UC Irvine
UC Irvine Electronic Theses and Dissertations

Title

Mechanotyping Cells through Microfluidics and Machine Learning

Permalink

https://escholarship.org/uc/item/5st0864k

Author

Combs, Cody Tyler

Publication Date

2022

Copyright Information

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,
IRVINE

Mechanotyping Cells through Microfluidics and Machine Learning

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Physics

by

Cody Tyler Combs

Dissertation Committee:
Professor Zuzanna S. Siwy, Chair
Professor Peter Taborek
Professor Xiaohui Xie

2022

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to extend my deepest thanks to my research advisor and committee chair, Professor Zuzanna Siwy, who's patience, guidance, and leadership has made all this work possible. I have learned a great deal from you both as a scientist and as a person, and for that I am thankful to have spent the last five years working together.

I would also like to thank Professor Peter Taborek and Professor Xiaohui Xie for serving on both my advancement committee and doctoral committee, as well as for their support of my research as both collaborators and mentors throughout my studies.

Thank you to all of my lab mates throughout the years for your help and kindness. I always looked forward to going to lab each day and you are all a large part of that.

Thank you to all of my collaborators throughout the years, without you none of this research would have been remotely possible.

I also would like to thank all of my family and friends who have been by my side throughout this journey, I would not have finished without you.

VITA

Cody Tyler Combs

EDUCATION

Doctor of Philosophy in Physics   **2022**
University of California Irvine   *Irvine, CA*

Bachelor of Science in Physics   **2016**
The College of New Jersey   *Ewing, NJ*

RESEARCH EXPERIENCE

Graduate Research Assistant   **2017–2022**
University of California, Irvine   *Irvine, California*

NSF Fellow - Machine Learning and Physical Sciences   **2018–2020**
University of California, Irvine   *Irvine, California*

Research Assistant   **2015–2016**
The College of New Jersey   *Ewing, NJ*

TEACHING EXPERIENCE

Teaching Assistant   **2016–2017**
University of California Irvine   *Irvine, CA*

REFEREED JOURNAL PUBLICATIONS

C. Combs, D. Seith, M. Bovyn, S. Gross, X. Xie, and Z. Siwy. Deep learning assisted mechanotyping of individual cells through repeated deformations and relaxations in undulating channels. *Biomicrofluidics*, 16:014104, (2022)

C.Y. Lin, C. Combs, Y. Su, L.H. Yeh, and Z. S. Siwy. Rectification of concentration polarization in mesopores leads to high conductance ionic diodes and high performance osmotic power. *J. Am. Chem. Soc.* 141(8):3691-3698, (2019)

Acar, E. T., Buchsbaum, S. F., Combs, C., Fornasiero, F., and Siwy, Z. S.. Biomimetic potassium-selective nanopores. *Science Advances*, 5(2), (2019)

ABSTRACT OF THE DISSERTATION

Mechanotyping Cells through Microfluidics and Machine Learning

By

Cody Tyler Combs

Doctor of Philosophy in Physics

University of California, Irvine, 2022

Professor Zuzanna S. Siwy, Chair

Traditionally, phenotyping cells has been approached through the biological lens of looking at proteins, DNA, RNA, or other molecular descriptions of cells. We present the use of physical properties of cells that emerge as as a consequence of these molecular properties, as a means to mechanically phenotype single cells. In this work, we phenotype two cell types, mammalian and bacterial cells, through separate approaches. In the first study, we create a high-throughput single cell microfluidic stretcher, where we track non-linear dynamics of deforming cells at a high spatial and temporal resolution. We apply image and sequence-based deep learning models for feature extraction that are trained using time sequences of cell shapes, and we found to achieve high classification accuracy based on cytoskeletal properties alone. The best model classified sub-populations with an accuracy over 90%, significantly higher than the 75% we achieved with traditional methods. This increase in accuracy corresponds to a five fold increase in potential enrichment of a sample for a target population. In the second study, we utilize an electrical impedance based approach, called resistive-pulse sensing, to classify bacteria cells based upon their size, shape, and surface potentials. In this method, as individual bacteria cells transit the polymer pore, we observe a changes in electrical impedance proportional to the particle volume, and differences in transit time that is proportional to the surface potential. These features are used to classify three bacteria populations which differ in shape at a high accuracy through unsupervised clustering. We

compare this approach to a shape based approach, upon which the resistive-pulse signal is averaged by bacteria of different lengths as they travel along an undulating micropore. This work establishes the application of sequence-based machine and deep learning models to dynamic deformability cytometry and resistive-pulse sensing of bacteria.

Chapter 1

Introduction

One of the fundamental questions of biology is how to identify a cell. Cells are typically identified by their molecular properties, such as the type of proteins that are being expressed. Advancements in sequencing technologies have allowed for the identification of single cells through reading RNA, or attaching surface bio markers with fluorophores. However, these methods can be costly and require a great deal of sample preparation. Identifying cells by physical properties such as capacitance, index of refraction, or elasticity, among others, provides a label-free alternative. Cells can exhibit a wide range of values for each of these physical properties, and can be seen complex materials. Often the physical descriptors align with the function of the cell, such as the ability of red blood cells to deform in order to travel through tight capillaries in the body. Mechanical phenotyping (mechanotyping) in particular has shown great promise for differentiating cell types. The advent of microfluidic based stretchers has enabled high-throughput mechanical characterization of single cells on the order of 1000 cells per second, compared to traditional methods like AFM or micropipette aspiration which are on the order of 1 cell per minute. The use of microfluidic channels to characterize single cell mechanics has been deemed deformability cytometry, and has seen a host of channel designs over the 10 years since its introduction.

In that same 10 year period, machine learning, in particular deep learning, has seen a resurgence and is ubiquitous in almost any data driven field. The combination of large data, hardware advances, and theoretical development of deep learning has lead to this recent rise. However, deep learning has only recently taken hold in the physical and biological sciences, and there remains a great deal of uncertainty in the application in these fields. Deep learning approaches have recently shown great promise in identifying cells types based upon physical descriptors such as morphology. In this thesis, we expand upon these physical descriptors to demonstrate the utility of machine and deep learning approaches to the field of mechanical phenotyping, by leveraging the high-throughput nature of single-cell microfluidics.

In this thesis, we will first present the theoretical backing for fluid mechanics in microfluidic systems, and derive equations for fluid flows and stresses. We will then compare these analytical predictions to simulations of fluid flow in our full experimental microfluidic device. We will then discuss the theoretical backing of both traditional machine learning algorithms and how they 'learn' from data, and how these compare to deep learning methods. We will then present explore two different techniques for phenotyping cells by their physical properties. In the first project, we study HL60 cells which have been treated with various drugs to induce cytoskeletal perturbations. We developed a single cell microfluidic stretcher to detect these mechanical perturbations, where we measure the dynamic deformation response of individual cells. Traditional machine learning are trained with derived features to establish a baseline for classification comparison to deep learning approaches which learn from time sequences of deformation shapes. In the subsequent chapter, we provide two examples of the application of this method to two problems: detection of activated immune cells called neutrophils, and characterization of a unique cell type found in mouse ears called lipochondrocytes. In the final chapter, we present an electrical approach to classify bacteria cells based upon their shape and surface potentials. We utilize the resistive-pulse principle along with polymer micropores to distinguish between bacteria of different shapes. The classification accuracy of two different machine learning approaches were compared; one using derived features, and

one comparing the unique resistive-pulse signal of each bacteria species. The projects in this thesis establish the use of machine learning to mechanotyping of single cells.

Chapter 2

Background and Theory

2.1 Fluid Mechanics Background

2.1.1 Fluid flow in micro-channels

In microchannels, the transport of fluid can be used for delivering chemical species, manipulation and measurement of various particulates, and even replicating biological organs, among others [3, 4, 5, 6]. The fluid flow profiles in these various applications are dependent on both channel geometry, as well as material properties of the fluid, such as viscosity. As a result, understanding how fluids behave in channels is fundamental to improving and creating new microfluidic technologies. In this section, fluid flow profiles which are driven in particular by pressure differences will be discussed for different geometries and length scales. The following text has been used as a reference for the derivations on fluid mechanics to follow, and is recommended for a more complete description[7].

When discussing fluid mechanics, one often has to make assumptions either about properties of the fluid under study, as well as interactions between the fluid and other surfaces. Here

4

we will introduce assumptions and analytically calculate the fluid profiles resulting from those assumptions. Later, we will discuss where those assumptions fail, and how the fluid flow profiles differ from when those assumptions are dropped. To begin, we will make the assumption that the fluid under study is incompressible. Fluid flow is considered to be incompressible if the density can be assumed to be constant and uniform. The continuity equations along with the Navier-Stokes equations alone can describe laminar fluid flow for an incompressible fluid. Through conservation of mass, the continuity equation can be written in integral form as:

$$\frac{\partial}{\partial t} \int_V \rho \, dV = - \int_S (\rho \vec{u}) \cdot \hat{n} \, dA \qquad (2.1)$$

The continuity equation (conservation of mass) states that the flux of mass crossing a surface of volume V is equal to the rate of change of mass in the same volume V. Utilizing the divergence theorem, we can rewrite this equation in differential form:

$$\frac{\partial \rho}{\partial t} - \nabla \cdot (\rho \vec{u}) = 0 \qquad (2.2)$$

As stated before, we assumed that ρ is constant and we find that the continuity equation for an incompressible fluid becomes:

$$\nabla \cdot \vec{u} = 0 \qquad (2.3)$$

Equation 2.3 states that the divergence of the velocity vector field is zero everywhere inside an incompressible fluid. Physically, this means that there are no sources or sinks of velocity, or in other words; any infinitesimal volume in the fluid has zero net flux.

The other equation needed to understand incompressible fluid flow is the Navier-Stokes equation, which in general form is the conservation of momentum in the continuum limit and is given by:

$$\frac{\partial}{\partial t}\int_V \rho\vec{u}\,dV = -\int_S (\rho\vec{u}\vec{u})\cdot\hat{n}\,dA + \int_S \vec{\vec{\tau}}\cdot\hat{n}\,dA + \int_V \sum_i \vec{f_i} \qquad (2.4)$$

Again assuming ρ is constant and using the divergence theorem along with the derived result of equation 2.3, we can simplify equation 2.4 to the form:

$$\rho\frac{\partial\vec{u}}{\partial t} + \rho\vec{u}\cdot\nabla\vec{u} = \nabla\cdot\vec{\vec{\tau}} + \sum_i \vec{f_i} \qquad (2.5)$$

Equation 2.5 is known as the cauchy momentum equation. The cauchy momentum equation describes how the changes in momentum in time ($\rho\frac{\partial\vec{u}}{\partial t}$) are created by convection of fluid out of a control volume ($\rho\vec{u}\cdot\nabla\vec{u}$), the surface stresses from volume forces ($\nabla\cdot\vec{\vec{\tau}}$), and the net body forces ($\vec{f_i}$).

The $\nabla\cdot\vec{\vec{\tau}}$ term describes both pressure and viscous forces. Pressure forces are perpendicular to the control surface and velocity independent. The viscous stress terms can be both perpendicular and tangential to the control surface and can depend on the local fluid velocity. Consequently, the stress term can be rewritten as:

6

$$\nabla \cdot \vec{\vec{\tau}} = -\nabla p + \nabla \cdot \vec{\vec{\tau}}_{visc} \tag{2.6}$$

To solve the Navier-Stokes equation, we need to find a relation between the viscous stress ($\vec{\vec{\tau}}_{visc}$) and the velocity field ($\vec{u}$). To link these quantities it is necessary to define a constitutive relation, which connect microscopic properties to the continuous macroscopic properties of matter. This relation cannot be derived and is a postulate of continuum fluid mechanics. For Newtonian fluids the stress is linearly related to the strain rate through the viscosity, where the viscosity is assumed to be independent of the strain rate. Viscosity is a property of a fluid that quantifies how much a fluid deforms from the the strain rate. We will see that in our actual system, we work with a fluid which contains high amounts of polymer molecules and has a viscosity which is dependent on the strain rate. The general constitutive relation for 3D flow between strain rate and stress is given by:

$$\vec{\vec{\tau}}_{visc} = 2\eta \vec{\vec{\epsilon}} \tag{2.7}$$

where the generalized 3D strain rate is:

$$\vec{\vec{\epsilon}} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{\partial v}{\partial y} & \frac{1}{2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \\ \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) & \frac{1}{2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) & \frac{\partial w}{\partial z} \end{bmatrix} \tag{2.8}$$

Using the fact that $\nabla \cdot 2\eta\vec{\vec{\epsilon}} = \nabla \cdot \eta\nabla\vec{u}$ along with Eq 2.6 and assuming the viscosity (η) is

uniform throughout the fluid, we can find the general 3D form of Navier-Stokes for Newtonian fluids:

$$\rho\frac{\partial \vec{u}}{\partial t} + \rho\vec{u}\cdot\nabla\vec{u} = -\nabla p + \eta\nabla^2\vec{u} \tag{2.9}$$

2.1.2   Pressure driven flow in microchannels

Now that we have derived Navier-Stokes, we will solve the equation for a simplified system where we assume a tube with a circular cross section where the flow is resulting from pressure gradients within the tube. This type of flow is referred to as Hagen-Poiseuille flow. Since we are able to analytically solve Hagen-Poiseuille flow, it can help estimate fluid velocities and viscous stresses occurring in our microchannels, whereas computational simulations are necessary to model fluid flow in more complex channel geometries.

First, we will use the symmetry of the system to make assumptions about the the fluid flow. Using a cylindrical coordinate system, we will assume that all of the fluid flow will be along the direction of the pipe (u_z) so that (u_θ, u_r)=0. Additionally we will assume the velocity gradients in those directions are also equal to zero. This results in a much more simple form of Navier-Stokes given by:

$$\nabla p = \eta\nabla^2\vec{u} \tag{2.10}$$

This can be analytically solved through separation of variables and applying the no-slip boundary condition ($u_t = 0$) resulting in:

$$u_z = -\frac{1}{4\eta}\frac{\partial p}{\partial z}(R^2 - r^2) \tag{2.11}$$

where R is the radius of the channel cross-section. The important result we find that the fluid velocity follows a parabolic profile reaching a maximum at the center of the channel. In this simple system there are only two non-zero terms for the viscous stress:

$$
\begin{aligned}
\tau_{zr} = \tau_{rz} &= \eta\frac{\partial u_z}{\partial r} \\
&= \frac{r}{2}\frac{\partial p}{\partial z}
\end{aligned}
\tag{2.12}
$$

We see that the stress various linearly with the radius for Hagen-Poiseuille flow and reaches a maximum at the wall of the channel ($r = R$). This radial gradient of stress can provide non-contact forces (e.g. friction with solid wall) to deform particles which will be the basis of our microfluidic stretching device.

2.1.3 Electro-osmotic Flow

In addition to pressure driven flow, fluid can also be transported through micro and nano channels through the application of electric fields, where this phenomena is referred to as electro-osmotic flow[8]. Electro-osmotic flow is the result of counter-ions in the fluid dragging surrounding molecules, i.e. water molecules. The movement of the counter-ions is driven by the electric force applied by an external field. The velocity of the fluid in the bulk due to electro-osmosis is a consequence of the interaction the solid-liquid interface that occurs at the channel walls, and is dependent on the material properties and solution bulk conductivity.

To understand this solid-liquid interface requires understanding the electrostatic interactions between ions in liquid and surface charges present on the solid material. There exists a characteristic region at this interface, referred to as the electrical double layer (EDL), which is used as an approximation for the length scales of interaction from the surface charges into the fluid. Conceptually, the EDL is a layer of counter-ions which screen the oppositely charged solid surface. The thickness of the EDL is often defined by the debye length which is given by:

$$\lambda = \left(\frac{\epsilon_0 \epsilon_r k_B T}{e^2 \sum_i n_i^\infty z_i^2} \right)^{1/2} \tag{2.13}$$

The debye length primarily depends on the bulk ion concentration and describes where the surface potential has decayed by a value of $\frac{1}{e}$. Within the debye length ion concentration is enhanced, and in small systems such as nanopores can lead to interesting non-linear behavior which acts as the basis for non-linear ionic circuit elements, such as ionic diodes or transistors. Typical debye lengths for biologically relevant salt conditions (0.1 M KCl) are on the order of around $\sim 1nm$, far below the length scales seen in our microfluidic systems $(1 - 20\mu m)$. While the enhancement of ion-concentration from the EDL has little effect on electrical characterization in our systems, the EDL is basis for electro-osmotic flow in microfluidic channels.

An approach to calculating the electro-osmotic flow velocity in a system with charged walls is to define an inner and outer solution, where at the boundary those two solutions converge to the same value. We can determine the inner solution by solving the Naiver-Stokes equation in the thin region defined by the EDL. Here, the body forces on the fluid are given by the externally applied electric field $\vec{E}_{ext}$ to the net charge density ρ_e. Assuming that the velocity is in the direction of the channel axis, and that the gradients in velocity and potential are

in the perpendicular direction, we can simplify the Navier-Stokes equation and solve for the inner solution:

$$u_{inner} = \frac{\epsilon E_{ext}}{\eta}(\phi - \phi_0) \tag{2.14}$$

We can see from this derived equation that in the thin region of the EDL, if we assume the electric field (E_{ext}) is uniform, that the fluid velocity changes linearly with respect to the electric potential (ϕ) as we move away from the wall. From this equation, we can find the boundary condition for the inner solution far from the wall, by using that ϕ should approach zero far from the wall. This result is know as the Helmhotz-Smolchowski equation. We can find the outer (or bulk) solution, and use this boundary condition to match the two solutions. In general to solve for the outer bulk solution, we need to solve Laplace's equation $\nabla\phi = 0$ since the bulk fluid is neutrally charged, in additiona to Navier-Stokes equation with the boundary condition from the inner solution. However, if we assume that the flow is irrotational, then solving Navier-Stokes is not necessary, and the velocity is given again by equation 2.14, where in this case $\vec{E}_{ext}$ is the local electric field. We see that the fluid velocity is proportional to the local electric field, where the proportionality constant is deemed the electroosmotic mobility (eq. 2.15).

$$\mu_{EO} = -\frac{\epsilon \Phi_0}{\eta} \tag{2.15}$$

One of the conditions that is required to assume that the flow is irrotational is that the system is operating at a low Reynolds number. Next, we will describe what the dimensionless quantity known as Reynolds number, along with its implications for various values.

2.1.4   Reynolds number in microfluidics

Flow can be classified into two main regimes, laminar and turbulent flow, where the regime in between is referred to as transitional flow. A Laminar flow is qualitatively described by layers of fluid that form streamlines and flow parallel to each other with little to no mixing between layers. Whereas laminar flow is stable, turbulent flow is unpredictable and has streamlines which intersect. Laminar or turbulent flow can be characterized by a dimensionless quantity known as the Reynolds number. The Reynolds number is defined by the characteristic length scale and velocity of the system, as well as the fluid density and viscosity. The general relation is given by eq. 2.16:

$$Re = \rho \frac{Ul}{\eta} \tag{2.16}$$

We see that the Reynolds number is proportional to the characteristic length scale. For this reason, laminar flow is most often observed in microfluidic devices, allowing the precise and predictable control of particle and fluid movement. We can use the Reynolds number as a way to determine if the flow in a system will be laminar. In general for cylindrical channels, $Re < 2000$ leads to laminar flow, while at $Re > 4000$ turbulent flow can begin to be observed. For the microfluidics channels used in the studies found in this thesis we can estimate the Reynolds number. Using a hydraulic diameter of $d = 20\mu m$, and the velocity of 0.1 ms^{-1}, we calculate $Re = 1$ for water, well below the threshold for laminar flow. The actual Reynolds number in our system is <<1, due to using a solution with dissolved polymers, resulting in a viscosity greater than that of water. Regardless, this demonstrates that even at large velocities found in our microfluidic system that we are very far from turbulent flow, allowing for the manipulation of single particles.

2.1.5 Electrophoresis

In addition to pressure driven flow and electroosmotic flow, electrophoresis is another method in which to transport particles through channels. Electrophoresis stems from the electric force on a charged particle in solution, allowing for the passive manipulation of charged particles. Electroosmosis and electrophoresis are essentially the same effect, however with electrophoresis the charged surface is moving relative to the surrounding fluid. As a result, if we perform a coordinate transformation so that the fluid is still in the frame of reference we see that the equation for electrophoretic velocity follows the same form as electroosmosis, assuming a thin EDL (eq. 2.17).

$$\vec{u}_{EP} = \frac{\epsilon \Phi_0}{\eta} \vec{E} \tag{2.17}$$

Here, $\vec{E}$ represents the local electric field where the particle is present. If a charged particle is traveling through a channel which is also charged, then we can observed both electrophoretic and electroosmotic transport. These effects could add or subtract depending on the relative magnitudes of signs of the surface potential, also known as the zeta potential. In such a case, the velocity of the particle is given by:

$$\vec{u}_{part} = \frac{\epsilon \Phi_{particle}}{\eta} \vec{E}_0 - \frac{\epsilon \Phi_{wall}}{\eta} \vec{E}_{wall} \tag{2.18}$$

In later studies presented in this work, we will use electrophoresis to transport bacteria species through PET microchannels, where both electrophoresis and electoosmotic transport are both present.

2.1.6   Finite Element Simulation of fluid dynamics in microfluidics

The previously presented sections on fluid transport have displayed the fundamental prin-
ciples of fluid dynamics. By making assumptions on the nature of the flow and assuming
symmetric cylindrical channels, we were able to derive eq. 2.11 which describes the parabolic
velocity profile in pressure driven flow. However, for all but the simplest cases can partial
differential equations (PDE) like Navier-Stokes be solved analytically, and we rely on nu-
merical techniques to solve such systems. Numerical approaches for solving PDEs work by
discretizing the space and numerically estimating the solution in each sub region. Through
the finite element method (FEM), we can numerically solve Naiver-Stokes (and many other
equations) for any geometry, flow rate, or material property. We utilize COMSOL Multi-
physics 5.3 to perform the FEM modeling. For our microfluidic channels, we are in particular
interested in the fluid velocities and stresses in order to estimate forces single cells may ex-
perience. To verify that our FEM modeling of Navier-Stokes is accurate, we will perform
the simulation on a cylindrical pipe with pressure driven flow, where we already know the
analytical solution.
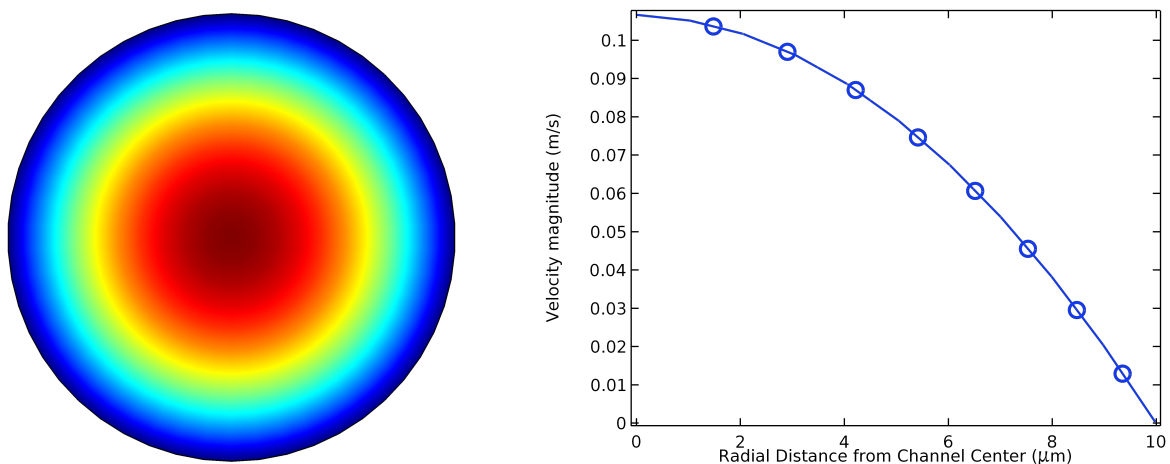
(a) Cross section of velocity in cylindrical channel   (b) Velocity as a function of radial distance

Figure 2.1: **FEM simulation of pressure driven flow in cylindrical channel.** The full
Navier-Stokes equation was solved with no-slip boundary condition for a cylindrical channel
with pressure driven flow. The velocity decays quadratically as a function of the radial
distance from the center.

We can solve the the full Naiver-Stokes equation for a cylindrical pipe with no slip boundary conditions at the walls. Here we will not make any assumptions about the flow profile. Using the experimental flow rate found in our systems of $Q = 1\mu Lmin^{-1}$, we can solve Navier-Stokes and look at the velocity magnitudes in a cross-sectional slice, as seen in figure 2.1. We see from the FEM simulation that the velocity profile assumes a parabolic profile as a function of distance from the center of the channel, with the maximum velocity found in the center. Without making any assumptions about the flow profile we were able to numerically find a solution that agrees with the analytical model, indicating that the assumptions about flow were valid. However, our actual channel geometry is much more complicated, and an analytical solution for the flow is not possible.
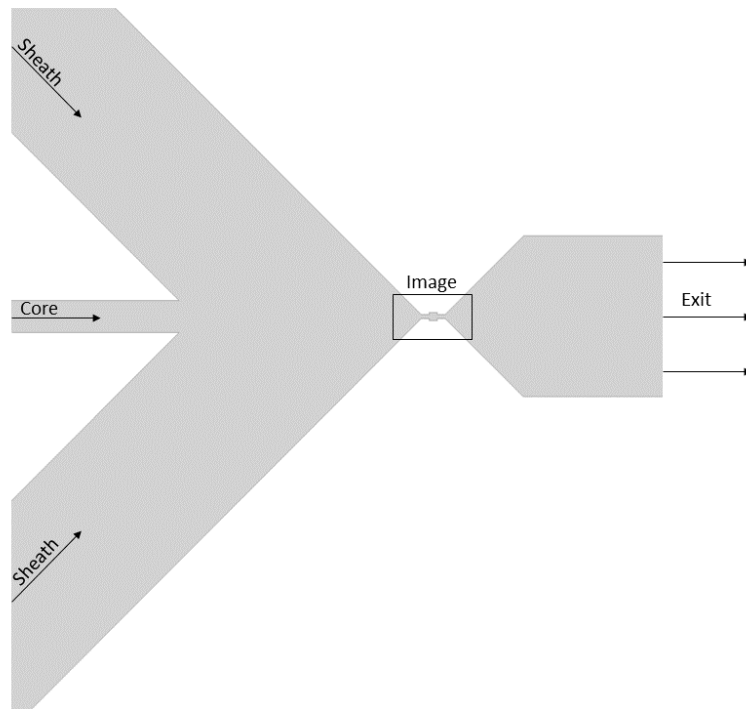
Figure 2.2: **Channel geometry of microfluidic device** Slice of true channel geometry used in microfluidic stretching device. Image was imported from CAD to COMSOL for fluid dynamics simulation. The two sheath flows focus the transport in the core, and the deforming particles are imaged in the rectangular region.

The full chip geometry utilized in our system consists of three inlets: the core where cells flow, and two sheath flows to focus the flow laterally. These three inlets combine to a single
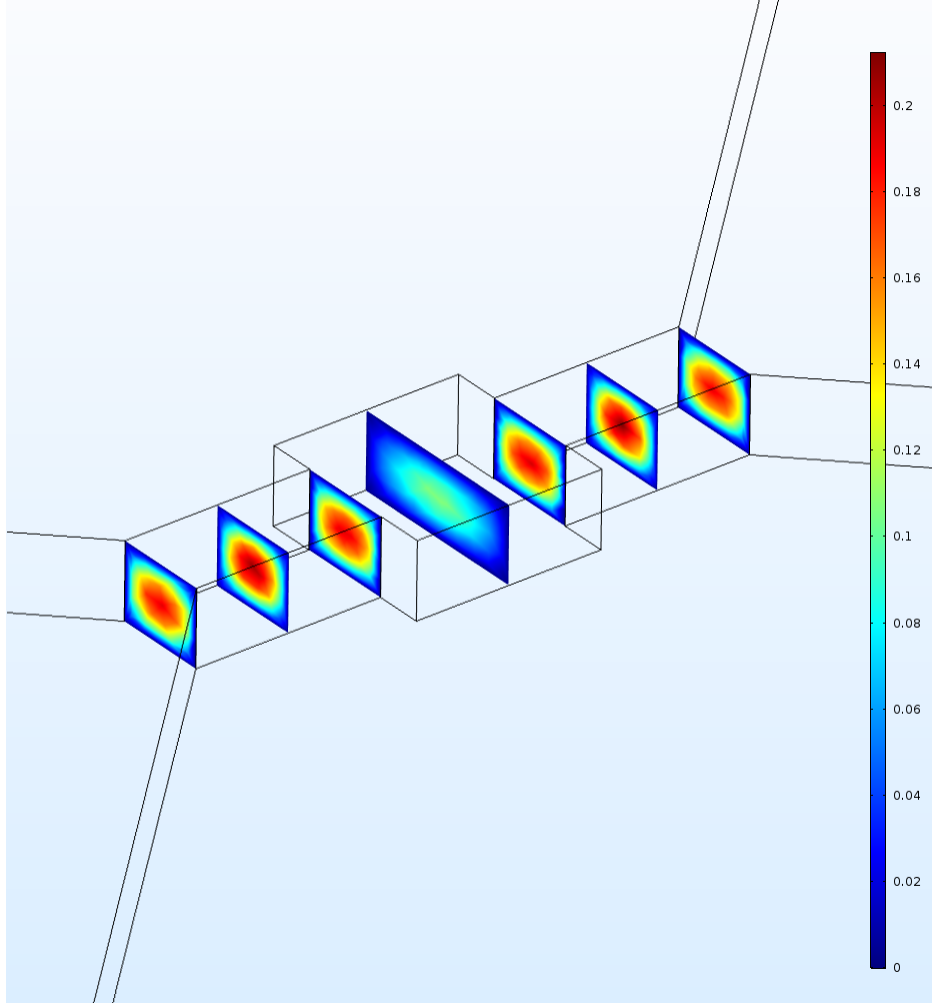
Figure 2.3: **Velocity distribution for undulating rectangular channels** A FEM simulation was performed on the full channel geometry (figure 2.2). Cross-sectional slices of velocity magnitude were take from 0 - 150μm in 25μm increments across the channel length.

channel with a constant height of 20μm of undulating width of 25μm, 50μm, and 25μm. The channel geometry can be seen in figure 2.2. The undulating channel widths creating velocity gradients along the channel axis. Figure 2.3 shows the velocity profile for the rectangular cross sections at various points in the undulating channel. We can see at the entrance that the velocity profile reaches a maximum at the center of the cross section, similar to the cylindrical channels. FEM is powerful in that it allows us to calculate fluid flows and stresses for different channel geometries and guide the design of channels used in experiment.

2.2 Machine learning background

2.2.1 Supervised learning

The field of machine learning can generally be broken down into three subcategories: supervised, unsupervised, and reinforcement learning. This thesis will utilize methods from both supervised and unsupervised learning. In supervised learning, the goal is to make a prediction based upon a set of features that are input into a model. These predictions can be either continuous or discrete values. Supervised models have access to the ground truth values or labels while learning. By having access to the ground truth values, supervised methods can compare how far their predicted outputs are from these ground truth values and use this error to intelligently update the model weights. In general, all supervised methods, regardless of their complexity, contain the following elements:

1. Model with weights
2. Cost function
3. Optimization Algorithm

Model choice is dependent on the data to be modeled, and the weights are the parameters to be learned or fit to the data. A cost function needs to be defined in order to determine how far a models prediction is from the ground truth value. By minimizing the cost function we can find the optimal models weights. An optimization method is needed to find these weights. This process of iteratively evaluating the model cost, and updating weights based upon the cost function value is the process of machine learning. The simplest example of supervised learning is linear regression. In this case the model is given by y = m x + b and would be used if your input data (x), follow a linear relation to your output variable (y). The model weights (m and b) need to be learned based upon the input data and the ground

17

truth label (y). The cost function for linear regression is known as the mean squared error and is given by :

$$J = \sum_{i=1}^{N} (\hat{y} - y_i)^2 \tag{2.19}$$

While this cost function has an analytical solution for best weights, this is most often not the case and an optimization algorithm is needed. Gradient descent is a popular choice for optimization, but many others are used, such as ADAM. The goal of gradient descent is to find how the cost function changes with respect to the model weights. Weights are updated by taking the partial gradient of the loss function and then multiplied by a learning rate (alpha) and added to the previous weights (eq. 2.20). This general process of defining a model, evaluating the cost function, and updating model weights are used from linear regression, to deep fully connected neural networks. The 'only' difference being larger, more complicated models require different cost functions and have harder optimization problems.

$$\theta_{i+1} = \theta_i - \alpha \frac{\partial J}{\partial \theta} \tag{2.20}$$

In the remaining sections, we will detail how specific supervised methods work which are used in this thesis. We will begin using a simple model, logistic regression, to introduce classification. We will then describe what we will refer to as 'traditional' machine learning models, and go into the details of support vector machines (SVM) and random forests (RF). We refer to these as 'traditional' methods as they have existed much longer as compared to the recent advancedments in neural networks and deep learning. These methods however should not be dismissed as their theory is well understood and perform well for many

applications. We will then describe 'deep' learning methods, specifically fully connected networks (FNN), and the fundamental unit of a FNN the perceptron. We will then expand to convolutional neural networks (CNN) and recurrent neural networks (RNN) which have applications in image and time-series data, respectively. Note that this background will be merely an introduction to each subject, and the reader is encouraged to the following sources as introductory texts on the subjects [9, 10].

2.2.2 Traditional Machine learning

Logistic Regression

Logistic regression can be seen as the linear regression of classification. Whereas linear regression predicts continuous variables, logistic regression predicts discrete labels for each data point (for example, 0 or 1, cat or dog, malignant or benign). Similar to linear regression, we must define a model, or a hypothesis function to describe our data. This hypothesis often takes the form:

$$H(\theta) = \theta X + b \tag{2.21}$$

Where θ and X could be multivariate and b is the bias. To translate this value to binary labels, we must pass the hypothesis function through an activation function. One activation function that is commonly used is the sigmoid function:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2.22}$$
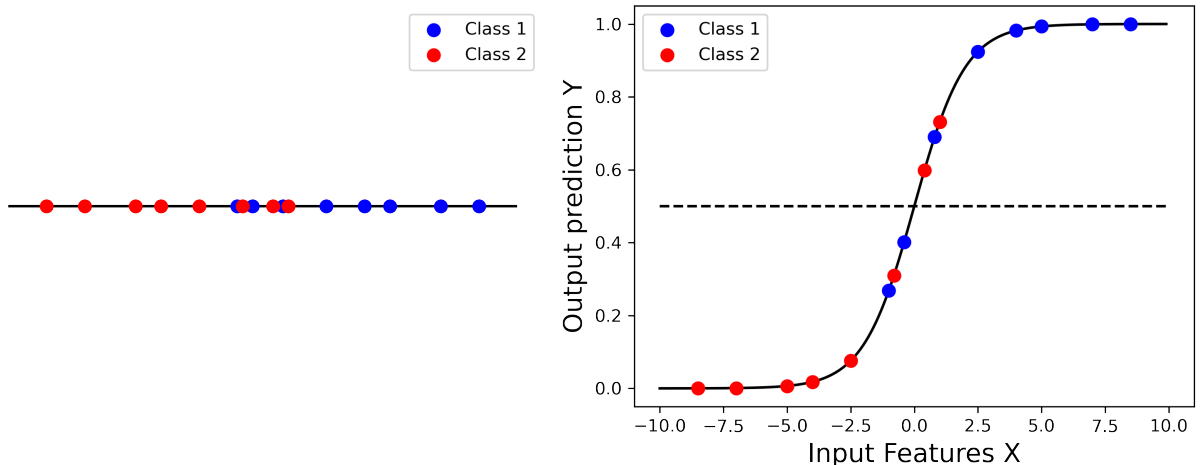
Figure 2.4: **Sigmoid activation for classification** a) Distribution of class label along an arbitrary feature vector b) Mapping of the class labels' feature through sigmoid activation. The dotted line represents a possible decision boundary for classification.

where the hypothesis function would replace z. The sigmoid maps the output of the hypothesis function between 0 and 1. Figure 2.4 shows the mapping of a feature x to a predicted label y. For binary classification, the cost function is given by binary cross-entropy. By minimizing binary cross-entropy, the model is learning to approximate the true distribution of labeled data. Binary cross entropy is a very common cost function used for models where the prediction is binary. The cost function can be cleverly written as:

$$J = \sum_{i=1}^{n} y_i \log y_i + (1 - y_i) \log (1 - y_i) \tag{2.23}$$

In the same fashion as linear regression, we can utilize gradient decent and write down an update rule for the weights (θ) by taking the partial derivative of the cost function with respect to the weights and bias. The gradient descent update rule takes the same form as eq. 2.20. The components we derived above for logistic regression are fundamental for understanding the basic elements of artificial neural networks. Before describing artificial neural networks, we will expand beyond logistic regression and describe support vector machines

(SVM) and random forest (RF).

Support Vector Machines
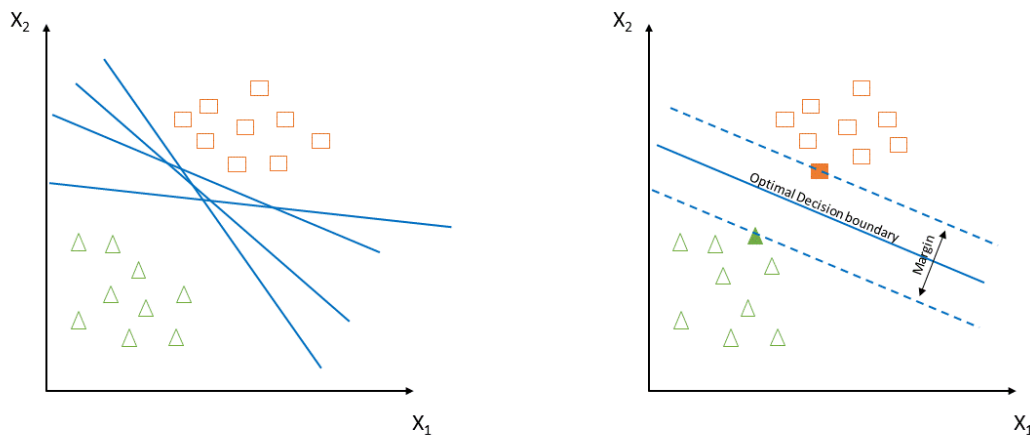
Figure 2.5: **Example SVM classifier boundary line** a) Both classes (green and orange) can be separated by many non-optimal boundary decision lines (shown in blue). b) Optimal decision boundary found by maximizing the margin to the support vectors (closest points to hyperplane).

The goal of support vector machines (SVM) is to find the optimal hyperplane to classify data points among populations. SVM use a a linear decision boundary and are applicable to higher dimensional features spaces. The definition of an optimal hyperplane is easiest to visualize in two dimensional space (R^2), where the hyperplane is a line. If the data under study is completely separable by a linear decision boundary, then there are an infinite number of lines which also satisfy this condition. An example of this can be seen in figure 2.5, where multiple green boundaries are drawn that can result in perfect classification accuracy. However, if new unseen data was added to the model, then these boundary lines do not generalize well and the model can be over fit to the training set. To find the optimal boundary, we can

define a region known as the margin, where the margin is the distance from the boundary line to the closest data points from each class (figure 2.5). The closest points to the margin are defined as the support vectors, and they are the only points used for model training. We can find the optimal hyperplane by maximizing the width of the margin. The equations that define the decision boundary and margins are given by:

$$w^T x - b = 0 \tag{2.24}$$

$$w^T x - b = 1 \tag{2.25}$$

$$w^T x - b = -1 \tag{2.26}$$
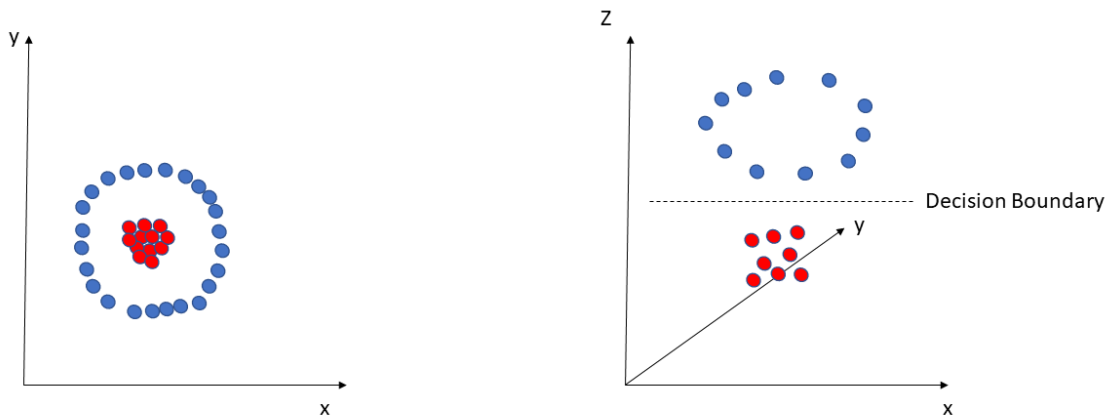
Figure 2.6: **Polynomial kernel trick applied to SVM** a) Two classes of data (red/blue) form clusters which cannot be linearly separated b) Applying a quadratic mapping into high-dimensional space separates the data where a linear hyper plane can fully segment the two classes.

where w is the weight vector which is perpendicular to the decision boundary. The distance

between the two margin hyperplanes is given by $\frac{2}{||w||}$. While the goal is to maximize the margin distance, often in optimization it is best to frame the problem in terms of minimization. By minimizing $||w||$, we obtain the same weights as maximizing the margin. The details of solving this minimization problem will not be presented here. SVM so far has been shown as a classifier with linear decision boundaries, but the algorithm has the ability to classify non-linear data through the use of different kernels. Briefly, the kernel trick can be used to map the feature space into higher dimensional space, where in that space a linear hyperplane is able to separate the data. We can see an example of this in figure 2.6. The higher dimensional hyperplane can then be transformed back to the original feature space where it may take on a non-linear boundary. Common non-linear kernel types used with SVM are polynomial and the radial basis function.

Random Forest

Random forest is an ensemble method, which uses a committee of decision trees to classify. To understand random forest, we must first understand how a decision tree works. Decision trees are tree-like structures that start from a single node called the root node. Each node in the tree represents a feature in which a conditional statement will decide the corresponding branch to traverse. The end nodes, also called the leaf nodes, specify the classification decision. The structure of the decision tree is decided by the information gain performed at each node, where the nodes or features with the most information gain start highest in the tree. To quantify information gain we can use utilize entropy, which is a measure of disorder. The information gain can be determined by the difference between the entropy of the parent and the sum of entropies of the children nodes (Eq. 2.27).

$$H(Y) = -\sum_{i}^{k} P(Y = y_i) \log_2 P(Y = y_i) \qquad (2.27)$$

The information gain for a feature X is given by:

$$IG(X) = H(Y) - H(Y|X) \qquad (2.28)$$

Decision trees are trained by recursively deciding the best feature split until there are no more features and the information gain is below a certain threshold. Decision trees with large depths often suffer from overfitting to training data, where the boundaries between classes can be small. A solution to overfitting is by creating an ensemble of decision trees, hence the name random forest. The 'random' describes the process in which the ensemble of decision trees are created. Each decision tree randomly selects data point or features to use for training. Once the ensemble of trees are trained, classification is determined through voting or averaging from each decision tree.

2.2.3 Deep learning

Here we will discuss the three main types of neural networks used in this work: fully connected neural networks (FCN), convolutional neural networks, and recurrent networks. The descriptions of their operation will be mainly qualitative. This section is simply an overview of the basic operating principles of each network type, as the implementation details are outside the scope of this work (but are highly encourage for the reader to pursue).

Fully connected networks

So far we have discussed what we deemed as traditional machine learning models. We have demonstrated how logistic regression can receive an input vector of features, and through a non-linear mapping of those features, transform the data into label prediction. The process of learning the optimal weights for this transformation can be accomplished through minimizing the loss function using gradient descent. The foundation of logistic regression, the perceptron, was created by modeling how neurons activate in the body; they receive feature inputs (or neurotransmitters in the body), and gate the signal activation depending on the weighted sum of these inputs. To create artificial neural networks, we can combine these logistic regression neurons or 'nodes', in a way where the outputs of one node, become the inputs of another. Furthermore, we can create layers of nodes, where every node outputs to the nodes in the next layer, where each output is weighted. Combining the nodes in such a fashion is called a fully connected artificial neural network (FCN). A diagram showing a simple FCN is shown in figure 2.7.

By fully connecting the nodes, FCN are able to learn non-linear representations of the input features, where each layer can be seen as a mapping of features to a new representation. The layer in between the input and output node are known as hidden layers, as the output from these nodes is never seen. The number of weights in the network is the number of total connections, and increases with size of each layer and the number of layers. Deep learning is essentially a neural network which contains many hidden layers, or a large number of parameters. Some of the largest models at the time of this writing contain over 175 billion parameters! The weights are updated in a similar fashion to the gradient descent algorithm described previously for logistic regression by a process known as back-propagation. However, in this case each layer's input is dependent on the previous layer out, creating a series of differentiation that can be written recursively using the chain rule. One of the largest disadvantages of FCN are the number of parameters contained in the networks. Another
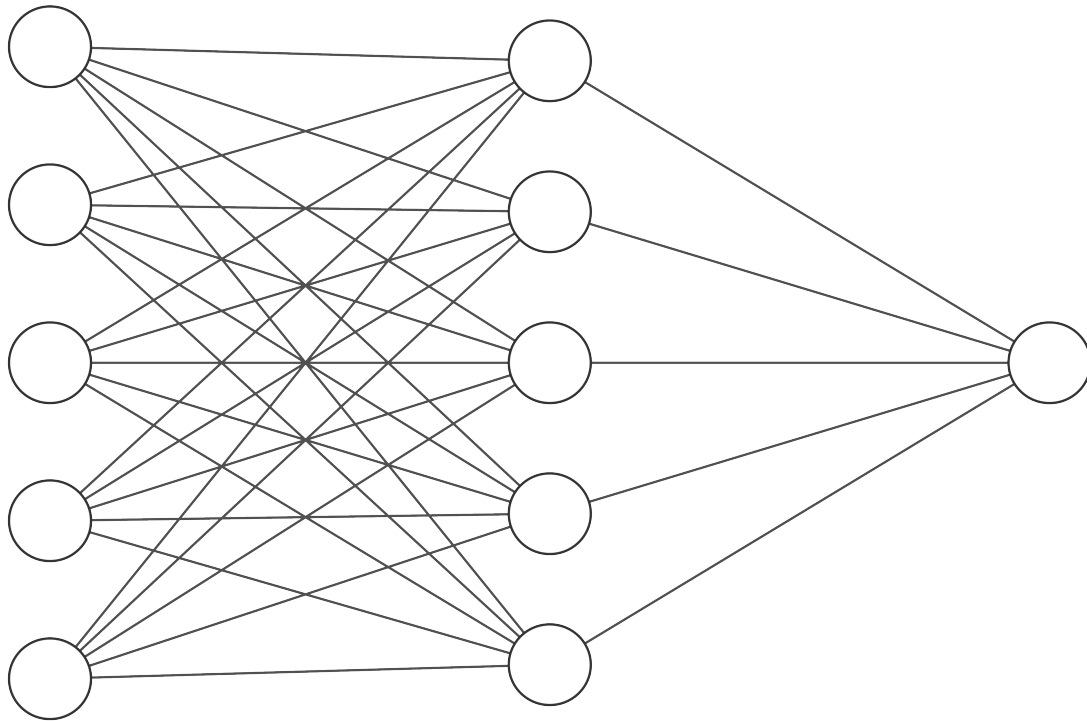
Input Layer   Hidden Layer   Output Layer

Figure 2.7: **Fully connected neural network architecture** An example architecture of a fully connected neural network. Each circle represents a node with inputs, an activation function, and singular output. Each line represents a weight connection between nodes.

architecture, known as convolutional neural networks solve this issue by reusing weights for feature extraction.

Convolutional networks

Convolutional neural networks (CNN) are most commonly applied to image based problems. Some of the advantages of CNNs are the smaller number of parameter compared to a FCN, as well as their applicability to a wide range of data. CNN works by learning features from an image, much like an edge detector. Whereas an edge detector convolves a filter or kernel with fixed weights across the input image, a CNN is able to learn the best kernel weights for each kernel through back propagation. The convolution operation is demonstrated in figure 2.8, where the kernel convolves, or slides across the input image. Note that here the filter only contains one channel, but in real applications there are multiple filters for each convolutional stage, each with their own weights.

At each step, the corresponding weight multiplies the input pixel, and is summed with the other pixels contained within the kernel. This summation is the output of the convolution, which is a singular value. This process of sliding the kernel across the image continues until all edges have been reached. We can see that the output image has a different dimension than the input image. The output dimensionality is dependent on the input image, kernel size, and how far the kernel slides in each step. A deep CNN combines multiple convolutional layers, usually in sequence, along with other layers such as pooling layers. The CNN acts as a generalized feature extraction, where the final outputs are typically fed into a FCN for final prediction. Figure 2.9 is an example of a simple CNN which contains both convolutional and pooling layers that feed into a dense fully connected network and consequently a final predicted output.
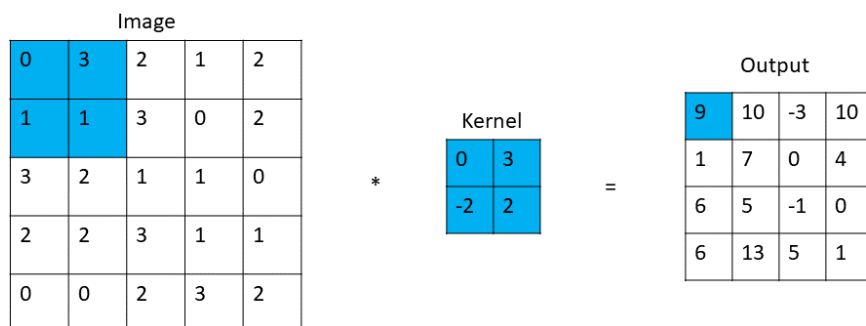
Figure 2.8: **Convolutional kernel output** Example output of a 2x2 kernel convolved over a 5x5 input image. The blues highlights a single calculation between the kernel and corresponding pixel, resulting in a single pixel output to a new image.
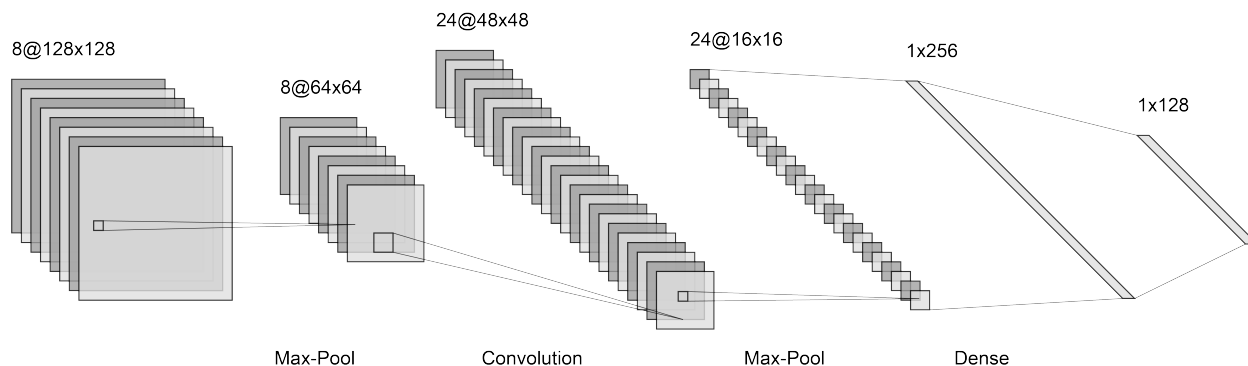
Figure 2.9: **Sample CNN architecture** Convolutional architecture containing max-pool and convolutional layers of different kernel sizes and depth. The CNN layers are fed into a FCN layer for final classification. [2]

28

Recurrent Networks

The last type of network we will cover are known as recurrent neural networks (RNN). RNNs are used for sequence based tasks, where a series of data points is input at each time step. RNN are capable of learning temporal relations across a sequence. The ordering of the input sequence is important for an RNN, while for a FCN the input sequence does not change the output. Applications for RNNs include natural language processing and time-series forecasting, such as the prediction of base-pair in nanopore sequencing. The construction of an RNN can be seen in figure 2.10. Figure 2.10 shows two different representations of the architecture, fold and unfolded. The unfolded RNN illustrates that at each time step, a node accepts both the current input data of the sequence, along with some output from the previous node. Depending on the task, the RNN could have an output at each time step, or a single output at the end of the sequence. For the work in this thesis, we utilize RNNs for singe cell classification, and therefore have a single output at the end of the sequence. The activation output of each node in an RNN takes the following form:

$$a^t = g1(W_{aa}a^{t-1} + W_{ax}x^t + b) \tag{2.29}$$

We can see that the activation function of an RNN has different weights for the input data and the activation from the previous data. These weights are shared across time, allowing the models to use sequences of arbitrary lengths. Sequences of longer lengths however suffer from small gradients during back propagation, resulting in poor model convergence. Furthermore, information from older data in the sequences tend to get lost and the basic RNN achitecture struggles with long term dependencies as a result. Other RNN models have been introduced such as Long Short Term Memory (LSTM) and Gated Recurrent Networks (GRU) which attempt to solve the problem of vanishing gradients and long term memory by introducing
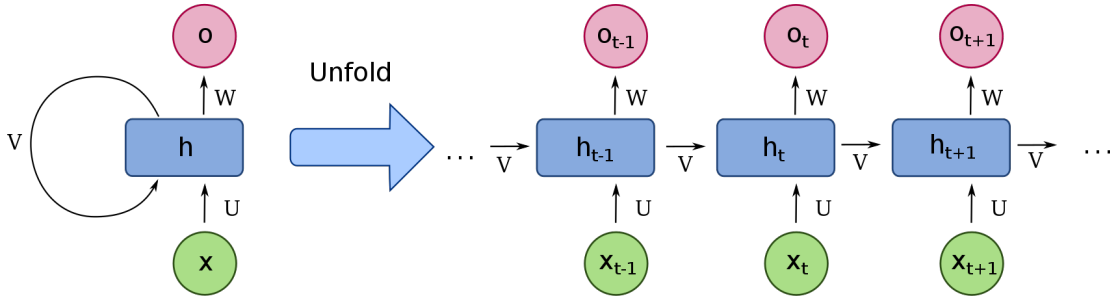
Figure 2.10: **Basic RNN architecture** Example model architecture and data flow in RNN. The unfolded representation demonstrates at each time step a gate receives an input feature, along with the activation from the previous state, and produces an output.[1]

further gates such as update, relevance and forget gates. We will demonstrate the use of a combination of both CNN and LSTM architure in this work for classification of single cells based on shape.

2.3   Unsupervised Learning

2.3.1   Gaussian Mixture Models for clustering

As opposed to supervised learning, where one is given data with ground truth labels, unsupervised learning attempts to find hidden structure, or clusters, within the data without any ground truth labels. Unsupervised learning is inherently a different task from supervised learning, whereas supervised learning can be thought of as finding the optimal decision boundaries between classes; unsupervised learning wants to find which data points are most 'similar'. Common traditional clustering algorithms include but not limited to k-means, spectral clustering, hierarchical clustering, and Gaussian-mixture models (GMM), among others. In this work, we will utilize GMMs for both the classification of bacteria by shape and mammalian cells based on their mechanical properties. Here, we will briefly discuss the

GMM algorithm and its advantages and disadvantages vs other algorithms like k-means.

Perhaps the best way to understand GMMs is to first understand the k-means algorithm. The k-means algorithm works in a two step process; first you measure the distance, often euclidean, of each data point to each cluster mean, and assign a label to that data point based upon the cluster with the minimum distance. Once each data point is assigned a cluster. The next step comprises of recalculating the cluster mean for each of the newly labeled samples. This process iterates until convergence is reached. To begin the algorithm, one must pick both number of clusters and a starting point for each cluster mean. The choices for cluster initiation can be random, or be chosen through certain criteria such as assigning points in the data set as the initial cluster centroids. K-means assumes that the clusters have equal variance in all direction and that each cluster has the same probability of assignment. GMMs address these drawbacks by fitting a Gaussian distribution to each cluster, and weighting the probability of assignment.

In the case of multivariate data, which is often the case for classification tasks, the multivariate Gaussian distribution is given by:

$$N(x; \mu, \sigma) = \frac{1}{(2\pi)^2} |\Sigma|^{-\frac{1}{2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)} \tag{2.30}$$

Σ and μ are the covariance matrix and vector of means, respectively. As with k-means, the number of clusters is picked a priori, and for each cluster there are separate covariance matrix and mean vector. By including the covariance in the model fitting, GMMs can find clusters which may contain different variances for each feature. In order to find the best fit GMM to the given data, the expectation-maximization algorithm (EM) is used.

Similar to k-means, the EM algorithm occurs in two steps. The first step (expectation), is

to calculate the probability that a given data point belongs to a specific cluster, where each cluster is denoted by c, and the total probability adds to 1. This probability is given by:

$$r_{ic} = \frac{\pi_c N(x_i; \mu_c, \Sigma_c)}{\sum_{c'} \pi_{c'} N(x_i; \mu_{c'}, \Sigma_{c'})} \tag{2.31}$$

where π_c is the 'size' of the cluster and is used to weight the assignment probability. The next step is to update model weights π_c, μ_c, Σ_c (maximization).

$$\pi_c = \frac{m_c}{m} \text{ where } m_c = \sum_i r_{ic}$$

$$\mu_c = \frac{1}{m_c} \sum_i r_{ic} x_i \tag{2.32}$$

$$\Sigma_c = \sum_i r_{ic}(x_i - \mu_c)^T (x_i - \mu_c)$$

GMMs are different from k-means in that produce a soft prediction for the assignment of each data point. Whereas k-means would define a hard label, GMMs produce a probability of a point belonging to each cluster. The EM algorithm is performed iteratively, and the convergence of the EM algorithm can be determined by looking at the log-likelihood function of the GMM.

Chapter 3

Deep Learning Assisted Mechanotyping of Individual Cells Through Repeated Deformations and Relaxations in Undulating Channels

3.1 Introduction

Mechanical properties of cells such as the ability to deform when an external force is applied, are directly linked to the structure of a cell's cytoskeleton. Changes in cytoskeleton have been correlated with cellular differentiation[11], malignant transformation[12], formation of biofilms[13] and even COVID-19 pathology[14]. Consequently, probing mechanical properties offers a label free method to learn about a cell's state such as its homeostasis or pathological conditions.

A multitude of techniques have been reported to probe mechanical properties of cells. Conventional methods include atomic force microscopy [15, 16], optical tweezers[17] and micropipette aspiration[18]. While these approaches are able to accurately measure mechanical properties of single cells, their throughput of roughly 1-10 cell(s) per minute is significantly slower than the ~10,000 cells per second achieved by most flow cytometers, which is needed to determine heterogeneities in cell populations and for single cell classification.

In recent years, various microfluidic based methods have been shown to measure the deformability of cells with significantly improved throughput, closing the gap between flow cytometry and mechanical phenotyping methods. In one class of methods, cells are squeezed through constrictions smaller than the cell's diameter[19, 20, 21, 22]. The passage time of such strongly deformed cells has been shown to depend on the cells' mechanical properties, with a faster passage corresponding to more deformable cells. This approach was applied to different cell lines, including cancer cells, stem cells and red blood cells [23, 21, 24, 25, 26, 27, 28, 29, 30, 31]. The transit through the channel is measured by recording electrical impedance or optical signals that enable one to relate parameters like transit time to the deformability, and in certain cases, measure quantities like Young's modulus[32]. However, often with these methods it is difficult to disentangle deformation and surface friction.

Another class of microfluidic approaches utilizes channels that are wider than the cells to be

analyzed, where the passing cells are subjected to hydrodynamic forces, and deformations of individual cells are probed optically. Two notable approaches include extensional deformability cytometry (xDC)[33] and real-time deformability cytometry (RT-DC)[34]. In xDC, a cross-channel is used to deform cells at ultra fast flow rates reaching $\sim 1000\,\mu\text{L min}^{-1}$. xDC has been used to classify malignant pleural effusions, differentiate multiple stem cells, as well as identify transitions in the cell cycle. The authors showed that deformation kinetics, among other features, are important in the classification of induced pluripotent stem cells (iPSCs)[35, 36]. Utilizing their full set of rheological and morphological features, Masaeli et. al. demonstrated the ability to classify between iPSCs and differentiated iPSCs with accuracies up to 95% using support vector machines (SVM)[36]. RT-DC, which is based on a straight, narrow microfluidic channel, operates at significantly lower flow rates than xDC ($\sim 1\,\mu\text{L min}^{-1}$), and induces constant shear stresses allowing cells to reach a steady state deformation. RT-DC quantifies deformability using steady-state images of the deformed cells captured at the end of the microfluidic constriction and is linked to a physical model to calculate Young's modulus[37]. RT-DC has also been recently extended to probe the cells' deformation kinetics as they approach steady state[38]. RT-DC has been shown to classify reticulocytes from mature red blood cells with an unsupervised approach with a mean accuracy of $\sim 74\%$[39].

While these approaches have shown great promise and success for a variety of problems, the classification of the cell populations rely heavily on morphological and size based features. While these features are useful in practical applications, the discriminative power of deformability cytometry on cell populations which differ in mechanical properties alone has not been fully explored. Additionally, the use of deep learning models, particularly convolutional and sequence-based models, provide higher classification potential than traditional machine learning models[40, 41]. In this manuscript, we propose to enhance classification accuracy by maximizing mechanotyping information through subjecting individual cells to repeated deformations and relaxations by hydrodynamic forces as well as through applica-

tion of deep learning methods. As the videos are recorded with the time resolution of at least 11,000 frames per second, we can probe the deformation dynamics with high precision. The dynamic observation reveals quantitative insights into the deformation/relaxation processes and provides a comprehensive mechanical fingerprint of each cell. The channel contains a cavity flanked by two narrower regions[42], with widths that at any position are wider than the cells to be analyzed. Our technique operates at a throughput comparable to the throughput of RT-DC that enables observations of hundreds of cells per minute and yet probe the cells sufficiently slowly to observe cytoskeletal changes [43]. Inducing repeated deformation and relaxation by hydrodynamic forces is a natural progression on the way to increase fidelity of cells characterization and classification based solely on mechanotyping[44, 45].

HL60 cells before and after treatment with either Cytochalasin D (CytoD) or Nocodazole (Noco) were used as a model system to probe classification potential of our method. Both chemicals were found to perturb HL60 deformability such that cytoD-treated HL60 cells (HL60d) were more and Noco-treated HL60 cells (HL60n) were less deformable than untreated HL60. Most importantly, cells belonging to the three sub-populations, HL60, HL60d, and HL60n, have the same average size, thus enabling us to test our classification strategy based on mechanical properties alone. Using these sub-populations subjected our method to a very challenging test, since classification is often aided by cells' size[35, 46].

The analysis of the recordings was supported by machine learning approaches with gradually increased levels of expressive power. We show that the mechanotyping dynamic features our method provides enable a significant increase of the classification accuracy of the HL60 populations when compared to using any single feature alone. We also calculated Shapley values, a technique known from economic game theory, to probe which deformability parameters contributed most to the classification accuracy[47, 48]. A significant improvement in classification accuracy was further observed when the time series of deformations was used as an input into deep learning models such as recurrent neural networks (RNNs). Our deep

36

learning methods utilizing time-based sequences of features showed an increase in classification accuracy to 90%, from the 75% accuracy we observed with the random forest (RF). This increase in accuracy translates to an order of magnitude increase in the potential ability to enrich a sample for a rare population of cells (Supplementary Material Note 1). Most importantly, a convolutional neural network (CNN) was used in conjunction with an RNN to utilize sequences of binary masks as input features. We also discuss the trade-off between interpretability achieved with more traditional machine learning approaches and accuracy that is offered by deep learning. Not only has such a comparison not been discussed before, deep learning has not been applied to deformability cytometry for classification based solely on mechanical properties.

3.2   Methods

Cell Culture

HL60 (ATCC CCL-240) cells were cultured in suspension with Iscove's Modified Dulbecco's Medium (IMDM) supplemented with 10%(v/v) FBS (Fisher brand) and 1% (v/v) penicillin-streptomycin and maintained in an incubator at 37°C with 5% CO_2. Cells were passaged through dilution every 2-3 days to maintain a density between 10^5 and 10^6 cells mL^{-1}. Prior to experiments, cells were centrifuged at 112 relative centrifugal force (RCF) for 5 minutes and resuspended to concentrations of 2-3x10^6 cells mL^{-1} in PBS with 1% (w/v) methylcellulose (Spectrum 4,000CP).

To create sub-populations of HL60 cells with perturbed actin and microtubule networks, cells were incubated for 10 minutes in 1 μM cytochalasin D (cytoD) (Sigma) or for 1 hour in 10 μM nocodazole (Noco) which had been diluted 10x from stock solution with dimethyl sulfoxide (DMSO)[49]. Cells were spun down at 112 RCF for 5 minutes and resuspended in
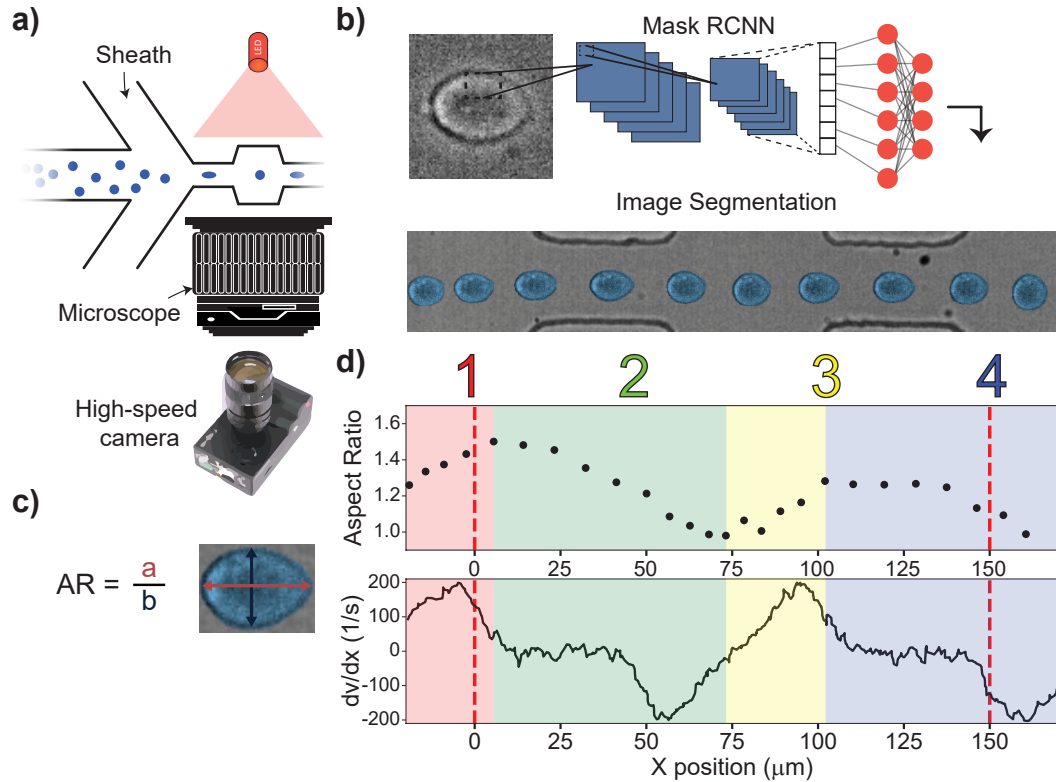
Figure 3.1: **Principles of repeated mechanotyping.** a) Channel design utilizing sheath flow, a high-powered LED and a microscope. The microfluidic channel that enables characterization and classification contains a cavity placed between two narrow zones. b) Data is captured by a high-speed camera, creating videos at 11k fps. Cell borders are detected and fit using Mask-RCNN. c) The cell deformation, AR, was quantified as the ratio of two axes of an ellipse that approximates the cell's shape. (d) (Top) The aspect ratio versus position relative to channel entrance of a single cell as it passes through the channel. (Bottom) COM-SOL simulation showing the derivative of velocity vs channel position, which is proportional to the shear stress. Region 1 (R1) and Region 3 (R3), denoted by red and yellow regions, are where the cells undergo deformation. Region 2 (R2) and Region 4 (R2), denoted by green and blue regions, are where the cells undergo relaxation.

1% (w/v) methylcellulose solution.

Channel Preparation and Data Acquisition

Microfluidic channels were prepared with a master mold using standard photolithography with negative SU-8 photoresist (Kayaku Advanced Materials Inc.). The channel used in the experiments was 150 µm long, with three equal length sub-regions (50 µm) with widths of 25 µm, 50 µm, and 25 µm. The height of the device was 20 µm along the whole length. The full chip geometry can be found in Supplementary Material (Note 2). 184-Sylgard polydimethyl siloxane (PDMS) was pipetted over the SU-8 master and baked for ∼4 hours at 75°C. Channel inlets and outlets were created with a 1.5 mm biopsy punch. PDMS channels were cleaned and dried with isopropyl alcohol, methanol and water before being bonded to a glass cover slip using a corona discharge wand (ETP). Methanol was used last before drying due to its low boiling point to ensure no residual alcohol remained in the device. Bonded PDMS/glass samples were heated for an additional hour at 90 °C to promote further adhesion.

Cells were suspended in methylcellulose solution to prevent settling to the bottom of the container. The solution was then pumped through a microfluidic channel using a Genie-plus syringe pump (Kent Scientific) at a rate of 1 µL min⁻¹. Cells were focused laterally in the channel using a sheath flow geometry with a flow rate of 2 µL min⁻¹. The sheath and core flows were allowed to equilibrate for 10 minutes before data was taken. Cells were illuminated using a high-powered Red Amber (613 nm) 36 Watt LED array (PT-121-RAX Luminus, Inc.) and imaged at 10x magnification in brightfield on an inverted microscope. A Chronos 1.4 high-speed camera (Krontech) imaged passing cells at a frame rate of ∼11,000 fps with 1 µs exposure time and a resolution of 880x140 to encapsulate the full channel length. The size of each pixel is 0.26 µm/pixel. Maximum blurring induced by cell movement is ∼ 0.1 µm. 8

second videos were recorded and saved which require ∼15 minutes to offload from camera memory. In order to ensure model generalizability, in total, ∼3,500 cell trajectories were recorded and analyzed. The data presented were collected over 10 technical replicates of HL60 (4 biological replicates), 8 technical replicates of HL60d (4 biological replicates), and 6 technical replicas of HL60n (2 biological replicates).

Detection, Segmentation, and Tracking

A convolutional neural network (CNN) was trained to identify frames containing cells to reduce the computational time of processing the large number of frames generated by the high-speed camera. Frames labeled as containing cells were then passed to a segmentation MASK-RCNN network that was used to segment cells from images and fit masks. MASK-RCNN can learn to accurately segment cells with differing focusing conditions and is able to segment multiple cells in a single frame. The Matterport implementation[50] was used with Tensorflow 2.2 due to its wide code availability and its support for Tensorflow. The network was trained using a NVIDIA 1070TI on a hand labeled dataset of ∼300 images of HL60 cells across multiple independent experiments using VGG Image annotator 1.0. The network was initialized with weights from the COCO dataset and the architecture was modified in order to increase the output resolution of the predicted masks. The training schedule consisted of training head layers for 20 epochs at a learning rate (LR) of 10^{-3}, 50 epochs training 4+ layers at LR/10, and 50 epochs training all layers at LR/10. Training curves and further details can be found in the Supplementary Material (Note 3). A custom tracking algorithm was written in python to follow individual cell trajectories.

Calculation of Features

Detected events with impossible trajectories (i.e. no event will begin in the middle of the channel) were discarded. To ensure the fits are accurate, a convex hull was fit to the cell mask and events were filtered out where a single frame had a ratio >1.1 between the original and convex hull fit. Detected particles with radii equal to three standard deviations from the mean were not included in analysis, as they often contained cell/channel debris, or clumps of multiple cells. Ellipses were fit to the detected masks and the aspect ratio of the ellipse was used to describe the cell deformation, AR. A full description of deformation parameters can be found in Supplementary Material (Note 4).

Comsol Simulation

A finite element simulation was conducted using Comsol Multiphysics 5.3 to simulate the velocities and stresses experienced in the undulating channel. A simplified 3-D model of the channel was modeled in the laminar flow module using the Navier-Stokes equation with creep flow in steady-state. An extra fine meshing was chosen for the area near the narrow section of the channel and normal meshing was chosen for the reservoir.

Machine learning model training

Of all the cell trajectories recorded, 3,552 were suitable for classification. They were distributed as 1114 trajectories of HL60, 1122 trajectories of HL60d and 1316 trajectories of HL60n.

The RF model and SVM were created and trained using python 3.6 and scikit-learn. The data were standardized, shuffled, and split according to the following ratio: 70:15:15 for

train, validation and test respectively.

The GRU and the CNN-GRU models were created and trained in python 3.6 using Keras and Tensorflow 2.2 libraries. The hyperparameters were optimized over 175 epochs using the test set. Model performance was assessed using a hold-out test set, in addition to 5-fold cross-validation. Additional details on both model architectures can be found in the Supplementary Material (Notes 5,6).

Data Preprocessing for Sequential Models

Sequences of aspect ratio, perimeter, deformability and area were selected as inputs for the GRU model. Data previously filtered were first aligned so that all sequences started and ended at a true x-position of -30 μm and 170 μm respectively. Data were then padded to length 50 and were shuffled and split according to the following ratio: 70:15:15 for train, validation and test respectively.

For the sequential models using image data, CNN-GRU, masks were first cropped to 96x96. An ellipse fitted to the mask shape was created using scikit-image and added to the second image channel. Each sequence of these two-channel images were padded to length 50. The data were shuffled and split according to the following ratio: 70:15:15 for train, validation and test respectively.

Code Availability

The python scripts implemented for data processing and training machine learning models are available at: https://github.com/siwylab/time-series-dc.

3.3 Results and Discussion

3.3.1 Channel Design and Data Acquisition

We designed a microfluidic channel that subjects individual cells to repeated compressions and relaxations. The multiple deformations of the cells are induced by the channel shape, specifically the presence of a cavity and two narrow zones. The varying channel width is expected to create inhomogeneous velocity profiles leading to temporal changes in the cells' shape[51]. To test this channel design for mechanotyping, we pumped a suspension of HL60 cells (ATCC-240) in methylcellulose (1% w/v) solution through a channel of consecutive constrictions of length equal to 50 µm (total length of 150 µm) and widths of 25 µm, 50 µm, and 25 µm at a flow rate of 1 µL min^{-1}. The more viscous methylcellulose solution induces larger deformations than buffer solution and prevents sedimentation of cells. Sheath flow focusing was used to ensure that all cells passed through the channel along its center axis and experienced the same forces. The microfluidic chip was placed on a 10x magnification inverted microscope. To enable probing dynamics of cells' shape at this high flow rate, we customized the microscope to allow for high speed imaging by replacing the light source with a high-powered LED array and accompanying 3D printed mount (Figure 3.1a). A high-speed camera was adapted to fit the microscope, and videos of translocating cells were recorded at 11,000 frames per second, resulting in ∼30 data points over 150 µm channel and revealing the shape dynamics as a cell moves along the channel axis.

After videos had been collected, the data was processed with a number of python scripts (Supplementary Material Note 7). The first script encompasses a lightweight CNN that filters out empty frames, and reduces the amount of data to be processed by ∼80%. Next we employed a modified architecture of the Matterport[50] implementation of Mask-RCNN[52], which is used to detect cell regions and their associated mask. Our version of Mask-RCNN is trained on 300 hand-labeled images of cells with varying levels of focus (Supplementary
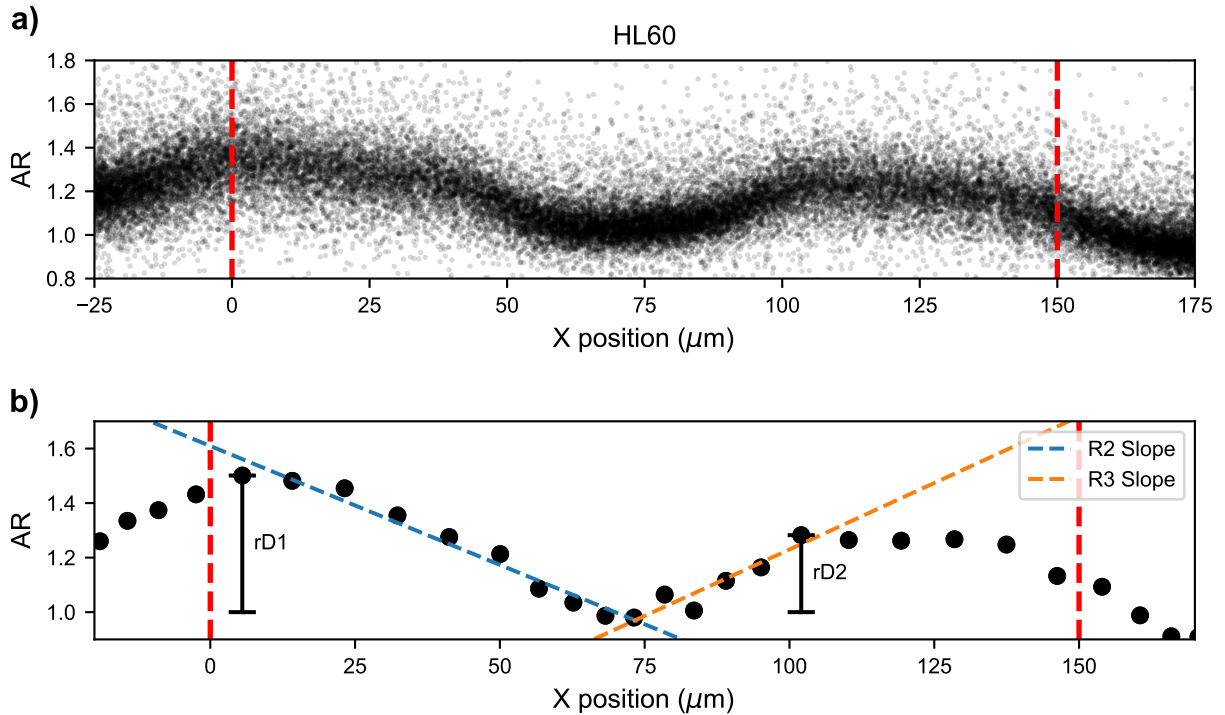
Figure 3.2: **Single cell deformation traces.** Deformation dynamics are shown for single cells translocating through the channel. The aspect ratio is determined by the best fit ellipse to the cell mask. Deformation is calculated by the difference between the aspect ratio at a given point and the minimum aspect ratio in the cavity. The x-position along the channel axis is determined by the centroid of the mask. Cells experience a smaller maximum deformation in the second narrow region, as compared to the first narrow zone. Channel inlets and outlets are marked by red dotted line. Deformation and relaxation occur twice within the channel. a) Full population of single cell traces of aspect ratio versus position for untreated HL60 cells. b) Single cell example of parameters that are determined: relative maximum deformations (rD1 and rD2) in the two narrow zones as well as relaxation and deformation slopes (R2 slope and R3 slope).

44

Material Note 3). The masks detected by this network are then processed using a custom tracking algorithm to find trajectories of individual cells. Figure 3.1b shows subsequent snapshots of one cell as it passes through the channel. MASK-RCNN enables us to track multiple cells in the same frame, allowing a high density of cells ($5x10^6$ cells mL $^{-1}$) to be used in the experiments. We found that the undulating channel design results in complex dynamics of the cell's shape, and leads to regions of differing deformations. Specifically, the cell underwent a strong deformation at the entrance of the channel and in the first narrow constriction; the cell then relaxed to a spherical shape in the cavity, and started to deform again when approaching the second narrow constriction. As seen in Figure 3.1d, we define the regions where the cell undergoes deformation as region 1 (R1) and region 3 (R3). The regions where the cell undergoes relaxation are denoted as region 2 (R2) and region 4 (R4).

We find that the shape of the deformed cells in our device can best be described as an ellipse where the deformation (D) is defined as the aspect ratio (AR) of the two axes of the ellipse: the axis parallel to the channel axis and the perpendicular axis (Figure 3.1c). The relative deformability (rD) is the difference between D at a given point in the trajectory, and D in the cavity, where the cells undergo no deformation. rD = 0 corresponds to lack of deformation relative to unperturbed cell shape, whereas rD>0 corresponds to an extension along the channel axis. Figure 3.1d summarizes how the magnitude of AR evolves as the cell shown in Figure 3.1b passes through the channel. In order to qualitatively understand the deformation trace, we performed a computational fluid dynamics simulation with COMSOL multiphysics in a cell-free undulating channel using the Navier-Stokes equations with creep flow at experimental flow rates. Shear stress in different parts of the channel can be analyzed through the derivative of velocity in the center of the channel with respect to axial position shown in Figure 3.1d. The cells experience a large velocity gradient at the entrance of the channel, leading to large stresses and deformations.

For the single cell trace shown in Figure 3.1d, we observe a peak deformation of AR =

1.49 in the first region (R1), marked in red. The velocity gradient then reaches a steady state value within the first narrow constriction, (10 μm - 50 μm) and the cell deformation decreases. Before the cell can reach a steady state deformed shape, it enters the cavity where the velocity gradient begins to decrease and then again rapidly increases. The cell returns to a spherical shape, AR = 1, where $dv/dx = 0$, which occurs at $\sim$75 μm. The cell then begins to deform again due to the velocity gradient at the entrance of the second narrow constriction, reaching a second peak deformation of AR = 1.30 in region three (R3). The maximum deformation observed in the second constriction is lower than the maximum deformation measured in the first constriction. We believe this is because the cell is subjected to positive shear stresses over a relatively short time and distance when transitioning from the cavity to the second narrow region, as compared to the transition from the bulk channel to first narrow constriction entrance. The distance from the middle cavity, where the shear stress is zero, to the position of peak deformation (and maximum positive shear stress) in the second narrow region is only 25 μm, whereas at the initial inlet the cell begins deforming from positive shear stresses $\sim$50 μm away from the entrance. Figure 3.2a shows values of AR for $\sim$1200 HL60 cells examined in the same conditions. The same trend for all cells has been observed: the cells reached the maximum deformations in the first narrow zone, relaxed to a sphere in the cavity, and underwent another deformation in the second constriction.

Taking advantage of the time-series of cells' positions and shapes, we also quantified the dynamics of deformation and relaxation. To this end, we used a linear model to fit the trace from the peak deformation in the first narrow zone to the position where the cell relaxes to a spherical shape in the cavity (R2), resulting in a slope of -8.71×10^{-3} μm^{-1}. The slope describes relaxation dynamics, and we refer to it as R2 slope. A similar analysis can be performed for the cell entering the second narrow zone by fitting a line between the spherical shape in the cavity to the maximum deformation in the second narrow zone, obtaining a value of 9.75×10^{-3} μm^{-1}. This slope, called R3 slope, describes the deformation dynamics. Figure 3.2b shows example fits of rD, R2 and R3 slopes to a trace of AR evolution for a

46

single cell passing through a channel.

3.3.2 Application of the undulating channel to probe perturbation of actin and microtubule networks

In order to evaluate the sensitivity of our method to detect cytoskeletal perturbations, we continued the experiments with HL60 cells and treated them with cytochalasin D (cytoD) and Nocodazole (Noco). These two chemicals have been previously used to create model populations of HL60 that allowed researchers to evaluate performance of different mechanotyping techniques[34, 33]. CytoD disrupts actin polymerization[53], and has been previously shown to increase deformability of HL60[34, 49, 32]. Noco targets microtubules causing rapid filament decomposition. For HL60 cells, Noco has previously been found to decrease the ability to deform [49]. Perturbing different components of the cytoskeleton allowed us to test whether the deformation/relaxation processes from our channels and the chosen flow rate are able to measure changes in whole cell deformation after actin and microtubules disturbance [43]. The three populations of HL60 cells, untreated, CytoD treated and Noco treated, were suspended in methylcellulose solution, and separately passed through our microfluidic channel. After post-processing, passages of more than 1000 cells of each sub-population were captured. Recordings were taken over 10 technical replicates of HL60 (4 biological replicates), 8 technical replicates of HL60d (4 biological replicates), and 6 technical replicas of HL60n (2 biological replicates). The repeated experiments and the amount of cells measured show consistent trends and enable statistical analysis of the various measured parameters across the populations.

Figure 3.3a,b shows the rD in the two narrow zones, and the two slopes for the three populations (Figure 3.3c,d) of HL60 cells. The magnitudes of rD are consistently the highest for the cytoD treated population, which confirms that the method is sensitive to actin network per-
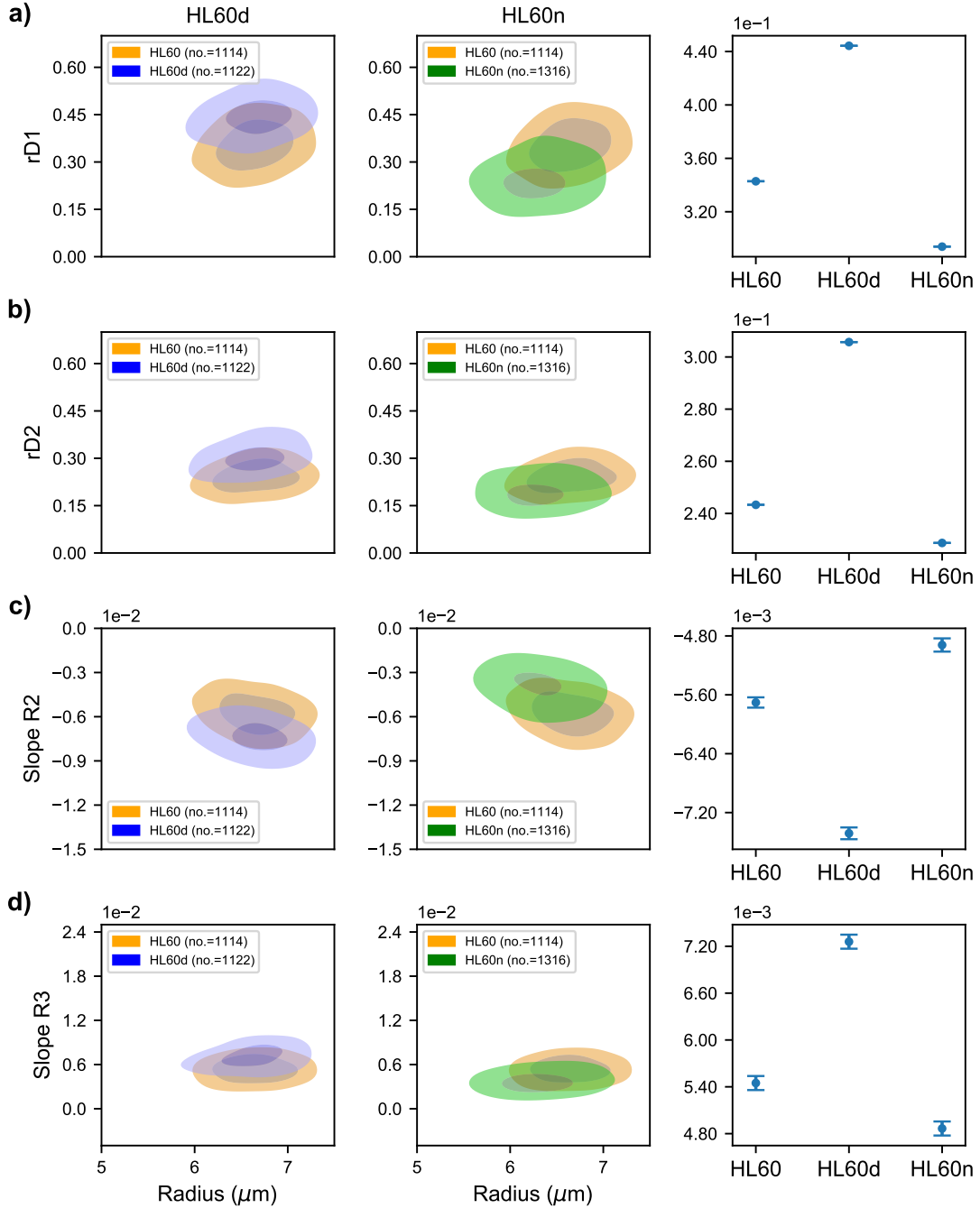
Figure 3.3: **Comparison of measured deformability features between untreated and treated HL60 cells** a) Contour plot of maximum rD in first narrow zone (R1) for untreated, cytoD treated, and HL60n cells. The outer contour represent 50% density and center contour represents 90% density. Mean of each population is reported where the reported error is standard error of the mean. b) Contour plots of maximum rD in second narrow region (R3). c) Linear fit slope from maximum deformation in first narrow region (R1) to relaxation to minimum deformation in cavity (R2). d) Linear fit slope from relaxed state in cavity (R2) until maximum deformation in second narrow region (R3).

turbations. The Noco treated populations exhibit the lowest mean rD, albeit with a larger variability than the other two populations. These findings are in agreement with earlier reports that used the same cell line and also observed increased (decreased) deformability of the cytoD (Noco) treated HL60 populations [49].

Interestingly, the R2 and R3 slopes that represent relaxation and deformation dynamics, respectively, are also the greatest for the cytoD-treated cells, suggesting that these cells are most responsive to external forces. In contrast, the slopes for the less deformable, Noco treated populations are characterized by significantly smaller magnitudes of both slopes than cytoD treated and untreated HL60 cells. Though the HL60n had a slower response to the shear stress, they relaxed to a spherical shape in the cavity, and deformed again in the second narrow region. It is important to note that the mean diameter of the three populations is nearly identical (Supplementary Material Note 8), confirming the cells experienced similar forces.

While the mean values of deformation are significantly different (Figure 3.3, third column), there is a significant overlap between the control HL60 and treated cell distributions (Figure 3.3, contour plots). This overlap makes it difficult to place an individual cell in one population or the other. We find that no single feature alone can provide accuracy significantly higher than 70% for single cell classification (Supplementary Material Note 4). For example, rD1 alone is able to classify between HL60 and HL60d populations at 72% accuracy using a logistic regression model, while the same parameter classifies HL60n with only 57% accuracy. The radius alone can be used to classify up to 56% and 59% for HL60d and HL60n, respectively, as expected from the nearly identical size of the populations. In the following sections we will show that in order to achieve discrimination between the two populations on a single cells basis, multiple features describing the cell deformation (Figure 3.3) must be considered simultaneously.

3.3.3   Feature extraction and machine learning model comparison

Our next goal was to utilize the complete mechanotyping fingerprint our method provides, to maximize classification accuracy. As mentioned above, the three populations chosen were nearly identical in size thus the differences in mechanical properties provide the only basis for classification. We first investigated the ability of traditional machine learning models to distinguish the two pairs of sub-populations, HL60 vs HL60d, and HL60 vs HL60n. To provide a baseline for classification potential, features were first manually extracted from the individual raw time series data as seen in Figure 3.2. A full list of extracted parameters can be found in the Supplementary Material (Note 4). These derived features were used to train a random forest classification model[54, 55, 56] (RF), which has been previously employed for classifying cytometry data[57]. The random forest models were implemented using scikit-learn and all relevant hyperparameters were optimized. Other models such as support vector machines (SVM) were tested and showed similar or less performance compared to RF (Supplementary Material Note 9). All models were trained across a mixture of multiple biological and technical replicates, and accuracy results were reported on a held on test set, in order to minimize bias.

We found that the trained RF model resulted in 75% classification accuracy for HL60 vs HL60d, and 71% classification accuracy for HL60 vs HL60n (Figure 3.4a,c. The lower accuracy of the model trained on HL60n can be attributed to the larger variation in cell radius and deformation of Noco treated cells. In the next step, we wanted to understand the impact of different mechanotyping features on the RF model's predictions. To this end, we utilized Shapley values, which were developed for coalitional game theory to inform how to fairly attribute success to the constituent parts[47]. Thus, Shapley values can help us understand which features are most informative for a single cell classification. In order to make use of this method, the SHAP python library[58] was employed in conjunction with the trained RF model to attribute overall model performance to individual features (Figure
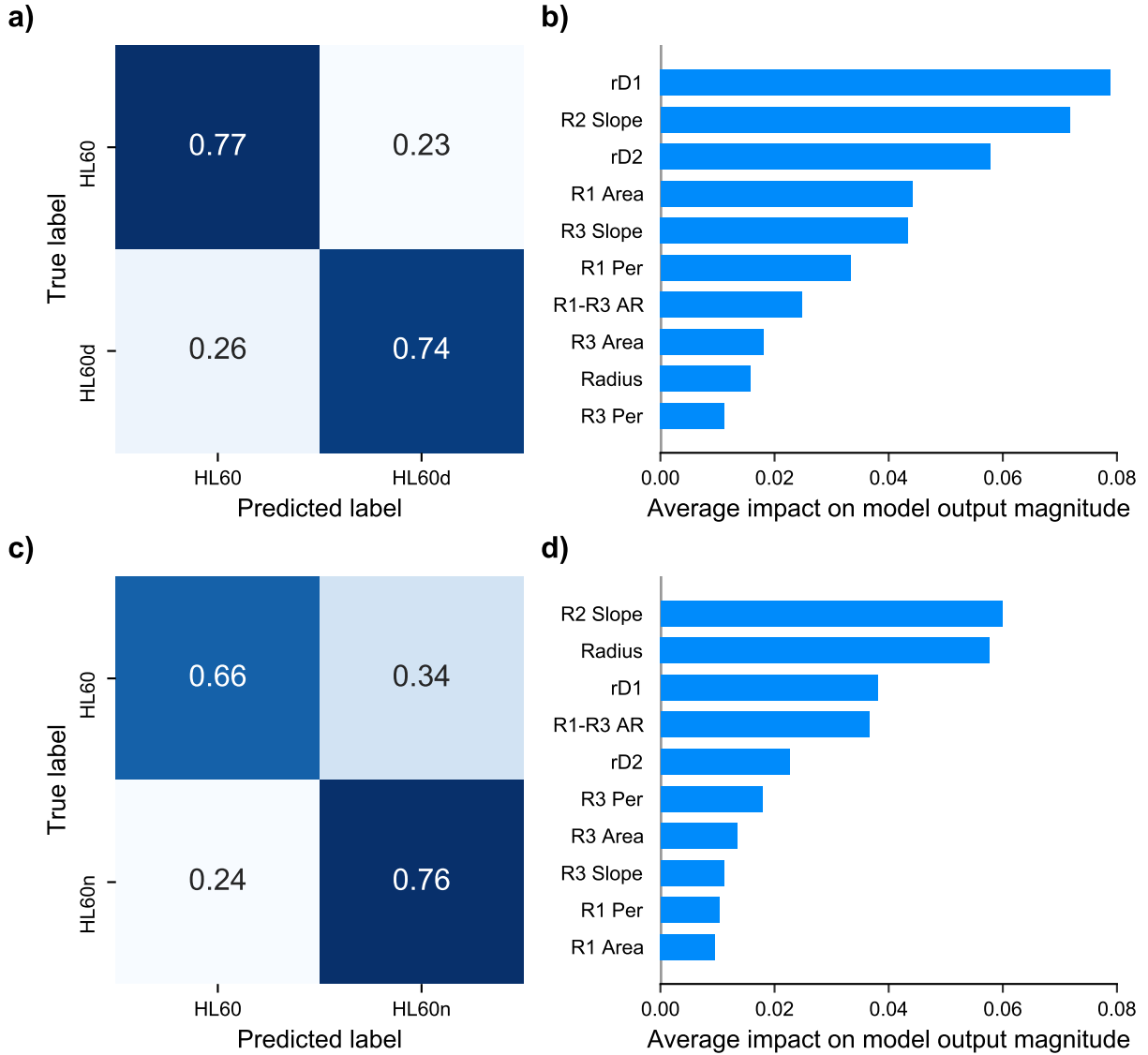
Figure 3.4: **Prediction results of random forest using derived mechanotyping features.** a) Confusion matrix of trained random forest predicting HL60 vs HL60d. The values are normalized by the true label count. Accuracy is equal to the average of diagonal. b) SHAP feature importance plot obtained using trained RF model for the HL60 vs HL60d classification. c) Confusion matrix for random forest trained on HL60 vs HL60n prediction. d) SHAP feature importance for HL60 vs HL60n.

3.4b). The values reported in Figure 4b and 4d represent the mean impact for an individual feature in determining cell type. From the Shapley values we find that the R2 slope and rD1 have the most weight in deciding classification between HL60 vs HL60d. Whereas for the HL60 vs HL60n case, we find the R2 slope to have the highest impact, followed by the radius. The introduction of Shapley values provides interpretability of the mechanotyping data and demonstrates that the region from peak AR to the relaxed state contains the most information for classification. The analysis also revealed the importance of another temporal feature, the R2 slope, in the classification (Figure 3.2b). Based on these observations, we hypothesized that classification could be further improved by incorporating the sequential nature of a cell's deformation into the model design. We then decided to explore deep learning approaches to create a model that can extract shape dynamics.

3.3.4   Deep learning for enhanced classification

RNNs are considered the optimal tools for handling sequential data[59] and are often used for translation and sequential prediction tasks. We chose RNNs since they can approximate a function describing the shape dynamics, which in our case is complex due to the repeated deformations and relaxations caused by the non-linear shear force (Figure 3.1c). In short, RNNs function by receiving a single input from the full sequence, processing it and feeding the output into a copy of itself along with the next time step. We implemented a variant of RNNs called the gated recurrent unit (GRU) and used the sequential time-series features such as aspect ratio, perimeter, and area as inputs to these models. The trained GRU displayed a moderate increase in classification performance, as seen in the confusion matrices in Figure 3.5b,c, resulting in a 4% accuracy increase for HL60 vs HL60d, and an 8% increase for HL60 vs HL60n. Another RNN variant, Long Short-Term Memory (LSTM), was tested but did not perform as well (Supplementary Material Note 10). GRU model architecture is presented in Supplementary Material (Note 5).
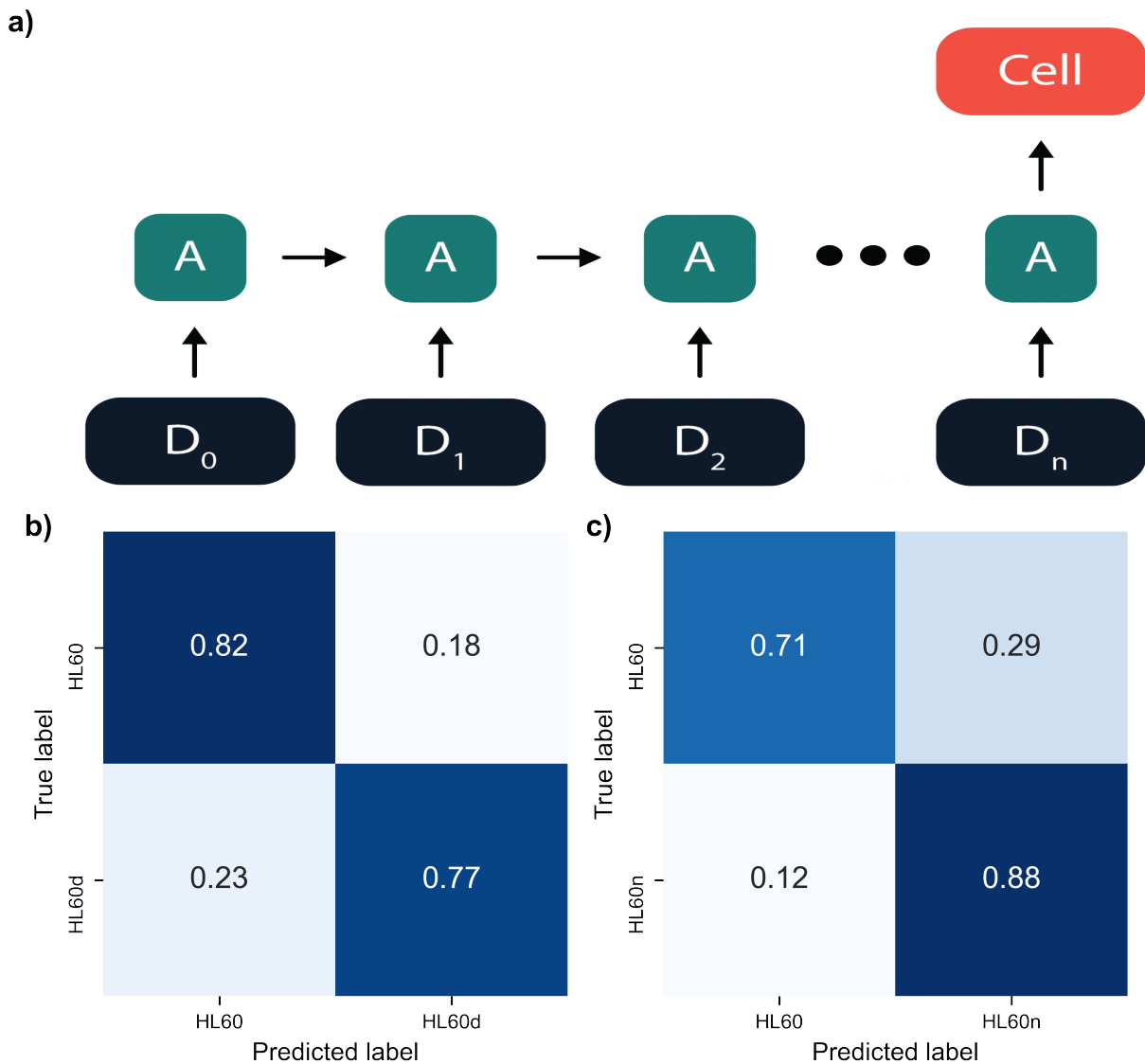
Figure 3.5: **Time series neural networks applied to mechanotyping features** a) Outline of recurrent neural network. Time series deformation data is used as input into GRUs. The output of the network predicts cell phenotype. b) Confusion matrix for RNN trained on HL60 vs HL60d. c) Confusion matrix for RNN trained on HL60 vs HL60n.

Since the application of RNNs yielded improved performance (Figure 3.5) compared to the RF models with manually derived features (Figure 3.4), we hypothesized that further improvement would require an approach that is not based on hand-selected features (or their combination). We therefore sought to improve accuracy by using the binary masks as inputs, obtained using our video-processing algorithm described above (see the sequence of

blue shaded regions in Figure 1). Note, that the sequence of the masks are the only inputs; the derived deformation values from the masks are not used here. To utilize the time-series of masks, we added a number of convolutional layers to the GRU model (CNN-GRU). Convolutional neural networks are known to be very well suited for image based classification tasks. A schematic diagram of the model is shown in Figure 3.6a.

While CNN-RNNs traditionally use raw frames as inputs, here our input complexity is drastically reduced since segmentation had already been performed. Consequently, the network only learned the shapes of the deformed cells, as opposed to differences in internal morphology or brightness contained in the raw images. Our CNN-GRU architecture was refined by searching different filter sizes for the CNN layers, changing the number of dense and GRU layers, as well as adjusting the dropout rates. The optimized model (Supplementary Material Note 6) shows superior performance of the CNN-GRU to both the RF and GRU models (Figure 3.6b, c). Namely, the CNN-GRU enabled an increase of accuracy by 11% to HL60 vs HL60d, and an increase of 6% to HL60 vs HL60n, resulting in final classification accuracies of 90% and 85%, respectively. To investigate the potential of over-fitting, 5-fold cross-validation was performed in addition to the use of a train, validation, and test split. The 5-fold cross-validation resulted in accuracies of $91.4\pm0.6\%$ and $83.7\pm0.9\%$, with similar results obtained for each training regime. The high validation set performance shows the richness of the deformability dynamics and their potential to aid in classification.
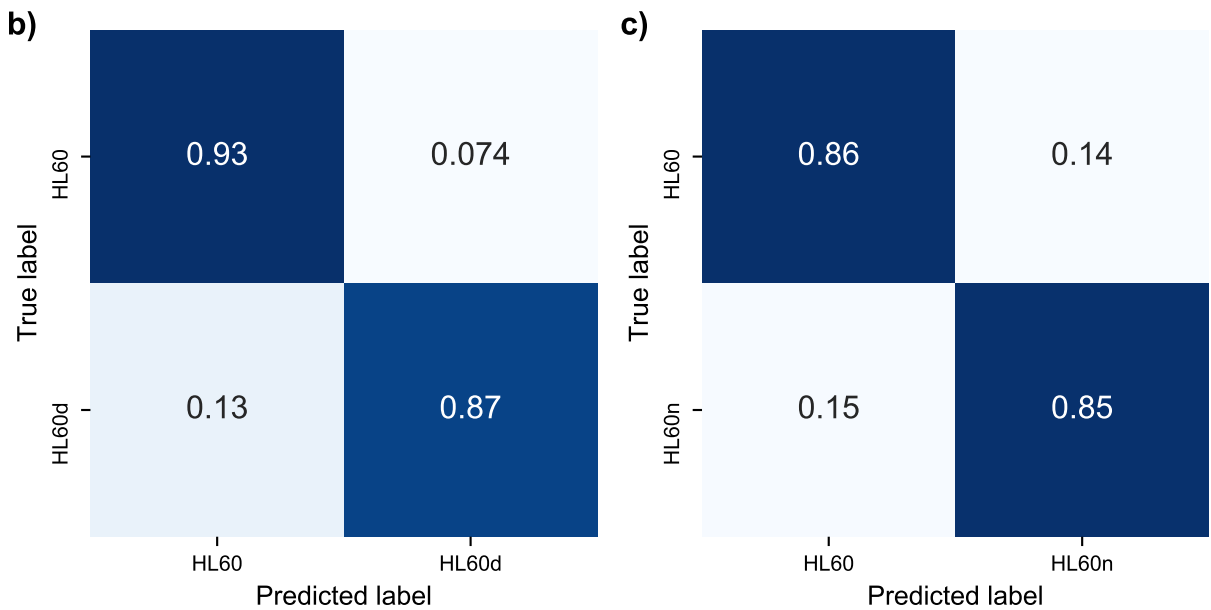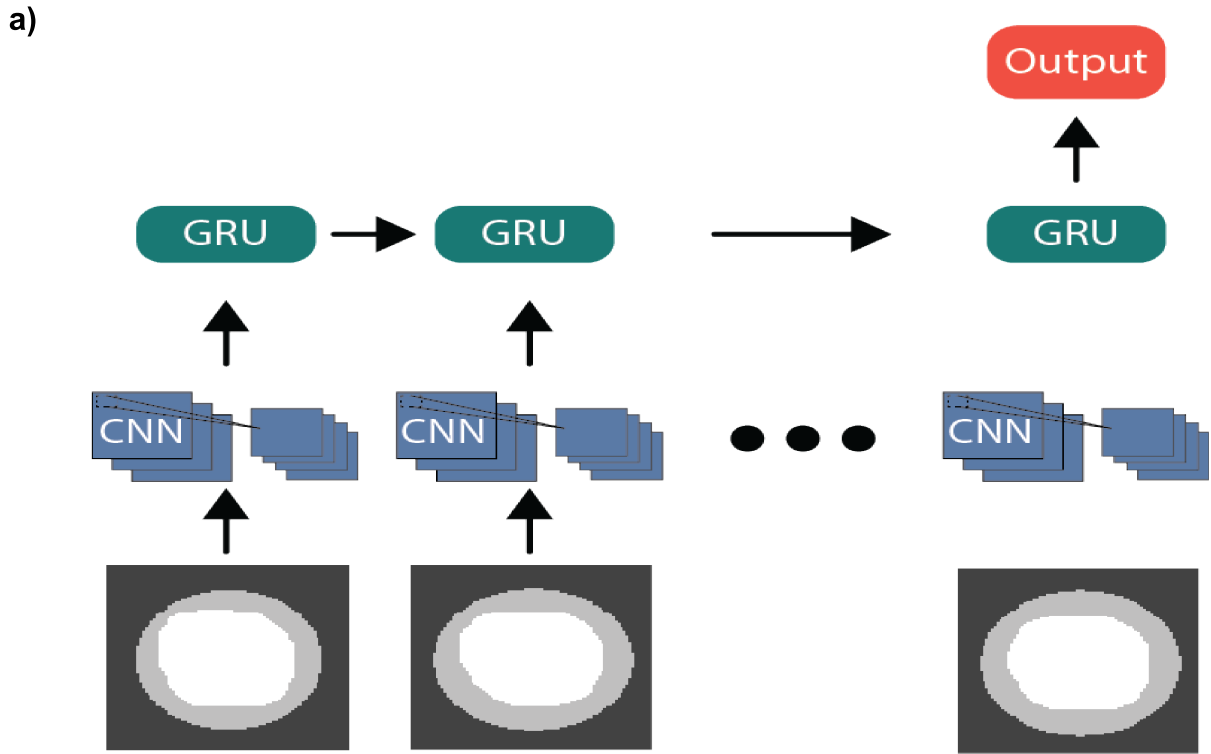
Figure 3.6: **Classification comparison using sequence of cell masks** a) General flow of CNN-GRU. Sequences of masks are padded and used as inputs. CNN and GRU layers use identical weights for each time step. b) Confusion matrix for HL60 vs HL60d. c) Confusion matrix for HL60 vs HL60n.

Finally, we wanted to confirm the importance of mechanotyping in classification of HL60 sub-populations, and asked whether these cells could be distinguished only through their morphological features, such as shape, prior to deformations. To this end, a CNN was trained to classify HL60 vs HL60d and HL60 vs HL60n sub-populations using masks from the channel cavity, where no induced deformations are present. The best test accuracy attainable was 65% for HL60 vs HL60d, indicating the poor classification potential of the undeformed shape (Supplementary Material Note 11). On the other hand, when the same analysis was performed for HL60 and HL60n, we saw a classification accuracy of 70%, indicating that the Noco treatment affected the cells' undeformed shape, relative to the control HL60 population. The modified morphology of HL60n could have also contributed to the large increase in accuracy when using the CNN-GRU. Combining mechanotyping and deformation dynamics with morphology of cells could lead to improved accuracy of cells classification in cases where the initial morphology differs.

3.3.5 Traditional ML with added Morphological Features vs Deep learning

To investigate the gain in performance obtained by deep learning over traditional machine learning methods, we extended the feature set of the manually derived features to include more descriptors of morphology. The added features were calculated in regions R1, R2, and R3 (Figure 1) and included the central moments of inertia along with Hu moments [60], which are a set of seven two-dimensional moments that are invariant to rotations, scaling, and translation (Supplementary Material Note 12). Using these new morphological descriptors, along with the previous derived features (Supplementary Material Note 4), a new SVM was trained and optimized through a grid search of relevant parameters. The resulting confusion matrix on a held-out test set and SHAP plots were calculated to understand the importance of these added features for classification (Supplementary Material

56

Note 12). The classification accuracy between HL60 and HL60d showed a marginal increase in performance (1%) compared to the SVM trained without the additional feature set. This, along with the poor classification of the undeformed shapes (65%) (Supplementary Material Note 11), suggest that the HL60 and HL60d cells are very similar morphologically, and that deformation-based features are the most informative for classification. The classification of HL60 vs HL60n on the other hand, was moderately improved by the newly added morphological features ($\sim$10%). The corresponding SHAP plot (Supplementary Material Note 12d) shows that the Hu moments in R2, where the cells are most relaxed, had the highest average impact on classification. The increase in classification accuracy along with the importance of features in R2 suggests that both morphological and deformation-based features contributed to classification of HL60 and HL60n.

While the addition of 30 new morphological features demonstrated an increase in SVM classification accuracy, it is impossible to ensure that our set of derived features forms a complete basis to describe all possible shapes the cells assume throughout their deformation and relaxation. Furthermore, we argue that features like the Hu moments, although useful for increasing classification, offer little interpretability and cannot be used to inform future physical models. In cases where one is most concerned with classification accuracy over interpretability, we found that deep learning, in particular a network with both convolutional and recurrent layers, outperforms traditional methods like an SVM. The classification accuracy of the CNN-GRU was $\sim$15% higher for HL60 vs HL60d, and $\sim$4% for HL60 vs HL60n. We assert this increase in performance is due to the ability of convolutional layers to generate a more complete feature set, along with the recurrent layers to learn the temporal dependence of the same features. It is important to highlight that the use of the deep learning models, like the CNN-GRU described here, currently presents almost no interpretability. However, this comes with a trade-off of improved classification accuracy.

3.3.6 Increased accuracy leads to higher enrichment

To explore the implications of an increased classification accuracy, we considered a hypothetical case in which a researcher has a heterogeneous sample of cells of many mechanical phenotypes and desires to characterize a sub-population within. The sub-population constitutes at most few percent of the total sample making other, more conventional analysis, such as single cell RNA sequencing, microscopy, etc., difficult. We envision our method can be implemented to identify the population of interest by combining deformability characterization, classification, and in the future sorting. In this case, classification accuracy determines how effectively rare cells can be sorted out of the mixed sample. In Supplementary Material (Note 1), we derive a relationship between the classification accuracy and the number of target cells recovered in a sorted sample. We define the enrichment as the increase in concentration of target cells in the sorted sample. We find that enrichment depends sensitively on classifier performance, increasing exponentially as false positive rate decreases below 0.5, then super-exponentially for false positive rate less than ~ 0.2 as shown in Figure S1. This dependence gives context to the importance of the increased classification performance we achieve. Using only morphological features, the classifier performance would allow enrichment of ~ 9 times. Using the mechanotyping data but with the inferior random forest classifier would allow an enrichment of ~ 12 times. Finally, using the CNN-GRU model would allow an enrichment of ~ 58 times (each averaging over prediction performance numbers for HL60d and HL60n, and assuming a rarity of 1/1000). Thus, the CNN-GRU classifier has about five times the potential effectiveness of the random forest classifier for future sorting applications.

3.4 Conclusions

In summary, we show the application of a channel with an undulating width that induces non-linear forces to individual cells at a high-throughput. The cells undergo multiple de-

formations and relaxations which reveal a multitude of information on cellular mechanics. We explore how much information is revealed from this dynamic and non-linear deformation process by comparing the classification accuracies of traditional machine learning models, with derived features, to deep learning models with automatic feature extraction. For the traditional machine learning models, the use of SHAP values unraveled mechanotyping parameters that contributed most to classification, which can be useful in gaining insight on how cell populations differ mechanically. At the cost of interpretability, the deep learning models showed an appreciable increase in classification accuracy, particularly in the case where the two cell populations were very similar morphologically.

While the deep learning characterization and classification are currently performed post-hoc, similar deep learning models with proper hardware can currently process $\sim$2,000 fps, leading to the possibility for real-time sorting. We envision this work being incorporated into currently existing technologies, such as imaging flow cytometry, and to be especially important for segregating rare cells, including circulating tumor cells[61] and even cancer stem cells[62].

Future work will include optimizing channel designs with different widths, lengths, and shapes, as well as extending the deep learning models to include unsupervised classification.

Chapter 4

Applications of deformability cytometry to neutrophil and lipocytes

In this chapter we will demonstrate two collaborative studies which have leveraged the deformability cytometry method presented in the previous chapter. We will first look at discriminating between subpopulations of a specific immune cell known as neutrophils. This work was done with Dr. Matthew Bovyn and Dr. Steven Gross. In the second section we will explore the mechanical heterogenities in lipids which construct cartilage in the ear of mice vs other mammals. This work was done with Dr. Raul Ramos and Dr. Maksim Plikus.

4.1 Classification of activated neutrophils

4.1.1 Motivation for studying neutrophils

Neutrophils are immune cells which are the bodies first response to infection and foreign invasion[63]. They are created in bone marrow and circulate through the blood, and make

up nearly 70 percent of the white blood cells in the body. Neutrophils possess an ability, known as chemotaxis, to sense chemical species which guides them to sites of infection[64]. They also possess multiple tools to fight infections and microbes in various ways, such as phagocytosis, reactive oxygen species (ROS), and extracellular traps (NETS)[65]. Phagocytosis is where a neutrophil engulfs a foreign microbe of interest and digests it internally using ROS. Neutrophils phagocytose microbes such as bacteria through rearranging their cytoskeletal filaments known as actin[66]. The ability of a neutrophil to phagoycytose can be regulated through activation, where a neutrophil is exposed to a specific chemical, or even activated through mechanical squeezing such as through a tight vessel[67, 68]. Understanding the mechanical phenotype of neutrophils can provide insights into the body's immune response and even has implications for cancer research and recently Covid-19[14].

4.1.2   Experimental methods

Creation of neutrophil populations and activation

Neutrophils are the most abundant white blood cells, and are constantly being created in the body. However, neutrophils are terminal and cannot differentiate. They are very short lived and typically can live only 4-12 hrs after creation. As a result, primary neutrophils from the blood can be difficult to work with experimentally. Another option for creating neutrophil assays is to culture the HL-60 cell line which is a leukemia stem-cell, from which neutrophils can be derived. In our study, we established a culture of HL-60 cells in media with 10% FBS as before, and differentiated them into neutrophils through incubation with DMSO at concentration of 1.5% (v/v). As seen in figure 4.1, flow cytometry confirmed that after incubation with DMSO for 4-5 days, the HL-60 population was positive for CD11b and CD15, which are common markers for identifying neutrophils.

We seek to compare the mechanical properties of primed neutrophil populations to those

(a) HL60

(b) Neutrophils derived from HL60
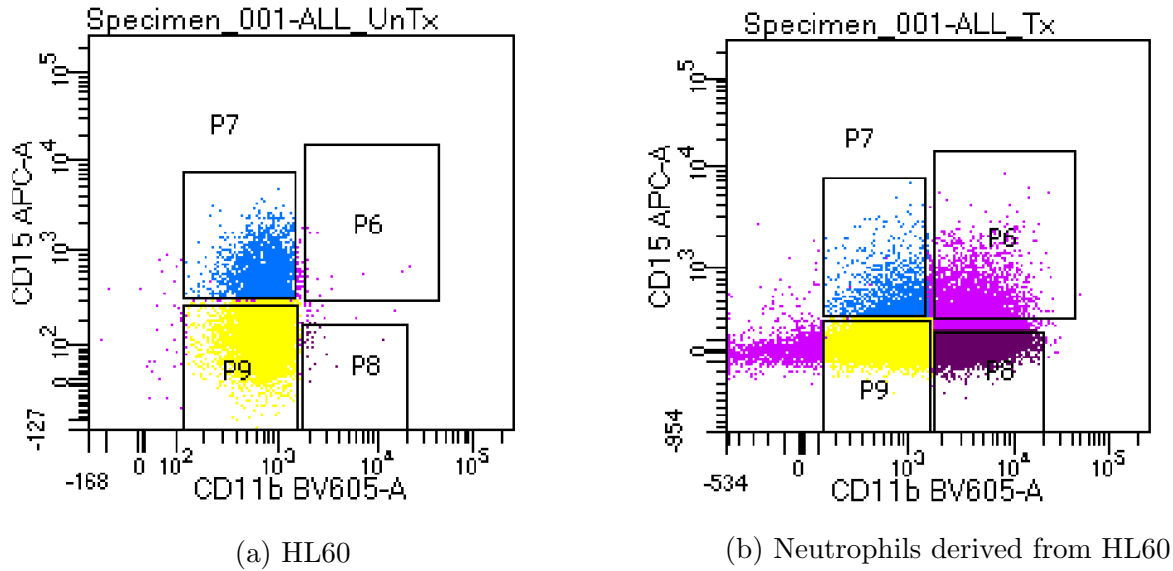
Figure 4.1: **Confirmation of neutrophil populations using FACS** A population of cells appear in region P6 and P8 positive for CD15 and CD11b after HL60 incubation with DMSO (0.1%) for three days, indicating that HL60 had differentiated into neutrophils.

which were unstimulated. To induce neutrophil activation and increase their antimicrobial response, derived neutrophil populations were incubated with PMA overnight. PMA has previously been shown to increase neutrophil phagoycytic and NET activity[69]. To provide evidence of neutrophil activation, we studied the phagocytosis activity of the activated neutrophils through florescence microscopy (figure 4.2). In short, activated neutrophil membranes were labeled with HCS CellMask Orange plasma membrane stain (ThermoFisher Scientific) and were incubated with cyan fluorescent bacteria (obtained from [70]). Cells were plated on a cover-slip and imaged using a confocal microscope at a magnification of 63x, where overlapping cyan and orange channels designate bacteria which have been engulfed. To confirm bacteria were engulfed, we performed a 3-D reconstruction of a Z-Stack and found that the neutrophil membranes entirely surrounded individual bacteria. After confirming the activation of the derived neutrophils through microscopy, we proceeded to study how the populations differed mechanically using the previously described microfluidic device.
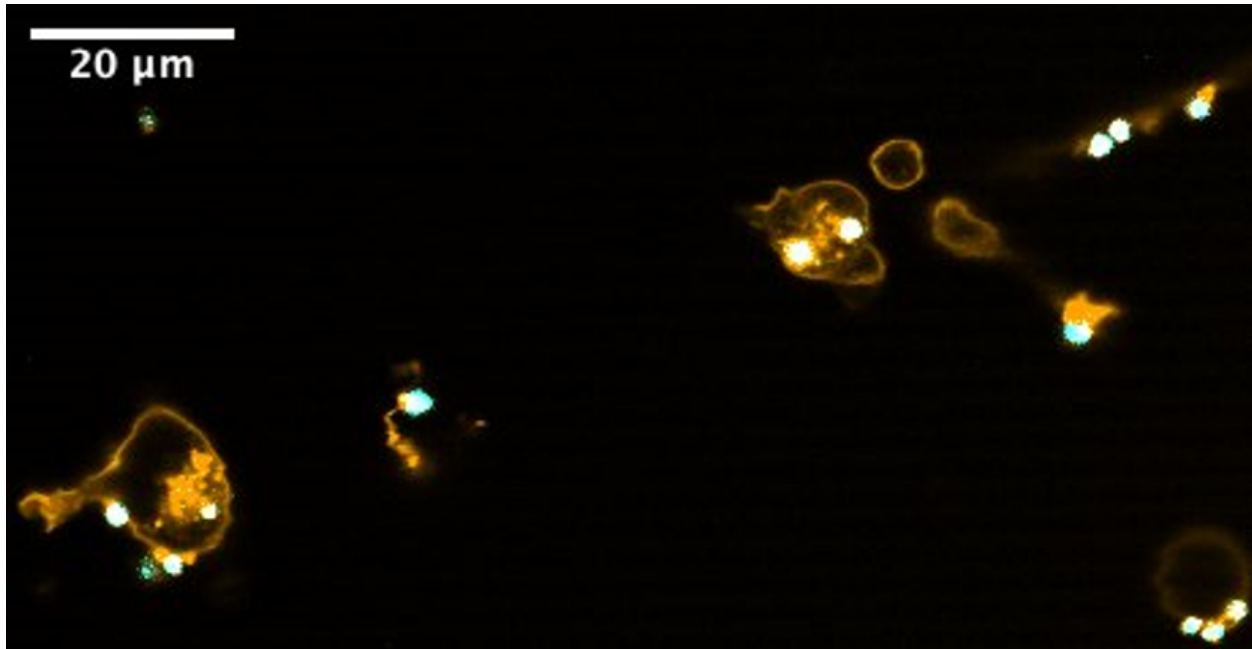
Figure 4.2: **Fluorescence confirmation of neutrophil phagocytic activity** Neutrophils labeled with CellMask Orange plasma membrane stain were imaged after activation with PMA stimulation. Images were taken after incubation with cyan fluorescent bacteria.

4.1.3 Results and discussion

Mechanical measurements and feature extraction

Due to the smaller nature of neutrophils compared to HL60, cells were pumped through a narrower microfluidic channel with length of 150 μm and three consecutive undulations of widths 15 μm, 30 μm, and 15 μm. Cells were again suspended in PBS supplemented with 1% (w/v) methylcellulose (4000 CP). In this experiment, cells were imaged using phase-contrast microscopy, as opposed to brightfield used in previous experiments described in Chapter 3. Phase contrast works on the principle of interference, rather than transmission intensity. Briefly, phase-contrast microscopy works by combining the light which passes through the specimen with unretarded light, resulting in interference between the two sources. For specimen which have an index of refraction close to that of surrounding media, like cells, the interference from phase shifts caused by the difference in optical path length create larger

63
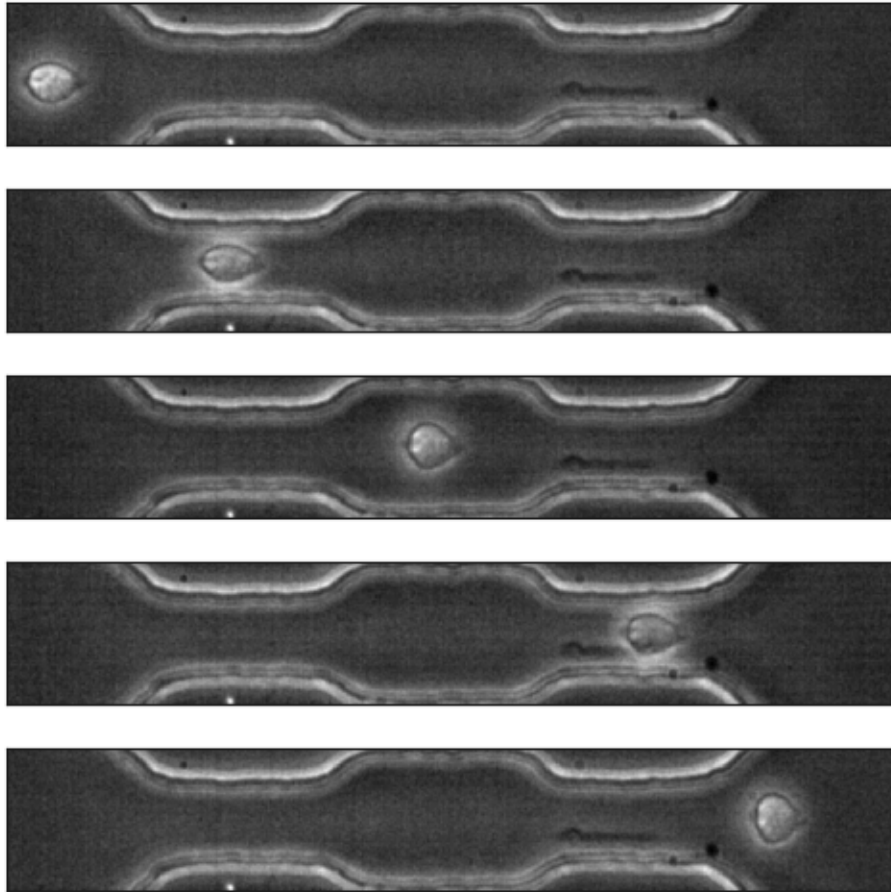
Figure 4.3: **Neutrophil deformation sequence** Snapshots of a single neutrophil deformation as it passes through undulating microfluidic channels. Neutrophils were imaged using phase-contrast microscopy.

changes in intensity as compared to transmission. As a result, phase contrast creates sharper edges, and the ability to even see structures present within the cell such as nuclei.

The two cell populations, untreated and PMA treated HL60 derived neutrophils, were independently suspended and pumped through the microfluidic system at a flow rate of $1\,\mu\mathrm{Ls}^{-1}$ with a sheath flow of the same rate. Videos were recorded at 11k fps with an exposure time of $1\,\mu\mathrm{s}$, at a resolution of 880x140 pixels, resulting in a pixel size of approximately $0.2\,\mu\mathrm{m}$. We observed during image segmentation that the neutrophils shapes were not always accurately described by an ellipse, but rather a 'bullet' like shape as observed in other

(a) Neutrophil Deformation	(b) PMA Neutrophil Deformation
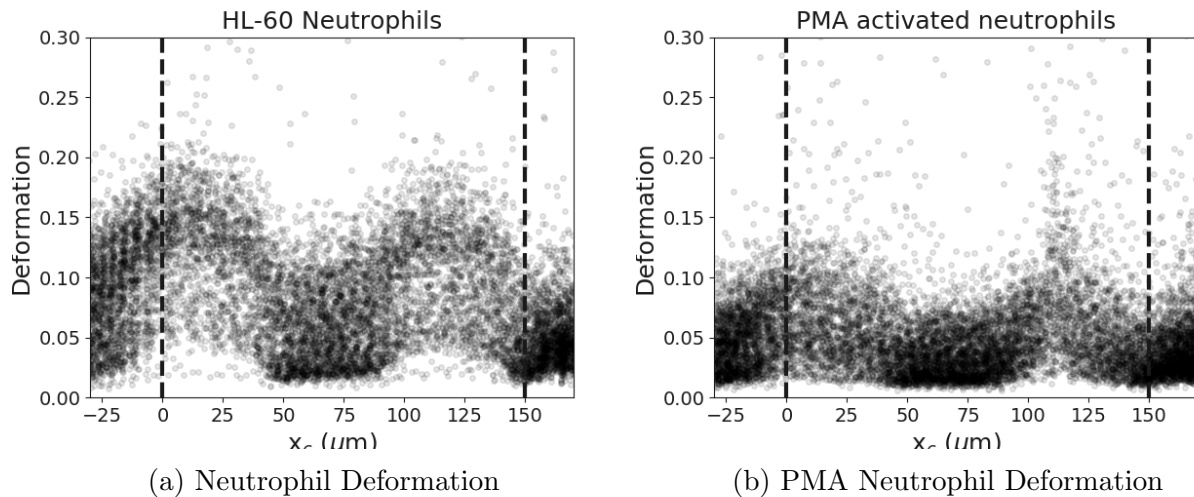
Figure 4.4: **Comparison of deformation traces between neutrophils and PMA activated neutrophils** The deformation as a function of position is plotted for both neutrophil populations.

microchannels[34]. To define the deformation of cells we used defined the deformation D as:

$$D = 1 - circularity = 1 - \frac{2\sqrt{\pi A}}{P} \quad (4.1)$$

where A and P are the area and perimeter, respectively. Here, a perfect circle would have the deformation defined as zero, and the max possible deformation is one. In addition to redefining the deformation metric, we added features utilizing internal cell morphology, i.e the standard deviation of intensity within a cell's boundary. Figure 4.3 demonstrates how the intensity of the cell changes as a single cell travels through the channel, going through deformation and relaxations. The change in brightness intensity of the cell is due to the stretching of the cell, which shortens the optical path length relative to the undeformed cell, resulting in different interference. Assuming the cell maintains a constant volume, this intensity will change depending on how much the cell is deformed. Plots of the deformation and intensity deviation as a function of position can be seen in both the untreated and PMA

| (a) Caption | (b) Caption |
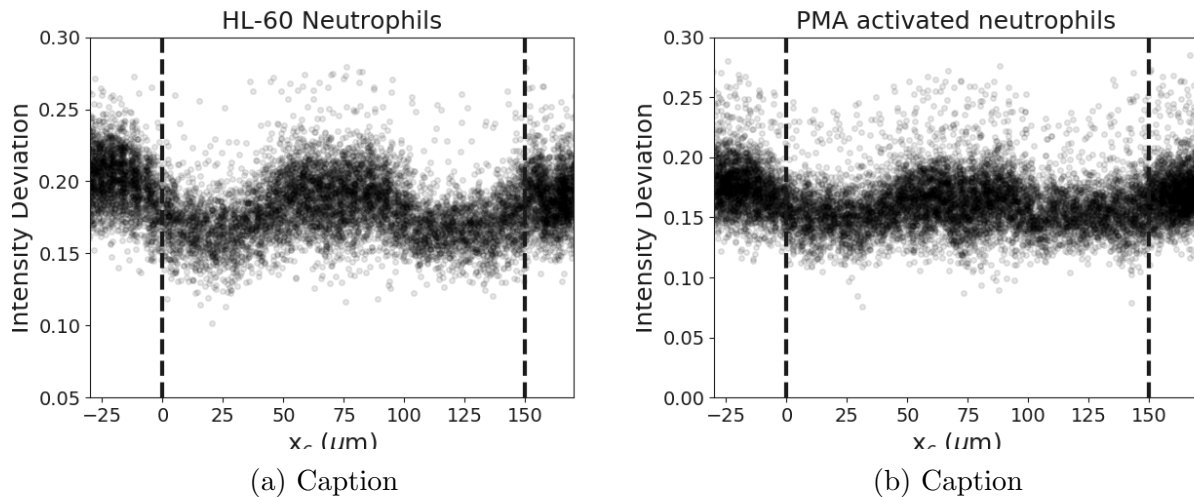
Figure 4.5: **Comparison of intensity variation traces between neutrophils and PMA activated neutrophils** The intensity variation as a function of position is plotted for both neutrophil populations.

treated neutrophils in Figures 4.4 and 4.5. As can be seen from figure 4.4, the neutrophils which had undergone activation with PMA resulted in overall lower peak deformation in the narrow constrictions as compared to the untreated populations.

With the decrease in deformation observed in the time traces for PMA treated neutrophils, it is possible that the deformation could be due to size differences between the two subpopulations, as cells of different sizes will experience different strain rates. Consequently, we must compare the radii distributions of the two populations in their underformed state to draw conclusions about the differences in deformation. The methods provided in this work are unique in that we can measure both the deformed and undeformed cell state, which occur in the narrow and cavity regions, respectively. As seen in Figure 4.6a, the means and distributions of the radii are very similar between neutrophils and PMA activated neutrophils. Since the two populations have the same radii distributions, we can assume that the cells experienced similar forces inside the channel. The changing mechanics from the treatment of the PMA are responsible for the differences we observe in deformation. The peak deformation in the first narrow constriction, where the max stress is applied,
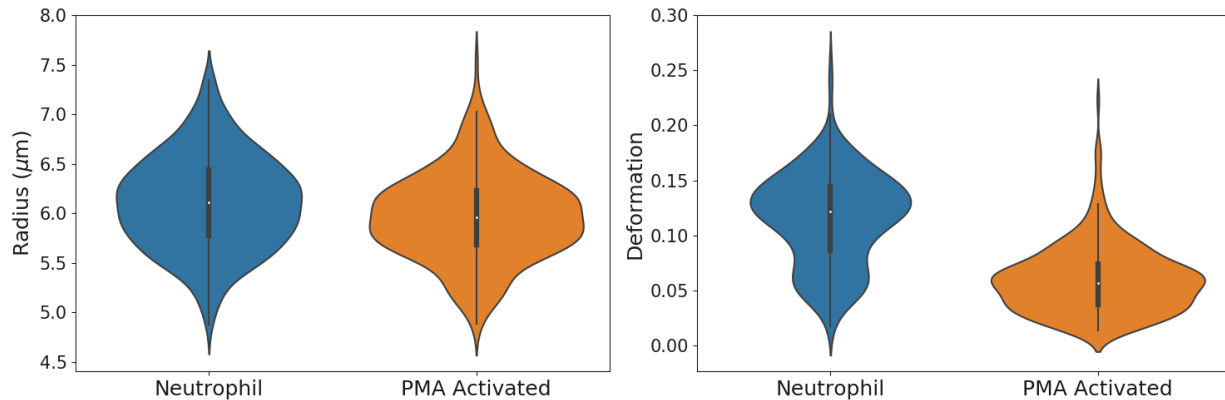
Figure 4.6: **Radii and Deformation distribution at channel opening** a) The distribution of cell radius is plotted for each cell type. The mean and variance of both populations are similar between cell types. b) The distribution of cell deformation at channel entrance is plotted for each cell type. The mean deformation of PMA activated neutrophils is significantly lower as compared to untreated neutrophils.

is compared between the two sub-populations in Figure 4.6b. Interestingly, the untreated neutrophils appear to have a bimodal distribution, where one of the peaks corresponds to that of the PMA treated neutrophils. This small sub-population of less deformable cells within the untreated population could be a result of neutrophil activation by mechanical perturbation.

Determining the amount of neutrophil activation populations has clinical applications for determination of disease such as cancer[71, 72] or monitoring the bodies' current state of immune response[68]. In such settings, fast diagnosis of the number of primed neutrophils is essential. The unlabeled approach, as apposed to molecular techniques such as flow cytomery, used in our studies provide an avenue to quickly determine the state of neutrophil polarization in real time. In order to find sub-populations of cell in an unlabeled fashion, such as activated neutrophils vs untreated, the use of unsupervised models are necessary since labels are not known a priorri. Here, we will explore the predictive power of unsupervised models utilizing the the full set of derived features extracted from the deformation time traces.
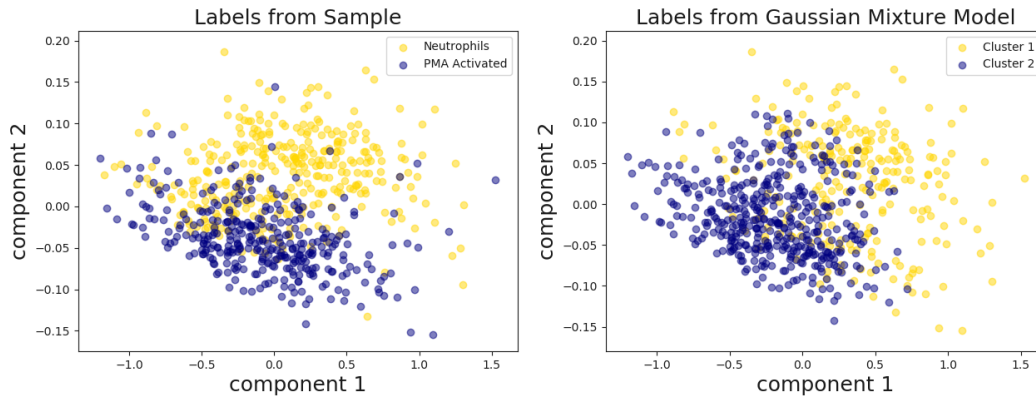
Figure 4.7: **Clustering of neutrophil populations** a) Reduced dimensionality of neutrophil populations using PCA b) Unsupervised GMM predicted labels.

Unsupervised classification of neutrophils

Unsupervised learning is often cast to the problem of how to cluster populations without previous knowledge of labels, as apposed to the supervised learning problem where labels are known and optimal classifications boundaries are found by training with the ground truth labels. A short but not comprehensive list of common clustering algorithms used in cell biology are k-means, spectral clustering, hierarchical clustering, and Gaussian-mixture-models (GMM), among others.

Before clustering, we reduced the feature space down to 2-D through principal component analysis (PCA). Principal components analysis works by finding a new basis that maximizes the variance in the data by finding the eigvenvectors for the covariance matrix. The first principal component contains the most information regarding the data, followed by the second and so on. We can select the first two vectors to reduce our space to 2-D, while retaining most of the information that was contained in the higher dimensional space. The result of performing PCA on the neutrophil data set can be seen in 4.7a, where the ground truth is labeled. Reducing the dimensionality of a feature space can improve model fitting, as well as provide a means for visualization.

After reducing the feature space, we utilized GMM for their ability to cluster multivariate data. GMM work by fitting both the mean and variance of a Gaussian distribution for each cluster. For multivariate data, we replace the variance with the covariance matrix. A GMM was fit to the PCA reduced features, with the number of clusters = 2. We can see the the results of the clustering in figure 4.7b. Qualitatively, the GMM was able to find the general structure of the two labels. Additionally, we see that it has clustered a population of neutrophils as PMA activated neutrophils. These incorrectly labeled neutrophils could belong to the second smaller population of neutrophils seen in figure 4.6b, which share the same mean deformation as the PMA activated neutrophils.

4.1.4   Conclusions

The work presented in this section demonstrates the possibility of identifying heterogeneous neutrophil populations from unlabeled data using only physical descriptors. This complements previous work which has shown the ability to classify between neutrophils and activated neutrophils through moprhological features[73]. This method could be extended to detection of immune cells in whole blood and has direct application to diagnosing various diseases, including cancer, leukemia, and even COVID-19.

4.2   Mechanical differences in cartilage of mice

4.2.1   Cartilage composition

The body contains three types of cartilage: hyaline, elastic and fibrocartilage. These three cartliges can be distinguished base upon the structure of the extracellular matrix (ECM)[74]. The elastic cartilage is in particular fascinating since it contains mostly cells and little

69

ECM[75]. The ECM is created and maintained by cells called condrocytes, which comprise all three type of cartilage. It has been found that the elastic cartilage in mice are unique compared to other mammals; they contain fat droplets that are similar to those found in adipose tissue[76]. Furthermore, the cartilage consists of very little ECM. These fatty chondrocytes have been named lipochondrocytes. Here we will explore how these lipochondrocytes differ mechanically from adipocytes at the single cell level when isolated from the ECM and tissue. This study has implications on understanding of the evolution of cartilage tissue.

4.2.2   Experimental methods for measuring lipocytes

Updating imaging parameters

We are interested in understanding the differences (if any) in mechanics between lipochondrocytes (LD) and adipocytes (AD), namely their whole cell deformation response. The key difference between these cells and the previously studied HL60 cells are that LD and AD cells are taken from tissue, whereas the HL60 cells can be cultured. Furthermore, the cell densities we obtain from these tissues are far lower than densities obtained from culturing. As a result, the final working densities for these studies is over 50-100x lower than used in our previous studies. These lower cell densities create a technical challenge, as our imaging parameters of 11k fps at a resolution of 880x140 only allow for a recording of 8s in the camera ring buffer. To identify a statistically significant number of cells, it is necessary to increase the recording time of our device. To increase the recording time we can reduce to field of view or resolution at the cost of losing cell dynamics. The smaller field of view was placed at the entrance of the channel, where the large stress occur. With these imaging conditions, were were able to capture single images of each cell type deforming at the channel entrance. Example images of each cell type deforming can be see been seen in figures 4.8 and 4.9.
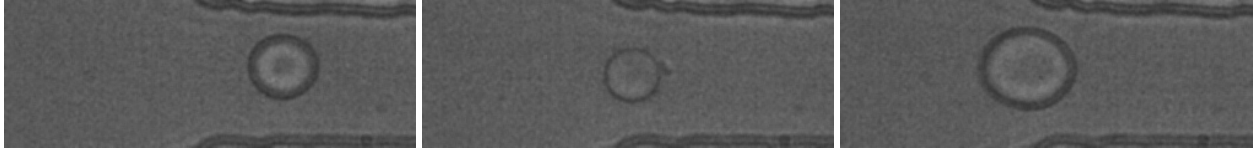
Figure 4.8: **Examples images of single adipocytes** Single frames of adipocytes deforming at the entrance of the microfluidic channel.
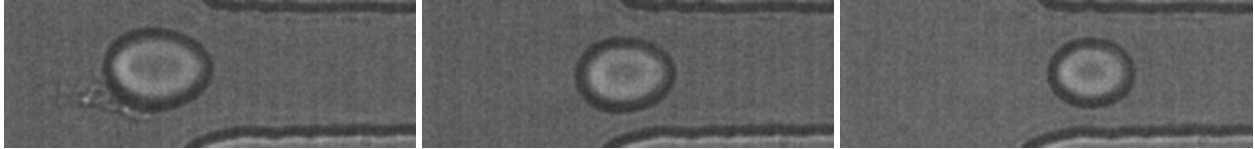
Figure 4.9: **Examples images of single lipochondrocytes** Single frames of lipochondrocytes deforming at the entrance of the microfluidic channel.

4.2.3 Results and Discussion

Gating cells by position and size

When the field of vision or resolution is reduced, we can only capture 1-2 frames of each cell near the entrance of the channel. As a single cell approaches the channel entrance, the stress that it experiences increases, until reaching a maximum at the channel entrance. By taking a single frame, we are left with a distribution of positions around the channel entrance. As can be seen in figure 4.10, the stress a cell experiences is dependent upon its position relative to the channel opening. Comparing cell deformability at different positions within the channel could lead to false conclusions on deformability, as the cells are not experiencing the same forces. Gating the cells to within a narrow range of positions around the channel entrance is necessary in order to draw conclusions. However, with each gating performed, we are also reducing the number of cells being analyzed which can effect statistical analysis. As we can see in figure 4.10, the average force on a cell reaches a maximum at $\sim 5\mu m$ into the channel. To compensate for these differences in forces, we gated the populations to only contain cell events where the position was located between 0-5 μ m.

In addition to gating by channel position, we must also gate the cells based upon size.

71
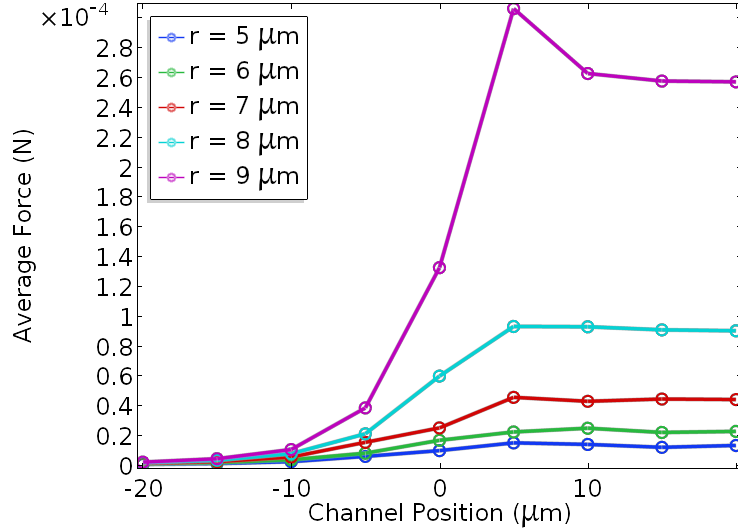
Figure 4.10: **Modeling of force on spherical particles** A FEM model was ran using COMSOL Multiphysics 5.3 solving stokes flow in a simplified channel geometry. The fluid flow was calculated for spheres of different radii at various channel position around the opening. Force was calculated by integrating the total stress over the surface of the spherical particle.

Figure 4.10 shows how the force experienced by cells at the entrance changes as a function of cell size. We can see that as the cell approaches the width of the channel $(20\mu m)$, the forces increase greatly, with a nearly 3x increase in average force between the $r = 8\mu m$ and $r = 9\mu m$. This is even more important when considering the size distributions of each cell type. We can see in figure 4.11 that the AD are high in number, but that they have a very large variance in size (area). Conversely, we were only able to capture a small number of LD, but the area of these cells were relatively uniform with low variance, as compared to AD. To ensure that the cells being compared experienced similar forces, we removed all cells from analysis which were smaller than the minimum area found in the LD distribution. Additionally, large outlier cells were removed from the analysis. The effects of the position and size gating on aspect ratio (or deformability) can be observed in figure 4.12.
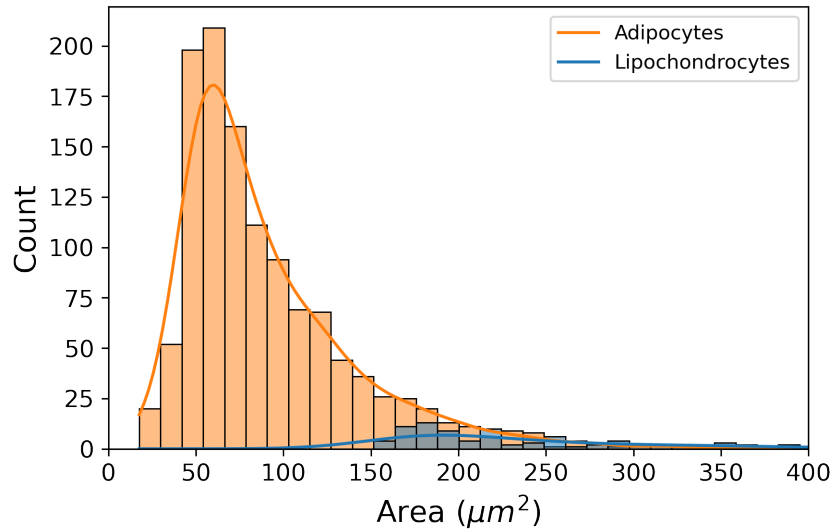
Figure 4.11: **Comparison of sizes of adipocytes and lipchondrocytes** Cell area was defined by the number of pixels identified belonging to the cell mask after segmentation. Area was converted from number of pixels to μm.

Deformability of adipocytes vs lipchodrocytes

After gating, we lose a large number of cells from the adipocyte population, mainly due to the large variation in size. We can see from figure 4.12, that the distribution of the LD aspect ratio is shifted to the right, with a slightly higher mean indicating that the LD are more deformable relative to the AD. To determine if this conclusion is statistically significant, we performed an independent t-test on each samples aspect ratio. We find that the sample means are not statistically significant and we cannot draw conclusions about whether LD or AD are more deformable. We must accept the null hypothesis that the means of the cell deformations are equivalent.

4.2.4 Conclusions

In this section, we compared the deformability of a unique cell type to mice called lipochon-droyctes, which resemble fat cells found in adipose tissue. We performed this analysis on
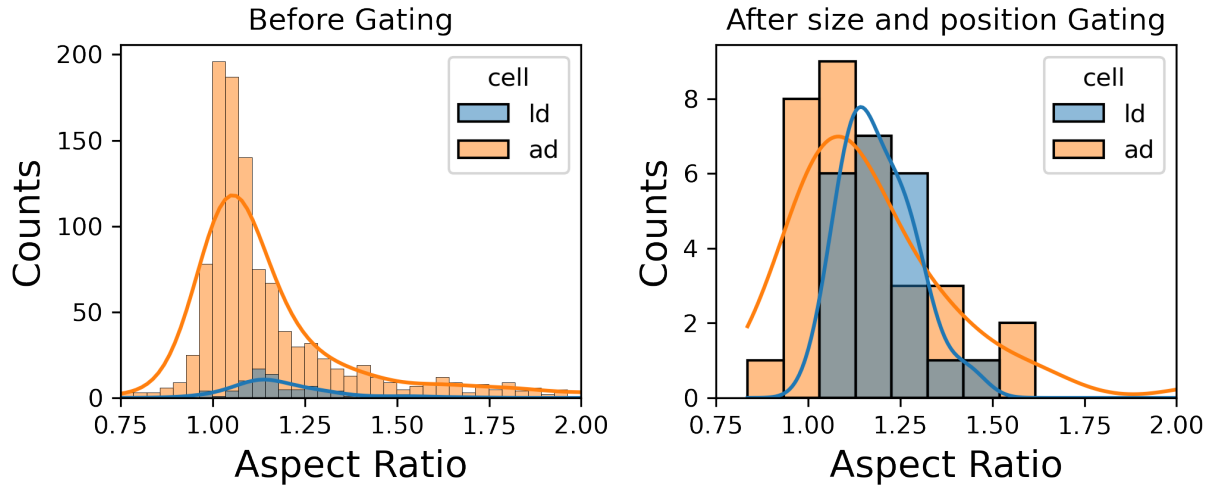
Figure 4.12: **Deformability before and after gating** a)Histogram of aspect ratios for both LD and AD before any gating is performed. b) Distribution of aspect ratios after gating for channel position and size.

a single cell level, as compared to measuring the deformation and elastic properties of the whole tissue. This work is part of a larger work, where this analysis is performed. To best understand the full picture of mechanics in mouse ear cartilage, the interaction between the lipchondrocytes and the ECM will have to be modeled and compared to adipose tissue. This work has found that their is no significant difference in deformation between lipochondrocyes and adipocytes, potentially supporting that the differences in mechanics between adipose tissue and elastic cartilage of mice lies in the ECM.

Chapter 5

Discriminating bacteria populations by shape using impedance cytometry

The previous studies presented in this worked focused on classifying on and measuring elastic properties of single cells at high throughputs and observed them optically as they pass through microfluidic channels. In this chapter, we will demonstrate the ability to classify bacteria based on physical properties such as shape and surface potential. In contrast to previous chapters, the methods presented here rely on electrical based characterization technique known as resistive-pulse. By combining the resistive-pulse technique with undulating polymer pores, we can classify bacteria based upon their shape, leading to potential applications in rapid detection of bacteria susceptibility to antibiotics.

This worked was performed collaboratively with Dr. Wilfred Russell, Leora Duong, Professor Albert Siryaporn, and Professor Zuzanna Siwy.

5.1 Introduction

The ability to quickly detect heterogeneous bacteria populations can have broad impacts in clinical settings for the the treatment of sepsis, or development of new antibiotics for resistant bacteria[77]. Genotypic identification can be used to classify bacteria populations; however, these methods are costly both monetarily and in time, preventing their implementation for rapid detection [78]. Another approach to classifying bacteria is through physical descriptors. Since bacteria contain rigid cell walls[79], populations can often be phenotyped into classes based upon their shape, namely spherical (Coccus) or rod like (Bacillus), among others, or by other physical descriptors such as size and morphology[80, 81, 82].

One attractive label-free approach for determination of cell size is resistive-pulse (RP) sensing[83, 84, 85, 86]. In RP sensing, current drop is measured as a particle passes through a channel, where this drop is proportional to the volume of the passing particle. This approach has been extended to determine particle shapes by measuring changes in electrical current as particles rotate to different orientations during channel translocation, changing the overall impedance of the system[87, 86]. Another approach for classifying particles based upon shape using RP sensing was suggested by Qiu et al[88], where they were able to distinguish between spherical and rod shaped silica particles of different lengths through RP sensing of channels with non-uniform geometries, or rather natural undulations. These natural undulations create a 'signature' characteristic RP signal based upon the local pore geometry[89]. Particles of different lengths can alter this characteristic RP signal by performing an averaging over the local pore geometry.

In addition to features like size and shape, other physical characteristics of bacteria such surface charge and surface potential have been used to phenotype bacteria populations[90]. Bacteria can pass through the pore by through the application of a trans-membrane voltage or pressure difference. When a voltage is applied, bacteria travel electrophoretically or

electroosmotically. The transit time of a bacteria passing electrophoretically is proportional to the surface potential of that bacteria. As a result, passage time presents another attractive feature to use for classifying bacteria in a label-free approach.

In this work, we will utilize RP sensing with 'rough' PET pores to determine how well we can discriminate between bacteria populations differing in size, shape, and surface potential. We apply this technique to three bacteria populations, E. coli, mid-log E. coli, and staphylycoccus aureus (S. aureus). The averaging technique combined with passage time demonstrate the capacity to phenotype these populations to 95% accuracy through unsupervised clustering by Gaussian-mixture models. An alternative shape-based approach is tested through the use of dynamic time-warping (DTW) with barycenter-averaging (DBA) as a similarity measure for k-means clustering, resulting in an accuracy of 80%. We then test the application of this technique for identifying untreated E. coli populations vs E. coli populations treated with antibiotic, and found the ability to classify these populations with 96% and 87% accuracy.

5.2 Methods

Chemicals

Polyoxyethylene 20 sorbitan monooleate (TWEEN) was purchased from Sigma Aldrich (St. Louis, MO, USA). Potassium chloride (KCl), sodium hydroxide (NaOH), TWEEN 80, and Gibco™ PBS (phosphate buffered saline) (1X), pH 7.4 were purchased from Fischer Scientific (Hampton, NH, USA). Doubly deionized (MilliQ) water purified to a resistivity of 18.2 MΩ using a Milli-Q IQ 7000 water purification system was purchased from MilliporeSigma (Burlington, MA, USA). Unfunctionalized 400 nm diameter polystyrene microspheres were purchased from Bangs Laboratory Inc. (Fishers, IN, USA). All specifications for the micro-

spheres were reported by the manufacturer. All chemicals were used as received and not purified any further.

Mesopore Fabrication

Single polyethylene terephthalate (PET) mesopores were fabricated using the track-etching technique described previously(REF). Briefly, 12 μm thick PET, polymer foils were irradiated using single track energetic (11.4 MeV/n) Au ions at the UNILAC: Universal Linear Accelerator (GSI Helmholtz Centre for Heavy Ion Research, Darmstadt, Germany). Each individual PET film was then etched chemically. First, by being subjected to additional irradiation, this time, from a 115V, UVGL-25 Compact UV Lamp from UVP, LLC (Upland, CA, USA). Both sides of the PET foil were irradiated with long wavelength (365 nm) ultraviolet light for one hour before being submerged into a 2 M NaOH solution; and heated to a temperature of 60 °C using an in-house water bath. The pore diameter increases linearly with the etching time. Once etched, the mesopore was left overnight in MilliQ water before being sized. Mesopores fabricated along the axis of the latent track in this manner are assumed to have a rough, cylindrical shaped geometry [91].

Mesopore Sizing

Prior to use, the average diameter of each PET mesopore was electrochemically sized using linear sweep voltammetry wherein; the measured conductance in a solution of known conductivity, was used to calculate the pore diameter. For determining the diameter of the PET mesopore, unbuffered 1 M KCl was prepared in MilliQ water. This solution was used to size all subsequent PET mesopores throughout this study. Electrochemical sizing measurements were collected using a Keithley 6487 Picoammeter/Voltage Source (Keithley Instruments, Solon, OH, USA), and software written in-house on MATLAB (MathWorks, Natick, MA,

USA). Using a two-electrode configuration, Ag/AgCl pellet electrodes provided from A-M Systems (Sequim, WA, USA) were used as the working and ground electrodes. These electrodes were placed in opposing chambers of a custom-built conductivity cell with a single pore PET film mounted between the chambers. The Ag/AgCl pellet electrodes and conductivity cell were used to perform all the electrochemical measurements hereafter. Once the experimental set-up was complete, a linear potential sweep (-1 to 1 V with 0.1 V steps) was applied to the working electrode for 1 forward scan. The current generated from this forward scan was constructed into a current-voltage (i-V) curve, used to calculate the conductance, and the diameter of the PET mesopore.

Bacterial growth conditions

Bacterial strains E. coli MG1655 and S. aureus RN4220 were streaked on Luria-Bertani (LB) Miller (BD Biosciences, Franklin Lakes, NJ) petri dishes, which contain 2% Bacto agar (BD Biosciences, Franklin Lakes, NJ). Plates were incubated overnight at 37 ^{o}C to obtain single colonies. Single colonies were inoculated in LB broth containing 0.002% v/v TWEEN 80 and grown overnight at 37 ^{o}C, shaking at 225 rpm. Bacterial cultures were either resuspended in PBS with 0.002% v/v TWEEN 80, or sub-cultured 1:100 in LB with 0.002% v/v TWEEN 80, cultured to absorbance at 600 nm (OD600) of 0.1, and resuspended in PBS with 0.002% v/v TWEEN 80.

Cephalexin Treatment

Single colonies of E. coli strain MG1655-seq 1 were inoculated in MinA media (1 liter of MinA media contains by weight, 10.5 g K $_2$ HPO $_4$, 4.5 g KH $_2$ PO $_4$, 1 g (NH $_4$) $_2$ SO $_4$, 0.5 g sodium citrate) with 1 mM MgSO4, 0.2% glucose, 0.1% casamino acids and grown overnight at 37°C, shaking at 225 rpm. Bacterial cultures were sub-cultured 1:100 in

the same medium, cultured to absorbance at 600 nm (OD 600) of 0.1, and treated at room temperature with cephalexin (Cayman Chemical, Ann Arbor, MI, USA) dissolved in Milli-Q water.

Microscopy

Following bacterial preparation, 5 μl of bacterial cultures were spotted onto glass slides or glass slides with 1% agarose pads to immobilize the bacteria. Images of bacteria were acquired immediately after using a Nikon Eclipse Ti-E microscope (Nikon, Melville, NY) containing a Nikon 100X Plan Apo (1.45 N.A.) objective, a phase contrast annulus, a Hamamatsu Orca Flash 4.0 V2 camera (Hamamatsu, Bridgewater, NJ), and Nikon NIS-Elements version 4.5. Images were analyzed using ImageJ (NIH, Bethesda, MD). A minimum of 100 cells were imaged and analyzed per experiment.

Resistive-Pulse Measurements

Resistive-pulse sensing of microsphere and bacteria translocation through the PET mesopores was performed with an Axon Instruments Axopatch 200B integrated patch clamp and 1550B1 Digidata 16-bit data acquisition system (Molecular Devices, LLC, San Jose, CA, USA). All resistive-pulse measurements were hardware low-pass Bessel filtered at 1 kHz (80 db/Decade), sampled at a frequency of 20 kHz, and digitized at 10 kHz. A 100 mM KCl solution containing 0.001% v/v TWEEN 20 was used for the polystyrene microsphere control experiments, and a PBS solution containing 0.002% TWEEN 80, for the bacteria experiments. Solutions containing the unfunctionalized polystyrene microspheres and bacteria cells were placed into the chamber of the conductivity cell containing the ground electrode and the PBS/TWEEN solution without either probe into the opposing chamber containing the working electrode. The concentration of polystyrene microspheres and bacteria cells

in the one chamber was $\approx 10^9$ particulates/mL. Once assembled, the entire apparatus was placed inside a faraday cage (Warner Instruments, Hamden, CT, USA) on top of a vibration cancellation table (TMC, Peabody, MA, USA). Data was analyzed on Clampfit 11.1 (Molecular Devices, LLC, San Jose, CA, USA).

Event Finding and Data Analysis

Determination of start and end of individual events was achieved through setting a current threshold using Clampfit 11.1 (Molecular Devices) and the baseline as adjusted to remain constant. Events where multiple particles traveled through the pore simultaneously were rejected from the analysis. A custom Python (3.6) script further refined start and end times for individual events. The start of an event was signified as the signal passes a threshold of 5σ of the baseline current magnitude. The resistive-pulse magnitude was determined by numerically integrating individual RP events and normalizing by the event time. Determination of bacteria sizes and shapes through microscopy were analyzed in ImageJ, where individual bacteria we were fit with ellipses. Estimation of the RP amplitude from of these bacteria populations was calculated through Eq. 5.2, assuming a sphere with same volume as the fit ellipsoid and using the measured average pore size.

5.3   Results and Analysis

5.3.1   Probing pore structure through resistive pulse sensing

Resistive-pulse sensing (RP) is an electronic based method that can be used for determining particle size, concentration, and even shape [83, 84, 85, 86]. Common properties among RP techniques include separating two reservoirs containing salt solutions, where one reservoir
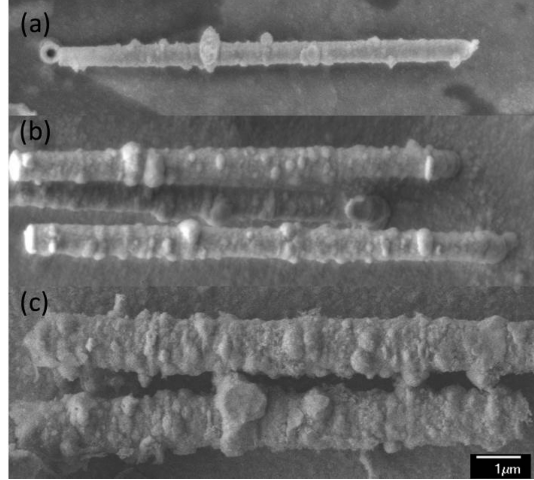
Figure 5.1: **SEM of metal replica of PET pore structure.** Multi-pore PET films were chemically etched and gold was deposited to fill the pore. PET was dissolved away leaving only gold nanorods. Figure reproduced from Innes et. al.[91]

contains the analyte of interest. The two reservoirs are separated by a channel, whose width is on the order of size of the analyte under study. RP sensing can be used on broad range of length scales, from tens of microns seen in the coulter-counter, down to the nanometer scale seen recently for the detection of nucelotides and amino acids [92, 93]. A variety of materials have been used for RP sensing and sequencing, including but not limited to: glass pipettes, silicon nitride membranes (SiN), protein nanopores, 2-D materials such as graphene and MoS2, and polyethylene terephthalate (PET) pores. Here we explore the use of PET pores for sensing bacteria, which are on the order of 1-2 microns. PET pores are created by chemically etching along a latent damage track created by passing single high-energy particles generated from a particle accelerator. Due to the structure of PET and the semi-crystalline structure of the polymer, PET pores contain natural undulations in size, where the pore width various as a function of channel position. This phenomena can be seen in figure 5.1.

To determine the size of such pores, we can approximate them as cylindrical resistors, where the conductivity ρ is dependent on the type of salt and and concentration of that salt. The average diameter of these PET forms can be determined by sweeping over voltage (V) values

and measuring current (I). The slope of this curve is the reciprocal of the resistance (R), where the resistance of a cylindrical resistance is given by:

$$R = \rho \frac{L}{A} \tag{5.1}$$

where L and A represent the length of the pore and cross-section area, respectively. By knowing the length of the pore, we can determine the average opening diameter (D). However, as can be seen in figure 5.1, PET pores can contain large undulations and cavities; these undulations are averaged together when determining the pore width using the above method. Pevarnik et al.[89] have previously shown the local pore geometry can be revealed by passing small particles through PET pores and observing the RP signal. The change in current amplitude due to the passage of a spherical particle through a cylindrical channel is dependent on the relative ratio of their volumes and is given by:

$$\frac{R_p - R_e}{R_e} = \frac{I_e - I_p}{I_p} = \frac{d^3}{D^2(L + 0.8D)} S\left(\frac{d}{D}\right) \tag{5.2}$$

where $R_p(I_p)$ and $R_e(I_e)$ are the resistance (current) in the pore when a particle is present and with no particle present. It is important to note that this value is independent of the ionic concentration, and is solely dependent on the geometry of the particle and pore. The additional term is called the shape-factor (S) and takes the form:

$$S\left(\frac{d}{D}\right) = \left[1 - 0.8\left(\frac{d}{D}\right)^3\right]^{-1} \tag{5.3}$$
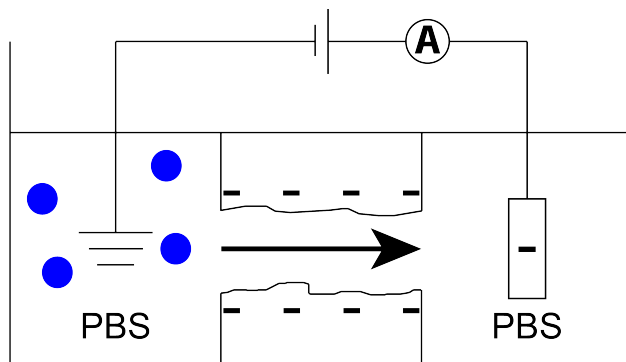
Figure 5.2: **Diagram of Resistive-Pulse experimental setup.** PBS buffer is placed is on both sides of a conductivity cell, connected through a micro channel. Analyte of study is placed opposite of the working electrode (ground) and where an electrical current induced electro-osmotic or electro-phoretic transport. Current is amplified and digitized before being measured.

When particles which are smaller than the undulating features of the pore are passed through the system, the current amplitude changes relative to the size of the local pore width as a result. These RP signal from these small particles can be seen as a mapping of the pore geometry, where each pore has a unique shape. To demonstrate this principle, we etched two separate PET pores, as described previously. 400 nm in diameter neutrally charged polystyrene particles suspended in 100mM KCl and 0.002% (v/v) Tween80 were placed on the ground side of the conductivity cell(schematic can be seen in figure 5.2). Particle passage was observed when a voltage of -0.2 V was applied to the working electrode, indicating that the particles traveled through electro-osmotic flow, which is in agreement with the direction induced by the negatively charged surface of PET. Example RP signals of individual events can be seen in figure 5.3 for two different pores of radii 409 nm and 484 nm. The fluctuations observed in these events are a consequence of the pore undulations.

While small particles have been shown for tracing local pore topography, one could ask what would happen if we used longer particles, such as rods. For a rod whose length is greater than the length of undulations, Qiu et. al.[88] demonstrated that silica rods 'smooth' the RP signal, relative to using a smaller spherical particle. The longer rod shaped particle can be seen as performing a windowed average across the RP signal, where the amount of averaging
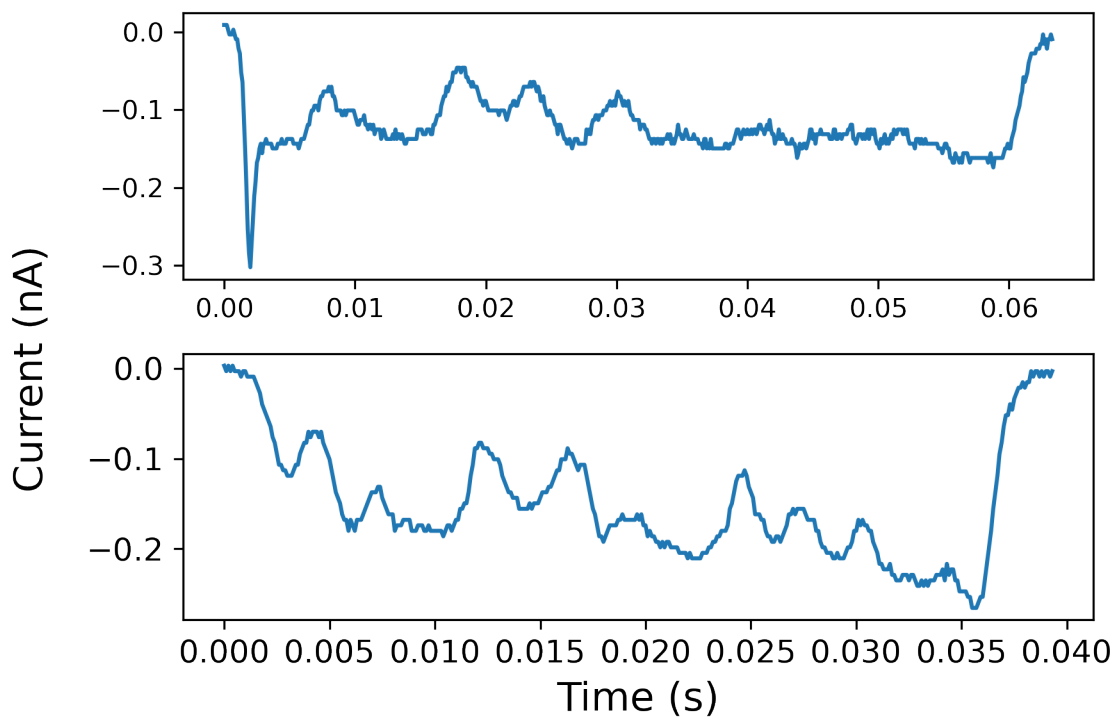
Figure 5.3: **Current trace of polystyrene particles.** RP signal of individual 400nm polysytrene events in 409nm and 484nm radii PET pores. The observed current fluctuations are changes in resistance the varying width of PET pores.

(a) E. Coli   (b) Mid-log E. Coli (E. Coli 2)   (c) S. Aureus
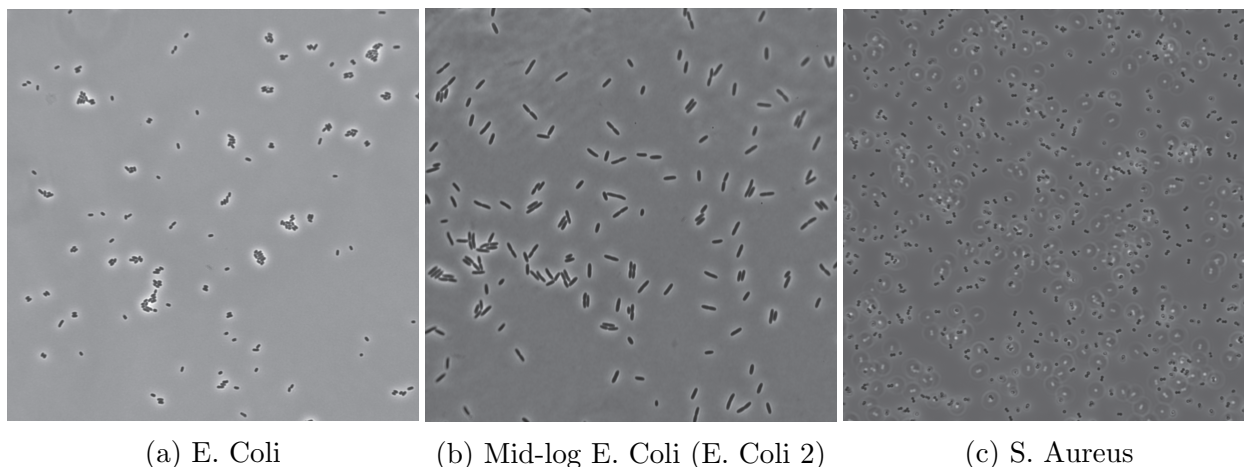
Figure 5.4: **Phase contrast images of bacteria populations.** Bacteria were imaged through phase-contrast microscopy at a magnification of 100x. Bacteria were immobololized on a 1% agarose pad. Cells boarders were segmented, fit and analyzed using ImageJ

will be proportional to the length of the translocating particle. As a result, one can utilize the 'smoothed' time signal, together with the amplitude of the RP amplitude, to determine the relative size and shape of different particles. In the remaining sections, we will determine the classification potential of this particle averaging method applied to biological analytes, specifically to bacteria populations of different shapes, combined with the unique RP signals characteristic of etched PET pores.

5.3.2 Imaging of bacteria

In this work, we created three bacteria populations as a means to test the classification potential of RP detection with PET pores on biological analytes differing in shape. We established three bacteria populations: E. coli, E. coli at mid-log growth (E. coli 2), and S. aureus (Staph), whose shapes can be described as both rod-like and spherical. The three populations were imaged using phase-contrast microscopy in order to determine populations shapes optically (Figure 5.4). To estimate the distribution of bacteria volumes, we used ImageJ to fit the boarders of individual cells with ellipses, and assumed the bacteria to be ellipsoids where the two semi-minors axes are equal. From these volumes, we can then
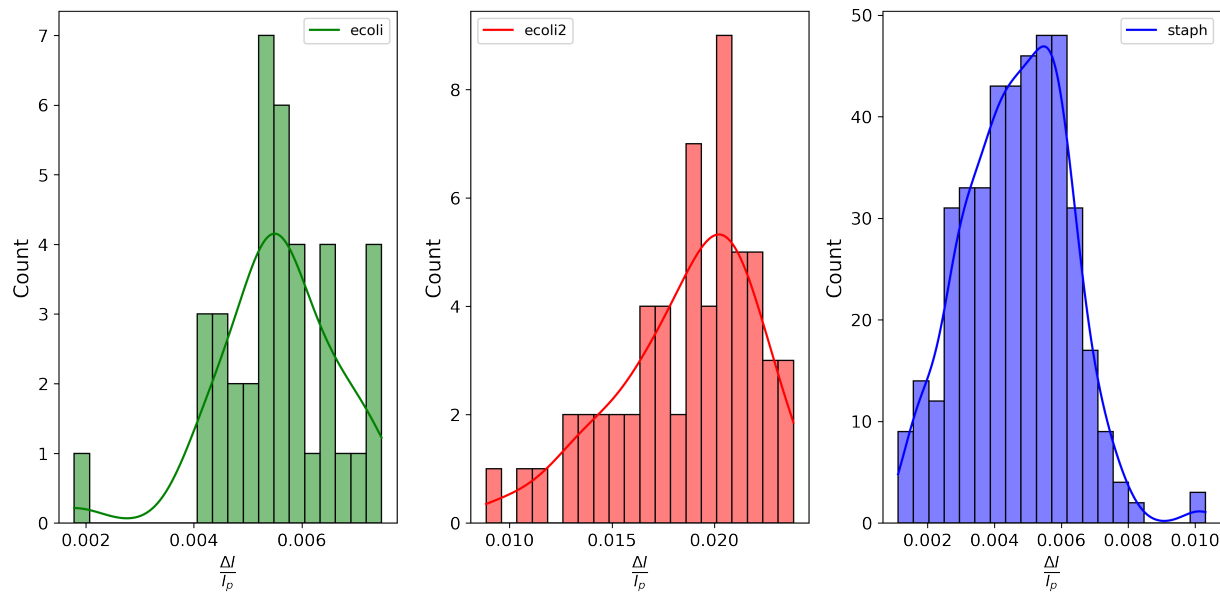
Figure 5.5: **Estimation of RP amplitude distributions for each bacteria populations.** Cell's borders from figure 5.4 were fit with ellipses and volumes were calculated assuming equal semi-minor axes. Radii for spheres of equal volume were calculated for each bacteria. Eq. 5.2 was used to calculate the predicted RP amplitude $\frac{\Delta I}{I_p}$.

calculate the predicted RP amplitude distribution using eq. 5.2. Note that here we assumed spheres of equal volume to the measured ellipsoids. In figure 5.5 we can observe the predicted distributions, where we see that the the E. Coli and S. aureus populations have significant overlap in their distributions, while the predicted values for E. Coli 2 show no overlap. These predicted values are only a measure of the current change due to differences in volume, and do not present a means for distinguishing the objects by shape.

5.3.3 Resistive-Pulse characterization of bacteria populations

The three bacteria populations were cultured and suspended in PBS buffer at concentrations of $10^8 - 10^9 particles mL^{-}1$ and supplemented with 0.002% Tween80 to prevent aggregation. Bacteria were placed on the same side as the ground electrode, where PBS buffer with no bacteria was placed on the working electrode side of the conductivity cell. Bacteria passages
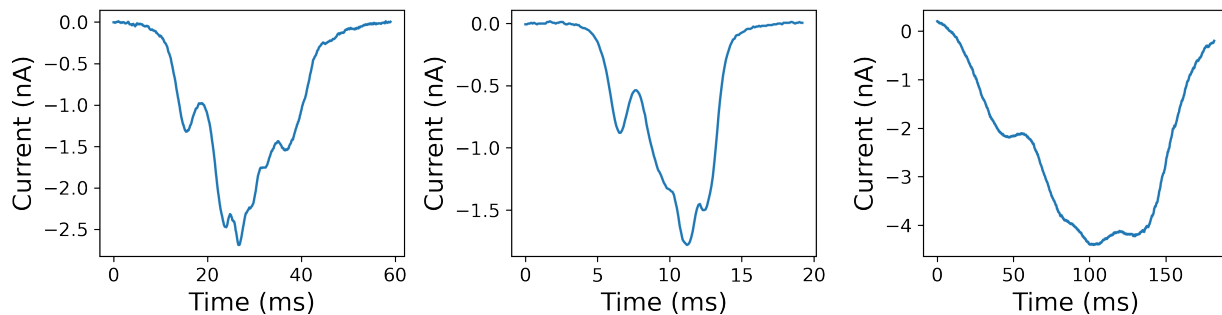
Figure 5.6: **Example event signatures for each bacteria.** a) Example event for S. Aureus. b)Example event for E. Coli. c) Example event for E. Coli 2

were observed at positive 0.2V indicting that the bacteria traveled electro-phoretically, implying that the E. Coli, and other populations, had negative zeta potentials (surface charge). This was confirmed through the use of a zetasizer, where we found the mean zeta potential to be -10mV for E. Coli.. Figure 5.6 shows the results of bacteria passage on a single PET pore of diameter 3.2 µm and length 12 µm.

In figure 5.6, we can see examples of single events for the three respective bacteria populations, where from left to right the signals are from S. aureus, E. Coli, and E. Coli 2, respectively. Qualitatively, we can see that all three signals share the same shape, a small dip in current, followed by a later larger dip, before returning to the original baseline. The sequences are also stretched to different time scales due to their differences in zeta potential and length. For S. aureues we can observe the smaller details within the RP signal, whereas the signal becomes smoother and detail is lost in the E. Coli signal. This smoothing or averaging becomes even more evident in the E. Coli 2, where almost all but the largest features in the signal become averaged together. We can also see that the event times and amplitudes differ between the signals. The difference in amplitudes of the signals is due to differences in volume, and in small part due to the shape. The difference in transit time is a function of zeta potential magnitude, and possibily size and shape. Often, the event amplitude and transit time are features used to distinguish between different particles in RP sensing. In figure 5.7, we can observe the distribution in RP amplitudes across the different
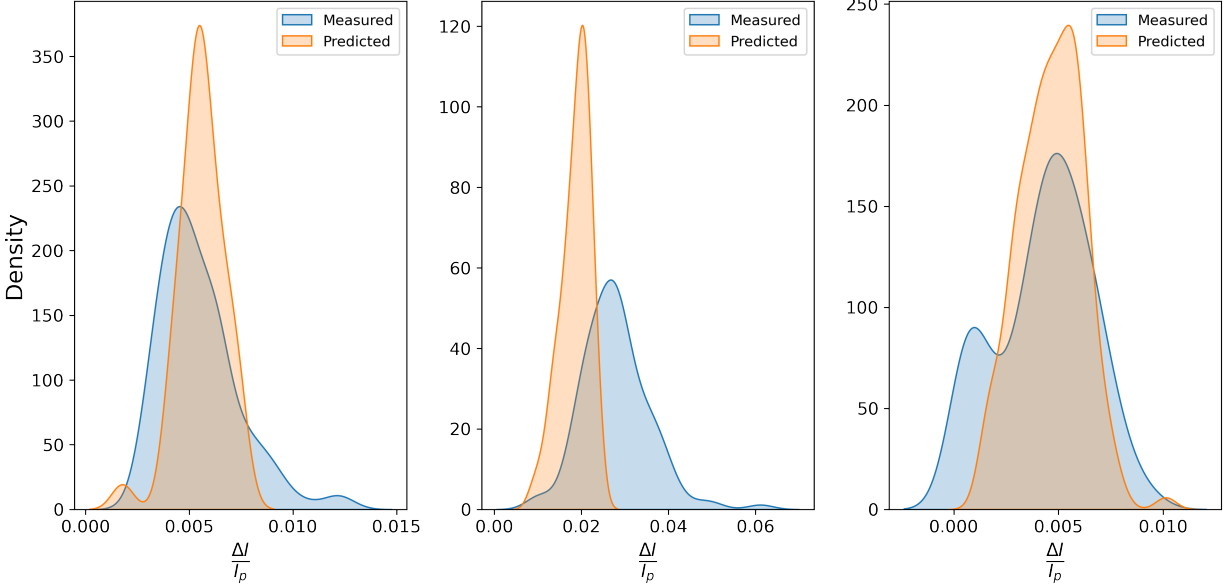
Figure 5.7: **Predicted RP amplitude distributions vs measured.** a) RP amplitude distributions for E. Coli. b) RP amplitude distributions for E. Coli 2. c) RP amplitude distributions of S. Aureus.

cell populations, and see how they compare to the predicted values calculated by observing the shapes optically. We can see that for the E. coli we predict a mean amplitude which is similar to the measured RP amplitude, whereas the predicted mean amplitude for the E. coli 2 is smaller than the measured mean. The discrepancies in the predicted values could be a result of to the eq. 5.2 being derived for spherical particles, whereas the bacteria are better described as prolate ellipsoids with aspects ratios ranging from 1 - 3. However, for S. aureus, we can see that there is good agreement between measured and predicted values, since these bacteria can be described as spheres, i.e. their aspect ratios are ~ 1. For the measured RP amplitudes, we see that the S. aureus has a bimodal distribution. This is due to spherical bacteria having a tendency of forming clusters of two (as seen in microscope images), and therefore we observe both single bacteria as well as pairs.

5.3.4 Classification of bacterial species using Gaussian Mixture-Models

If we plot the RP amplitudes versus the transit time for each bacteria species, we can see distinct clusters appear for each population, where we label the ground truth from independent experimental runs. We are at first interested in how well we can classify the three populations using just these two features, which depend on particle volume and passage time. Gaussian mixture-models (GMM), discussed previously, were implemented to perform unsupervised classification. To determine the classification accuracy, we can use the known labels, and determine how many were correctly identified from the clustering performed by the GMM. When assuming the number of clusters = 3, we find that the GMM is able to produce a score of 95% accuracy. Interestingly, when using four cluster components, we find similar accuracy scores. The fourth identified clusters appear to be outliers in event time, and could be explained by E. Coli events where two bacteria are traveling through the pore simultaneously. We can see in figure 5.8, the true labels for the bacteria populations and the label learned from the GMM. The confusion matrix (figure 5.9) displays the number of correctly identified bacteria in each population, normalized by the total count of bacteria in the true population.

While the GMM was able to classify the three populations with moderate success, the features used (RP amplitude and transit time) do not utilize the full information contained within the signal. The signal averaging of the undulating RP amplitude from the particle length is not accounted for in these two features. Next, we will use an alternative clustering method which classifies time-series based upon shape, rather than amplitude and time lengths.

Figure 5.8: **GMM classification of bacteria populations** a)Scatterplot of E. Coli, E. Coli 2. S. Aureus of RP amplitude vs event time. Cells are labeled by experimental ground truth. b) Predicted labels from unsupervised GMM using RP amplitude and event time as features.
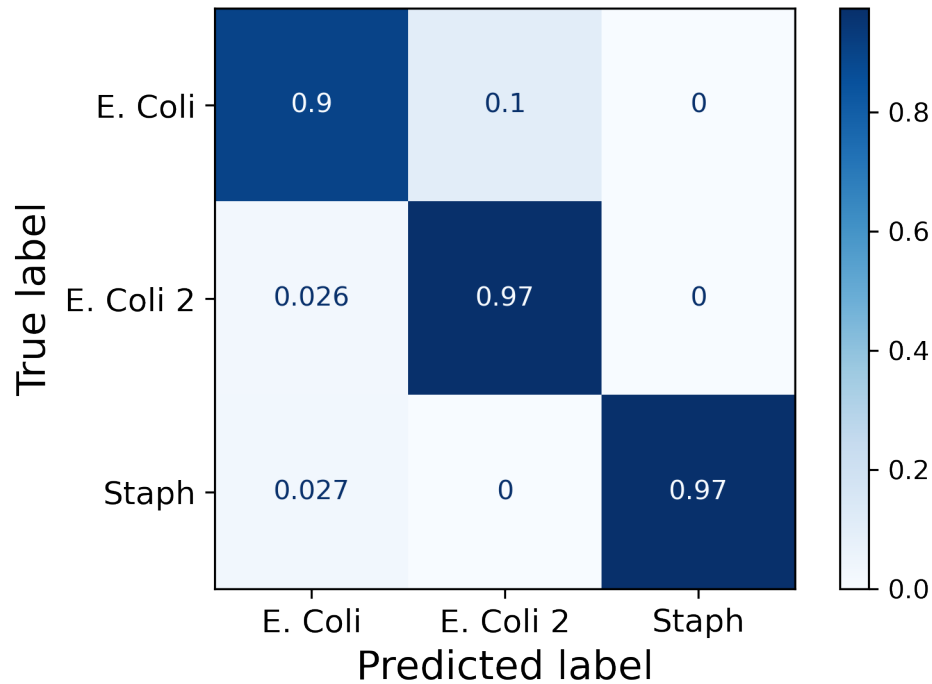
Figure 5.9: **Confusion matrix for GMM clustering** Horizontal and vertical axis of confusion matrix represent the GMM predicted label and ground truth label, respectively. Values are normalized across row by the total sample number of each true label. Diagonal elements represent the correct prediction accuracies for each class. Off diagonal represent incorrect cluster assignments.

5.3.5 K-Means with Dynamic Time Warping for clustering from Signal Shape

An alternative to clustering time-series by feature analysis is to cluster time-series which are similar in shape. Euclidean distance seems to be the natural choice to measure of distance between two signals. However, euclidean distance fails as a similarity measure if the signals occur over different time scales i.e. comparing sine waves of different periods. One approach to finding a similarity between two time-series is through the use of dynamic time warping (DTW). DTW is able to match signals which occur over different time scales, but share the same structure. We can utilize DTW as a means for clustering time-series based upon their similarity. Bacteria of similar length scales will produce a unique RP signal that average the undulations found in the PET pores. By leveraging the particle averaging and DTW, we can attempt to cluster based off the signal shape alone.

Here we will utilize k-means clustering together with DTW. Note that k-means contains the following optimization steps: 1. Find distance from each point to cluster center 2. Assign closest cluster to each data point 3. Recalculate mean of newly formed cluster. This process is performed iteratively until convergence is reached (clusters centers remain stationary). Steps 1 and 2 can be accomplished through measuring DTW distance and for cluster assignment. However, finding a time-series which is mean representative of a population of time-series is not trivial. One approach is to calculate pair-wise averaging, however this creates an average which is order dependent. Petitjean et. al [94, 95] created an algorithm known as DTW Barrycenter Averaging (DBA), which is not sensitive to ordering, and utilizes DTW to calculate an average of a set of time-series. We demonstrate the use of DBA by averaging the RP signals of each bacteria species (figure 5.10). We can see that DBA captures the unique shaped caused by each bacteria species.

By utilizing DBA and DTW, k-means clustering can be used on the raw time signal without
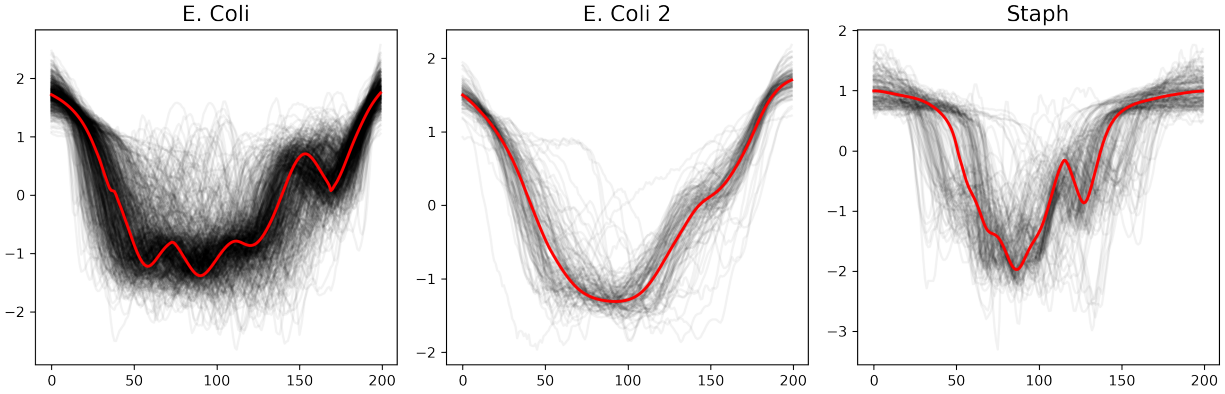
Figure 5.10: **Dynamic time-warping barrycenter averaging of RP signals** Signals were compressed to be of equal length and scaled to have a mean of zero and variance of one. a) The average signal of E. Coli computed using DBA b) Average signal of E. Coli. 2 computed using DBA. c) Average signal of S. aureus using DBA

any feature extraction. While DBA and DTW can be used for averaging signals and computing similarities, computing theses values can be very computationally expensive, both in time and space complexity, where the time complexity grows quadratically with sequence length. To make these studies computationally feasible, each time-series event was scaled down in time by a factor of 10, preserving relative feature lengths. Scaling all series to a constant length showed similar classification results. Additionally, the amplitude of each sequence was scaled to have a mean of zero and variance of one. K-means classification with DBA was performed utilizing the tslearn package in Python 3.6. To visualize the performance of k-means DBA clustering, we used the learned clusters and labeled the RP amplitude vs event time graph. We see that k-means DBA is able to capture the general data structure, where nearly all of the E Coli. 2 and Staph populations are identified correctly. However, this algorithm falsely identifies E. Coli as E. Coli 2 at a significant rate. To understand this phenomena further work will have to be done to explore these overlapping populations.

From the confusion matrix presented in 5.12, we can see that the k-means DBA algorithm overall performs worse in terms of classification accuracy as compared to using GMM with the two derived features (ΔI and event time). It seems from these conclusions that the unique

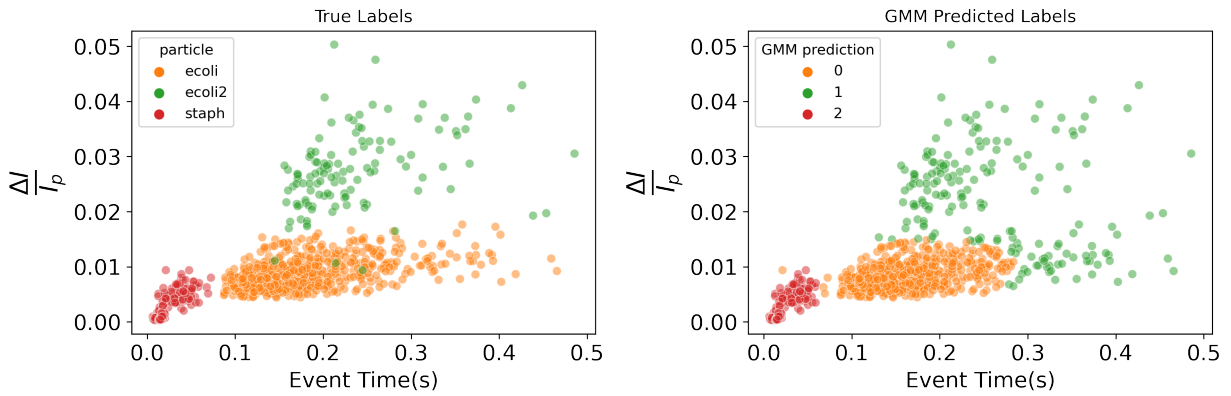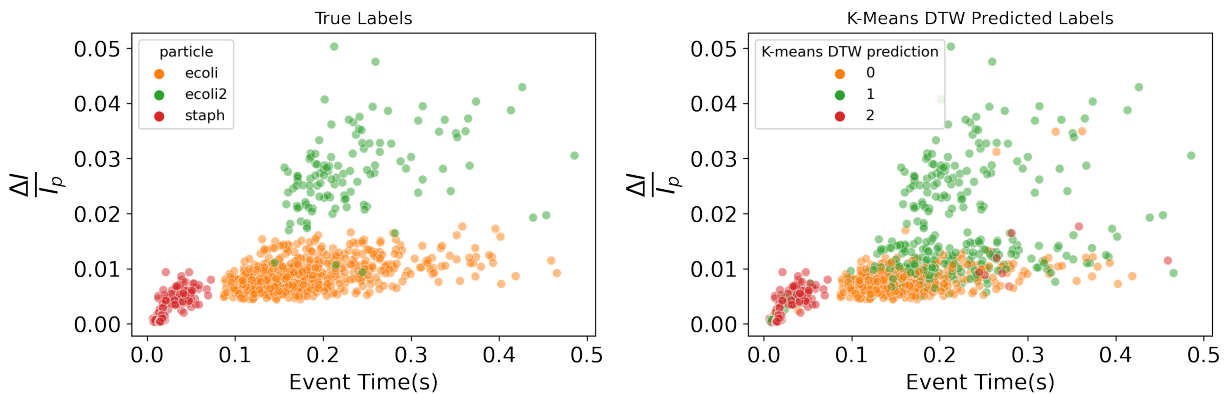Figure 5.11: **DBA k-means classification of bacteria populations** a)Scatterplot of E. Coli, E. Coli 2. S. Aureus of RP amplitude vs event time. Cells are labeled by experimental ground truth. b) Predicted labels from unsupervised DBA k-means using event signals as input data

shape created by particle averaging adds no intrinsic value to classification. However, solution conditions can greatly alter transit time through changing surface potentials. In such a case, the GMM approach would classify objects of similar shapes into different clusters. Whereas by classification by shape analysis, as done by k-means DBA, the stretching/shrinking of event times would have no impact on classifying particles of similar shapes.

5.3.6 Detection of bacteria populations after antibiotic treatment

Here, we will demonstrate a use case of the previously described methods to identify how a bacteria population shifts after treatment with antibiotics. As bacteria continue adapt to become resistant to antibiotic treatments, new antibiotics must be developed. Rapidly understanding how antibiotics treatments physically change bacteria can be used for determining susceptibility, and guide the development of new antibiotics. To test the potential of our technique, we created two bacteria populations, untreated E. Coli, and treated E. Coli which have been incubated with 10 μM Cephalexin, an antibiotic which inhibits growth. For this experiment, bacteria was suspended in MinA+ media, and a PET pore of 36 μm in length was chemically etched. Interestingly, events were observed with an applied voltage of
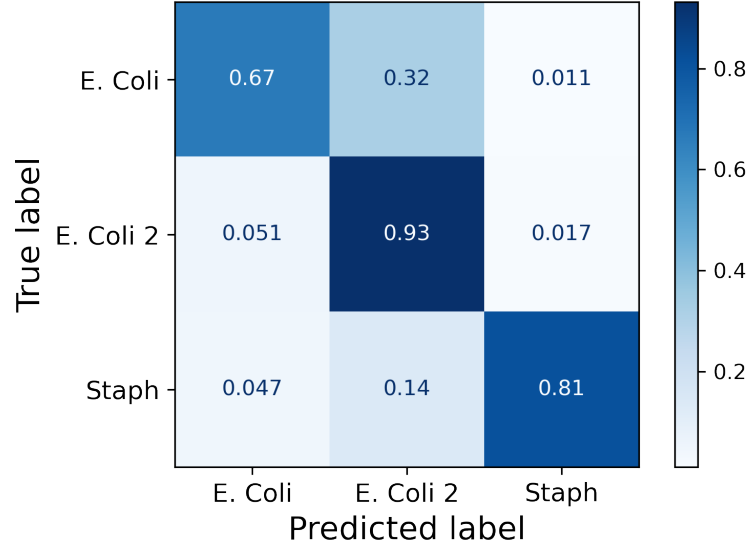
Figure 5.12: **Confusion matrix for DBA k-means clustering** Horizontal and vertical axis of confusion matrix represent the DBA k-means predicted label and ground truth label, respectively.

-0.1V, indicating that the particles became neutralized and traveled through electo-osmosis, or became positively charged and by electro-phoresis though this is unlikely as positively charged bacteria would stick to the pore surface during transit. This effect is a result from suspending the bacteria in minA media instead of pbs. We find the the distribution of RP amplitudes (ΔI_p) for both populations are similar, and that the populations differ significantly in event time. Further analysis will have to be performed to understand the shift in event time between the two populations.

When performing clustering through the GMM approach, we find that we can classify the populations at very high rates, with an accuracy of 92 % and 100 % for the untreated and treated populations, respectively. When using the shape based approach, namely k-means with DBA, we again see lower classification potential as compared to using the GMM, resulting in accuracies of of 83% and 90%. When looking at the k-means DBA identified labels in the figure 5.15b, we see that the algorithm classifies points within the untreated cluster as samples belong to the treated cluster. While in both instances the shape based
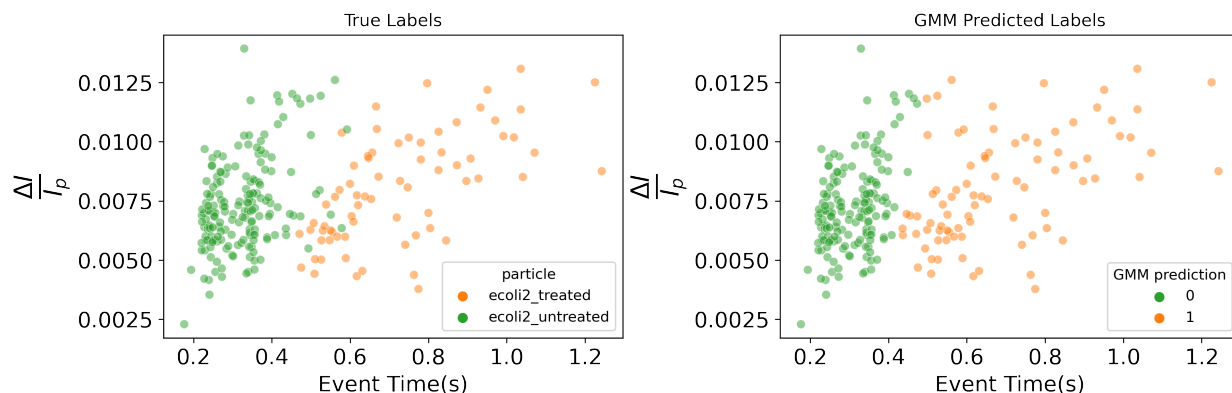
Figure 5.13: **GMM classification of antibiotic treated bacteria** a)Scatterplot of untreated E. Coli and Cephalexin treated E. Coli of RP amplitude vs event time. Cells are labeled by experimental ground truth. b) Predicted labels from unsupervised GMM using RP amplitude and event time as features.

classification approach performs worse than using GMM approach, there are still applications where using k-means DBA could be attractive. We can see the neutralizing of surface potential from suspending the bacteria in MinA+ media as opposed to PBS, demonstrating that the ions present in the liquid in which the samples are suspended can effect the surface potential of the bacteria in magnitude or sign. This change in surface potential, results in changes in event transit time. As a result, the GMM approach is most useful when the species understudy are suspended with the same solution conditions. However, if one was to introduce new ions or molecules to the suspended cells, the shaped based approach is independent of changes in surface potential and event transit time; only the shape of the cell contributes to the averaging of the RP signal.

5.4 Discussion and conclusions

We have shown the ability to use resistive-pulse sensing to detect bacteria populations of different shapes and surface potentials in PET micropores, and classified them based upon these features to high accuracy. Furthermore, we have compared this to an alternative
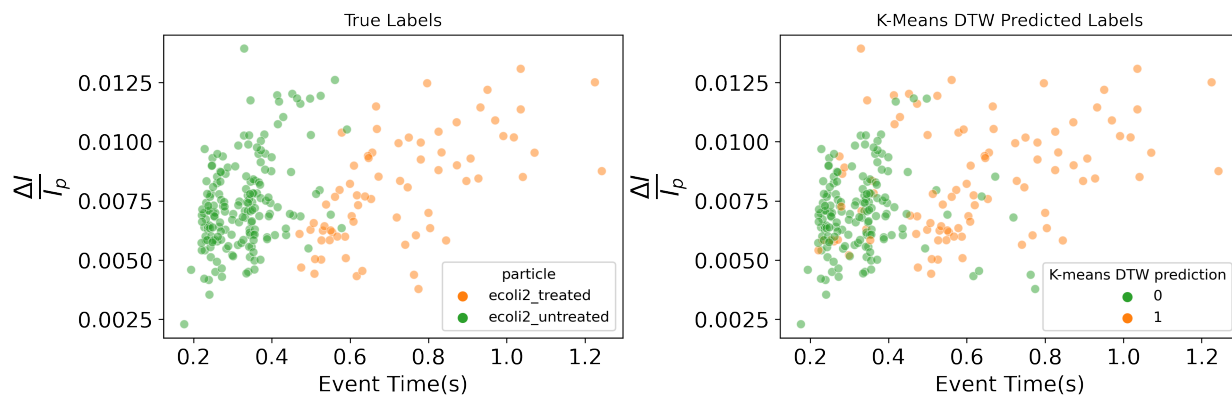
Figure 5.14: **DBA k-means classification of antibiotic treated bacteria** a)Scatterplot of untreated E. Coli and Cephalexin treated E. Coli of RP amplitude vs event time. Cells are labeled by experimental ground truth. b) Predicted labels from DBA k-means using event signals as input data

(a) GMM prediction

(b) DBA k-means prediction
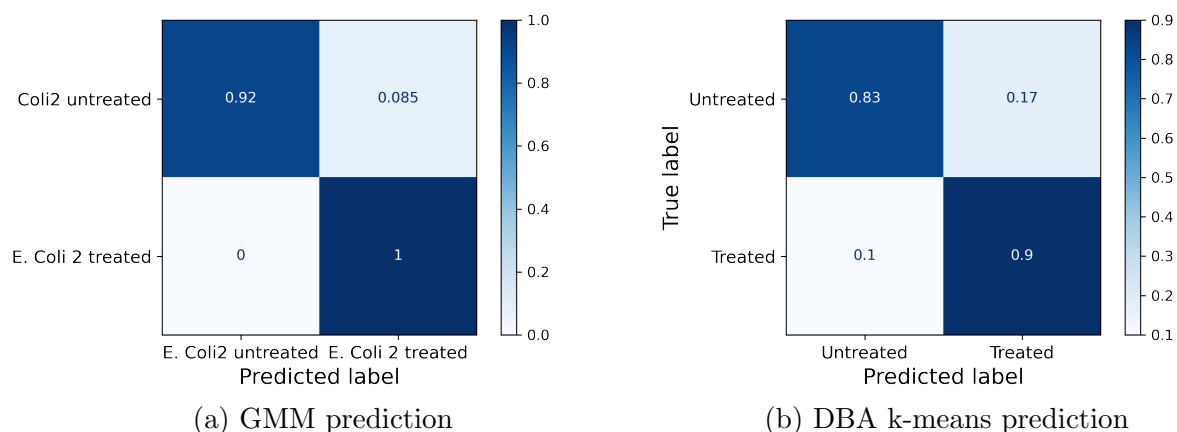
Figure 5.15: **GMM vs DBA k-means classification of antibiotic treated E. Coli**

approach which leverages the undulating diameter in PET pores occuring from chemical etching, resulting in a unique RP 'signature'. This RP signature becomes averaged depending on the size and length of the passing particle, resulting in different signal shapes. We utilized this effect together with dynamic time-warping barrycenter averaging and k-means to cluster cells based upon their shape. While this work was performed at the micron scale on bacteria, the same principles can be scaled down to other biological analytes such as viruses, exosomes, and even proteins. Methods for developing 'rough' channels on the smaller scale will have to be explored to make use of the shape based averaging.

97

Chapter 6

Conclusions

To summarize the work presented, we have developed two different mircofluidic platforms to characterize mammalian and bacteria cells based on their physical properties such as mechanics, shape, and surface charge. To begin we have explained the physical backing for fluid base mechanical stretchers, as well as demonstrated the use of FEM modeling to simulate the fluid dynamics in more complicated microfluidic geometries. We then provided an overview of machine learning models and deep learning architectures which are utilized in this thesis. The following chapter describes the application of dynamic deformability cytometry with deep learning for the classification of mammalian cells. Thereafter, we provide two examples where this method has shown promise. In the last chapter, we present a an electrical impedance based platform for the classification of bacteria cells based upon their shape and surface potential.

6.1 Deep learning applied to dynamic deformabilty cytometry

Single cell microfluidics stretchers have emerged recently as way to discover and classify cell populations, leading to a new form of cytometry known as deformability cytometry. We have expanded upon the previous work in this field through a combination of the following innovations. We have demonstrated the ability to track the deformation dynamics of single cells to a high spatial and temporal resolution. By subjecting cells to non-linear stresses and tracking the deformation dynamics, frequency based mechanical information is revealed. Through establishing control subpopulations of HL60 cells, each treated with a different cytoskeletal perturbing drug, we are able to determine how much information is gained through these methods. We utilized traditional machine learning algorithms, along with image sequence based deep learning architectures and found classification accuracies approaching 90% on mechanical features alone. We then demonstrated brief application of these methods for the use of immune cells and lipochondrocytes.

It is my hope that the work presented in this chapter opens up the application of deep learning to mechanotyping of single cells. The methods presented here are not necessarily optimized, and there are ample opportunities for improvement in both microfluidic channel design, deep learning, or the combination of both. There are also many opportunities to improve the methods presented here. Enabling real-time image analysis for continuous acquisition would enable large amounts of data to be acquired in a short time period. This would result in better trained models and insights into rare cell types, as well as open up the possibility for real-time sorting of cells based upon mechanical properties. Other experimental improvements include measuring further physical features which are perpendicular to mechanics simultaneously, such as electrical measurements of capacitance. There are also a great deal of machine learning possibilities for improvement, mainly used unsupervised approaches for detection

of unknown cell populations. While we have shown the examples of detecting immune cell populations, there remains much work at the intersection of biology and mechanics to find problems where this tool would be most useful, either in a lab setting or as a clinical tool for detection. The vision is for deformability cytometry to be integrated into current biological tool sets such as imaging flow cytometry or even flow cytometry.

6.2 Classification of bacteria by shape, size, and surface properties

Detection of bacterial species has direct implication for global healthcare and focused treatments. Sepsis, in particular, is marked by the infection of the blood by bacteria and where a diagnosis represents nearly a 20% mortality rate in the US. Furthermore, as antibiotics become over prescribed, antibiotic resistant bacteria pose a major concern for treatment of infections. Monitoring the physical properties of bacteria provides a rapid label-free test for determining the susceptibility of bacteria to treatments. We have presented the ability to distinguish between bacteria populations which differ in shape, size, and surface potential through resistive-pulse sensing using PET micropores. We compare this method to another which utilizes the natural undulations in PET pores with particle averaging to classify bacteria based upon their unique resistive-pulse signals. It is the hope that these two methods can be expanded to further tests with antibiotic treated bacteria. There is also the possibility to scaling down the technology for classification of virus and even proteins.

Bibliography

[1] Wikimedia Commons. Compressed (left) and unfolded (right) basic recurrent neural network., 2017.

[2] Alexander LeNail. Nn-svg: Publication-ready neural network architecture schematics. *J. Open Source Softw.*, 4(33):747, 2019.

[3] Chia-Yen Lee, Chin-Lung Chang, Yao-Nan Wang, and Lung-Ming Fu. Microfluidic mixing: a review. *International journal of molecular sciences*, 12(5):3263–3287, 2011.

[4] Daniel R Gossett, Westbrook M Weaver, Albert J Mach, Soojung Claire Hur, Henry Tat Kwong Tse, Wonhee Lee, Hamed Amini, and Dino Di Carlo. Label-free cell separation and sorting in microfluidic systems. *Analytical and bioanalytical chemistry*, 397(8):3249–3267, 2010.

[5] Andreas Lenshof and Thomas Laurell. Continuous separation of cells and particles in microfluidic systems. *Chemical Society Reviews*, 39(3):1203–1217, 2010.

[6] Boyang Zhang, Anastasia Korolj, Benjamin Fook Lun Lai, and Milica Radisic. Advances in organ-on-a-chip engineering. *Nature Reviews Materials*, 3(8):257–278, 2018.

[7] Brian J Kirby. *Micro-and nanoscale fluid mechanics: transport in microfluidic devices.* Cambridge university press, 2010.

[8] Á V Delgado, F González-Caballero, RJ Hunter, LK Koopal, and J Lyklema. Measurement and interpretation of electrokinetic phenomena. *Journal of colloid and interface science*, 309(2):194–224, 2007.

[9] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[10] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

[11] Ewa Guzniczak, Maryam Mohammad Zadeh, Fiona Dempsey, Melanie Jimenez, Henry Bock, Graeme Whyte, Nicholas Willoughby, and Helen Bridle. High-throughput assessment of mechanical properties of stem cell derived red blood cells, toward cellular downstream processing. *Sci. Rep.*, 7(1):1–11, 2017.

[12] Jochen Guck, Stefan Schinkinger, Bryan Lincoln, Falk Wottawah, Susanne Ebert, Maren Romeyke, Dominik Lenz, Harold M Erickson, Revathi Ananthakrishnan, Daniel Mitchell, et al. Optical deformability as an inherent cell marker for testing malignant transformation and metastatic competence. *Biophys. J.*, 88(5):3689–3698, 2005.

[13] Aaron P Mosier, Alain E Kaloyeros, and Nathaniel C Cady. A novel microfluidic device for the in situ optical and mechanical analysis of bacterial biofilms. *J. Microbiol. Methods*, 91(1):198–204, 2012.

[14] Markéta Kubánková, Bettina Hohberger, Jakob Hoffmanns, Julia Fürst, Martin Hermann, Jochen Guck, and Martin Krater. Physical phenotype of blood cells is altered in covid-19. *bioRxiv*, 2021.

[15] Albercht L Weisenhorn, Mitra Khorsandi, Sandor Kasas, Vassilis Gotzos, and H-J Butt. Deformation and height anomaly of soft surfaces studied with an afm. *Nanotechnology*, 4(2):106, 1993.

[16] Manfred Radmacher, Monika Fritz, Claudia M Kacher, Jason P Cleveland, and Paul K Hansma. Measuring the viscoelastic properties of human platelets with the atomic force microscope. *Biophys. J.*, 1996.

[17] Jianwu Dai and Michael P Sheetz. Mechanical properties of neuronal growth cone membranes studied by tether formation with laser optical tweezers. *Biophys. J.*, 68(3):988–996, 1995.

[18] EA Evans. New membrane concept applied to the analysis of fluid shear-and micropipette-deformed red blood cells. *Biophys. J.*, 13(9):941–954, 1973.

[19] Waseem Asghar, Yuan Wan, Azhar Ilyas, Robert Bachoo, Young-tae Kim, and Samir M Iqbal. Electrical fingerprinting, 3d profiling and detection of tumor cells with solid-state micropores. *Lab Chip*, 12(13):2345–2352, 2012.

[20] Junghyun Kim, Sewoon Han, Andy Lei, Masaru Miyano, Jessica Bloom, Vasudha Srivastava, Martha R Stampfer, Zev J Gartner, Mark A LaBarge, and Lydia L Sohn. Characterizing cellular mechanical phenotypes with mechano-node-pore sensing. *Microsyst. & Nanoeng.*, 4(1):1–12, 2018.

[21] Sangwon Byun, Sungmin Son, Dario Amodei, Nathan Cermak, Josephine Shaw, Joon Ho Kang, Vivian C Hecht, Monte M Winslow, Tyler Jacks, Parag Mallick, et al. Characterizing deformability and surface friction of cancer cells. *Proc. Natl. Acad. Sci. U.S.A.*, 110(19):7580–7585, 2013.

[22] Ying Zhou, Dahou Yang, Yinning Zhou, Bee Luan Khoo, Jongyoon Han, and Ye Ai. Characterizing deformability and electrical impedance of cancer cells in a microfluidic device. *Anal. Chem.*, 90(1):912–919, 2018.

[23] Andrew C Hodgson, Christophe M Verstreken, Cynthia L Fisher, Ulrich F Keyser, Stefano Pagliara, and Kevin J Chalut. A microfluidic device for characterizing nuclear deformations. *Lab on a Chip*, 17(5):805–813, 2017.

[24] Jian Chen, Yi Zheng, Qingyuan Tan, Ehsan Shojaei-Baghini, Yan Liang Zhang, Jason Li, Preethy Prasad, Lidan You, Xiao Yu Wu, and Yu Sun. Classification of cell types using a microfluidic device for mechanical and electrical measurement on single cells. *Lab on a Chip*, 11(18):3174–3181, 2011.

[25] Han Wei Hou, QS Li, GYH Lee, AP Kumar, CN Ong, and Chwee Teck Lim. Deformability study of breast cancer cells using microfluidics. *Biomedical microdevices*, 11(3):557–564, 2009.

[26] Xiaonan Yang, Zhuofa Chen, Jun Miao, Liwang Cui, and Weihua Guan. High-throughput and label-free parasitemia quantification and stage differentiation for malaria-infected red blood cells. *Biosensors and Bioelectronics*, 98:408–414, 2017.

[27] Andrea Adamo, Armon Sharei, Luigi Adamo, ByungKun Lee, Shirley Mao, and Klavs F Jensen. Microfluidics-based assessment of cell deformability. *Analytical chemistry*, 84(15):6438–6443, 2012.

[28] Xiang Ren, Parham Ghassemi, Hesam Babahosseini, Jeannine S Strobl, and Masoud Agah. Single-cell mechanical characteristics analyzed by multiconstriction microfluidic channels. *ACS sensors*, 2(2):290–299, 2017.

[29] Yi Zheng, John Nguyen, Chen Wang, and Yu Sun. Electrical measurement of red blood cell deformability on a microfluidic device. *Lab on a Chip*, 13(16):3275–3283, 2013.

[30] Chia-Hung Dylan Tsai, Shinya Sakuma, Fumihito Arai, and Makoto Kaneko. A new dimensionless index for evaluating cell stiffness-based deformability in microchannel. *IEEE transactions on Biomedical Engineering*, 61(4):1187–1195, 2014.

[31] Kendra D Nyberg, Samuel L Bruce, Angelyn V Nguyen, Clara K Chan, Navjot K Gill, Tae-Hyung Kim, Erica K Sloan, and Amy C Rowat. Predicting cancer cell invasion by single-cell physical phenotyping. *Integr. Biol.*, 10(4):218–231, 2018.

[32] Kendra D Nyberg, Kenneth H Hu, Sara H Kleinman, Damir B Khismatullin, Manish J Butte, and Amy C Rowat. Quantitative deformability cytometry: rapid, calibrated measurements of cell mechanical properties. *Biophys. J.*, 113(7):1574–1584, 2017.

[33] Daniel R Gossett, TK Henry, Serena A Lee, Yong Ying, Anne G Lindgren, Otto O Yang, Jianyu Rao, Amander T Clark, and Dino Di Carlo. Hydrodynamic stretching of single cells for large population mechanical phenotyping. *Proc. Natl. Acad. Sci. U.S.A.*, 109(20):7630–7635, 2012.

[34] Oliver Otto, Philipp Rosendahl, Alexander Mietke, Stefan Golfier, Christoph Herold, Daniel Klaue, Salvatore Girardo, Stefano Pagliara, Andrew Ekpenyong, Angela Jacobi, et al. Real-time deformability cytometry: on-the-fly cell mechanical phenotyping. *Nat. Methods*, 12(3):199–202, 2015.

[35] Jonathan Lin, Donghyuk Kim, T Tse Henry, Peter Tseng, Lillian Peng, Manjima Dhar, Saravanan Karumbayaram, and Dino Di Carlo. High-throughput physical phenotyping of cell differentiation. *Microsyst. & Nanoeng.*, 3(1):1–7, 2017.

[36] Mahdokht Masaeli, Dewal Gupta, Sean O'Byrne, TK Henry, Daniel R Gossett, Peter Tseng, Andrew S Utada, Hea-Jin Jung, Stephen Young, Amander T Clark, et al. Multi-parameter mechanical and morphometric screening of cells. *Scientific reports*, 6(1):1–11, 2016.

[37] Alexander Mietke, Oliver Otto, Salvatore Girardo, Philipp Rosendahl, Anna Taubenberger, Stefan Golfier, Elke Ulbricht, Sebastian Aland, Jochen Guck, and Elisabeth Fischer-Friedrich. Extracting cell stiffness from real-time deformability cytometry: theory and experiment. *Biophysical journal*, 109(10):2023–2036, 2015.

[38] Bob Fregin, Fabian Czerwinski, Doreen Biedenweg, Salvatore Girardo, Stefan Gross, Konstanze Aurich, and Oliver Otto. High-throughput single-cell rheology in complex samples by dynamic real-time deformability cytometry. *Nat. Commun.*, 10(1):1–11, 2019.

[39] Yan Ge, Philipp Rosendahl, Claudio Durán, Nicole Töpfner, Sara Ciucci, Jochen Guck, and Carlo Vittorio Cannistraci. Cell mechanics based computational classification of red blood cells via machine intelligence applied to morpho-rheological markers. *IEEE/ACM transactions on computational biology and bioinformatics*, 2019.

[40] Jason Riordon, Dušan Sovilj, Scott Sanner, David Sinton, and Edmond WK Young. Deep learning with microfluidics for biotechnology. *Trends in biotechnology*, 37(3):310–324, 2019.

[41] Shaobo Luo, Yuzhi Shi, Lip Ket Chin, Paul Edward Hutchinson, Yi Zhang, Giovanni Chierchia, Hugues Talbot, Xudong Jiang, Tarik Bourouina, and Ai-Qun Liu. Machine-learning-assisted intelligent imaging flow cytometry: A review. *Advanced Intelligent Systems*, 3(11):2100073, 2021.

[42] Preston Hinkle, Trisha M Westerhof, Yinghua Qiu, David J Mallin, Matthew L Wallace, Edward L Nelson, Peter Taborek, and Zuzanna S Siwy. A hybrid resistive pulse-optical detection platform for microfluidic experiments. *Sci. Rep.*, 7(1):1–14, 2017.

[43] Marta Urbanska, Hector E Muñoz, Josephine Shaw Bagnall, Oliver Otto, Scott R Manalis, Dino Di Carlo, and Jochen Guck. A comparison of microfluidic methods for high-throughput cell deformability measurements. *Nat. Methods*, 17(6):587–593, 2020.

[44] Dahou Yang, Ying Zhou, Yinning Zhou, Jongyoon Han, and Ye Ai. Biophysical phenotyping of single cells using a differential multiconstriction microfluidic device with self-aligned 3d electrodes. *Biosensors and Bioelectronics*, 133:16–23, 2019.

[45] Gihoon Choi, Zifan Tang, and Weihua Guan. Microfluidic high-throughput single-cell mechanotyping: Devices and applications. *Nanotechnology and Precision Engineering*, 4(4):045002, 2021.

[46] Ahmad Ahsan Nawaz, Marta Urbanska, Maik Herbig, Martin Nötzel, Martin Kräter, Philipp Rosendahl, Christoph Herold, Nicole Toepfner, Markéta Kubánková, Ruchi Goswami, et al. Intelligent image-based deformation-assisted cell sorting with molecular specificity. *Nat. Methods*, 17(6):595–599, 2020.

[47] Lloyd Shapley. A value fo n-person games. *Ann. Math. Study28, Contributions to the Theory of Games, ed. by HW Kuhn, and AW Tucker*, pages 307–317, 1953.

[48] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2(1):2522–5839, 2020.

[49] Stefan Golfier, Philipp Rosendahl, Alexander Mietke, Maik Herbig, Jochen Guck, and Oliver Otto. High-throughput cell mechanical phenotyping for label-free titration assays of cytoskeletal modifications. *Cytoskeleton*, 74(8):283–296, 2017.

[50] Waleed Abdulla. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. `https://github.com/matterport/Mask_RCNN`, 2017.

[51] Monica Piergiovanni, Valeria Galli, Gregor Holzner, Stavros Stavrakis, Andrew DeMello, and Gabriele Dubini. Deformation of leukaemia cell lines in hyperbolic microchannels: investigating the role of shear and extensional components. *Lab Chip*, 20(14):2539–2548, 2020.

[52] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[53] John A Cooper. Effects of cytochalasin and phalloidin on actin. *J. Cell Biol.*, 105(4):1473–1478, 1987.

[54] Ali Bashashati, Kenneth Lo, Raphael Gottardo, Randy D Gascoyne, Andrew Weng, and Ryan Brinkman. A pipeline for automated analysis of flow cytometry data: preliminary results on lymphoma sub-type diagnosis. In *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4945–4948. IEEE, 2009.

[55] Tycho M Scholtens, Frederik Schreuder, Sjoerd T Ligthart, Joost F Swennenhuis, Jan Greve, and Leon WMM Terstappen. Automated identification of circulating tumor cells by image cytometry. *Cytom. A.*, 81(2):138–148, 2012.

[56] Mira Valkonen, Kimmo Kartasalo, Kaisa Liimatainen, Matti Nykter, Leena Latonen, and Pekka Ruusuvuori. Metastasis detection from whole slide images using local features and random forests. *Cytom. A.*, 91(6):555–565, 2017.

[57] Carolien Duetz, Sofie Van Gassen, Theresia M Westers, Margot F van Spronsen, Costa Bachas, Yvan Saeys, and Arjan A van de Loosdrecht. Computational flow cytometry as a diagnostic tool in suspected-myelodysplastic syndromes. *Cytometry Part A*, 2021.

[58] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.

[59] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *Preprint at https://arxiv.org/abs/1412.3555*, 2014.

[60] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE transactions on information theory*, 8(2):179–187, 1962.

[61] Laura Keller and Klaus Pantel. Unravelling tumour heterogeneity by single-cell profiling of circulating tumour cells. *Nat. Rev. Cancer*, 19(10):553–567, 2019.

[62] Eduard Batlle and Hans Clevers. Cancer stem cells revisited. *Nat. Med.*, 23(10):1124, 2017.

[63] V Kumar and A Sharma. Neutrophils: Cinderella of innate immune system. *International immunopharmacology*, 10(11):1325–1334, 2010.

[64] Björn Petri and Maria-Jesús Sanz. Neutrophil chemotaxis. *Cell and tissue research*, 371(3):425–436, 2018.

[65] Tanya N Mayadas, Xavier Cullere, and Clifford A Lowell. The multifaceted functions of neutrophils. *Annual Review of Pathology: Mechanisms of Disease*, 9:181–218, 2014.

[66] Robin C May and Laura M Machesky. Phagocytosis and the actin cytoskeleton. *Journal of cell science*, 114(6):1061–1077, 2001.

[67] Andrew E Ekpenyong, Nicole Toepfner, Edwin R Chilvers, and Jochen Guck. Mechanotransduction in neutrophil activation and deactivation. *Biochimica et Biophysica Acta (BBA)-Molecular Cell Research*, 1853(11):3105–3116, 2015.

[68] Andrew E Ekpenyong, Nicole Toepfner, Christine Fiddler, Maik Herbig, Wenhong Li, Gheorghe Cojoc, Charlotte Summers, Jochen Guck, and Edwin R Chilvers. Mechanical deformation induces depolarization of neutrophils. *Science advances*, 3(6):e1602536, 2017.

[69] Charles N Serhan, M Johan Broekman, Helen M Korchak, Aaron J Marcus, and Gerald Weissman. Endogenous phospholipid metabolism in stimulated neutrophils differential activation by fmlp and pma. *Biochemical and Biophysical Research Communications*, 107(3):951–958, 1982.

[70] Albert Siryaporn and Mark Goulian. Cross-talk suppression between the cpxa–cpxr and envz–ompr two-component systems in e. coli. *Molecular microbiology*, 70(2):494–506, 2008.

[71] Fan Feng, Li Sun, Gaozan Zheng, Shushang Liu, Zhen Liu, Guanghui Xu, Man Guo, Xiao Lian, Daiming Fan, and Hongwei Zhang. Low lymphocyte-to-white blood cell ratio and high monocyte-to-white blood cell ratio predict poor prognosis in gastric cancer. *Oncotarget*, 8(3):5281, 2017.

[72] Hendrik JM de Jonge, Peter JM Valk, Eveline SJM de Bont, Jan Jacob Schuringa, Gert Ossenkoppele, Edo Vellenga, and Gerwin Huls. Prognostic impact of white blood cell count in intermediate risk acute myeloid leukemia: relevance of mutated npm1 and flt3-itd. *haematologica*, 96(9):1310, 2011.

[73] Xiwei Huang, Hyungkook Jeon, Jixuan Liu, Jiangfan Yao, Maoyu Wei, Wentao Han, Jin Chen, Lingling Sun, and Jongyoon Han. Deep-learning based label-free classification of activated and inactivated neutrophils for rapid immune state monitoring. *Sensors*, 21(02):512, 2021.

[74] R Mallinger and P Böck. Differentiation of extracellular matrix in the cellular cartilage ("zellknorpel") of the mouse pinna. *Anatomy and embryology*, 172(1):69–74, 1985.

[75] Isamu Ito, Masato Imada, Minoru Ikeda, Kouhei Sueno, Tomio Arikuni, and Akinori Kida. A morphological study of age changes in adult human auricular cartilage with special emphasis on elastic fibers. *The Laryngoscope*, 111(5):881–886, 2001.

[76] Charles F Sanzone and Edward J Reith. The development of the elastic cartilage of the mouse pinna. *American Journal of Anatomy*, 146(1):31–71, 1976.

[77] Sojib Bin Zaman, Muhammed Awlad Hussain, Rachel Nye, Varshil Mehta, Kazi Taib Mamun, and Naznin Hossain. A review on antibiotic resistance: alarm bells are ringing. *Cureus*, 9(6), 2017.

[78] Dawn Field, Jennifer Hughes, and E Richard Moxon. Using the genome to understand pathogenicity. In *Genomics, Proteomics, and Clinical Bacteriology*, pages 261–287. Springer, 2004.

[79] George K Auer and Douglas B Weibel. Bacterial cell mechanics. *Biochemistry*, 56(29):3710–3724, 2017.

[80] Dawid Połap and Marcin Woźniak. Bacteria shape classification by the use of region covariance and convolutional neural network. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2019.

[81] Matthew T Cabeen and Christine Jacobs-Wagner. Bacterial cell shape. *Nature Reviews Microbiology*, 3(8):601–610, 2005.

[82] Noor Amaleena Mohamad, Noorain A Jusoh, Zaw Zaw Htike, Shoon Lei Win, et al. Bacteria identification from microscopic morphology: a survey. *International Journal on Soft Computing, Artificial Intelligence and Applications (IJSCAI)*, 3(1):2319–1015, 2014.

[83] Shota Hattori, Rintaro Sekido, Iat Wai Leong, Makusu Tsutsui, Akihide Arima, Masayoshi Tanaka, Kazumichi Yokota, Takashi Washio, Tomoji Kawai, and Mina Okochi. Machine learning-driven electronic identifications of single pathogenic bacteria. *Scientific reports*, 10(1):1–10, 2020.

[84] Hagan Bayley and Charles R Martin. Resistive-pulse sensing from microbes to molecules. *Chemical reviews*, 100(7):2575–2594, 2000.

[85] Tao Sun and Hywel Morgan. Single-cell microfluidic impedance cytometry: a review. *Microfluidics and Nanofluidics*, 8(4):423–443, 2010.

[86] Erik C Yusko, Brandon R Bruhn, Olivia M Eggenberger, Jared Houghtaling, Ryan C Rollings, Nathan C Walsh, Santoshi Nandivada, Mariya Pindrus, Adam R Hall, David Sept, et al. Real-time shape approximation and fingerprinting of single proteins using a nanopore. *Nature nanotechnology*, 12(4):360–367, 2017.

[87] DC Golibersuch. Observation of aspherical particle rotation in poiseuille flow via the resistance pulse technique: I. application to human erythrocytes. *Biophysical journal*, 13(3):265–280, 1973.

[88] Yinghua Qiu, Preston Hinkle, Crystal Yang, Henriette E Bakker, Matthew Schiel, Hong Wang, Dmitriy Melnikov, Maria Gracheva, Maria Eugenia Toimil-Molares, Arnout Imhof, et al. Pores with longitudinal irregularities distinguish objects by shape. *ACS nano*, 9(4):4390–4397, 2015.

[89] Matthew Pevarnik, Ken Healy, Maria Eugenia Toimil-Molares, Alan Morrison, Sonia E Létant, and Zuzanna S Siwy. Polystyrene particles reveal pore substructure as they translocate. *Acs Nano*, 6(8):7295–7302, 2012.

[90] W William Wilson, Mary Margaret Wade, Steven C Holman, and Franklin R Champlin. Status of methods for assessing bacterial cell surface charge properties based on zeta potential measurements. *Journal of microbiological methods*, 43(3):153–164, 2001.

[91] Laura M Innes, Chin-Hsuan Chen, Matthew Schiel, Matthew Pevarnik, Florian Haurais, Maria Eugenia Toimil-Molares, Ivan Vlassiouk, Luke Theogarajan, and Zuzanna S Siwy. Velocity profiles in pores with undulating opening diameter and their importance for resistive-pulse experiments. *Analytical chemistry*, 86(20):10445–10453, 2014.

[92] C Chad Harrell, Youngseon Choi, Lloyd P Horne, Lane A Baker, Zuzanna S Siwy, and Charles R Martin. Resistive-pulse dna detection with a conical nanopore sensor. *Langmuir*, 22(25):10837–10843, 2006.

[93] Meni Wanunu. Nanopores: A journey towards dna sequencing. *Physics of life reviews*, 9(2):125–158, 2012.

[94] François Petitjean, Alain Ketterlin, and Pierre Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3):678–693, 2011.

[95] François Petitjean, Germain Forestier, Geoffrey I Webb, Ann E Nicholson, Yanping Chen, and Eamonn Keogh. Dynamic time warping averaging of time series allows faster and more accurate classification. In *Data Mining (ICDM), 2014 IEEE International Conference on*, pages 470–479. IEEE, 2014.

Appendix A

Supplementary figures for Chapter 3

A.1   Calculation of Enrichment

We imagine a use case in which we have a mixed sample of several cell populations, and seek to characterize one rare population of cells within. We begin with n_{sample} cells from the biological sample of interest, perhaps from culture or dissociated tissue. The optimal strategy to analyze the most rare cells would be to characterize every cell in the sample, from which one would recover $r \; n_{\text{sample}}$ rare cells, where r is the fraction of target cells in the sample. However, characterizing every cell to find the rare ones could be prohibitive or even untenable. For example, if one sought a rare population with a frequency of 1/1000, a typical single cell RNA sequencing run of 10,000 cells would yield data on only 10 of the rare cells. If the initial population of cells only numbered 10,000, this would be the best one could do. However, if one started with millions of cells but only had the resources to sequence 10,000 of them, many more of the target rare cells could be characterized if the sample were first sorted for the rare cells.

We imagine a future version of our device which could enrich a sample by mechanically

characterizing the cells, then sorting the output into positives and negatives for a defined profile of the mechanical characteristics of the rare cells. The positive sub-sample would be composed of true positives (TP) and false positives (FP), characterized by the true positive rate TPR and the false positive rate FPR.

Resource limitations dictate that the downstream analysis can accept a limited number of cells, $n_{\text{downstream}} < n_{\text{sample}}$. We seek an enriched sample of size $n_{\text{downstream}}$ which contains the maximum possible number of true positives. To do so, we mechanically characterize n_{RDRUC} cells. Therefore we maximize the number of true positives, $n_{\text{RDRUC}} \; r$ TPR, subject to the constraint $n_{\text{RDRUC}} \leq n_{\text{sample}}$. We find the number of true positives recovered

$$
n_{\text{enriched}} = \begin{cases} n_{\text{sample}} r \text{TPR}, & 0 < n_{\text{sample}} \leq \frac{n_{\text{downstream}}}{\text{FPR} - r\text{FPR} + r\text{TPR}} \\[2ex] \frac{n_{\text{downstream}} r \text{TPR}}{\text{FPR} - r\text{FPR} + r\text{TPR}}, & n_{\text{sample}} > \frac{n_{\text{downstream}}}{\text{FPR} - r\text{FPR} + r\text{TPR}} \end{cases} .
\tag{A.1}
$$

The number of recovered true positives increases with sample size up to a maximum value given by the second case. If an experimenter knows the number of cells possible to analyze downstream and can estimate both the rarity of the cells they seek and classification performance, the sample size threshold could be useful in designing experiments.

Using this result, we now seek to quantify the impact of the classification performance on the quality of the sample passed on from mechanical characterization. We define the maximum enrichment,

$$
\text{enrichment} = \frac{n_{\text{enriched}}}{n_{\text{downstream}} r} = \frac{\text{TPR}}{\text{FPR} - r\text{FPR} + r\text{TPR}}.
\tag{A.2}
$$

We find the maximum enrichment increases must faster than linearly with decreasing FPR, as shown in figure A.1. Therefore, seemingly small increases in classification performance
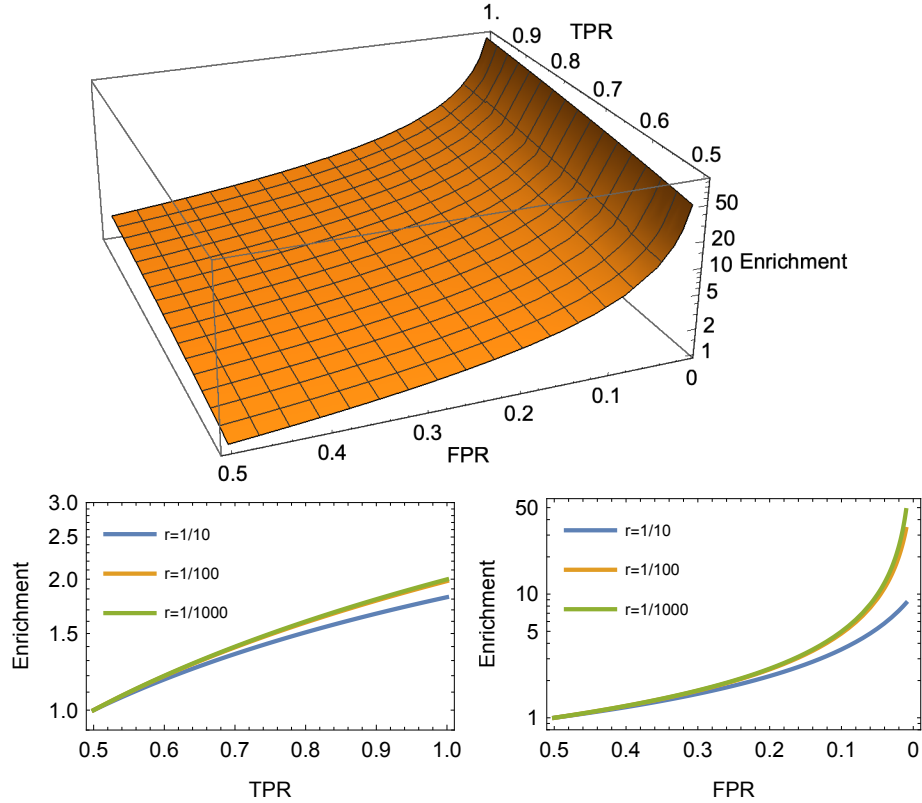
Figure A.1: Maximum enrichment.
Top: Enrichment as a function of both true positive rate and false positive rate for target cell rarity $r = 1/1000$.
Bottom left: Maximum enrichment as a function of TPR shown for FPR fixed at .5 and a range of target cell rarities, r.
Bottom right: Maximum enrichment as a function of FPR shown for TPR fixed at .5 and a range of target cell rarities, r.
Enrichment shown on a log scale in all three subfigures.

can have large effects on the usefulness of a sorter.

Each machine learning model is capable of a range of true positive and false positive rates, visualized as a receiver operating curve (ROC) in figure A.2, left. Because the enrichment is much more sensitive to false positive rate than true positive rate, the maximum enrichment does not necessarily occur at the same point along this curve as the maximum accuracy, where the model is evaluated for the confusion matrices in the main text. To find the best possible max enrichment for a given model, we evaluate max enrichment at each point along the ROC, as shown in figure A.2, right. The greatest enrichment occurs near the edge of the

ROC, where numbers of samples tend to be lower than further along the curve due to high rejection rates. Therefore, we evaluate the best enrichment for a given model by averaging the three highest max enrichment values generated along the ROC curve.
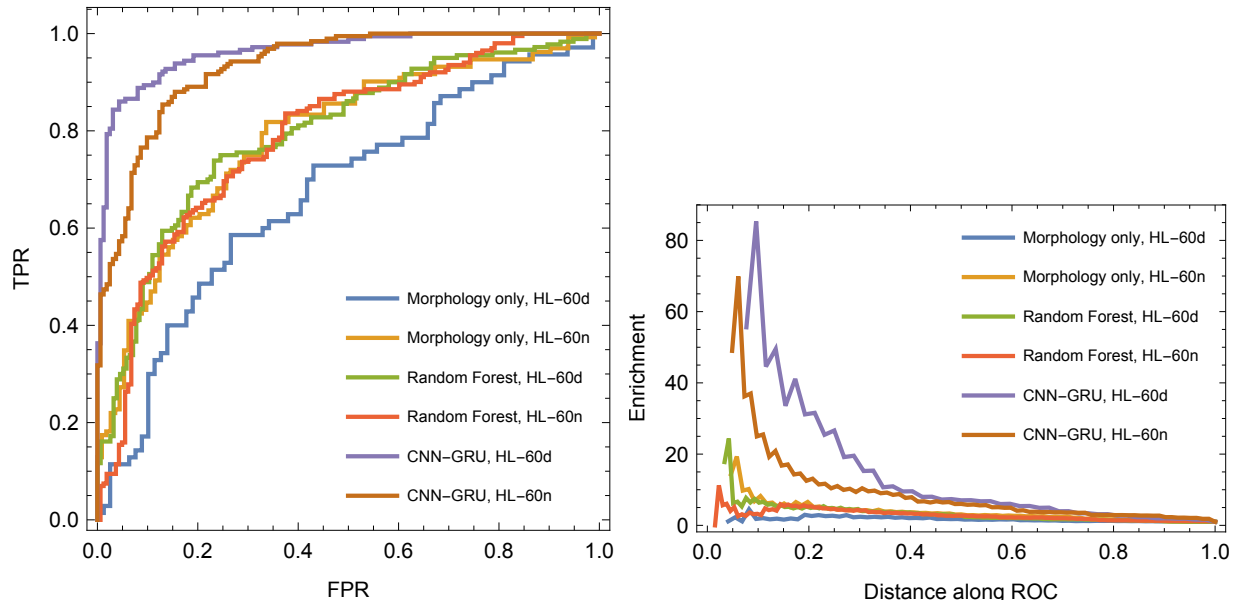

Figure A.2: Evaluation of best enrichment.
Left: Receiver operating curves (ROCs) for several of the models presented in the text.
Right: Max enrichment evaluated for each point along the ROC. Data is excluded for points where true positive rate (TPR) or false positive rate (FPR) were 0. Shown for rarity $r=1/1000$.

We find that for a sorter with the performance of the random forest classifier for HL60d cells (figure 4) has a best maximum enrichment of 17, while a sorter with the accuracy of the CNN-GRU classifier for the same cells (figure 6) has a best maximum enrichment of 63 ($r=1/1000$). This corresponds to an increase from 166 rare cells found to 635 rare cells found for a case where $n_{\text{sample}} = 1,000,000$, $n_{\text{downstream}} = 10,000$, and $r = 1/1000$. Note that only $n_{\text{downstream}} \, r = 10$ rare cells would be found without sorting.
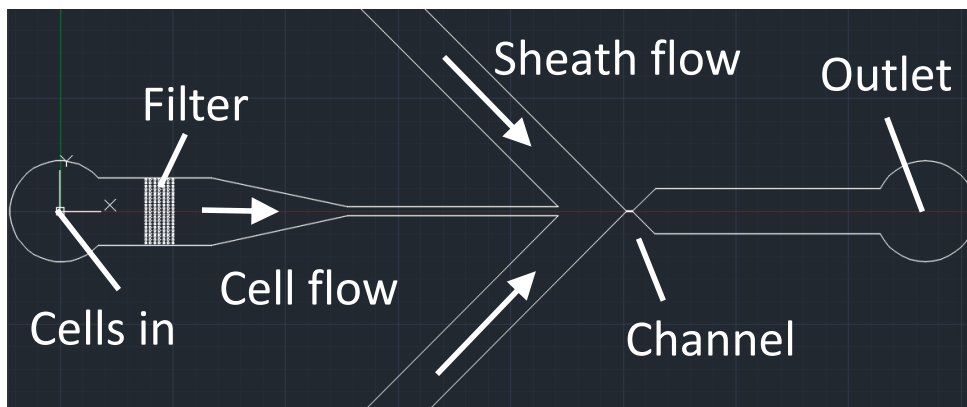
A.2   Full Chip Design

Figure A.3: CAD layout of microfluidic chip. Cells enter through the core and are focused by two sheath flows. Cells are imaged in the channel region and then collected from outlet.

A.3   MASK-RCNN Training

MASK-RCNN network was trained to segment cells from images and determine masks. Network was trained on hand labeled dataset of HL60 and HL60d across multiple independent experiments using VGG Image annotator 1.0. MASK-RCNN is able to accurately fit masks at various focusing and lighting conditions with a mAP and mIOU > 0.9, reducing bias in overall mask fitting between experiments. Matterport implementation which was updated to run on Tensorflow 2.2 was used. The training schedule consisted of training head layers for 20 epochs at a learning rate (LR) of 10^{-3}, 50 epochs training 4+ layers at LR/10, and 50 epochs training all layers at LR/10.
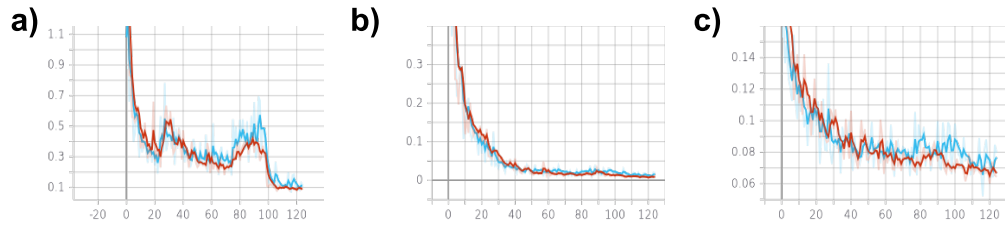
Figure A.4: Training curves for Mask-RCNN. a) Total loss per epoch. b) Bounding box loss per epoch. c) Mask loss per epoch.

A.4   Feature Table

Feature	Description	Hl60n	HL60d
Radius	Radius of cell in cavity where deformation is minimal. Average radius of circle of equivalent perimeter and circle of equivalent area.	59%	56%
rD1	Maximum relative aspect ratio of fitted ellipse measured in first narrow constriction.	57%	72%
rD2	Maximum aspect ratio of fitted ellipse measured in second narrow constriction	55%	71%
R2 Slope	Slope of linear fit between position of maximum deformation in first narrow constriction and position of least deformation in cavity. Represents cell relaxation.	58%	68%
R3 Slope	Slope of linear fit between position of least deformation in cavity and position of maximum deformation in second narrow constriction. Represents cell compression.	52%	70%
R1-R3 AR	Difference between AR1 and AR2	58%	59%
R1 Per	Average perimeter of mask in first narrow constriction.	53%	55%
R3 Per	Average perimeter of mask in second narrow constriction	51%	54%
R1 Area	Average area of mask in first narrow constriction	55%	57%
R3 Area	Average area of mask in second narrow constriction	54%	56%

A.5 Gated Recurrent Unit Network

A GRU network was trained using Tensorflow version 2.3.1. A model with the layers shown below was generated. The optimizer used was RMSprop and the loss function was binary cross-entropy. The hyperparameters were optimized using the test set. Once hyperparameter optimization was finished, test accuracy was recorded and no further training was conducted.

1. Masking

2. GRU - Dimension 50

3. GRU - Dimension 50

4. Dense - Length 24 - Activation: relu

5. Dense - Length 1 - Activation: sigmoid

A.6 CNN-GRU

Data were filtered, aligned and partitioned using the same methods described above for the GRU. Masks were cropped from the frame and padded to 96x96. A ellipse was fit to the cell and the two channels were concatenated. The optimizer used was RMSprop and the loss function was binary cross-entropy. The hyperparameters were optimized over 175 epochs using the validation set. Once hyperparameter optimization was finished, test set accuracy was recorded and no further training was conducted. In order to further validate model performance, 5-fold cross validation was also conducted. Peak performances from each fold are listed below. The following model was used:

1. Conv2D - Filters 8

2. MaxPooling2D

3. Conv2D - Filters 16

4. MaxPooling2D

5. Conv2D - Filters 16

6. MaxPooling2D

7. Flatten

8. GRU - Dimension 50

9. GRU - Dimension 50

10. Dense - Length 1 - Activation: sigmoid

5-Fold Cross-Validation Results HL60 v HL60d

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
0.917	0.922	0.908	0.906	0.917	0.914 ± 0.006

5-Fold Cross-Validation Results HL60 v HL60n

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
0.833	0.821	0.840	0.848	0.842	0.837 ± 0.009

A.7 Detection, Segmentation and Tracking

A 4-layer CNN was created with each Conv2D layer containing a relu activation function, 3x3 kernel and 16 filters. The final layer was a single dense node with the sigmoid activation function. RMSprop was used to optimize the binary cross-entropy loss with a base learning rate of 0.0001. Roughly 9000 total images were used as inputs with 4000 background images

and 5000 cell images. The input images were resized with padding down to 256x256. Pixel values were scaled down from 16 bits to 8 bits using the formula below.

$$\text{new frame} = 255 \times (\text{frame - min(frame)})/(\text{range(frame)})$$

After frames have been labeled by the CNN prefilter they are passed into the trained Mask RCNN where fits are obtained. Fits with a pixel area less than 600 or greater than 2500 are thrown out. The algorithm will then take data from the cell such as x-position, frame time and the binary mask to a dictionary with active events. The new mask data is appended to a list of a given event's mask data based on the new mask's change in frame time and x-position. If there are no cells that match the incoming data, the incoming data is assigned a new key in the dictionary. After the algorithm attempts to match incoming data with existing events, it checks to see if all incoming data has been assigned. If it has not, all incoming masks that have not been assigned are matched to the closest event. At the end of the video, active events are closed off. Events are only saved if there are more than 20 frames in the list.

A.8 Comparison of radii between classes

We measure the radii of cells by measuring the area and perimeter of cell masks where shear stress is zero, which occurs at $75\,\mu m$. The radii is the average of fitting a circle with equivalent area and perimeter respectively. We find a small discrepancy between the mean radii of HL60 cells and HL60 cells treated with cytoD, while HL60 cell treated with noco have a slightly larger variation in size. Cells with radii larger than 3σ from the mean were removed from analysis, as they were often small cell or PDMS debris, or clumps of two or more cells.
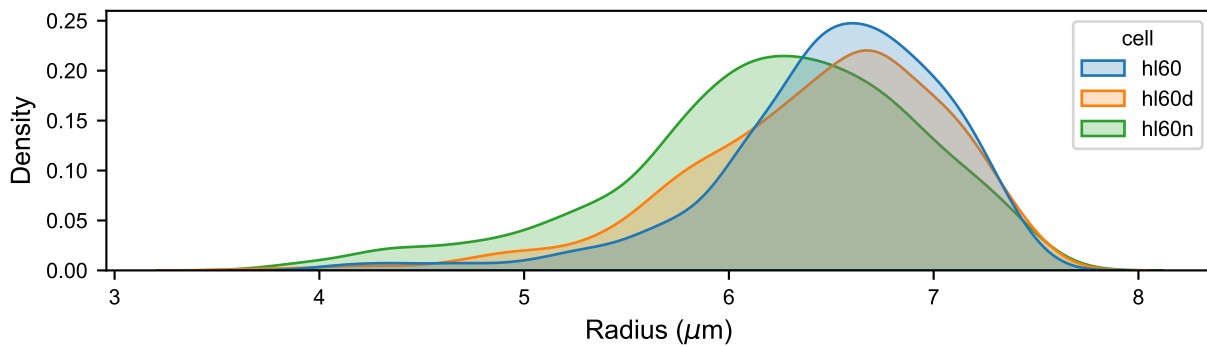

Figure A.5: HL60 radii comparison. Distribution of radii for three cell populations.

A.9 Traditional Model Performance and Training

All models were created and trained using python 3.6 and scikit-learn. The features used are listed in the Feature Table above. The data were shuffled and split according to the following ratio: 70:15:15 for train, validation and test respectively. A function was built for the standardization based on the train data and then used to transform the validation and test data so as to not leak information. Accuracy is reported using the test data.

A.9.1 Random Forest

Using the gini criterion, different depths were tested in a grid search.

A.9.2 Support Vector Machine

The regularization strength, kernel, and degree for the polynomial kernel were optimized in a grid search.
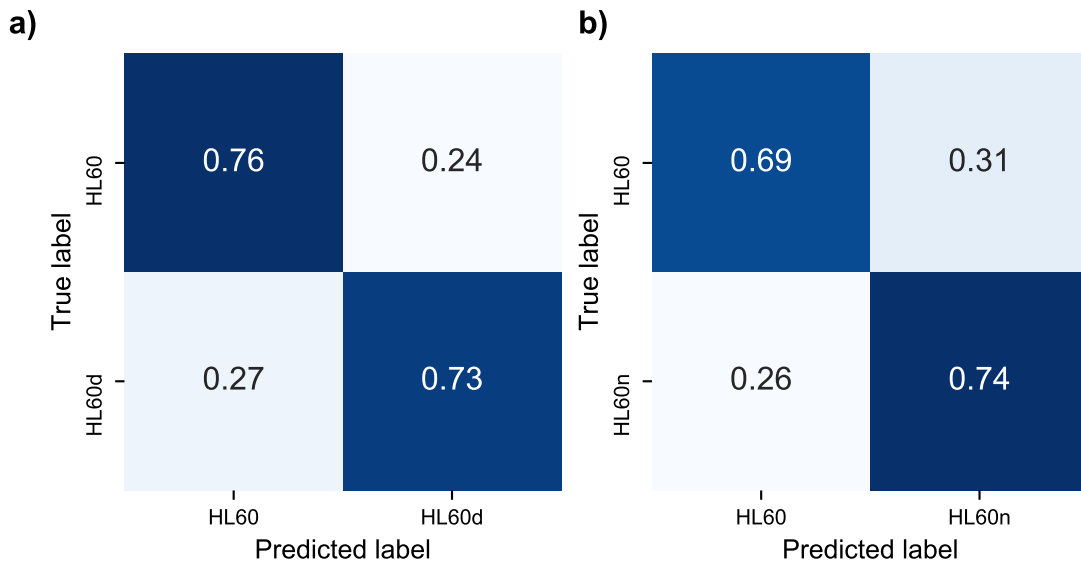
SVM Performance

Figure A.6: Confusion matrix for SVM. Accuracy equals average of diagonal elements.

A.10 Long Short-Term Memory Network and Accuracy

An LSTM network was trained using Tensorflow version 2.3.1. A model with the layers shown below was generated. The optimizer used was RMSprop and the loss function was binary cross-entropy. The hyperparameters were optimized using the validation set. Once hyperparameter optimization was finished, test accuracy was recorded and no further training was conducted.

1. Masking

2. LSTM - Dimension 50

3. LSTM - Dimension 50

4. Dense - Length 24 - Activation: relu

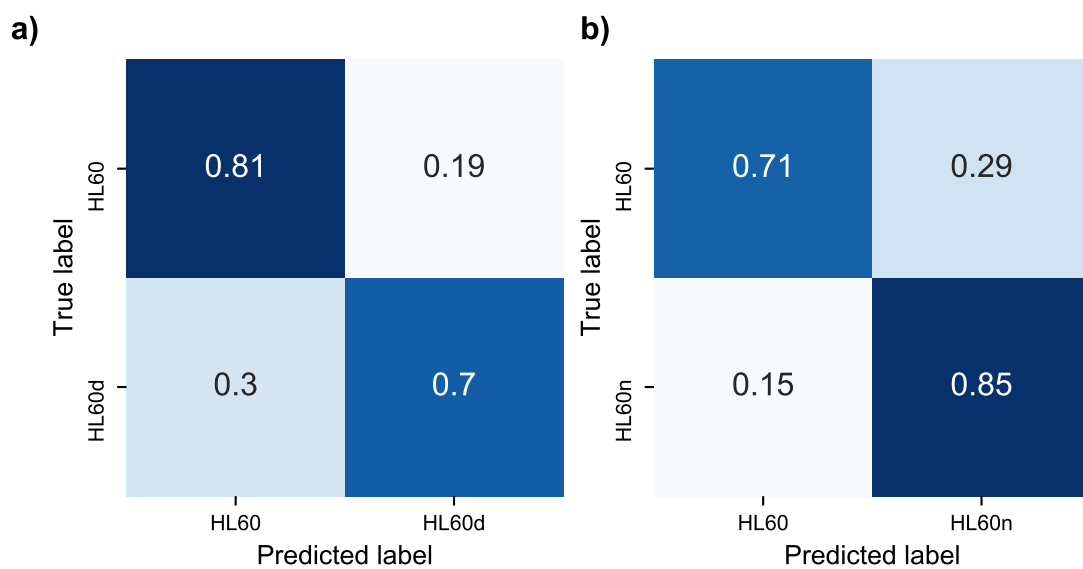5. Dense - Length 1 - Activation: sigmoid

Figure A.7: Confusion matrix for LSTM. Accuracy equals average of diagonal elements. a) Confusion matrix for HL60 vs HL60d. b) Confusion matrix for HL60 vs HL60n

A.11 CNN model trained to detect morphology

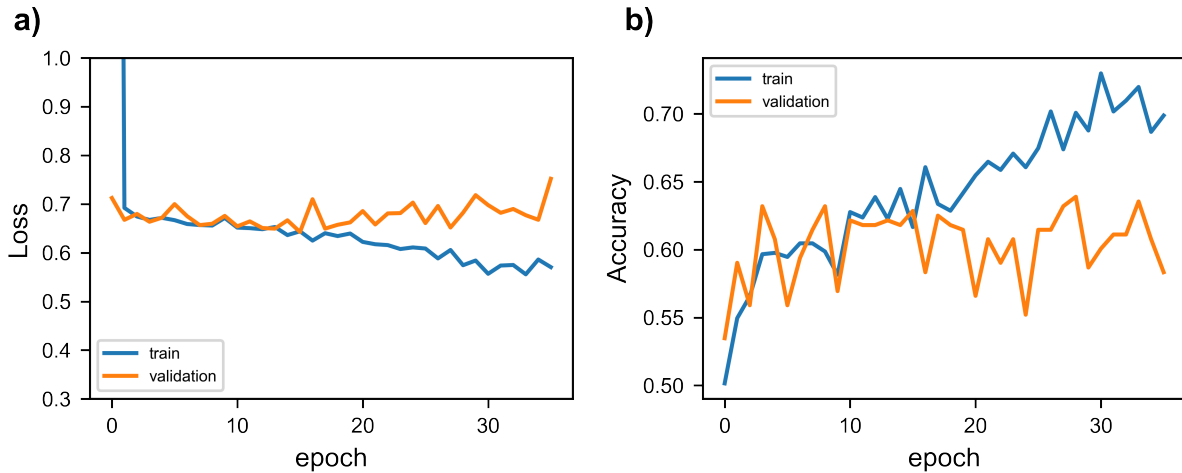**Morphology HL60 v HL60d**

Figure A.8: Training curve and validation set accuracy for CNN to detection morphology differences. a) Total loss b) Validation accuracy during training. Accuracy was measured on a held out test set.
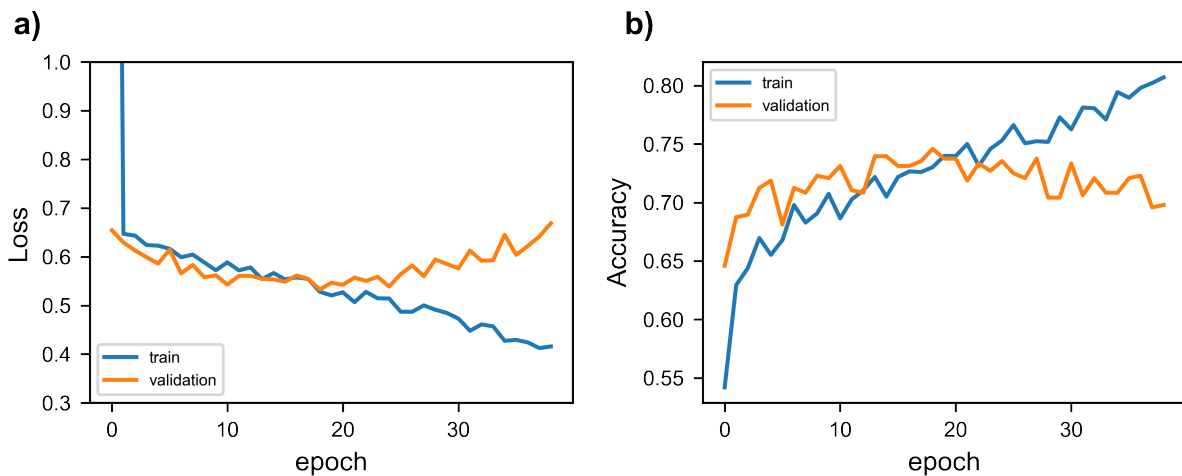
Morphology HL60 v HL60n

Figure A.9: Training curve and validation set accuracy for CNN to detection morphology differences.

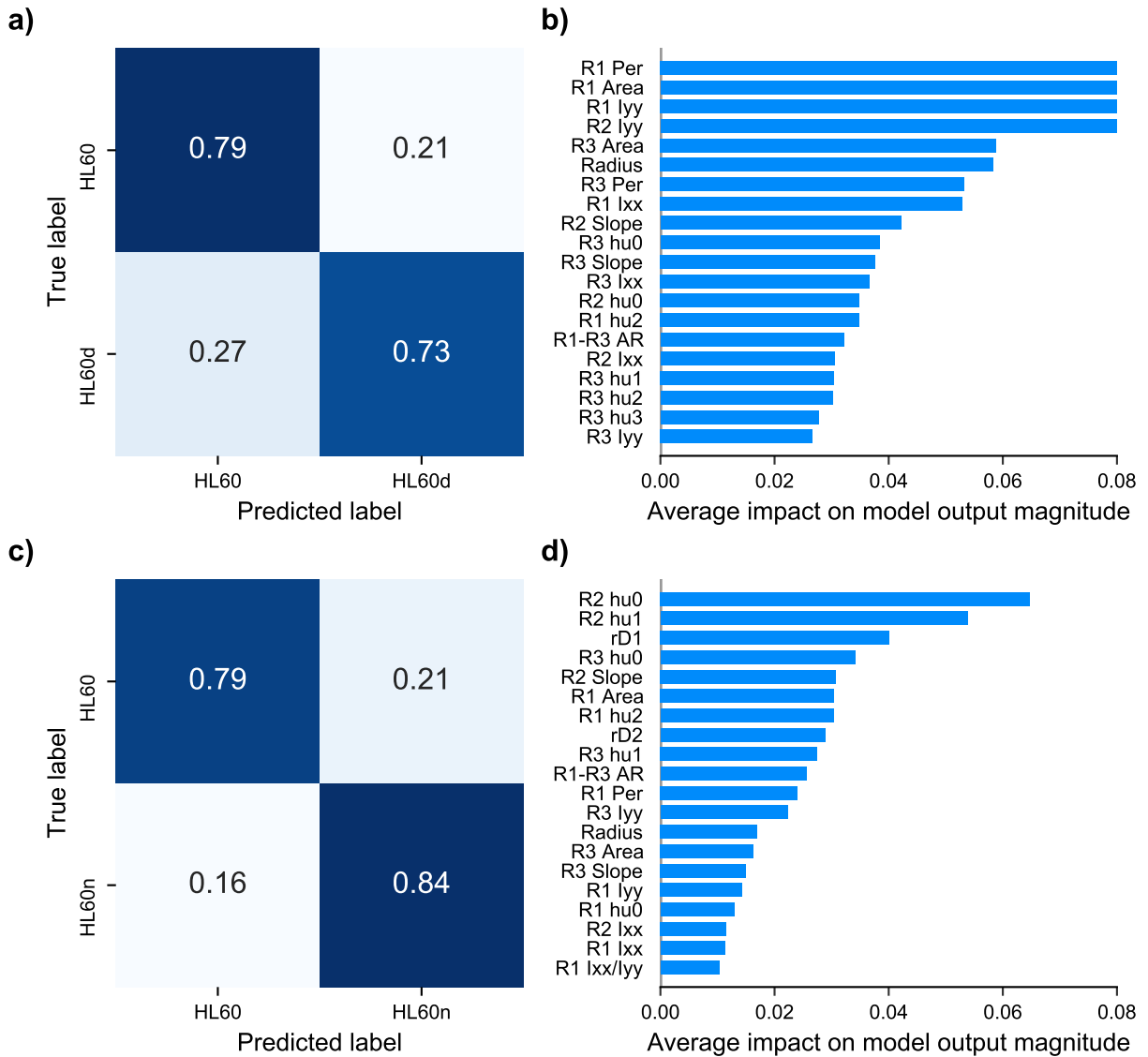# A.12 SVM trained with mechanical and additional morphological features

Figure A.10: Performance for optimized SVM on set of full mechanical and morphologically derived features. a) Confusion matrix for HL60 vs HL60d. b) Corresponding SHAP plot for HL60 vs HL60d. c) Confusion matrix for HL60 vs Hl60n. d) Corresponding SHAP plot for HL60 vs HL60n

Feature	Description (Calculated at R1 max deformation, R2 minimum deformation, and R3 max deformation.)
I_{xx}	Central moment of intertia along channel in direction of fluid flow.
I_{yy}	Central moment of intertia perpendicular to direction of fluid flow.
I_{xx}/I_{yy}	Ratio of diagonal elements of central intertia tensor.
H[0],H[1],... H[6]	Hu's set of image moments