdots \\
(a_{k-1}, b_{k-1}) &= \alpha(a_{k-2}, b_{k-2}) + \beta\omega^{k-1}(a_{k-2}, b_{k-2})^\perp \\
(1, 0) = (a_0, b_0) &= \alpha(a_{k-1}, b_{k-1}) + \beta\omega^k(a_{k-1}, b_{k-1})^\perp.
\end{aligned}$$

Identify $\mathbf{C}^2$ with $\mathbf{H}$ via $(a, b) \mapsto a + bj$. Thus $(a, b)^\perp \mapsto j(a + bj)$ and, for $\phi \in \mathbf{C}$, $\phi(a, b) \mapsto \phi(a + bj) = (a + bj)\bar{\phi}$. In particular, multiplication on the left by a complex number is unambiguous.

Our recurrence relation becomes

$$\begin{aligned}
a_1 + b_1j &= (\alpha + \beta j)(1 + 0j) \\
a_2 + b_2j &= (\alpha + \beta\omega j)(a_1 + b_1j) \\
a_3 + b_3j &= (\alpha + \beta\omega^2 j)(a_2 + b_2j) \\
&\vdots \\
a_{k-1} + b_{k-1}j &= (\alpha + \beta\omega^{k-2} j)(a_{k-2} + b_{k-2}j) \\
1 + 0j &= (\alpha + \beta\omega^k j)(a_{k-1} + b_{k-1}j).
\end{aligned}$$

So the vectors (a_m, b_m) must be the partial products

$$(a_m, b_m) = (\alpha + \beta\omega^{m-1}j)(\alpha + \beta\omega^{m-1}j) \dots (\alpha + \beta\omega^0j)$$

for $m = 0, \dots, k-1$ (defined as the empty product if $m = 0$), where α, β satisfy the equation

$$1 = (\alpha + \beta\omega^{k-1}j)(\alpha + \beta\omega^{k-2}j) \dots (\alpha + \beta\omega^0j).$$

We now show that any sequence of vectors $(a_0, b_0), \dots, (a_{k-1}, b_{k-1})$ satisfying the above system of equations can be obtained by slightly modifying a solution in the case $\omega = 1$ (possibly with a different value of k).

Let ζ_k be a (not necessarily primitive) complex k th root of unity. The sequence

$$(a_0, b_0), (a_1\zeta_k, b_1\zeta_k), (a_2\zeta_k^2, b_2\zeta_k^2), \dots, (a_{k-1}\zeta_k^{k-1}, b_{k-1}\zeta_k^{k-1}).$$

also satisfies the two conditions in Lemma 2.9, so we can run the same logic as above. This yields a another system of equations in the same form, but with different choices of α, β, ω . Specifically, the new values of α, β are ζ_k times the old values, and the new value of ω is ζ_k^2 times the old value, so by choosing ζ_k appropriately we may freely modify ω by ζ_k^2 for any k th root of unity ζ_k , or equivalently any power of $e^{2\pi i/k}$.

If k is odd, this is enough to show that any nonzero critical value can be obtained by $\omega = 1$, and furthermore that any critical point can be expressed as

$$(a_0, b_0), (a_1\zeta_k, b_1\zeta_k), (a_2\zeta_k^2, b_2\zeta_k^2), \dots, (a_{k-1}\zeta_k^{k-1}, b_{k-1}\zeta_k^{k-1})$$

for some k th root of unity ζ_k .

If k is even, let ζ_{2k} be a $2k$ th root of unity. Consider the sequence

$$(a_0, b_0), (a_1\zeta_{2k}, b_1\zeta_{2k}), (a_2\zeta_{2k}^2, b_2\zeta_{2k}^2), \dots, (a_{k-1}\zeta_{2k}^{k-1}, b_{k-1}\zeta_{2k}^{k-1}), \\ (a_0\zeta_{2k}^k, b_0\zeta_{2k}^k), (a_1\zeta_{2k}^{k+1}, b_1\zeta_{2k}^{k+1}), \dots, (a_{k-1}\zeta_{2k}^{2k-1}, b_{k-1}\zeta_{2k}^{2k-1}).$$

This satisfies the conditions in Lemma 2.9 for $2k$ in place of k . The corresponding values of α, β in the resulting system are the same as in the original multiplied by ζ_{2k} , and the resulting value of ω is the same as the original multiplied by ζ_{2k}^2 . As ζ_{2k}^2 can be chosen to be ω^{-1} , we can reduce to the case

$$1 = (\zeta_{2k}(\alpha + \beta\omega^{2k-1}j))(\zeta_{2k}(\alpha + \beta\omega^{2k-2}j)) \dots (\zeta_{2k}(\alpha + \beta\omega^0j))$$

where $\omega = 1$, or equivalently

$$1 = (\zeta_{2k}(\alpha + \beta j))^{2k}.$$

Remark 2.11. It is possible to write these manipulations using quaternions, which we demonstrate in the appendix.

For any integer m , the m th roots of unity in $\mathbf{H}$ are precisely those numbers whose real parts are the same as those of the real parts of the m th roots of unity in $\mathbf{C}$. So $(\alpha + \beta j)^m = 1$ if and only if α lies on one of the vertical diagonals of the m -gon whose vertices are the m th roots of unity (where the vertical diagonals are those with constant real part, including sides of the n -gon).

For the phase-shifted system, α consequently ranges over the vertical diagonals of the regular k -gon when k is odd, and over every other vertical diagonal of the regular $2k$ -gon when k is even; in each case the vertices are the appropriate roots of unity. Inverting the phase shift gives that, regardless of whether k is even or odd, the values of α in the solutions

to

$$1 = (\alpha + \beta\omega^{k-1}j)(\alpha + \beta\omega^{k-2}j) \dots (\alpha + \beta\omega^0j)$$

are the points on the diagonals (including the sides), of the regular k -gon whose vertices are the k th roots of unity.

As the inner product of consecutive vectors at such a sequence is α , the value of P at such a sequence is α^k . Therefore the critical values of f are then k th powers of these diagonals. Furthermore, when k is odd the critical points are sequences of the form $(a_m, b_m) = \phi_m(\alpha + \beta j)^m$ where $\alpha + \beta j$ is a quaternionic k th root of unity and ϕ_m are units in $\mathbf{C}$; When k is even, the critical points are sequences of the form $(a_m, b_m) = \phi_m(\alpha + \beta j)^m$ where $\alpha + \beta j$ are quaternionic $2k$ th roots of unity and ϕ_m are units in $\mathbf{C}$,

By the maximum principle for holomorphic maps, the outer boundary of Γ_k is obtained by taking the k th powers of the sides (where each side maps to the same curve). Using polar coordinates, the resulting curve is the sinusoidal spiral $\cos(\frac{\pi}{k})r^{-1/k} = \cos(\frac{\theta-\pi}{k})$. $\square$

The same proof shows that all other critical points other than those that P sends to 0 arise from the other diagonals of the polygons and their corresponding critical values are their k th powers, which are also sinusoidal spirals. A computation shows that a sequence of vectors that P sends to 0 is a critical point if and only if at least two of the inner products are 0.

2.4 Application to cyclic forgetting bounds

As an immediate application of Theorem 2.10, we can control the numerical range of a product of k orthogonal projections.

Corollary 2.12. Let $A = P_k P_{k-1} \dots P_1$ be a product of k orthogonal complex projections $\mathbf{C}^d \rightarrow \mathbf{C}^d$. Then $W(A) \subseteq \Gamma_{k+1}$.

Proof. We first observe a general fact. Let $P: \mathbf{C}^d \rightarrow \mathbf{C}^d$ be an orthogonal projection and let $\vec{z} \in \mathbf{C}^d$. Let $\vec{q} = \frac{P\vec{z}}{\|P\vec{z}\|}$. Then

$$P\vec{z} = \vec{q}\langle\vec{q}, \vec{z}\rangle = \vec{q}\vec{q}^*\vec{z}.$$

Now let $\vec{z}$ be an arbitrary unit vector and let

$$\vec{q}_i = \frac{P_i P_{i-1} \dots P_1 \vec{z}}{\|P_i P_{i-1} \dots P_1 \vec{z}\|}.$$

Iterating the fact stated above,

$$(\vec{q}_i \vec{q}_i^*) \dots (\vec{q}_1 \vec{q}_1^*) \vec{z} = P_i \dots P_1 \vec{z},$$

for all $i \in [k]$. Hence

$$\vec{z}^* A \vec{z} = \vec{z}^* (\vec{q}_k \vec{q}_k^*) (\vec{q}_{k-1} \vec{q}_{k-1}^*) \dots (\vec{q}_1 \vec{q}_1^*) \vec{z} = \langle \vec{z}, \vec{q}_k \rangle \langle \vec{q}_k, \vec{q}_{k-1} \rangle \dots \langle \vec{q}_2, \vec{q}_1 \rangle \langle \vec{q}_1, \vec{z} \rangle,$$

so $W(A) \subseteq \Gamma_{k+1}$. □

In the appendix for this chapter, we use our characterization of $P^{-1}(\partial\Gamma_{k+1})$ to characterize the pairs $(A, \vec{z})$ such that $\vec{z}^* A \vec{z} \in \partial\Gamma_{k+1}$.

We next convert the numerical range estimate into a uniform bound for $\|A^m(I - A)\|$. By [29], there is a constant $Q \leq 1 + \sqrt{2}$ such that every polynomial p satisfies

$$\|p(A)\| \leq Q \sup_{z \in W(A)} |p(z)|.$$

For a product $A: \mathbf{C}^n \rightarrow \mathbf{C}^n$ of k complex projections, we apply this inequality with the numerical range bound above and with $p(z) = z^m(1 - z)$. We show the calculation in the appendix.

Lemma 2.13.

$$\sup_{z \in \Gamma_k} |z^m(1-z)| \leq \frac{k}{m} \left(\frac{4}{e\pi^2} + o_{k,m \rightarrow \infty}(1) \right).$$

A more precise version of this bound in which $\frac{4}{e\pi^2}$ can be improved in terms of $\frac{k}{m}$ will be given in the appendix.

Combining this with Crouzeix's inequality gives Lemma 2.6, which we restate below more precisely, and hence Theorem 2.4.

Lemma 2.6. For any linear map $A: \mathbf{C}^n \rightarrow \mathbf{C}^n$ expressible as a product of k complex projections,

$$\|A^m(I - A)\| \leq \frac{k}{m} \left(\frac{4}{e\pi^2} + o_{k,m \rightarrow \infty}(1) \right) Q$$

where Q is the constant of Crouzeix's inequality as presented above.

2.4.1 Real projections

The projections arising in the forgetting problem are real, so one may ask whether all points in our numerical range bound can be realized using real projection.

The appendix for this chapter shows that they all can, in fact using only real orthogonal projections $\mathbf{R}^4 \rightarrow \mathbf{R}^4$. These projections also realize the induced asymptotic bound on $\|A^m - A^{m+1}\|$. There is nevertheless additional structure at the boundary. Any real or complex collection of orthogonal projections whose product has a numerical-range point on $\partial\Gamma_k$ admits an orthogonal decomposition $U \oplus V$ into subspaces invariant under every projection, with $\|A^m \vec{u}\| - \|A^{m+1} \vec{u}\|$ decaying rapidly on U . Therefore, any indecomposable such sequence must have quickly decaying forgetting.

2.5 Appendix for Cyclic Forgetting

2.5.1 A geometric interpretation of the forgetting bound in [41]

[41] gave an algebraic proof of the bound

$$F_S(mT) \leq \frac{T-1}{2} \Delta_m(A).$$

We give a geometric interpretation of the proof.

Recall that $F_S(mT)$ is at most the average squared distance from $\vec{w}_{mT}$ to $P_1, P_2, \dots, P_T$, and $\vec{w}_{mT}$ is on P_T . The past iterates $\vec{w}_{mT+1}, \vec{w}_{mT+2}, \dots, \vec{w}_{mT+(T-1)}$ lie on $P_1, \dots, P_{T-1}$, respectively, so the distance from $\vec{w}_{mT}$ to P_t is at most $\|\vec{w}_{mT} - \vec{w}_{mT+t}\|$. The point will be that if $\|\vec{w}_t\|^2$ does not decrease much from $t = mT$ to $t = (m+1)T$, then $\vec{w}_t$ cannot move much along its path from $t = mT$ to $t = (m+1)T$.

Let $d_t := \|\vec{w}_t - \vec{w}_{t+1}\|$. Then $\|\vec{w}_{mT} - \vec{w}_{mT+t}\| \leq d_{mT} + d_{mT+1} + \dots + d_{mT+t-1}$.

As $\vec{w}_{t+1}$ is a projection of $\vec{w}_t$ onto a subspace, $d_t^2 = \|\vec{w}_t\|^2 - \|\vec{w}_{t+1}\|^2$, so $\|\vec{w}_{mT}\|^2 - \|\vec{w}_{(m+1)T}\|^2 = d_{mT}^2 + d_{mT+1}^2 + \dots + d_{(m+1)T-1}^2$.

$$\frac{F_S(mT)}{\|\vec{w}_{mT} - \vec{w}_{(m+1)T}\|^2} \leq \frac{\frac{1}{T} \sum_{t=1}^{T-1} \left(\sum_{s=0}^t d_{mT+s} \right)^2}{\sum_{t=0}^{T-1} d_{mT+t}^2}.$$
 While we could now optimize this quadratic form precisely, we opt for a simpler bound, deferring a more precise approach to section 2.5.3

$$\left(\sum_{s=0}^t d_{mT+s} \right)^2 = \|(d_{mT+s})_{s=1}^t\|_{L^1}^2 \leq t \|(d_{mT+s})_{s=1}^t\|_{L^2}^2$$
 (with equality when all d_{mT+s} are equal).

Our bound becomes

$$\begin{aligned}
\frac{\frac{1}{T} \sum_{t=1}^{T-1} \left(\sum_{s=0}^t d_{mT+s} \right)^2}{\sum_{t=0}^{T-1} d_{mT+t}^2} &\leq \frac{\frac{1}{T} \sum_{t=1}^{T-1} t \sum_{s=0}^t d_{mT+s}^2}{\sum_{t=0}^{T-1} d_{mT+t}^2} \\
&= \frac{\frac{1}{T} \sum_{s=0}^T \sum_{t=s}^{T-1} t d_{mT+s}^2}{\sum_{t=0}^{T-1} d_{mT+t}^2} \\
&\leq \frac{\frac{1}{T} \sum_{s=0}^T \sum_{t=0}^{T-1} t d_{mT+s}^2}{\sum_{t=0}^{T-1} d_{mT+t}^2} \\
&= \frac{T-1}{2}.
\end{aligned}$$

2.5.2 A tighter normalization

Suppose we did not normalize $\max_t \|X_t\| = 1$. Then each loss would be scaled by $\|X_t\|$. The argument in the previous section applies directly with an additional factor of $\frac{1}{T} \sum_{t=1}^T \|X_t\|$, so we can replace the normalizing of $\max_t \|X_t\| = 1$ with $\sum_{t=1}^T \frac{1}{T} \|X_t\| = 1$ without affecting the results.

2.5.3 Optimizing the quadratic form more precisely

[41] observed that one could improve the bound on forgetting in terms of $\sup_{A, \vec{u}} \|A^{m-1} \vec{u}\|_2^2 - \|A^m \vec{u}\|_2^2$ by a factor of 2 (compared to the bound above in terms of $\|A^m \vec{u}\|_2^2 - \|A^{m+1} \vec{u}\|_2^2$ (with a different exponent) for the specific $A, \vec{u} = \vec{w}_0$ in our forgetting) by using that $\vec{w}_{mT}$ can be shown to be even closer to the latter half of the planes $P_{T-1}, P_{T-2}, \dots$ by considering its distance to $\vec{w}_{mT-1}, \vec{w}_{mT-2}, \dots$ and instead choosing A to be the update map shifted by half the iterates, and choosing $\vec{u}$ to be $\vec{w}_{\frac{T}{2}}$ (up to a floor or ceiling). Applying a similar analysis gives the factor of 2.

But it is possible to improve by more than a factor of 2 by computing the optimizer of the ratio of the quadratic forms: The relevant quantity to bound can now be expressed as $\frac{\vec{v}^T Q \vec{v}}{\vec{v}^T \vec{v}}$ for a block matrix Q split along the middle, so it's enough to optimize a single block which

is half the length of the full matrix. This is optimized by computing the largest eigenvalue of Q , which can be computed by looking at the characteristic polynomial of its inverse to gain another factor of 2 (asymptotically).

This analysis can be combined with removing the normalization of the X_t , in which case Q depends on $\|X_t\|$ but the same procedure would work as long as you can compute the eigenvalues of Q .

2.5.4 Proof of Lemma 2.9

We start with the following simple fact that we will use below.

Proposition 2.14. Let $x, y \in \mathbf{C}$ be such that $\{\gamma x + \bar{\gamma} y | \gamma \in \mathbf{C}\}$ consists of real multiples of some fixed complex number. Then $|x| = |y|$.

Proof. If $x + y = 0$ then we are done. Otherwise $\frac{ix - iy}{x + y} \in \mathbf{R}$. So

$$\frac{ix - iy}{x + y} = \frac{-i\bar{x} + i\bar{y}}{\bar{x} + \bar{y}},$$

which rearranges to $|x| = |y|$. □

Now prove Lemma 2.9.

Proof. For $\vec{z} \in \mathbf{C}^2$ let $\nabla_{m, \vec{z}} P$ denote the directional derivative of P as the m th coordinate is varied in the $\vec{z}$ direction. That is,

$$\nabla_{m, \vec{z}} P(\vec{v}_0, \dots, \vec{v}_{k-1}) := \lim_{t \rightarrow 0} \frac{1}{t} (P(\vec{v}_0, \dots, \vec{v}_m + t\vec{z}, \dots, \vec{v}_{k-1}) - P(\vec{v}_0, \dots, \vec{v}_{k-1})).$$

A direct computation shows

$$\nabla_{m, \vec{z}} P(\vec{v}_0, \dots, \vec{v}_{k-1}) = \left(\frac{\langle \vec{v}_{m-1}, \vec{z} \rangle}{\langle \vec{v}_{m-1}, \vec{v}_m \rangle} + \frac{\langle \vec{z}, \vec{v}_{m+1} \rangle}{\langle \vec{v}_m, \vec{v}_{m+1} \rangle} \right) P(\vec{v}_0, \dots, \vec{v}_{k-1}).$$

Here, and throughout the rest of the argument, indices are treated mod k . The only case where a denominator can be zero is if $P(\vec{v}_0, \dots, \vec{v}_{k-1}) = 0$. For $k \geq 3$, we will see that 0 does not lie on the outer boundary of Γ_k , so we may ignore this case.

Each $\vec{v}_m^\perp$, along with its scalar multiples, lies in the tangent space of the unit sphere at $\vec{v}_m$. So if $(\vec{v}_0, \dots, \vec{v}_{k-1})$ is a boundary point, then the directional derivatives $\nabla_{m, \gamma \vec{v}_m^\perp}$ must all be parallel as m varies over $\{1, \dots, k\}$ and γ varies over the unit circle. Hence for any fixed m ,

$$\left\{ \gamma \frac{\langle \vec{v}_{m-1}, \vec{v}_m^\perp \rangle}{\langle \vec{v}_{m-1}, \vec{v}_m \rangle} + \bar{\gamma} \frac{\langle \vec{v}_m^\perp, \vec{v}_{m+1} \rangle}{\langle \vec{v}_m, \vec{v}_{m+1} \rangle} \middle| \gamma \in \mathbf{C} \right\}$$

consists of real multiples of some fixed complex number. By Proposition 2.14 along with the Pythagorean theorem,

$$\frac{1 - |\langle \vec{v}_{m-1}, \vec{v}_m \rangle|^2}{|\langle \vec{v}_{m-1}, \vec{v}_m \rangle|^2} = \left| \frac{\langle \vec{v}_{m-1}, \vec{v}_m^\perp \rangle}{\langle \vec{v}_{m-1}, \vec{v}_m \rangle} \right|^2 = \left| \frac{\langle \vec{v}_m^\perp, \vec{v}_{m+1} \rangle}{\langle \vec{v}_m, \vec{v}_{m+1} \rangle} \right|^2 = \frac{1 - |\langle \vec{v}_m, \vec{v}_{m+1} \rangle|^2}{|\langle \vec{v}_m, \vec{v}_{m+1} \rangle|^2},$$

so $|\langle \vec{v}_{m-1}, \vec{v}_m \rangle| = |\langle \vec{v}_m, \vec{v}_{m+1} \rangle|$ for all m .

Replacing $\vec{v}_m$ by $\phi_m \vec{v}_m$, for any choices of $|\phi_m| = 1$, leaves P unchanged while multiplying the adjacent inner product $\langle \vec{v}_{m-1}, \vec{v}_m \rangle$ by $\overline{\phi_{m-1}} \phi_m$. We may choose these phases to make all resulting adjacent inner products agree. Indeed, after identifying unit complex numbers with their arguments modulo 2π , this is a linear statement: The vectors $\vec{e}_0 - \vec{e}_1, \vec{e}_1 - \vec{e}_2, \dots, \vec{e}_{k-1} - \vec{e}_0 \in \mathbf{R}^k$ span the subspace of vectors whose coordinates sum to zero, so by varying the ϕ_m , the list

$$\{\overline{\phi_{m-1}} \phi_m \langle \vec{v}_{m-1}, \vec{v}_m \rangle\}_{m=1}^k$$

can be set to be any list of complex numbers of norm α with product $P(\vec{v}_0, \dots, \vec{v}_{k-1})$. In particular, we can make this list constant.

Thus any boundary point is achieved by a sequence $(\vec{v}_0, \dots, \vec{v}_{k-1})$ with $\langle \vec{v}_m, \vec{v}_{m+1} \rangle = \alpha$ for a single $\alpha \in \mathbf{C}$, so for the remainder of this proof, we assume $(\vec{v}_0, \dots, \vec{v}_{k-1})$ is a critical

point with this property.

Set

$$\beta_m = \langle \vec{v}_m, \vec{v}_{m+1}^\perp \rangle = -\overline{\langle \vec{v}_m^\perp, \vec{v}_{m+1} \rangle},$$

so

$$\frac{\alpha}{P(\vec{v}_0, \dots, \vec{v}_{k-1})} \nabla_{m, \vec{v}_m^\perp} P(\vec{v}_0, \dots, \vec{v}_{k-1}) = (\beta_{m-1} - \overline{\beta_m}).$$

Again using that the derivatives are parallel at a critical point,

$$(\beta_0 - \overline{\beta_1}), (\beta_1 - \overline{\beta_2}), \dots, (\beta_{k-1} - \overline{\beta_0}) \in \{\lambda z \mid \lambda \in \mathbf{R}\} =: \ell_z$$

for some $z \in \mathbf{C}$ of unit norm. By the Pythagorean theorem, $|\beta_m|^2 = 1 - \alpha^2$, so all β_m 's have the same norm. This implies that either (i) β_{m-1} and $-\overline{\beta_m}$ are reflections about ℓ_z , or (ii) $\beta_{m-1} = \overline{\beta_m}$.

If (ii) holds for some m , then

$$\frac{\alpha}{P(\vec{v}_0, \dots, \vec{v}_{k-1})} \nabla_{m, i\vec{v}_m^\perp} P(\vec{v}_0, \dots, \vec{v}_{k-1}) = (i\beta_{m-1} + i\overline{\beta_m}) = 2i\beta_{m-1}.$$

So as long as the β 's are nonzero, $2i\beta_{m-1}$ must be a real multiple of z (because

$$\frac{\alpha}{P(\vec{v}_0, \dots, \vec{v}_{k-1})} \nabla_{m, \vec{v}_m^\perp} P(\vec{v}_0, \dots, \vec{v}_{k-1}) = (\beta_{m-1} - \overline{\beta_m})$$

all lie on ℓ_z). This means that z is perpendicular to β_{m-1} . When $\beta_{m-1} = \overline{\beta_m}$ this implies that $-\overline{\beta_m}$ is the reflection of β_{m-1} over ℓ_z . So condition (i) continues to hold in this case (and it holds trivially if the β 's are 0).

Thus we may assume (i) holds for all m . Then β_m is the image of β_{m-1} under a reflection about ℓ_z composed with a reflection about the imaginary axis. The composition of these reflections is a rotation about the origin, so there is some $\omega \in \mathbf{C}$ with $\|\omega\| = 1$ with $\beta_m = \omega^m \beta_0$ for all m , and also $\beta_0 = \beta_k = \omega^k \beta_0$. So either ω is a (not necessarily primitive) k th

root of unity or $\beta_0 = \dots = \beta_{k-1} = 0$. Either way, one can write $\beta_m = \omega^m \beta_0$ where ω is a k th root of unity. $\square$

2.5.5 Any (possibly inconsistent) cyclic sequence of T tasks converges to a cycle of length T

Recall that the direction of an affine subspace is the vector space spanned by any two vectors in that subspace.

The claim is equivalent to the following by setting w to be w_{i+kT} and cycling $P_1, \dots, P_T$ so that $P_{i \% T+1}$ comes first. (Or alternatively, by setting $w = w_0$ and using that the claim is preserved under applying any affine map.)

Proposition 2.15. Let $P_1, P_2, \dots, P_T$ be affine subspaces of $\mathbf{R}^d$, and let $\vec{w} \in \mathbf{R}^d$. Let $M : \mathbf{R}^d \rightarrow \mathbf{R}^d$ be the affine map given by the composition of orthogonal projections onto $P_1, P_2, \dots, P_T$ in that order. Then $\vec{w}, M\vec{w}, M^2\vec{w} \dots$ converges to a fixed point of M , either linearly or after a finite number of iterations.

Proof. We first show that the restriction of M to the affine hull A of $\vec{w}, M\vec{w}, M^2\vec{w} \dots$ is a strict contraction.

Indeed, the only projections that do not decrease $\|M^{k+1}\vec{w} - M^k\vec{w}\|$ are those onto affine subspaces whose direction contains $M^{k+1}\vec{w} - M^k\vec{w}$, but any sequence of such projections sends $M^k\vec{w}$ to points in the affine subspace orthogonal to A through $M^k\vec{w}$, so it is not possible that all the projections $P_i, P_{i+1}, \dots$ are parallel to this vector or else it would be impossible for their composition to send $M^k\vec{w}$ to $M^{k+1}\vec{w}$.

Next, the sequence $\vec{w}, M\vec{w}, M^2\vec{w}$ are the partial sums of $\vec{w} + (M\vec{w} - \vec{w}) + M(M\vec{w} - \vec{w}) + M^2(M\vec{w} - \vec{w}) + \dots$ and $\|M^k(M\vec{w} - \vec{w})\|$ is at most $\|(M\vec{w} - \vec{w})\|$ times the operator norm of the linear part of $M|_A$ to the power k . This operator norm is strictly less than 1, as it's a strict contraction, so the series converges linearly or faster.

To get the lower bound, apply the same argument to the eventual affine hull (the intersection of all affine hulls of subsequences obtained by ignoring a prefix), on which M must act invertibly. If it's only one point, then it must have converged after a finite number of steps; Otherwise, the invertibility of M implies that the smallest singular value is positive. $\square$

This convergence implies that the forgetting converges to some positive value along each subsequence $w_i, w_{i+T}, w_{i+2T}, \dots$, but it doesn't necessary converge for the whole sequence: Consider three lines bounding a right triangle. The cycle will include both the right-angle vertex and a point on the hypotenuse, and the forgetting at the point on the hypotenuse is strictly larger than the forgetting at the vertex.

Proposition 2.16. Using the notation of the previous proposition, let $w_{*,1}$ and $w_{*,2}$ be fixed points of M . Then $w_{*,2} - w_{*,1}$ is contained in the direction of each P_i . In other words, letting D be the intersections of the directions for the P_i , the fixed points are $w_{*,1} + D$.

In particular, if $P := \bigcap_i P_i \neq \emptyset$, then all fixed points of M are in P .

Proof. $\|M(w_{*,2} - w_{*,1})\| = \|w_{*,2} - w_{*,1}\|$, but all projections that do not decrease $\|w_{*,2} - w_{*,1}\|$ contain $w_{*,2} - w_{*,1}$ in their direction.

In particular, if $P \neq \emptyset$, then any point $w_* \in P$ is a fixed point, so all fixed points of P can be expressed as $w_* + d$ where d is in the direction of all the P_i , so $w_* + d \in P$. $\square$

2.5.6 Proof of Lemma 2.13

The following lemma gives a bound that depends on $\frac{k}{m}$, from which Lemma 2.13 will follow.

Lemma 2.17. Define $S : [0, \frac{1}{2}] \rightarrow [\frac{1}{4}, \infty)$ by $S(t) := \frac{\cot(\pi t)}{2\pi(1-2t)}$ (which is strictly monotonically decreasing). Then

$$\sup_{z \in \Gamma_k} |z^m(1-z)| \leq \begin{cases} e^{-\frac{m}{k}((1-S^{-1}(\frac{m}{k}))S^{-1}(\frac{m}{k}))(2\pi^2)} 2 \sin(\pi S^{-1}(\frac{k}{m})) + o_{k \rightarrow \infty}(1) & k \leq 4m \\ 2e^{-\frac{m}{k}\frac{\pi^2}{2}} + o_{k \rightarrow \infty}(1) & k > 4m \end{cases}.$$

where the little-o terms are uniform in m .

Proof. We parameterize the boundary of Γ_{k+1} by $((1-t) + te^{\frac{2\pi i}{k}})^k$ as t ranges from 0 to 1; By the maximum principle, $\sup_{z \in \Gamma_{k+1}} |(1-z)z^m|$ is attained for some value of t .

The first factor $1-z$ becomes

$$\begin{aligned} \lim_{k \rightarrow \infty} \left| 1 - ((1-t) + te^{\frac{2\pi i}{k}})^k \right| &= \left| 1 - \lim_{k \rightarrow \infty} ((1-t) + te^{\frac{2\pi i}{k}})^k \right| \\ &= \left| 1 - \lim_{k \rightarrow \infty} (e^{\frac{2\pi i t}{k}})^k \right| \\ &= |1 - e^{2\pi i t}| \\ &= 2|\sin(\pi t)| \end{aligned}$$

uniformly in t .

The second factor z^m becomes

$$\begin{aligned}
& \left| \left((1-t) + te^{\frac{2\pi i}{k}} \right)^k \right|^m \\
&= \left| (1-t) + te^{\frac{2\pi i}{k}} \right|^{km} \\
&= \left(\left((1-t) + t \cos\left(\frac{2\pi}{k}\right) \right)^2 + \left(t \sin\left(\frac{2\pi}{k}\right) \right)^2 \right)^{\frac{km}{2}} \\
&= \left((1-t)^2 + 2(1-t)t \cos\left(\frac{2\pi}{k}\right) + t^2 \right)^{\frac{km}{2}} \\
&= \left(((1-t) + t)^2 - 2(1-t)t(1 - \cos\left(\frac{2\pi}{k}\right)) \right)^{\frac{km}{2}} \\
&= \left(1 - 2(1-t)t(1 - \cos\left(\frac{2\pi}{k}\right)) \right)^{\frac{km}{2}} \\
&= \left(1 - 2(1-t)t \left(\frac{\left(\frac{2\pi}{k}\right)^2}{2} + O(k^{-4}) \right) \right)^{\frac{km}{2}} \\
&= \left(1 - \frac{1}{\left(\frac{k^2}{2(1-t)t(2\pi^2 + O(k^{-2}))} \right)} \right)^{\frac{km}{2}} \\
&= \left(\left(1 - \frac{1}{\left(\frac{k^2}{2(1-t)t(2\pi^2 + O(k^{-2}))} \right)} \right)^{\frac{k^2}{2(1-t)t(2\pi^2 + O(k^{-2}))}} \right)^{\frac{m}{2k} (2(1-t)t(2\pi^2 + O(k^{-2})))}
\end{aligned}$$

where the $O(k^{-2})$ term is $-k^2(\cos(\frac{2\pi}{k}) - 1 + \frac{(\frac{2\pi}{k})^2}{2})$ in all occurrences.

Letting $\alpha := \frac{m}{k}$, this is

$$(e^{-1} - o_{k \rightarrow \infty}(1))^{\alpha((1-t)t(2\pi^2 + O(k^{-2})))}.$$

Furthermore, $\left(1 - 2(1-t)t(1 - \cos(\frac{2\pi}{k})) \right)^{\frac{k}{2}}$ is increasing in k for k sufficient large k .

(indeed, the derivative of its log with respect to k is

$$\frac{2\pi m \cdot (1-t)t \sin\left(\frac{2\pi}{k}\right)}{\left(1 - 2(1-t)t \cdot \left(1 - \cos\left(\frac{2\pi}{k}\right) \right) \right) k} + \frac{m \ln\left(1 - 2(1-t)t \cdot \left(1 - \cos\left(\frac{2\pi}{k}\right) \right) \right)}{2}$$

which is positive for large k (the first addend is positive and the second is negative for large

k ; The limit of the ratio is -2 uniformly in $t \in [0, \frac{1}{2}]$ as $k \rightarrow \infty$ so the first term is larger than the second.)) So the limit is from below. That is, the second factor is

$$\left(e^{-((1-t)t)(2\pi^2)} - o_{k \rightarrow \infty}(1)\right)^\alpha$$

where the little-o is positive.

Putting the two factors together, letting $q(z) = z^{mk}(1 - z^k)$,

$$\begin{aligned} |q(1 - t + te^{\frac{2\pi i}{k}})| &= \left(e^{-((1-t)t)(2\pi^2)} - o_{k \rightarrow \infty}(1)\right)^\alpha 2|\sin(\pi t)| \\ &= \left(e^{-((1-t)t)(2\pi^2)} - o_{k \rightarrow \infty}(1)\right)^\alpha 2\sin(\pi t) \end{aligned}$$

uniformly on $t \in [0, 1]$. As the little-o is positive, the limit as $k \rightarrow \infty$ is also uniform in m .

For any fixed α , the maximum is attained either at an endpoint or where the derivative with respect to t is 0. By symmetry about $\frac{1}{2}$, it suffices to bound this for $t \in [0, \frac{1}{2}]$.

The derivative is

$$2\pi e^{-\alpha((1-t)t)(2\pi^2)}(\cos(\pi t) - 2\pi\alpha(1 - 2t)\sin(\pi t)),$$

which, for $t \in (0, \frac{1}{2})$, has the same sign as

$$\frac{\cot(\pi t)}{2\pi(1 - 2t)} - \alpha.$$

As $\frac{\cot(\pi t)}{2\pi(1 - 2t)}$ is decreasing in t (its derivative has the same sign as $2\pi t + \sin(2\pi t) - \pi$, so letting $s = 2\pi t$, this is $s + \sin(s) - \pi$ which is increasing in s), $e^{-\alpha((1-t)t)(2\pi^2)} 2\sin(\pi t)$ is increasing in t from 0 until $\frac{\cot(\pi t)}{2\pi(1 - 2t)} = \alpha$, and after this it is decreasing in t . In particular, if there is no t such that $\frac{\cot(\pi t)}{2\pi(1 - 2t)} = \alpha$ (or equivalently, $\alpha < \min_{[0, \frac{1}{2}]} \frac{\cot(\pi t)}{2\pi(1 - 2t)} = \frac{1}{4}$), then $\operatorname{argmax}_t e^{-\alpha((1-t)t)(2\pi^2)} 2\sin(\pi t) = \frac{1}{2}$. $\square$

We now prove Lemma 2.13, that

$$\sup_{z \in \Gamma_k} |z^m(1-z)| \leq \frac{k}{m} \left(\frac{4}{e\pi^2} + o_{k,m \rightarrow \infty}(1) \right).$$

Proof. If $k \leq 4m$ then the maximum of the limit as $k, m \rightarrow \infty$ with α fixed is attained at a point where $\frac{m}{k} = \alpha = \frac{\cot(\pi t)}{2\pi(1-2t)}$. Plugging this in gives

$$\lim_{k \rightarrow \infty, m = \alpha k} |q(1-t + te^{\frac{2\pi i}{k}})| \frac{m}{k} = e^{-\frac{\cot(\pi t)}{2\pi(1-2t)}((1-t)t)(2\pi^2)} 2 \sin(\pi t) \frac{\cot(\pi t)}{2\pi(1-2t)},$$

which is bounded by

$$\frac{1}{e\pi} \leq e^{-\frac{\cot(\pi t)}{2\pi(1-2t)}((1-t)t)(2\pi^2)} 2 \sin(\pi t) \frac{\cot(\pi t)}{2\pi(1-2t)} \leq \frac{1}{2e^{\frac{\pi^2}{8}}},$$

where equality is attained for the first inequality at $t = 0$ (which corresponds to $\alpha = \infty$) and for the second at $t = \frac{1}{2}$ (which corresponds to $\alpha = \frac{1}{4}$).

If $m > \frac{k}{4}$ then the maximum is attained at $t = \frac{1}{2}$, giving the value $2e^{-\frac{m}{k}\frac{\pi^2}{2}}$. Since

$$2\frac{m}{k}e^{-\frac{m}{k}\frac{\pi^2}{2}} \leq \frac{4}{e\pi^2},$$

the claim follows. □

2.5.7 Reducing to $\omega = 1$ via quaternions

We rewrite the arguments that reduce our problem to solving the equation when $\omega = 1$ (possibly replacing k with $2k$) purely in terms of quaternions.

For k odd, then for any solution to

$$1 = (\alpha + \beta\omega^{k-1}j)(\alpha + \beta\omega^{k-2}j) \dots (\alpha + \beta\omega^0j),$$

it holds that

$$1 = ((\alpha\zeta_k) + (\beta\zeta_k)(\zeta_k^{-2}\omega)^k j)((\alpha\zeta_k) + (\beta\zeta_k)(\zeta_k^{-2}\omega)^{k-1} j) \dots ((\alpha\zeta_k) + (\beta\zeta_k)(\zeta_k^{-2}\omega)^0 j).$$

For k even, the original proof could be translated directly into quaternions, but we find the following slightly modified version cleaner: For any solution to

$$1 = (\alpha + \beta\omega^{k-1}j)(\alpha + \beta\omega^{k-2}j) \dots (\alpha + \beta\omega^0j),$$

it holds that

$$\begin{aligned} ((\alpha\zeta_{2k}) + (\beta\zeta_{2k})(\zeta_{2k}^{-2}\omega)^k j)((\alpha\zeta_{2k}) + (\beta\zeta_{2k})(\zeta_{2k}^{-2}\omega)^{k-1} j) \dots ((\alpha\zeta_{2k}) + (\beta\zeta_{2k})(\zeta_{2k}^{-2}\omega)^0 j) \\ = \zeta_{2k}^k = \pm 1. \end{aligned}$$

Therefore the square of the left hand side is 1.

The whole problem has a simple expression in terms of quaternions: letting $\mathfrak{C}(a + bi + cj + dk) = a + bi$ be the complex part and $\mathfrak{H}(a + bi + cj + dk) = c + di$ be the “quaternionic part”, the desired range is the same as the range of $\prod_{m=0}^{k-1} \mathfrak{C}p_m$ subject to the constraint $\prod_{m=0}^{k-1} p_m = 1$. (An equivalent definition is $\mathfrak{C}(p) = \frac{p+ip^i}{2}$.)

Indeed, the inner product $\langle p, q \rangle$ where $p, q \in \mathbf{H}$ is $\mathfrak{C}q^{-1}p$, so letting our sequence of vectors be $p_0, p_0p_1, p_0p_1p_2, \dots, p_0p_1 \dots p_k$ for $p_m \in \mathbf{H}$ gives the claim.

The constraint that all inner products are equal becomes the constraint that $\mathfrak{C}p_m$ is the same for all m .

2.5.8 The range of P includes the interior

To directly show that the range of $P : (\mathbf{C}^2)^k \rightarrow \mathbf{C}$ includes the interior, we inductively (in k) give a geometric description of the range. For clarity, we let P_k denote the version of P

with domain $(\mathbf{C}^2)^k$.

The range of P_1 is $\{1\}$.

For any $k \in \mathbf{Z}_{\geq 0}$, any point in the range of P_{k+1} can be obtained by picking a number $p \in \Gamma_k$, taking a sequence $(a_0, b_0), (a_1, b_1), \dots, (a_{k-1}, b_{k-1}) \in \mathbf{C}^2$ with

$$P_k((a_0, b_0), (a_1, b_1), \dots, (a_{k-1}, b_{k-1})) = p,$$

and adding a point (a_k, b_k) to the end of the sequence.

Fix any sequence $(a_0, b_0), (a_1, b_1), \dots, (a_{k-1}, b_{k-1}) \in \mathbf{C}^2$. The set of possible ratios $\frac{P_{k+1}((a_m, b_m)_{m=0}^k)}{P_k((a_m, b_m)_{m=0}^{k-1})} = \frac{\langle (a_{k-1}, b_{k-1}), (a_k, b_k) \rangle \langle (a_k, b_k), (a_0, b_0) \rangle}{\langle (a_{k-1}, b_{k-1}), (a_0, b_0) \rangle}$ depends only on $\langle (a_0, b_0), (a_{k-1}, b_{k-1}) \rangle$.

This range is the union of circles centered at each point in $[0, 1]$, where the radii vary like an ellipse, as a rescaled version of $R(x) = l\sqrt{1-x^2}$, and the center gives the new squared inner product with (a_0, b_0) .

By using a unitary transformation sending (a_2, b_2) to $(1, 0)$, every such point can be expressed in 2 ways (up to multiplicity). These two coordinates form the new squared inner product magnitudes.

To show that this set is contractible, we show that its intersection with any line with fixed real part is contractible and show that applying the contraction to each intersection gives a line segment. More generally, we use the following lemma and corollary:

Lemma 2.18. Let $\{f_i\}_{i \in I_{\leq 0}}, \{f_j\}_{j \in I_{\geq 0}}$ be families of (not necessarily continuous) partial functions $\mathbf{R}^n \rightarrow \mathbf{R}$ where $f_i \leq 0, f_j \geq 0$. Assume these families are connected under the norm $\|f - g\| = \sup_{x \in \mathbf{R}^n} |f_0(x) - g_0(x)|$ where h_0 denotes the extension of the partial function h to all of $\mathbf{R}^n$ by 0.

Let $D_{\leq 0} = \bigcup_{i \in I_{\leq 0}} D(f_i), D_{\geq 0} = \bigcup_{j \in I_{\geq 0}} D(f_j)$ where $D(f)$ denotes the domain of definition of f , and let $D = D_{\leq 0} \cup D_{\geq 0}$.

Define $f_{\leq 0, \sup} : D_{\leq 0} \rightarrow \mathbf{R}$ by $f_{\leq 0, \sup}(x) := \sup_{\substack{i \in I_{\leq 0} \\ x \in D(f_i)}} f(x)$ and $f_{\geq 0, \inf} : D'_{\geq 0} \rightarrow \mathbf{R}$ by $f_{\geq 0, \inf}(x) := \inf_{\substack{j \in I_{\geq 0} \\ x \in D(f_j)}} f(x)$.

Then the region bounded (including extrema for each x) by $\Gamma(f_{\leq 0, \sup}) \cup \Gamma(f_{\geq 0, \inf}(x))$ where $\Gamma(f)$ denotes the graph of f in $\mathbf{R}^{n+1}$ (which may have multiple components) can be expressed as

$$D \times \{0\} \cup \left(\bigcap_{i \in I_{\leq 0}} \{(x, y) : x \in D(f_i), f_i(x) \leq y \leq 0\} \right) \cup \left(\bigcap_{j \in I_{\geq 0}} \{(x, y) : x \in D(f_j), 0 \leq y \leq f_j(x)\} \right).$$

$$\text{Let } X = \left(\bigcup_{i \in I_{\leq 0}} \Gamma(f_i) \right) \cup \left(\bigcup_{j \in I_{\geq 0}} \Gamma(f_j) \right).$$

Let $\pi : \mathbf{R}^{n+1} \rightarrow \mathbf{R}^n$ denote projection away from the last coordinate. If $f_{\leq 0, \sup}$ and $f_{\geq 0, \inf}(x)$ are continuous and $\pi(\overline{X_{\leq 0}}) \subseteq D_{\leq 0}$, $\pi(\overline{X_{\geq 0}}) \subseteq D_{\geq 0}$, then $\overline{X}$ deformation retracts to

$$G := \left(\bigcup_{x \in D_{\leq 0}} \sup\{f_i(x) : i \in I_{\leq 0}\} \right) \cup \left(\bigcup_{x \in D_{\geq 0}} \inf\{f_j(x) : j \in I_{\geq 0}\} \right).$$

A deformation retract is given by linearly decreasing the magnitude of the last coordinate.

Proof. The first claim factors over $\mathbf{R}^n$, so it suffices to prove this claim when D is a single point, in which case it is trivial.

For the second claim, by the connectedness of each family, their images on a single value of x are connected with constant sign, so the proposed deformation retract is well-defined. By the continuity of $f_{\leq 0, \sup}$ and $f_{\geq 0, \inf}$, the proposed deformation retract is continuous. $\square$

We leave finding the correct generalization when $f_{\leq 0, \sup}$ and $f_{\geq 0, \inf}(x)$ are discontinuous to the interested reader.

Corollary 2.19. Let $D \subset \mathbf{R}^n \subset \mathbf{R}^{n+1}$ be compact and connected, where the embedding sends the last coordinate to 0. Let $R : D \rightarrow \mathbf{R}_{\geq 0}$ be continuous.

Let $X = \bigcup_{x \in D} S_{R(x)}(x)$ where $S_r(x)$ denotes the sphere of radius r centered at x .

Letting $P_{n+1} : \mathbf{R}^{n+1} \rightarrow \mathbf{R}^n$ be the projection sending the last coordinate to 0, define

$G_{\min} : P_{n+1}(X) \rightarrow \mathbf{R}_{\geq 0}$ by $G_{\min}(x) := \min\{y \in \mathbf{R}_{\geq 0} : (x, y) \in X\}$, where the minimum exists because D is closed. Similarly define $G_{\max} : P_{n+1}(X) \rightarrow \mathbf{R}_{\geq 0}$ be defined by $G_{\max}(x) := \max\{y \in \mathbf{R}_{\geq 0} : (x, y) \in X\}$. Then:

1. $X = \bigcup_{x \in P_{n+1}(X)} \{(x, y) : G_{\min}(x) \leq |y| \leq G_{\max}(x)\}$.
2. X deformation retracts to $\bigcup_{x \in P_{n+1}(X)} \{(x, y) : G_{\min}(x) = |y|\}$.
3. The region bounded by this surface, $\bigcup_{x \in P_{n+1}(X)} \{(x, y) : G_{\min}(x) < |y|\}$, is

$$\bigcap_{x \in D} B_{R(x)}(x).$$

In particular, if $R(x)$ is ever 0, then X deformation retracts to $P_{n+1}(X)$.

In this case, if D is convex, then X is contractible.

Proof. Because D is closed, $X = \overline{X}$. Then the lemma gives the three enumerated items.

For what remains, it suffices to find a deformation retract from X to D . It is given by sending every point in $X \setminus D$ in the direction towards the nearest point in D , which is unique because D is convex. $\square$

One can generalize to higher dimensional spheres (in which case the dimension of the codomain also increases) by working on each copy of $\mathbf{R}^{n+1}$ containing $\mathbf{R}^n$ independently. A similar generalization applies to the lemma.

In our case, we can find the range by doing a calculation on each vertical line. Alternatively, if the radius at t is given by $R(t)$ a (not necessarily strict) superset of the boundary is the union of the circles at each endpoint plus the curve parameterized by $(t, 0) + R(t)(-R'(t), \pm\sqrt{1 - R'(t)^2})$ where t ranges over the line segment (this curve parameterizes, for each t in the interior of the interval where $|R'(t)| < 1$, the unique point accessible from that point and not any nearby points. If $|R'(t)| > 1$, then no such points exist.).

This range forms an ellipse $\frac{x^2}{l^2+1} + \frac{y^2}{l^2} = 1$ with foci at $(0,0)$ and $(1,0)$ by working one vertical line at a time.

The condition that all consecutive inner products are equal corresponds to the condition that the extremal points all come from the same point twice in each step of the geometric construction.

One gets another constraint because any two ellipses constructed in the above way passing through a specified point have different directions of tangency at that point, so the inner products $\langle (a_m, b_m), (a_{m+2}, b_{m+2}) \rangle$ must also be equal in magnitude.

2.5.9 Optimizing sequences for P are coplanar

Proposition 2.20. Any sequence of unit vectors $v_0, v_1, \dots, v_{k-1} \in \mathbf{C}^n$ with the property that $P(v_0, v_1, \dots, v_{k-1}) \in \partial\Gamma_k$ must be coplanar (i.e., lie on a complex plane).

Proof. We may assume $k \geq 3$, because any 2 vectors are coplanar.

We prove the contrapositive: If $v_0, v_1, v_2, \dots, v_{k-1}$ are not coplanar, then we will show $P(v_0, v_1, v_2, \dots, v_{k-1})$ is in the interior of Γ_k .

As Γ_k is radial and $\partial\Gamma_k$ is continuous (as a function of the complex argument), it suffices to show that there exists a sequence of k vectors whose image under P has the same argument but a larger magnitude.

As 0 is in the interior of Γ_k , we may assume none of the v_i are 0.

Assume without loss of generality that v_0, v_1, v_2 are not coplanar. Let v'_1 denote the projection of v_1 onto the plane spanned by v_0, v_2 . Then

$$\frac{P(v_0, \frac{v'_1}{\|v'_1\|}, v_2, \dots, v_{k-1})}{P(v_0, v_1, v_2, \dots, v_{k-1})} = \frac{1}{\|v'_1\|^2} \in \mathbf{R}_{>1}.$$

□

2.5.10 Products of projections with numerical range intersecting

$$\partial\Gamma_k$$

As shown in the proof of Theorem 2.10, any sequence of unit vectors $v_0, v_1, \dots, v_{k-1} \in \mathbf{H}$ realizing $P(v_0, \dots, v_{k-1}) \in \partial\Gamma_k$ must be obtainable from a sequence $v_0, v_0u, v_0u^2, \dots, v_0u^{k-1}$ (where u is a quaternionic k th root of unity) by multiplying each vector by a complex unit.

The interpretation in the sense of projections is that, if v_0 realizes $\partial\Gamma_k$, then the sequence of projections must send v_0 to the sequence of vectors formed by projecting onto $v_0u, v_0u^2, \dots$ in that order (multiplying by a complex unit does not change the line we project onto at each step). Up to a unitary transformation, we may take $v_0 = 1$.

Writing $u = a + bi$ with $a, b \in \mathbf{C}$, by direct computation, the sequence of projections is

$$1, au, a^2u^2, \dots$$

Combining this with the result from the previous section that any sequence of vectors optimizing P must be coplanar, we get that if $A = P_k P_{k-1} \dots P_1$ is a product of k projections on $\mathbf{C}^d$ with $W(A) \cap \Gamma_{k+1} \neq \emptyset$, then for any vector $v_0 \in \mathbf{C}^d$ with $v_0^* A v_0 \in \partial\Gamma_k$, the vectors $P_i P_{i-1} \dots P_1 v_0$ must all be coplanar, and furthermore there must exist a unitary transformation $\mathbf{H} \rightarrow \mathbf{C}^d$ such that the sequence of projections is the image of $1, \bar{a}u, \bar{a}^2u^2, \dots$

2.5.11 Real projections

We can directly show that any product A of k real projections whose numerical range contains a point in $\partial\Gamma_k$ must be decomposable into a direct sum $U \oplus V$ of subspaces, invariant under each projection, such that $\|A^m \vec{u}\|^2 - \|A^{m+1} \vec{u}\|^2$ is small for all $\vec{u}$.

Proposition 2.21. If $P_k, \dots, P_1: \mathbf{R}^n \rightarrow \mathbf{R}^n$ are orthogonal projections satisfying $\gamma \in W(P_k \dots P_1)$ for some $\gamma \in \partial\Gamma_{k+1}$, then there is an ≤ 4 -dimensional subspace $V \subseteq \mathbf{R}^n$ invariant under each P_i such that $\gamma \in W(P_k|_V \dots P_1|_V)$.

Proof. As we show in the appendix for this chapter, if $P_k, \dots, P_1 : \mathbf{C}^n \rightarrow \mathbf{C}^n$ are complex projections and $v \in \mathbf{C}^n$ such that $\gamma = v^* P_k \dots P_1 v \in \partial\Gamma_k$, then $v, P_1 v, \dots, P_k \dots P_1 v$ must lie in a complex plane. Combining this with the above equality case gives that there must exist an isometry of complex vector spaces $f : \mathbf{H} \rightarrow \mathbf{C}^n$ such that the action of the projections on the image corresponds to an equality case.

In particular, if $P_1, \dots, P_k$ are real projections, then the span of the real parts of the image of f is invariant under all P_t , is ≤ 4 -dimensional, and the numerical range of the restriction to this span also contains γ . $\square$

To get the orthogonal decomposition, the orthogonal complement of any subspace invariant under all projections is invariant under all projections. The reason is that this is true for each projection individually (i.e., for any orthogonal projection, the orthogonal complement of any invariant subspace is invariant).

Taking the invariant subspace from the proposition, we get that there is an invariant subspace where the projections act as described in the previous section. But the norms of these vectors decay geometrically, and therefore cannot do asymptotically better than the lower bound given by [41], and furthermore any collection of projections that does better can be done without having numerical range of the product intersect $\partial\Gamma_k$ by removing all orthogonal summands of this form.

2.5.12 Existence of real realizations

Despite the previous section, one may independently wonder whether the bound on the numerical range can be improved by restricting to real projections, thus improving our forgetting bound. The answer is that it cannot: Real projections can have product with a numerical range including any point of $\partial\Gamma_k$.

This is not immediate from our complex construction: Though any product of complex projections $\mathbf{C}^d \rightarrow \mathbf{C}^d$ can be interpreted as a product of real projections $\mathbf{R}^{2d} \rightarrow \mathbf{R}^{2d}$, the

numerical ranges of these two interpretations are different in general.

We will show that, for any unit quaternion $u = \alpha + \beta j \in \mathbf{H}$ (with $\alpha, \beta \in \mathbf{C}$), there exists a sequence of vectors $\mathbf{u}_{n,\Re}, \mathbf{u}_{n,\Im} \in \mathbf{C}^4$, such that:

1. The unitary map of complex vector spaces $\Phi : \mathbf{H} \rightarrow \mathbf{C}^4$ sending $u^0 = 1 \mapsto \mathbf{u}_{0,\Re} + i\mathbf{u}_{0,\Im}$ and $u^1 \mapsto \mathbf{u}_{1,\Re} + i\mathbf{u}_{1,\Im}$ sends $u^n \mapsto \mathbf{u}_{n,\Re} + i\mathbf{u}_{n,\Im}$. (If $\beta = 0$ then Φ is not determined by the $\Phi(u^0)$ and $\Phi(u^1)$, but $\Phi(u^n)$ always is.)
2. The real projection onto $\text{Span}(\mathbf{u}_{n,\Re}, \mathbf{u}_{n,\Im})$ (complexified to a map $\mathbf{C}^4 \rightarrow \mathbf{C}^4$) sends $\Phi(\bar{\alpha}^{n-1}u^{n-1})$ to $\Phi(\bar{\alpha}^n u^n)$ (equivalently, it sends the real (resp. imaginary) part to the real (resp. imaginary) part).

The second condition is equivalent to saying that the real projection onto $\text{Span}(\mathbf{u}_{n,\Re}, \mathbf{u}_{n,\Im})$ sends $\Phi(u^{n-1})$ to $\Phi(\bar{\alpha}u^n)$.

Let

$$\mathbf{u}_{0,\Re} + i\mathbf{u}_{0,\Im} = \frac{1}{\sqrt{2}} \left(\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + i \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \right)$$

and

$$\bar{\alpha}(\mathbf{u}_{1,\Re} + i\mathbf{u}_{1,\Im}) = \frac{1}{\sqrt{2}} \left(\begin{pmatrix} |\alpha|^2 \\ 0 \\ \sqrt{|\alpha|^2 - |\alpha|^4} \\ 0 \end{pmatrix} + i \begin{pmatrix} 0 \\ |\alpha|^2 \\ 0 \\ \sqrt{|\alpha|^2 - |\alpha|^4} \end{pmatrix} \right).$$

As all powers of u are real linear combinations of 1 and u (via the recurrence relation $u^2 = -1 + (2\Re u)u$, which holds for any unit quaternion), the first property above determines $\mathbf{u}_{n,\Re}, \mathbf{u}_{n,\Im}$.

We next check that there exists a unitary map Φ is unitary. Indeed, all we need to check

is that

$$\alpha = \langle \mathbf{u}_{0,\Re} + i\mathbf{u}_{0,\Im}, \mathbf{u}_{1,\Re} + i\mathbf{u}_{1,\Im} \rangle,$$

or equivalently

$$|\alpha|^2 = \bar{\alpha}\alpha = \langle \mathbf{u}_{0,\Re} + i\mathbf{u}_{0,\Im}, \bar{\alpha}(\mathbf{u}_{1,\Re} + i\mathbf{u}_{1,\Im}) \rangle$$

which is true.

Finally, we check the second property. This is invariant under (real) rotations.

Definition 2.22. Two unit vectors $\mathbf{1}, \mathbf{u} \in \mathbf{C}^4$ are **compatible** with respect to $\alpha \in \mathbf{C}$ if both of the following hold:

- The complex projection of $\mathbf{1}$ onto $\mathbf{u}$ is $\bar{\alpha}\mathbf{u}$, and is a real projection
- The complex projection of $\bar{\alpha}\mathbf{u}$ onto $\mathbf{u}^2$ is $\bar{\alpha}^2\mathbf{u}^2$, and is a real projection.

Two vectors being compatible means that $\mathbf{1}, \mathbf{u}, \mathbf{u}^2$ (with the last defined by the recurrence relation) doesn't violate the second condition (though $\mathbf{u}^3, \dots$ might).

The property of two vectors being compatible is invariant under rotations of $\mathbf{R}^4$ and multiplication by any complex unit (where both vectors must be multiplied by the same complex unit).

As before, in the second condition we may replace $\bar{\alpha}\mathbf{u}$ with $\mathbf{u}$, $\mathbf{u}^2$ with any complex multiple of $\mathbf{u}^2$, and $\bar{\alpha}^2\mathbf{u}^2$ with $\bar{\alpha}\mathbf{u}^2$.

Write $\mathbf{1} = \bar{\alpha}\mathbf{u} + \mathbf{V}$, so

$$\Re \mathbf{V}, \Im \mathbf{V} \perp \Re \mathbf{u}, \Im \mathbf{u}$$

(not respectively: All four pairs are orthogonal. Indeed, because the complex projection of $\mathbf{1}$ onto $\bar{\alpha}\mathbf{u}$ is a real projection, the real and imaginary parts of $\mathbf{V}$ must be orthogonal to the plane spanned by $\Re \mathbf{u}, \Im \mathbf{u}$.) Then

$$\mathbf{u}^2 = -\mathbf{1} + 2\Re \alpha \mathbf{u} = (2\Re \alpha - \bar{\alpha})\mathbf{u} - \mathbf{V} = \alpha \mathbf{u} - \mathbf{V}$$

$$\bar{\alpha}\mathbf{u}^2 = |\alpha|^2\mathbf{u} - \bar{\alpha}\mathbf{V}.$$

So the second condition in the definition of compatibility is equivalent to the complex projection of $\mathbf{u}$ onto $|\alpha|^2\mathbf{u} - \bar{\alpha}\mathbf{V}$ being $|\alpha|^2\mathbf{u} - \bar{\alpha}\mathbf{V}$ and being a real projection.

So the condition is that there is a real projection sending $\Re\mathbf{u}$ to $|\alpha|^2\Re\mathbf{u} - \Re(\bar{\alpha}\mathbf{V})$ and $\Im\mathbf{u}$ to $|\alpha|^2\Im\mathbf{u} - \Im(\bar{\alpha}\mathbf{V})$.

As $\Re\mathbf{V}, \Im\mathbf{V} \perp \Re\mathbf{u}, \Im\mathbf{u}$, there exists such a real projection if and only if both of the following hold:

- $\Re(\bar{\alpha}\mathbf{V}) \perp \Im(\bar{\alpha}\mathbf{V})$, or equivalently $\Re\mathbf{V} \perp \Im\mathbf{V}$. This is true by construction.
- $|\Re(\bar{\alpha}\mathbf{V})|^2 = 1 - |\alpha|^2|\Re\mathbf{u}|^2$ and $|\Im(\bar{\alpha}\mathbf{V})|^2 = 1 - |\alpha|^2|\Im\mathbf{u}|^2$.

(Indeed, being of the right length means there exists a real projection sending $\Re\mathbf{u}$ to $\Re(\bar{\alpha}\mathbf{u}^2)$; Conditional on this, the possible projections of $\Im\mathbf{u}$ are the sphere with a diameter formed by its projection onto the projection of $\Re\mathbf{u}$ (which is 0 because everything is perpendicular) and its projection onto the orthogonal complement of $\Re\mathbf{u}$.)

A sequence of vectors works if and only if all consecutive pairs except the last two are compatible with $\bar{\alpha}$. For this, we need the real and imaginary components of everything to be orthogonal, but we multiply by $\bar{\alpha}$ each time so the only way being orthogonal like this is preserved is if $\bar{\alpha}$ is purely real or purely imaginary (which only corresponds to a nontrivial point on the boundary if $k = 2$) or if $\|\Re\mathbf{u}\| = \|\Im\mathbf{u}\|$. The latter case uniquely determines the projections.

The constructed realization also shows that the asymptotic supremum of $\|A^m - A^{m+1}\|$ is the same for real projections as for complex projections.

As an aside, this also implies that this can't be obtained using projections onto subspaces of codimension 1, because the real and imaginary parts of the $\mathbf{u}^n$ have to be orthogonal, and that can't be preserved under taking a projection onto a subspace of codimension 1 unless one of the vectors is in the subspace.

2.5.13 Remark on the task dependency on forgetting

One might also be interested in exploring the question of how task dependency affects forgetting in general. One way to capture the task dependency is through the Friedrichs number or its like as they govern the geometric decay rate of residual errors and forgetting for a fixed set of tasks. Specifically, if you consider any sequence of T number of fixed tasks, their Friedrichs angle (for $T = 2$) and its extension, the Friedrichs number (for $T > 2$), are always less than 1 [7, 12]. This causes the residual error to converge geometrically [12]. As a result, the rate of forgetting also converges geometrically, and not inversely proportional to the number of iterations, as suggested by our bounds or by Evrons' [41] for worst-case scenarios. We refer readers interested in such results to [12].

2.5.14 Forgetting vs. Regret

In our context (assuming consistent tasks), one could define the regret for a sequence of tasks S at iteration n as

$$R_S(n) := \frac{1}{n} \sum_{t=1}^n \|X_t w_t - y_t\|^2,$$

in contrast to the forgetting which was defined as

$$F_S(n) := \frac{1}{n} \sum_{t=1}^n \|X_t w_n - y_t\|^2.$$

While superficially similar, analyzing regret is quite different (and much simpler) than analyzing forgetting in our setting. Indeed the regret over the first k iterations is simply the sum of squares of the update distances. By iterating the Pythagorean Theorem, one can see

$$\sum_{t=1}^n \|X_t w_t - y_t\|^2 = \|w_n\|^2 + \sum_{t=1}^n \|w_t - w_{t-1}\|^2 = \|w_n - w_0\|^2 \leq 4\|w_0\|^2$$

since $w_t - w_{t-1}$ is orthogonal to w_t for $t \geq 1$. This means $R_S(n) \leq O(1/n)$, which is tight

even if convergence occurs after a single iteration. (Meaning that w_1 satisfies all constraints.)

Chapter 3

Greedy Orderings

This chapter is based on [122]. All figures in this chapter either come directly from or are adapted versions of figures from that paper.

3.1 Introduction

Continual learning is a subfield of machine learning in which a learner is exposed to tasks or datasets sequentially. In such setups, only a single task is fully accessible at any given time. This models, for instance, computational limitations, data retention or privacy constraints, or the temporal nature of the environment. Much of continual learning research focuses on mitigating forgetting or improving transfer. The role of task ordering is not yet fully understood.

Studying how task order affects learning and what characterizes optimal orderings has both theoretical and practical relevance. Such study can suggest failure modes and suggest a model of interplay between forgetting and transfer. It can also guide the design of environments and algorithms for continual learning. Furthermore, it can inform active control over task sequences in settings that permit it (e.g., robotic environments), relating the problem to continual learning, multitask learning, curriculum learning, and active learning.

One can use the following high-level questions about task ordering as directions of study:

- Can task ordering by itself mitigate forgetting, even under vanilla continual training?
- What constitutes an “optimal” task ordering?
- Is it better to learn when adjacent tasks are similar or dissimilar?
- Can greedy strategies systematically outperform random task orderings?

One direction in the continual learning literature is the design of task orderings informed by task similarity. This idea appears in several earlier works, with varying degrees of emphasis and differing motivations [e.g., 55, 69, 76, 88, 89, 97, 108, 109]. Most closely related to our work is Bell and Lawrence [17], who were among the first to explicitly and systematically examine such orderings in continual learning. They hypothesized that optimal performance would arise when adjacent tasks are *similar*. They empirically found the opposite—orderings with *dissimilar* adjacent tasks led to better performance. More recently, Li and Hiratani [75] reached a similar conclusion and further proposed arranging tasks from the least to the most “typical”.

The above studies are either empirical [17, 88, 89], based on restrictive data assumptions [75, 76], or focused solely on task-incremental settings [97], with some of their findings appearing inconclusive or with no clear unification.

In this paper, we study one formalization of “similarity-guided” orderings through greedy task selection.

Building on a projection-based perspective of continual learning [41, 42], we introduce two greedy orderings—Maximum Distance and Maximum Residual—commonly studied in the Kaczmarz [90, 92] and projection methods literatures [2, 49].

In this paper, we describe some of the geometry of iterates under greedy orderings, as illustrated in fig. 3.1. We then make some observations by experimenting on synthetic and CIFAR-100 tasks and proving optimality results for high-rank tasks.

Under general tasks, although without-replacement random orderings are known to converge [11, 43], we show that single-pass greedy orderings may fail catastrophically. We further prove that this drawback is resolved under greedy orderings with repetition, for which we establish a dimension-independent convergence bound. Finally, we introduce a hybrid scheme combining greedy and random orderings.

We hope that the theoretical foundations—perspectives, tools, and findings—laid out in this paper will inspire future work on task orderings and their potential to mitigate catastrophic forgetting.

Summary of our contributions.

1. We give one formalization of similarity-guided orderings in continual linear regression via greedy strategies, drawing on tools and intuitions from the Kaczmarz and projection literature (section 3.3).
2. We run a few experiments in which greedy orderings converge faster than random orderings, both on synthetic regression tasks and on CIFAR-100-based classification tasks (section 3.4.1).
3. We prove optimality and convergence guarantees for high-rank tasks (section 3.4.2).
4. For general-rank data, we design adversarial task collections in which single-pass greedy orderings provably induce catastrophic forgetting, i.e., yield an $\Omega(1)$ loss even as $T \rightarrow \infty$ (section 3.5.1).
5. In contrast, we prove an $\mathcal{O}(1/\sqrt[3]{k})$ upper bound for greedy orderings with repetition (section 3.5.2).
6. We combine greedy and random orderings into a hybrid strategy that inherits the bounds of random orderings, avoiding catastrophic forgetting in expectation and performing better than random in our experiments (section 3.5.3).

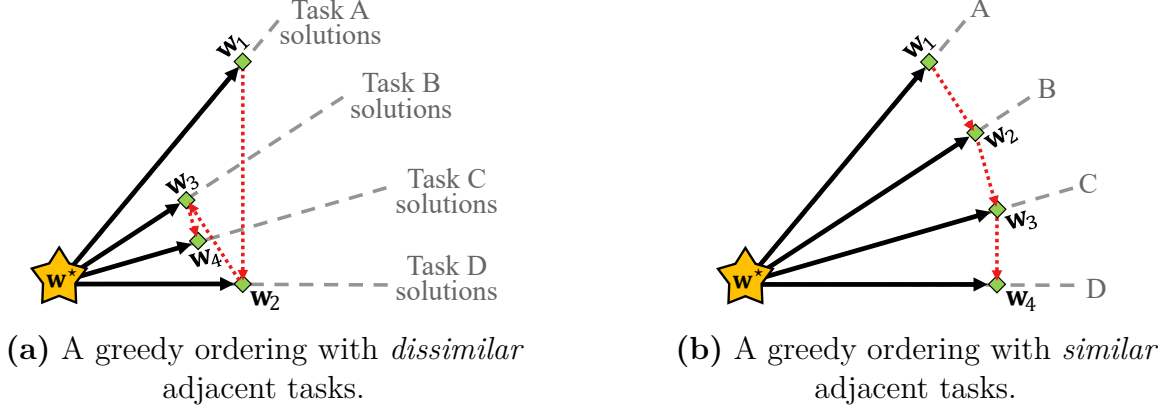


Figure 3.1: **Intuition.** Consider a collection of jointly-realizable linear regression tasks (e.g., A, B, C, D). Each task has an affine solution space (e.g., where $\mathbf{X}_A \mathbf{w} = \mathbf{y}_A$), and $\mathbf{w}_*$ is an “offline” joint solution at the intersection of all tasks. Employing a projection perspective on learning in continual models [41, 42], transitions between *dissimilar* tasks ($A \rightarrow D \rightarrow B \rightarrow C$) intuitively lead to faster convergence toward the intersection compared to transitions between *similar* tasks ($A \rightarrow B \rightarrow C \rightarrow D$).

3.2 Setting and Background: Continual Linear Regression

We focus on continual linear regression, common in theoretical continual learning [e.g., 8, 36, 41, 43, 46, 57, 76, 96]. This setting, though simple, already gives rise to several continual learning phenomena, such as interactions between forgetting, task similarity, and overparameterization [see 47].

Notation. We reserve bold symbols for matrices and vectors, e.g., $\mathbf{X}, \mathbf{w}$. We use $\|\cdot\|$ to denote the Euclidean norm of vectors and the spectral (L2) norm of matrices. $\mathbf{X}^+$ denotes the Moore–Penrose pseudoinverse of a matrix. Finally, we denote $[n] = 1, \dots, n$.

Formally, the learner is given access to a task collection of T linear regression tasks, i.e., $(\mathbf{X}_1, \mathbf{y}_1), \dots, (\mathbf{X}_T, \mathbf{y}_T)$ where $\mathbf{X}_m \in \mathbb{R}^{n_m \times d}$, $\mathbf{y}_m \in \mathbb{R}^{n_m}$. We denote the data “radius” by $R \triangleq \max_{m \in [T]} \|\mathbf{X}_m\|$. For k iterations, the learner sequentially learns the tasks according to a task ordering $\tau : [k] \rightarrow [T]$.

Scheme 3.1 (Continual linear regression (to convergence)).

Initialize $\mathbf{w}_0 = \mathbf{0}_d$

For each iteration $t = 1, \dots, k$: $\mathbf{w}_t \leftarrow$ Start from $\mathbf{w}_{t-1}$ and minimize the current task's

$$\text{loss } \mathcal{L}_{\tau(t)}(\mathbf{w}) \triangleq \left\| \mathbf{X}_{\tau(t)} \mathbf{w} - \mathbf{y}_{\tau(t)} \right\|^2 \text{ with (S)GD to convergence}$$

Output $\mathbf{w}_k$

We assume throughout the paper that there exists at least one joint solutions that solves all T tasks jointly with 0 loss. This assumption is common¹ in many theoretical continual learning papers and makes the analysis easier mathematically [e.g., 41–43, 47, 63, 119]. Joint realizability also holds in some highly overparameterized settings, such as deep networks in the neural tangent kernel (NTK) regime under certain interpolation conditions [see 27, 60].

Assumption 3.1 (Joint Linear Realizability of Training Data). Assume the intersection of *all* task solution subspaces is non-empty, i.e.,

$$\mathcal{W}_\star \triangleq \bigcap_{m=1}^T \mathcal{W}_m \triangleq \bigcap_{m=1}^T \left\{ \mathbf{w} \in \mathbb{R}^d \mid \mathbf{X}_m \mathbf{w} = \mathbf{y}_m \right\} \neq \emptyset.$$

We focus on the joint solution with the minimum norm, often linked to improved generalization.

Definition 3.2 (Minimum-norm joint solution). Denote specifically

$$\mathbf{w}_\star \triangleq \underset{\mathbf{w} \in \mathcal{W}_\star}{\operatorname{argmin}} \|\mathbf{w}\|.$$

¹A different trend in continual learning theory is to assume an underlying linear model, like we do, but allow additive label noise [e.g., 35, 46, 73, 74, 76, 138]. Such papers often add assumptions such as commuting covariance matrices or i.i.d. features across tasks. However, if the covariance matrices commute, then by definition the final iterate depends only on the set of tasks learnt, not their order. We do not study this alternative further in this paper.

At times in this paper we will, without mention, translate our setting so that $\mathbf{0}_d$ is a common solution, and in particular $\mathbf{w}_\star = \mathbf{0}_d$. In this case we also translate the initialization away from $\mathbf{0}_d$.

We follow prior theoretical work [e.g., 36, 41–43, 47] and study the model’s ability to not “forget” previously seen data, and accumulate expertise on the training data of all tasks. This focus isolates continual dynamics from statistical generalization effects that also arise in non-continual, stationary settings. That is, we do not study how well the final model generalizes to tasks not provided in the original task list.

Definition 3.3 (Average loss). The (training) loss of an individual task $m \in [T]$ is defined as $\mathcal{L}_m(\mathbf{w}) \triangleq \|\mathbf{X}_m \mathbf{w} - \mathbf{y}_m\|^2$. The loss we analyze is the average across all T tasks, which in our realizable setting takes the following form:

$$\mathcal{L}(\mathbf{w}_k) \triangleq \frac{1}{\|\mathbf{w}_\star\|^2 R^2} \cdot \frac{1}{T} \sum_{m=1}^T \mathcal{L}_m(\mathbf{w}_k) = \frac{1}{\|\mathbf{w}_\star\|^2 R^2} \cdot \frac{1}{T} \sum_{m=1}^T \|\mathbf{X}_m (\mathbf{w}_k - \mathbf{w}_\star)\|^2 ,$$

where we also normalize by the scaling factors $\|\mathbf{w}_\star\|$ and $R \triangleq \max_{m \in [T]} \|\mathbf{X}_m\|$.

Such scaling factors are natural as rescaling the space or the data affects the geometry of the iterates only by a dilation by the corresponding factor. For this normalization factor to be defined, we assume $\|w^*\|, R \neq 0$.

Remark 3.4 (Forgetting vs. loss). An alternative quantity considered in continual learning [23, 41] is the forgetting, defined as the loss *degradation* at iteration k across previously seen tasks only, i.e., $\frac{1}{k} \sum_{t=1}^k (\mathcal{L}_{\tau(t)}(\mathbf{w}_k) - \mathcal{L}_{\tau(t)}(\mathbf{w}_t))$, or simply as $\frac{1}{k} \sum_{t=1}^k \|\mathbf{X}_{\tau(t)} \mathbf{w}_k - \mathbf{y}_{\tau(t)}\|^2$ in our realizable setting. For single-pass orderings where each task is seen once, the forgetting ultimately coincides with the average loss. For with-replacement settings, these do not agree, and in that case we study the loss and not the forgetting.

One can also study the distance to $\mathbf{w}_\star$.

Definition 3.5 (Distance to the joint solution). After k iterations, the (squared) distance is,

$$D^2(\mathbf{w}_k) = \frac{1}{\|\mathbf{w}_\star\|^2} \cdot \|\mathbf{w}_k - \mathbf{w}_\star\|^2.$$

This distance upper bounds the loss, as can be shown using simple norm inequalities.

Proposition 3.6 (Linking the quantities). After k iterations of Scheme 3.1 on jointly realizable tasks, the loss is upper bounded by the distance to the joint solution.

$$\mathcal{L}(\mathbf{w}_k) = \frac{1}{\|\mathbf{w}_\star\|^2 R^2} \cdot \frac{1}{T} \sum_{m=1}^T \|\mathbf{X}_m (\mathbf{w}_k - \mathbf{w}_\star)\|^2 \leq \frac{1}{\|\mathbf{w}_\star\|^2} \cdot \|\mathbf{w}_k - \mathbf{w}_\star\|^2 = D^2(\mathbf{w}_k).$$

It is possible for this distance to remain large while the loss is small (but nonzero) even if there is a unique solution, showing that converging to $\mathbf{w}_\star$ is not mandatory for continual learning [41].

The distance from the final iterate to $\mathbf{w}_\star$ has clear dependence on the geometry of the tasks beyond the number of iterations, such as its condition number [104]. Thus we focus on the loss.

Geometric interpretation to learning. In each iteration of Scheme 3.1, the learner minimizes the squared loss of the current task to convergence.² Each iterate $\mathbf{w}_t$ of this scheme above is known [41] to implicitly follow the following closed-form update rule,

$$\mathbf{w}_t = \mathbf{X}_{\tau(t)}^+ \mathbf{y}_{\tau(t)} + (\mathbf{I} - \mathbf{X}_{\tau(t)}^+ \mathbf{X}_{\tau(t)}) \mathbf{w}_{t-1}. \quad (3.1)$$

In our realizable setting, this update rule admits a geometric interpretation.

²This simplifies the geometry and the analysis. Other choices exist as well, e.g., a fixed number of steps per task [63, 72].

Evron et al. [41] identified an orthogonal projection operator,

$$\mathbf{P}_m \triangleq \mathbf{I} - \mathbf{X}_m^+ \mathbf{X}_m \in \mathbb{R}^{d \times d},$$

which we use for analysis only (Scheme 3.1 never explicitly computes pseudoinverses or SVDs).

Under the realizability assumption, $\mathbf{y}_{\tau(t)} = \mathbf{X}_{\tau(t)} \mathbf{w}_\star$. We plug it into (3.1) and obtain:

$$\begin{aligned} \mathbf{w}_t &= \mathbf{X}_{\tau(t)}^+ \mathbf{X}_{\tau(t)} \mathbf{w}_\star + (\mathbf{I} - \mathbf{X}_{\tau(t)}^+ \mathbf{X}_{\tau(t)}) \mathbf{w}_{t-1} \\ \mathbf{w}_t - \mathbf{w}_\star &= \mathbf{P}_{\tau(t)} (\mathbf{w}_{t-1} - \mathbf{w}_\star). \end{aligned} \tag{3.2}$$

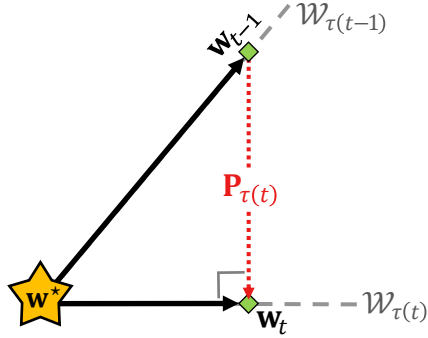


Figure 3.2: **Projection dynamics.**

Geometrically, $\mathbf{w}_{t-1}$ is projected by an affine projection onto the solution space of task $\tau(t)$. In our paper, we adopt this projection-based perspective, which has been used previously [41, 42], to build intuition about greedy orderings.

3.3 Greedy task orderings: A formal approach and intuition

As discussed in section 3.1, the task order affects the dynamics of many machine learning settings. This phenomenon has also been observed in continual learning, both analytically

and empirically. Several works have proposed leveraging “similarity-guided” task orderings—placing dissimilar tasks consecutively.

This section draws on connections between other research areas and our formalization of greedy task orderings.

Geometric intuition. As illustrated in fig. 3.2, the projection perspective allows us to decompose $\|\mathbf{w}_t - \mathbf{w}_\star\|^2$ using projection properties and the Pythagorean theorem as:

$$\begin{aligned}\|\mathbf{w}_t - \mathbf{w}_\star\|^2 &= \|\mathbf{w}_{t-1} - \mathbf{w}_\star\|^2 - \|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2 \\ &= \|\mathbf{w}_{t-1} - \mathbf{w}_\star\|^2 - \|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2.\end{aligned}\tag{3.3}$$

So to try and minimize $\|\mathbf{w}_t - \mathbf{w}_\star\|^2$, one could greedily maximize $\|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2$.

This has inspired a myriad of studies on Kaczmarz³ and projection methods [e.g., 2, 19, 92] which employed ordering schemes that greedily maximize $\|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2$, in the following spirit.

Definition 3.7 (Maximum Distance Ordering). Greedily maximize the distance between iterates, i.e.,

$$\begin{aligned}\tau_{\text{MD}}(t) &= \operatorname{argmax}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} \|\mathbf{X}_m^+ (\mathbf{X}_m \mathbf{w}_{t-1} - \mathbf{y}_m)\|^2 \\ &= \operatorname{argmax}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2\end{aligned}$$

at each iteration $t \in [T]$, where $\tau_{\text{MD}}(1:t-1) \triangleq \{\tau_{\text{MD}}(1), \dots, \tau_{\text{MD}}(t-1)\}$.⁴

Our earlier fig. 3.1 illustrates the MD ordering and how it could lead to faster convergence

³The Kaczmarz method [40, 64], further explained in section 3.7, iteratively solves a linear system of equations.

⁴In practice, the MD rule is easy to compute for rank-1 tasks, since it reduces to $\frac{1}{\|\mathbf{x}_m\|^2} \|\mathbf{x}_m^\top \mathbf{w}_{t-1} - y_m\|^2$. In higher ranks, this rule is harder to compute exactly, but can be estimated, e.g., with a subsample of the task.

to $\mathbf{w}_\star$.

Distance and task similarity. One can heuristically model the distance between iterates $\mathbf{w}_{\tau(t-1)}$ and $\mathbf{w}_{\tau(t)}$ as positively correlated with the principal angles between these subspaces [41]. These angles can be used to quantify task similarity, as illustrated in the setting of section 3.4.2 and fig. 3.1.

An alternative greedy ordering found in the literature is the Maximum Residual ordering [e.g., 2, 48, 92, 136]. In some settings, this rule may be easier to compute in full, or to estimate using a small validation set.

Definition 3.8 (Maximum Residual Ordering). Greedily select the task exhibiting the greatest error:

$$\tau_{\text{MR}}(t) = \underset{m \in [T] \setminus \tau_{\text{MR}}(1:t-1)}{\operatorname{argmax}} \quad \|\mathbf{X}_m \mathbf{w}_{t-1} - \mathbf{y}_m\|^2, \quad \forall t \in [T].$$

The MD and MR orderings have the following relation:

$$\mathbf{X}_m = \mathbf{X}_m \mathbf{X}_m^+ \mathbf{X}_m = \mathbf{X}_m (\mathbf{I} - \mathbf{P}_m),$$

and

$$\|\mathbf{X}_m \mathbf{w}_{t-1} - \mathbf{y}_m\|^2 = \|\mathbf{X}_m (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \leq \|\mathbf{X}_m\|^2 \|(\mathbf{I} - \mathbf{P}_m) (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2.$$

Single-pass orderings. Although disallowing repetitions departs slightly from the motivating literature on projection methods, it has been studied in continual learning [see 17, 75]. In curriculum or multitask learning, limiting tasks to a single pass can reduce training costs. We defer the study of the effect of repetitions to section 3.5.2.

Computational tractability of greedy policies. Naively computing the greedy rules in definitions 3.7 and 3.8 has a higher computational cost than a random or cyclic rule. Before continuing our investigation of these orderings, we briefly address their computational feasibility.

1. **Estimation:** Greedy rules can often be estimated efficiently in practical scenarios. For example, the maximum residual rule (definition 3.8) requires the current loss of each available task. This quantity can be approximated using a small subset of samples or via dimensionality reduction techniques [e.g., see 39]. In section 3.9, we show empirically that the maximum residual rule performs comparably to the maximum distance rule, and that its performance remains unharmed even when approximated using only 1% of the data. In deep networks, computing that rule requires only forward passes and may reduce the number of gradient steps—thereby lowering overall time and memory costs by limiting costly backward passes [58].
2. **Heuristics:** Our greedy rules rely on residuals to quantify the similarity between the current task and the remaining ones. This approach is exemplified in fig. 3.1 and (3.5) of section 3.4.2, and is related to principal angles between subspaces [see 41]. Alternatively, one could use heuristic notions of task similarity, such as predefined [69] or computed metrics that use Hessians [17], zero-shot performance [75], or task embeddings [1, 88].
3. **Structured tasks:** If each step updates relatively few residuals (e.g., in a Kaczmarz setting with sparse columns and rows), only few residuals must be recomputed [92].

The “estimation” and “heuristic” considerations suggest that one may wish to generalize our bounds to various notions of approximately greedy task orderings, which we leave to future work.

3.4 Benefits of greedy orderings

In this paper, the competitor to greedy strategies we select is the random strategy, uniformly sampling tasks (rows or blocks in the Kaczmarz context) from the task collection $[T]$, either with or without replacement. That is,

$$\tau_{\text{Unif}}(1), \dots, \tau_{\text{Unif}}(k) \sim \text{Uniform}([T]) \quad (3.4)$$

or the without-replacement variant. As mentioned earlier, greedy strategies have previously been studied for the Kaczmarz method [92, 93] and its block variants [83, 90, 129, 136, 137] employing either deterministic [92] or probabilistic [13, 14, 116, 135] selection rules. Such work often found better provable bounds on the distance to $\mathbf{w}_\star$ (definition 3.5) for greedy orderings compared to random orderings.

In contrast, we study convergence of the loss (definition 3.3), as has also been studied in the continual learning literature [e.g., 41, 42, 63, 119].

While loss is upper bounded by the distance to $\mathbf{w}_\star$ (proposition 3.6), the existing Kaczmarz rates fall short in two ways:

- (i) they rely on repeating rows/blocks and do not apply in the single-pass regime central to continual learning
- (ii) even when repetition is allowed, their rates depend on data eigenvalues, potentially making them much slower than our data-independent $\mathcal{O}(1/\sqrt[3]{k})$ guarantee of theorem 3.13.

In this section, we examine how well the advantages of greedy over random orderings carry over to the loss in continual settings, first empirically and then analytically.

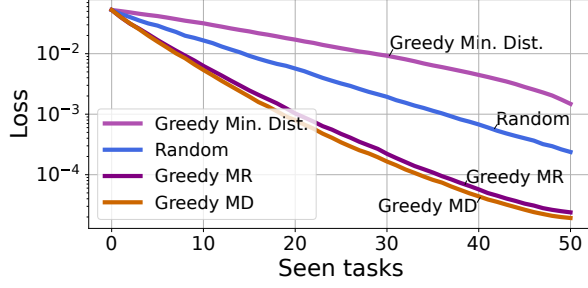
3.4.1 Motivating experiments: Greedy outperforms random ordering

Here, we test different task ordering strategies on synthetic regression tasks and a more complex classification setup: Using linear probing in a domain-incremental CIFAR-100 setting.⁵

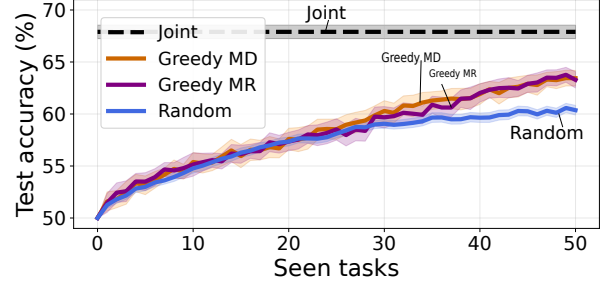
Regression tasks: Random data. The feature matrices $\mathbf{X}_1, \dots, \mathbf{X}_T$ are drawn from a Gaussian distribution. We compare the two “dissimilarity-maximizing” greedy strategies (MD, MR) to the random ordering ((3.4)) and a complementary minimum distance strategy (defined as in definition 3.7, replacing argmax with argmin). Full details, including more combinations of task count T , dimension d , and rank r , as well as experiments on anisotropic data, are provided in section 3.8.

Classification tasks: CIFAR-100. We randomly partition classes into continual binary classification tasks, similarly to Li and Hiratani [75]. We train a linear probe on top of a ResNet-20 embedder, pretrained on the original CIFAR-100 multiclass task [56, 68]. We optimize the cross-entropy loss of each task while employing L2 regularization towards the previous parameters [79]. We compare the performance to a ‘joint’ baseline, trained on all tasks together (not continually). See section 3.9 for full details. There, we conduct further experiments on continual CIFAR-100 tasks as before, but using embeddings pretrained on (i) CIFAR-10 and (ii) CIFAR-100 with only half of the samples from each class (in this case, continual learning is performed on tasks formed from the other half). We also examine more computationally efficient greedy orderings, determined from only a fraction of the data—down to 1% (5 samples per class in CIFAR-100).

⁵We provide a code snippet for the regression experiments in section 3.14. The code for the classification experiments is accessible at https://github.com/matants/greedy_ordering.



(a) **Regression (random data):** Average loss over $T = 50$ regression tasks of rank $r = 10$ in $d = 100$ dimensions, sampled from an isotropic Gaussian distribution. Details in Appendix 3.8.



(b) **Classification (CIFAR-100):** Average test accuracy over $T = 50$ binary classification tasks, generated by randomly partitioning CIFAR-100 classes (a domain-incremental setting). Details in Appendix 3.9.

Figure 3.3: **Task ordering comparison.** Transitioning between dissimilar tasks consistently outperforms random transitions, with Greedy MR and MD achieving comparable performance.

3.4.2 Provable benefits for high-rank, “nearly determined” tasks

To further motivate greedy orderings, we analyze a simple setup where each task’s matrix is of nearly full rank, i.e., $\text{rank}(\mathbf{X}_m) = d - 1$, $\forall m \in [T]$. Even in such a setup, it has been shown that arbitrary orderings of $T \rightarrow \infty$ may lead to catastrophic forgetting and maximal losses [41].

In this setup, each projector can be expressed as a rank-1 operator, i.e., $\mathbf{P}_m = \mathbf{I} - \mathbf{X}_m^+ \mathbf{X}_m = \mathbf{v}_m \mathbf{v}_m^\top$ for a unit vector $\mathbf{v}_m \in \mathbb{R}^d$ that spans the solution space of task m . Then, we can rewrite the Maximum Distance (definition 3.7) rule to explicitly maximize dissimilarity between consecutive tasks, i.e.,

$$\tau_{\text{MD}}(t) = \underset{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)}{\text{argmin}} \left(\mathbf{v}_m^\top \mathbf{v}_{\tau(t-1)} \right)^2 \quad (3.5)$$

$$= \underset{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)}{\text{argmax}} \theta_{m, \tau(t-1)}, \quad (3.6)$$

where we define $\mathbf{v}_{\tau(0)} \triangleq \frac{\mathbf{w}_0 - \mathbf{w}_*}{\|\mathbf{w}_0 - \mathbf{w}_*\|}$; see (3.7) in section 3.10.

Optimality of greedy orderings in terms of distance to $\mathbf{w}_\star$. Earlier in (3.3), we motivated the MD ordering as greedily maximizing the decrease in $\|\mathbf{w}_t - \mathbf{w}_\star\|$. Does this guarantee a minimal distance $\|\mathbf{w}_T - \mathbf{w}_\star\|$ at the end of the sequence? Here, we prove that the MD ordering yields a square-root approximation of the optimal distance at the end of learning.⁶

Lemma 3.9 (Optimality guarantee when $r = d - 1$). Let $\mathbf{w}_T^{\tau_{\text{MD}}}$ and $\mathbf{w}_T^{\tau_\star}$ be the iterates after learning T jointly realizable tasks of rank $d - 1$ under the Maximum Distance ordering τ_{MD} and an optimal ordering $\tau_\star$ that leads to a minimal distance to the joint solution $\mathbf{w}_\star$. Then, their distances hold,

$$0 \leq D^2(\mathbf{w}_T^{\tau_\star}) \leq D^2(\mathbf{w}_T^{\tau_{\text{MD}}}) \triangleq \frac{\|\mathbf{w}_T^{\tau_{\text{MD}}} - \mathbf{w}_\star\|^2}{\|\mathbf{w}_\star\|^2} \leq \frac{\|\mathbf{w}_T^{\tau_\star} - \mathbf{w}_\star\|}{\|\mathbf{w}_\star\|} \triangleq D(\mathbf{w}_T^{\tau_\star}) \leq 1.$$

The full proofs for this section are given in Appendix 3.10.

What about the loss? The optimality of the distance does not imply optimality of the average loss, as exemplified in fig. 3.7 in section 3.6. Instead, we now derive an upper bound for the loss.

Lemma 3.10 (Loss bound when $r = d - 1$). Under the Maximum Distance greedy ordering over T jointly-realizable tasks of rank $d - 1$, the loss of scheme 3.1 after T iterations is upper bounded as,

$$\mathcal{L}(\mathbf{w}_T) = \frac{1}{\|\mathbf{w}_\star\|^2 R^2} \cdot \frac{1}{T} \sum_{m=1}^T \|\mathbf{X}_m \mathbf{w}_T - \mathbf{y}_m\|^2 \leq \frac{1}{eT}.$$

⁶Our optimality result is related to the optimal Hamiltonian path in a predefined similarity graph, as studied in related work [17, 75]. Specifically, in this section’s high-rank case, similarity can be statically defined as $s_{m,m'} = \cos^2(\theta_{m,m'})$. Then, the greedy MD ordering approximates the Hamiltonian path $\tau_\star$ that maximizes $\prod_{t=2}^T s_{\tau_\star(t-1), \tau_\star(t)}$ [see 45, 86]. However, under general-rank tasks, our greedy rules (definitions 3.7 and 3.8) are computed online at each iteration and depend not only on the previous task $\tau(t-1)$ but also on the previous iterate $\mathbf{w}_{t-1}$. Consequently, these rules do not correspond to a Hamiltonian path on any predefined graph.

This rate matches a recent $\mathcal{O}(\frac{d-r}{T})$ bound for random orderings without replacement [43], whereas the best known rate for such orderings with general-rank tasks is $\mathcal{O}(1/\sqrt{T})$ [11]. This raises the question: *for general-rank tasks, can single-pass greedy orderings still compete with random ones, or even outperform them, in worst-case analysis?* Next, we show that they cannot.

3.5 Failure modes in greedy orderings

Under random orderings, with or without replacement, Attia et al. [11] proved a universal, dimensionality-independent rate of

$$\mathbb{E}_{\tau_{\text{Unif}}} \mathcal{L}(\mathbf{w}_k^{\tau_{\text{Unif}}}) \leq \frac{13}{\sqrt{k}}.$$

We prove a clear separation in our setup: while single-pass greedy orderings can fail catastrophically, i.e., not decrease with the number of iterations $k = T$, greedy orderings with repetition enjoy a bound of $\mathcal{O}(1/\sqrt[3]{k})$.

3.5.1 Greedy orderings can fail where random ones do not

We now present cases where the single-pass greedy ordering forgets catastrophically, i.e., suffers an $\Omega(1)$ loss, even after fitting a collection of $T \rightarrow \infty$ tasks. Specifically, we present an example in $d = 3$ dimensions where the loss does not diminish, and a construction in $d = T + 1$ dimensions that exploits dimensionality to yield larger forgetting. Full details and proofs⁷ are given in Appendix 3.11.

Example 3.11 (Adversarial 3D construction). For all $T \in \{4 \cdot 10^i - 1 \mid i = 1, 2, \dots, 7\}$,

⁷The proof of the $d = T + 1$ -dimensional result has some known errors, but numerical evidence suggests the claim is true and I find it plausible that a proof essentially following the lines given in the Appendix works.

there exists a task collection of jointly-realizable tasks in $d = 3$, such that

$$\mathcal{L}(\mathbf{w}_T^{\tau_{\text{MD}}}), \mathcal{L}(\mathbf{w}_T^{\tau_{\text{MR}}}) > 2.78 \cdot 10^{-5}.$$

Theorem 3.12 (Greedy lower bound). For any $d \geq 30$, there exists an adversarial task collection with $T = d - 1$ jointly-realizable tasks (of different ranks) such that both greedy orderings (MD, MR) forget catastrophically. That is, the loss at the end of the sequence is, $\mathcal{L}(\mathbf{w}_T^{\tau_{\text{MD}}}), \mathcal{L}(\mathbf{w}_T^{\tau_{\text{MR}}}) \geq \frac{1}{8} - \frac{1}{4d}$.

We demonstrate the behavior of an adversarial task collection using $T = 999$ tasks in $d = 1000$ dimensions. Our constructed collection “tricks” the greedy orderings: slowly increasing not only the loss on all tasks, but also the forgetting of previous tasks. The model is thus unable to accumulate knowledge and practically forgets everything it learns.

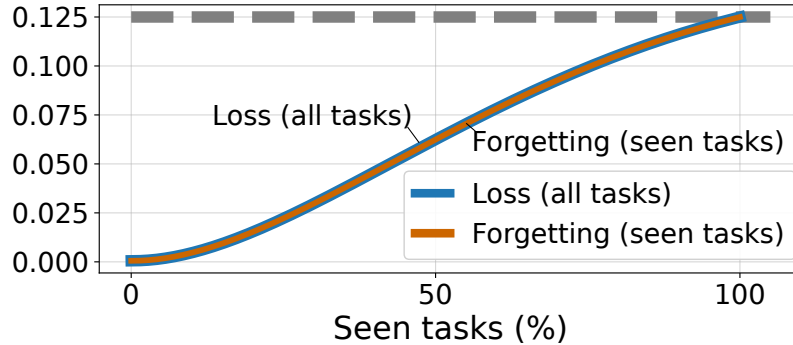


Figure 3.4: Learning an adversarial collection.

3.5.2 Single-pass vs. repetition in greedy orderings

So far, we have focused on single-pass greedy orderings, in which each task is learned exactly once. These are conceptually related to without-replacement sampling and (re)shuffling techniques in SGD and the Kaczmarz method, where repetition-free strategies often converge faster than with-replacement sampling, both empirically [20, 93, 117] and in theory [18, 51,

54, 61, 84; but see 33, 103]. We ask: Does the advantage of orderings without repetition extend to greedy orderings?

Now, we show that repetition in greedy orderings avoids the failure mode of single-pass ones.

Theorem 3.13 (Dimensionality-independent bound for greedy orderings with repetition). Under a Maximum Distance greedy ordering with repetition ($\tau_{\text{MD-R}}$) over T jointly-realizable tasks, the loss of scheme 3.1 after $k \geq 2$ iterations is upper bounded as $\mathcal{L}(\mathbf{w}_k^{\tau_{\text{MD-R}}}) = \mathcal{O}(1/\sqrt[3]{k})$.

section 3.12 provides the proof, a comparison to prior rates (table 3.1), and details for the experiment below.

We evaluate the effect of repetition across orderings under random data. As in prior work, random sampling without replacement outperforms with replacement. In contrast, repetition benefits greedy orderings, likely due to larger updates and faster convergence to $\mathbf{w}_\star$. The slowdown in the single-pass case likely reflects the exhaustion of high dissimilarities.

Intuitively, repetition in random orderings exposes the learner to less data, while in greedy selection it allows considering all tasks at each step.

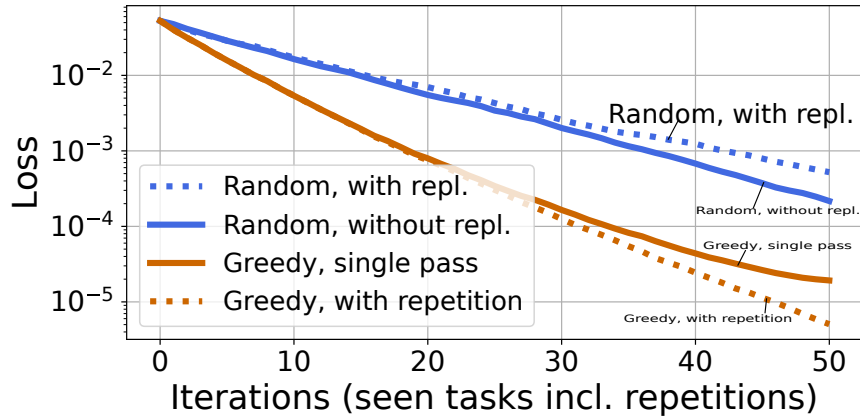


Figure 3.5: **The effect of repetitions.**

However, these findings do not always hold, as seen in our classification example (section 3.9.5).

3.5.3 Extension: Hybrid task orderings

To leverage both the fast empirical convergence of greedy orderings and the analytical convergence guarantees of random ones, we introduce a “hybrid” strategy: begin with greedy selection and switch to random once the decrements $\|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2$ fall below a threshold. Analytically, using a suitable threshold, we prove in lemma 3.24 that any bound for without-replacement random orderings, e.g., $\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$ [11], extends to our hybrid scheme, showing it avoids the failure mode of section 3.5.1.

Empirically, the hybrid ordering performs better than random but worse than greedy. This matches our intuition from eq. (3.3) and fig. 3.1: greedy selection takes larger “steps” (or projections), particularly early on, when most tasks are still available. Once these projections diminish, we switch to the random ordering, which—unlike the greedy approach—cannot be adversarially “tricked” into failure.

Further details and experiments appear in section 3.13.

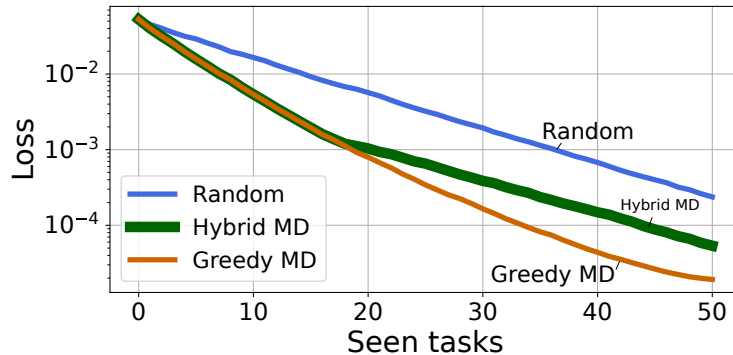


Figure 3.6: Hybrid ordering experiment.

3.6 Discussion and related work

So far, we have studied greedy task orderings, demonstrating empirical and analytical benefits of transitioning between *dissimilar* tasks. Here, we expand on connections and ideas not yet fully covered to better situate our work within the existing literature. In section 3.7,

we discuss further links to Kaczmarz methods, curriculum and active learning, coordinate descent, and example selection in SGD.

Task orderings in continual learning theory. Continual learning theory often treats task orderings as arbitrary. However, several analytical works [e.g., 22, 41–43, 63, 119] show that certain orderings—typically cyclic or random—can mitigate forgetting (matching empirical findings [71]). While some works downplay ordering effects—arguing they are often minor—and defer their study to future work [111], others design continual learning algorithms specifically for evolving sequences with very similar adjacent tasks [6]. We follow a different line of work that studies how pairwise task similarities, or dissimilarities, influence continual schemes.

A particularly relevant work by Bell and Lawrence [17] advocates pairwise task dissimilarity as a guiding principle and was among the first to empirically investigate similarity-guided task orderings. Tasks are represented as vertices in a complete graph, where edge weights correspond to a predefined distance between tasks, and Hamiltonian paths represent full task orderings. They hypothesized that a minimum-weight path (favoring similar tasks in succession) would yield the best continual performance. Yet their experiments on simple neural networks indicated the opposite: maximum-weight paths, placing dissimilar tasks adjacently, often led to improved performance. Still, these results were not always statistically significant (see their Figure 5)—motivating us to revisit similarity-guided task orderings from a more analytical perspective.

Li and Hiratani [75] conduct a deeper investigation into similarity-guided task orderings, obtaining more statistically robust empirical results. They likewise find that adjacent tasks should be *dissimilar*, and further explore task “typicality” (discussed below). While deriving results for a linear regression model to support their empirical observations (on neural networks), they rely on a restrictive analytical data model in which features are randomly drawn from a simplified distribution across tasks. In contrast, our analysis accommodates

arbitrary feature matrices, allowing richer and more realistic forms of task similarity. Like Bell and Lawrence [17], their goal is to characterize optimal orderings in general, whereas we formalize and analyze *greedy* orderings specifically, both as a practical strategy and as a proxy for optimal ones.

Ruvolo and Eaton [107] propose an “information maximization” approach to task ordering, using a diversity-based heuristic related to our maximum residual strategy (definition 3.8). However, their complex model limits rigorous theoretical analysis of the kind we provide.

Lin et al. [76] examine the role of task similarity and reach conclusions broadly aligned with ours. While influential, their work differs from ours in several ways. First, their analysis relies on a restrictive i.i.d. feature assumption across tasks. They also assume a distinct *teacher* model per task, unlike our setting, where all tasks share a single overparameterized model—as is common in modern deep learning. Consequently, their notion of task similarity relies on teacher similarity, rather than more practical measures such as residuals (as in definition 3.8) or feature similarity. Although they note ordering effects in their expressions and briefly support them with classification experiments, task ordering is not their primary focus. In contrast, we offer a comprehensive treatment of similarity-guided orderings specifically—providing formal definitions, geometric intuitions, greedy strategies, optimality results, empirical validation, failure modes, and repetition analysis.

Can similarity-guided task orderings alone mitigate forgetting? Methods such as replay, regularization, and parameter isolation are widely used to mitigate forgetting in continual learning [24, 31, 66, 72, 102, 106]. However, they depart somewhat from standard (“vanilla”) deep learning practices that apply plain gradient methods to the (current) loss. Both our work and prior studies show that even without such mechanisms, task ordering *alone* can affect forgetting. For instance, randomizing the task order—with or without replacement—can alleviate forgetting [11, 41, 43, 71]. In contrast, we show that single-pass

greedy ordering can exacerbate forgetting (section 3.5.1), while allowing task repetitions mitigates this effect (section 3.5.2). Moreover, ordering strategies can be combined with other approaches; for example, in our classification experiments we also employ regularization (section 3.4.1). This underscores the importance of studying task ordering as a complementary way to mitigate forgetting while potentially keeping continual learning closer to standard deep learning practice.

Task typicality at the end of learning. Li and Hiratani [75] suggest that tasks should be arranged from least to most “typical”. While we did not focus on this aspect of orderings, our geometric interpretation can illustrate it.

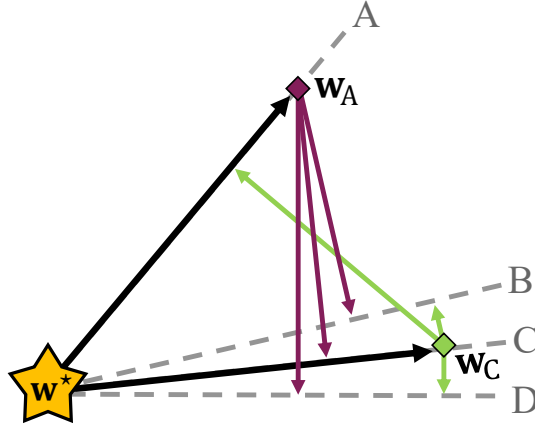


Figure 3.7: **Task typicality.**

Our motivation was to minimize the distance $\|\mathbf{w}_k - \mathbf{w}_\star\|^2$, which upper bounds the *loss* $\frac{1}{T} \sum_{m=1}^T \|\mathbf{X}_m (\mathbf{w}_k - \mathbf{w}_\star)\|^2$. However, this bound can be loose, and minimizing the distance does not guarantee the lowest loss. For example, in fig. 3.7, although $\|\mathbf{w}_A - \mathbf{w}_\star\|^2 = \|\mathbf{w}_C - \mathbf{w}_\star\|^2$, the point $\mathbf{w}_C$ is a better ending point than $\mathbf{w}_A$, inducing a lower loss (the arrows represent the residuals). This happens because task C is more typical—i.e., more similar to other tasks—than task A. The empirical advantage of greedy ordering may stem from a tendency to postpone typical tasks, perhaps causing its benefits to emerge only in later stages of training (see fig. 3.3).

Regret today or loss tomorrow? In section 3.3, we motivated the use of greedy orderings to minimize the distance to the joint solution $\|\mathbf{w}_k - \mathbf{w}_\star\|^2$, which in turn upper-bounds the average loss over *all* tasks: $\frac{1}{T} \sum_{m=1}^T \|\mathbf{X}_m (\mathbf{w}_k - \mathbf{w}_\star)\|^2$. This objective is related, but not identical, to the notion of *regret*, which quantifies the loss along the optimization *path* on *consecutive* tasks, i.e., $\frac{1}{k} \sum_{t=1}^k \|\mathbf{X}_{\tau(t)} (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2$. From this definition and fig. 3.1, we observe that regret—though also upper-bounded by the distances $\|\mathbf{w}_{t-1} - \mathbf{w}_\star\|^2$ —can often benefit from transitions between *similar* tasks rather than *dissimilar* ones. In other words, to make accurate predictions *during* learning—e.g., in decision-making—transitioning between *similar* tasks may be preferable. Conversely, to minimize average loss *across* tasks—e.g., in curriculum or multitask learning—our findings suggest that transitioning between *dissimilar* tasks is preferable.

Other continual setups. The majority of studies support our conclusion that sequential task *dissimilarity* is beneficial [e.g., 17, 41, 76, 81, 88, 100, 107, 110]. Still, the specific continual setup can influence the behavior of task orderings. We consider a “domain-incremental” setting: learning the same problem across different domains, i.e., $\mathcal{P}(X)$ changes but $\mathcal{P}(Y|X)$ is fixed [70, 124]. Alternatively, “task-incremental” setups involve distinct tasks—possibly with different $\mathcal{P}(Y|X)$ —with task identity known at both train *and* test time. There, prior work [89, 97] trained a *separate* model per task and found that *similarity-maximizing* orderings prevail, seemingly contradicting our findings. However, in such scenarios, the focus shifts from *forgetting* to inter-task *transfer*, benefiting from *similar* consecutive tasks (see discussion on regret). Hence, their results complement ours.

Others have studied “class-incremental” learning (CIL), where each task introduces new objects or classes, aiming for strong overall performance (e.g., in split benchmarks [118]). While some CIL papers suggest that consecutive task *similarity* is preferable [55, 82], a closer look reveals that they modify the class composition *within* each task, inducing high *intra-task* *heterogeneity* [9, 55]. This likely leads to wider minima and stronger “transferability” to other

tasks, thus explaining their improved results. Such configurations resemble curriculum more than continual learning.⁸ To our knowledge, only Yang and Li [132] report contradictory results, possibly due to their empirical design.⁹ Finally, we note that the effects discussed here are related to the *interleaving effect* in educational psychology [94, 105].

Future work. One could extend our findings to other settings—such as class- and task-incremental, discussed earlier—and to more complex continual learning methods, such as replay and regularization. Moreover, our linear realizability assumption could be relaxed to accommodate label noise or even extend to nonlinear models, possibly borrowing tools from Kaczmarz literature [15, 136]. One may also be able to combine our approach with common wisdom in curriculum learning—i.e., to design orderings that account for both task similarity and difficulty.

Finally, another potential direction for achieving tighter upper bounds in continual linear regression (see table 3.1) lies in probabilistic selection rules, inspired by randomized greedy Kaczmarz methods [13, 14, 116, 135], which could combine the strengths of greedy orderings with the robustness of randomness, akin to our proposed hybrid scheme.

3.7 Further related work

Here, we elaborate on additional connections not fully addressed in section 3.6.

Alternative viewpoint: The Kaczmarz method. The Kaczmarz method [40, 64], a classical iterative projection algorithm for solving linear systems of equations. Our continual

⁸The learner controls the internal task composition to create “easier” tasks, as in curriculum learning [128].

⁹They construct the first task using a random half of the classes. This strong “pretraining” leads to low initial loss, as the model already learns *half* the classes. This resembles the failure modes discussed in section 3.5.1.

linear regression scheme maps directly to this. In our context, the system is $\mathbf{X}\mathbf{w} = \mathbf{y}$, where

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_T \end{pmatrix} \in \mathbb{R}^{N \times d}, \quad \mathbf{y} = \begin{pmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_T \end{pmatrix} \in \mathbb{R}^N, \quad \text{where } N = \sum_{m=1}^T n_m.$$

Evron et al. [41] pointed out that Kaczmarz methods iteratively solve the “block” systems of the form $\mathbf{X}_{\tau(t)}\mathbf{w} = \mathbf{y}_{\tau(t)}$ using an update rule *equivalent* to our continual update in (3.1). As a result, the observations and results in our paper extend naturally to the greedy Kaczmarz method. However, whereas Kaczmarz studies typically analyze convergence in terms of the distance to the intersection $\mathbf{w}_\star$, we focus on the *loss*, i.e., the residuals (Definitions 3.5 and 3.3, respectively). For example, in the $r = d - 1$ case of section 3.4.2, this distinction allowed proving an upper bound on the *loss* and an “approximation” result on the optimal *distance* (Lemmas 3.10 and 3.9, respectively).

It is worth noting that, via its connection to the Kaczmarz method, our continual linear regression scheme is also related to coordinate descent methods [101]. While prior work in this area shows that greedy selection can outperform random sampling, these results often rely on strong convexity assumptions [44, 91], and typically apply to the Kaczmarz method through a primal-dual lens [41, 134]—again, yielding only convergence to the intersection point $\mathbf{w}_\star$.

Curriculum learning. Broadly, curriculum learning enhances training by controlling the order in which data are presented, to accelerate convergence or improve accuracy. The prevailing view is that examples should be ordered from *easy to hard* by their “difficulty” [113, 128]. In contrast, we study *similarity*-guided orderings, aligning with recent findings in continual learning [17, 57].

A distinction between curriculum and continual learning lies in the unit of ordering: while curriculum learning typically orders individual samples or batches, we focus on orderings of

entire tasks. Moreover, curriculum learning often takes a single gradient step per sample, whereas continual learning optimizes each task to a low loss before proceeding. Nonetheless, given the maturity of curriculum learning and the interdisciplinary nature of our work, some curriculum studies that operate at the task level are directly relevant [e.g., 97] and were discussed in section 3.6.

Example selection in SGD. Evron et al. [43] show that learning an *entire* (continual) linear regression task in our scheme 3.1 reduces to taking a *single* large gradient step on a modified objective. While they use this reduction to analyze *random* orderings via last-iterate SGD analysis, we leverage it here to draw connections between *greedy* task orderings and greedy example selection in SGD.

Most of the example selection literature considers multi-epoch settings, where each sample is seen multiple times. In such regimes, it is common to *randomly shuffle* the dataset once or at the start of each epoch [e.g., 51, 84], but this is not necessarily *optimal* [99]. For instance, Lu et al. [78] show that greedy permutations—computed at the beginning of each epoch—can yield faster convergence than random ones. However, their analysis requires (1) multiple epochs and (2) very small step sizes, making it inapplicable to our single-pass continual setting.

Das et al. [30] show that a selection rule akin to our maximum residual rule (definition 3.8) accelerates *early* convergence but may underperform random orderings asymptotically—aligning with our findings on the hybrid approach in section 3.5.3. They also analyze an approximate rule, supporting our observations on computational tractability in section 3.3. Others select greedily by gradient magnitude instead of loss [130], or “mine” examples at the *mini-batch* level—selecting “hard” samples with a high loss or ones that lead to a significant decrease in loss [112, 126].

Active Learning. Active learning aims to reduce labeling cost by querying the most informative samples for labeling, typically from a large unlabeled pool. This setting resembles ours, where the learner may apply a greedy maximum distance rule (definition 3.7) to select the task or sample expected to induce the greatest model update. For example, a related idea is explored in Cai et al. [21], who propose a greedy maximum distance variant for regression. Since labels are unknown at selection time in their active learning setup, they approximate the expected model change using a bootstrap method. Empirically, this approach identifies informative examples and consistently improves generalization across datasets.

3.8 Appendix to section 3.4.1: Regression experiments

All figures report averages over 10 runs. In each run, we randomly sample a task collection to evaluate the different ordering strategies. Shaded regions (see sections 3.12.2 and 3.13.3) indicate ± 1 standard error intervals, even when not visually discernible. In section 3.8.3 we further discuss the statistical significance of our experiments.

Computational resources. All regression experiments—including those not shown—were completed within 4 hours on a home PC equipped with an Intel i5-9400F CPU and 16 GB of RAM.

3.8.1 Isotropic data

figs. 3.8 and 3.9 extend the previous experiment on isotropic data (fig. 3.3) to varying dimensions d , ranks r and task counts T . Results confirm consistent patterns: greedy (dis-similarity maximizing) methods outperform random, and MD is better than MR across all settings (sometimes only slightly).

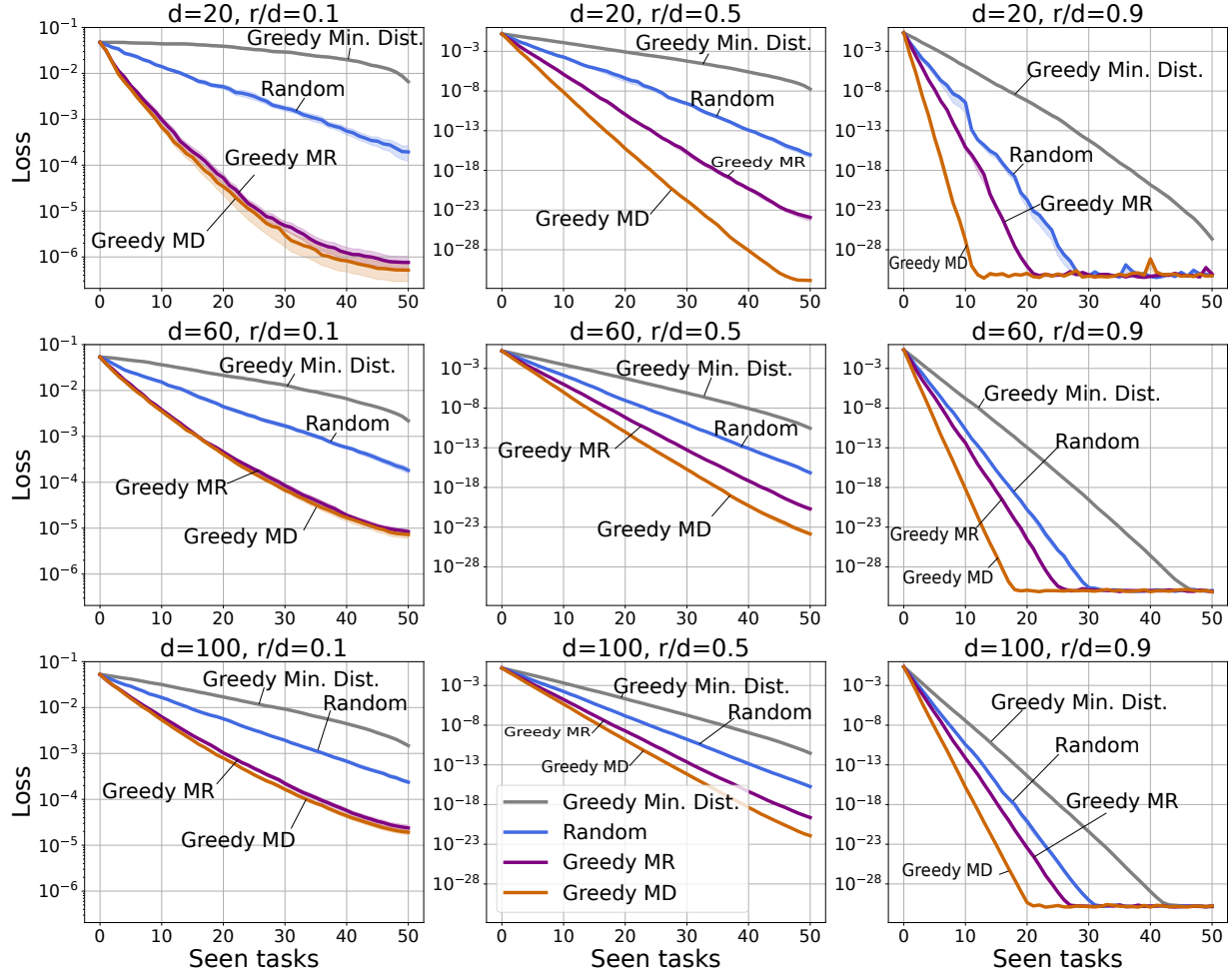


Figure 3.8: **Comparing orderings for varying dimensions d and ranks r of the data matrices, for isotropic data.** $T = 50$. For such isotropic data, for fixed ratio r/d , the random ordering performance changes relatively little. In contrast, greedy orderings that prioritize dissimilarity benefit from a lower dimension when r/d is fixed, and Greedy Min Dist benefits from a higher dimension (to see this from the plots, focus on single columns in the grid). This may be related to some notion of increased task “density” in lower dimensions: when r/d is fixed, increasing d increases $d - r$, expanding the set of possible task projections (see (3.2)). As a result, a fixed number of tasks T covers this space more sparsely in higher dimensions. In lower dimensions, the same T tasks yield denser coverage, which may cause greedy dissimilarity-based selection to take steps with large projections. In all strategies, for fixed ambient dimension, higher task rank consistently yields improved performance (focus on single rows). This is because the solution subspaces are of rank $d - r$, so increasing r (with fixed d) lowers the subspace rank, increasing the distances between them and resulting in larger projections.

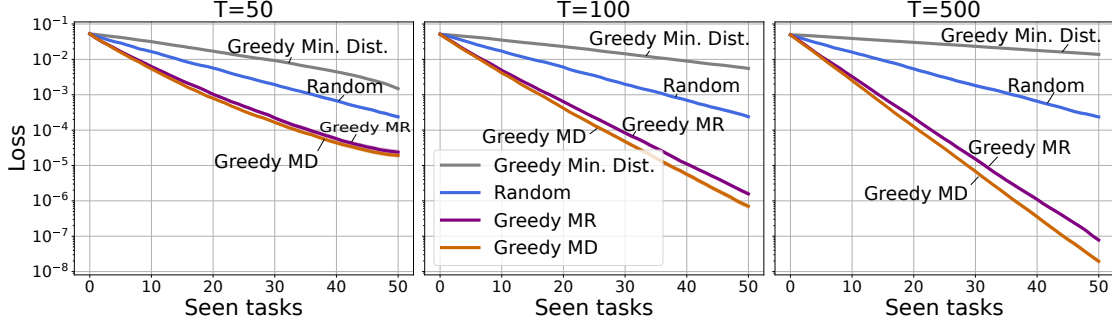


Figure 3.9: **Comparing orderings for varying task count T , for isotropic data.** $d = 100$, $r = 10$. Dissimilarity-based greedy strategies become more effective as the number of tasks increases, while Greedy-Min-Dist becomes less effective. We can give a heuristic explanation for the improvement to greedy strategies: In an isotropic setting (so task directions are sampled uniformly), increasing the number of tasks increases the coverage of the unit sphere. This results in a higher probability of encountering task pairs with large angular separation between their solution subspaces, which greedy ordering utilizes.

3.8.2 Anisotropic data

The following experiments were performed with anisotropic data, sampled from a Gaussian distribution with exponentially decaying eigenvalues, as detailed in scheme 3.2, resulting in high task correlation. This arises because tasks tend to align with the dominant eigen-directions, leading to strong pairwise similarity.

Scheme 3.2 (Generating tasks with high correlation).

Input: Input dimension d , task rank r , number of tasks T , and edge eigenvalues $\lambda_1 = 10^{-3}$ and $\lambda_d = 10^3$:

1. Sample $\mathbf{A} \sim \mathcal{N}(0, 1)^{d \times d}$ and symmetrize: $\mathbf{A}_{\text{sym}} \leftarrow \frac{1}{2}(\mathbf{A} + \mathbf{A}^\top)$.
2. Compute SVD: $\mathbf{A}_{\text{sym}} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$.
3. Define diagonal spectrum:

$$\Lambda \leftarrow \text{diag} \left(\lambda_1 \exp \left(\ln(\lambda_d/\lambda_1) \frac{i}{d-1} \right) \right)_{i=0}^{d-1}.$$

4. Construct covariance: $\Sigma \leftarrow \mathbf{U}\Lambda\mathbf{U}^\top$.

5. For $t = 1, \dots, T$:

(a) Sample $\mathbf{Z}_t \sim \mathcal{N}(0, 1)^{r \times d}$.

(b) Set $\mathbf{X}_t \leftarrow \mathbf{Z}_t \Sigma^{1/2}$.

Output: $\{\mathbf{X}_t\}_{t=1}^T$.

figs. 3.10 and 3.11 below reveal some trends compared to the isotropic case.

3.8.3 A note on statistical significance

All figures in this section 3.8 except fig. 3.12 include standard error bands of ± 1 standard error, although these are often too narrow to be visible. While different task collections introduce slight variations in outcomes, the overall trends are highly consistent. This is illustrated in the following figure, where we replicate the plot from fig. 3.3, overlaying individual runs from all 10 repeated experiments. Despite some run-to-run variability, the standard error remains small at only 10 runs, so the standard deviation is small, reinforcing the robustness of our qualitative conclusions.

3.9 Appendix to section 3.4.1: Classification experiments

3.9.1 Code

The code for the classification experiments with CIFAR-100 is available at https://github.com/matants/greedy_ordering.

Computational resources. All classification experiments were completed within a month on four NVIDIA GeForce GTX 1080 Ti GPUs.

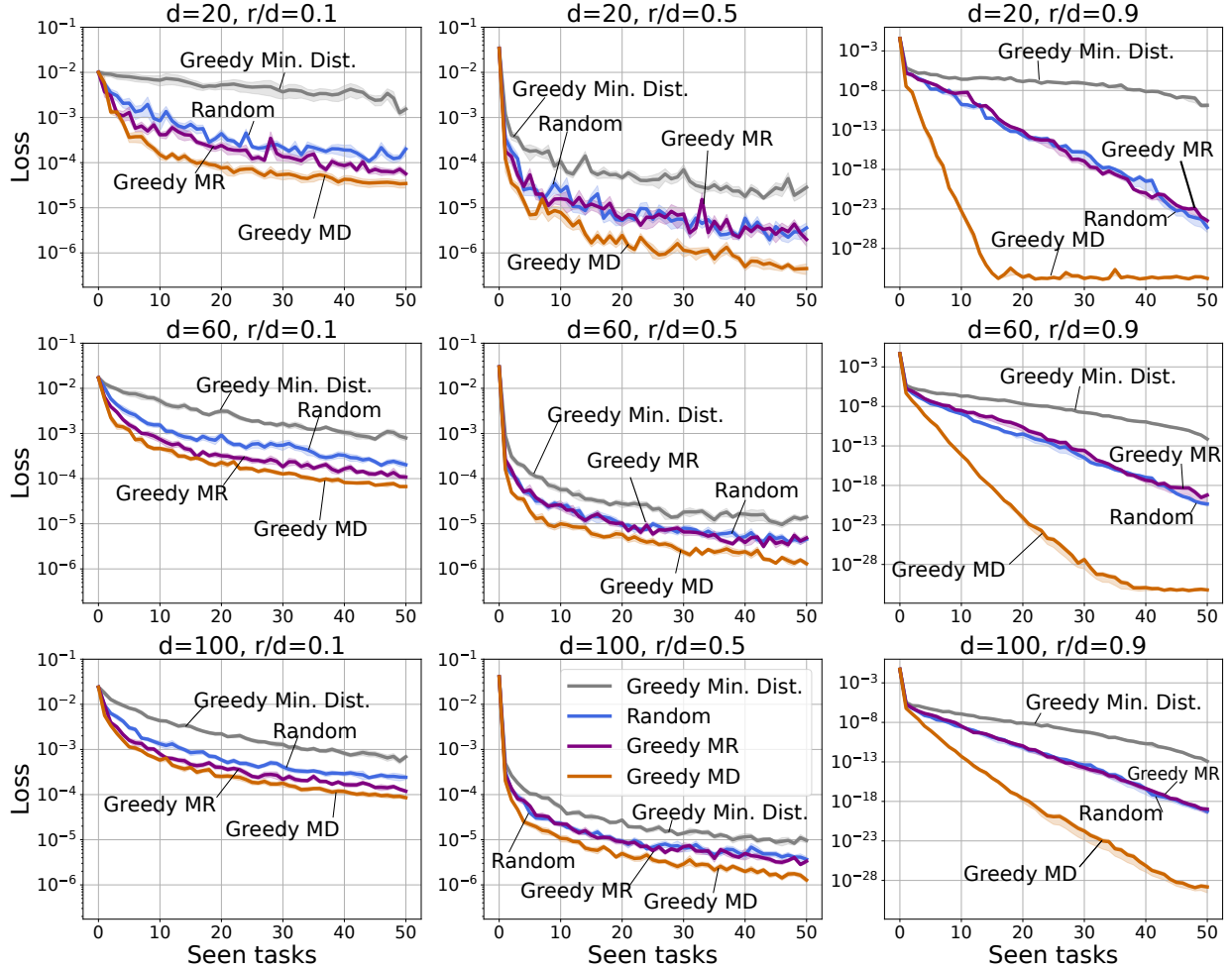


Figure 3.10: **Comparing orderings for varying dimensions d and ranks r of the data matrices, for anisotropic data.** $T = 50$. Compared to the isotropic case (fig. 3.8), we observe slower rates for all strategies. This may be related to how the pairwise distances between task solution subspaces are smaller, due to the higher correlation in the anisotropic case, though such a causation was not experimentally tested. As rank increases (focusing on a single row in the grid), the Maximum Residual (MR; definition 3.8) ordering underperforms and aligns with the random one very closely. We do not have an explanation of this alignment. The underperformance may be related to the combination of high rank and strong intra-task correlation, which may lead to ill-conditioned data matrices (for each task). In such a case, small perturbations, or steps, in the solution space may cause disproportionately large changes in residuals, which may cause MR to select tasks with large residuals that advance the iterate only marginally toward the intersection ($\mathbf{w}_*$).

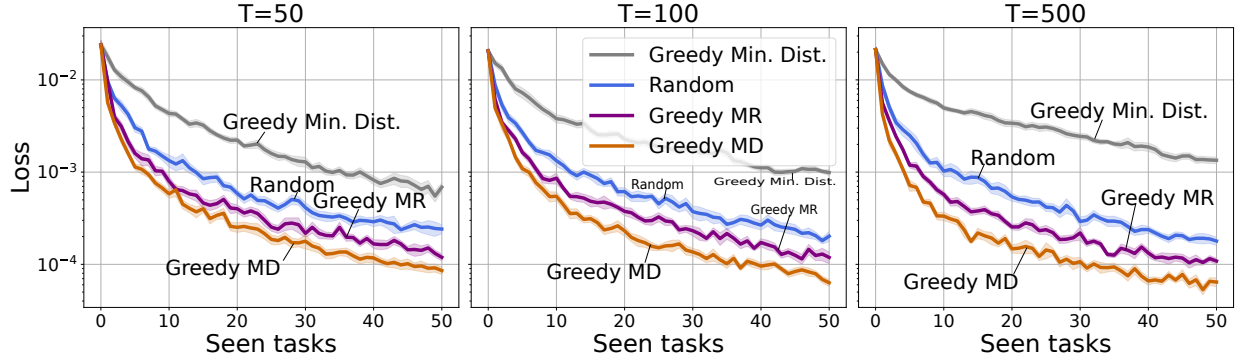


Figure 3.11: **Comparing orderings for varying task count T , for anisotropic data.** $d = 100$, $r = 10$. Compared to the isotropic case (fig. 3.9), the benefit to greedy orderings from increasing the number of tasks T is much smaller. The harm to Greedy min dist is also smaller than it was in the isotropic case. We can heuristically explain this as follows: In the anisotropic case, a large number of tasks must be added to induce the substantial “angles” that greedy orderings can exploit. Thus, while more tasks does cover more directions, it covers the space of directions more slowly in the anisotropic case.

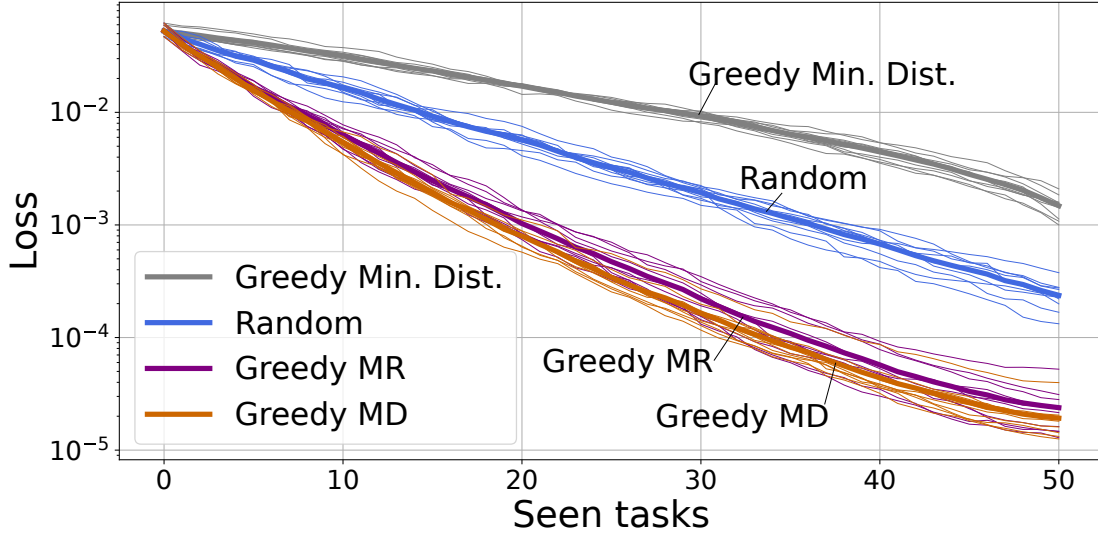


Figure 3.12: **Showing variations across different experiments.** Same as fig. 3.3, we have $T = 50$, $r = 10$, $d = 100$, with random isotropic data. Light lines represent each individual experiment. While minor variations exist across experiments, the low standard error at only 10 runs confirms the consistency of the results. We did not test whether, for fixed data, Greedy MD tends to beat Greedy MR, even though this appears true for the averages.

3.9.2 Experiment details

Model: Linear probing on pretrained ResNet-20

Our experiments employ a frozen pretrained ResNet-20 classifier [56], where the final classification head was removed and replaced with a binary classification head that we train [3, 37].

Tasks and benchmarks

We employ three benchmarks of domain-incremental CIFAR-100-based binary classification:

- (A) We use a model pretrained on CIFAR-100 [68] taken from Chen [25], which achieves 68.83% top-1 classification accuracy on CIFAR-100 multiclass classification according to Chen [25]. The continual learning tasks are composed of CIFAR-100 classes split into 50 pairs of binary classification tasks, with each pair having one task labeled 0 and one task labeled 1. The splitting procedure can be found in the code and does not respect superclass labels (that is, two fine classes from the same CIFAR-100 superclass do not necessarily receive the same 0/1 label)¹⁰. This is the setting used in all experiments unless stated otherwise, including the results presented in fig. 3.3.
- (B) We partition the 500 training samples of each CIFAR-100 class into two distinct groups of 250 samples, and use one of the groups to train the ResNet-20 embedder on the original CIFAR-100 multiclass task, using the same training recipe as Chen [25] and achieving 61.57% top-1 classification accuracy on the CIFAR-100 test set after 200 training epochs. The partitioning and training code is included in our provided repository. We then employ a linear probe on top of the resulting model (with the classification head removed), and construct the continual learning tasks using the 250 samples per class

¹⁰We originally intended to randomly split the 20 superclasses into two groups of 10, give one group the label 0 and the other the label 1, then randomly pair one fine label per group. However, we incorrectly assumed that the fine labels were grouped by superclass, so the first five fine labels form one superclass, etc. This is incorrect: In CIFAR-100, the fine labels are ordered alphabetically, independently of superclass.

that were not used for training the embedder. The class pairs and binary target labels defining the continual learning tasks are the same as in (A).

- (C) Using a model pretrained on CIFAR-10, taken from Chen [25], with the same CIFAR-100-based tasks as (A).

All presented results used 25 randomly generated task sets.

Training

Training was performed with cross-entropy loss on the softmax of the classifier’s output with label smoothing of 0.05 [120], with additive L2 regularization towards the previous parameters controlled by the hyperparameter λ , which is common in continual learning and is necessary to facilitate the projections view, as shown in Evron et al. [42]. In fig. 3.3, $\lambda = 5$, and we present an ablation study for λ in section 3.9.4, to address how it affects the performance of different ordering rules.

For each task we used the SGD optimizer with a learning rate of 0.01 and we used `ReduceLROnPlateau` on epoch losses, trained for 40 epochs with a batch size of 64. As a baseline, we jointly trained a classifier on all tasks together, without regularization.

Evaluation

We evaluate the performance of each ordering by calculating the average test (generalization) loss of all tasks after each seen task. Results are presented with 95% confidence intervals, calculated over the different randomly generated task sets, and over the permutations as well for random ordering.

Ordering computation

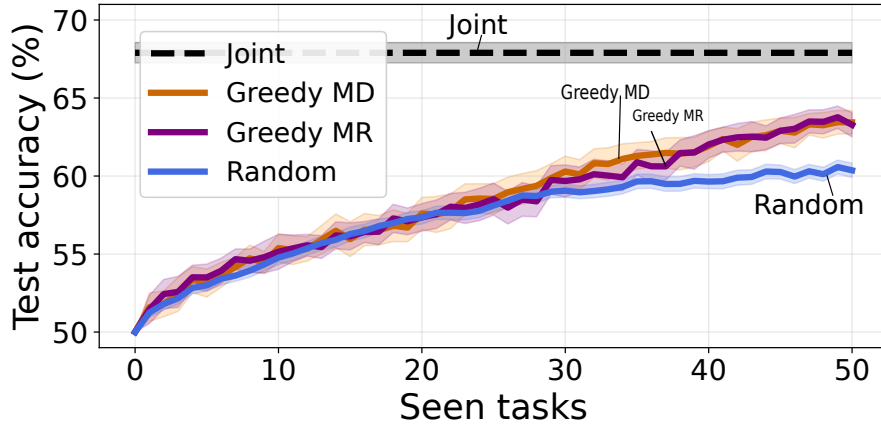
For the Random rule (eq. (3.4)), we use random sampling *without* replacement from the task set, unless stated otherwise. When presenting results for it, we use four random task

permutations per task set. The Greedy MR rule (definition 3.8) requires calculating the loss (without regularization) of all tasks after each task training, and choosing the task with the maximal loss. In section 3.9.6 we show its performance does not degrade when the losses are evaluated on a fraction of the dataset, as small as 1% of the data—5 samples per class in CIFAR-100. The Greedy MD rule (definition 3.7) was calculated by performing full training on each task, as elaborated above (section 3.9.2)—choosing the task that resulted in model parameters farthest from the current model parameters in terms of Euclidean distance. Although computing the MD rule requires training on each candidate task, the Greedy MR rule requires only forward passes to evaluate candidate-task losses and achieves comparable empirical performance.

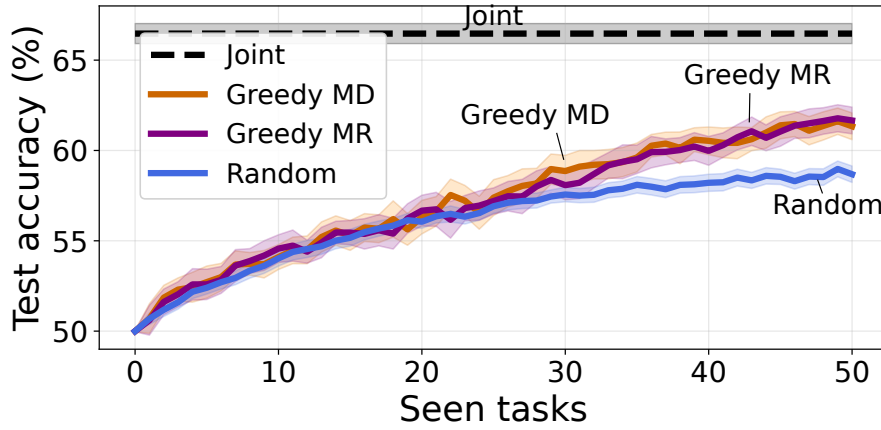
3.9.3 Out-of-domain feature extractors

To evaluate how our proposed method extends to more general transfer learning settings, we employ multiple benchmarks as elaborated in section 3.9.2.

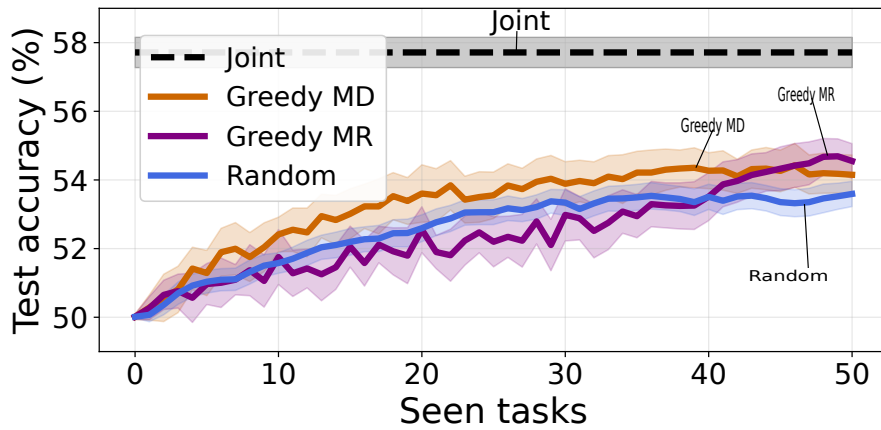
- (A) fig. 3.13 (a) is the same figure as fig. 3.3, shown here for completeness.
- (B) As shown in fig. 3.13 (b), this transfer-learning setting behaves similarly to the case where tasks are drawn from the training data (fig. 3.3), though with slightly weaker performance for both the joint baseline and all orderings.
- (C) As shown in fig. 3.13 (c), dissimilarity-guided orderings still outperform random orderings, though less prominently than in other experiments. This may be related to the lower joint baseline, suggesting that the pretrained initialization behaves more unlike our joint-realizability assumption (assumption 3.1).



(a) Pretraining: Full CIFAR-100; continual learning: Full CIFAR-100.



(b) Pretraining: Partitioned CIFAR-100; continual learning: Remaining CIFAR-100.



(c) Pretraining: CIFAR-10; continual learning: CIFAR-100.

Figure 3.13: Comparison of orderings under different pretraining setups.

3.9.4 Regularization ablation study

Regularization toward previous model parameters is a standard method to mitigate catastrophic forgetting [5, 66, 72], and is crucial for a projections view to emerge in continual classification [42]. Because of its central role, we perform an ablation study on the effect of regularization strength for completeness. As shown in fig. 3.14, without regularization our continual learning scheme collapses across all orderings, and greedy methods consistently outperform random ordering across all strengths examined. Interestingly, the optimal performance for greedy orderings occurs at smaller regularization strengths (λ) than for random ordering. This makes sense: greedy methods deliberately select tasks that push parameters further from their current values, so the regularization term has a stronger influence on the loss, requiring smaller λ for optimal effect.

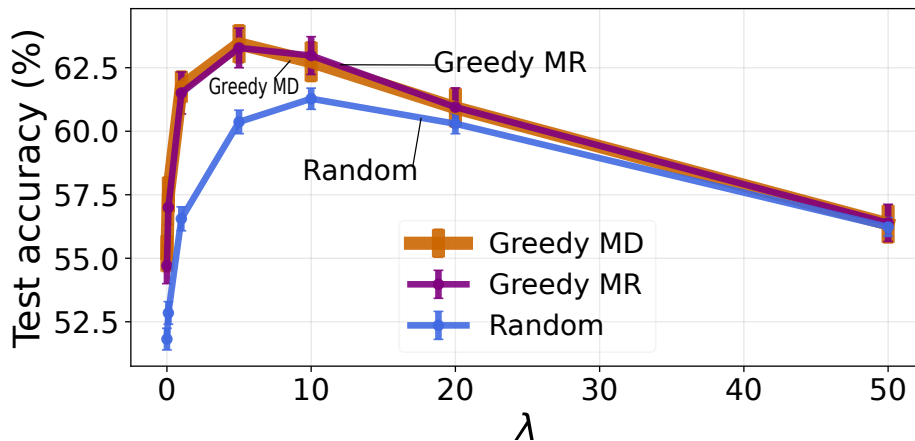


Figure 3.14: Regularization strength ablation study.

3.9.5 Allowing task repetition

As discussed in section 3.5.2, in continual linear regression, allowing repetitions helps greedy orderings avoid failure modes and guarantees provable convergence. In our classification experiments, however, repetitions do not improve performance: as shown in fig. 3.15, where each scheme could select from all tasks for 100 iterations (instead of 50), repetitions actually

harm the performance of greedy orderings, effectively canceling their advantage over random orderings. We observe a similar phenomenon in linear regression with anisotropic low-rank data, as detailed in section 3.12.2.

In the classification case, which departs substantially from jointly realizable linear regression, multiple factors could underlie this behavior. It would be interesting to examine how this relates to the connection between greedy ordering and “periphery-to-core” ordering [75], which may break down when repetitions are allowed, or to the distance-to-teacher perspective explored in section 3.12.2 (fig. 3.23). As this lies beyond the scope of our paper, we leave it for future work.

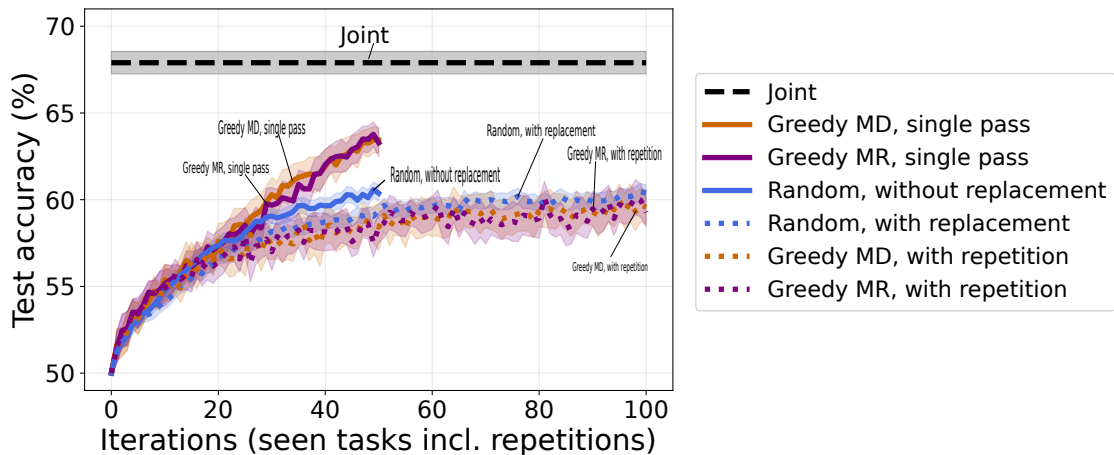


Figure 3.15: The effect of task repetition on CIFAR-100 continual classification.

3.9.6 Rule calculation with partial data

To assess practicality, we evaluate the more efficient Greedy MR method, which requires only forward passes, using fractions of the data and compute. Even with just 1% of the data (5 samples per CIFAR-100 class, i.e., 10 per binary task) to compute each task’s loss, Greedy MR maintains its performance and remains stronger than random ordering.

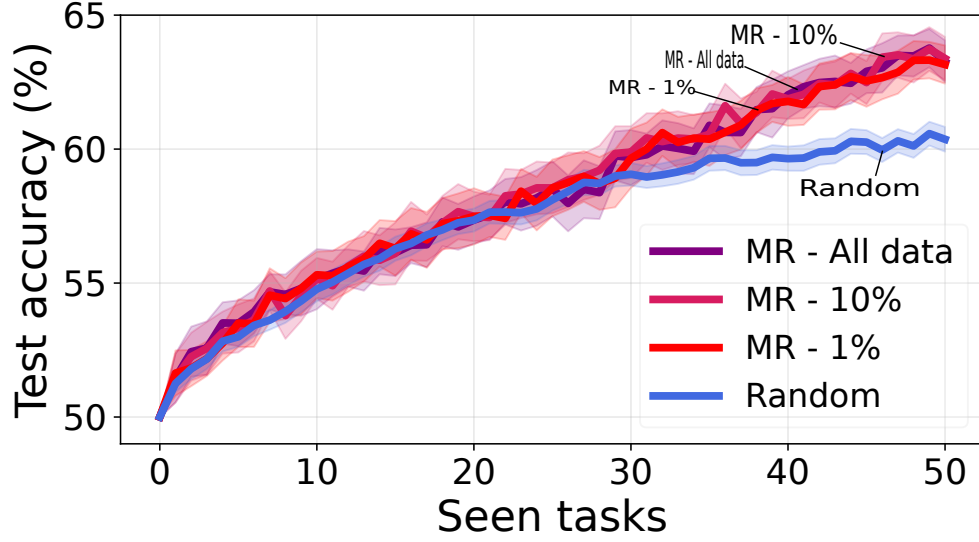


Figure 3.16: Greedy MR rule calculation using partial data.

3.10 Appendix to section 3.4.2: Proofs for “nearly determined” tasks

3.10.1 Optimality guarantee when $r = d - 1$

Recall Lemma 3.9. Let $\mathbf{w}_T^{\tau_{\text{MD}}}$ and $\mathbf{w}_T^{\tau_\star}$ be the iterates after learning T jointly realizable tasks of rank $d - 1$ under the Maximum Distance ordering τ_{MD} and an optimal ordering $\tau_\star$ that leads to a minimal distance to the joint solution $\mathbf{w}_\star$. Then

$$0 \leq D^2(\mathbf{w}_T^{\tau_\star}) \leq D^2(\mathbf{w}_T^{\tau_{\text{MD}}}) \triangleq \frac{\|\mathbf{w}_T^{\tau_{\text{MD}}} - \mathbf{w}_\star\|^2}{\|\mathbf{w}_\star\|^2} \leq \frac{\|\mathbf{w}_T^{\tau_\star} - \mathbf{w}_\star\|}{\|\mathbf{w}_\star\|} \triangleq D(\mathbf{w}_T^{\tau_\star}) \leq 1.$$

Proof. The distance at the end of an ordering τ is

$$\begin{aligned} D^2(\mathbf{w}_T^\tau) &\triangleq \frac{\|\mathbf{w}_T^\tau - \mathbf{w}_\star\|^2}{\|\mathbf{w}_\star\|^2} = \frac{1}{\|\mathbf{w}_\star\|^2} \|\mathbf{v}_{\tau(T)}^\top \mathbf{v}_{\tau(T)}^\top \cdots \mathbf{v}_{\tau(1)}^\top \mathbf{v}_{\tau(1)}^\top (\mathbf{w}_0 - \mathbf{w}_\star)\|^2 \\ &= \frac{1}{\|\mathbf{w}_\star\|^2} (\mathbf{v}_{\tau(1)}^\top (\mathbf{w}_0 - \mathbf{w}_\star))^2 \cdot \prod_{t=1}^{T-1} (\mathbf{v}_{\tau(t)}^\top \mathbf{v}_{\tau(t+1)})^2. \end{aligned}$$

Let $\tau = \tau_{\text{MD}}, \tau_*$ be the greedy MD ordering and an optimal ordering leading to the minimal distance (respectively). Let $c(i, j) = \begin{cases} \frac{1}{\|\mathbf{w}_*\|^2} (\mathbf{v}_j^\top (\mathbf{w}_0 - \mathbf{w}_*))^2 & i = 0, j \in [T] \\ (\mathbf{v}_i^\top \mathbf{v}_j)^2 & i, j \in [T] \end{cases}$.

Then

$$\begin{aligned} D^2(\mathbf{w}_T^{\tau_*}) &= \frac{1}{\|\mathbf{w}_*\|^2} (\mathbf{v}_{\tau_*(1)}^\top (\mathbf{w}_0 - \mathbf{w}_*))^2 \cdot \prod_{t=1}^{T-1} (\mathbf{v}_{\tau_*(t)}^\top \mathbf{v}_{\tau_*(t+1)})^2 \\ &= c(0, \tau_*(1)) \prod_{t=1}^{T-1} c(\tau_*(t), \tau_*(t+1)) \\ &= c(0, \tau_*(1)) \prod_{t \in \mathcal{C}} c\left(\tau(\tau^{-1}(\tau_*(t))), \tau_*(t+1)\right) \cdot \prod_{t \notin \mathcal{C}} c\left(\tau_*(t), \tau(\tau^{-1}(\tau_*(t+1)))\right), \end{aligned}$$

where we define the index set $\mathcal{C} = \{t \mid 1 \leq t \leq T-1, \tau^{-1}(\tau_*(t)) < \tau^{-1}(\tau_*(t+1))\}$.

By greediness,

$$\begin{aligned} D^2(\mathbf{w}_T^{\tau_*}) &\geq c(0, \tau(1)) \underbrace{\prod_{t \in \mathcal{C}} c\left(\tau(\tau^{-1}(\tau_*(t))), \tau(1 + \tau^{-1}(\tau_*(t)))\right)}_{\text{here, } \tau^{-1}(\tau_*(t)) < T} \\ &\quad \cdot \underbrace{\prod_{t \notin \mathcal{C}} c\left(\tau(1 + \tau^{-1}(\tau_*(t+1))), \tau(\tau^{-1}(\tau_*(t+1)))\right)}_{\text{here, } \tau^{-1}(\tau_*(t+1)) < T}. \end{aligned}$$

Then, since $\tau^{-1}(\tau_*(\cdot))$ “covers” $[T]$ and $c(i, j) \leq 1$, iterating over $1, \dots, T-1$ adds elements to the product that cannot make it larger. That is,

$$\begin{aligned} D^2(\mathbf{w}_T^{\tau_*}) &\geq c(0, \tau(1)) \cdot \prod_{\ell=1}^{T-1} c(\tau(\ell), \tau(1 + \ell)) \cdot \prod_{\ell=1}^{T-1} c(\tau(1 + \ell), \tau(\ell)) \\ &\geq (c(0, \tau(1)) \prod_{\ell=1}^{T-1} c(\tau(\ell), \tau(1 + \ell)))^2 \\ &= \left(\frac{1}{\|\mathbf{w}_*\|^2} (\mathbf{v}_{\tau(1)}^\top (\mathbf{w}_0 - \mathbf{w}_*))^2 \prod_{t=1}^{T-1} (\mathbf{v}_{\tau(t)}^\top \mathbf{v}_{\tau(t+1)})^2 \right)^2 = (D^2(\mathbf{w}_T^{\tau_*}))^2 \\ &\Rightarrow 1 \geq D(\mathbf{w}_T^{\tau_*}) \geq D^2(\mathbf{w}_T^{\tau_{\text{MD}}}) \geq D^2(\mathbf{w}_T^{\tau_*}) \geq 0. \end{aligned}$$

□

3.10.2 Loss bound when $r = d - 1$

Recall Lemma 3.10. For the Maximum Distance greedy ordering over T jointly-realizable tasks of rank $d - 1$, the loss of scheme 3.1 after T iterations is upper bounded as

$$\mathcal{L}(\mathbf{w}_T) = \frac{1}{\|\mathbf{w}_\star\|^2 R^2} \cdot \frac{1}{T} \sum_{m=1}^T \|\mathbf{X}_m \mathbf{w}_T - \mathbf{y}_m\|^2 \leq \frac{1}{eT}.$$

Proof. We aim to bound the average loss using projection matrices,

$$\begin{aligned} \mathcal{L}_{\tau_{\text{MD}}}(\mathbf{w}_T) &= \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m \mathbf{w}_T - \mathbf{y}_m\|^2 = \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2 \\ &= \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m \mathbf{X}_m^+ \mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2 \\ &\leq \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m\|^2 \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_T - \mathbf{w}_\star)\|^2 \\ [\text{eq. (3.2)}] &\leq \frac{1}{\|\mathbf{w}_\star\|^2 T} \sum_{t=1}^T \left\| (\mathbf{I} - \mathbf{P}_{\tau(t)}) \prod_{s=1}^t \mathbf{P}_{\tau(s)} (\mathbf{w}_0 - \mathbf{w}_\star) \right\|^2. \end{aligned}$$

Since each task matrix $\mathbf{X}_i$ has rank $d - 1$, each projection $\mathbf{P}_i$ is rank 1 and can be written as $\mathbf{P}_i = \mathbf{v}_i \mathbf{v}_i^\top$ for a unit vector $\mathbf{v}_i$. Substituting this and $\mathbf{v}_{\tau(0)} = \frac{1}{\|\mathbf{w}_\star\|} (\mathbf{w}_0 - \mathbf{w}_\star)$, the bound becomes:

$$\begin{aligned} \mathcal{L}_{\tau_{\text{MD}}}(\mathbf{w}_T) &\leq \frac{1}{T} \sum_{t=1}^T \left\| (\mathbf{I} - \mathbf{v}_{\tau(t)} \mathbf{v}_{\tau(t)}^\top) \mathbf{v}_{\tau(t)} \mathbf{v}_{\tau(t)}^\top \cdots \mathbf{v}_{\tau(1)} \mathbf{v}_{\tau(1)}^\top \mathbf{v}_{\tau(0)} \right\|^2 \\ &\leq \underbrace{(\mathbf{v}_{\tau(1)}^\top \mathbf{v}_{\tau(0)})^2}_{\leq 1} \frac{1}{T} \sum_{t=1}^T \left\| (\mathbf{I} - \mathbf{v}_{\tau(t)} \mathbf{v}_{\tau(t)}^\top) \mathbf{v}_{\tau(t)} \right\|^2 \prod_{s=1}^{t-1} (\mathbf{v}_{\tau(s+1)}^\top \mathbf{v}_{\tau(s)})^2 \\ [\text{projection properties}] &\leq \left(1 - \frac{1}{T} \sum_{s=1}^T (\mathbf{v}_{\tau(s)}^\top \mathbf{v}_{\tau(s)})^2 \right) \prod_{s=1}^{T-1} (\mathbf{v}_{\tau(s+1)}^\top \mathbf{v}_{\tau(s)})^2. \end{aligned}$$

Then, we use algebraic and projection properties to rewrite the greedy ordering as

$$\begin{aligned}
\tau_{\text{MD}}(t) &= \operatorname{argmax}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \\
&= \operatorname{argmax}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} (\|\mathbf{w}_{t-1} - \mathbf{w}_\star\|^2 - \|\mathbf{P}_m(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2) \\
&= \operatorname{argmin}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} \|\mathbf{P}_m(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \\
&= \operatorname{argmin}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} \|\mathbf{v}_m \mathbf{v}_m^\top \mathbf{v}_{\tau(t-1)} \mathbf{v}_{\tau(t-1)}^\top (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \\
&= \operatorname{argmin}_{m \in [T] \setminus \tau_{\text{MD}}(1:t-1)} (\mathbf{v}_m^\top \mathbf{v}_{\tau(t-1)})^2.
\end{aligned} \tag{3.7}$$

Then, employing greediness as reformulated above and inequality of arithmetic and geometric mean, we obtain:

$$\prod_{s=1}^{T-1} (\mathbf{v}_{\tau(s+1)}^\top \mathbf{v}_{\tau(s)})^2 \leq \prod_{s=1}^T (\mathbf{v}_{\tau(T)}^\top \mathbf{v}_{\tau(s)})^2 \leq \left(\frac{1}{T} \sum_{s=1}^T (\mathbf{v}_{\tau(T)}^\top \mathbf{v}_{\tau(s)})^2 \right)^T.$$

Substituting back into the forgetting, it is now bounded as,

$$\mathcal{L}_{\tau_{\text{MD}}}(\mathbf{w}_T) \leq \left(1 - \frac{1}{T} \sum_{s=1}^T (\mathbf{v}_{\tau(T)}^\top \mathbf{v}_{\tau(s)})^2 \right) \left(\frac{1}{T} \sum_{s=1}^T (\mathbf{v}_{\tau(T)}^\top \mathbf{v}_{\tau(s)})^2 \right)^T \leq \frac{1}{eT},$$

where we invoked an algebraic property that $(1-x)x^T \leq \frac{1}{eT}, \forall x \in [0, 1]$. $\square$

3.11 Appendix to section 3.5.1: Lower bound’s “adversarial” constructions

3.11.1 General dimension construction and proof (theorem 3.12)

Recall Theorem 3.12. For any $d \geq 30$, there exists an adversarial task collection with $T = d - 1$ jointly-realizable tasks of different ranks such that both greedy orderings (MD,

MR) forget catastrophically. That is, the loss at the end of the sequence satisfies

$$\mathcal{L}(\mathbf{w}_T^{\tau_{\text{MD}}}), \mathcal{L}(\mathbf{w}_T^{\tau_{\text{MR}}}) \geq \frac{1}{8} - \frac{1}{4d}.$$

Proof outline. For a given dimension d , we construct a sequence of d iterates $(\mathbf{w}_t)_{t=1}^d$, corresponding to $T = d - 1$ tasks $(\mathbf{X}_t)_{t=2}^d$ of decreasing rank, which are jointly-realizable with $\mathbf{w}_\star = \mathbf{0}$ (i.e., $\mathbf{y}_t = \mathbf{0}$ for all $t \in \{2 \dots T\}$) with the following properties:

1. Given this specific choice of tasks and matching iterates, the loss (equivalently, forgetting) satisfies the lower bound stated in the theorem.
2. The chosen iterates are valid—i.e., they can be obtained from a specific selection rule given the constructed task collection.
3. The chosen ordering adheres to greedy selection rules, both MD and MR, under the chosen tasks.

For notational convenience, the construction indexes iterates from $t = 1$ and tasks from $t = 2$. We set $\mathbf{w}_\star = \mathbf{0}$ and start from $\mathbf{w}_1 = \mathbf{e}_1$. The same construction holds for a shifted frame of reference where all iterates (and $\mathbf{w}_\star$) are shifted by $-\mathbf{e}_1$.

Construction details

We define the iterates as follows:

$$\mathbf{w}_1 = \mathbf{e}_1 = \left[1, \underbrace{0, \dots, 0}_{d-1 \text{ times}} \right]^\top,$$

$$\forall t \in \{2 \dots d\} : \mathbf{w}_t = \left[\frac{(\mathbf{w}_{t-1})_1 + \sqrt{(\mathbf{w}_{t-1})_1^2 - 4\beta_t}}{2}, \underbrace{c^{t-2} \frac{1}{\sqrt{d}}, \dots, c^{t-2} \frac{1}{\sqrt{d}}}_{t-1 \text{ times}}, 0, \dots, 0 \right]^\top,$$

where $c \triangleq 2^{-1/d}$ and $\beta_t \triangleq \frac{((t-1)c - (t-2))c^{2t-5}}{d}$.

$x_t \triangleq (\mathbf{w}_t)_1$ is defined recursively by

$$x_1 = 1, \quad x_t = \frac{x_{t-1} + \sqrt{x_{t-1}^2 - 4\beta_t}}{2} \quad \forall t \in \{2 \dots d\}.$$

Since $\mathbf{w}_t \neq \mathbf{w}_{t-1}$, define the unit vector

$$\mathbf{u}_t = \frac{\mathbf{w}_t - \mathbf{w}_{t-1}}{\|\mathbf{w}_t - \mathbf{w}_{t-1}\|} \in \text{span}(\mathbf{e}_1, \dots, \mathbf{e}_t).$$

Define the tasks by

$$\mathbf{X}_t = \begin{bmatrix} -\mathbf{u}_t^\top - \\ -\mathbf{e}_{t+1}^\top - \\ \vdots \\ -\mathbf{e}_d^\top - \end{bmatrix} = \begin{bmatrix} -\mathbf{u}_t^\top - \\ \mathbf{I}_{t+1:d} \end{bmatrix} \in \mathbb{R}^{(d-t+1) \times d}, \quad \forall t \in \{2 \dots d\}.$$

Then $\mathbf{P}_t \triangleq \mathbf{I}_d - \mathbf{X}_t^+ \mathbf{X}_t = \mathbf{I}_d - \mathbf{I}_{t+1:d} - \mathbf{u}_t \mathbf{u}_t^\top = \underbrace{\mathbf{I}_t}_{\text{rank } t} - \mathbf{u}_t \mathbf{u}_t^\top$.

Lower bounding the loss

For each task $\mathbf{X}_m$, its individual loss at time $t = d$ is

$$\begin{aligned} \mathcal{L}_m(\mathbf{w}_d) &\triangleq \|\mathbf{X}_m \mathbf{w}_d\|^2 = \left\| \begin{bmatrix} -\mathbf{u}_m^\top - \\ \mathbf{I}_{m+1:d} \end{bmatrix} \mathbf{w}_d \right\|^2 = (\mathbf{u}_m^\top \mathbf{w}_d)^2 + \|\mathbf{I}_{m+1:d} \mathbf{w}_d\|^2 \\ &\geq \|\mathbf{I}_{m+1:d} \mathbf{w}_d\|^2 = \sum_{j=m+1}^d (\mathbf{w}_d)_j^2 \\ [j \geq 2] &= (d-m) \frac{c^{2d-4}}{d} = \left(1 - \frac{m}{d}\right) c^{2d-4} = \left(1 - \frac{m}{d}\right) 2^{-(2d-4)/d} \\ &= \frac{1}{4} \left(1 - \frac{m}{d}\right) 2^{4/d} \geq \frac{1}{4} \left(1 - \frac{m}{d}\right). \end{aligned}$$

The average loss after all iterates, which coincides with the forgetting (see remark 3.4), is

$$\begin{aligned}
\mathcal{L}(\mathbf{w}_d) &= \frac{1}{T} \sum_{m \in \{2 \dots d\}} \mathcal{L}_m(\mathbf{w}_d) = \frac{1}{d-1} \sum_{m=2}^d \mathcal{L}_m(\mathbf{w}_d) \\
&\geq \frac{1}{4(d-1)} \sum_{m=2}^d \left(1 - \frac{m}{d}\right) = \frac{1}{4(d-1)} \left(d-1 - \frac{\sum_{m=2}^d m}{d}\right) \\
&= \frac{1}{4} - \frac{d+2}{8d} = \frac{1}{8} - \frac{1}{4d}.
\end{aligned}$$

Proving that the iterates and tasks exist

Lemma 3.46 proves that for all $d \geq 30$, $t \in \{2, \dots, d\}$, $x_{t-1}^2 - 4\beta_t \geq 0$, so the square root in the recursive definition of x_t (section 3.11.1) exists.

Proving that the iterates can be formed from projections of the given tasks

We first do a sanity check: $\mathbf{P}_t$ is real symmetric, and we can verify its idempotence as follows:

$$\begin{aligned}
\mathbf{P}_t^2 &= (\mathbf{I}_t - \mathbf{u}_t \mathbf{u}_t^\top)^2 = \mathbf{I}_t^2 - \mathbf{u}_t \mathbf{u}_t^\top \mathbf{I}_t - \mathbf{I}_t \mathbf{u}_t \mathbf{u}_t^\top + \mathbf{u}_t \mathbf{u}_t^\top \mathbf{u}_t \mathbf{u}_t^\top \\
&= \mathbf{I}_t - \mathbf{u}_t \mathbf{u}_t^\top - \mathbf{u}_t \mathbf{u}_t^\top + \mathbf{u}_t \mathbf{u}_t^\top = \mathbf{I}_t - \mathbf{u}_t \mathbf{u}_t^\top = \mathbf{P}_t.
\end{aligned}$$

The orthogonality condition gives $\mathbf{w}_t^\top (\mathbf{w}_t - \mathbf{w}_{t-1}) = 0$:

$$\begin{aligned}
\mathbf{w}_t^\top (\mathbf{w}_t - \mathbf{w}_{t-1}) &= \sum_{i=1}^d (\mathbf{w}_t)_i^2 - \sum_{i=1}^d (\mathbf{w}_t)_i (\mathbf{w}_{t-1})_i = (\mathbf{w}_t)_1^2 + \sum_{i=2}^t (\mathbf{w}_t)_i^2 - \sum_{i=1}^{t-1} (\mathbf{w}_t)_i (\mathbf{w}_{t-1})_i \\
&= (\mathbf{w}_t)_1^2 - (\mathbf{w}_t)_1 (\mathbf{w}_{t-1})_1 + \sum_{i=2}^t \frac{c^{2t-4}}{d} - \sum_{i=2}^{t-1} \frac{c^{t-2} c^{t-3}}{d} \\
&= (\mathbf{w}_t)_1^2 - (\mathbf{w}_t)_1 (\mathbf{w}_{t-1})_1 + \frac{(t-1)c^{2t-4} - (t-2)c^{2t-5}}{d} \\
&= (\mathbf{w}_t)_1^2 - (\mathbf{w}_t)_1 (\mathbf{w}_{t-1})_1 + \underbrace{\frac{((t-1)c - (t-2))c^{2t-5}}{d}}_{=\beta_t} \\
&= (\mathbf{w}_t)_1^2 - (\mathbf{w}_t)_1 (\mathbf{w}_{t-1})_1 + \beta_t,
\end{aligned}$$

The construction choice $(\mathbf{w}_t)_1 = \frac{(\mathbf{w}_{t-1})_1 + \sqrt{(\mathbf{w}_{t-1})_1^2 - 4\beta_t}}{2}$ satisfies

$$\mathbf{w}_t^\top (\mathbf{w}_t - \mathbf{w}_{t-1}) = 0.$$

The iterates are a sequence of the corresponding projections:

$$\begin{aligned} \mathbf{P}_t \mathbf{w}_{t-1} &= (\mathbf{I}_t - \mathbf{u}_t \mathbf{u}_t^\top) \mathbf{w}_{t-1} = \mathbf{I}_t \mathbf{w}_{t-1} - \mathbf{u}_t \mathbf{u}_t^\top \mathbf{w}_{t-1} \\ &= \mathbf{w}_{t-1} - \left(\frac{(\mathbf{w}_t - \mathbf{w}_{t-1})^\top}{\|\mathbf{w}_t - \mathbf{w}_{t-1}\|} \mathbf{w}_{t-1} \right) \mathbf{u}_t = \mathbf{w}_{t-1} - \frac{(\mathbf{w}_t - \mathbf{w}_{t-1})^\top \mathbf{w}_{t-1}}{\|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2} (\mathbf{w}_t - \mathbf{w}_{t-1}) \\ &= \mathbf{w}_{t-1} - \frac{(\mathbf{w}_t - \mathbf{w}_{t-1})^\top \mathbf{w}_{t-1} - \overbrace{(\mathbf{w}_t - \mathbf{w}_{t-1})^\top \mathbf{w}_t}^{=0}}{\|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2} (\mathbf{w}_t - \mathbf{w}_{t-1}) \\ &= \mathbf{w}_{t-1} + \frac{(\mathbf{w}_t - \mathbf{w}_{t-1})^\top (\mathbf{w}_t - \mathbf{w}_{t-1})}{\|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2} (\mathbf{w}_t - \mathbf{w}_{t-1}) \\ &= \mathbf{w}_{t-1} + (\mathbf{w}_t - \mathbf{w}_{t-1}) = \mathbf{w}_t. \end{aligned}$$

Proving that the iterates adhere to greedy ordering rules

Maximum Distance (MD). We wish to prove that the greedy MD rule agrees with the ordering we chose. That is,

$$\tau_t \triangleq \operatorname{argmax}_{t' \in [T] \setminus \{\tau_2, \dots, \tau_{t-1}\}} \|(\mathbf{I} - \mathbf{P}_{t'}) \mathbf{w}_{t-1}\|^2 = t.$$

We proceed by induction.

Assuming the rule holds for $\tau_2, \dots, \tau_{t-1}$ (with the same calculation for the base case

$t = 2$):

$$\begin{aligned}
\tau_t &\triangleq \operatorname{argmax}_{t' \in [T] \setminus \{\tau_2, \dots, \tau_{t-1}\}} \|(\mathbf{I} - \mathbf{P}_{t'}) \mathbf{w}_{t-1}\|^2 \\
[\text{induction assumption}] &= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \|(\mathbf{I}_d - \mathbf{P}_{t'}) \mathbf{w}_{t-1}\|^2 \\
&= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \|(\mathbf{I}_d - \mathbf{I}_{t'} + \mathbf{u}_{t'} \mathbf{u}_{t'}^\top) \mathbf{w}_{t-1}\|^2 \\
[t' > t - 1] &= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \left\| \underbrace{(\mathbf{I}_d - \mathbf{I}_{t'}) \mathbf{w}_{t-1}}_{=0} + \mathbf{u}_{t'} \mathbf{u}_{t'}^\top \mathbf{w}_{t-1} \right\|^2 \\
[\|\mathbf{u}_{t'}\|^2 = 1] &= \operatorname{argmax}_{t' \in \{t, \dots, T\}} (\mathbf{u}_{t'}^\top \mathbf{w}_{t-1})^2 \\
&= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \left(\frac{(\mathbf{w}_{t'} - \mathbf{w}_{t'-1})^\top}{\|\mathbf{w}_{t'} - \mathbf{w}_{t'-1}\|} \mathbf{w}_{t-1} \right)^2.
\end{aligned}$$

Maximum Residual (MR). We wish to prove that the greedy MR rule agrees with the ordering we chose. That is,

$$\tau_t \triangleq \operatorname{argmax}_{t' \in [T] \setminus \{\tau_2, \dots, \tau_{t-1}\}} \|\mathbf{X}_{t'} \mathbf{w}_{t-1}\|^2 = t.$$

We proceed by induction.

Assuming the rule holds for $\tau_2, \dots, \tau_{t-1}$ (with the same calculation for the base case

$t = 2$):

$$\begin{aligned}
\tau_t &\triangleq \operatorname{argmax}_{t' \in [T] \setminus \{\tau_2, \dots, \tau_{t-1}\}} \|\mathbf{X}_{t'} \mathbf{w}_{t-1}\|^2 \\
[\text{induction assumption}] &= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \|\mathbf{X}_{t'} \mathbf{w}_{t-1}\|^2 = \operatorname{argmax}_{t' \in \{t, \dots, T\}} \left\| \begin{bmatrix} -\mathbf{u}_{t'}^\top - \\ \mathbf{I}_{t'+1:d} \end{bmatrix} \mathbf{w}_{t-1} \right\|^2 \\
[t' > t-1] &= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \left((\mathbf{u}_{t'}^\top \mathbf{w}_{t-1})^2 + \underbrace{\|\mathbf{I}_{t'+1:d} \mathbf{w}_{t-1}\|^2}_{=0} \right) \\
&= \operatorname{argmax}_{t' \in \{t, \dots, T\}} \left(\frac{(\mathbf{w}_{t'} - \mathbf{w}_{t'-1})^\top}{\|\mathbf{w}_{t'} - \mathbf{w}_{t'-1}\|} \mathbf{w}_{t-1} \right)^2.
\end{aligned}$$

Thus, the MR and MD rules coincide in this case.

How we prove greediness holds: Delta positivity. It remains to show that, as a function of $k \geq t$, $\left(\frac{((\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1})^2}{\|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2} \right)_k$ decreases monotonically, which will prove that the iterates satisfy the greedy MD and MR rules in definitions 3.7 and 3.8.

The difference between consecutive iterates is

$$\mathbf{w}_{k-1} - \mathbf{w}_k = \left[x_{k-1} - x_k, \underbrace{\frac{c^{k-3}(1-c)}{\sqrt{d}}, \dots, \frac{c^{k-3}(1-c)}{\sqrt{d}}}_{k-2 \text{ times}}, -\frac{c^{k-2}}{\sqrt{d}}, 0, \dots, 0 \right].$$

For all $k \geq t$, $(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1}$ is positive because

$$(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} = \underbrace{(x_{k-1} - x_k)}_{>0, \text{ from 3.30}} \underbrace{x_{t-1}}_{>0} + (t-2) \underbrace{\frac{c^{k-3}(1-c)}{\sqrt{d}}}_{>0} \frac{c^{t-3}}{\sqrt{d}} > 0.$$

Thus, it suffices to show monotonic decrease for all $k \geq t$ of

$$\left(\frac{(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1}}{\|\mathbf{w}_{k-1} - \mathbf{w}_k\|} \right)_k.$$

So it suffices to show that the following quantity is positive for all $t \in \{2, \dots, d-1\}$ and

$k \in \{t, \dots, d-1\}$ (recall our convention that the first step is $t = 2$, and at the last step there is only one choice):

$$\begin{aligned} & \frac{(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1}}{\|\mathbf{w}_{k-1} - \mathbf{w}_k\|} - \frac{(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1}}{\|\mathbf{w}_k - \mathbf{w}_{k+1}\|} \\ & \propto \|\mathbf{w}_k - \mathbf{w}_{k+1}\| (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\| (\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} \triangleq \Delta_{t,k}. \end{aligned}$$

Next, we will show this holds numerically for $d < 25,000$ and prove it analytically for $d \geq 25,000$.

Showing delta positivity numerically for low dimensions. We use the following facts to write code that verifies $\Delta_{t,k} > 0$ for all $d < 25,000$, $t \in \{2, \dots, d-1\}$, and $k \in \{t, \dots, d-1\}$:

$$\begin{aligned} \|\mathbf{w}_{k-1} - \mathbf{w}_k\| &= \sqrt{(x_{k-1} - x_k)^2 + (k-2) \left(\frac{c^{k-3}(1-c)}{\sqrt{d}} \right)^2 + \left(\frac{c^{k-3}}{\sqrt{d}} \right)^2} \\ &= \sqrt{(x_{k-1} - x_k)^2 + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{1}{d} c^{2k-6}}, \\ (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} &= (x_{k-1} - x_k) x_{t-1} + (t-2) \frac{c^{k-3}(1-c)}{\sqrt{d}} \frac{c^{t-3}}{\sqrt{d}}. \end{aligned}$$

For each dimension d , we calculate the sequence $(x)_k$ using its recursive definition, and we calculate $\Delta(d) \triangleq \min_{\{t,k|t \in \{2,\dots,d-1\}, k \in \{t,\dots,d-1\}\}} \Delta_{t,k}$ using these formulas. Figure 3.17 shows that $\Delta(d)$ remains positive for all $d \in \{30\dots 47,000\}$ (dimensions above 25,000 are not needed for the numerical verification).

As will be seen analytically (eq. (3.13)), $\Delta(d)$ should correlate with $d^{-\frac{5}{2}}$. Our graphs below show that, numerically, this holds for lower dimensions as well by observing that $\Delta(d) \cdot d^{\frac{5}{2}}$ is approximately constant.

Computational resources. This numerical validation took 4 days to run on a home PC with i5-9400F CPU and 16GB RAM.

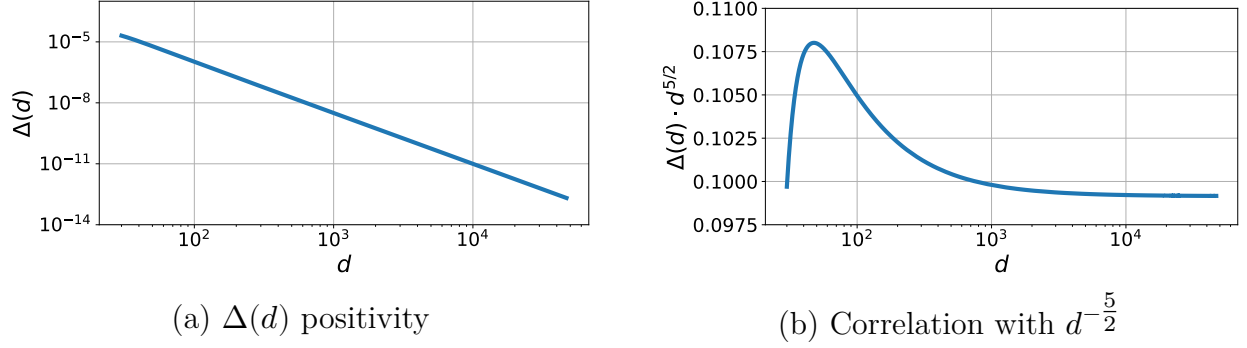


Figure 3.17: Numerical positivity of $\Delta(d) \triangleq \min_{\{t,k|t \in \{2,\dots,d-1\}, k \in \{t,\dots,d-1\}\}} \Delta_{t,k}$

Showing delta positivity analytically for high dimensions. Due to the length of this part, we defer it to section 3.15, where we prove that, for all $d \geq 25,000$, $t \in \{2, \dots, d-1\}$, and $k \in \{t, \dots, d-1\}$,

$$\Delta_{t,k} \triangleq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} > 0.$$

Conclusion. Together with the numerical verification, we have established that $\Delta_{t,k} > 0$ for all $k \geq t$ and all $d \geq 30$. This completes the proof that the iterates satisfy the greedy ordering rules and thereby the proof that the adversarial construction yields a lower bound on the loss under single-pass greedy orderings.

3.11.2 Adversarial 3d construction (example 3.11)

Recall Example 3.11. For all $T \in \{4 \cdot 10^i - 1 \mid i = 1, 2, \dots, 7\}$, there exists a task collection of jointly-realizable tasks in $d = 3$ such that $\mathcal{L}(\mathbf{w}_T^{\text{MD}}), \mathcal{L}(\mathbf{w}_T^{\text{MR}}) > 2.78 \cdot 10^{-5}$.

Construction details

For simplicity, we employ the joint solution $\mathbf{w}_\star = \mathbf{0}$, which means that for all tasks $m \in [T]$, $\mathbf{y}_m = \mathbf{0}$. For some $K \in \mathbb{N}^+$, we construct $T = 4K - 1$ tasks by defining their solution subspaces as follows:

- $K - 1$ copies of $\text{Span}((0, \sin(\frac{1}{\sqrt{K}}), \cos(\frac{1}{\sqrt{K}})))$,
- K copies of $\text{Span}((0, -\sin(\frac{1}{\sqrt{K}}), \cos(\frac{1}{\sqrt{K}})))$,
- K copies of $\text{Span}((0, \sin(\frac{1}{2\sqrt{K}}), \cos(\frac{1}{2\sqrt{K}})), (\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}))$,
- K copies of $\text{Span}((0, -\sin(\frac{1}{2\sqrt{K}}), \cos(\frac{1}{2\sqrt{K}})), (\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}))$.

For a given solution subspace of rank $d - r$, a task feature matrix $\mathbf{X} \in \mathbb{R}^{r \times d}$ of rank r (i.e., with linearly independent rows) is defined such that each row of $\mathbf{X}$ is orthogonal to the solution subspace.

We initialize $\mathbf{w}_0 = (0, \sin(\frac{1}{\sqrt{K}}), \cos(\frac{1}{\sqrt{K}}))$.

Lower bound explanation

We do not give a formal proof, but here is an explanation of why the construction works. We stick to using greedy MD for the intuition.

While learning tasks consecutively using greedy MD ordering, we start by alternating between projecting onto the 1-D subspaces, since each 2-D subspace contains a line (either $\text{Span}((0, \sin(\frac{1}{2\sqrt{K}}), \cos(\frac{1}{2\sqrt{K}})))$ or $\text{Span}((0, -\sin(\frac{1}{2\sqrt{K}}), \cos(\frac{1}{2\sqrt{K}})))$) between them and hence cannot be the farthest away. Once those are used up, we are left with the 2-D subspaces, which we alternate between.

Morally, the angle between the 1-D subspaces is $\mathcal{O}(\frac{1}{\sqrt{K}})$, so as K grows large, these first $2K - 1$ steps should bring us to about $(0, 0, \Theta(1))$. The 2-D subspaces intersect in $\text{Span}((\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}))$, and the angle between them is also $\Theta(\frac{1}{\sqrt{K}})$, so they move us a constant fraction of the way toward the closest point on $\text{Span}((\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}))$, which is $(\Theta(1), 0, \Theta(1))$. Since the first half of the tasks are projections onto 1-D subspaces near $(0, 0, 1)$, the x -coordinate contributes $\Theta(1)$ loss due to these tasks.

Quantifying the asymptotic loss. For each of the first $2K - 1$ steps, we multiply $\|\mathbf{w}\|$ by $\cos(2/\sqrt{K})$ each time, giving

$$\cos(2/\sqrt{K})^{2K-1} = (1 - 2/K + O(K^{-2}))^{2K-1} = e^{-4} + o(1).$$

The direction is $(0, -\sin(1/\sqrt{K}), \cos(1/\sqrt{K}))$, so we end at $(0, o(1), e^{-4} + o(1))$.

Next we project onto a 2-D subspace; this is $o(1)$ movement, so $\mathbf{w}$ becomes

$$\mathbf{w} = (o(1), o(1), e^{-4} + o(1)) = (0, 0, e^{-4}) + o(1).$$

Let $\mathbf{v}$ be the closest point on $\text{Span}((\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}))$ to $\mathbf{w} = (0, 0, e^{-4}) + o(1)$. Then

$$\mathbf{v} = (e^{-4}/2, 0, e^{-4}/2) + o(1)(1, 0, 1).$$

All remaining 2-D subspaces contain the line $L = \text{Span}((\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}))$, so the projections are onto lines through $\mathbf{v}$ perpendicular to L .

The angle between these lines equals the angle between the planes. Let $\mathbf{u}$ be the closest point on $\text{Span}(1, 0, 1)$ to $(0, \sin(1/\sqrt{K}), \cos(1/\sqrt{K}))$; then $\mathbf{u} = (\frac{1}{2} + o(1))(1, 0, 1)$. Thus the plane angle is

$$2 \tan^{-1} \left(\frac{\sin(1/(2\sqrt{K}))}{1/\sqrt{2} + o(1)} \right) = (1 + o(1)) \frac{\sqrt{2}}{\sqrt{K}}.$$

Each of the remaining $2K - 1$ projections multiplies $\text{dist}(\mathbf{w}, \mathbf{v})$ by $\cos((1 + o(1))\sqrt{2}/\sqrt{K})$, so overall it is scaled by $e^{-2} + o(1)$.

Originally,

$$\text{dist}((0, 0, e^{-4}), (e^{-4}/2, 0, e^{-4}/2)) = \frac{e^{-4}\sqrt{2}}{2} + o(1),$$

so afterwards the distance is $\frac{e^{-6}\sqrt{2}}{2} + o(1)$.

The direction from $\mathbf{v}$ is parallel to

$$o(1) + (0, 0, e^{-4}) - (e^{-4}/2, 0, e^{-4}/2) = o(1) + (-e^{-4}/2, 0, e^{-4}/2) = o(1) + (-\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}),$$

so the final position of $\mathbf{w}_T$ is

$$\begin{aligned} & o(1) + (e^{-4}/2, 0, e^{-4}/2) + (\frac{e^{-6}\sqrt{2}}{2} + o(1))(o(1) + (-\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}})) \\ &= o(1) + (e^{-4}/2 - e^{-6}/2, 0, e^{-4}/2 + e^{-6}/2). \end{aligned}$$

Hence, the x -coordinate contributes the following approximate loss due to the first half of the tasks:

$$\mathcal{L}(\mathbf{w}_T) \approx \frac{1}{2}(e^{-4}/2 - e^{-6}/2)^2 \approx 3.135 \cdot 10^{-5}. \quad (3.8)$$

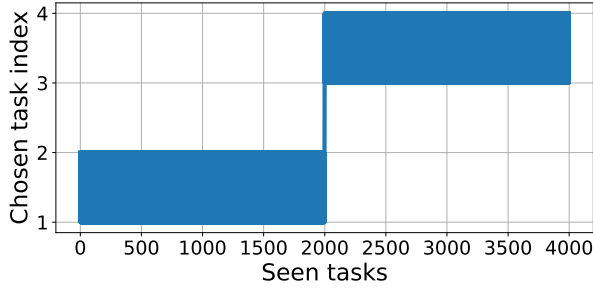
Experimental results

Constructing the tasks as explained in section 3.11.2, using QR decomposition to acquire $\mathbf{X}$ for each solution subspace, we observe that the task orderings for greedy MD (definition 3.7) and greedy MR (definition 3.8) coincide, as explained in section 3.11.2, for all $T \in \{4 \cdot 10^i - 1 \mid i = 1, 2, \dots, 7\}$. As observed in fig. 3.18, the greedy ordering alternates between the first two task groups until these are depleted, then switches to the other two task groups. Loss diminishes linearly while learning the first half of the tasks, then increases to the predicted value (eq. (3.8)) while learning the second half.

In fig. 3.19, we show that example 3.11 holds and that the theoretical value (eq. (3.8)) is valid.

Code for reproducibility

```
import numpy as np
```



(a) Chosen task group under greedy ordering.



(b) Average loss during continual learning under greedy ordering.

Figure 3.18: Example of continual learning the 3d adversarial construction with greedy ordering. Chosen tasks coincide for MD and MR greedy orderings. $K = 1000, T = 3999$.

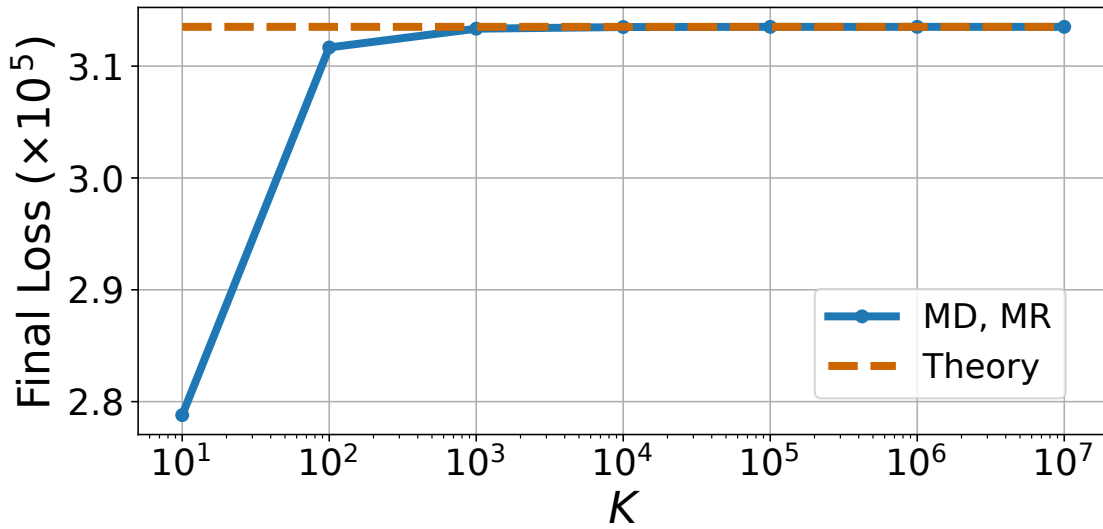


Figure 3.19: Final average loss after greedy ordering for the 3d adversarial construction, for different task counts. $T = 4K - 1$.

```

def orth_complement_rows(W, *, rtol=1e-12):
    """ Given W in  $\mathbb{R}^{(d-r) \times d}$  (full row rank), return X in  $\mathbb{R}^{r \times d}$ 
    whose rows form an orthonormal basis of the orthogonal complement
    of the row-space of W. """
    W = np.asarray(W, dtype=float)
    d = W.shape[1]
    # Full QR of W.T  $\rightarrow$  Q is d×d and orthogonal
    # The first rank columns of Q span the rows of W;
    # the remaining columns span their orthogonal complement.
    Q, _ = np.linalg.qr(W.T, mode='complete') # Q in  $\mathbb{R}^{d \times d}$ 
    rank = np.linalg.matrix_rank(W, tol=rtol)
    # Take the last r = d - rank columns of Q, transpose to get rows
    X = Q[:, rank:].T # X in  $\mathbb{R}^{r \times d}$ 
    return X

ORDERING = 'MD' # 'MR' / 'MD'

K = 1000
w1 = np.array([[0, np.sin(1/np.sqrt(K)), np.cos(1/np.sqrt(K))]])
w2 = np.array([[0, -np.sin(1/np.sqrt(K)), np.cos(1/np.sqrt(K))]])
w3 = np.array([
    [0, np.sin(0.5/np.sqrt(K)), np.cos(0.5/np.sqrt(K))],
    [1/np.sqrt(2), 0, 1/np.sqrt(2)]
])
w4 = np.array([
    [0, -np.sin(0.5/np.sqrt(K)), np.cos(0.5/np.sqrt(K))],
    [1/np.sqrt(2), 0, 1/np.sqrt(2)]
])

```

```

ws = [w1, w2, w3, w4]
tasks = [orth_complement_rows(w) for w in ws]
projections = [np.eye(3) - np.linalg.pinv(X) @ X for X in tasks]
total_collection_count = [K-1, K, K, K]
T = np.sum(total_collection_count)
collection_count = total_collection_count.copy()
w = np.array([0, np.sin(1/np.sqrt(K)), np.cos(1/np.sqrt(K))])
residual_per_projection = [np.linalg.norm(X @ w)**2 for X in tasks]
total_loss = (
    (1/T)
    * np.asarray(residual_per_projection)
    @ np.asarray(total_collection_count)
)
losses = [total_loss]
for i in range(T):
    chosen_task = None
    distance = -np.inf
    new_w, new_w_candidate = None, None
    for task_index in range(4):
        if collection_count[task_index] > 0:
            if ORDERING == 'MD':
                new_w_candidate = projections[task_index] @ w
                new_distance = np.linalg.norm(new_w_candidate - w)**2
            elif ORDERING == 'MR':
                new_distance = np.linalg.norm(tasks[task_index] @ w)**2
        if new_distance > distance:
            chosen_task = task_index

```

```

        distance = new_distance

        new_w = new_w_candidate

    if new_w is None:

        new_w = projections[chosen_task] @ w

    w = new_w

    collection_count[chosen_task] -= 1

    residual_per_projection = [np.linalg.norm(X @ w)**2 for X in tasks]

    total_loss = (

        (1/T)

        * np.asarray(residual_per_projection)

        @ np.asarray(total_collection_count)

    )

    losses.append(total_loss)

```

3.12 Appendix to section 3.5.2: Single-pass vs. repetition

3.12.1 Appendix to the upper bound for greedy orderings with repetition (theorem 3.13)

Recall Theorem 3.13. Under a Maximum Distance greedy ordering with repetition (denoted $\tau_{\text{MD-R}}$) over T jointly-realizable tasks, the loss of scheme 3.1 after $k \geq 2$ iterations is upper bounded as

$$\mathcal{L}(\mathbf{w}_k^{\tau_{\text{MD-R}}}) = \mathcal{O}\left(\frac{1}{\sqrt[3]{k}}\right).$$

The main propositions leading to the proof of theorem 3.13 are in section 3.12.1, with auxiliary claims in section 3.12.1. We use geometric analysis to derive an upper bound on

how fast iterates can grow, and then bound the iterates relative to the original distance to the teacher $\mathbf{w}_\star$. Finally, we upper bound the loss by the greedy iterate size when repetitions are allowed.

Comparison to convergence rates of other task orderings

Table 3.1: Loss bounds in continual linear regression over jointly realizable tasks (based on Table 1 of Evron et al. [43]). The presented bounds are “worst case”: upper bounds apply to any collection of T jointly realizable tasks, while lower bounds are achieved by specific constructions. Bounds for random orderings apply to the expected loss. We omit scaling terms ($\|\mathbf{w}_\star\|^2 R^2$) and constant multiplicative factors (which are mild). Notation: k = iterations; d = dimensions; $\bar{r}, r_{\max}$ = average/maximum data matrix ranks; $a, b \triangleq \min(a, b)$.

Bound	Paper / Ordering	Single-pass Greedy	Greedy with Repetition	Random w/o Replacement	Random with Replacement
Upper	Evron et al. [2022]	—	—	—	$\frac{d - \bar{r}}{k}$
	Evron et al. [2026]	—	—	$\frac{1}{\sqrt[4]{T}}, \frac{d - \bar{r}}{T}$	$\frac{1}{\sqrt[4]{k}}, \frac{\sqrt{d - \bar{r}}}{k}, \frac{\sqrt{T\bar{r}}}{k}$
	Attia et al. [2025]	—	—	$\frac{1}{\sqrt{T}}$	$\frac{1}{\sqrt{k}}$
	Ours	None	$\frac{1}{\sqrt[3]{k}}$	—	—
Lower	Evron et al. [2022] (*)	$\frac{1}{T}$	$\frac{1}{k}$	$\frac{1}{T}$	$\frac{1}{k}$
	Ours	$\Omega(1)$	—	—	—

(*) Although Evron et al. [41] did not state these lower bounds explicitly, their proof of Theorem 10 provides a 2-task construction which, when replicated $\lfloor T/2 \rfloor$ times, produces a $\Theta(1/k)$ bound for both random and greedy orderings with any general T .

Deriving the bound

Proposition 3.14. Let $\mathbf{w}_0, \mathbf{w}_1 \in \mathbb{R}^d$ and let $\mathbf{P} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ be an orthogonal projection. Then

$$\|\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1\|^2 \leq \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \|(\mathbf{I} - \mathbf{P})\mathbf{w}_1\|^2 + 2\|(\mathbf{I} - \mathbf{P})\mathbf{w}_1\|\|(\mathbf{I} - \mathbf{P})\mathbf{w}_0\|.$$

Proof. Denote

$$\mathbf{u} \triangleq (\mathbf{I} - \mathbf{P})\mathbf{w}_0, \quad \mathbf{v} \triangleq (\mathbf{I} - \mathbf{P})\mathbf{w}_1.$$

Then $\mathbf{u}, \mathbf{v} \in \ker(\mathbf{P})$ and are orthogonal to $\mathbf{P}\mathbf{w}_0, \mathbf{P}\mathbf{w}_1 \in \text{im}(\mathbf{P})$, respectively.

Decompose:

$$\mathbf{w}_0 - \mathbf{w}_1 = (\mathbf{P}\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1) + (\mathbf{u} - \mathbf{v}),$$

$$\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1 = (\mathbf{P}\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1) + \mathbf{u}.$$

Using orthogonality:

$$\|\mathbf{w}_0 - \mathbf{w}_1\|^2 = \|\mathbf{P}\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1\|^2 + \|\mathbf{u} - \mathbf{v}\|^2, \quad (3.9)$$

$$\|\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1\|^2 = \|\mathbf{P}\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1\|^2 + \|\mathbf{u}\|^2. \quad (3.10)$$

Subtracting (3.10) from (3.9) gives

$$\|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \|\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1\|^2 = \|\mathbf{u} - \mathbf{v}\|^2 - \|\mathbf{u}\|^2 = \|\mathbf{v}\|^2 - 2\mathbf{u}^\top \mathbf{v}.$$

So,

$$\|\mathbf{w}_0 - \mathbf{P}\mathbf{w}_1\|^2 = \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \|\mathbf{v}\|^2 + 2\mathbf{u}^\top \mathbf{v}$$

$$[\text{Cauchy-Schwarz}] \leq \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \|(\mathbf{I} - \mathbf{P})\mathbf{w}_1\|^2 + 2\|(\mathbf{I} - \mathbf{P})\mathbf{w}_1\| \|(\mathbf{I} - \mathbf{P})\mathbf{w}_0\|.$$

□

Corollary 3.15. Consider the case in eq. (3.2), i.e., $(\mathbf{w}_t)_{t=0}^k$ are iterates such that $\mathbf{w}_0 = \mathbf{0}$

and $\forall t \in [k], (\mathbf{w}_t - \mathbf{w}_\star) = \mathbf{P}_{\tau(t)}(\mathbf{w}_{t-1} - \mathbf{w}_\star)$, then $\forall k \geq 2$,

$$\begin{aligned} \|\mathbf{w}_0 - \mathbf{w}_k\|^2 &\leq \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 \\ &\quad + 2 \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_0 - \mathbf{w}_\star)\|. \end{aligned}$$

Proof. We repeatedly apply the above proposition:

$$\begin{aligned} \|\mathbf{w}_0 - \mathbf{w}_k\|^2 &= \|(\mathbf{w}_0 - \mathbf{w}_\star) - (\mathbf{w}_k - \mathbf{w}_\star)\|^2 \\ &= \|(\mathbf{w}_0 - \mathbf{w}_\star) - \mathbf{P}_{\tau(k)}(\mathbf{w}_{k-1} - \mathbf{w}_\star)\|^2 \\ &\leq \underbrace{\|(\mathbf{w}_0 - \mathbf{w}_\star) - (\mathbf{w}_{k-1} - \mathbf{w}_\star)\|^2}_{\text{continue inductively}} - \|(\mathbf{I} - \mathbf{P}_{\tau(k)})(\mathbf{w}_{k-1} - \mathbf{w}_\star)\|^2 \\ &\quad + 2\|(\mathbf{I} - \mathbf{P}_{\tau(k)})(\mathbf{w}_{k-1} - \mathbf{w}_\star)\| \|(\mathbf{I} - \mathbf{P}_{\tau(k)})(\mathbf{w}_0 - \mathbf{w}_\star)\| \\ &\leq \|(\mathbf{w}_0 - \mathbf{w}_\star) - (\mathbf{w}_1 - \mathbf{w}_\star)\|^2 - \sum_{t=2}^k \|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \\ &\quad + 2 \sum_{t=2}^k \|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_{t-1} - \mathbf{w}_\star)\| \|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_0 - \mathbf{w}_\star)\| \\ &= \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 + 2 \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \|(\mathbf{I} - \mathbf{P}_{\tau(t)})(\mathbf{w}_0 - \mathbf{w}_\star)\|. \end{aligned}$$

□

Corollary 3.16. If the first step is MD greedy, i.e., $\forall m \in [T], \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_0 - \mathbf{w}_\star)\| \leq \|\mathbf{w}_0 - \mathbf{w}_1\|$, then $\forall k \geq 2$,

$$\|\mathbf{w}_0 - \mathbf{w}_k\|^2 \leq \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 + 2\|\mathbf{w}_0 - \mathbf{w}_1\| \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\|.$$

Proposition 3.17. Under greedy MD ordering, either single-pass or with repetition, for all $k \geq 1$,

$$\|\mathbf{w}_{k+1} - \mathbf{w}_k\| \leq \|\mathbf{w}_0 - \mathbf{w}_1\| \cdot 2e^{4/3} k^{1/3}.$$

Proof.

$$\begin{aligned}
\|\mathbf{w}_{k+1} - \mathbf{w}_k\| &= \|(\mathbf{w}_k - \mathbf{w}_\star) - \mathbf{P}_{\tau(k+1)}(\mathbf{w}_k - \mathbf{w}_\star)\| \\
[\text{orth. proj.}] &\leq \|(\mathbf{w}_k - \mathbf{w}_\star) - \mathbf{P}_{\tau(k+1)}(\mathbf{w}_0 - \mathbf{w}_\star)\| \\
[\text{triangle inequality}] &\leq \|(\mathbf{w}_k - \mathbf{w}_\star) - (\mathbf{w}_0 - \mathbf{w}_\star)\| + \|(\mathbf{w}_0 - \mathbf{w}_\star) - \mathbf{P}_{\tau(k+1)}(\mathbf{w}_0 - \mathbf{w}_\star)\| \\
[\text{greediness}] &\leq \|\mathbf{w}_0 - \mathbf{w}_k\| + \|\mathbf{w}_0 - \mathbf{w}_1\| \\
\implies \quad &\|\mathbf{w}_{k+1} - \mathbf{w}_k\| - \|\mathbf{w}_0 - \mathbf{w}_1\| \leq \|\mathbf{w}_0 - \mathbf{w}_k\|,
\end{aligned}$$

where we used the fact that the orthogonal projection of a given point on a subspace is the closest point in this subspace to the given point in the Euclidean-norm sense.

If $\|\mathbf{w}_{k+1} - \mathbf{w}_k\| < \|\mathbf{w}_0 - \mathbf{w}_1\|$, the proposition follows immediately. Otherwise $\|\mathbf{w}_{k+1} - \mathbf{w}_k\| \geq \|\mathbf{w}_0 - \mathbf{w}_1\|$, and thus

$$(\|\mathbf{w}_{k+1} - \mathbf{w}_k\| - \|\mathbf{w}_0 - \mathbf{w}_1\|)^2 \leq \|\mathbf{w}_0 - \mathbf{w}_k\|^2.$$

Plugging this into corollary 3.16 gives

$$\begin{aligned}
(\|\mathbf{w}_{k+1} - \mathbf{w}_k\| - \|\mathbf{w}_0 - \mathbf{w}_1\|)^2 &\leq \|\mathbf{w}_0 - \mathbf{w}_k\|^2 \\
&\leq \|\mathbf{w}_0 - \mathbf{w}_1\|^2 - \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 + 2\|\mathbf{w}_0 - \mathbf{w}_1\| \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \\
\|\mathbf{w}_{k+1} - \mathbf{w}_k\|^2 + \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 &\leq 2\|\mathbf{w}_0 - \mathbf{w}_1\| \left(\|\mathbf{w}_{k+1} - \mathbf{w}_k\| + \sum_{t=2}^k \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \right) \\
\|\mathbf{w}_0 - \mathbf{w}_1\| &\geq \frac{\sum_{t=2}^{k+1} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2}{2 \sum_{t=2}^{k+1} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|}.
\end{aligned}$$

Using the same derivation, when all steps are greedy, $\|\mathbf{w}_2 - \mathbf{w}_1\| \geq \frac{\sum_{t=3}^{k+1} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2}{2 \sum_{t=3}^{k+1} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|}, \dots$,
 $\|\mathbf{w}_k - \mathbf{w}_{k-1}\| \geq \frac{\sum_{t=k+1}^{k+1} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2}{2 \sum_{t=k+1}^{k+1} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|} = \frac{\|\mathbf{w}_{k+1} - \mathbf{w}_k\|^2}{2 \|\mathbf{w}_{k+1} - \mathbf{w}_k\|} = \frac{1}{2} \|\mathbf{w}_{k+1} - \mathbf{w}_k\|$, acquiring a lower bound
 $\forall t \in [k]: \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \geq \frac{\sum_{j=t+1}^{k+1} \|\mathbf{w}_j - \mathbf{w}_{j-1}\|^2}{2 \sum_{j=t+1}^{k+1} \|\mathbf{w}_j - \mathbf{w}_{j-1}\|}$. Hence, we define the sequence $(C_t)_{t=1}^{k+1}$ by the

backward recurrence $C_{k+1} = \|\mathbf{w}_{k+1} - \mathbf{w}_k\|$ and $\forall t \in [k], C_t = \frac{\sum_{j=t+1}^{k+1} C_j^2}{2 \sum_{j=t+1}^{k+1} C_j}$. By applying claim 3.20, using the sequences $a_i \triangleq \|\mathbf{w}_{k+1-i} - \mathbf{w}_{k-i}\|$, $b_i \triangleq C_{k+1-i}$ for all $i \in \{0, \dots, k\}$, we observe that $\|\mathbf{w}_0 - \mathbf{w}_1\| \geq C_1$.

We investigate the sequence (C_t) in corollary 3.23, where we prove that if $C_{k+1} = \|\mathbf{w}_{k+1} - \mathbf{w}_k\| > 0$, this sequence maintains $\forall k \geq 1$,

$$C_1 \geq C_{k+1} \cdot \frac{1}{2e^{4/3}} k^{-1/3},$$

(and indeed if $\|\mathbf{w}_{k+1} - \mathbf{w}_k\| = 0$, the proposition follows immediately), and thus we get that

$$\begin{aligned} \|\mathbf{w}_0 - \mathbf{w}_1\| &\geq C_1 \geq \|\mathbf{w}_{k+1} - \mathbf{w}_k\| \frac{1}{2e^{4/3}} k^{-1/3} \\ \implies \|\mathbf{w}_{k+1} - \mathbf{w}_k\| &\leq \|\mathbf{w}_0 - \mathbf{w}_1\| \cdot 2e^{4/3} k^{1/3}. \end{aligned}$$

□

Lemma 3.18. Under greedy MD ordering, either single pass or with repetition, for all $k \geq 1$,

$$\|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2 \leq \frac{4e^{8/3}}{3} \frac{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2}{k^{1/3}}.$$

Proof. Applying the above proposition for a starting index $t-1$ (instead of 0), for all $k > t$,

$$\begin{aligned} \|\mathbf{w}_{k-1} - \mathbf{w}_k\| &\leq \|\mathbf{w}_{t-1} - \mathbf{w}_t\| \cdot 2e^{4/3} (k-t)^{1/3} \\ \implies \frac{\|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2}{4e^{8/3} (k-t)^{2/3}} &\leq \|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2. \end{aligned}$$

The Pythagorean theorem gives

$$\|\mathbf{w}_k - \mathbf{w}_\star\|^2 = \|\mathbf{w}_{k-1} - \mathbf{w}_\star\|^2 - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2 = \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 - \sum_{t=1}^k \|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2.$$

Plugging in the above proposition gives

$$\begin{aligned}
0 &\leq \|\mathbf{w}_k - \mathbf{w}_\star\|^2 = \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 - \sum_{t=1}^k \|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2 \\
&\leq \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2 - \sum_{t=1}^{k-1} \frac{\|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2}{4e^{8/3}(k-t)^{2/3}} \\
&= \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2 \left(1 + \frac{1}{4e^{8/3}} \sum_{i=1}^{k-1} \frac{1}{i^{2/3}} \right) \\
\left[\sum_{i=1}^{k-1} \frac{1}{i^{2/3}} \geq 3(k^{1/3} - 1) \right] &\leq \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2 \left(1 + \frac{3(k^{1/3} - 1)}{4e^{8/3}} \right) \\
\Rightarrow \|\mathbf{w}_{k-1} - \mathbf{w}_k\|^2 &\leq \frac{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2}{\left(1 + \frac{3(k^{1/3} - 1)}{4e^{8/3}} \right)} = \frac{4e^{8/3}}{3} \frac{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2}{k^{1/3} + \frac{4e^{8/3}}{3} - 1} \\
&\leq \frac{4e^{8/3}}{3} \frac{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2}{k^{1/3}},
\end{aligned}$$

where we used $\sum_{i=1}^{k-1} \frac{1}{i^{2/3}} \geq \int_1^k \frac{1}{x^{2/3}} dx = 3(k^{1/3} - 1)$. □

Proof of theorem 3.13. Under greedy MD ordering with repetitions,

$$\begin{aligned}
\mathcal{L}(\mathbf{w}_k) &= \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m \mathbf{w}_k - \mathbf{y}_m\|^2 = \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m (\mathbf{w}_k - \mathbf{w}_\star)\|^2 \\
&\leq \frac{1}{\|\mathbf{w}_\star\|^2 R^2 T} \sum_{m=1}^T \|\mathbf{X}_m\|^2 \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_k - \mathbf{w}_\star)\|^2 \\
&\leq \frac{1}{\|\mathbf{w}_\star\|^2 T} \sum_{m=1}^T \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_k - \mathbf{w}_\star)\|^2 \\
[\text{greedy+repetitions}] &\leq \frac{1}{\|\mathbf{w}_\star\|^2} \|\mathbf{w}_k - \mathbf{w}_{k+1}\|^2 \\
[\text{above, } \mathbf{w}_0 = \mathbf{0}] &\leq \frac{1}{\|\mathbf{w}_\star\|^2} \frac{4e^{8/3}}{3} \frac{\|\mathbf{w}_\star\|^2}{(k+1)^{1/3}} = \frac{4e^{8/3}}{3} \frac{1}{(k+1)^{1/3}} \\
&= \mathcal{O}(k^{-1/3}).
\end{aligned}$$

□

Auxiliary claims

Claim 3.19. For $s, r > 0$ and $\alpha \geq \frac{1}{2}$ define

$$f_{s,r}(\alpha) \triangleq r \frac{s + \alpha^2 r}{s + \alpha r}.$$

- (a) For all $s, r > 0$, $\frac{\partial f_{s,r}(\alpha)}{\partial \alpha} \geq 0$ on $\alpha \in [\frac{1}{2}, \infty)$; hence f is non-decreasing in α .
- (b) For all $\alpha \geq \frac{1}{2}$ and $s > 0$, $\frac{\partial f_{s,r}(\alpha)}{\partial r} > 0$; hence $f_{s,r}(\alpha)$ is strictly increasing in r .
- (c) For all $\alpha \in [\frac{1}{2}, 1]$ and $r > 0$, $\frac{\partial f_{s,r}(\alpha)}{\partial s} \geq 0$, hence $f_{s,r}(\alpha)$ is non-decreasing in s on that α -range.

Proof.

- (a) Derivative with respect to α :

$$\frac{\partial f_{s,r}}{\partial \alpha} = r \frac{(2\alpha r)(s + \alpha r) - (s + \alpha^2 r)(r)}{(s + \alpha r)^2}.$$

Expanding the numerator:

$$(2\alpha r)(s + \alpha r) = 2\alpha rs + 2\alpha^2 r^2,$$

$$(s + \alpha^2 r)r = rs + \alpha^2 r^2,$$

$$\text{difference} = (2\alpha - 1)rs + \alpha^2 r^2.$$

Thus

$$\frac{\partial f_{s,r}}{\partial \alpha} = \frac{r^2 [(2\alpha - 1)s + \alpha^2 r]}{(s + \alpha r)^2} \geq 0.$$

- (b) Derivative with respect to r :

$$\frac{\partial f_{s,r}}{\partial r} = \frac{(s + 2\alpha^2 r)(s + \alpha r) - \alpha(rs + \alpha^2 r^2)}{(s + \alpha r)^2}.$$

Expand the numerator term-by-term:

$$(s + 2\alpha^2 r)(s + \alpha r) = s^2 + s\alpha r + 2\alpha^2 rs + 2\alpha^3 r^2,$$

$$\alpha(rs + \alpha^2 r^2) = \alpha rs + \alpha^3 r^2,$$

$$\text{difference} = s^2 + 2\alpha^2 rs + \alpha^3 r^2 > 0.$$

Hence $\partial f_{s,r}/\partial r > 0$.

(c) Derivative with respect to s :

$$\frac{\partial f_{s,r}}{\partial s} = r \frac{(s + \alpha r) - (s + \alpha^2 r)}{(s + \alpha r)^2} = \frac{r^2 \alpha (1 - \alpha)}{(s + \alpha r)^2}.$$

For $\alpha \in [\frac{1}{2}, 1]$ the factor $\alpha(1 - \alpha) \geq 0$, yielding a non-negative derivative.

□

Claim 3.20. Let $n \geq 1$ and let $(a_i)_{i=0}^n$ be a sequence of strictly positive numbers such that

$$\forall i \geq 1, \quad a_i \geq \frac{\sum_{j=0}^{i-1} a_j^2}{2 \sum_{j=0}^{i-1} a_j}.$$

Define a second sequence $(b_i)_{i=0}^n$ recursively by

$$b_0 = a_0 > 0, \quad b_i = \frac{\sum_{j=0}^{i-1} b_j^2}{2 \sum_{j=0}^{i-1} b_j} \quad (i \geq 1).$$

Then $a_n \geq b_n$.

Proof. For either sequence $x \in \{a, b\}$ and each $k \geq 1$ set

$$S_k^{(x)} \triangleq \sum_{j=0}^k x_j, \quad Q_k^{(x)} \triangleq \sum_{j=0}^k x_j^2, \quad R_k^{(x)} \triangleq \frac{Q_k^{(x)}}{S_k^{(x)}} > 0.$$

For $k \geq 1$ we can express the defining relations as

$$a_k \geq \frac{R_{k-1}^{(a)}}{2}, \quad b_k = \frac{R_{k-1}^{(b)}}{2},$$

and, using $Q_{k-1}^{(x)} = R_{k-1}^{(x)} S_{k-1}^{(x)}$,

$$\begin{aligned} R_k^{(x)} &= \frac{Q_k^{(x)}}{S_k^{(x)}} = \frac{x_k^2 + Q_{k-1}^{(x)}}{x_k + S_{k-1}^{(x)}} = \frac{x_k^2 + R_{k-1}^{(x)} S_{k-1}^{(x)}}{x_k + S_{k-1}^{(x)}} = R_{k-1}^{(x)} \frac{S_{k-1}^{(x)} + \left(\frac{x_k}{R_{k-1}^{(x)}}\right)^2 R_{k-1}^{(x)}}{S_{k-1}^{(x)} + \left(\frac{x_k}{R_{k-1}^{(x)}}\right) R_{k-1}^{(x)}} \\ &= f_{S_{k-1}^{(x)}, R_{k-1}^{(x)}} \left(\frac{x_k}{R_{k-1}^{(x)}} \right), \end{aligned} \quad (*)$$

where $f_{s,r}(\alpha)$ is defined as in claim 3.19.

We prove by induction that $\forall k \in \{0, \dots, n\}$,

$$a_k \geq b_k, \quad S_k^{(a)} \geq S_k^{(b)}, \quad R_k^{(a)} \geq R_k^{(b)}.$$

Base case $k = 0$. All equalities $a_0 = b_0$, $R_0^{(a)} = R_0^{(b)} = a_0$, $S_0^{(a)} = S_0^{(b)} = a_0$ hold.

Induction step $k \rightarrow k + 1$. Assume the inequalities hold for k .

(i) Comparing a_{k+1} and b_{k+1} . By the recursion and the inductive hypothesis,

$$a_{k+1} \geq \frac{R_k^{(a)}}{2} \geq \frac{R_k^{(b)}}{2} = b_{k+1}.$$

(ii) Comparing $S_{k+1}^{(a)}$ and $S_{k+1}^{(b)}$. From the above and the inductive hypothesis,

$$S_{k+1}^{(a)} = S_k^{(a)} + a_{k+1} \geq S_k^{(b)} + b_{k+1} = S_{k+1}^{(b)}.$$

(iii) Comparing $R_{k+1}^{(a)}$ and $R_{k+1}^{(b)}$. From (*),

$$R_{k+1}^{(b)} = f_{S_k^{(b)}, R_k^{(b)}}\left(\frac{1}{2}\right), \quad R_{k+1}^{(a)} = f_{S_k^{(a)}, R_k^{(a)}}\left(\frac{a_{k+1}}{R_k^{(a)}}\right),$$

where $\frac{a_{k+1}}{R_k^{(a)}} \geq \frac{1}{2}$. By the inductive hypothesis $S_k^{(a)} \geq S_k^{(b)}$ and $R_k^{(a)} \geq R_k^{(b)}$, and by claim 3.19 the function $f_{s,r}(\alpha)$ is (a) increasing in α , (b) increasing in r for all $\alpha \geq \frac{1}{2}$, and (c) increasing in s when $\alpha = \frac{1}{2}$. Therefore

$$R_{k+1}^{(a)} = f_{S_k^{(a)}, R_k^{(a)}}\left(\frac{a_{k+1}}{R_k^{(a)}}\right) \geq f_{S_k^{(a)}, R_k^{(a)}}\left(\frac{1}{2}\right) \geq f_{S_k^{(b)}, R_k^{(b)}}\left(\frac{1}{2}\right) = R_{k+1}^{(b)}.$$

Hence the inequalities hold for $k+1$, concluding the proof by induction. Taking $k = n$ yields $a_n \geq b_n$, proving the claim. $\square$

Claim 3.21. $\forall x > -1, \log(1+x) \geq \frac{x}{1+x}$.

Proof. For $x > -1$, define $f(x) = \ln(1+x) - \frac{x}{1+x}$, then

$$f'(x) = \frac{1}{1+x} - \frac{1+x-x}{(1+x)^2} = \frac{x}{(1+x)^2}.$$

Since $f'(x) < 0$ for $x < 0$ and $f'(x) > 0$ for $x > 0$, the point $x = 0$, where $f(0) = 0$, is the global minimum of f . Hence $\ln(1+x) \geq \frac{x}{1+x}$ for all $x > -1$. $\square$

Claim 3.22. Let $\lambda > \mu > 0, \kappa > 0$. Define the sequences $(A_n)_{n=0}^\infty, (B_n)_{n=0}^\infty, (\tilde{C}_n)_{n=0}^\infty$ by $A_0 > 0, B_0 > 0, \tilde{C}_0 = \kappa \frac{B_0}{A_0}$, and for $n \geq 1$,

$$A_n \triangleq A_{n-1} + \lambda \frac{B_{n-1}}{A_{n-1}}, \quad B_n \triangleq B_{n-1} + \mu \left(\frac{B_{n-1}}{A_{n-1}}\right)^2, \quad \tilde{C}_n \triangleq \kappa \frac{B_n}{A_n},$$

then $\forall n \geq 1, \tilde{C}_n \geq \tilde{C}_0 \cdot \gamma n^{-\frac{\lambda-\mu}{2\lambda-\mu}}$, where $\gamma \triangleq \exp\left(-\frac{2\lambda(\lambda-\mu)}{\mu(2\lambda-\mu)}\right)$.

Proof. By induction, $A_n > 0$ and $B_n > 0$ for all $n \geq 0$. Define the helper sequence for all

$n \geq 0$, $f_n = \frac{A_n^2}{B_n} > 0$, then $A_n = \sqrt{B_n f_n}$ and $\forall n \geq 1$,

$$\begin{aligned}
f_n &= \frac{\left(A_{n-1} + \lambda \frac{B_{n-1}}{A_{n-1}}\right)^2}{B_{n-1} + \mu \left(\frac{B_{n-1}}{A_{n-1}}\right)^2} = \frac{\left(\sqrt{B_{n-1} f_{n-1}} + \lambda \frac{B_{n-1}}{\sqrt{B_{n-1} f_{n-1}}}\right)^2}{B_{n-1} + \mu \left(\frac{B_{n-1}}{\sqrt{B_{n-1} f_{n-1}}}\right)^2} \\
&= \frac{B_{n-1} \left(\sqrt{f_{n-1}} + \frac{\lambda}{\sqrt{f_{n-1}}}\right)^2}{B_{n-1} \left(1 + \frac{\mu}{f_{n-1}}\right)} \\
[B_{n-1} > 0] &= \frac{\frac{1}{f_{n-1}}(f_{n-1} + \lambda)^2}{1 + \frac{\mu}{f_{n-1}}} = \frac{f_{n-1}^2 + 2\lambda f_{n-1} + \lambda^2}{f_{n-1} + \mu} \\
&= f_{n-1} + 2\lambda - \mu + \frac{(\lambda - \mu)^2}{f_{n-1} + \mu} \\
&\geq f_{n-1} + 2\lambda - \mu,
\end{aligned}$$

then $f_n \geq f_{n-1} + 2\lambda - \mu \geq f_0 + n(2\lambda - \mu) = \frac{A_0^2}{B_0} + n(2\lambda - \mu)$.

Now, observe $\forall n \geq 1$,

$$\begin{aligned}
\tilde{C}_n &= \kappa \frac{B_n}{A_n} = \kappa \frac{B_{n-1} + \mu \left(\frac{B_{n-1}}{A_{n-1}}\right)^2}{A_{n-1} + \lambda \frac{B_{n-1}}{A_{n-1}}} = \kappa \frac{B_{n-1}}{A_{n-1}} \frac{1 + \mu \frac{B_{n-1}}{A_{n-1}^2}}{1 + \lambda \frac{B_{n-1}}{A_{n-1}^2}} = \tilde{C}_{n-1} \frac{1 + \frac{\mu}{f_{n-1}}}{1 + \frac{\lambda}{f_{n-1}}} \\
&= \tilde{C}_0 \prod_{i=0}^{n-1} \frac{f_i + \mu + \lambda - \lambda}{f_i + \lambda} = \tilde{C}_0 \prod_{i=0}^{n-1} \left(1 - \frac{\lambda - \mu}{f_i + \lambda}\right),
\end{aligned}$$

and note that $\forall i \geq 0, 0 < \frac{\lambda-\mu}{f_i+\lambda} < 1$. Taking logarithms,

$$\begin{aligned}
\log \tilde{C}_n &= \log \tilde{C}_0 + \sum_{i=0}^{n-1} \log \left(1 - \frac{\lambda-\mu}{f_i+\lambda} \right) \\
[\text{claim 3.21}] &\geq \log \tilde{C}_0 + \sum_{i=0}^{n-1} \frac{-\frac{\lambda-\mu}{f_i+\lambda}}{1 - \frac{\lambda-\mu}{f_i+\lambda}} = \log \tilde{C}_0 - \sum_{i=0}^{n-1} \frac{\lambda-\mu}{f_i+\mu} \\
[\lambda > \mu] &\geq \log \tilde{C}_0 - \sum_{i=0}^{n-1} \frac{\lambda-\mu}{f_0 + i(2\lambda-\mu) + \mu} \geq \log \tilde{C}_0 - \sum_{i=0}^{n-1} \frac{\lambda-\mu}{i(2\lambda-\mu) + \mu} \\
[\text{assuming } n \geq 2] &= \log \tilde{C}_0 - \frac{\lambda-\mu}{\mu} - \sum_{i=1}^{n-1} \frac{\lambda-\mu}{i(2\lambda-\mu) + \mu} \\
&\geq \log \tilde{C}_0 - \frac{\lambda-\mu}{\mu} - \sum_{i=1}^{n-1} \frac{\lambda-\mu}{i(2\lambda-\mu)} \\
\left[\sum_{i=1}^m \frac{1}{i} \leq 1 + \log m \right] &\geq \log \tilde{C}_0 - \frac{\lambda-\mu}{\mu} - \frac{\lambda-\mu}{2\lambda-\mu} (1 + \log(n-1)) \\
&\geq \log \tilde{C}_0 - \frac{\lambda-\mu}{\mu} - \frac{\lambda-\mu}{2\lambda-\mu} - \frac{\lambda-\mu}{2\lambda-\mu} \log n \\
&\Rightarrow \tilde{C}_n \geq \tilde{C}_0 n^{-\frac{\lambda-\mu}{2\lambda-\mu}} \exp\left(-\frac{2\lambda(\lambda-\mu)}{\mu(2\lambda-\mu)}\right).
\end{aligned}$$

When $n = 1$, $\log \tilde{C}_1 \geq \log \tilde{C}_0 - \frac{\lambda-\mu}{\mu} > \log \tilde{C}_0 - \frac{\lambda-\mu}{\mu} - \frac{\lambda-\mu}{2\lambda-\mu}$, hence $\tilde{C}_1 \geq \tilde{C}_0 \exp\left(-\frac{2\lambda(\lambda-\mu)}{\mu(2\lambda-\mu)}\right)$, concluding the proof for all $n \geq 1$. $\square$

Corollary 3.23. For $(C_t)_{t=1}^{k+1}$ defined by the backwards recurrence $C_{k+1} > 0$ and $\forall t \in [k], C_t = \frac{\sum_{j=t+1}^{k+1} C_j^2}{2 \sum_{j=t+1}^{k+1} C_j}$, for all $k \geq 1$,

$$C_1 \geq C_{k+1} \cdot \frac{1}{2e^{4/3}} k^{-1/3}.$$

Proof. By defining $\forall n \in \{0, \dots, k\} : A_n = \sum_{j=k+1-n}^{k+1} C_j, B_n = \sum_{j=k+1-n}^{k+1} C_j^2$, we note that

$A_0 = C_{k+1} > 0, B_0 = C_{k+1}^2 > 0$, and for all $1 \leq n \leq k$,

$$\begin{aligned} A_n &= A_{n-1} + C_{k+1-n} = A_{n-1} + \frac{\sum_{j=k+1-n+1}^{k+1} C_j^2}{2 \sum_{j=k+1-n+1}^{k+1} C_j} = A_{n-1} + \frac{1}{2} \frac{B_{n-1}}{A_{n-1}}, \\ B_n &= B_{n-1} + C_{k+1-n}^2 = B_{n-1} + \left(\frac{1}{2} \frac{B_{n-1}}{A_{n-1}} \right)^2 = B_{n-1} + \frac{1}{4} \left(\frac{B_{n-1}}{A_{n-1}} \right)^2. \end{aligned}$$

Defining the sequence $\tilde{C}_n \triangleq \kappa \frac{B_n}{A_n}$, with $\kappa = \frac{1}{2}$, the above claim gives for all $n \geq 1$,

$$\tilde{C}_n \geq \tilde{C}_0 \cdot \gamma n^{-\frac{\lambda-\mu}{2\lambda-\mu}},$$

where $\lambda = \frac{1}{2}, \mu = \frac{1}{4}, \gamma = \exp\left(-\frac{2\lambda(\lambda-\mu)}{\mu(2\lambda-\mu)}\right) = e^{-4/3}$, and $\tilde{C}_0 = \frac{1}{2} \frac{B_0}{A_0} = \frac{1}{2} C_{k+1}$. Thus,

$$\tilde{C}_n \geq C_{k+1} \cdot \frac{e^{-4/3}}{2} n^{-1/3}, \quad \forall n \geq 1.$$

Finally, observe that $\tilde{C}_n \triangleq \frac{1}{2} \frac{B_n}{A_n} = \frac{\sum_{j=k+1-n}^{k+1} C_j^2}{2 \sum_{j=k+1-n}^{k+1} C_j} = C_{k-n}$, for all $0 \leq n \leq k-1$, and plugging in $n = k-1 \geq 1$, when $k \geq 2$, we get

$$C_1 \geq C_{k+1} \cdot \frac{e^{-4/3}}{2} (k-1)^{-1/3} \geq C_{k+1} \cdot \frac{e^{-4/3}}{2} k^{-1/3}.$$

Specifically, when $k = 1$, $C_1 = \frac{C_2}{2} = C_{k+1} \cdot \frac{1}{2} \geq C_{k+1} \cdot \frac{e^{-4/3}}{2} k^{-1/3}$, concluding the proof for all $k \geq 1$. $\square$

3.12.2 Regression experiments on single-pass vs. repetition

We extend the experiment on the effect of repetition (fig. 3.5) to additional data regimes. Figure 3.5 was produced using the same data as fig. 3.3, i.e., $d = 100, r = 10, T = 50$. Throughout this section, the Maximum Distance ordering (definition 3.7) is denoted by “Greedy” for brevity.

Isotropic data. We find that the conclusions of section 3.5.2 extend to more regimes: In the settings of our experiments, repetitions are beneficial in greedy ordering while replacement harms random ordering.

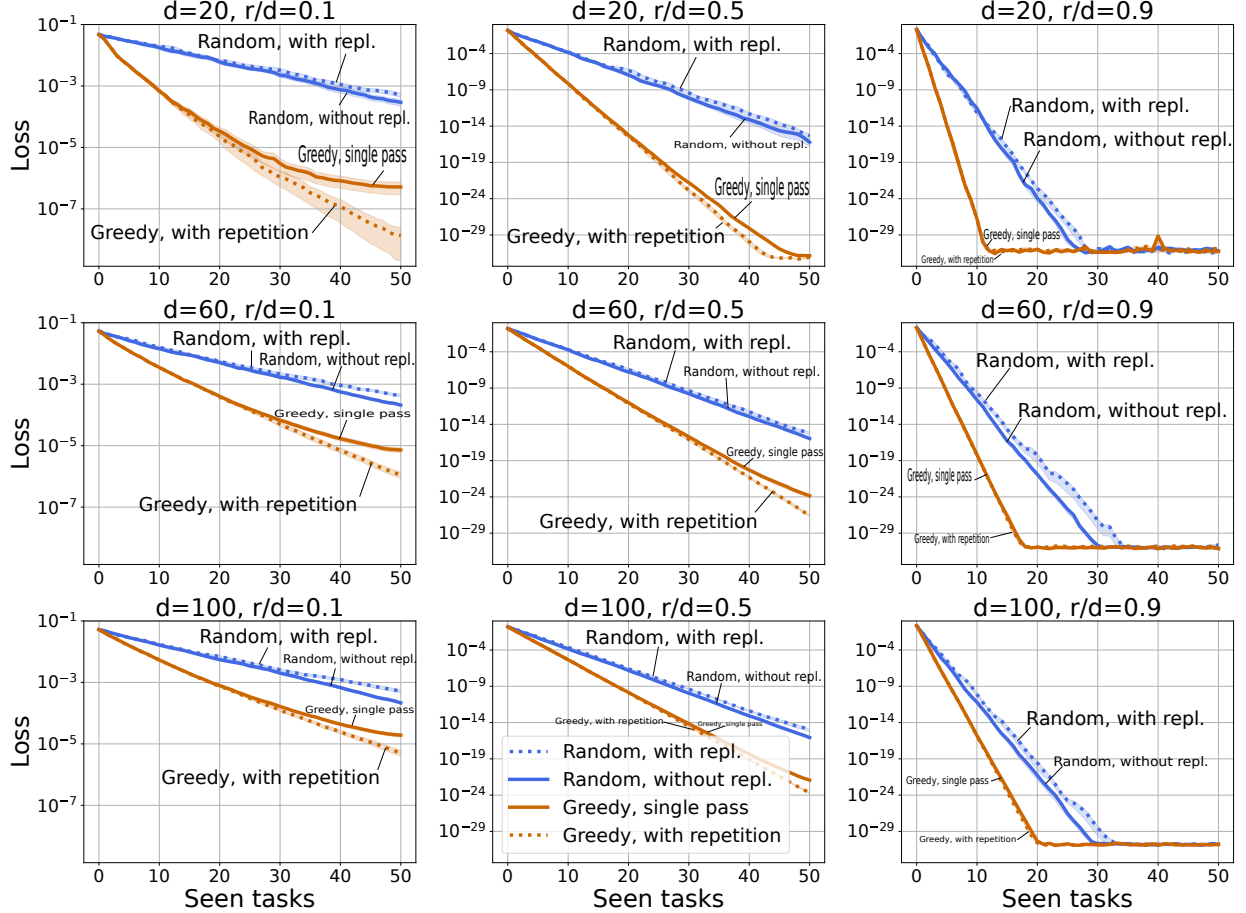


Figure 3.20: The effect of repetitions for varying dimensions d and ranks r of the data matrices, for isotropic data. $T = 50$. Random orderings without replacement consistently outperform their with-replacement counterparts. In contrast, greedy orderings with repetition outperform the single-pass variant. As explained in section 3.5.2, one can heuristically argue that repetition in greedy orderings is beneficial because it enables larger steps (and faster convergence to the joint solution $\mathbf{w}_\star$).

Anisotropic data. Next, we observe that the effect of repetitions diminishes for correlated data.

Below, we observe another aspect related to the “diminished” effect of repetitions in this setting.

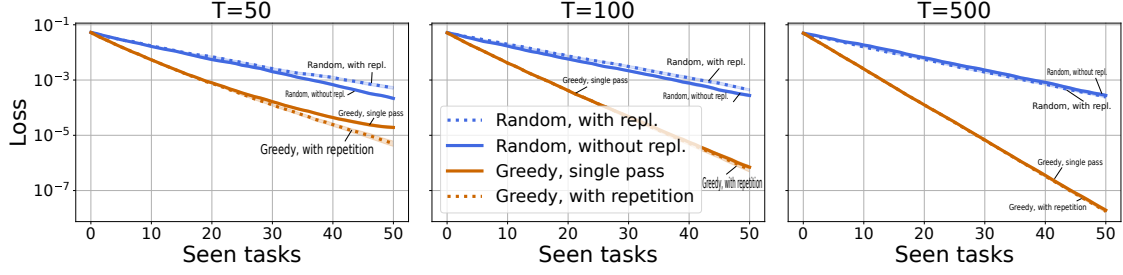


Figure 3.21: The effect of repetitions for varying task count T , for isotropic data. $d = 100, r = 10$. As the task count increases, the differences between with and without repetition diminish. However, in all subplots we only learn the first 50 tasks. In the left subplot, the effect of repetition becomes pronounced in the latter parts of the task sequences. Repetition offers less benefit when many diverse, unexplored tasks remain.

We omit the figure for the corresponding experiment with varying numbers of tasks T , as it offers no additional insights beyond those shown in fig. 3.21.

3.13 Appendix to section 3.5.3: Hybrid task ordering

3.13.1 Hybrid ordering scheme

Motivated by greedy Kaczmarz and importance sampling methods [4, 92], as well as recent convergence bounds for random orderings in continual learning [11, 43], we introduce a “hybrid” strategy in section 3.5.3. Hybrid schemes have also been explored in the contexts of Kaczmarz methods [32, 92], coordinate descent [44], and multiplicative Schwarz methods [48].

In this approach, tasks are selected greedily as long as the decrements $\|\mathbf{w}_{t-1} - \mathbf{w}_t\|^2$ (see eq. (3.3)) remain above a threshold; afterward, selection switches to random sampling. The proposed hybrid method can be used with either the greedy Maximum Distance rule (definition 3.7), as in scheme 3.3, or the greedy Maximum Residual rule (definition 3.8) as in scheme 3.4.

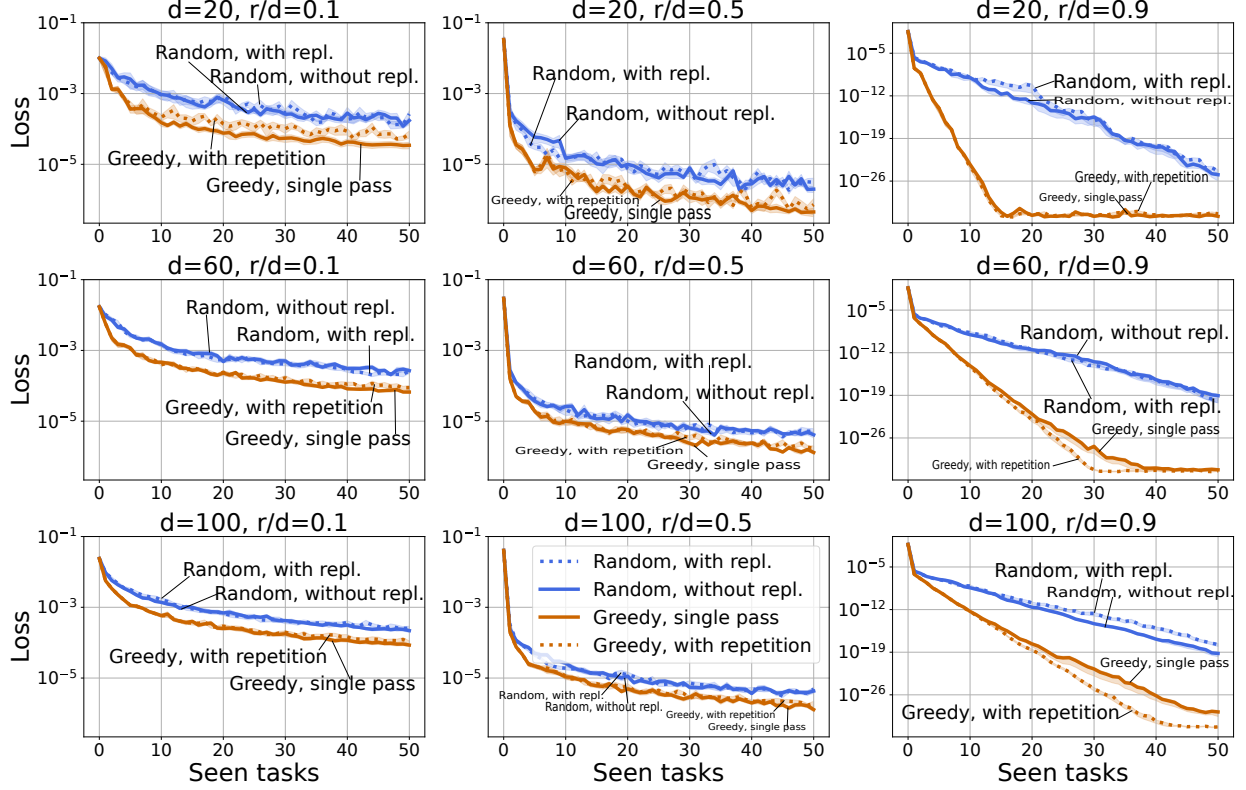


Figure 3.22: The effect of repetitions for varying dimensions d and ranks r of the data matrices, for anisotropic data. $T = 50$. Previously in section 3.8 (fig. 3.10), we explained that the performance of all ordering strategies deteriorates when the pairwise distances between task solution subspaces are small. This effect is even more pronounced for low-rank tasks (left columns), where the complementary high-rank solution subspaces overlap substantially. We also observe that in those low-rank regimes, different orderings exhibit more similar performance, and consequently, repetitions become less impactful. While we cannot fully explain the small performance degradation observed when allowing repetitions in greedy ordering for low-dimensional, low-rank settings (top-left subfigure), it may be related to the slower convergence to the joint solution $\mathbf{w}_*$, nullifying the effect of the loss upper bound induced by the distance to $\mathbf{w}_*$ (proposition 3.6; see also fig. 3.23 below).

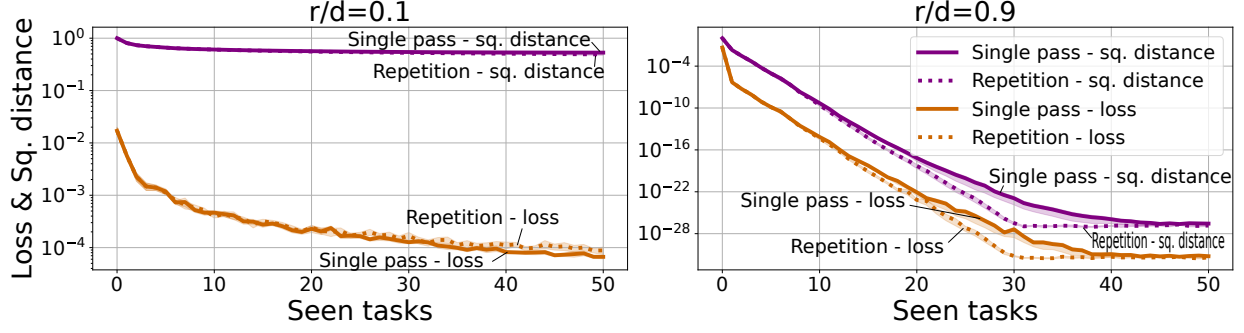


Figure 3.23: The distance to $\mathbf{w}_\star$ is a loose upper bound on the loss for high-similarity tasks. Greedy ordering with $d = 60$, $T = 50$. While repetitions do lead to a slightly faster decrease in the squared distance to $\mathbf{w}_\star$, this decrease remains slow when tasks are highly similar (as in the low-rank setting on the left). Consequently, the upper bound of proposition 3.6 becomes looser, as the loss itself decreases more rapidly. This discrepancy makes it difficult to draw firm conclusions about the convergence of the loss, including the exact impact of repetitions. A similar gap between the loss and the distance to the joint solution $\mathbf{w}_\star$ in highly similar tasks was also noted by Evron et al. [41, Section 5.1 therein].

Scheme 3.3 (MD hybrid ordering ($\tau_{\text{H-MD}}$)).

Input: $\beta_{\text{MD}} \in [0, \|\mathbf{w}_0 - \mathbf{w}_\star\|^2]$

For each iteration $t = 1, \dots, T$: # Use greedy selection as long as the threshold is met

$m' \leftarrow \operatorname{argmax}_{m \in [T] \setminus \tau_{\text{H-MD}}(1:t-1)} \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2$ # Compute greedy selection

If $\|(\mathbf{I} - \mathbf{P}_{m'}) (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \geq \beta_{\text{MD}}$ **Then** $\tau_{\text{H-MD}}(t) \leftarrow m'$ **Else Break**

$\tau_{\text{H-MD}}(t : T) \sim \text{Unif}([T] \setminus \tau_{\text{H-MD}}(1 : t - 1))$ # Choose remaining tasks randomly w/o replacement

Scheme 3.4 (MR hybrid ordering ($\tau_{\text{H-MR}}$)).

Input: $\beta_{\text{MR}} \in [0, R^2 \|\mathbf{w}_0 - \mathbf{w}_\star\|^2]$ # Reminder: $R \triangleq \max_{m \in [T]} \|\mathbf{X}_m\|$

For each iteration $t = 1, \dots, T$: # Use greedy selection as long as the threshold is met

$m' \leftarrow \operatorname{argmax}_{m \in [T] \setminus \tau_{\text{H-MR}}(1:t-1)} \|\mathbf{X}_m \mathbf{w}_{t-1} - \mathbf{y}_m\|^2$ # Compute greedy selection

If $\|\mathbf{X}_{m'} \mathbf{w}_{t-1} - \mathbf{y}_{m'}\|^2 \geq \beta_{\text{MR}}$ **Then** $\tau_{\text{H-MR}}(t) \leftarrow m'$ **Else Break**

$\tau_{\text{H-MR}}(t : T) \sim \text{Unif}([T] \setminus \tau_{\text{H-MR}}(1 : t - 1))$ # Choose remaining tasks randomly w/o replacement

While our analysis fixes the value of the threshold β in terms of $\|\mathbf{w}_0 - \mathbf{w}_\star\|$ and R , the hybrid methods can be used with any value of β .

Analytically, using a suitable threshold β , any upper bound for without-replacement random orderings, e.g., an $\mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$ bound [11], can extend to our hybrid schemes, avoiding the failure mode of section 3.5.1; see lemma 3.24. The way we derive our upper bound is by finding an upper bound at each iteration based on the current geometry, and this bound improves for each step as long as the step is large enough. So we set our stopping criterion to be when the step size stop being large enough for our bound on the final loss to improve.

3.13.2 Hybrid ordering upper bound

Lemma 3.24 (Hybrid ordering bound). Consider any known upper bound for the expected normalized loss (definition 3.3) in random ordering without replacement over T jointly-realizable tasks, of the form $\mathbb{E}_{\tau_{\text{Unif}}} [\mathcal{L}(\mathbf{w}_T^{\tau_{\text{Unif}}})] \leq \frac{C}{T^\alpha}$ with $C > 0$ and $0 < \alpha \leq 1$, such that $\frac{C}{T^\alpha} \leq \frac{1}{2-\alpha}$. Then, defining $\tilde{\beta}_{\min} \triangleq \frac{T^{\alpha-C(1-\alpha)}}{CT}$, the following holds. When $\beta_{\text{MD}} \geq \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 \tilde{\beta}_{\min}$ (or $\beta_{\text{MR}} \geq R^2 \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 \tilde{\beta}_{\min}$), the loss under scheme 3.3 (or scheme 3.4) is upper bounded as $\mathbb{E}_{\tau_{\text{H}}} [\mathcal{L}(\mathbf{w}_T^{\tau_{\text{H}}})] \leq \frac{C}{T^\alpha}$. Furthermore, choosing $\beta_{\text{MD}} = \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 \tilde{\beta}_{\min}$ (or $\beta_{\text{MR}} = R^2 \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 \tilde{\beta}_{\min}$), i.e., postponing the stopping time as much as our (data-dependent) condition allows, leads to the smallest upper bound in these derivations.

Proof. For MD and MR hybrid orderings, we denote $\beta_{\text{MD}} = \tilde{\beta} \|\mathbf{w}_0 - \mathbf{w}_\star\|^2$ and $\beta_{\text{MR}} = \tilde{\beta} R^2 \|\mathbf{w}_0 - \mathbf{w}_\star\|^2$, respectively. For all $m \in [T]$ and $\mathbf{w} \in \mathbb{R}^d$:

$$\|\mathbf{X}_m \mathbf{w} - \mathbf{y}_m\|^2 = \|\mathbf{X}_m (\mathbf{w} - \mathbf{w}_\star)\|^2 = \|\mathbf{X}_m \mathbf{X}_m^+ \mathbf{X}_m (\mathbf{w} - \mathbf{w}_\star)\|^2 \leq R^2 \|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w} - \mathbf{w}_\star)\|^2.$$

If $\|\mathbf{X}_m \mathbf{w}_{t-1} - \mathbf{y}_m\|^2 \geq \tilde{\beta} R^2 \|\mathbf{w}_0 - \mathbf{w}_\star\|^2$, then $\|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \geq \tilde{\beta} \|\mathbf{w}_0 - \mathbf{w}_\star\|^2$. Thus, if the condition for continuing with greedy MR steps in scheme 3.4 holds, $\|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_{t-1} -$

$\mathbf{w}_\star\|^2 \geq \tilde{\beta}\|\mathbf{w}_0 - \mathbf{w}_\star\|^2$. For scheme 3.3, if the condition for continuing with greedy holds then $\|(\mathbf{I} - \mathbf{P}_{m'}) (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \geq \tilde{\beta}\|\mathbf{w}_0 - \mathbf{w}_\star\|^2$ holds by definition.

The last step t for which $\max_{m \in [T] \setminus \tau(1:t-1)} \|(\mathbf{I} - \mathbf{P}_m) (\mathbf{w}_{t-1} - \mathbf{w}_\star)\|^2 \geq \tilde{\beta}\|\mathbf{w}_0 - \mathbf{w}_\star\|^2$ consecutively holds is some $t = s$, where $0 \leq s \leq T$. For this s ,

$$\|\mathbf{w}_s - \mathbf{w}_\star\|^2 = \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 - \sum_{t=1}^s \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 \leq \|\mathbf{w}_0 - \mathbf{w}_\star\|^2 (1 - \tilde{\beta}s).$$

Recall that the normalized loss for a solution vector $\mathbf{w}$ with task collection $\mathcal{T}$, starting point $\mathbf{w}_0$, and minimum norm joint solution $\mathbf{w}_\star$ is

$$\mathcal{L}^{(\mathcal{T}, \mathbf{w}_0)}[\mathbf{w}] \triangleq \frac{1}{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2 R^2} \frac{1}{|\mathcal{T}|} \sum_{m \in \mathcal{T}} \|\mathbf{X}_m (\mathbf{w} - \mathbf{w}_\star)\|^2.$$

Running the hybrid scheme on the task collection $[T]$ yields the following expected loss:

$$\begin{aligned} \mathbb{E}_\tau \mathcal{L}^{([T], \mathbf{w}_0)}[\mathbf{w}_T] &= \frac{1}{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2 R^2} \frac{1}{T} \sum_{m=1}^T \mathbb{E} [\|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2] \\ &= \frac{1}{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2 R^2} \frac{1}{T} \left[\sum_{t=1}^s \mathbb{E} [\|\mathbf{X}_{\tau(t)} (\mathbf{w}_T - \mathbf{w}_\star)\|^2] + \sum_{m \in [T] \setminus \tau(1:s)} \mathbb{E} [\|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2] \right] \\ &\leq \frac{1}{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2 R^2} \frac{1}{T} \left[R^2 \sum_{t=1}^s \mathbb{E} [\|\mathbf{w}_T - \mathbf{w}_\star\|^2] + \sum_{m \in [T] \setminus \tau(1:s)} \mathbb{E} [\|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2] \right] \\ &\stackrel{(1)}{\leq} \frac{1}{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2 R^2} \frac{1}{T} \left[R^2 \sum_{t=1}^s \mathbb{E} [\|\mathbf{w}_s - \mathbf{w}_\star\|^2] + \sum_{m \in [T] \setminus \tau(1:s)} \mathbb{E} [\|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2] \right] \\ &\stackrel{(2)}{=} \frac{1}{\|\mathbf{w}_0 - \mathbf{w}_\star\|^2 R^2} \frac{1}{T} \left[R^2 s \|\mathbf{w}_s - \mathbf{w}_\star\|^2 + \sum_{m \in [T] \setminus \tau(1:s)} \mathbb{E} [\|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2] \right] \\ &= \frac{\|\mathbf{w}_s - \mathbf{w}_\star\|^2}{T \|\mathbf{w}_0 - \mathbf{w}_\star\|^2} \left(s + (T-s) \left(\frac{1}{\|\mathbf{w}_s - \mathbf{w}_\star\|^2 R^2} \frac{1}{T-s} \sum_{m \in [T] \setminus \tau(1:s)} \mathbb{E} [\|\mathbf{X}_m (\mathbf{w}_T - \mathbf{w}_\star)\|^2] \right) \right) \\ &= \frac{\|\mathbf{w}_s - \mathbf{w}_\star\|^2}{T \|\mathbf{w}_0 - \mathbf{w}_\star\|^2} (s + (T-s) \mathbb{E}_\tau \mathcal{L}^{([T] \setminus \tau(1:s), \mathbf{w}_s)}[\mathbf{w}_T]) \\ &\leq \frac{1 - \tilde{\beta}s}{T} (s + (T-s) \mathbb{E}_\tau \mathcal{L}^{([T] \setminus \tau(1:s), \mathbf{w}_s)}[\mathbf{w}_T]), \end{aligned}$$

where (1) is since $s \leq T$, and (2) is since $\mathbf{w}_s$ is deterministic. This means we can plug in any

upper bound for the expected normalized loss of the random ordering, for the collection of $T - s$ tasks $[T] \setminus \tau(1 : s)$ with the starting point $\mathbf{w}_s$, replacing dependence on T with $T - s$. If we have an upper bound $f(T)$ for the expected normalized loss of random ordering of T tasks, where f is positive and decreasing, we obtain:

$$\mathbb{E}_{\tau} \mathcal{L}^{([T], \mathbf{w}_0)} [\mathbf{w}_T] \leq \frac{1 - \tilde{\beta}s}{T} (s + (T - s)f(T - s)). \quad (3.11)$$

As a sanity check, setting $s = 0$ removes the greedy iterates, and the bound reduces to that of random ordering.

For the rest of the proof, we work with bounds of the following form:

$$f(T) = \begin{cases} \frac{C}{T^\alpha} & \frac{C}{T^\alpha} \leq 1 \\ 1 & \text{else} \end{cases}, \quad (3.12)$$

such that $f(T) \leq 1$ for all C, α, T . This excludes cases where the random-ordering bound is larger than the trivial bound 1.

We want a condition on $\tilde{\beta}$ for which continuing with greedy iterates as long as $\|(\mathbf{I} - \mathbf{P}_m)(\mathbf{w}_{t-1} - \mathbf{w}_*)\|^2 \geq \tilde{\beta}\|\mathbf{w}_0 - \mathbf{w}_*\|^2$, necessarily improves the bound. This means we want the bound to decrease with s . Thus, we demand $\forall s \in [T] : \frac{d}{ds} \left(\frac{1 - \tilde{\beta}s}{T} (s + (T - s)f(T - s)) \right) \leq 0$:

$$\begin{aligned} & \frac{d}{ds} \left(\frac{1 - \tilde{\beta}s}{T} (s + (T - s)f(T - s)) \right) \\ &= \frac{1}{T} \left(-\tilde{\beta}(s + (T - s)f(T - s)) + (1 - \tilde{\beta}s)(1 + (-f(T - s) - (T - s)f'(T - s))) \right) \\ &= \frac{1}{T} \left(-\tilde{\beta}s - \tilde{\beta}Tf(T - s) + \tilde{\beta}s f(T - s) + 1 - f(T - s) - (T - s)f'(T - s) - \tilde{\beta}s \right. \\ & \quad \left. + \tilde{\beta}s f(T - s) + \tilde{\beta}s(T - s)f'(T - s) \right) \\ &= \frac{1}{T} \left(1 - 2\tilde{\beta}s - (1 + \tilde{\beta}T - 2\tilde{\beta}s)f(T - s) - (1 - \tilde{\beta}s)(T - s)f'(T - s) \right). \end{aligned}$$

Requiring this expression to be ≤ 0 gives:

$$\begin{aligned}\tilde{\beta}(-2s - (T - 2s)f(T - s) + s(T - s)f'(T - s)) &\leq -1 + f(T - s) + (T - s)f'(T - s) \\ \tilde{\beta} &\geq \frac{1 - f(T - s) - (T - s)f'(T - s)}{Tf(T - s) + 2s(1 - f(T - s)) - s(T - s)f'(T - s)}.\end{aligned}$$

As a sanity check, since $f(T - s) \leq 1$ and $f'(T - s) \leq 0$, the numerator is non-negative and the denominator is positive.

Continuing,

$$\begin{aligned}\tilde{\beta} &\geq \frac{1 - f(T - s) - (T - s)f'(T - s)}{Tf(T - s) + 2s(1 - f(T - s)) - s(T - s)f'(T - s)} \\ &= \frac{1 - f(T - s) - (T - s)f'(T - s)}{s(1 - f(T - s) - (T - s)f'(T - s)) - s + Tf(T - s) + s - sf(T - s)} \\ &= \left(s + \frac{(T - s)f(T - s)}{1 - f(T - s) - (T - s)f'(T - s)} \right)^{-1} \\ \tilde{\beta}^{-1} &\leq s + \frac{(T - s)f(T - s)}{1 - f(T - s) - (T - s)f'(T - s)}.\end{aligned}$$

We demand this holds for all $s \in [T]$. First assume $\frac{C}{(T-s)^\alpha} \leq 1$; We consider other case is later. Plugging in $f(T - s) = \frac{C}{(T-s)^\alpha}$:

$$\begin{aligned}\tilde{\beta}^{-1} &\leq s + \frac{(T - s)\frac{C}{(T-s)^\alpha}}{1 - \frac{C}{(T-s)^\alpha} + (T - s)\frac{\alpha C}{(T-s)^{\alpha+1}}} = s + \frac{C(T - s)^{1-\alpha}}{1 - \frac{C}{(T-s)^\alpha} + \frac{\alpha C}{(T-s)^\alpha}} \\ &= s + \frac{C(T - s)}{(T - s)^\alpha \left(1 - \frac{C(1-\alpha)}{(T-s)^\alpha}\right)} = s + \frac{C(T - s)}{(T - s)^\alpha - C(1 - \alpha)} \triangleq g(s).\end{aligned}$$

Under our assumption $\frac{C}{(T-s)^\alpha} \leq 1$, $(T - s)^\alpha \geq C > C(1 - \alpha)$, so the denominator is positive.

Denote $\tilde{\beta}^{-1} \leq g(s) \triangleq s + \frac{C(T-s)}{(T-s)^\alpha - C(1-\alpha)}$. To find an upper bound on $\tilde{\beta}^{-1}$ that holds for all

s , we look for the minimum of $g(s)$. Differentiating,

$$\begin{aligned}
\frac{dg(s)}{ds} &= 1 + \frac{-C((T-s)^\alpha - C(1-\alpha)) - C(T-s)(-\alpha(T-s)^{\alpha-1})}{((T-s)^\alpha - C(1-\alpha))^2} \\
&= 1 - C \frac{-\alpha(T-s)^\alpha + (T-s)^\alpha - C(1-\alpha)}{((T-s)^\alpha - C(1-\alpha))^2} \\
&= 1 + \alpha C \frac{(T-s)^\alpha}{((T-s)^\alpha - C(1-\alpha))^2} - C \frac{1}{(T-s)^\alpha - C(1-\alpha)} \\
&\geq 1 + \alpha C \frac{(T-s)^\alpha - C(1-\alpha)}{((T-s)^\alpha - C(1-\alpha))^2} - C \frac{1}{(T-s)^\alpha - C(1-\alpha)} \\
&= 1 - \frac{C(1-\alpha)}{(T-s)^\alpha - C(1-\alpha)} = \frac{(T-s)^\alpha - 2C(1-\alpha)}{(T-s)^\alpha - C(1-\alpha)}.
\end{aligned}$$

Hence $g'(s) \geq 0$ for s small enough such that $(T-s)^\alpha \geq 2C(1-\alpha)$, and $g'(s) < 0$ for larger values, up to the maximum value under the current assumption of $s = T - C^{1/\alpha}$. Hence, the minimum of $g(s)$ in $[0, T - C^{1/\alpha}]$ will be one of the boundary points:

$$\begin{aligned}
g(T - C^{1/\alpha}) &= T - C^{1/\alpha} + \frac{C \cdot C^{1/\alpha}}{C - C(1-\alpha)} = T - C^{1/\alpha} + \frac{C^{1/\alpha}}{\alpha} \\
&= T + C^{1/\alpha}(\alpha^{-1} - 1) \geq T \\
g(0) &= \frac{CT}{T^\alpha - C(1-\alpha)} \\
g(T - C^{1/\alpha}) - g(0) &\geq T \left(1 - \frac{C}{T^\alpha - C(1-\alpha)} \right) = T \left(\frac{T^\alpha - C(2-\alpha)}{T^\alpha - C(1-\alpha)} \right).
\end{aligned}$$

This can only be negative when $T < (C(2-\alpha))^{1/\alpha}$, i.e., $f(T) = \frac{C}{T^\alpha} > \frac{1}{2-\alpha}$, which is not covered by this lemma, since we assumed $\frac{C}{T^\alpha} \leq \frac{1}{2-\alpha}$ (and losing this regime is not too bad because in this regime, you are trying to plug in a bound for random that is $> 1/2$ which is a very weak bound). Thus, the lowest upper bound for $\tilde{\beta}^{-1}$ occurs at $s = 0$, and we get

$$\begin{aligned}
\tilde{\beta}^{-1} &\leq \frac{CT}{T^\alpha - C(1-\alpha)} \\
\tilde{\beta} &\geq \tilde{\beta}_{\min} = \frac{T^\alpha - C(1-\alpha)}{CT}.
\end{aligned}$$

Under this choice of $\tilde{\beta}$, the upper bound of eq. (3.11) is monotonically decreasing with s as long as $\frac{C}{(T-s)^\alpha} \leq 1$.

We now address the case of s large enough that $\frac{C}{(T-s)^\alpha} > 1$. In this case, our upper bound from eq. (3.11) becomes: $\mathbb{E}_\tau \mathcal{L}^{([T], \mathbf{w}_0)}[\mathbf{w}_T] \leq \frac{1-\tilde{\beta}s}{T} (s + (T-s) \cdot 1) = 1 - \tilde{\beta}s$, which is also monotonically decreasing with s . From continuity of the bound at $s = T - C^{1/\alpha}$ (it does not matter that this might not be an integer), we get that the bound is monotonically decreasing with s for all $s > 0$ when $\tilde{\beta} \geq \tilde{\beta}_{\min} = \frac{T^\alpha - C(1-\alpha)}{CT}$, and is therefore no larger than the random ordering bound at $s = 0$. $\square$

3.13.3 Hybrid ordering experiments

fig. 3.6 was acquired using the same data as fig. 3.3, and using the dimension and rank-dependent upper bound of $\frac{2(d-r)}{k}$ from Evron et al. [43] to set β , since the universal bound of $\frac{14}{k^{1/4}}$ requires more than 50 iterations to be effective. The hybrid method yields intermediate performance between random and greedy. The figures combine trends reported earlier (section 3.8) for random and greedy MD, in terms of the effect of dimension, rank, task count and task correlation on the performance.

Anisotropic data. Similar trends were observed under anisotropic data, and we therefore omit the corresponding figures for brevity.

3.14 Appendix to section 3.4.1: Code snippet for regression experiments

For completeness and reproducibility, the following code snippet generates a linear regression experiment on isotropic data, comparing random and greedy orderings (eq. (3.4), definitions 3.7 and 3.8). Small modifications of this with seeds $1, \dots, 10$ reproduce some of the figures in this paper.

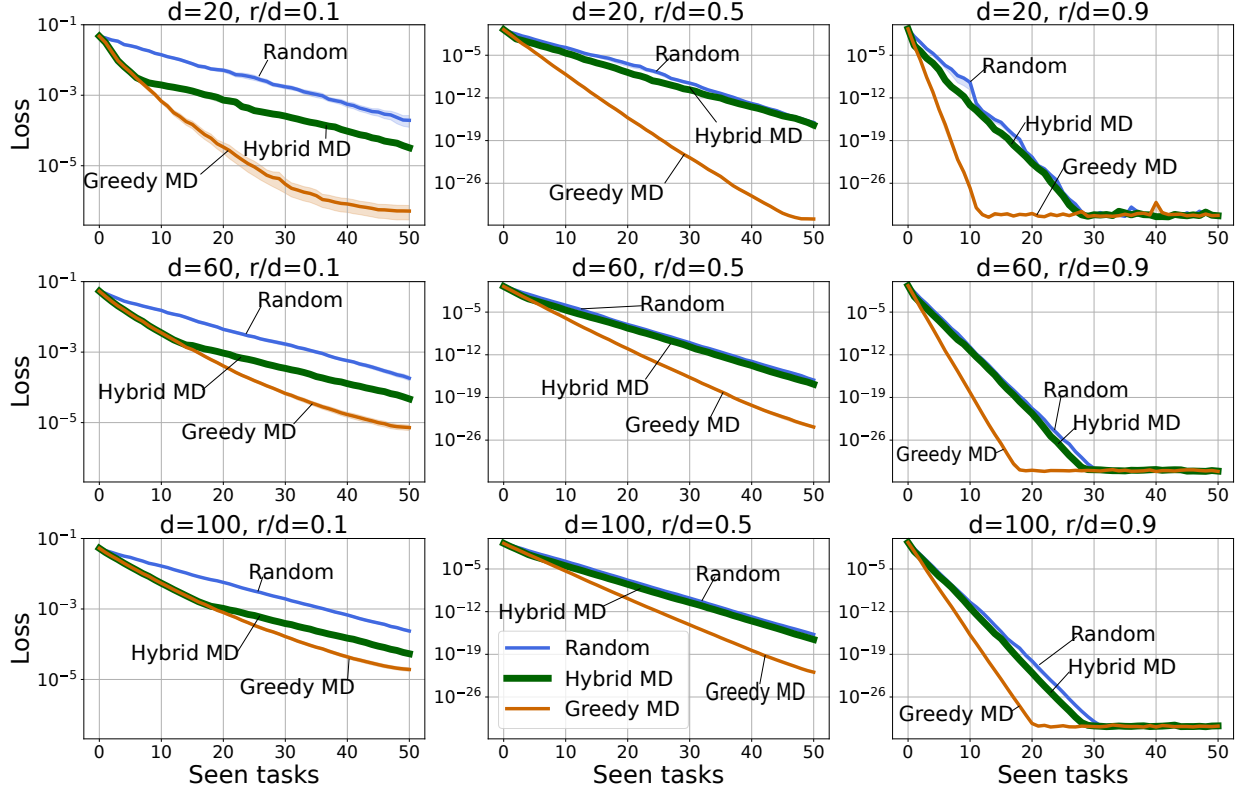


Figure 3.24: **Hybrid performance for varying dimensions d and ranks r of the data matrices, for isotropic data.** $T = 50$. In high-rank and/or low-dimensional settings, the rank-dependent upper bound employed by the hybrid strategy in this case is lower, prompting an earlier transition from the greedy to the random phase. For $r/d = 0.1$, the random phase within the hybrid method has slightly lower slope than fully random ordering (so the loss decreases slower), possibly because the initial greedy steps remove tasks with large decrements from the remaining task set. The slopes are hard to compare for $r/d = 0.5$ and very slightly faster for $r/d = 0.9$.

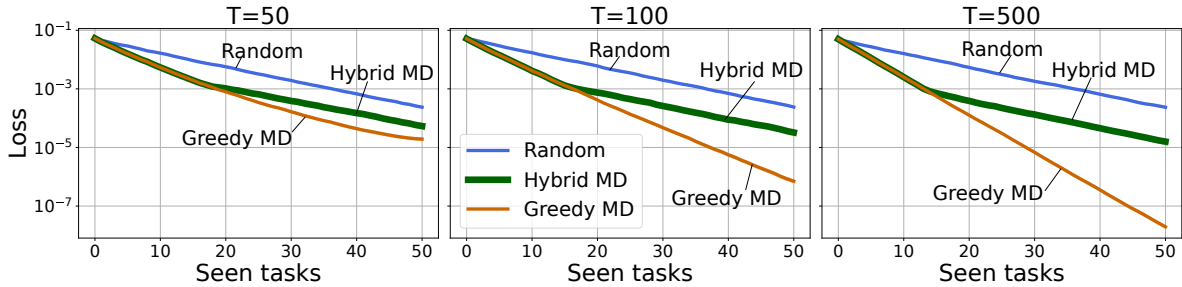


Figure 3.25: **Hybrid performance for varying task count T , for isotropic data.** $d = 100$, $r = 10$. We see similar trends. The previously observed difference between the random phase following the greedy phase and fully random ordering is smaller at higher task counts, possibly because more tasks with large decrements remain available for selection.

Data generation

```
# Minimal Block Kaczmarz experiment with a runnable demo and plot

import numpy as np
import matplotlib.pyplot as plt
from numpy.linalg import pinv
from typing import List, Tuple, Optional

# ----- Core utilities -----

def generate_data(r: int, d: int, T: int, seed: Optional[int] = None) -> Tuple[
    np.ndarray, np.ndarray, np.ndarray, List[tuple], List[np.ndarray],
    List[np.ndarray]]:
    """
    Returns:
        X: (T*r, d) matrix; b: (T*r,) labels; w_true: (d,) teacher.
        blocks: list of (X_t, b_t) with X_t in  $\mathbb{R}^{r \times d}$ , b_t in  $\mathbb{R}^r$ .
        pinv_blocks: list of pseudoinverses  $X_t^+$ .
        md_proj: list of  $X_t^+ X_t$  (used by MD-based selection).
    """
    if seed is not None:
        np.random.seed(seed)

    X_blocks = [np.random.randn(r, d) for _ in range(T)]
    max_rad = max(np.linalg.norm(B, 2) for B in X_blocks)
    X_blocks = [B / max_rad for B in X_blocks]

    w_true = np.random.randn(d)
    w_true /= np.linalg.norm(w_true)
```

```

X = np.vstack(X_blocks)

b = X @ w_true

X_blocks = [X[i*r:(i+1)*r, :] for i in range(T)]
b_blocks = [b[i*r:(i+1)*r] for i in range(T)]
pinv_blocks = [pinv(Xt) for Xt in X_blocks]
md_proj = [pinv_blocks[t] @ X_blocks[t] for t in range(T)]
blocks = list(zip(X_blocks, b_blocks))

return X, b, w_true, blocks, pinv_blocks, md_proj

```

Ordering strategies

```

def _pick_uniform(
    T: int, used: Optional[List[int]], allow_repetition: bool
) -> Optional[int]:
    if allow_repetition:
        return int(np.random.randint(0, T))
    pool = list(set(range(T)) - set(used or []))
    return int(np.random.choice(pool)) if pool else None

def _pick_greedy_mr(
    blocks: List[tuple], w: np.ndarray, used: Optional[List[int]],
    allow_repetition: bool
) -> Optional[int]:
    best, best_val = None, -np.inf
    for i, (Xt, bt) in enumerate(blocks):
        if (not allow_repetition) and i in used:
            continue
        val = np.linalg.norm(Xt @ w - bt) ** 2

```



```

        if val >= best_val:
            best, best_val = i, val
    return best

def _pick_greedy_md(
    md_proj: List[np.ndarray], w: np.ndarray, w_true: np.ndarray,
    used: Optional[List[int]], allow_repetition: bool
) -> Optional[int]:
    best, best_val = None, -np.inf
    for i, P in enumerate(md_proj):
        if (not allow_repetition) and used and i in used:
            continue
        val = np.linalg.norm(P @ (w - w_true)) ** 2
        if val >= best_val:
            best, best_val = i, val
    return best

```

Experiment loop

```

def block_kaczmarz(
    blocks: List[tuple],
    pinv_blocks: List[np.ndarray],
    md_proj: List[np.ndarray],
    w_true: np.ndarray,
    T: int,
    d: int,
    max_iters: int,
    selection: str = "uniform", # one of {"uniform", "greedy_mr", "greedy_md"}
    allow_repetition: bool = False,
) -> Tuple[np.ndarray, List[float], List[float]]:

```

```

"""Runs Block Kaczmarz and returns (w, out_losses, dist_to_teacher) per
iteration."""
w = np.zeros(d)
out_losses: List[float] = []
dist: List[float] = []
used: List[int] = []

for _ in range(max_iters):
    if selection == "uniform":
        t = _pick_uniform(T, used, allow_repetition)
    elif selection == "greedy_mr":
        t = _pick_greedy_mr(blocks, w, used, allow_repetition)
    elif selection == "greedy_md":
        t = _pick_greedy_md(md_proj, w, w_true, used, allow_repetition)
    else:
        raise ValueError("selection must be one of
            {'uniform', 'greedy_mr', 'greedy_md'}"
        )

    if t is None: # no available block under no-replacement
        out_losses.append(np.nan)
        dist.append(np.linalg.norm(w - w_true) ** 2)
        continue

    Xt, bt = blocks[t]
    w += pinv_blocks[t] @ (bt - Xt @ w) # single block least squares step

    if not allow_repetition:

```

```

        used.append(t)

    # Metrics

    loss = 0.0

    for Xs, bs in blocks:

        r = Xs @ w - bs

        loss += np.linalg.norm(r) ** 2

    out_losses.append(loss / T)

    dist.append(np.linalg.norm(w - w_true) ** 2)

return w, out_losses, dist

```

Ordering comparison experiment

```

# One reproducible experiment and comparison figure

# Problem size / schedule
d = 100

r = max(1, int(0.1 * d)) # rows per block (r/d = 0.1)

T = 50 # number of blocks

max_iters = T # single pass without replacement

seed = 42 # reproducible

# Data and precomputations
_, _, w_true, blocks, pinv_blocks, md_proj = generate_data(r, d, T, seed=seed)

# Compare three orderings without replacement
strategies = [
    ("Random", {"selection": "uniform", "allow_repetition": False}),
    ("Greedy MR", {"selection": "greedy_mr", "allow_repetition": False}),

```

```

    ("Greedy MD", {"selection": "greedy_md", "allow_repetition": False}),
]

results = {}

for name, opts in strategies:
    _, out_losses, dist = block_kaczmarz(
        blocks=blocks,
        pinv_blocks=pinv_blocks,
        md_proj=md_proj,
        w_true=w_true,
        T=T,
        d=d,
        max_iters=max_iters,
        **opts
    )
    results[name] = (np.asarray(out_losses), np.asarray(dist))

# Print final metrics
print("Final loss and distance-to-teacher:")

for name in strategies:
    key = name[0]
    L, D = results[key]
    # Pick the last non-nan value
    last_idx = np.where(~np.isnan(L))[0][-1]
    print(f" {key:10s}  loss={L[last_idx]:.4e}  dist={D[last_idx]:.4e}")

# Comparison figure
plt.figure(figsize=(6, 4))

```

```

xs = np.arange(1, max_iters + 1)
for name, (L, _) in results.items():
    plt.plot(xs, L[:max_iters], label=name)
plt.xlabel("Iterations (seen tasks)")
plt.ylabel("Average loss")
plt.yscale("log")
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()

```

3.15 Lower bound technical appendix: Delta positivity proof

This section supplements section 3.11.1, which we recommend reviewing in advance. Here, we prove that $\forall d \geq 25,000, \forall t \in \{2, \dots, d-1\}, \forall k \in \{t, \dots, d-1\}$,

$$\Delta_{t,k} \triangleq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} > 0.$$

In some places in our proofs, we will need a closed-form approximation of the first coordinates $x_k \triangleq (\mathbf{w}_k)_1$ which we obtain recursively. Such an approximation was suggested in Hucht [59]:

$$\tilde{x}_k = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\frac{k}{d}} \left(\frac{1}{\ln 4} - \frac{k}{d} \right)}.$$

This will be formalized and proven in section 3.16. In addition, this gives a lower bound $x_k \geq 0.45, \forall k \in [d]$ when $d \geq 25,000$ (corollary 3.48).

3.15.1 Proof outline

The proof is as follows: we decompose $\Delta_{t,k}$ to smaller parts, and lower bound each of these parts. We then combine all of these lower bounds to achieve an overall lower bound on $\Delta_{t,k}$ and find a sufficient condition on d for which this lower bound is positive. This condition, revealed in eq. (3.13), is satisfied when $d \geq 25,000$, concluding the proof. We begin by bounding some intermediate quantities that appear later in the derivation, and starting in section 3.15.3 we decompose and lower bound $\Delta_{t,k}$.

3.15.2 Auxiliary: Algebraic inequalities

Claim 3.25. $\forall d \in \mathbb{N}$ and $1 \leq n \leq d$, it holds that $1 - c^n \triangleq 1 - 2^{-n/d} \in [\frac{n \ln(2)}{d} - \frac{n^2 \ln^2(2)}{2d^2}, \frac{n \ln(2)}{d}]$. Particularly, this shows $1 - c \in [\frac{\ln(2)}{d} - \frac{\ln^2(2)}{2d^2}, \frac{\ln(2)}{d}]$.

Proof. To show the upper bound, we define $\alpha = n/d \in (0, 1]$ and $f(\alpha) = 1 - 2^{-\alpha} - \alpha \ln(2)$, and f is decreasing in $(0, 1]$ since

$$f'(\alpha) = (2^{-\alpha} - 1) \ln(2) \propto 2^{-\alpha} - 1 < 0, \forall \alpha \in (0, 1].$$

Then, this means $f(\alpha) = 1 - 2^{-n/d} - \frac{n \ln(2)}{d} \leq \lim_{\alpha \rightarrow 0^+} f(\alpha) = 0$ as required.

Conversely, the lower bound follows by showing that the function $g(\alpha = \frac{n}{d}) = 1 - 2^{-n/d} - (\frac{n \ln(2)}{d} - \frac{n^2 \ln^2(2)}{2d^2})$ is increasing in $(0, 1]$,

$$g(\alpha) = 1 - 2^{-\alpha} - \left(\alpha \ln(2) - \frac{\alpha^2 \ln^2(2)}{2} \right), \lim_{\alpha \rightarrow 0^+} g(0) = 1 - 2^{-0} - 0 = 0,$$

$$g'(\alpha) = \ln(2)(2^{-\alpha} + \alpha \ln(2) - 1) \propto 2^{-\alpha} + \alpha \ln(2) - 1 = -f(\alpha) + 1$$

$$\geq - \lim_{\alpha \rightarrow 0^+} f(\alpha) + 1 = -(1 - 2^{-0} - 0) + 1 = 1 > 0.$$

□

Claim 3.26. For $\forall d, n, m \in \mathbb{N}$ and $k \in [d]$, it holds that $c^{nk-m} \geq 2^{-n}$.

Proof. The function $c^z = 2^{-z/d}$ is decreasing with z . Plugging in $z = nk - m \leq nd$ gives $c^z \geq c^{nd} = 2^{-n}$. $\square$

Claim 3.27. $\forall k \in [1, d]$ it holds that $1 - (1 - c)(k - 1) = ((k - 1)c - (k - 2)) \in [0, 1]$.

Proof. $(1 - c)(k - 1) \triangleq (1 - 2^{-1/d})(k - 1) \geq 0$. From claim 3.25:

$$\underbrace{(1 - c)(k - 1)}_{\geq 0} \leq (1 - c)(d - 1) \leq \frac{\ln(2)}{d}(d - 1) < \ln(2) < 1.$$

$\square$

Claim 3.28. $\forall k \in [1, d]$ it holds that $kc - (k - 1) > 0$.

Proof. From claim 3.25, $1 - c \leq \frac{\ln 2}{d} \Rightarrow c \geq 1 - \frac{\ln 2}{d}$. Substituting gives:

$$kc - (k - 1) \geq k \left(1 - \frac{\ln 2}{d}\right) - k + 1 = 1 - \ln 2 \frac{k}{d} \geq 1 - \ln 2 > 0.$$

$\square$

Claim 3.29. $\forall k \in [d]$ it holds that $\beta_k \in \left[\frac{0.3c^{2k-5}}{d}, \frac{c^{2k-5}}{d}\right]$.

Proof.

$$\beta_k = \frac{((k - 1)c - (k - 2))c^{2k-5}}{d} = \left(1 - \underbrace{(1 - c)(k - 1)}_{\in [0, \ln(2)] \subset [0, 0.7]}\right) \cdot \frac{c^{2k-5}}{d} \in \left[\frac{0.3c^{2k-5}}{d}, \frac{c^{2k-5}}{d}\right].$$

$\square$

Claim 3.30. x_k is decreasing and $\forall k \in [d]$, $x_k \leq 1$.

Proof. Decreasing follows immediately from positivity of β_k (see claim 3.29) and the construction, and since $x_1 = 1$, $\forall k \in [d]$, $x_k \leq 1$. $\square$

Claim 3.31. $\forall k \in [2, d]$ it holds that $\beta_k \leq \frac{1}{cd}$.

Proof. Since $c < 1$, $\beta_k \leq \frac{c^{2k-5}}{d} \leq \frac{c^{2 \cdot 2-5}}{d} = \frac{1}{cd}$. □

Claim 3.32. $\forall a > 0, b \in \mathbb{R} \setminus \{0\}$ such that $a + b \geq 0$, it holds that $\sqrt{a+b} < \sqrt{a} + \frac{b}{2\sqrt{a}}$.

Proof. $0 < b^2 \iff 4a(a+b) < 4a^2 + 4ab + b^2 = (2a+b)^2 \iff \sqrt{a+b} < \sqrt{a} + \frac{b}{2\sqrt{a}}$. □

Claim 3.33. $\forall d \geq 1 : 2^{1/d} \geq 1 + \frac{\ln 2}{d}$.

Proof. Using Taylor's expansion: $2^{1/d} = e^{\frac{\ln 2}{d}} = 1 + \frac{\ln 2}{d} + \sum_{i=2}^{\infty} \frac{1}{i!} \left(\frac{\ln 2}{d} \right)^i \geq 1 + \frac{\ln 2}{d}$. □

Claim 3.34. If $|x_k - \tilde{x}_k| \leq \epsilon$ and $x_k \geq 0$, $|x_k^2 - \tilde{x}_k^2| \leq 2x_k\epsilon_d + \epsilon_d^2$.

Proof. Defining $r = \tilde{x}_k - x_k$,

$$\begin{aligned} |x_k^2 - \tilde{x}_k^2| &= |x_k^2 - (x_k + r)^2| = |2x_k r - r^2| \leq |2x_k r| + r^2 = 2|x_k(\tilde{x}_k - x_k)| + (\tilde{x}_k - x_k)^2 \\ &\leq 2x_k\epsilon + \epsilon^2. \end{aligned}$$

□

3.15.3 Proof body

Analyzing $x_{k-1} - x_k, (x_{k-1} - x_k)^2, \frac{x_k}{x_{k-1}}$

Proposition 3.35. For any $k \geq 2$, it holds that,

$$\begin{aligned} x_{k-1} - x_k &\triangleq f_{x_{k-1}}(\beta_k) \in \left[\frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{2\beta_k^3}{x_{k-1}^5}, \frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{2\beta_k^3}{(x_{k-1}^2 - 4\beta_k)^{5/2}} \right] \\ [\text{when } d \geq 25,000] &\subseteq \left[\frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3}, \frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{113c^{6k-15}}{d^3} \right]. \end{aligned}$$

Proof. By construction,

$$x_{k-1} - x_k = \frac{x_{k-1} - \sqrt{x_{k-1}^2 - 4\beta_k}}{2}.$$

Define $f_z(x) = \frac{1}{2} \left(z - \sqrt{z^2 - 4x} \right)$ for $z \in [0.45, 1]$ (see claim 3.30, corollary 3.48) and $z^2 \gg x > 0$. Expand with Taylor:

$$\begin{aligned} f_z(0) &= f_z(0) \\ f_z^{(1)}(x) &= -\frac{1}{4} \frac{-4}{\sqrt{z^2 - 4x}} = \frac{1}{\sqrt{z^2 - 4x}} &\implies f_z^{(1)}(0) &= \frac{1}{z}, \\ f_z^{(2)}(x) &= 2(z^2 - 4x)^{-3/2} &\implies f_z^{(2)}(0) &= \frac{2}{z^3}, \\ f_z^{(3)}(x) &= 2 \frac{3}{2} \cdot 4(z^2 - 4x)^{-5/2} = 12(z^2 - 4x)^{-5/2} &\implies f_z^{(3)}(0) &= \frac{12}{z^5}, \end{aligned}$$

and generally $\forall z^2 \gg x > 0$, $f_z^{(n)}(x) > 0$.

Then, by Lagrange's form of the remainder, the error of the quadratic approximation (around $x = 0$) is given by

$$f_z(x) = \frac{f(0)}{0!}x^0 + \frac{f^{(1)}(0)}{1!}x^1 + \frac{f^{(2)}(0)}{2!}x^2 + R_2(x) = \frac{x}{z} + \frac{x^2}{z^3} + R_2(x),$$

where

$$R_2(x) = \frac{f^{(3)}(x_0)}{3!}(x - 0)^3 = \frac{12(z^2 - 4x_0)^{-5/2}}{6}x^3 \in \left[\frac{2x^3}{z^5}, \frac{2x^3}{(z^2 - 4x)^{5/2}} \right].$$

Since $x_0 \in [0, x]$.

Therefore,

$$x_{k-1} - x_k = f_{x_{k-1}}(\beta_k) \in \left[\frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{2\beta_k^3}{x_{k-1}^5}, \frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{2\beta_k^3}{(x_{k-1}^2 - 4\beta_k)^{5/2}} \right].$$

Finally, since $\beta_k \leq \frac{c^{2k-5}}{d}$ and $x_{k-1} \in [0.45, 1]$, the bound is

$$\begin{aligned} \frac{2\beta_k^3}{(x_{k-1}^2 - 4\beta_k)^{5/2}} &\leq \frac{2\left(\frac{c^{2k-5}}{d}\right)^3}{\left(0.45^2 - 4\frac{c^{2k-5}}{d}\right)^{5/2}} \leq \frac{2c^{6k-15}}{d^3 \left(0.45^2 - 4\frac{1}{cd}\right)^{5/2}} \\ &\leq \frac{2c^{6k-15}}{d^{1/2}(0.2d - 4 \cdot 2^{1/d})^{5/2}} \\ [d \geq 10,000 \Rightarrow 4 \cdot 2^{1/d} \leq 0.00041d] &\leq \frac{2c^{6k-15}}{d^{1/2}(0.2d - 0.00041d)^{5/2}} \leq \frac{113c^{6k-15}}{d^3}. \end{aligned}$$

□

Proposition 3.36. For any $k \geq 2$ (and $d \geq 25,000$), it holds that,

$$\frac{x_k}{x_{k-1}} \in \left(1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}, 1 - \frac{\beta_k}{x_{k-1}^2}\right).$$

Proof. We employ the bounds we found for $x_{k-1} - x_k$:

$$\begin{aligned} \frac{x_k}{x_{k-1}} &= \frac{1}{x_{k-1}}(x_k - x_{k-1} + x_{k-1}) = 1 - \frac{1}{x_{k-1}}(x_{k-1} - x_k) \\ &\in \left[1 - \frac{\beta_k}{x_{k-1}^2} - \frac{\beta_k^2}{x_{k-1}^4} - \frac{2\beta_k^3}{x_{k-1}(x_{k-1}^2 - 4\beta_k)^{5/2}}, 1 - \frac{\beta_k}{x_{k-1}^2} - \frac{\beta_k^2}{x_{k-1}^4} - \frac{2\beta_k^3}{x_{k-1}^6}\right] \\ &\subset \left[1 - \frac{\beta_k}{x_{k-1}^2} - \frac{\beta_k^2}{x_{k-1}^4} - \frac{2\beta_k^3}{x_{k-1}(x_{k-1}^2 - 4\beta_k)^{5/2}}, 1 - \frac{\beta_k}{x_{k-1}^2}\right]. \end{aligned}$$

From the bounds on β_k, x_{k-1} :

$$\begin{aligned} \frac{2\beta_k^3}{x_{k-1}(x_{k-1}^2 - 4\beta_k)^{5/2}} &= \frac{2\beta_k^2 \beta_k x_{k-1}^3}{x_{k-1}^4 (x_{k-1}^2 - 4\beta_k)^{5/2}} \leq \frac{2\beta_k^2 \frac{c^{2k-5}}{d} 1^3}{x_{k-1}^4 \left(0.45^2 - 4\frac{c^{2k-5}}{d}\right)^{5/2}} \\ &\leq \frac{2\beta_k^2 \frac{1}{d}}{x_{k-1}^4 \left(0.45^2 - 4\frac{1}{cd}\right)^{5/2}} = \frac{2\beta_k^2}{d \cdot 0.45^2 x_{k-1}^4 \left(1 - \frac{4}{0.45^2} \frac{1}{2^{-1/d}d}\right)^{5/2}} \\ [d \geq 10,000] &\leq \frac{2\beta_k^3}{d \cdot 0.45^2 x_{k-1}^4 \left(1 - \frac{4}{0.45^2} \frac{1}{2^{-1/10000} \cdot 10000}\right)^{5/2}} \leq \frac{\beta_k^3}{x_{k-1}^4} \cdot \frac{10}{d}. \end{aligned}$$

Since $d \geq 10,000$, we obtain $\frac{2\beta_k^3}{x_{k-1}(x_{k-1}^2 - 4\beta_k)^{5/2}} \leq \frac{10}{d} \frac{\beta_k^2}{x_{k-1}^4}$. Therefore,

$$\frac{x_k}{x_{k-1}} \in \left(1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}, 1 - \frac{\beta_k}{x_{k-1}^2}\right).$$

□

Proposition 3.37. For $k \geq 2$ (and $d \geq 25,000$), it holds that,

$$(x_{k-1} - x_k)^2 = \left(\frac{x_{k-1} - \sqrt{x_{k-1}^2 - 4\beta_k}}{2}\right)^2 \in \left[\frac{\beta_k^2}{x_{k-1}^2}, \frac{\beta_k^2}{x_{k-1}^2} + \frac{113c^{6k-15}}{d^3}\right].$$

Proof. We exploit the Taylor expansion of the following function (for $z^2 \gg x > 0$),

$$f(x) = \left(\frac{z - \sqrt{z^2 - 4x}}{2}\right)^2, \\ f^{(1)}(x) = \frac{z}{\sqrt{z^2 - 4x}} - 1, f^{(2)}(x) = \frac{2z}{(z^2 - 4x)^{3/2}}, f^{(3)}(x) = \frac{12z}{(z^2 - 4x)^{5/2}}.$$

Then, by Lagrange's form of the remainder, the error of the quadratic approximation (around $x = 0$) is given by

$$f(x) = \frac{f(0)}{0!}x^0 + \frac{f^{(1)}(0)}{1!}x^1 + \frac{f^{(2)}(0)}{2!}x^2 + R_2(x) \\ = 0 + 0 \cdot x^1 + \frac{2}{2z^2}x^2 + R_2(x) = \frac{x^2}{z^2} + R_2(x),$$

where

$$R_2(x) = \frac{f^{(3)}(x_0)}{3!}(x - 0)^3 = \frac{12z}{6(z^2 - 4x_0)^{5/2}}x^3 = \frac{2z}{(z^2 - 4x_0)^{5/2}}x^3 \in \left[\frac{2x^3}{z^4}, \frac{2z \cdot x^3}{(z^2 - 4x)^{5/2}}\right],$$

Since $x_0 \in [0, x]$.

Then, setting $z = x_{k-1} \in [0.45, 1]$, we can now conclude that,

$$\begin{aligned} (x_{k-1} - x_k)^2 &= \left(\frac{x_{k-1} - \sqrt{x_{k-1}^2 - 4\beta_k}}{2} \right)^2 \triangleq f(\beta_k) \\ &\in \left[\frac{\beta_k^2}{x_{k-1}^2} + \frac{2\beta_k^3}{x_{k-1}^4}, \frac{\beta_k^2}{x_{k-1}^2} + \frac{2\beta_k^3 x_{k-1}}{(x_{k-1}^2 - 4\beta_k)^{5/2}} \right]. \end{aligned}$$

We simplify the lower bound as $(x_{k-1} - x_k)^2 \geq \frac{\beta_k^2}{x_{k-1}^2} + \frac{2\beta_k^3}{x_{k-1}^4} \geq \frac{\beta_k^2}{x_{k-1}^2}$.

Finally, for the upper bound, since $\beta_k \leq \frac{c^{2k-5}}{d}$ and $x_{k-1} \in [0.45, 1]$, the upper bound satisfies

$$\begin{aligned} (x_{k-1} - x_k)^2 &\leq \frac{\beta_k^2}{x_{k-1}^2} + \frac{2\beta_k^3 x_{k-1}}{(x_{k-1}^2 - 4\beta_k)^{5/2}} \leq \frac{\beta_k^2}{x_{k-1}^2} + \frac{2c^{6k-15} x_{k-1}}{d^3 (x_{k-1}^2 - 4\frac{c^{2k-5}}{d})^{5/2}} \\ &\leq \frac{\beta_k^2}{x_{k-1}^2} + \frac{2 \cdot c^{6k-15}}{d^3 (0.45^2 - 4\frac{1}{cd})^{5/2}} \leq \frac{\beta_k^2}{x_{k-1}^2} + \frac{2c^{6k-15}}{d^{1/2} (0.2d - 4 \cdot 2^{1/d})^{5/2}} \\ \left[\Rightarrow \frac{d \geq 10,000}{4 \cdot 2^{1/d} \leq 0.00041d} \right] &\leq \frac{\beta_k^2}{x_{k-1}^2} + \frac{2c^{6k-15}}{d^{1/2} (0.2d - 0.00041d)^{5/2}} \leq \frac{\beta_k^2}{x_{k-1}^2} + \frac{113c^{6k-15}}{d^3}. \end{aligned}$$

□

Expanding the inner product $(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1}$

Proposition 3.38. Let $t \in [d]$ and $t < k \leq d$ (and $d \geq 25,000$). Then,

$$\begin{aligned} (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} &\in \left[\frac{x_{t-1}}{x_{k-1}} \beta_k + \frac{x_{t-1}}{x_{k-1}^3} \beta_k^2 + \frac{t-2}{d} c^{k+t-6} (1-c), \right. \\ &\quad \left. \frac{x_{t-1}}{x_{k-1}} \beta_k + \frac{x_{t-1}}{x_{k-1}^3} \beta_k^2 + \frac{t-2}{d} c^{k+t-6} (1-c) + \frac{113c^{6k-15}}{d^3} \right]. \end{aligned}$$

Proof. We use the expanded form of the inner product, that is,

$$\begin{aligned} 0 < (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} &= (x_{k-1} - x_k)x_{t-1} + (t-2) \frac{c^{k-3}(1-c)}{\sqrt{d}} \frac{c^{t-3}}{\sqrt{d}} \\ &= (x_{k-1} - x_k)x_{t-1} + \frac{t-2}{d} c^{k+t-6}(1-c). \end{aligned}$$

Since $x_{k-1} - x_k \in \left[\frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3}, \frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{113c^{6k-15}}{d^3} \right]$, we obtain

$$(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} \geq \frac{x_{t-1}}{x_{k-1}} \beta_k + \frac{x_{t-1}}{x_{k-1}^3} \beta_k^2 + \frac{t-2}{d} c^{k+t-6}(1-c),$$

and,

$$\begin{aligned} (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} &\leq x_{t-1} \left(\frac{\beta_k}{x_{k-1}} + \frac{\beta_k^2}{x_{k-1}^3} + \frac{113c^{6k-15}}{d^3} \right) + \frac{t-2}{d} c^{k+t-6}(1-c) \\ [x_{t-1} \leq 1] &\leq \frac{x_{t-1}}{x_{k-1}} \beta_k + \frac{x_{t-1}}{x_{k-1}^3} \beta_k^2 + \frac{t-2}{d} c^{k+t-6}(1-c) + \frac{113c^{6k-15}}{d^3}. \end{aligned}$$

□

Bounding $h(k) \triangleq \sqrt{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6}(1-c)^2 + \frac{1}{d} c^{2k-6}}$

Proposition 3.39. For any $k \geq 2$ (when $d \geq 25,000$), $h(k) \in \left[\frac{c^{k-3}}{\sqrt{d}}, \frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}} \right]$.

Proof. The lower bound is:

$$h(k) = \sqrt{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6}(1-c)^2 + \frac{1}{d} c^{2k-6}} > \sqrt{\frac{1}{d} c^{2k-6}} \geq \frac{c^{k-3}}{\sqrt{d}}.$$

To get the upper bound, we employ the inequality $(1 - c) \triangleq 1 - 2^{-1/d} \leq \frac{\ln(2)}{d}$, and get,

$$\begin{aligned}
h(k) &= \sqrt{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{c^{2k-6}}{d}} \\
&\leq \sqrt{\frac{\beta_k^2}{(0.45)^2} + (1-c)^2 + \frac{c^{2k-6}}{d}} \leq \sqrt{\frac{\beta_k^2}{(0.45)^2} + \frac{\ln^2(2)}{d^2} + \frac{c^{2k-6}}{d}} \\
[3.31] &\leq \sqrt{\frac{1}{(0.45)^2 c^2 d^2} + \frac{\ln^2(2)}{d^2} + \frac{c^{2k-6}}{d}} \\
\left[\begin{array}{l} d \geq 10,000 \\ \Rightarrow c^2 \geq 0.99986 \end{array} \right] &\leq \sqrt{\frac{c^{2k-6}}{d} + \frac{5.42}{d^2}} \\
[3.32] &< \sqrt{\frac{c^{2k-6}}{d}} + \frac{1}{2\sqrt{\frac{c^{2k-6}}{d}}} \cdot \frac{5.42}{d^2} = \frac{c^{k-3}}{\sqrt{d}} + \frac{1}{2c^{k-3}} \cdot \frac{5.42}{d^{3/2}} \\
[c^{k-m} \geq 2^{-1}] &\leq \frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}}.
\end{aligned}$$

□

Bounding $\frac{h(k+1)}{h(k)}$

Proposition 3.40. For any $k \geq 2$ (when $d \geq 500$),

$$\frac{h(k+1)}{h(k)} \in \left[c - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2}, c + \frac{2.44}{x_k^4 c^{2k-3} d^2} \right].$$

Proof. We start by expanding the expression in a way that will be useful for both the upper

and the lower bounds,

$$\begin{aligned}
\frac{h(k+1)}{h(k)} &= \sqrt{\frac{\frac{\beta_{k+1}^2}{x_k^2} + \frac{k-1}{d}c^{2k-4}(1-c)^2 + \frac{1}{d}c^{2k-4}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}} \\
&= \sqrt{\frac{\frac{\beta_{k+1}^2}{x_k^2} + \frac{k-2}{d}c^{2k-4}(1-c)^2 + \frac{1}{d}c^{2k-4}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}} + \frac{\frac{1}{d}c^{2k-4}(1-c)^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}} \\
&= \sqrt{c^2 + \frac{\frac{\beta_{k+1}^2}{x_k^2} - \frac{c^2\beta_k^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}} + \frac{\frac{1}{d}c^{2k-4}(1-c)^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}.
\end{aligned}$$

For the upper bound,

$$\begin{aligned}
&\frac{h(k+1)}{h(k)} \\
&= \sqrt{c^2 + \frac{\frac{\beta_{k+1}^2}{x_k^2} - \frac{c^2\beta_k^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}} + \frac{\frac{1}{d}c^{2k-4}(1-c)^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}} \\
&\stackrel{(1)}{\leq} \sqrt{c^2 + \beta_k^2 \frac{\frac{1}{x_k^2} - \frac{c^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}} + \frac{\frac{1}{d}(1-c)^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}} \\
&\stackrel{(2)}{\leq} \sqrt{c^2 + \frac{1}{c^2 d^2} \frac{\frac{1}{x_k^2} - \frac{c^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}} + \frac{\frac{1}{d} \left(\frac{\ln(2)}{d}\right)^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}} \\
&\leq \sqrt{c^2 + \frac{1}{c^2 d^2} \frac{\frac{1}{x_k^2} - \frac{c^2}{x_{k-1}^2}}{\frac{1}{d}c^{2k-6}} + \frac{\ln^2(2)}{c^{2k-6}d^2}} = \sqrt{c^2 + \frac{x_{k-1}^2 - c^2 x_k^2}{x_{k-1}^2 x_k^2 c^{2k-4} d} + \frac{\ln^2(2)}{c^{2k-6} d^2}} \\
&\stackrel{(3)}{\leq} \sqrt{c^2 + \frac{x_{k-1}^2 - c^2 x_k^2}{x_k^4 c^{2k-4} d} + \frac{\ln^2(2)}{c^{2k-6} d^2}} = \sqrt{c^2 + \frac{x_{k-1}^2 - c^2 x_{k-1}^2}{x_k^4 c^{2k-4} d} + \frac{c^2 x_{k-1}^2 - c^2 x_k^2}{x_k^4 c^{2k-4} d} + \frac{\ln^2(2)}{c^{2k-6} d^2}} \\
&\stackrel{(4)}{\leq} \sqrt{c^2 + (1-c^2) \frac{1}{x_k^4 c^{2k-4} d} + c^2 \frac{x_{k-1}^2 - x_k^2}{x_k^4 c^{2k-4} d} + \frac{\ln^2(2)}{c^{2k-6} d^2}},
\end{aligned}$$

where (1) is since $\beta_{k+1} < \beta_k$, $c < 1$; (2) is since $\beta_k < \frac{1}{cd}$, $1-c \leq \frac{\ln 2}{d}$; (3) is since $x_k \leq x_{k-1}$; and (4) is since $x_{k-1} \leq 1$. To upper bound $x_{k-1}^2 - x_k^2$ we use the recursive formula of x_k ,

showing that

$$\begin{aligned}
x_{k-1}^2 - x_k^2 &= x_{k-1}^2 - \frac{x_{k-1}^2 + 2x_{k-1}\sqrt{x_{k-1}^2 - 4\beta_k} + x_{k-1}^2 - 4\beta_k}{4} \\
&= \frac{x_{k-1}^2}{2} - \frac{x_{k-1}^2\sqrt{1 - 4\frac{\beta_k}{x_{k-1}^2}} - 2\beta_k}{2} \\
[1 - z \leq \sqrt{1 - z}] &\leq \frac{x_{k-1}^2}{2} - \frac{x_{k-1}^2\left(1 - 4\frac{\beta_k}{x_{k-1}^2}\right) - 2\beta_k}{2} = \frac{4\beta_k + 2\beta_k}{2} = 3\beta_k.
\end{aligned}$$

Back to our expression,

$$\begin{aligned}
\frac{h(k+1)}{h(k)} &\leq \sqrt{c^2 + (1 - c^2)\frac{1}{x_k^4 c^{2k-4} d} + \frac{3\beta_k}{x_k^4 c^{2k-6} d} + \frac{\ln^2(2)}{c^{2k-6} d^2}} \\
\left[\beta_k \leq \frac{1}{cd}\right] &\leq \sqrt{c^2 + (1 - c^2)\frac{1}{x_k^4 c^{2k-4} d} + \frac{3}{x_k^4 c^{2k-5} d^2} + \frac{\ln^2(2)}{c^{2k-6} d^2}} \\
\left[1 - c^2 \leq \frac{\ln(4)}{d}\right] &\leq \sqrt{c^2 + \frac{\ln(4)}{x_k^4 c^{2k-4} d^2} + \frac{3}{x_k^4 c^{2k-5} d^2} + \frac{\ln^2(2)}{c^{2k-6} d^2}} \\
[c < 1] &\leq \sqrt{c^2 + \frac{\ln(4)}{x_k^4 c^{2k-4} d^2} + \frac{3}{x_k^4 c^{2k-4} d^2} + \frac{\ln^2(2)}{c^{2k-4} d^2}} \leq \sqrt{c^2 + \frac{4.39 + 0.49x_k^4}{x_k^4 c^{2k-6} d^2}} \\
[x_k \leq 1] &\leq \sqrt{c^2 + \frac{4.88}{x_k^4 c^{2k-4} d^2}} = c\sqrt{1 + \frac{4.88}{x_k^4 c^{2k-2} d^2}} \leq c + \frac{2.44}{x_k^4 c^{2k-3} d^2},
\end{aligned}$$

where in the last inequality we used the fact that $\forall z > 0, \sqrt{1 + z} \leq 1 + \frac{z}{2}$ (since $(1 + \frac{z}{2})^2 = 1 + z + \frac{z^2}{4} \geq 1 + z = (\sqrt{1 + z})^2$).

For the lower bound,

$$\begin{aligned}
& \frac{h(k+1)}{h(k)} \\
&= \sqrt{c^2 + \frac{\frac{\beta_{k+1}^2}{x_k^2} - \frac{c^2 \beta_k^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{1}{d} c^{2k-6}}} + \frac{\frac{1}{d} c^{2k-4} (1-c)^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{1}{d} c^{2k-6}} \\
&\geq \sqrt{c^2 + \frac{\frac{\beta_{k+1}^2}{x_k^2} - \frac{c^2 \beta_k^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{1}{d} c^{2k-6}}} \\
&\geq c \sqrt{1 + \frac{\frac{\beta_{k+1}^2}{x_k^2} - \frac{\beta_k^2}{x_{k-1}^2}}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{1}{d} c^{2k-6}}} \\
&\stackrel{(1)}{\geq} c \sqrt{1 + \frac{1}{x_{k-1}^2} \frac{\beta_{k+1}^2 - \beta_k^2}{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d} c^{2k-6} (1-c)^2 + \frac{1}{d} c^{2k-6}}},
\end{aligned}$$

where (1) is since $x_k \leq x_{k-1}$. Since $(\beta_k)_k$ is positive and decreasing, $\beta_{k+1}^2 - \beta_k^2 < 0$, and so we can simplify the expression using the fact that $\sqrt{1-z} \geq 1-z, \forall z \in (0, 1)$:

$$\frac{h(k+1)}{h(k)} \geq c \sqrt{1 - \frac{|\beta_k^2 - \beta_{k+1}^2|}{\frac{1}{d} c^{2k-6} x_{k-1}^2}} \geq c - \frac{|\beta_k^2 - \beta_{k+1}^2|}{\frac{1}{d} c^{2k-5} x_{k-1}^2}.$$

Focusing on $\frac{|\beta_{k+1}^2 - \beta_k^2|}{\frac{1}{d} c^{2k-5}}$, and since $1 - (1-c)(k-1) = ((k-1)c - (k-2))$,

$$\begin{aligned}
\frac{|\beta_{k+1}^2 - \beta_k^2|}{\frac{1}{d} c^{2k-5}} &= \frac{\beta_k^2 - \beta_{k+1}^2}{\frac{1}{d} c^{2k-5}} = \frac{\left(\frac{((k-1)c - (k-2))c^{2k-5}}{d}\right)^2 - \left(\frac{(kc - (k-1))c^{2k-3}}{d}\right)^2}{\frac{1}{d} c^{2k-5}} \\
&= \frac{c^{2k-5}}{d} ((1 - (1-c)(k-1))^2 - (1 - (1-c)k)^2 c^4) \\
&= \frac{c^{2k-5}}{d} \left((1 - c^4) - 2k \underbrace{(1-c)(1-c^4)}_{\geq 0} + k^2 (1-c)^2 (1-c^4) + \right. \\
&\quad \left. + 2(1-c) + \underbrace{(1-c)^2 (-2k+1)}_{\geq 0} \right) \\
&\leq \frac{c^{2k-5}}{d} ((1 - c^4) + k^2 (1-c)^2 (1-c^4) + 2(1-c) + (1-c)^2).
\end{aligned}$$

The previously derived bounds $1 - c \in \left[\frac{\ln(2)}{d} - \frac{\ln^2(2)}{2d^2}, \frac{\ln(2)}{d} \right]$ give

$$\begin{aligned} \frac{|\beta_{k+1}^2 - \beta_k^2|}{\frac{1}{d}c^{2k-5}} &\leq \frac{c^{2k-5}}{d} \left((1 - c^4) + k^2 \frac{\ln^2(2)}{d^2} (1 - c^4) + 2 \frac{\ln(2)}{d} + \frac{\ln^2(2)}{d^2} \right) \\ [d \geq 500 \geq 1000 \ln^2(2)] &\leq \frac{c^{2k-5}}{d} \left((1 - c^4) + k^2 \frac{\ln^2(2)}{d^2} (1 - c^4) + 2 \frac{\ln(2)}{d} + \frac{0.001}{d} \right) \\ [k \leq d] &\leq \frac{c^{2k-5}}{d} \left((1 - c^4) + \ln^2(2)(1 - c^4) + \frac{\ln(4) + 0.001}{d} \right) \\ &\leq \frac{c^{2k-5}}{d} \left((1 + \ln^2(2))(1 - c^4) + \frac{\ln(4) + 0.001}{d} \right). \end{aligned}$$

The previously derived bound $1 - c^n \leq \frac{n \ln(2)}{d}$ gives

$$\frac{|\beta_{k+1}^2 - \beta_k^2|}{\frac{1}{d}c^{2k-5}} \leq \frac{c^{2k-5}}{d} \left((1 + \ln^2(2)) \frac{4 \ln(2)}{d} + \frac{\ln(4) + 0.001}{d} \right) \leq 5.5 \frac{c^{2k-5}}{d^2}.$$

Finally,

$$\frac{h(k+1)}{h(k)} \geq c - \frac{|\beta_k^2 - \beta_{k+1}^2|}{\frac{1}{d}c^{2k-5}x_{k-1}^2} \geq c - \frac{5.5c^{2k-5}}{x_{k-1}^2d^2}.$$

□

Expanding the norm

Proposition 3.41. For any $k \geq 2$, $\|\mathbf{w}_{k-1} - \mathbf{w}_k\| \in \left[h(k), h(k) + \frac{56.5c^{6k-15}}{c^{k-3}d^{5/2}} \right]$, where $h(k) \triangleq \sqrt{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}$.

Proof. By construction,

$$\begin{aligned} \|\mathbf{w}_{k-1} - \mathbf{w}_k\| &= \sqrt{(x_{k-1} - x_k)^2 + (k-2) \left(\frac{c^{k-3}(1-c)}{\sqrt{d}} \right)^2 + \left(\frac{c^{k-3}}{\sqrt{d}} \right)^2} \\ &= \sqrt{(x_{k-1} - x_k)^2 + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}}. \end{aligned}$$

Before, we proved that $(x_{k-1} - x_k)^2 \in \left[\frac{\beta_k^2}{x_{k-1}^2}, \frac{\beta_k^2}{x_{k-1}^2} + \frac{113c^{6k-15}}{d^3} \right]$. Now, we show the resulting bounds for $\|\mathbf{w}_{k-1} - \mathbf{w}_k\|$ which employ that bound.

The lower bound is immediate, since

$$\begin{aligned}\|\mathbf{w}_{k-1} - \mathbf{w}_k\| &= \sqrt{(x_{k-1} - x_k)^2 + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}} \\ &\geq \sqrt{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6}} \triangleq h(k).\end{aligned}$$

The upper bound requires an additional algebraic inequality of $\forall a, b > 0 : \sqrt{a+b} < \sqrt{a} + \frac{b}{2\sqrt{a}}$ and the inequality of $h(k) \geq \frac{1}{\sqrt{d}}c^{k-3}$, i.e.,

$$\begin{aligned}\|\mathbf{w}_{k-1} - \mathbf{w}_k\| &\leq \sqrt{\frac{\beta_k^2}{x_{k-1}^2} + \frac{k-2}{d}c^{2k-6}(1-c)^2 + \frac{1}{d}c^{2k-6} + \frac{113c^{6k-15}}{d^3}} \\ &= \sqrt{h^2(k) + \frac{113c^{6k-15}}{d^3}} \leq h(k) + \frac{113c^{6k-15}}{2h(k)d^3} \leq h(k) + \frac{56.5c^{6k-15}}{c^{k-3}d^{5/2}}.\end{aligned}$$

□

Combining the expansions

Proposition 3.42. When $d \geq 25,000$,

$$\begin{aligned}&\|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} \\ &\geq A_1(k) + A_2(k) + A_3(k) \\ &\quad - \frac{1}{d^{7/2}} \left(\frac{96c^{6k-15}}{x_k} + 113c^{7k-18} \right) - \frac{1}{d^{9/2}} \left(\frac{56.5c^{9k-18}}{x_k^3} + 614c^{6k-30} \right),\end{aligned}$$

where $A_1(k) \triangleq x_{t-1}(\frac{h(k+1)}{x_{k-1}}\beta_k - \frac{h(k)}{x_k}\beta_{k+1})$, $A_2(k) \triangleq x_{t-1}(\frac{h(k+1)}{x_{k-1}^3}\beta_k^2 - \frac{h(k)}{x_k^3}\beta_{k+1}^2)$, $A_3(k) \triangleq \frac{t-2}{d}(1-c)c^{k+t-6}(h(k+1) - ch(k))$.

Proof. To bound

$$\|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1},$$

we lower bound the right expression. The bounds for $(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1}$ and $\|\mathbf{w}_{k-1} - \mathbf{w}_k\|$ derived above give

$$\begin{aligned}
& -\|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} \\
& \geq -\|\mathbf{w}_{k-1} - \mathbf{w}_k\| \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{(t-2)c^{k+t-5}(1-c)}{d} + \frac{113c^{6k-15}}{d^3} \right) \\
& \geq -\|\mathbf{w}_{k-1} - \mathbf{w}_k\| \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{(t-2)c^{k+t-5}(1-c)}{d} \right) \\
& \quad - \|\mathbf{w}_{k-1} - \mathbf{w}_k\| \frac{113c^{6k-15}}{d^3} \\
& \geq - \underbrace{\left(h(k) + \frac{56.5c^{6k-15}}{c^{k-3}d^{5/2}} \right) \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{t-2}{d} c^{k+t-5}(1-c) \right)}_{\triangleq a(k)} \\
& \quad - \underbrace{\|\mathbf{w}_{k-1} - \mathbf{w}_k\| \frac{113c^{6k-15}}{d^3}}_{\triangleq b(k)}.
\end{aligned}$$

The right function is bounded as

$$\begin{aligned}
b(k) &= -\|\mathbf{w}_{k-1} - \mathbf{w}_k\| \frac{113c^{6k-15}}{d^3} \\
&\geq - \left(h(k) + \frac{56.5c^{6k-15}}{c^{k-3}d^{5/2}} \right) \frac{113c^{6k-15}}{d^3} \\
&= -\frac{113c^{6k-15}}{d^3} h(k) - \frac{6384.5c^{12k-30}}{d^{11/2}c^{k-3}} \\
&\geq -\frac{113c^{6k-15}}{d^3} \left(\frac{\sqrt{c^{2k-6}}}{\sqrt{d}} + \frac{5.42}{d^{3/2}} \right) - \frac{6384.5c^{12k-30}}{d^{11/2}c^{k-3}} \\
&= -\frac{113c^{6k-15}c^{k-3}}{d^{7/2}} - \frac{612.46c^{6k-15}}{d^{9/2}} - \frac{6384.5c^{12k-30}}{d^{11/2}c^{k-3}} \\
&[c^{k-3} \geq c^d = 0.5] \geq -\frac{113c^{7k-18}}{d^{7/2}} - \frac{612.46c^{6k-15}}{d^{9/2}} - \frac{12769c^{12k-30}}{d^{11/2}}.
\end{aligned}$$

The left function is further decomposed as

$$a(k) = \underbrace{-h(k) \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{t-2}{d} c^{k+t-5} (1-c) \right)}_{\triangleq a_1(k)} - \underbrace{\frac{56.5c^{5k-12}}{d^{5/2}} \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{t-2}{d} c^{k+t-5} (1-c) \right)}_{\triangleq a_2(k)}.$$

Then,

$$\begin{aligned} a_2(k) &= -\frac{56.5c^{5k-12}}{d^{5/2}} \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{t-2}{d} c^{k+t-5} (1-c) \right) \\ \left[x_{t-1} \leq 1, \frac{t-2}{d} < 1 \right] &\geq -\frac{56.5c^{5k-12}}{d^{5/2}} \left(\frac{1}{x_k} \beta_{k+1} + \frac{1}{x_k^3} \beta_{k+1}^2 + c^{k+t-5} (1-c) \right) \\ \left[\beta_{k+1} \leq \frac{c^{2k-3}}{d} \right] &\geq -\frac{56.5c^{5k-12}}{d^{5/2}} \left(\frac{c^{2k-3}}{x_k d} + \frac{c^{4k-6}}{x_k^3 d^2} + c^{k+t-5} (1-c) \right) \\ &\geq -\frac{56.5c^{5k-12}}{d^{5/2}} \left(\frac{c^{2k-3}}{x_k d} + \frac{c^{4k-6}}{x_k^3 d^2} + \frac{\ln(2)}{d} c^{k+t-5} \right) \\ [x_k \leq 1] &\geq -\frac{56.5c^{5k-12}}{d^{5/2}} \left(\frac{c^{2k-3} + \ln(2)c^{k+t-5}}{x_k d} + \frac{c^{4k-6}}{x_k^3 d^2} \right) \\ &= -\frac{56.5c^{6k-15}}{d^{5/2} \cdot x_k d} \left(c^k + \ln(2)c^{t-2} + \frac{c^{3k-3}}{x_k^2 d} \right) \\ [c < 1] &\geq -\frac{56.5c^{6k-15}}{d^{7/2} \cdot x_k} \left(1 + \ln(2) + \frac{c^{3k-3}}{x_k^2 d} \right) \\ &\geq -\frac{96c^{6k-15}}{d^{7/2} \cdot x_k} - \frac{56.5c^{9k-18}}{d^{9/2} \cdot x_k^3}. \end{aligned}$$

Overall,

$$\begin{aligned}
& \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1}, \\
& \geq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} + a_1(k) + a_2(k) + b(k) \\
& \geq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} + a_1(k) \\
& \quad - \frac{96c^{6k-15}}{d^{7/2} \cdot x_k} - \frac{56.5c^{9k-18}}{d^{9/2} \cdot x_k^3} - \frac{113c^{7k-18}}{d^{7/2}} - \frac{612.46c^{6k-15}}{d^{9/2}} - \frac{12769c^{12k-30}}{d^{11/2}} \\
& \geq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} + a_1(k) \\
& \quad - \frac{96c^{6k-15}}{d^{7/2} \cdot x_k} - \frac{56.5c^{9k-18}}{d^{9/2} \cdot x_k^3} - \frac{113c^{7k-18}}{d^{7/2}} - \frac{612.46c^{6k-30}}{d^{9/2}} - \frac{12769c^{6k-30}}{d^{11/2}} \\
& \stackrel{(1)}{\geq} \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} \\
& \quad + a_1(k) - \frac{96c^{6k-15}}{d^{7/2} \cdot x_k} - \frac{56.5c^{9k-18}}{d^{9/2} \cdot x_k^3} - \frac{113c^{7k-18}}{d^{7/2}} - \frac{614c^{6k-30}}{d^{9/2}} \\
& \geq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} \\
& \quad + a_1(k) - \frac{1}{d^{7/2}} \left(\frac{96c^{6k-15}}{x_k} + 113c^{7k-18} \right) - \frac{1}{d^{9/2}} \left(\frac{56.5c^{9k-18}}{x_k^3} + 614c^{6k-30} \right),
\end{aligned}$$

where (1) is since $d \geq 10,000$. Focusing on the left terms gives the overall expression, which

we need to show is positive. The previously derived inequalities show

$$\begin{aligned}
& \| \mathbf{w}_k - \mathbf{w}_{k+1} \| (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} + a_1(k) \\
&= \| \mathbf{w}_k - \mathbf{w}_{k+1} \| (\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} \\
&\quad - h(k) \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{t-2}{d} c^{k+t-5} (1-c) \right) \\
&\geq h(k+1) \left(\frac{x_{t-1}}{x_{k-1}} \beta_k + \frac{x_{t-1}}{x_{k-1}^3} \beta_k^2 + \frac{t-2}{d} c^{k+t-6} (1-c) \right) \\
&\quad - h(k) \left(\frac{x_{t-1}}{x_k} \beta_{k+1} + \frac{x_{t-1}}{x_k^3} \beta_{k+1}^2 + \frac{t-2}{d} c^{k+t-5} (1-c) \right) \\
&= \underbrace{x_{t-1} \left(\frac{h(k+1)}{x_{k-1}} \beta_k - \frac{h(k)}{x_k} \beta_{k+1} \right)}_{\triangleq A_1(k)} + \underbrace{x_{t-1} \left(\frac{h(k+1)}{x_{k-1}^3} \beta_k^2 - \frac{h(k)}{x_k^3} \beta_{k+1}^2 \right)}_{\triangleq A_2(k)} \\
&\quad + \underbrace{\frac{t-2}{d} (1-c) c^{k+t-6} (h(k+1) - ch(k))}_{\triangleq A_3(k)},
\end{aligned}$$

which we will bound separately below. □

The second term, $A_2(k)$, is insignificant $\mathcal{O}(\frac{1}{d^{7/2}})$

Proposition 3.43. When $d \geq 25,000$,

$$A_2(k) = x_{t-1} \left(\frac{h(k+1)}{x_{k-1}^3} \beta_k^2 - \frac{h(k)}{x_k^3} \beta_{k+1}^2 \right) \geq -\frac{14.88 x_{t-1} c^{3k-12}}{x_k^3 d^{7/2}}.$$

Proof. We start from,

$$A_2(k) = x_{t-1} \left(\frac{h(k+1)}{x_{k-1}^3} \beta_k^2 - \frac{h(k)}{x_k^3} \beta_{k+1}^2 \right) = \frac{x_{t-1}}{x_k^3} h(k) \beta_{k+1}^2 \underbrace{\left(\frac{h(k+1)}{h(k)} \frac{x_k^3}{x_{k-1}^3} \frac{\beta_k^2}{\beta_{k+1}^2} - 1 \right)}_{\triangleq a(k)}.$$

Dissecting the terms in $a(k)$,

$$\frac{\beta_k}{\beta_{k+1}} = \frac{\frac{((k-1)c-(k-2))c^{2k-5}}{d}}{\frac{(kc-(k-1))c^{2k-3}}{d}} = \frac{1}{c^2} + \underbrace{\frac{1-c}{(kc-(k-1))}}_{>0, \text{ from 3.28}} c^2 \geq \frac{1}{c^2},$$

$$\frac{\beta_k^2}{\beta_{k+1}^2} \geq \frac{1}{c^4}.$$

We already showed that $\frac{x_k}{x_{k-1}} \in \left(1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}, 1 - \frac{\beta_k}{x_{k-1}^2}\right)$, and we simplify it further:

$$\frac{x_k}{x_{k-1}} \geq 1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4} \stackrel{\beta_k \leq \frac{1}{d}}{\geq} 1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k}{x_{k-1}^4 d}$$

$$[x_{k-1} \geq 0.45] \geq 1 - \beta_k \left(\frac{1}{(0.45)^2} + \frac{\left(1 + \frac{10}{d}\right)}{(0.45)^4 d} \right) \stackrel{d \geq 10,000}{\geq} 1 - 4.95\beta_k.$$

Using the algebraic inequality $\forall z \in (0, 1), (1-z)^3 = 1-3z+3z^2-z^3 > 1-3z$, it follows that

$$\frac{x_k^3}{x_{k-1}^3} > 1 - 14.85\beta_k.$$

Moreover, recall that we already showed that $\frac{h(k+1)}{h(k)} \geq c - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2}$. Now, focusing on $a(k)$,

$$\begin{aligned}
a(k) &= \frac{h(k+1)}{h(k)} \cdot \frac{x_k^3}{x_{k-1}^3} \frac{\beta_k^2}{\beta_{k+1}^2} - 1 \\
&\geq \left(c - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2} \right) (1 - 14.85\beta_k) \frac{1}{c^4} - 1 \\
&= \left(\frac{1}{c^3} - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2} \right) (1 - 14.85\beta_k) - 1 \\
&\geq \left(1 - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2} \right) (1 - 14.85\beta_k) - 1 \\
&= -14.85\beta_k - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2} + \frac{81.675c^{2k-9}}{x_{k-1}^2 d^2} \beta_k \\
&\geq -14.85\beta_k - \frac{5.5c^{2k-9}}{x_{k-1}^2 d^2} \\
\left[\beta_k \leq \frac{c^{2k-5}}{d} \right] &\geq -\frac{14.85c^{2k-5}}{d} - \frac{5.5c^{2k-9}}{x_{k-1}^2 d^2} \geq -\frac{14.85c^{2k-5}}{d} - \frac{5.5c^{2k-9}}{(0.45)^2 d^2} \\
&\geq -\frac{14.85c^{2k-5}}{d} - \frac{27.17c^{2k-9}}{d^2} \geq -\frac{14.85c^{2k-9}}{d} - \frac{27.17c^{2k-9}}{d^2} \\
[d \geq 10,000] &\geq -\frac{14.86c^{2k-9}}{d}.
\end{aligned}$$

And finally,

$$\begin{aligned}
\frac{1}{x_{t-1}} A_2(k) &= \frac{1}{x_k^3} h(k) \beta_{k+1}^2 \cdot a(k) \geq -\frac{1}{x_k^3} h(k) \beta_{k+1}^2 \cdot \frac{14.86c^{2k-9}}{d} \\
\left[\begin{array}{l} \beta_{k+1} \leq \frac{1}{d}, \\ h(k) \leq \frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}} \end{array} \right] &\geq -\frac{1}{x_k^3} \frac{\frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}}}{d^2} \frac{14.86c^{2k-9}}{d} = -\frac{1}{x_k^3} \frac{c^{3k-12}}{d^{5/2}} \frac{14.86}{d} - \frac{1}{x_k^3} \frac{5.42}{d^{7/2}} \frac{14.86c^{2k-9}}{d} \\
[c < 1] &\geq -\frac{14.86c^{3k-12}}{x_k^3 d^{7/2}} - \frac{80.55c^{2k-12}}{x_k^3 d^{9/2}} \\
\left[\begin{array}{l} d \geq 10,000, \\ c^{k-3} \geq c^d = 0.5 \end{array} \right] &\geq -\frac{14.86c^{3k-12}}{x_k^3 d^{7/2}} - \frac{0.0081 \cdot c^{3k-12}}{x_k^3 d^{7/2} c} \geq -\frac{(14.86 + \frac{0.0081}{0.5}) c^{3k-12}}{x_k^3 d^{7/2}} \\
A_2(k) &\geq -\frac{14.88x_{t-1}c^{3k-12}}{x_k^3 d^{7/2}},
\end{aligned}$$

thus concluding this part. □

The third term, $A_3(k)$, is insignificant $\mathcal{O}\left(\frac{1}{d^{7/2}}\right)$

Proposition 3.44. When $d \geq 25,000$,

$$|A_3(k)| = \left| \frac{t-2}{d} (1-c)c^{k+t-6} (h(k+1) - ch(k)) \right| \leq \frac{6.77c^{k-6}}{x_k^4} \left(\frac{c^{k-3}}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right).$$

Proof.

$$\begin{aligned} |A_3(k)| &= \left| \frac{t-2}{d} (1-c)c^{k+t-6} (h(k+1) - ch(k)) \right| \\ &= \frac{t-2}{d} (1-c)c^{k+t-6} |h(k+1) - ch(k)| \leq (1-c)c^{k+t-6} |h(k+1) - ch(k)| \\ &\leq \ln(2)c^{k+t-6} \frac{h(k)}{d} \left| \frac{h(k+1)}{h(k)} - c \right| \leq \ln(2)c^{k+t-6} \left(\frac{c^{k-3}}{d^{3/2}} + \frac{5.42}{d^{5/2}} \right) \left| \frac{h(k+1)}{h(k)} - c \right|, \end{aligned}$$

where we used the facts that $1-c \leq \frac{\ln 2}{d}$ and $h(k) \leq \frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}}$.

Using $\frac{h(k+1)}{h(k)} \in \left[c - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2}, c + \frac{2.44}{x_k^4 c^{2k-3} d^2} \right]$, we finally get,

$$\begin{aligned} |A_3(k)| &\leq \ln(2)c^{k+t-6} \left(\frac{c^{k-3}}{d^{3/2}} + \frac{5.42}{d^{5/2}} \right) \left| \frac{h(k+1)}{h(k)} - c \right| \\ &\leq \ln(2)c^{k+t-6} \left(\frac{c^{k-3}}{d^{3/2}} + \frac{5.42}{d^{5/2}} \right) \max \left(\frac{5.5c^{2k-5}}{x_{k-1}^2 d^2}, \frac{2.44}{x_k^4 c^{2k-3} d^2} \right) \\ [x_k < x_{k-1}] &\leq \ln(2)c^{k+t-6} \left(\frac{c^{k-3}}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right) \max \left(\frac{5.5c^{2k-5}}{x_k^4}, \frac{2.44}{x_k^4 c^{2k-3}} \right) \\ [c < 1] &\leq \frac{\ln(2)c^{k+t-6}}{x_k^4} \left(\frac{c^{k-3}}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right) \max \left(5.5c^{2k-5}, \frac{2.44}{c^{2k}} \right) \\ [c^{nk-m} \geq 2^{-n}, k \geq 2, c < 1] &\leq \frac{\ln(2)c^{k+t-6}}{x_k^4} \left(\frac{1}{cd^{7/2}} + \frac{5.42}{d^{9/2}} \right) \max \left(\frac{5.5}{c}, 9.76 \right) \\ [c \geq 2^{-1/10000} \geq 0.9999] &\leq \frac{\ln(2)c^{k+t-6}}{x_k^4} \left(\frac{1}{0.9999d^{7/2}} + \frac{5.42}{d^{9/2}} \right) \max \left(\frac{5.5}{0.9999}, 9.76 \right) \\ &\leq \frac{6.77c^{k+t-6}}{x_k^4} \left(\frac{1}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right). \end{aligned}$$

□

Back to the first term, $A_1(k)$

Proposition 3.45. When $d \geq 25,000$,

$$A_1(k) = x_{t-1} \left(\frac{h(k+1)}{x_{k-1}} \beta_k - \frac{h(k)}{x_k} \beta_{k+1} \right) \geq x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{173.07c^{3k-9}}{x_k^2 d^{7/2}} \right).$$

Proof.

$$A_1(k) = x_{t-1} \left(\frac{h(k+1)}{x_{k-1}} \beta_k - \frac{h(k)}{x_k} \beta_{k+1} \right) = \underbrace{\frac{x_{t-1}}{x_k} h(k) \beta_{k+1}}_{=\Theta(d^{-3/2})} \underbrace{\left(\frac{x_k}{x_{k-1}} \frac{h(k+1)}{h(k)} \frac{\beta_k}{\beta_{k+1}} - 1 \right)}_{\triangleq a(k)}.$$

We use the previously derived lower bounds $\frac{x_k}{x_{k-1}} \geq 1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}$ and $\frac{h(k+1)}{h(k)} \geq c - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2}$. To lower bound $\frac{\beta_k}{\beta_{k+1}} = \frac{1}{c^2} + \frac{1-c}{(1-k(1-c))c^2}$, we need a slightly stronger bound than before. Specifically, for any $z \in (0, 1)$, $\frac{z}{1-z} \geq z$. Then, since $1-c \in \left[\frac{\ln(2)}{d} - \frac{\ln^2(2)}{2d^2}, \frac{\ln(2)}{d}\right] \Rightarrow k(1-c) \in \left[\frac{k}{d} \ln(2) - \frac{k}{2d^2} \ln^2(2), \frac{k}{d} \ln(2)\right] \subseteq (0, 1)$, and

$$\frac{1-c}{(1-k(1-c))c^2} = \frac{1}{c^2 k} \frac{k(1-c)}{(1-k(1-c))} \geq \frac{1}{c^2 k} k(1-c) = \frac{1-c}{c^2}.$$

Therefore,

$$\frac{\beta_k}{\beta_{k+1}} \geq \frac{1}{c^2} + \frac{1-c}{c^2} = \frac{2-c}{c^2}.$$

We are now ready to lower bound $a(k)$ as

$$\begin{aligned} a(k) &= \frac{x_k}{x_{k-1}} \frac{h(k+1)}{h(k)} \frac{\beta_k}{\beta_{k+1}} - 1 \\ &\geq \left(1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}\right) \left(c - \frac{5.5c^{2k-5}}{x_{k-1}^2 d^2}\right) \left(\frac{2-c}{c^2}\right) - 1 \\ &= \left(1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}\right) \left(1 - \frac{5.5c^{2k-6}}{x_{k-1}^2 d^2}\right) \left(\frac{2-c}{c}\right) - 1. \end{aligned}$$

Using claim 3.33: $\frac{2-c}{c} = \frac{2}{c} - 1 = 2 \cdot 2^{1/d} - 1 \geq 2 \left(1 + \frac{\ln(2)}{d}\right) - 1 = 1 + \frac{\ln(4)}{d}$, this gives:

$$\begin{aligned}
a(k) &\geq \left(1 - \frac{\beta_k}{x_{k-1}^2} - \left(1 + \frac{10}{d}\right) \frac{\beta_k^2}{x_{k-1}^4}\right) \left(1 - \frac{5.5c^{2k-6}}{x_{k-1}^2 d^2}\right) \left(1 + \frac{\ln(4)}{d}\right) - 1 \\
&= \underbrace{\frac{\ln(4)}{d} - \frac{\beta_k}{x_{k-1}^2}}_{\mathcal{O}(d^{-1})} - \underbrace{\left(\frac{5.5c^{2k-6}}{x_{k-1}^2 d^2} + \frac{\left(1 + \frac{10}{d}\right) \beta_k^2}{x_{k-1}^4} + \frac{\ln(4) \beta_k}{x_{k-1}^2 d}\right)}_{\Theta(d^{-2})} \\
&\quad + \underbrace{\frac{5.5c^{2k-6} \beta_k}{x_{k-1}^4 d^2} - \frac{5.5 \ln(4) c^{2k-6}}{x_{k-1}^2 d^3} - \frac{\left(1 + \frac{10}{d}\right) \ln(4) \beta_k^2}{x_{k-1}^4 d}}_{\mathcal{O}(d^{-3})} \\
&\quad + \underbrace{\frac{5.5 \left(1 + \frac{10}{d}\right) c^{2k-6} \beta_k^2}{x_{k-1}^6 d^2} + \frac{5.5 \ln(4) c^{2k-6}}{x_{k-1}^2 d^3} \left(\frac{\beta_k}{x_{k-1}^2} + \frac{\left(1 + \frac{10}{d}\right) \beta_k^2}{x_{k-1}^4}\right)}_{\mathcal{O}(d^{-4})}.
\end{aligned}$$

Lower bounding the negligible positive terms by 0 gives

$$\begin{aligned}
a(k) &\geq \underbrace{\frac{\ln(4)}{d} - \frac{\beta_k}{x_{k-1}^2}}_{\mathcal{O}(d^{-1})} - \underbrace{\left(\frac{5.5c^{2k-6}}{x_{k-1}^2 d^2} + \frac{\left(1 + \frac{10}{d}\right) \beta_k^2}{x_{k-1}^4} + \frac{\ln(4) \beta_k}{x_{k-1}^2 d}\right)}_{\Theta(d^{-2})} \\
&\quad - \underbrace{\left(\frac{5.5 \ln(4) c^{2k-6}}{x_{k-1}^2 d^3} + \frac{\left(1 + \frac{10}{d}\right) \ln(4) \beta_k^2}{x_{k-1}^4 d}\right)}_{\Theta(d^{-3})}.
\end{aligned}$$

We now simplify the least significant terms above further. We start from an upper bound to the $\Theta(d^{-2})$ term (since its sign is negative in the expression above),

$$\begin{aligned}
\frac{5.5c^{2k-6}}{x_{k-1}^2 d^2} + \frac{\left(1 + \frac{10}{d}\right) \beta_k^2}{x_{k-1}^4} + \frac{\ln(4) \beta_k}{x_{k-1}^2 d} &\leq \frac{5.5c^{2k-6}}{x_{k-1}^2 d^2} + \frac{1.001 \beta_k^2}{x_{k-1}^2 x_{k-1}^2} + \frac{\ln(4) \beta_k}{x_{k-1}^2 d} \\
\left[\beta_k \leq \frac{c^{2k-5}}{d} \leq \frac{1}{cd}, x_{k-1} \geq 0.45\right] &\leq \frac{5.5c^{2k-6}}{x_{k-1}^2 d^2} + \frac{1.001 c^{4k-10}}{x_{k-1}^2 d^2 0.45^2} + \frac{\ln(4) c^{2k-5}}{x_{k-1}^2 d^2} \\
&\leq \frac{c^{2k-6}}{x_{k-1}^2 d^2} (5.5 + 4.95 c^{2k-4} + \ln 4 \cdot c) \\
[k \geq 2, c \leq 1] &\leq \frac{c^{2k-6}}{x_{k-1}^2 d^2} (5.5 + 4.95 + \ln 4) \leq \frac{11.84 c^{2k-6}}{x_{k-1}^2 d^2}.
\end{aligned}$$

Similarly, for the $\Theta(d^{-3})$ term, we again employ the upper bound $\beta_k \leq \frac{c^{2k-5}}{d} \leq \frac{1}{cd}$, and

obtain,

$$\begin{aligned} \frac{5.5 \ln(4) c^{2k-6}}{x_{k-1}^2 d^3} + \frac{(1 + \frac{10}{d}) \ln(4) \beta_k^2}{x_{k-1}^4 d} &\leq \frac{5.5 \ln(4) c^{2k-6}}{x_{k-1}^2 d^3} + \frac{1.001 \ln(4) c^{2k-5}}{x_{k-1}^4 d^3 c^3} \\ &\leq \frac{c^{2k-6}}{x_{k-1}^4 d^3} (5.5 \ln(4) + 1.001 \ln(4) c^{-2}) \leq \frac{c^{2k-8}}{x_{k-1}^4 d^3} (5.5 \ln(4) + 1.001 \ln(4)) \leq \frac{9.02 c^{2k-8}}{x_{k-1}^4 d^3}. \end{aligned}$$

Thus, the following lower bound holds:

$$\begin{aligned} a(k) &\geq \frac{\ln(4)}{d} - \frac{\beta_k}{x_{k-1}^2} - \left(\frac{11.84 c^{2k-6}}{x_{k-1}^2 d^2} + \frac{9.02 c^{2k-8}}{x_{k-1}^4 d^3} \right) \\ [x_k \leq x_{k-1}] &\geq \frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} - \frac{c^{2k-6}}{x_k^2 d^2} \left(11.84 + \frac{9.02 c^{-2}}{x_k^2 d} \right). \end{aligned}$$

Back to the overall term we are trying to lower bound,

$$\begin{aligned} \frac{1}{x_{t-1}} A_1(k) &= \frac{h(k+1)}{x_{k-1}} \beta_k - \frac{h(k)}{x_k} \beta_{k+1} = \underbrace{\frac{1}{x_k} h(k) \beta_{k+1}}_{=\Theta(d^{-3/2})} \underbrace{\left(\frac{x_k}{x_{k-1}} \frac{h(k+1)}{h(k)} \frac{\beta_k}{\beta_{k+1}} - 1 \right)}_{\triangleq a(k)} \\ &\geq \frac{1}{x_k} h(k) \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} - \frac{c^{2k-6}}{x_k^2 d^2} \left(11.84 + \frac{9.02 c^{-2}}{x_k^2 d} \right) \right) \\ &= \frac{1}{x_k} h(k) \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{1}{x_k} h(k) \beta_{k+1} \frac{c^{2k-6}}{x_k^2 d^2} \left(11.84 + \frac{9.02 c^{-2}}{x_k^2 d} \right) \\ &\stackrel{(1)}{\geq} \frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}} c^{2k-6} \left(11.84 + \frac{9.02 c^{-2}}{x_k^2 d} \right) \\ &\stackrel{(2)}{\geq} \frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{c^{3k-9}}{x_k^3 d^{7/2}} \left(11.84 + \frac{9.02 \cdot 2^{2/10000}}{0.45^2 d} \right) \\ &\quad - \frac{5.42 c^{2k-6}}{x_k^3 d^{9/2}} \left(11.84 + \frac{9.02 \cdot 2^{2/10000}}{0.45^2 d} \right), \end{aligned}$$

where (1) is since $h(k) \in \left[\frac{c^{k-3}}{\sqrt{d}}, \frac{c^{k-3}}{\sqrt{d}} + \frac{5.42}{d^{3/2}} \right]$, $\beta_{k+1} \leq \frac{1}{d}$; (2) is since $d \geq 10,000$ and $x_k \geq 0.45$.

Furthermore,

$$\begin{aligned}
& \frac{1}{x_{t-1}} A_1(k) \\
& \geq \frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{44.55c^{3k-9}}{x_k^3 d^{9/2}} - \frac{64.18c^{2k-6}}{x_k^3 d^{9/2}} - \frac{241.46c^{2k-6}}{x_k^3 d^{11/2}} \\
& \geq \frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{44.55c^{2k-9}}{x_k^3 d^{9/2}} - \frac{64.18c^{2k-9}}{x_k^3 d^{9/2}} - \frac{241.46c^{2k-9}}{x_k^3 d^{11/2}} \\
& \stackrel{(3)}{\geq} \frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{109c^{2k-9}}{x_k^3 d^{9/2}},
\end{aligned}$$

where (3) is since $d \geq 10,000$. Therefore,

$$A_1(k) \geq \frac{x_{t-1}}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{11.84x_{t-1}c^{3k-9}}{x_k^3 d^{7/2}} - \frac{109x_{t-1}c^{2k-9}}{x_k^3 d^{9/2}}.$$

It remains to get a lower bound for $\beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right)$.

First, we show

$$\begin{aligned}
b(k) & \triangleq \frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} = \frac{\ln(4)}{d} - \frac{(1 - (1-c)(k-1))c^{2k-5}}{x_k^2 d} \\
& = \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{1}{x_k^2 c^3} \cdot (1-c) \left(\frac{k-1}{d} \right) 4^{-\frac{k-1}{d}} \\
& \geq \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{1}{x_k^2 c^3} \cdot \frac{1-c}{4} \left(\frac{k-1}{d} \right),
\end{aligned}$$

where we used an algebraic property that $4^{-z} \geq \frac{1}{4}, \forall z \in [0, 1]$. Continuing,

$$\begin{aligned}
b(k) &\geq \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{1}{4x_k^2 c^3} \cdot \left(\frac{\ln(2)}{d} - \frac{\ln^2(2)}{2d^2} \right) \left(\frac{k-1}{d} \right) \\
&= \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{\ln(2)}{4x_k^2 c^3 d} \left(\frac{k-1}{d} \right) - \frac{\ln^2(2)}{8x_k^2 c^3 d^2} \left(\frac{k-1}{d} \right) \\
[c \leq 1] &\geq \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{\ln(2)}{4x_k^2 d} \left(\frac{k-1}{d} \right) - \frac{\ln^2(2)}{8x_k^2 c^3 d^2} \left(\frac{k-1}{d} \right) \\
&\geq \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{\ln(2)}{4x_k^2 d} \left(\frac{k-1}{d} \right) - \frac{\ln^2(2)}{8(0.45)^2 c^3 d^2} \cdot 1 \\
&\geq \frac{\ln(4)}{d} - \frac{c^{2k-5}}{x_k^2 d} + \frac{\ln(2)}{4x_k^2 d} \cdot \frac{k-1}{d} - \frac{0.3}{c^3 d^2} \geq \frac{\ln(4)}{d} + \frac{\ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k-5}}{4x_k^2 d} - \frac{0.3}{c^3 d^2}.
\end{aligned}$$

Below, we use the closed-form approximation of x_k , for which we have established $|x_k - \tilde{x}_k| \leq \frac{170.4}{d} = \epsilon$ (lemma 3.47) and $|x_k^2 - \tilde{x}_k^2| \leq 2x_k \epsilon + \epsilon^2$ (claim 3.34). Also, recall that, $\tilde{x}_k = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\frac{k}{d}} \left(\frac{1}{\ln 4} - \frac{k}{d} \right)} = \sqrt{1 - \frac{1}{\ln 4} + c^{2k} \left(\frac{1}{\ln 4} - \frac{k}{d} \right)}$. These relations further refine the lower bound on $b(k)$:

$$\begin{aligned}
b(k) + \frac{0.3}{c^3 d^2} &\geq \frac{\ln(4)}{d} + \frac{\ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k-5}}{4x_k^2 d} = \frac{4x_k^2 \ln(4) + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k-5}}{4x_k^2 d} \\
&\geq \frac{4\tilde{x}_k^2 \ln(4) + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k-5}}{4x_k^2 d} - \frac{4\ln(4)}{4x_k^2 d} |x_k^2 - \tilde{x}_k^2| \\
&\geq \frac{4\tilde{x}_k^2 \ln(4) + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k-5}}{4x_k^2 d} - \frac{\ln(4)}{x_k^2 d} (2x_k \epsilon + \epsilon^2) \\
&= \frac{4 \left(\ln(4) - 1 + c^{2k} \left(1 - \frac{k}{d} \ln(4) \right) \right) + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k-5}}{4x_k^2 d} - \frac{\epsilon^2}{x_k^2 d} - \frac{\ln(16)\epsilon}{x_k d} \\
&\geq \frac{1.545 + \ln(2) \left(\frac{k-1}{d} \right) + 4c^{2k} \left(1 - \frac{k}{d} \ln(4) \right) - 4c^{2k-5}}{4x_k^2 d} - \frac{\epsilon^2}{x_k^2 d} - \frac{\ln(16)\epsilon}{x_k d} \\
&= \frac{1.545 + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k} \ln(4) \frac{k}{d}}{4x_k^2 d} - \frac{c^{2k-5}(1 - c^5)}{x_k^2 d} - \frac{\epsilon^2}{x_k^2 d} - \frac{\ln(16)\epsilon}{x_k d} \\
[3.25] &\geq \frac{1.545 + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k} \ln(4) \frac{k}{d}}{4x_k^2 d} - \frac{5\ln(2)c^{2k-5}}{x_k^2 d^2} - \frac{\epsilon^2}{x_k^2 d} - \frac{\ln(16)\epsilon}{x_k d}.
\end{aligned}$$

Focusing on the left numerator,

$$\begin{aligned}
1.545 + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k} \ln(4) \frac{k}{d} &= 1.545 + \underbrace{\ln(2)}_{>0} \left(\frac{1}{d} (k-1) - 8c^{2k} \frac{k}{d} \right) \\
&= 1.545 + \ln(2) \left((1 - 8c^{2k}) \frac{k}{d} - \frac{1}{d} \right) = 1.545 - \ln(2) \left((8c^{2k} - 1) \frac{k}{d} + \frac{1}{d} \right) \\
&= 1.545 - \ln(2) \left(\left(8 \cdot 4^{-\frac{k}{d}} - 1 \right) \frac{k}{d} + \frac{1}{d} \right).
\end{aligned}$$

To upper bound $g(x) = 8x \cdot 4^{-x}$ (inside $x \in [0, 1]$), we show that

$$0 \stackrel{!}{=} g'(x) = 8 \cdot 4^{-x} - 8x \ln(4) 4^{-x} = 8 \cdot 4^{-x} (1 - x \ln(4)),$$

solved by $x = \frac{1}{\ln(4)}$, which falls inside $x \in [0, 1]$, meaning it is a global optimum.

The second derivative is

$$\begin{aligned}
g''(x) &= (8 \cdot 4^{-x} - 8x \ln(4) \cdot 4^{-x})' = -4^{2-x} \ln(2) - 8 \ln(4) \cdot 4^{-x} (1 - x \ln(4)) \\
g'' \left(\frac{1}{\ln(4)} \right) &= -4^{2-\frac{1}{\ln(4)}} \ln(2) = -\frac{16 \ln(2)}{e} < 0,
\end{aligned}$$

meaning that the $x = \frac{1}{\ln(4)}$ is the global maximum. Also note: $\left(4^{\frac{1}{\ln 4}}\right)^{\ln 4} = 4 \Rightarrow 4^{\frac{1}{\ln 4}} = e$.

Therefore,

$$\begin{aligned}
1.545 + \ln(2) \left(\frac{k-1}{d} \right) - 4c^{2k} \ln(4) \frac{k}{d} &\geq 1.545 - \ln(2) \left(\left(8 \cdot 4^{-\frac{k}{d}} - 1 \right) \frac{k}{d} + \frac{1}{d} \right) \\
&\geq 1.545 - \ln(2) \left(\left(8 \cdot 4^{-\frac{1}{\ln 4}} - 1 \right) \frac{1}{\ln 4} + \frac{1}{d} \right) = 1.545 - \frac{\ln(2)}{d} - \frac{\ln(2)}{2 \ln(2)} \left(\frac{8}{e} - 1 \right) \\
&\geq 1.545 - 0.972 - \frac{0.7}{d} \geq 0.573 - \frac{0.7}{d}.
\end{aligned}$$

Finally,

$$\begin{aligned}
b(k) + \frac{0.3}{c^3 d^2} + \frac{\epsilon^2}{x_k^2 d} + \frac{\ln(16)\epsilon}{x_k d} + \frac{5 \ln(2) c^{2k-5}}{x_k^2 d^2} &\geq \frac{1.545 + \ln(2) \left(\frac{k-1}{d}\right) - 4c^{2k} \ln(4) \frac{k}{d}}{4x_k^2 d} \\
&\geq \frac{0.573 - \frac{0.7}{d}}{4x_k^2 d} \geq \frac{0.14325}{x_k^2 d} - \frac{0.175}{x_k^2 d^2}.
\end{aligned}$$

Going back to $\beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) = \beta_{k+1} b(k)$, the following bound holds:

$$\begin{aligned}
\beta_{k+1} b(k) &\geq \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \left(\frac{0.3}{c^3 d^2} + \frac{\epsilon^2}{x_k^2 d} + \frac{\epsilon \ln(16)}{x_k d} + \frac{5 \ln(2) c^{2k-5}}{x_k^2 d^2} + \frac{0.175}{x_k^2 d^2} \right) \right) \\
[c < 1, k \geq 2] &\geq \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \left(\frac{0.3}{c^3 d^2} + \frac{\epsilon^2}{x_k^2 d} + \frac{\epsilon \ln(16)}{x_k d} + \frac{5 \ln(2)}{c x_k^2 d^2} + \frac{0.175}{x_k^2 d^2} \right) \right) \\
&= \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \frac{1}{x_k} \left(\frac{0.3 x_k}{c^3 d^2} + \frac{\epsilon^2}{x_k d} + \frac{\epsilon \ln(16)}{d} + \frac{5 \ln(2)}{c x_k d^2} + \frac{0.175}{x_k d^2} \right) \right) \\
[0.45 \leq x_k \leq 1] &\geq \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \frac{1}{x_k} \left(\frac{0.3}{c^3 d^2} + \frac{\epsilon \ln(16)}{d} + \frac{5 \ln(2)}{c \cdot 0.45 d^2} + \frac{0.175}{0.45 d^2} + \frac{\epsilon^2}{0.45 d} \right) \right) \\
&\geq \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \frac{1}{x_k} \left(\frac{0.3}{c^3 d^2} + \frac{2.78\epsilon}{d} + \frac{7.71}{c d^2} + \frac{0.39}{d^2} + \frac{2.3\epsilon^2}{d} \right) \right).
\end{aligned}$$

Since $c = 2^{-1/d} \geq 0.9999, \forall d \geq 10,000$, and plugging in $\epsilon = \frac{170.4}{d}$:

$$\begin{aligned}
&\beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) \\
&\geq \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \frac{1}{x_k} \left(\frac{0.3}{0.9999^3 d^2} + \frac{2.78 \cdot 170.4}{d^2} + \frac{7.71}{0.9999 d^2} + \frac{0.39}{d^2} + \frac{2.3 \cdot 170.4^2}{d^3} \right) \right) \\
&\geq \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \frac{1}{x_k} \left(\frac{482.2}{d^2} + \frac{66783.17}{d^3} \right) \right) \\
&\stackrel{(1)}{\geq} \beta_{k+1} \left(\frac{0.14325}{x_k^2 d} - \frac{1}{x_k} \left(\frac{488.88}{d^2} \right) \right) \geq \frac{\beta_{k+1}}{x_k d} \left(\frac{0.14325}{x_k} - \frac{488.88}{d} \right),
\end{aligned}$$

where (1) is since $d \geq 10,000$. The inside of the parenthesis is positive $\forall d \geq \left\lceil \frac{488.88}{0.14325} \right\rceil = 3413$,

so we can bound the expression by lower bounding $\frac{\beta_{k+1}}{x_k^2}$.

$$\begin{aligned}\beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) &\geq \frac{\beta_{k+1}}{x_k d} \left(\frac{0.14325}{x_k} - \frac{488.88}{d} \right) \geq \frac{0.3c^{2k-3}}{d^2} \left(\frac{0.14325}{x_k^2} - \frac{488.88}{x_k d} \right) \\ &\geq c^{2k-3} \left(\frac{0.0429}{x_k^2 d^2} - \frac{146.7}{x_k d^3} \right).\end{aligned}$$

Therefore,

$$\begin{aligned}A_1(x) &\geq x_{t-1} \left(\frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} \beta_{k+1} \left(\frac{\ln(4)}{d} - \frac{\beta_k}{x_k^2} \right) - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{109c^{2k-9}}{x_k^3 d^{9/2}} \right) \\ &\geq x_{t-1} \left(\frac{1}{x_k} \frac{c^{k-3}}{\sqrt{d}} c^{2k-3} \left(\frac{0.0429}{x_k^2 d^2} - \frac{146.7}{x_k d^3} \right) - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{109c^{2k-9}}{x_k^3 d^{9/2}} \right) \\ &= x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{146.7c^{3k-6}}{x_k^2 d^{7/2}} - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{109c^{2k-9}}{x_k^3 d^{9/2}} \right) \\ [c < 1] &\geq x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{146.7c^{3k-9}}{x_k^2 d^{7/2}} - \frac{11.84c^{3k-9}}{x_k^3 d^{7/2}} - \frac{109c^{3k-9}}{c^k x_k^3 d^{9/2}} \right) \\ &\stackrel{(1)}{\geq} x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{146.7c^{3k-9}}{x_k^2 d^{7/2}} - \frac{11.84c^{3k-9}}{0.45x_k^2 d^{7/2}} - \frac{109c^{3k-9}}{0.5 \cdot 0.45x_k^2 d^{9/2}} \right) \\ &\geq x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{173.02c^{3k-9}}{x_k^2 d^{7/2}} - \frac{484.45c^{3k-9}}{x_k^2 d^{9/2}} \right) \\ [d \geq 10,000] &\geq x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{173.07c^{3k-9}}{x_k^2 d^{7/2}} \right),\end{aligned}$$

where (1) is since $x_k \geq 0.45$, $c^k \geq c^d = 0.5$.

□

3.15.4 Conclusion

We want to show positivity of $\Delta_{t,k} \triangleq \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1}$. Applying proposition 3.42, we show $\forall k \geq t$:

$$\begin{aligned}
\Delta_{t,k} &= \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} \\
&\geq A_1(k) + A_2(k) + A_3(k) \\
&\quad - \frac{1}{d^{7/2}} \left(\frac{96c^{6k-15}}{x_k} + 113c^{7k-18} \right) - \frac{1}{d^{9/2}} \left(\frac{56.5c^{9k-18}}{x_k^3} + 614c^{6k-30} \right) \\
&\stackrel{(1)}{\geq} A_1(k) + A_2(k) + A_3(k) \\
&\quad - \frac{1}{d^{7/2}} \left(\frac{96c^{6k-15}}{x_k} + 113c^{7k-18} \right) - \frac{1}{d^{9/2}} \left(\frac{56.5c^{7k-18}}{x_k^3} + 614c^{7k-18}c^{-k-12} \right) \\
&\stackrel{(2)}{\geq} A_1(k) + A_2(k) + A_3(k) \\
&\quad - \frac{1}{d^{7/2}} \left(\frac{96c^{6k-15}}{x_k} + 113c^{7k-18} \right) - \frac{c^{7k-18}}{d^{9/2}} \left(\frac{56.5}{x_k^3} + 614 \cdot 1.00007^{12} \cdot 2 \right) \\
&\stackrel{(3)}{\geq} A_1(k) + A_2(k) + A_3(k) \\
&\quad - \frac{1}{d^{7/2}} \left(\frac{96c^{6k-15}}{0.45} + 113c^{7k-18} \right) - \frac{c^{7k-18}}{d^{9/2}} \left(\frac{56.5}{0.45^3} + 1238.36 \right) \\
&\geq A_1(k) + A_2(k) + A_3(k) - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113c^{7k-18}) - \frac{1858.39c^{7k-18}}{d^{9/2}} \\
&\stackrel{(4)}{\geq} A_1(k) + A_2(k) + A_3(k) - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113.19c^{7k-18}),
\end{aligned}$$

where (1) is since $c < 1$, $k \geq 2$; (2) is since $2^{1/10000} \leq 1.00007$, $c^{-k} \leq c^{-d} = 2$; (3) is since $x_{k-1} \geq 0.45$; and (4) is since $d \geq 10,000$. Plugging in the results of Propositions 3.43, 3.44 and 3.45, we derive

$$\begin{aligned}
\Delta_{t,k} &= \|\mathbf{w}_k - \mathbf{w}_{k+1}\|(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1} - \|\mathbf{w}_{k-1} - \mathbf{w}_k\|(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1} \\
&\geq \underbrace{x_{t-1} \left(\frac{0.0429c^{3k-6}}{x_k^3 d^{5/2}} - \frac{173.07c^{3k-9}}{x_k^2 d^{7/2}} \right)}_{A_1(k)} \underbrace{- \frac{14.88x_{t-1}c^{3k-12}}{x_k^3 d^{7/2}}}_{A_2(k)} \underbrace{- \frac{6.77c^{k+t-6}}{x_k^4} \left(\frac{1}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right)}_{A_3(k)} \\
&\quad - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113.19c^{7k-18})
\end{aligned}$$

$$\begin{aligned}
&= \frac{0.0429x_{t-1}c^{3k-6}}{x_k^3d^{5/2}} - \frac{173.07x_{t-1}c^{3k-9}}{x_k^2d^{7/2}} - \frac{14.88x_{t-1}c^{3k-12}}{x_k^3d^{7/2}} - \frac{6.77c^{k+t-6}}{x_k^4} \left(\frac{1}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right) \\
&\quad - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113.19c^{7k-18}) \\
&\stackrel{(1)}{\geq} \frac{0.0429c^{3k-6}}{x_k^2d^{5/2}} - \frac{173.07x_{t-1}c^{3k-9}}{x_k^2d^{7/2}} - \frac{14.88x_{t-1}c^{3k-12}}{x_k^3d^{7/2}} - \frac{6.77c^{k+t-6}}{x_k^4} \left(\frac{1}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right) \\
&\quad - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113.19c^{7k-18}) \\
&\stackrel{(2)}{\geq} \frac{0.0429c^{3k-6}}{x_k^2d^{5/2}} - \frac{173.07c^{3k-9}}{x_k^2d^{7/2}} - \frac{14.88c^{3k-12}}{x_k^3d^{7/2}} - \frac{6.77c^{k+t-6}}{x_k^4} \left(\frac{1}{d^{7/2}} + \frac{5.42}{d^{9/2}} \right) \\
&\quad - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113.19c^{7k-18}) \\
&\stackrel{(3)}{\geq} \frac{0.0429c^{3k-6}}{x_k^2d^{5/2}} - \frac{173.07c^{3k-9}}{x_k^2d^{7/2}} - \frac{14.88c^{3k-12}}{x_k^3d^{7/2}} - \frac{6.78c^{k+t-6}}{x_k^4d^{7/2}} \\
&\quad - \frac{1}{d^{7/2}}(213.34c^{6k-15} + 113.19c^{7k-18}) \\
&\stackrel{(4)}{\geq} \frac{0.0429c^{3k-6}}{x_k^2d^{5/2}} - \frac{173.07c^{3k-9}}{x_k^2d^{7/2}} - \frac{14.88c^{3k-12}}{x_k^3d^{7/2}} - \frac{6.78c^{k+t-6}}{x_k^4d^{7/2}} \\
&\quad - \frac{213.34c^{6k-15} + 113.19c^{7k-18}}{x_k^2d^{7/2}} \\
&= \frac{1}{x_k^2d^{5/2}} \left(0.0429c^{3k-6} - \frac{1}{d} \left(173.07c^{3k-9} \right. \right. \\
&\quad \left. \left. + \frac{14.88c^{3k-12}}{x_k} + \frac{6.78c^{k+t-6}}{x_k^2} + 213.34c^{6k-15} + 113.19c^{7k-18} \right) \right) \\
&\stackrel{(5)}{\geq} \frac{1}{x_k^2d^{5/2}} \left(0.0429c^{3k-6} - \frac{1}{d} \left(173.07c^{3k-9} \right. \right. \\
&\quad \left. \left. + \frac{14.88c^{3k-12}}{0.45} + \frac{6.78c^{k+t-6}}{0.45^2} + 213.34c^{6k-15} + 113.19c^{7k-18} \right) \right) \\
&\geq \frac{1}{x_k^2d^{5/2}} \left(0.0429c^{3k-6} - \frac{1}{d} (173.07c^{3k-9} \right. \\
&\quad \left. + 33.07c^{3k-12} + 33.49c^{k+t-6} + 213.34c^{6k-15} + 113.19c^{7k-18}) \right) \\
&= \frac{c^{3k-6}}{x_k^2d^{5/2}} \left(0.0429 - \frac{1}{d} (173.07c^{-3} \right. \\
&\quad \left. + 33.07c^{-6} + 33.49c^{-2k+t} + 213.34c^{3k-9} + 113.19c^{4k-12}) \right) \\
&\stackrel{(6)}{\geq} \frac{c^{3k-6}}{x_k^2d^{5/2}} \left(0.0429 - \frac{1}{d} (173.07c^{-3} + 33.07c^{-6} + 33.49c^{-2k} + 213.34c^{-3} + 113.19c^{-4}) \right)
\end{aligned}$$

$$\begin{aligned}
&\stackrel{(7)}{\geq} \frac{c^{3k-6}}{x_k^2 d^{5/2}} \left(0.0429 - \frac{1}{d} (173.07c^{-3} + 33.07c^{-6} + 33.49 \cdot 4 + 213.34c^{-3} + 113.19c^{-4}) \right) \\
&\stackrel{(8)}{\geq} \frac{c^{3k-6}}{x_k^2 d^{5/2}} \left(0.0429 - \frac{1}{d} (173.07 \cdot 0.9999^{-3} \right. \\
&\quad \left. + 33.07 \cdot 0.9999^{-6} + 33.49 \cdot 4 + 213.34 \cdot 0.9999^{-3} + 113.19 \cdot 0.9999^{-4}) \right),
\end{aligned}$$

where (1) is since $x_{t-1} > x_k$; (2) is since $x_{t-1} \leq 1$; (3) is since $d \geq 10,000$; (4) is since $x_k \leq 1$; (5) is since $x_k \geq 0.45$; (6) is since $c < 1$, $k \geq 2$; (7) is since $c^{-2k} = 4^{k/d} \leq 4$; and (8) is since $d \geq 10,000 \Rightarrow c \geq 0.9999$.

Finally,

$$\Delta_{t,k} \geq \frac{c^{3k-6}}{x_k^2 d^{5/2}} \left(0.0429 - \frac{666.82}{d} \right). \quad (3.13)$$

Hence, a sufficient condition for $\frac{(\mathbf{w}_{k-1} - \mathbf{w}_k)^\top \mathbf{w}_{t-1}}{\|\mathbf{w}_{k-1} - \mathbf{w}_k\|} - \frac{(\mathbf{w}_k - \mathbf{w}_{k+1})^\top \mathbf{w}_{t-1}}{\|\mathbf{w}_k - \mathbf{w}_{k+1}\|}$ to be positive and monotonicity to hold, is that $d \geq \lceil \frac{666.82}{0.0429} \rceil = 15,544$. Since this is smaller than 25,000, this concludes our proof of positivity of $\Delta_{t,k}$.

3.16 Lower bound technical appendix: Properties of the recursive construction

This section complements section 3.11.1 by confirming that the recursive construction introduced there is well-defined; readers are encouraged to review it first for context.

Specifically, we prove that the recurrence defining the sequence (x_k) is well-posed, in the sense that the square root is always taken over a nonnegative quantity:

Lemma 3.46 (Existence of the recursive sequence). Given the sequence $(x)_k$ recursively defined by $x_1 = 1$, $x_k = \frac{x_{k-1} + \sqrt{x_{k-1}^2 - 4\beta_k}}{2}$, $\forall k \in \{2, \dots, d\}$ where $c \triangleq 2^{-1/d}$ and $\beta_k \triangleq$

$\frac{((k-1)c-(k-2))c^{2k-5}}{d}$, for all $d \geq 30$ and $k \in \{2, \dots, d\}$,

$$x_{k-1}^2 - 4\beta_k \geq 0.$$

In addition, we prove the following lemma:

Lemma 3.47 (Approximation by closed-form reference). Given the sequence $(x)_k$ recursively defined by $x_1 = 1$, $x_k = \frac{x_{k-1} + \sqrt{x_{k-1}^2 - 4\beta_k}}{2}$, $\forall k \in \{2, \dots, d\}$ where $c \triangleq 2^{-1/d}$ and $\beta_k \triangleq \frac{((k-1)c-(k-2))c^{2k-5}}{d}$, and the sequence $\tilde{x}_k = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\frac{k}{d}} \left(\frac{1}{\ln 4} - \frac{k}{d}\right)}$, for all $d \geq 30$ and $k \in [d]$:

$$|x_k - \tilde{x}_k| \leq \frac{170.4}{d}.$$

Before proving this lemma, we state the following corollary:

Corollary 3.48 (Lower bound on x_k). $\forall d \geq 25,000$, $\forall k \in [d]$: $x_k \geq 0.45$.

Proof. x_k is decreasing (claim 3.29), so $\forall k \in [d]$:

$$x_k \geq x_d \geq \tilde{x}_d - \frac{170.4}{d} = \sqrt{1 - \frac{1}{\ln 4} + 4^{-1} \left(\frac{1}{\ln 4} - 1\right)} - \frac{170.4}{d}$$

$$[d \geq 25,000] \geq 0.45.$$

□

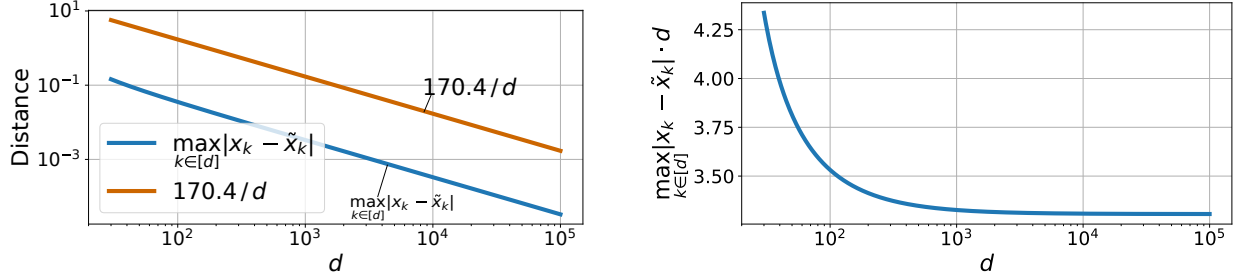
This bound is used in the proof of section 3.15.

3.16.1 Proof outline

First, we show the above holds numerically for $30 \leq d < 100,000$, as can be seen in fig. 3.26. We then prove analytically for $d \geq 100,000$, by constructing an ODE for which the sequence $(x)_k$ serves as an Euler trajectory. We then bound the distance between the solution to this ODE and a known function $\tilde{x}(\tau)$. Combining this bound with Euler's method global

truncation error bound, we obtain a bound for the distance between $(x)_k$ and $(\tilde{x})_k$. We then use this bound to show the existence of the sequence $(x)_k$ for all $k \in \{2, \dots, d\}$.

Computational resources The numerical validation took 6 hours to run on a home PC with i5-9400F CPU and 16GB RAM.



(a) $|x_k - \tilde{x}_k| \leq \frac{170.4}{d}$. This is an analytically derived upper bound.

(b) Actual upper bound is $< \frac{4.5}{d}$.

Figure 3.26: **Numerical proof of lemma 3.47 for $d < 100,000$.** Using the recursive definition of x_k , we calculated the sequence for each value of d , $\forall k \in [d]$, and compared with $\tilde{x}_k$.

Euler's method construction and bottom line

Here, we leverage the global truncation error of Euler's method to establish the bound. The auxiliary propositions supporting this result are proved in the following section. Define

$$f(\tau, x) = d \frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2},$$

$$\beta(\tau) = \frac{((d\tau - 1)2^{-1/d} - (d\tau - 2))2^{(5-2d\tau)/d}}{d}.$$

Then, using step size $h = \frac{1}{d}$, Euler's method gives the iterates

$$x_{k+1} = x_k + h \cdot f(\tau_k, x_k),$$

$$\tau_{k+1} = \tau_k + h,$$

and thus

$$x_{k+1} = x_k + \frac{\sqrt{x_k^2 - 4\beta\left(\frac{k+1}{d}\right)} - x_k}{2} = \frac{x_k + \sqrt{x_k^2 - 4\beta_{k+1}}}{2},$$

which are the desired iterates.

These are the Euler's iterates for the differential equation

$$x'(\tau) = f(\tau, x(\tau)), \quad x(0) = 1. \quad (3.14)$$

We do not find an exact solution to this equation. Instead, for $d \geq 100,000$, proposition 3.64 will prove

$$|x(\tau) - \tilde{x}(\tau)| \leq \frac{38.9822}{d}$$

where the function $\tilde{x}$, suggested in Hucht [59], is defined by

$$\tilde{x}(\tau) = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\tau} \left(\frac{1}{\ln 4} - \tau \right)},$$

so that $\tilde{x}_k = \tilde{x}\left(\frac{k}{d}\right)$. Next, we bound the iterates using the global truncation error of Euler's method, obtaining

$$\left| x_k - x\left(\frac{k}{d}\right) \right| \leq \frac{131.3685}{d}.$$

Combining this with the previous result, proposition 3.71 yields

$$|x_k - \tilde{x}_k| \leq \frac{131.3685}{d} + \frac{38.9822}{d} \leq \frac{170.4}{d}.$$

Finally, we use this bound to show the iterates exist $\forall k \in [d]$.

3.16.2 Full proof

Auxiliary propositions

We begin with preliminary claims and move on to the propositions used in the previous section.

Claim 3.49. $f(d) \triangleq -d(1 - 2^{-1/d})$ is decreasing $\forall d \geq 1$.

Proof. $f'(d) = -(1 - 2^{-1/d}) - d((-1) \cdot \frac{1}{d^2} \ln 2 \cdot 2^{-1/d}) = -(1 - 2^{-1/d}) + \frac{\ln 2}{d} 2^{-1/d} = -1 - (1 - \frac{\ln 2}{d}) 2^{-1/d} < 0$. □

Claim 3.50. $\forall d \geq 1 : d(2^{1/d} - 1) \geq \ln 2$.

Proof. Using Taylor's expansion:

$$\begin{aligned} 2^{1/d} &= e^{\frac{\ln 2}{d}} = 1 + \frac{\ln 2}{d} + \sum_{i=2}^{\infty} \frac{1}{i!} \left(\frac{\ln 2}{d} \right)^i \\ \Rightarrow d(2^{1/d} - 1) &= \ln 2 + \sum_{i=2}^{\infty} \frac{1}{i!} \frac{(\ln 2)^i}{d^{i-1}} \geq \ln 2. \end{aligned}$$

□

Claim 3.51. $-d(1 - 2^{-1/d}) \geq -\ln 2$ (alternatively: $2^{-1/d} \geq 1 - \frac{\ln 2}{d}$).

Proof. By claim 3.49, $-d(1 - 2^{-1/d})$ is decreasing with d , so

$$-d(1 - 2^{-1/d}) \geq \lim_{d \rightarrow \infty} -d(1 - 2^{-1/d}) = \lim_{h \rightarrow 0^+} \frac{2^{-h} - 1}{h} = \lim_{h \rightarrow 0^+} \frac{2^{-h} - 2^0}{h}.$$

We recognize this as the definition of the derivative of 2^{-x} for $x = 0^+$. Thus:

$$\lim_{d \rightarrow \infty} -d(1 - 2^{-1/d}) = \frac{d(2^{-x})}{dx}(x = 0^+) = -\ln 2 \cdot 2^0 = -\ln 2.$$

□

Claim 3.52. $\beta(\tau) > 0$, decreasing and convex for $\tau < \frac{1}{\ln 2}$.

Proof. Recall that $\beta(\tau) = \frac{((d\tau-1)2^{-1/d} - (d\tau-2))2^{(5-2d\tau)/d}}{d}$, and $d \geq 1$.

Denote $\beta(\tau) = \frac{1}{d}f(\tau)g(\tau)$, where $f(\tau) = (d\tau-1)2^{-1/d} - (d\tau-2)$ and $g(\tau) = 2^{\frac{5-2d\tau}{d}}$.

$g(\tau) > 0$ for all τ .

From claim 3.51, $1 - \frac{\ln 2}{d} \leq 2^{-1/d} \leq 1$, so:

$$f(\tau) = 2^{-1/d}d\tau - 2^{-1/d} - d\tau + 2 \geq \left(1 - \frac{\ln 2}{d}\right)d\tau - 1 - d\tau + 2 = -\tau \ln 2 + 1,$$

so $f(\tau) > 0$ for $\tau < \frac{1}{\ln 2}$. Thus $\beta(\tau) > 0$ for $\tau < \frac{1}{\ln 2}$.

Also, $f'(\tau) = d(2^{-1/d} - 1) < 0$, $\forall d \geq 1$, $\forall \tau$ and $g'(\tau) = -2 \ln 2 \cdot 2^{\frac{5-2d\tau}{d}} < 0$, $\forall d$, $\forall \tau$

So:

$$\beta'(\tau) = \frac{1}{d}(f'(\tau)g(\tau) + g'(\tau)f(\tau)) < 0,$$

as long as $g(\tau) > 0$ and $f(\tau) \geq 0$ which holds for $\tau < \frac{1}{\ln 2}$.

Now note $f''(\tau) = 0$ and $g''(\tau) = 4 \ln^2 2 \cdot 2^{\frac{5-2d\tau}{d}} > 0$, so:

$$\begin{aligned} \beta''(\tau) &= \frac{1}{d}(f''(\tau)g(\tau) + f'(\tau)g'(\tau) + g''(\tau)f(\tau) + f'(\tau)g'(\tau)) \\ &= \frac{1}{d}(2f'(\tau)g'(\tau) + g''(\tau)f(\tau)) > 0, \end{aligned}$$

as long as $f(\tau) \geq 0$ which holds for $\tau < \frac{1}{\ln 2}$. □

Claim 3.53. $x(\tau)$ defined by the ODE in eq. (3.14) satisfies the ODE $x(0) = 1$, $x'(\tau) = d\sqrt{\frac{x^2 - g(\tau) - x}{2}}$ with $g(\tau) = 4\beta(\tau + \frac{1}{d})$. We also have:

$$\forall d \geq 3, \quad \max_{s \in [0,1]} g(s) = g(0) = 4\frac{2^{3/d}}{d}.$$

Proof. Substituting $x'(\tau) = d\sqrt{\frac{x^2 - 4\beta(\tau + \frac{1}{d}) - x}{2}}$ from eq. (3.14) into $x'(\tau) = d\sqrt{\frac{x^2 - g(\tau) - x}{2}}$ gives $g(\tau) = 4\beta(\tau + \frac{1}{d})$.

For $\tau \in [0, 1]$ and $d \geq 3$, $\tau + \frac{1}{d} \leq \frac{1}{\ln 2}$. By claim 3.52, β is decreasing, so:

$$\max_{s \in [0, 1]} g(s) = g(0) = 4\beta\left(\frac{1}{d}\right) = 4 \frac{((1-1)2^{-1/d} - (1-2))2^{(5-2)/d}}{d} = 4 \frac{2^{3/d}}{d}.$$

□

Remark 3.54. The solution to the ODE $x(\tau)x'(\tau) = -f(x)$, $x(0) = 1$ is

$$x(\tau) = \sqrt{1 - 2 \int_0^\tau f(s) ds}.$$

Claim 3.55. $\forall 0 \leq a \leq \mu \leq 1 : 1 - \frac{1-\sqrt{1-\mu}}{\mu}a \leq \sqrt{1-a} \leq 1 - \frac{a}{2}$.

Proof. For the right-side inequality, $(1 - \frac{a}{2})^2 = 1 - a + \frac{a^2}{4} \geq 1 - a = (\sqrt{1-a})^2$.

For the left side: denote $f(a) = \sqrt{1-a}$. f is concave: $f'(a) = -\frac{1}{2\sqrt{1-a}}$, $f''(a) = \frac{-2}{4(1-a)^{3/2}} \leq 0$. Thus, for $0 \leq a \leq \mu \leq 1$:

$$\begin{aligned} \sqrt{1-a} = f(a) &\geq \frac{(\mu-a)f(0) + af(\mu)}{\mu} \\ &= 1 - \frac{a}{\mu} + \frac{a}{\mu}\sqrt{1-\mu} = 1 - \frac{1-\sqrt{1-\mu}}{\mu}a. \end{aligned}$$

□

Remark 3.56. The solution to the ODE in eq. (3.14) is only defined when the quantity under the square root remains nonnegative, i.e.,

$$x^2(\tau) \geq 4\beta\left(\tau + \frac{1}{d}\right).$$

Given that $x(0) = 1$ and that $4\beta\left(\frac{1}{d}\right) = 4 \frac{2^{3/d}}{d} < 1$ for all $d \geq 6$, i.e., $x^2(0) > 4\beta\left(\frac{1}{d}\right)$ strictly, from continuity we have that $x(\tau) \geq \sqrt{4\beta\left(\tau + \frac{1}{d}\right)} > 0$ for all $\tau \in [0, \zeta]$, for some $\zeta \in (0, 1]$.

Going forward we focus on $\tau \in [0, \zeta]$ when stating facts about $x(\tau)$, and eventually show that $\zeta = 1$ in corollary 3.68.

Proposition 3.57. Assuming $\forall \tau \in [0, \zeta] : 0 \leq \frac{g(\tau)}{x^2} \leq 1$ and $x > 0$, the solution of $x(0) = 1$, $x'(\tau) = d \frac{\sqrt{x^2 - g(\tau)} - x}{2}$ satisfies

$$x(\tau) \in \left[\sqrt{1 - d \frac{1 - \sqrt{1 - \mu}}{\mu} \int_0^\tau g(s) ds}, \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} \right],$$

for $\mu \triangleq \max_{s \in [0, \zeta]} \frac{g(s)}{x(s)^2}$.

Proof. Let $0 \leq \mu \leq 1$ such that $\forall \tau \in [0, \zeta] : 0 \leq \frac{g(\tau)}{x^2} \leq \mu \leq 1$. By claim 3.55:

$$\begin{aligned} 1 - \frac{1 - \sqrt{1 - \mu}}{\mu} \frac{g(\tau)}{x^2} &\leq \sqrt{1 - \frac{g(\tau)}{x^2}} \leq 1 - \frac{g(\tau)}{2x^2} \\ \left(1 - \frac{1 - \sqrt{1 - \mu}}{\mu} \frac{g(\tau)}{x^2}\right) x &\leq x \sqrt{1 - \frac{g(\tau)}{x^2}} \leq \left(1 - \frac{g(\tau)}{2x^2}\right) x \\ x - \frac{1 - \sqrt{1 - \mu}}{\mu} \frac{g(\tau)}{x} &\leq \sqrt{x^2 - g(\tau)} \leq x - \frac{g(\tau)}{2x} \\ -\frac{1 - \sqrt{1 - \mu}}{\mu} \frac{g(\tau)}{x} &\leq \sqrt{x^2 - g(\tau)} - x \leq -\frac{g(\tau)}{2x} \\ -d \frac{1 - \sqrt{1 - \mu}}{2\mu} \frac{g(\tau)}{x} &\leq d \frac{\sqrt{x^2 - g(\tau)} - x}{2} \leq -\frac{dg(\tau)}{4x} \\ \frac{(\sqrt{1 - \mu} - 1) dg(\tau)}{2\mu x(\tau)} &\leq x'(\tau) \leq \frac{-dg(\tau)}{4x(\tau)} \\ \frac{(\sqrt{1 - \mu} - 1) dg(\tau)}{2\mu} &\leq x(\tau) x'(\tau) \leq \frac{-dg(\tau)}{4}, \end{aligned}$$

where the last transition is valid since $x > 0$. From remark 3.54, we know that the solution to the ODE $x(\tau) x'(\tau) = -f(x)$, $x(0) = 1$ is $x(\tau) = \sqrt{1 - 2 \int_0^\tau f(s) ds}$. Put differently this means $x(\tau) = \sqrt{1 + 2 \int_0^\tau x(s) x'(s) ds}$ (when $x(0) = 1$). We aim to plug this into the

inequalities, so we now achieve the required form:

$$\begin{aligned}
& \int_0^\tau \frac{(\sqrt{1-\mu}-1) dg(s)}{2\mu} ds \leq \int_0^\tau x(s) x'(s) ds \leq \int_0^\tau \frac{-dg(s)}{4} ds \\
& 1 - 2 \int_0^\tau \frac{d}{2} \frac{(1-\sqrt{1-\mu})}{\mu} g(s) ds \leq 1 + 2 \int_0^\tau x(s) x'(s) ds \leq 1 - 2 \int_0^\tau \frac{d}{4} g(s) ds \\
& 1 - d \frac{1-\sqrt{1-\mu}}{\mu} \int_0^\tau g(s) ds \leq 1 + 2 \int_0^\tau x(s) x'(s) ds \leq 1 - \frac{d}{2} \int_0^\tau g(s) ds.
\end{aligned}$$

Denoting $A \triangleq d \int_0^\tau g(s) ds \geq 0$, for the LHS,

$$1 - \frac{1-\sqrt{1-\mu}}{\mu} A = \frac{1}{\mu} (\mu - A + A\sqrt{1-\mu}) \geq \frac{1}{\mu} (\mu - A + A(1-\mu)) = 0.$$

Hence it is legal to take a square root:

$$\sqrt{1 - d \frac{1-\sqrt{1-\mu}}{\mu} \int_0^\tau g(s) ds} \leq \sqrt{1 + 2 \int_0^\tau x(s) x'(s) ds} \leq \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds}.$$

Finally, plugging in $x(\tau) = \sqrt{1 + 2 \int_0^\tau x(s) x'(s) ds}$, we get:

$$\sqrt{1 - d \frac{1-\sqrt{1-\mu}}{\mu} \int_0^\tau g(s) ds} \leq x(\tau) \leq \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds}.$$

□

Proposition 3.58. For $d \geq 100,000$ and $\tau \in [0, 1]$, $0 \leq d \int_0^\tau g(s) ds \leq 1.5821$.

Proof. For $\tau \in [0, 1]$ and $d \geq 3$, $\tau + \frac{1}{d} < \frac{1}{\ln 2}$. By claim 3.52, β is positive and thus

$d \int_0^\tau g(s) ds = 4d \int_0^\tau \beta\left(s + \frac{1}{d}\right) ds \geq 0$. For the right-side inequality:

$$\begin{aligned}
d \int_0^\tau g(s) ds &= 4d \int_0^\tau \beta\left(s + \frac{1}{d}\right) ds \\
[\beta \geq 0] &\leq 4d \int_0^1 \beta\left(s + \frac{1}{d}\right) ds \\
&= 4d \int_0^1 \left(\frac{((ds + 1 - 1) 2^{-1/d} - (ds + 1 - 2)) 2^{(5-2ds-2)/d}}{d} \right) ds \\
&= 4 \cdot 2^{3/d} \left[\int_0^1 2^{-2s} ds - d(1 - 2^{-1/d}) \int_0^1 s 2^{-2s} ds \right] \\
&= 4 \cdot 2^{3/d} \left[\left[-\frac{2^{-2s}}{\ln 4} \right]_0^1 - d(1 - 2^{-1/d}) \left[-\frac{2^{-2s}(s \ln 4 + 1)}{\ln^2 4} \right]_0^1 \right] \\
&= 4 \cdot 2^{2/d} \left[2^{1/d} \frac{3}{4 \ln 4} - d(2^{1/d} - 1) \left[\frac{4 - (\ln 4 + 1)}{4 \ln^2 4} \right] \right].
\end{aligned}$$

By claim 3.50, $d(2^{1/d} - 1) \geq \ln 2$, so:

$$\begin{aligned}
d \int_0^\tau g(s) ds &\leq 4d \int_0^1 \beta\left(s + \frac{1}{d}\right) ds \leq 4 \cdot 2^{2/d} \left[2^{1/d} \frac{3}{4 \ln 4} - \ln 2 \left[\frac{3 - \ln 4}{4 \ln^2 4} \right] \right] \\
[d \geq 100,000] &\leq 4 \cdot 2^{2/100000} \left[2^{1/100000} \frac{3}{4 \ln 4} - \ln 2 \left[\frac{3 - \ln 4}{4 \ln^2 4} \right] \right] \leq 1.5821.
\end{aligned}$$

□

Claim 3.59. $\frac{1-\sqrt{1-x}}{x} - \frac{1}{2} \leq \frac{x}{2}$ for $x \in (0, 1]$.

Proof.

$$\frac{1 - \sqrt{1-x}}{x} = \frac{1 - \sqrt{1-x}}{x} \frac{1 + \sqrt{1-x}}{1 + \sqrt{1-x}} = \frac{x}{x(1 + \sqrt{1-x})},$$

so we define $a(x) \triangleq \frac{1}{1+\sqrt{1-x}}$.

This function is monotonically increasing, continuous and convex for $x \in [0, 1]$:

$$\begin{aligned} a'(x) &= \frac{\frac{1}{2\sqrt{1-x}}}{(1 + \sqrt{1-x})^2}, \\ a''(x) &= \frac{-1 \cdot \left(2 \frac{-1}{2\sqrt{1-x}} (1 + \sqrt{1-x})^2 + 2 (1 + \sqrt{1-x}) \frac{-1}{2\sqrt{1-x}} \cdot 2\sqrt{1-x}\right)}{4(1-x)(1 + \sqrt{1-x})^4} \\ &= \frac{\frac{1}{\sqrt{1-x}} (1 + \sqrt{1-x})^2 + 2 (1 + \sqrt{1-x})}{4(1-x)(1 + \sqrt{1-x})^4} \geq 0, \end{aligned}$$

and thus by convexity, for $x \in (0, 1]$:

$$\begin{aligned} \frac{1 - \sqrt{1-x}}{x} &= a(x) \leq (1-x)a(0) + xa(1) = (1-x)\frac{1}{2} + x = \frac{1}{2} + \frac{1}{2}x \\ \implies 2 \left(\frac{1 - \sqrt{1-x}}{x} - \frac{1}{2} \right) &\leq x. \end{aligned}$$

□

Proposition 3.60. For $d \geq 100,000$, $\mu \triangleq \max_{s \in [0, \zeta]} \frac{g(s)}{x(s)^2} \leq \frac{19.158}{d}$.

Proof. By remark 3.56, $x(\tau)$ is positive, and since $\beta(\tau + \frac{1}{d}) > 0$, $x(\tau)$ is decreasing in $[0, \zeta]$ (claim 3.52), and thus the minimum of $x(\tau)^2$ is $x(\zeta)^2$, and $x(\zeta)^2 \leq x(0)^2 = 1$.

Applying the upper bound on $g(s)$ from claim 3.53 gives

$$\mu \triangleq \max_{s \in [0, \zeta]} \frac{g(s)}{x(s)^2} \leq \frac{\max_{s \in [0, \zeta]} g(s)}{\min_{s \in [0, \zeta]} x(s)^2} = \frac{4 \frac{2^{3/d}}{d}}{x(\zeta)^2}.$$

By proposition 3.57, $\sqrt{1 - d \frac{1 - \sqrt{1-\mu}}{\mu} \int_0^\zeta g(s) ds} \leq x(\zeta)$.

Squaring both (positive) sides, substituting and denoting $A = d \int_0^\zeta g(s) ds$, we get

$$1 - \frac{1 - \sqrt{1-\mu}}{\mu} A \leq x(\zeta)^2.$$

$A \geq 0$ by proposition 3.58, and $f(\mu) \triangleq \frac{1 - \sqrt{1-\mu}}{\mu}$ is increasing, since using claim 3.55, $f'(\mu) =$

$\frac{2-\mu-2\sqrt{1-\mu}}{2\mu^2\sqrt{1-\mu}} \geq \frac{2-\mu-2(1-\mu/2)}{2\mu^2\sqrt{1-\mu}} = 0$. Combining these and $\mu \leq \frac{4^{2^{3/d}}}{x(\zeta)^2}$, we get

$$\begin{aligned} 1 - \frac{1 - \sqrt{1 - \frac{4^{2^{3/d}}}{x(\zeta)^2}}}{4^{2^{3/d}}/x(\zeta)^2} A &\leq 1 - \frac{1 - \sqrt{1 - \mu}}{\mu} A \leq x(\zeta)^2 \\ \frac{4^{2^{3/d}}}{x(\zeta)^2} - A + A \sqrt{1 - \frac{4^{2^{3/d}}}{x(\zeta)^2}} &\leq 4 \frac{2^{3/d}}{d} \\ A \sqrt{1 - \frac{4^{2^{3/d}}}{x(\zeta)^2}} &\leq -4 \frac{2^{3/d}}{d} \left(\frac{1}{x(\zeta)^2} - 1 \right) + A. \end{aligned}$$

Denote $z = \frac{1}{x(\zeta)^2}$, $r = 4 \frac{2^{3/d}}{d}$. Recall that $z \geq 1$, and we are looking for an upper bound for it, so we can have a lower bound for $x(\zeta)^2$. We have:

$$\begin{aligned} A^2 (1 - rz) &\leq (-r(z - 1) + A)^2 = r^2 (z - 1)^2 - 2Ar(z - 1) + A^2 \\ -A^2 rz &\leq r^2 z^2 - 2r^2 z + r^2 - 2Arz + 2Ar \\ 0 &\leq rz^2 + (A^2 - 2A - 2r)z + 2A + r. \end{aligned}$$

Finding the roots, $z_{1,2} = \frac{2r + A(2 - A) \pm \sqrt{(2r + A(2 - A))^2 - 4r(2A + r)}}{2r}$.

Since we are looking for an upper bound, we care about the smaller root:

$$\begin{aligned} z &\leq \frac{2r + A(2 - A) - \sqrt{4r^2 + 4rA(2 - A) + A^2(2 - A)^2 - 8rA - 4r^2}}{2r} \\ &= \frac{2r + A(2 - A) - \sqrt{-4rA^2 + A^2(2 - A)^2}}{2r} = 1 + \frac{A(2 - A)}{2r} \left(1 - \sqrt{1 - \frac{4r}{(2 - A)^2}} \right). \end{aligned}$$

For $d \geq 100,000$, $\frac{4r}{(2-A)^2} = \frac{4 \cdot 4 \frac{2^{3/d}}{d}}{(2-A)^2} \leq \frac{4 \cdot 4 \cdot \frac{2^{3/100000}}{100000}}{(2-1.5821)^2} \leq 10^{-3} < 1$, (we used proposition 3.58),

so we can apply claim 3.59:

$$\begin{aligned}
z &\leq 1 + \frac{A(2-A)}{2r} \left(\frac{4r}{2(2-A)^2} \left(\frac{4r}{(2-A)^2} + 1 \right) \right) \\
&= 1 + \frac{A}{2-A} + \frac{4Ar}{(2-A)^3} \\
\left[\begin{array}{l} d \geq 100,000 \\ \Rightarrow r \leq 4.1 \cdot 10^{-5} \end{array} \right] &\leq 1 + \frac{A}{2-A} + 4 \cdot 4.1 \cdot 10^{-5} \frac{A}{(2-A)^3} \\
[A \leq 1.5821, 3.58] &\leq 1 + \frac{1.5821}{0.4179} + 4 \cdot 4.1 \cdot 10^{-5} \cdot \frac{1.5821}{0.4179^3} \leq 4.7894.
\end{aligned}$$

Then, for $d \geq 10^5$:

$$\frac{1}{x(\zeta)^2} \leq 4.7894 \Rightarrow \mu \leq 4.7894 \cdot 4 \frac{2^{3/d}}{d} \leq \frac{19.1576 \cdot 2^{3/10^5}}{d} \leq \frac{19.158}{d}.$$

□

Proposition 3.61. For $d \geq 100,000$ and all $\tau \in [0, \zeta]$, $\left| x(\tau) - \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} \right| \leq \frac{33.1539}{d}$.

Proof. Denote $A = d \int_0^\tau g(s) ds$. Then

$$\begin{aligned}
\sqrt{1 - \frac{(1 - \sqrt{1-\mu})}{\mu} A} &\leq x(\tau) \leq \sqrt{1 - \frac{1}{2} A} \\
\left| x(\tau) - \sqrt{1 - \frac{1}{2} A} \right| &\leq \sqrt{1 - \frac{1}{2} A} - \sqrt{1 - \frac{(1 - \sqrt{1-\mu})}{\mu} A} \\
&\leq \frac{1 - \frac{1}{2} A - 1 + \frac{(1 - \sqrt{1-\mu})}{\mu} A}{\sqrt{1 - \frac{1}{2} A} + \sqrt{1 - \frac{(1 - \sqrt{1-\mu})}{\mu} A}} \\
&\leq \frac{\left(\frac{(1 - \sqrt{1-\mu})}{\mu} - \frac{1}{2} \right) A}{\sqrt{1 - \frac{1}{2} A} + \sqrt{1 - \frac{(1 - \sqrt{1-\mu})}{\mu} A}} \\
&\leq \frac{A}{\sqrt{1 - \frac{1}{2} A}} \left(\frac{(1 - \sqrt{1-\mu})}{\mu} - \frac{1}{2} \right).
\end{aligned}$$

By proposition 3.58, for $d \geq 100,000$, $A \leq 1.5821$, then $\frac{A}{\sqrt{1-\frac{1}{2}A}} \leq \frac{1.5821}{\sqrt{1-\frac{1.5821}{2}}} \leq 3.4611$.

We further know from claim 3.59 that $\frac{1-\sqrt{1-\mu}}{\mu} - \frac{1}{2} \leq \frac{\mu}{2}$ for $\mu \in (0, 1]$.

Combining these with proposition 3.60 gives:

$$\left| x(\tau) - \sqrt{1 - \frac{1}{2}A} \right| \leq 3.4611 \frac{\mu}{2} \leq \frac{3.4611 \cdot 19.158}{2d} \leq \frac{33.1539}{d}.$$

□

Claim 3.62. $\forall x \in [0, 1] : 2^x \leq 1 + x$.

Proof. 2^x is convex, so on $[0, 1]$:

$$2^x \leq (1-x)2^0 + x2^1 = 1 - x + 2x = 1 + x.$$

□

Proposition 3.63. For $d \geq 100,000$, $\left| \tilde{x}(\tau) - \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} \right| \leq \frac{5.8283}{d}$.

Proof. Recall:

$$\begin{aligned} \tilde{x}(\tau) &= \sqrt{1 - \frac{1}{\ln 4} + 4^{-\tau} \left(\frac{1}{\ln 4} - \tau \right)}, \\ \beta(\tau) &= \frac{((d\tau - 1)2^{-1/d} - (d\tau - 2))2^{(5-2d\tau)/d}}{d}, \\ g(\tau) &= 4\beta \left(\tau + \frac{1}{d} \right) = 4 \frac{(d\tau 2^{-1/d} - (d\tau - 1))2^{(3-2d\tau)/d}}{d}. \end{aligned}$$

Define $A(\tau) \triangleq \frac{d}{2} \int_0^\tau g(s) ds$, $B(\tau) = \frac{1}{\ln 4} - 4^{-\tau} \left(\frac{1}{\ln 4} - \tau \right)$.

$A(0) = B(0) = 0$, and $A(\tau) - B(\tau)$ is nonnegative and increasing for $\tau \in [0, 1]$:

$$\begin{aligned} \frac{d(A(\tau) - B(\tau))}{d\tau} &= \frac{d}{2} g(\tau) - 4^{-\tau} (2 - \tau \ln 4) \\ &= \frac{d}{2} 4 \frac{(d\tau 2^{-1/d} - (d\tau - 1))2^{(3-2d\tau)/d}}{d} - 4^{-\tau} (2 - \tau \ln 4) \\ &= 4^{-\tau} (2(2^{3/d} - 1) + \tau (\ln 4 - 2 \cdot 2^{3/d} d (1 - 2^{-1/d}))). \end{aligned}$$

If $\ln 4 \geq 2 \cdot 2^{3/d} d (1 - 2^{-1/d})$, then the derivative is positive. Otherwise:

$$\begin{aligned}
\frac{d(A(\tau) - B(\tau))}{d\tau} &= 4^{-\tau} (2(2^{3/d} - 1) - \tau(2 \cdot 2^{3/d} d (1 - 2^{-1/d}) - \ln 4)) \\
[\tau \in [0, 1]] &\geq 4^{-\tau} (2(2^{3/d} - 1) - (2 \cdot 2^{3/d} d (1 - 2^{-1/d}) - \ln 4)) \\
&= 4^{-\tau} (2 \cdot 2^{3/d} (1 - d(1 - 2^{-1/d})) - 2 + \ln 4) \\
[3.51] &\geq 4^{-\tau} (2 \cdot 2^{3/d} (1 - \ln 2) - 2 + \ln 4) \\
&\geq 4^{-\tau} (2(1 - \ln 2) - 2 + \ln 4) = 4^{-\tau} (2 - 2 \ln 2 - 2 + \ln 4) = 0.
\end{aligned}$$

This means that

$$0 \leq A(\tau) - B(\tau) \leq A(1) - B(1).$$

In proposition 3.58 we saw that:

$$\begin{aligned}
d \int_0^1 g(s) ds &\leq 4 \cdot 2^{2/d} \left[2^{1/d} \frac{3}{4 \ln 4} - \ln 2 \left[\frac{3 - \ln 4}{4 \ln^2 4} \right] \right] \\
&\Rightarrow A(1) \leq 2 \cdot 2^{2/d} \left[2^{1/d} \frac{3}{4 \ln 4} - \ln 2 \left[\frac{3 - \ln 4}{4 \ln^2 4} \right] \right] \\
[3.62, d \geq 2, \ln 4 = 2 \ln 2] &\leq 2 \left(1 + \frac{2}{d} \right) \left[\frac{3(1 + \frac{1}{d})}{8 \ln 2} - \ln 2 \left[\frac{3 - \ln 4}{16 \ln^2 2} \right] \right] \\
&= 2 \left(1 + \frac{2}{d} \right) \left[\frac{6 + \frac{6}{d}}{16 \ln 2} - \frac{3 - \ln 4}{16 \ln 2} \right] = \left(1 + \frac{2}{d} \right) \left[\frac{3 + 2 \ln 2 + \frac{6}{d}}{8 \ln 2} \right].
\end{aligned}$$

Subtracting $B(1)$ gives:

$$\begin{aligned}
A(1) - B(1) &\leq \left(1 + \frac{2}{d} \right) \left[\frac{3 + 2 \ln 2 + \frac{6}{d}}{8 \ln 2} \right] - \left(\frac{1}{\ln 4} - \frac{1}{4} \left(\frac{1}{\ln 4} - 1 \right) \right) \\
&= \left(1 + \frac{2}{d} \right) \left[\frac{3 + 2 \ln 2 + \frac{6}{d}}{8 \ln 2} \right] - \frac{3 + 2 \ln 2}{8 \ln 2} \\
&= \frac{6}{d \cdot 8 \ln 2} + \frac{2}{d} \left[\frac{3 + 2 \ln 2 + \frac{6}{d}}{8 \ln 2} \right] \\
[d \geq 100,000] &\leq \frac{2.6641}{d},
\end{aligned}$$

leading to

$$0 \leq A(\tau) - B(\tau) \leq \frac{2.6641}{d}.$$

Also,

$$\begin{aligned} \tilde{x}(\tau) - \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} &= \sqrt{1 - B(\tau)} - \sqrt{1 - A(\tau)} = \frac{1 - B(\tau) - (1 - A(\tau))}{\sqrt{1 - B(\tau)} + \sqrt{1 - A(\tau)}} \\ &= \frac{A(\tau) - B(\tau)}{\sqrt{1 - A(\tau)} + \sqrt{1 - B(\tau)}}. \end{aligned}$$

Since $0 \leq A(\tau) - B(\tau)$, $\tilde{x}(\tau) \geq \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds}$. Lower bounding the denominator:

$$\begin{aligned} \sqrt{1 - A(\tau)} + \sqrt{1 - B(\tau)} &\geq \sqrt{1 - B(\tau)} = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\tau} \left(\frac{1}{\ln 4} - \tau \right)} \\ &\geq \sqrt{1 - \frac{1}{\ln 4} + 4^{-1} \left(\frac{1}{\ln 4} - 1 \right)} \geq 0.4571, \end{aligned}$$

and so,

$$0 \leq \tilde{x}(\tau) - \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} \leq \frac{2.6641}{0.4571d} \leq \frac{5.8283}{d}.$$

□

Proposition 3.64. For $d \geq 100,000$ and all $\tau \in [0, \zeta]$, $|\tilde{x}(\tau) - x(\tau)| \leq \frac{38.9822}{d}$.

Proof. From proposition 3.63 and proposition 3.61:

$$\begin{aligned} |\tilde{x}(\tau) - x(\tau)| &= \left| \tilde{x}(\tau) - \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} + \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} - x(\tau) \right| \\ &\leq \left| \tilde{x}(\tau) - \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} \right| + \left| \sqrt{1 - \frac{d}{2} \int_0^\tau g(s) ds} - x(\tau) \right| \\ &\leq \frac{5.8283}{d} + \frac{33.1539}{d} = \frac{38.9822}{d}. \end{aligned}$$

□

Corollary 3.65. For $d \geq 100,000$ and all $\tau \in [0, \zeta]$, $x(\tau) \geq 0.4567$.

Proof. The function $\tilde{x}(\tau) = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\tau} \left(\frac{1}{\ln 4} - \tau \right)}$ is decreasing for $\tau \in [0, 1]$:

$$\tilde{x}'(\tau) = \frac{-4^{-\tau} (2 - \tau \ln 4)}{2\sqrt{1 - \frac{1}{\ln 4} + 4^{-\tau} \left(\frac{1}{\ln 4} - \tau \right)}} \leq 0.$$

It is lowest at $\tau = 1$. Combining this with $|x(\tau) - \tilde{x}(\tau)| \leq \frac{38.9822}{d}$ gives:

$$\begin{aligned} x &\geq \sqrt{1 - \frac{1}{\ln 4} + \frac{1}{4} \left(\frac{1}{\ln 4} - 1 \right)} - \frac{38.9822}{d} \geq 0.4571 - \frac{38.9822}{d} \\ [d \geq 100,000] &\geq 0.4571 - \frac{38.9822}{100,000} \geq 0.4567. \end{aligned}$$

□

Claim 3.66. For $a \geq b \geq 0$, $\sqrt{a-b} \geq \sqrt{a} - \sqrt{b}$.

Proof.

$$\sqrt{a-b} = \sqrt{(\sqrt{a} - \sqrt{b})(\sqrt{a} + \sqrt{b})} \geq \sqrt{(\sqrt{a} - \sqrt{b})^2} = \sqrt{a} - \sqrt{b}.$$

□

Proposition 3.67 (Existence of the solution to the ODE). For $d \geq 100,000$ and all $\tau \in [0, 1]$, $x^2(\tau) \geq 4\beta \left(\tau + \frac{1}{d} \right)$.

Proof. Note the following for $\tau = \zeta$:

$$x^2(\zeta) \geq 0.4567^2 > 0.2 > 4\frac{2}{d} > 4\beta \left(\zeta + \frac{1}{d} \right),$$

for $d \geq 100,000$, from claim 3.53 and corollary 3.65. From continuity and the strict inequality, there exists $\delta > 0$ such that $\forall \tau \in [\zeta, \zeta + \delta]$, $x^2(\tau) \geq 4\beta \left(\tau + \frac{1}{d} \right)$. From the definition of the

ODE in eq. (3.14), for all $\tau \in [\zeta, \zeta + \delta]$:

$$\begin{aligned}
x'(\tau) &= d \frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2} \\
[3.66] \geq d \frac{x - 2\sqrt{\beta(\tau + \frac{1}{d})} - x}{2} &= -d\sqrt{\beta(\tau + \frac{1}{d})} \geq -d\sqrt{\frac{2}{d}} = -\sqrt{2d} \\
\Rightarrow x(\tau) &\geq x(\zeta) - \sqrt{2d}(\tau - \zeta) \geq 0.4567 - \sqrt{2d}(\tau - \zeta).
\end{aligned}$$

We can show that $\delta \geq \frac{0.3}{\sqrt{d}}$, by showing the solution must exist at $\tau = \zeta + \frac{0.3}{\sqrt{d}}$:

$$\begin{aligned}
x(\zeta + \tilde{\delta}) &\geq 0.4567 - \sqrt{2d}\tilde{\delta} \geq 2\sqrt{\frac{2}{d}} > \sqrt{4\beta(\zeta + \tilde{\delta} + \frac{1}{d})} \\
\Leftrightarrow \tilde{\delta} &\leq \frac{1}{\sqrt{2d}} \left(0.4567 - 2\sqrt{\frac{2}{d}} \right) \quad \Leftarrow \quad \tilde{\delta} = \frac{0.3}{\sqrt{d}}.
\end{aligned}$$

Hence, $x^2(\tau) \geq 4\beta(\tau + \frac{1}{d})$ for $\tau \in [0, \zeta + \frac{0.3}{\sqrt{d}}]$, and thus we can apply all previous claims replacing ζ with $\zeta + \frac{0.3}{\sqrt{d}}$, and specifically,

$$x^2(\zeta + \frac{0.3}{\sqrt{d}}) \geq 0.4567^2 > 0.2 > 4\frac{2}{d} > 4\beta\left(\zeta + \frac{0.3}{\sqrt{d}} + \frac{1}{d}\right),$$

This allows repeating all the previous steps without alteration. After repeating $\lceil \frac{\sqrt{d}}{0.3} \rceil$ times, $x^2(\tau) \geq 4\beta(\tau + \frac{1}{d})$ for $\tau \in [0, 1]$, concluding the proof. $\square$

Corollary 3.68. All previous propositions, and specifically proposition 3.64 and corollary 3.65, apply $\forall \tau \in [0, 1]$ (indicating $\zeta = 1$).

Proposition 3.69. For $d \geq 100,000$, $L \triangleq \max_{x, \tau \in [0, 1]} \left| \frac{d}{dx} f(\tau, x) \right| \leq 4.7955$.

Proof. L , the Lipschitz constant of f , is given by

$$L \triangleq \max_{x, \tau \in [0, 1]} \left| \frac{d}{dx} f(\tau, x) \right|.$$

$$\frac{d}{dx}f(\tau, x) = \frac{d}{dx}\left[d\frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2}\right] = \frac{d}{2}\left(\frac{x}{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})}} - 1\right).$$

Assume that $x \geq x_{\min}$, $\tau \in [0, 1]$. from $d \geq 3$, $\tau + \frac{1}{d} \leq \frac{1}{\ln 2}$. From claim 3.52, $\beta(\tau + \frac{1}{d}) \geq 0$, so $\frac{d}{dx}f(\tau, x) \geq 0$. So

$$L = \max_{x, \tau \in [0, 1]} \frac{d}{dx}f(\tau, x).$$

For any fixed x , the maximum $\beta(\tau + \frac{1}{d})$ will maximize L . By claim 3.52, β is decreasing with τ , so to maximize L , $\tau = 0$. To maximize $\frac{d}{2}\left(\frac{x}{\sqrt{x^2 - 4\beta(\frac{1}{d})}} - 1\right)$, this function is decreasing with respect to x :

$$\begin{aligned} \frac{d}{dx}\left[\frac{d}{2}\left(\frac{x}{\sqrt{x^2 - 4\beta(\frac{1}{d})}} - 1\right)\right] &= \frac{d}{2}\left(\frac{\sqrt{x^2 - 4\beta(\frac{1}{d})} - x\frac{x}{\sqrt{x^2 - 4\beta(\frac{1}{d})}}}{x^2 - 4\beta(\frac{1}{d})}\right) \\ &= \frac{d}{2}\left(\frac{x^2 - 4\beta(\frac{1}{d}) - x^2}{(x^2 - 4\beta(\frac{1}{d}))^{\frac{3}{2}}}\right) = \frac{d}{2}\left(\frac{-4\beta(\frac{1}{d})}{(x^2 - 4\beta(\frac{1}{d}))^{\frac{3}{2}}}\right) \leq 0, \end{aligned}$$

hence the optimal x is $x_{\min}$. Thus

$$L = \frac{d}{2}\left(\frac{x_{\min}}{\sqrt{x_{\min}^2 - 4\beta(\frac{1}{d})}} - 1\right).$$

Now,

$$4\beta\left(\frac{1}{d}\right) = 4\frac{((1-1)2^{-1/d} - (1-2))2^{(5-2)/d}}{d} = 4\frac{2^{3/d}}{d},$$

and applying corollary 3.65 gives:

$$\begin{aligned}
L &\leq \frac{d}{2} \left(\frac{0.4567}{\sqrt{0.4567^2 - 4^{\frac{23}{d}}}} - 1 \right) \\
[d \geq 100,000] &\leq \frac{d}{2} \left(\frac{0.4567}{\sqrt{0.4567^2 - 4^{\frac{23}{100000}}}} - 1 \right) \leq \frac{d}{2} \left(\frac{0.4567}{\sqrt{0.4567^2 - \frac{4.0001}{d}}} - 1 \right) \\
&\leq \frac{d}{2} \left(\frac{1}{\sqrt{1 - \frac{19.1783}{d}}} - 1 \right) = \frac{d}{2} \left(\frac{1}{\sqrt{1 - \frac{19.1783}{d}}} - 1 \right) \frac{\frac{1}{\sqrt{1 - \frac{19.1783}{d}}} + 1}{\frac{1}{\sqrt{1 - \frac{19.1783}{d}}} + 1} \\
&= \frac{d}{2} \left(\frac{\frac{1}{1 - \frac{19.1783}{d}} - 1}{\frac{1}{\sqrt{1 - \frac{19.1783}{d}}} + 1} \right) \leq \frac{d}{2} \left(\frac{\frac{19.1783}{1 - \frac{19.1783}{d}}}{1 + 1} \right) = \frac{19.1783}{4} \frac{1}{1 - \frac{19.1783}{d}} \\
&\leq \frac{19.1783}{4} \frac{1}{1 - \frac{19.1783}{100000}} \leq 4.7955.
\end{aligned}$$

□

Proposition 3.70. For $d \geq 100,000$, $M \triangleq \max_{\tau \in [0,1]} \left| \frac{d^2}{d\tau^2} x(\tau) \right| \leq 10.5027$

Proof. M is defined as an upper bound on the second derivative (absolute value) of $x(\tau)$ in the relevant interval:

$$M \triangleq \max_{\tau \in [0,1]} \left| \frac{d^2}{d\tau^2} x(\tau) \right| = \max_{\tau \in [0,1]} \left| \frac{d}{d\tau} f(\tau, x(\tau)) \right|.$$

$$\begin{aligned}
\frac{d}{d\tau} f(\tau, x(\tau)) &= \frac{\partial}{\partial \tau} \left[d \frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2} \right] + x'(\tau) \frac{\partial}{\partial x} \left[d \frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2} \right] \\
&= \frac{\partial}{\partial \tau} \left[d \frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2} \right] + f(\tau, x(\tau)) \frac{\partial}{\partial x} \left[d \frac{\sqrt{x^2 - 4\beta(\tau + \frac{1}{d})} - x}{2} \right].
\end{aligned}$$

For the first term:

$$\begin{aligned}\frac{\partial}{\partial \tau} \left[d \frac{\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)} - x}{2} \right] &= \frac{d}{2} \left(\frac{-4}{2\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)}} \right) \frac{\partial}{\partial \tau} \beta \left(\tau + \frac{1}{d} \right) \\ &= -\frac{d}{\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)}} \frac{\partial}{\partial \tau} \beta \left(\tau + \frac{1}{d} \right).\end{aligned}$$

By claim 3.52, β is positive and decreasing, so $\frac{d}{\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)}}$ is maximized at $\tau = 0$; In addition $\frac{\partial}{\partial \tau} \beta \left(\tau + \frac{1}{d} \right) \leq 0$, and thus the entire expression is nonnegative. Also, β is convex, so the absolute value of the negative $\frac{\partial}{\partial \tau} \beta \left(\tau + \frac{1}{d} \right)$ is also maximized at $\tau = 0$. All in all, the entire expression is maximized at $\tau = 0$, and is bounded by:

$$\begin{aligned}0 &\leq -\frac{d}{\sqrt{x^2 - 4\beta \left(\frac{1}{d} \right)}} \frac{\partial}{\partial \tau} \beta \left(\tau + \frac{1}{d} \right) \Big|_{\tau=0} \\ &= -\frac{d}{\sqrt{x^2 - 4\beta \left(\frac{1}{d} \right)}} \frac{1}{d} \left(d \left(2^{-1/d} - 1 \right) 2^{\frac{5-2d(1/d)}{d}} \right. \\ &\quad \left. - 2 \ln 2 \cdot 2^{\frac{5-2d(1/d)}{d}} \left((d(1/d) - 1) 2^{-1/d} - (d(1/d) - 2) \right) \right) \\ &= \frac{2^{3/d}}{\sqrt{x^2 - 4\beta \left(\frac{1}{d} \right)}} (2 \ln 2 + d(1 - 2^{-1/d})) \\ [3.51] &\leq \frac{2^{3/d}}{\sqrt{x^2 - 4 \frac{2^{3/d}}{d}}} (2 \ln 2 + \ln 2) \\ &\leq \frac{2^{3/100000}}{\sqrt{x^2 - 4 \frac{2^{3/100000}}{100000}}} (3 \ln 2) \\ &\leq \frac{2.08}{\sqrt{x^2 - 4.1 \cdot 10^{-5}}} \\ [3.65] &\leq \frac{2.08}{\sqrt{0.4567^2 - 4.1 \cdot 10^{-5}}} \leq 4.555.\end{aligned}$$

For the second term, by proposition 3.69,

$$0 \leq \frac{\partial}{\partial x} \left[d \frac{\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)} - x}{2} \right] \leq L = 4.7955.$$

Now we need to bound $f(\tau, x) = d \frac{\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)} - x}{2}$, which is always negative. By claim 3.52, β is positive and decreasing, so its maximum, which minimizes this and thus maximizes its absolute value, is received at $\tau = 0$. We also know that $f(0, x)$ increases with x (see the beginning of the proof for proposition 3.69), so its absolute value decreases with x . Thus:

$$\begin{aligned} 0 &\geq d \frac{\sqrt{x^2 - 4\beta \left(\tau + \frac{1}{d} \right)} - x}{2} \geq d \frac{\sqrt{x^2 - 4\beta \left(\frac{1}{d} \right)} - x}{2} \\ [d \geq 100,000 \Rightarrow x \geq 0.4567] &\geq d \frac{\sqrt{0.4567^2 - 4\frac{2^{3/d}}{d}} - 0.4567}{2} \\ &\geq d \frac{\sqrt{0.4567^2 - 4\frac{2^{3/100,000}}{d}} - 0.4567}{2} \\ &\geq \frac{0.4567}{2} \frac{\sqrt{1 - \frac{19.1782}{d}} - 1}{\frac{1}{d}} \\ &= -\frac{0.4567}{2} \cdot 19.1782 \frac{1 - \sqrt{1 - \frac{19.1782}{d}}}{\frac{19.1782}{d}} \\ [3.59] &\geq -\frac{0.4567}{2} \cdot 19.1782 \left(\frac{1}{2} + \frac{19.1782}{2d} \right) \\ [d \geq 100,000] &\geq -\frac{0.4567}{2} \cdot 19.1782 \left(\frac{1}{2} + \frac{19.1782}{2 \cdot 100000} \right) \geq -2.1901. \end{aligned}$$

To summarize, $-2.1901 \leq f(\tau, x) \leq 0$, and thus

$$\frac{d}{d\tau} f(\tau, x(\tau)) \in [0, 4.555] + [-2.1901, 0] \cdot [0, 4.7955] \subseteq [-10.5027, 4.555],$$

and in absolute value,

$$M \leq 10.5027.$$

□

Proposition 3.71. For $d \geq 100,000$, $k \in \{2, \dots, d\}$ for which the sequence $(x)_k$ exists, i.e., $x_{j-1}^2 - 4\beta\left(\frac{j}{d}\right) \geq 0$ for all $2 \leq j \leq k$,

$$|x_k - \tilde{x}_k| \leq \frac{170.4}{d},$$

where $\tilde{x}_k = \sqrt{1 - \frac{1}{\ln 4} + 4^{-\frac{k}{d}} \left(\frac{1}{\ln 4} - \frac{k}{d}\right)}$.

Proof. As noted in section 3.16.1, the sequence $(x)_k$ are Euler's iterates for the ODE in eq. (3.14). Using the global truncation error of Euler's method [10, chapter 6.2]:

$$\left| x_k - x\left(\frac{k}{d}\right) \right| \leq \frac{hM}{2L} \left(\exp\left(L\left(\frac{k}{d} - 0\right)\right) - 1 \right) \leq \frac{M}{2Ld} (e^L - 1),$$

where

$$L = \max_{x, \tau \in [0, 1]} \left| \frac{d}{dx} f(\tau, x) \right| \quad (\text{where } \tau \text{ is treated as a constant}),$$

$$M = \max_{\tau \in [0, 1]} \left| \frac{d^2}{d\tau^2} x(\tau) \right| = \left| \frac{d}{d\tau} f(\tau, x(\tau)) \right|.$$

For $d \geq 100,000$, $L \leq 4.7955$ by proposition 3.69, and proposition 3.70 gives $M \leq 10.5027$.

So in total

$$\left| x_k - x\left(\frac{k}{d}\right) \right| \leq \frac{10.5027}{2 \cdot 4.7955d} \cdot (e^{4.7955} - 1) \leq \frac{131.3685}{d}.$$

Combining this with proposition 3.64 gives

$$\begin{aligned}
|x_k - \tilde{x}_k| &= \left| x_k - \tilde{x} \left(\frac{k}{d} \right) \right| = \left| x_k - x \left(\frac{k}{d} \right) + x \left(\frac{k}{d} \right) - \tilde{x} \left(\frac{k}{d} \right) \right| \\
&\leq \left| x_k - x \left(\frac{k}{d} \right) \right| + \left| x \left(\frac{k}{d} \right) - \tilde{x} \left(\frac{k}{d} \right) \right| \\
&\leq \frac{131.3685}{d} + \frac{38.9822}{d} \leq \frac{170.4}{d}.
\end{aligned}$$

□

Corollary 3.72. For $d \geq 100,000$, $k \in \{2, \dots, d\}$ for which the sequence $(x)_k$ exists, i.e., $x_{j-1}^2 - 4\beta \left(\frac{j}{d} \right) \geq 0$ for all $2 \leq j \leq k$: $x_k \geq 0.45$.

Proof. x_k is decreasing (claim 3.29), so $\forall k \in [d]$:

$$x_k \geq x_d \geq \tilde{x}_d - \frac{170.4}{d} = \sqrt{1 - \frac{1}{\ln 4} + 4^{-1} \left(\frac{1}{\ln 4} - 1 \right)} - \frac{170.4}{d}$$

$$[d \geq 100,000] \geq 0.45.$$

□

Proposition 3.73 (Existence of the sequence). For $d \geq 100,000$, the sequence $(x)_k$ exists for all $k \in [d]$, i.e., $x_{j-1}^2 - 4\beta_j \geq 0$, for all $2 \leq j \leq d$.

The proof is similar to that of proposition 3.67 but simpler. First,

$$x_1^2 = 1 > 4\frac{2}{d} > 4\beta \left(\frac{2}{d} \right) = 4\beta_2,$$

which means x_2 exists. Now assuming for some $k \in \{2, \dots, d\}$, x_k exists, then from corollary 3.72, $x_k \geq 0.45$, and we have

$$x_k^2 \geq 0.45^2 > 0.2 > 4\frac{2}{d} > 4\beta \left(\frac{k+1}{d} \right) = 4\beta_{k+1},$$

which means that x_{k+1} exists, and by induction the proof is done.

Chapter 4

Conclusion

We have seen in this dissertation several ways to get upper and lower bounds on average and max loss of the final iterate on variants of the procedure described in the introduction.

In the cyclic case, we showed that if one iteratively projects a unit vector $w_0 \in \mathbf{R}^d$ onto arbitrary subspaces $S_1, \dots, S_T \subseteq \mathbf{R}^d$ in that order, then does that same sequence of projections repeatedly for a total of K cycles, then no matter what $d, S_1, \dots, S_T$ are, the average squared distance from the final vector to each of $S_1, \dots, S_T$ is $O(T^2/K)$, where the implied constant may be taken to be $0.4 + o_{T, K \rightarrow \infty}$.

We also showed that the numerical range of the product of T complex projections is always contained in the sinusoidal spiral $\Omega_n := \{re^{i\theta} : \cos(\pi/k)r^{-1/k} \geq \cos(\frac{\theta-\pi}{k})\} \subset \mathbf{C}$, and that this may be realized by the product of T real projections onto $2D$ subspaces in $\mathbf{R}^4$.

In the greedy case, we gave a proof that if one iteratively projects a unit vector $w_0 \in \mathbf{R}^d$ onto greedily selected subspaces from an indexed family of subspaces of $\mathbf{R}^d$ for a total of I iterations, then the final vector has a squared distance of at most $O(1/\sqrt[3]{I})$ from each of the subspaces in the family.

However, if one does the greedy selection without replacement (i.e., one selects an index from the indexed family of subspaces that is maximally far from the current iterate among those that have not been used) then we give constructions with arbitrarily many iterations

for which the average squared distance to the subspaces in the family is bounded away from 0: One in $3D$ and one achieving larger final average square distance with unbounded dimension.

Also, experiments using Gaussian and CIFAR data suggested some trends which we leave as open problems to give heuristic explanations for.

To conclude, we shift perspective to studying this as a problem in pure linear algebra and suggest some problems that arose naturally during the study of this problem.

1. Is the upper bound obtained in the cyclic chapter asymptotically tight? For example, calculate

$$2\pi^4 \limsup_{T \rightarrow \infty} \limsup_{K \rightarrow \infty} \frac{K}{T^2} \sup_{\substack{d \in \mathbf{Z}_{\geq 0} \\ S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ w_0 \in \mathbf{R}^d, \|w_0\|=1}} \text{avg}(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I))$$

where $I = TK$ and w_I is defined via the cyclic ordering.

One can ask a similar question for max loss.

2. Our bound from [119] does not asymptotically¹ beat the bound in [41] if all subspaces have codimension 1 (if one restricts the ambient space to the span of the initial guess and the normals of the subspaces). Can one improve the bound asymptotically if one adds the assumption that all subspaces have codimension 1?
3. The following conjecture arose when experimenting in the cyclic case:

Conjecture 4.1. For each $n \in \mathbf{Z}_{\geq 3}$, define the sinusoidal spiral

$$\Omega_n := \{re^{i\theta} : \cos(\pi/n)r^{-1/n} \geq \cos(\frac{\theta - \pi}{n})\} \subset \mathbf{C}.$$

¹i.e., by more than a constant factor

Let X be a Hilbert space and $A : X \rightarrow X$ a linear transformation whose numerical range $W(A)$ satisfies

$$W(A) \subseteq \Omega_n.$$

Then for any $k \in \mathbf{Z}_{\geq 0}$,

$$W(A^k) \subseteq \Omega_n.$$

4. Our setup, as described in the introduction, does not include every conceivable way of aggregating the loss. For example, one could imagine taking the max/average loss over only those subspaces that were selected (and, in the case of average loss, possibly weighted by how many times each subspace was selected), which would make sense if interpreting aggregate loss as some measurement of “forgetting”. For the cyclic selection rule, as long as $I \geq T$ all subspaces are visited, and as $I/T \rightarrow \infty$ (or if one assumes I is a multiple of T) each subspace becomes used an essentially equal number of times. For the greedy selection rule, this does not apply.

Are the best bounds for these variants the same as those described in the introduction?

One can interpret an upper bound for the max version as “if a greedy selection rule happens to visit a subspace, it will be at most [bound] away from it at the end” and a similar but more complicated interpretation for the average version.

5. As shown in [133], there exists an optimizer for the greedy (with replacement) selection rule where each iterate is equidistant from all subspaces that will be used in a future step, as well as the subspace that will be used to measure the final loss (but not any of the subspaces previously visited). Can one show geometrically that this will occur for an optimizer?
6. What algebraic structure can be placed on the space of all greedy (with or without replacement) trajectories (not necessarily starting at a vector of norm 1)?

7. An easy calculation shows that, for a greedy (with or without replacement, possibly with affine subspaces with no common intersection) trajectory, if one of the steps $\|w_i - w_{i-1}\|$ is at least twice as long as an earlier step $\|w_{i'} - w_{i'-1}\|$ with $i' < i$, then w_i is at least as close to $w_{i'-1}$ as w_{i-1} is. In other words, large late steps must get closer to points involved in early short steps.

(a) This connects the greedy case to the following Ramsey-theoretic questions:

- In terms of the dimension d , what is the length of the longest sequence of points $x_1, \dots, x_N \in \mathbf{R}^d$ such that each point x_i is closer to all of $x_1, \dots, x_{i-2}$ than x_{i-1} ?
- In terms of the dimension d , what is the length of the longest sequence of points $x_1, \dots, x_N \in \mathbf{R}^d$ such that, if $d_{\min}$ denotes the minimum distance between any two points in the sequence, then for all $1 \leq i < j < k \leq N$,

$$\|x_i - x_k\|^2 - \|x_i - x_j\|^2 < d_{\min}^2?$$

- In terms of the dimension d , what is the length of the longest sequence of points $x_1, \dots, x_N \in \mathbf{R}^d$ such that, if $d_{\min}$ denotes the minimum distance between any two points in the sequence, then for all $1 \leq i < j < k \leq N$,

$$\|x_i - x_k\|^2 - \|x_i - x_j\|^2 < \|x_j - x_k\|^2?$$

- In terms of the dimension d , what is the length of the longest sequence of points $x_1, \dots, x_N \in \mathbf{R}^d$ such that all (not necessarily consecutive) subsequences of length 3 form acute angles (and the remaining angles may or may not be acute)?

(b) For each fixed dimension d , calculate

$$\limsup_{I \rightarrow \infty} \sup_{\substack{T \in \mathbf{Z}_{\geq 1} \\ S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ w_0 \in \mathbf{R}^d, \|w_0\|=1}} \max(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I))$$

and

$$\limsup_{I \rightarrow \infty} \sup_{\substack{d', T \in \mathbf{Z}_{\geq 1} \\ S_1, \dots, S_T \subseteq \mathbf{R}^{d'} \text{ subspaces} \\ \dim(S_1)=d \\ w_0 \in \mathbf{R}^{d'}, \|w_0\|=1}} \max(L_{S_1}(w_I), L_{S_2}(w_I), \dots, L_{S_T}(w_I))$$

where w_I is defined via the greedy (with replacement) ordering. In words, the best bound given the ambient dimension and the best bound given the lowest dimension of one of the subspaces. I find the latter natural, as having even one subspace of low dimension forces, by greediness, all iterates to almost lie on that subspace, making it act almost as if the ambient space is low-dimensional.

8. (a) The geometric interpretation of Corollary 3.15 is that we selected a “basepoint” at w_0 and project it to all line segments between consecutive steps and bound based on the distances from these projections to the associated endpoints. Can one generalize to a better basepoint to recover the bound in [133]?
- (b) Many good vector equations have bilinear generalizations. What is the bilinear version of the geometric interpretation of Corollary 3.15?
- (c) What is the relationship between our upper bound proof and the method in [133]?
9. As mentioned previously, one reason to study greedy orderings is as a proxy for other orderings that approximate a greedy selection rule. It therefore makes sense to try to generalize our results to orderings that are not quite greedy to serve as a more robust proxy. Which methods in this dissertation generalize if, say, we only require steps to

go to a subspace that is at least C times as far away as the farthest, for some fixed constant $0 \leq C \leq 1$? What if we only require a positive fraction of the steps be greedy steps? Does any algebraic structure mentioned in an earlier question carry over?

10. **Conjecture 4.2.** For any $d \in \mathbf{Z}_{\geq 0}$,

$$\begin{aligned} & \limsup_{T \rightarrow \infty} T \sup_{\substack{S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ w_0 \in \mathbf{R}^d, \|w_0\|=1}} \text{avg}(L_{S_1}(w_T), L_{S_2}(w_T), \dots, L_{S_T}(w_T)) \\ &= \limsup_{T \rightarrow \infty} T \sup_{\substack{S_1, \dots, S_T \subseteq \mathbf{R}^d \text{ subspaces} \\ \dim(S_1) = \dots = \dim(S_T) = d-1 \\ w_0 \in \mathbf{R}^d, \|w_0\|=1}} \text{avg}(L_{S_1}(w_T), L_{S_2}(w_T), \dots, L_{S_T}(w_T)) \end{aligned}$$

where w_T is defined via the greedy-without-replacement ordering. In words, in any fixed dimension and using the greedy-without-replacement selection rule, the asymptotic worst-case average loss can be obtained using only codimension 1 subspaces.

References

- [1] Alessandro Achille, Michael Lam, Rahul Tewari, Avinash Ravichandran, Subhansu Maji, Charles C Fowlkes, Stefano Soatto, and Pietro Perona. Task2vec: Task embedding for meta-learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6430–6439, 2019.
- [2] Shmuel Agmon. The relaxation method for linear inequalities. *Canadian Journal of Mathematics*, 6:382–392, 1954.
- [3] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *ICLR 2017 Workshop Track*, 2017.
- [4] Guillaume Alain, Alex Lamb, Chinnadhurai Sankar, Aaron Courville, and Yoshua Bengio. Variance reduction in sgd by distributed importance sampling. *arXiv preprint arXiv:1511.06481*, 2015.
- [5] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 139–154, 2018.
- [6] Verónica Alvarez, Santiago Mazuelas, and Jose A Lozano. Supervised learning with evolving tasks and performance guarantees. *Journal of Machine Learning Research*, 26(17):1–59, 2025.
- [7] Yossi Arjevani and Ohad Shamir. Dimension-free iteration complexity of finite sum optimization problems. *Advances in Neural Information Processing Systems*, 29, 2016.
- [8] Haruka Asanuma, Shiro Takagi, Yoshihiro Nagano, Yuki Yoshida, Yasuhiko Igarashi, and Masato Okada. Statistical mechanical analysis of catastrophic forgetting in continual learning with teacher and student networks. *Journal of the Physical Society of Japan*, 90(10):104001, Oct 2021.
- [9] Neil Ashtekar, Jingxi Zhu, and Vasant G Honavar. Class incremental learning from first principles: A review. *Transactions on Machine Learning Research*, 2025.
- [10] K.E. Atkinson. *An Introduction to Numerical Analysis, 2nd Ed.* Wiley India Pvt.

Limited, 2008. ISBN 9788126518500. URL <https://books.google.com/books?id=1PV8Fv2XEosC>.

- [11] Amit Attia, Matan Schliserman, Uri Sherman, and Tomer Koren. Fast last-iterate convergence of sgd in the smooth interpolation regime. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [12] Catalin Badea and David Seifert. Ritt operators and convergence in the method of alternating projections. *Journal of Approximation Theory*, 205:133–148, 2016.
- [13] Zhong-Zhi Bai and Wen-Ting Wu. On greedy randomized kaczmarz method for solving large sparse linear systems. *SIAM Journal on Scientific Computing*, 40(1):A592–A606, 2018.
- [14] Zhong-Zhi Bai and Wen-Ting Wu. On relaxed greedy randomized kaczmarz methods for solving large sparse linear systems. *Applied Mathematics Letters*, 83:21–26, 2018.
- [15] Zhong-Zhi Bai and Wen-Ting Wu. On greedy randomized augmented kaczmarz method for solving large sparse inconsistent linear systems. *SIAM Journal on Scientific Computing*, 43(6):A3892–A3911, 2021.
- [16] Heinz H Bauschke and Jonathan M Borwein. On projection algorithms for solving convex feasibility problems. *SIAM review*, 38(3):367–426, 1996.
- [17] Samuel J Bell and Neil D Lawrence. The effect of task ordering in continual learning. *arXiv preprint arXiv:2205.13323*, 2022.
- [18] Pierfrancesco Beneventano. On the trajectories of sgd without replacement. *arXiv preprint arXiv:2312.16143*, 2023.
- [19] Petr A. Borodin and Eva Kopecká. Alternating projections, remotest projections, and greedy approximation. *Journal of Approximation Theory*, 260:105486, 2020. ISSN 0021-9045.
- [20] Léon Bottou. Curiously fast convergence of some stochastic gradient descent algorithms. Unpublished open problem offered to the attendance of the SLDS 2009 conference, 2009. URL <http://leon.bottou.org/papers/bottou-slds-open-problem-2009>.
- [21] Wenbin Cai, Ya Zhang, and Jun Zhou. Maximizing expected model change for active learning in regression. In *2013 IEEE 13th international conference on data mining*, pages 51–60. IEEE, 2013.
- [22] Xufeng Cai and Jelena Diakonikolas. Last iterate convergence of incremental methods

- and applications in continual learning. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [23] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision (ECCV)*, pages 532–547, 2018.
 - [24] Arslan Chaudhry, Marc’Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-GEM. In *International Conference on Learning Representations*, 2019.
 - [25] Yaofo Chen. Pytorch cifar models. <https://github.com/chenyaofo/pytorch-cifar-models>, 2021.
 - [26] Zhiyuan Chen and Bing Liu. Lifelong machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 12(3):1–207, 2018.
 - [27] Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, pages 2933–2943. Curran Associates, Inc., 2019. URL https://papers.neurips.cc/paper_files/paper/2019/hash/ae614c557843b1df326cb29c57225459-Abstract.html.
 - [28] Michel Crouzeix. A functional calculus based on the numerical range: applications. *Linear and Multilinear Algebra*, 56(1-2):81–103, 2008.
 - [29] Michel Crouzeix and César Palencia. The numerical range is a $(1 + \sqrt{2})$ -spectral set. *SIAM Journal on Matrix Analysis and Applications*, 38(2):649–655, 2017. doi: 10.1137/17M1116672.
 - [30] Rudrajit Das, Xi Chen, Bertram Ieong, Parikshit Bansal, and sujay sanghavi. Understanding the training speedup from sampling with approximate losses. In *Forty-first International Conference on Machine Learning*, 2024.
 - [31] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2022. doi: 10.1109/TPAMI.2021.3057446.
 - [32] Jesus A De Loera, Jamie Haddock, and Deanna Needell. A sampling kaczmarz–motzkin algorithm for linear feasibility. *SIAM Journal on Scientific Computing*, 39(5):S66–S87, 2017.

- [33] Christopher De Sa. Random reshuffling is not always better. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5957–5967. Curran Associates, Inc., 2020. URL <https://proceedings.neurips.cc/paper/2020/hash/42299f06ee419aa5d9d07798b56779e2-Abstract.html>.
- [34] Ronald A DeVore and Vladimir N Temlyakov. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.
- [35] Meng Ding, Kaiyi Ji, Di Wang, and Jinhui Xu. Understanding forgetting in continual learning with linear regression. In *Forty-first International Conference on Machine Learning*, 2024.
- [36] Thang Doan, Mehdi Abbana Bennani, Bogdan Mazouze, Guillaume Rabusseau, and Pierre Alquier. A theoretical analysis of catastrophic forgetting through the NTK overlap matrix. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1072–1080. PMLR, 13–15 Apr 2021. URL <https://proceedings.mlr.press/v130/doan21a.html>.
- [37] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655. PMLR, 2014.
- [38] David L Donoho, Michael Elad, and Vladimir N Temlyakov. On lebesgue-type inequalities for greedy approximation. *Journal of Approximation Theory*, 147(2):185–195, 2007.
- [39] Yonina C Eldar and Deanna Needell. Acceleration of randomized kaczmarz method via the johnson–lindenstrauss lemma. *Numerical Algorithms*, 58:163–177, 2011.
- [40] Tommy Elfving. Block-iterative methods for consistent and inconsistent linear equations. *Numerische Mathematik*, 35(1):1–12, 1980.
- [41] Itay Evron, Edward Moroshko, Rachel Ward, Nathan Srebro, and Daniel Soudry. How catastrophic can catastrophic forgetting be in linear regression? In *Conference on Learning Theory*, pages 4028–4079. PMLR, 2022.
- [42] Itay Evron, Edward Moroshko, Gon Buzaglo, Maroun Khriesh, Badea Marjieh, Nathan Srebro, and Daniel Soudry. Continual learning in linear classification on separable data. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 9440–9484. PMLR, 23–29 Jul 2023.

- [43] Itay Evron, Ran Levinstein, Matan Schliserman, Uri Sherman, Tomer Koren, Daniel Soudry, and Nathan Srebro. From continual learning to SGD and back: Better rates for continual linear models. In *Proceedings of the 37th International Conference on Algorithmic Learning Theory*, volume 313 of *Proceedings of Machine Learning Research*, pages 1–50. PMLR, 2026.
- [44] Huang Fang, Guanhua Fang, Tan Yu, and Ping Li. Efficient greedy coordinate descent via variable partitioning. In Cassio de Campos and Marloes H. Maathuis, editors, *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pages 547–557. PMLR, 27–30 Jul 2021.
- [45] Marshall L Fisher, George L Nemhauser, and Laurence A Wolsey. An analysis of approximations for finding a maximum weight hamiltonian circuit. *Operations Research*, 27(4):799–809, 1979.
- [46] Daniel Goldfarb and Paul Hand. Analysis of catastrophic forgetting for random orthogonal transformation tasks in the overparameterized regime. In *International Conference on Artificial Intelligence and Statistics*, pages 2975–2993. PMLR, 2023.
- [47] Daniel Goldfarb, Itay Evron, Nir Weinberger, Daniel Soudry, and Paul Hand. The joint effect of task similarity and overparameterization on catastrophic forgetting - an analytical model. In *The Twelfth International Conference on Learning Representations*, 2024.
- [48] Michael Griebel and Peter Oswald. Greedy and randomized versions of the multiplicative schwarz method. *Linear Algebra and its Applications*, 437(7):1596–1610, 2012.
- [49] LG Gubin, Boris T Polyak, and EV Raik. The method of projections for finding the common point of convex sets. *USSR Computational Mathematics and Mathematical Physics*, 7(6):1–24, 1967.
- [50] Mert Gurbuzbalaban, Asu Ozdaglar, and Pablo A Parrilo. Convergence rate of incremental gradient and incremental newton methods. *SIAM Journal on Optimization*, 29(4):2542–2565, 2019.
- [51] Mert Gürbüzbalaban, Asuman Ozdaglar, and Pablo A. Parrilo. Why random reshuffling beats stochastic gradient descent. *Mathematical Programming*, 186(1):49–84, Mar 2021. ISSN 1436-4646. doi: 10.1007/s10107-019-01440-w.
- [52] Raia Hadsell, Dushyant Rao, Andrei A Rusu, and Razvan Pascanu. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences*, 24(12):1028–1040, 2020.

- [53] Israel Halperin. The product of projection operators. *Acta Sci. Math.(Szeged)*, 23(1): 96–99, 1962.
- [54] Deren Han and Jiaxin Xie. A simple linear convergence analysis of the randomized reshuffling Kaczmarz method. *Journal of the Operations Research Society of China*, 2025. doi: 10.1007/s40305-025-00623-x.
- [55] Chen He, Ruiping Wang, and Xilin Chen. Rethinking class orders and transferability in class incremental learning. *Pattern Recognition Letters*, 161:67–73, 2022. ISSN 0167-8655.
- [56] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [57] Naoki Hiratani. Disentangling and mitigating the impact of task similarity for continual learning. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [58] Elad Hoffer, Berry Weinstein, Itay Hubara, Sergei Gofman, and Daniel Soudry. In-fer2train: leveraging inference for better training of deep networks. In *NeurIPS 2018 Workshop on Systems for ML*, page 40, 2018.
- [59] Fred Hucht. Solving a specific difference equation. MathOverflow, 2024. URL <https://mathoverflow.net/q/474430>. URL:https://mathoverflow.net/q/474430 (version: 2024-07-04).
- [60] Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://papers.nips.cc/paper_files/paper/2018/hash/5a4be1fa34e62bb8a6ec6b91d2462f5a-Abstract.html.
- [61] Halyun Jeong and Deanna Needell. Linear convergence of reshuffling kaczmarz methods with sparse constraints. *SIAM Journal on Scientific Computing*, 47(4):A2323–A2352, 2025. doi: 10.1137/23M1565206.
- [62] Shanmu Jin. The numerical range is a 2-spectral set. Preprints.org, 2026. Version 4.
- [63] Hyunji Jung, Hanseul Cho, and Chulhee Yun. Convergence and implicit bias of gradient descent on continual linear classification. In *The Thirteenth International Conference on Learning Representations*, 2025.

- [64] Stefan Kaczmarz. Angenäherte auflösung von systemen linearer gleichungen. *Bulletin International de l’Académie Polonaise des Sciences et des Lettres. Classe des Sciences Mathématiques et Naturelles. Série A, Sciences Mathématiques*, 35:355–357, 1937.
- [65] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler L. Hayes, and Christopher Kanan. Measuring catastrophic forgetting in neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1):3390–3398, 2018. doi: 10.1609/aaai.v32i1.11651.
- [66] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.
- [67] Jason M Klusowski and Jonathan W Siegel. Sharp convergence rates for matching pursuit. *IEEE Transactions on Information Theory*, 71(7):5556–5569, 2025.
- [68] Alex Krizhevsky. Learning multiple layers of features from tiny images. Master’s thesis, University of Toronto, 2009. URL <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>.
- [69] Abhimanyu Lad, Rayid Ghani, Yiming Yang, and Bryan Kisiel. Toward optimal ordering of prediction tasks. In *Proceedings of the 2009 SIAM International Conference on Data Mining*, pages 884–893. SIAM, 2009.
- [70] Timothée Lesort, Massimo Caccia, and Irina Rish. Understanding continual learning settings with data distribution drift analysis. *arXiv preprint arXiv:2104.01678*, 2021.
- [71] Timothée Lesort, Oleksiy Ostapenko, Pau Rodríguez, Diganta Misra, Md Rifat Arefin, Laurent Charlin, and Irina Rish. Challenging common assumptions about catastrophic forgetting and knowledge accumulation. In *Conference on Lifelong Learning Agents*, pages 43–65. PMLR, 2023.
- [72] Ran Levinstein, Amit Attia, Matan Schliserman, Uri Sherman, Tomer Koren, Daniel Soudry, and Itay Evron. Optimal rates in continual linear regression via increasing regularization. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [73] Haoran Li, Jingfeng Wu, and Vladimir Braverman. Fixed design analysis of regularization-based continual learning. In Sarath Chandar, Razvan Pascanu, Hanie Sedghi, and Doina Precup, editors, *Proceedings of The 2nd Conference on Lifelong Learning Agents*, volume 232 of *Proceedings of Machine Learning Research*, pages 513–533. PMLR, 22–25 Aug 2023.

- [74] Haoran Li, Jingfeng Wu, and Vladimir Braverman. Memory-statistics tradeoff in continual learning with structural regularization. In *International Conference on Learning Representations*, 2026.
- [75] Ziyang Li and Naoki Hiratani. Optimal task order for continual learning of multiple tasks. In *Forty-second International Conference on Machine Learning*, 2025.
- [76] Sen Lin, Peizhong Ju, Yingbin Liang, and Ness Shroff. Theory on forgetting and generalization of continual learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 21078–21100. PMLR, 23–29 Jul 2023.
- [77] Emiel Lorist and Felix Schwenninger. A solution to crouzeix’s conjecture. *arXiv preprint arXiv:2608.03841*, 2026.
- [78] Yucheng Lu, Si Yi Meng, and Christopher De Sa. A general analysis of example-selection for stochastic gradient descent. In *International Conference on Learning Representations (ICLR)*, volume 10, 2022.
- [79] Ekdeep Singh Lubana, Puja Trivedi, Danai Koutra, and Robert P. Dick. How do quadratic regularizers prevent catastrophic forgetting: The role of interpolation. In *ICML Workshop on Theory and Foundations of Continual Learning*, 2021.
- [80] Stéphane G Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on signal processing*, 41(12):3397–3415, 1993.
- [81] Gabriel Mantione-Holmes, Justin Leo, and Jugal Kalita. Utilizing priming to identify optimal class ordering to alleviate catastrophic forgetting. In *2023 IEEE 17th International Conference on Semantic Computing (ICSC)*, pages 57–64. IEEE, 2023.
- [82] Marc Masana, Bartłomiej Twardowski, and Joost Van de Weijer. On class orderings for incremental learning. *arXiv preprint arXiv:2007.02145*, 2020.
- [83] Cun-Qiang Miao and Wen-Ting Wu. On greedy randomized average block kaczmarz method for solving large linear systems. *Journal of Computational and Applied Mathematics*, 413:114372, 2022.
- [84] Konstantin Mishchenko, Ahmed Khaled, and Peter Richtárik. Random reshuffling: Simple analysis with vast improvements. *Advances in Neural Information Processing Systems*, 33:17309–17320, 2020.
- [85] Amirkeivan Mohtashami, Sebastian Stich, and Martin Jaggi. Characterizing & finding good data orderings for fast convergence of sequential gradient methods. *arXiv preprint arXiv:2202.01838*, 2022.

- [86] Jérôme Monnot. Approximation algorithms for the maximum hamiltonian path problem with specified endpoint (s). *European Journal of Operational Research*, 161(3): 721–735, 2005.
- [87] Deanna Needell and Joel A Tropp. Paved with good intentions: analysis of a randomized block kaczmarz method. *Linear Algebra and its Applications*, 441:199–221, 2014.
- [88] Cuong V Nguyen, Alessandro Achille, Michael Lam, Tal Hassner, Vijay Mahadevan, and Stefano Soatto. Toward understanding catastrophic forgetting in continual learning. *arXiv preprint arXiv:1908.01091*, 2019.
- [89] Thinh Nguyen, Cuong N Nguyen, Quang Pham, Binh T Nguyen, Savitha Ramasamy, Xiaoli Li, and Cuong V Nguyen. Sequence transferability and task order selection in continual learning. *arXiv preprint arXiv:2502.06544*, 2025.
- [90] Yu-Qi Niu and Bing Zheng. A greedy block kaczmarz algorithm for solving large-scale linear systems. *Applied Mathematics Letters*, 104:106294, 2020.
- [91] Julie Nutini, Mark Schmidt, Issam Laradji, Michael Friedlander, and Hoyt Koepke. Coordinate descent converges faster with the gauss-southwell rule than random selection. In *International Conference on Machine Learning*, pages 1632–1641. PMLR, 2015.
- [92] Julie Nutini, Behrooz Sepehry, Issam Laradji, Mark Schmidt, Hoyt Koepke, and Alim Virani. Convergence rates for greedy kaczmarz algorithms, and faster randomized kaczmarz rules using the orthogonality graph. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, UAI’16, page 547–556, Arlington, Virginia, USA, 2016. AUAI Press. ISBN 9780996643115.
- [93] Peter Oswald and Weiqi Zhou. Convergence analysis for kaczmarz-type methods in a hilbert space framework. *Linear Algebra and its Applications*, 478:131–161, 2015.
- [94] Steven C Pan. The interleaving effect: mixing it up boosts learning. *Scientific American*, 313(2), 2015.
- [95] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.
- [96] Liangzu Peng, Paris Giampouras, and Rene Vidal. The ideal continual learner: An agent that never forgets. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings*

- of *Machine Learning Research*, pages 27585–27610. PMLR, 23–29 Jul 2023. URL <https://proceedings.mlr.press/v202/peng23a.html>.
- [97] Anastasia Pentina, Viktoriia Sharmanska, and Christoph H Lampert. Curriculum learning of multiple tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5492–5500, 2015.
 - [98] Chao Qin and Daniel Russo. Adaptivity and confounding in multi-armed bandit experiments. *arXiv preprint arXiv:2202.09036*, 2022.
 - [99] Shashank Rajput, Kangwook Lee, and Dimitris Papailiopoulos. Permutation-based SGD: Is random optimal? In *International Conference on Learning Representations*, 2022.
 - [100] Vinay Venkatesh Ramasesh, Ethan Dyer, and Maithra Raghu. Anatomy of catastrophic forgetting: Hidden representations and task semantics. In *International Conference on Learning Representations*, 2021. URL <https://iclr.cc/virtual/2021/poster/2907>.
 - [101] Aaditya Ramdas. Rows vs columns for linear systems of equations-randomized kaczmarz or coordinate descent? *arXiv preprint arXiv:1406.5295*, 2014.
 - [102] S. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert. icarl: Incremental classifier and representation learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5533–5542, Los Alamitos, CA, USA, jul 2017. IEEE Computer Society. doi: 10.1109/CVPR.2017.587.
 - [103] Benjamin Recht and Christopher Ré. Toward a noncommutative arithmetic-geometric mean inequality: Conjectures, case-studies, and consequences. In Shie Mannor, Nathan Srebro, and Robert C. Williamson, editors, *Proceedings of the 25th Annual Conference on Learning Theory*, volume 23 of *Proceedings of Machine Learning Research*, pages 11.1–11.24, Edinburgh, Scotland, 25–27 Jun 2012. PMLR. URL <https://proceedings.mlr.press/v23/recht12.html>.
 - [104] Simeon Reich and Rafał Zalas. Polynomial estimates for the method of cyclic projections in hilbert spaces. *Numerical Algorithms*, 94(3):1217–1242, 2023. doi: 10.1007/s11075-023-01533-w.
 - [105] Doug Rohrer, Robert F Dedrick, and Sandra Stershic. Interleaved practice improves mathematics learning. *Journal of Educational Psychology*, 107(3):900–908, 2015. doi: 10.1037/edu0000001.
 - [106] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural

- networks. *arXiv preprint arXiv:1606.04671*, 2016.
- [107] Paul Ruvolo and Eric Eaton. Active task selection for lifelong machine learning. In *Twenty-seventh AAAI conference on artificial intelligence*, 2013.
 - [108] Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Yonatan Belinkov, and Stephan Vogel. Neural machine translation training in a multi-domain scenario. *arXiv preprint arXiv:1708.08712*, 2017.
 - [109] Nikolaos Sarafianos, Theodore Giannakopoulos, Christophoros Nikou, and Ioannis A Kakadiaris. Curriculum learning for multi-task classification of visual attributes. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2608–2615, 2017.
 - [110] Coen Schouten. Investigating task order in online class-incremental learning. Master’s thesis, Department of Mathematics and Computer Science, AutoML Group, Eindhoven University of Technology, Netherlands, 2024.
 - [111] Haozhe Shan, Qianyi Li, and Haim Sompolsky. Order parameters and phase transitions of continual learning in deep neural networks. *Proceedings of the National Academy of Sciences*, 123(6):e2501899123, 2026. doi: 10.1073/pnas.2501899123.
 - [112] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 761–769, 2016. doi: 10.1109/CVPR.2016.89.
 - [113] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. *International Journal of Computer Vision*, 130(6):1526–1565, 2022. doi: 10.1007/s11263-022-01611-x.
 - [114] Jonathan E. Spingarn. Monotone operations in mathematical programming. Osp final research report, Georgia Institute of Technology, 1984.
 - [115] Thomas Strohmer and Roman Vershynin. A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.
 - [116] Yansheng Su, Deren Han, Yun Zeng, and Jiaxin Xie. On the convergence analysis of the greedy randomized Kaczmarz method. *Numerical Algorithms*, 2026. doi: 10.1007/s11075-026-02352-5.
 - [117] Ruo-Yu Sun. Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, 8(2):249–294, 2020.

- [118] Siddharth Swaroop, Cuong V Nguyen, Thang D Bui, and Richard E Turner. Improving and understanding variational continual learning. *arXiv preprint arXiv:1905.02099*, 2019.
- [119] William Swartworth, Deanna Needell, Rachel Ward, Mark Kong, and Halyun Jeong. Nearly optimal bounds for cyclic forgetting. In *Advances in Neural Information Processing Systems*, volume 36, pages 68197–68206, 2023. doi: 10.52202/075280-2982.
- [120] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.
- [121] Sebastian Thrun and Tom M Mitchell. Lifelong robot learning. *Robotics and autonomous systems*, 15(1-2):25–46, 1995.
- [122] Matan Tsipory, Ran Levinstein, Itay Evron, Mark Kong, Deanna Needell, and Daniel Soudry. Are greedy task orderings better than random in continual linear regression? In *Advances in Neural Information Processing Systems*, volume 38, pages 172355–172456, 2025. doi: 10.52202/085713-5192.
- [123] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.
- [124] Gido M Van de Ven, Tinne Tuytelaars, and Andreas S Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 4(12):1185–1197, 2022.
- [125] John Von Neumann. On rings of operators. reduction theory. *Annals of Mathematics*, pages 401–485, 1949.
- [126] Jiachen T. Wang, Tong Wu, Dawn Song, Prateek Mittal, and Ruoxi Jia. GREATS: Online selection of high-quality data for LLM training in every iteration. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [127] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329.
- [128] Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):4555–4576, 2022. doi: 10.1109/TPAMI.2021.3069908.
- [129] A-Qin Xiao, Jun-Feng Yin, and Ning Zheng. On fast greedy block kaczmarz methods for solving large consistent linear systems. *Computational and Applied Mathematics*,

42(3):119, 2023.

- [130] Yuetong Xu and Baharan Mirzasoleiman. Ordering for non-replacement sgd. *arXiv preprint arXiv:2306.15848*, 2023.
- [131] Yingxiang Yang, Zhihan Xiong, Tianyi Liu, Taiqing Wang, and Chong Wang. Fourier learning with cyclical data. In *International Conference on Machine Learning*, pages 25280–25301. PMLR, 2022.
- [132] Zhaonan Yang and Huiyun Li. Task ordering matters for incremental learning. In *2021 International Symposium on Networks, Computers and Communications (ISNCC)*, pages 1–6, 2021.
- [133] Moslem Zamani and François Glineur. Exact convergence rate of the subgradient method by using polyak step size. *arXiv preprint arXiv:2407.15195*, 2024.
- [134] Yun Zeng, Deren Han, Yansheng Su, and Jiaxin Xie. Stochastic dual coordinate descent with adaptive heavy ball momentum for linearly constrained convex optimization. *Numerische Mathematik*, 158(2):749–794, 2026. doi: 10.1007/s00211-026-01526-6.
- [135] Jian-Jun Zhang. A new greedy kaczmarz algorithm for the solution of very large linear systems. *Applied Mathematics Letters*, 91:207–212, 2019.
- [136] Jianhua Zhang, Yuqing Wang, and Jing Zhao. On maximum residual nonlinear kaczmarz-type algorithms for large nonlinear systems of equations. *Journal of Computational and Applied Mathematics*, 425:115065, 2023.
- [137] Yanjun Zhang and Hanyu Li. Greedy motzkin–kaczmarz methods for solving linear systems. *Numerical Linear Algebra with Applications*, 29(4):e2429, 2022.
- [138] Xuyang Zhao, Huiyuan Wang, Weiran Huang, and Wei Lin. A statistical theory of regularization-based continual learning. In *Forty-first International Conference on Machine Learning*, 2024.