



Transparent Single-Cell Set Classification with Kernel Mean Embeddings

Siyuan Shan
siyuanshan@cs.unc.edu
Department of Computer Science,
UNC-Chapel Hill
USA

Vishal Athreya Baskaran
athreya@cs.unc.edu
Department of Computer Science,
UNC-Chapel Hill
USA

Haidong Yi
haidyi@cs.unc.edu
Department of Computer Science,
UNC-Chapel Hill
USA

Jolene Ranek
ranekj@live.unc.edu
Computational Medicine Program,
Curriculum in Bioinformatics and
Computational Biology, UNC-Chapel
Hill
USA

Natalie Stanley
natalies@cs.unc.edu
Computational Medicine Program,
Department of Computer Science,
UNC-Chapel Hill
USA

Junier B. Oliva
joliva@cs.unc.edu
Department of Computer Science,
UNC-Chapel Hill
USA

ABSTRACT

Modern single-cell flow and mass cytometry technologies measure the expression of several proteins of the individual cells within a blood or tissue sample. Each profiled biological sample is thus represented by a set of hundreds of thousands of multidimensional cell feature vectors, which incurs a high computational cost to predict each biological sample's associated phenotype with machine learning models. Such a large set cardinality also limits the interpretability of machine learning models due to the difficulty in tracking how each individual cell influences the ultimate prediction. We propose using Kernel Mean Embedding to encode the cellular landscape of each profiled biological sample. Although our foremost goal is to make a more transparent model, we find that our method achieves comparable or better accuracies than the state-of-the-art gating-free methods through a simple linear classifier. As a result, our model contains few parameters but still performs similarly to deep learning models with millions of parameters. In contrast with deep learning approaches, the linearity and sub-selection step of our model makes it easy to interpret classification results. Analysis further shows that our method admits rich biological interpretability for linking cellular heterogeneity to clinical phenotype.

CCS CONCEPTS

• **Computing methodologies** → *Machine learning algorithms*; • **Applied computing** → **Computational biology**.

KEYWORDS

Cytometry, Single-Cell Bioinformatics, Clinical Prediction, Flow Cytometry, Mass Cytometry, Kernel Methods

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

BCB '22, August 7–10, 2022, Northbrook, IL, USA

© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-9386-7/22/08...\$15.00

<https://doi.org/10.1145/3535508.3545538>

ACM Reference Format:

Siyuan Shan, Vishal Athreya Baskaran, Haidong Yi, Jolene Ranek, Natalie Stanley, and Junier B. Oliva. 2022. Transparent Single-Cell Set Classification with Kernel Mean Embeddings. In *13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '22)*, August 7–10, 2022, Northbrook, IL, USA. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3535508.3545538>

1 INTRODUCTION

Modern immune profiling techniques such as flow and mass cytometry (CyTOF) enable comprehensive profiling of immunological heterogeneity across a multi-patient cohort [12, 20]. In recent years, such technologies have been applied to numerous clinical applications. In particular, these assays allow for both the phenotypic and functional characterization of immune cells based on the simultaneous measurement of 10–45 protein markers [3]. To further connect the diversity, abundance, and functional state of specific immune cell-types to clinical outcomes or external variables, modern bioinformatics approaches have focused on how to engineer or learn a set of “immune features” that adequately encodes a profiled individual's immunological landscape. In general, existing approaches for creating immunological features either rely heavily on manual human effort [14, 16, 20], clustering [5, 26, 32], or do not produce immunological features that are readily interpretable or informative for follow-up experiments, diagnostics, or treatment strategies [39]. To efficiently and accurately link cellular heterogeneity to clinical outcomes or external variables, here we introduce Cell Kernel Mean Embedding (CKME), a method based on kernel mean embeddings [23] that directly featurizes cells in samples according to their aggregate distribution. We show that CKME is simple enough to be readily interpreted by a human, and learns individual cell scores. Empirical studies show that CKME achieves competitive to state-of-the-art classification accuracy in clinical outcome prediction tasks.

Flow and Mass Cytometry Assays Produce a Dataset of Sets

To comprehensively profile the immune systems of biological samples collected across multiple individuals, a unique data structure of multiple *sample-sets* is ultimately produced. Each sample-set is

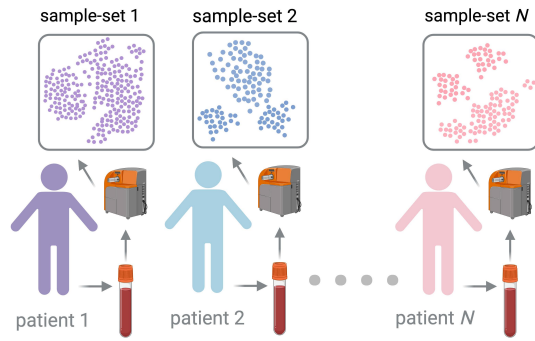


Figure 1: Single-cell profiles produce a *sample-set* of (typically thousands of) multidimensional feature vectors of measured features per cell (illustrated as scatter plots). Thus, a dataset of multiple biological samples will result in a dataset of multiple sets (shown above).

likely to contain hundreds of thousands of cells collected from an individual, and machine learning models are often used to predict their associated clinical or experimental labels. This concept is illustrated in Fig. 1. In profiling multiple individuals with a single-cell technology, a *dataset of sets* is produced and is a non-traditional data structure that breaks the standard convention of a *dataset of feature vectors*. This produces complexity for learning over individuals (e.g. patients), as single-cell data requires a model to characterize and compare across multiple sets of multiple vectors (where each vector represents a cell). Another challenge of analyzing single-cell data is the large number of cells in each sample-set, which incurs a computational cost and obscures the decision-making process of machine learning models.

Related Work The first class of methods for identifying coherent cell-populations and specifying their associated immunological features are *gating-based*¹, and operate by first assigning cells to populations either manually, or in an automated manner by applying an unsupervised clustering approach [1, 20]. After an individual’s cells have been assigned to their respective cell-populations, corresponding immunological features are specified by 1) computing frequencies or the proportion of cells assigned to each population and 2) computing functional readouts as the median expression of a particular functional marker [32]. Though gating-based approaches can be automatic without human intervention, they have several drawbacks: 1) they tend to be sensitive to variation in the parameters of the underlying clustering algorithms [32]. 2) the underlying clustering algorithms typically do not scale well to millions of cells. 3) they typically need to rerun the expensive clustering process whenever a new sample-set is given [18]. Furthermore, partitioning cells into discrete clusters can be inadequate in the context of continuous phenotypes such as drug treatment response and ultimately limits the capacity of having a single-cell resolution of the data.

To address these limitations, the second class of methods are *gating-free* and rely on representing, or making predictions based on individual cells [2, 17, 39]. For example, CytoDx [17] uses a linear model to first compute the response of every single cell individually, then the responses are aggregated by a mean pooling

function. Finally, the aggregated result is used by another linear model to predict clinical outcomes. CellCNN [2] uses a 1-d convolution layer to learn filters that are responsive to marker profiles of cells and aggregates the responses of cells to different filters via a pooling layer for prediction. Since both CytoDx and CellCNN only contain a single linear or convolution layer before the pooling operation, they enable human interpretation. However, this property also limits their expressive power. To enhance model flexibility, CytoSet [39] employs a deep learning model similar to Deep Set [40] to handle set data with a permutation invariant architecture that stacks multiple intermediate permutation *equivariant* neural network layers. As a result, the sample-set featurization that CytoSet achieves, while discriminative and accurate, is opaque and a black box, making it difficult to analyze downstream for biological discoveries. In addition, CytoSet usually contains millions of parameters, further obfuscating the underlying predictive mechanisms. Our method CKME belongs to this *gating-free* class; we show that CKME achieves interpretability without harming model expressiveness.

Summary and Contribution of Results CKME tackles the challenges highlighted above by computing a kernel mean embedding (KME) [23] of sample-sets to represent the characteristics of each sample-set’s composition (and distribution). For classification tasks, we train a linear classifier (e.g. linear SVM) in the KME feature space. Despite its simplicity, CKME achieves comparable to state-of-the-art gating-free performance on three flow and mass cytometry datasets with multiple sample-sets and associated clinical outcomes. We show that CKME may be used to obtain an individual cell score for each cell in a sample-set, where the ultimate prediction is the result of the average cell score. As a result, CKME is highly interpretable since one can quantify the individual contribution of every cell to the final prediction. Predictions are then synthesized further using kernel herding [9], which identifies key cells to retain in a subsample. We leverage the transparency of CKME in a thorough analysis to understand predictions. First, we study the semantics of cell scores from CKME, and show that they follow several desirable, intuitive properties. After, we explore cell scores to obtain biological insights from our models, and show that discoveries from CKME are consistent with existing knowledge. We release publicly available code at <https://github.com/shansiliu95/CKME>.

2 METHODS

Notation for Sample-sets A *sample* refers to the collection of cells from an individual’s blood or tissue, and is further represented as a *sample-set* of many, n , cells: $\mathcal{X} = \{x^{(i)}\}_{i=1}^n$, where $x^{(i)} \in \mathbb{R}^d$ denotes the vector of d features (e.g. proteins or genes) measured in cell i . A multi-sample dataset $\mathcal{D}$ contains multiple sample-sets (across multiple, N , individuals and conditions): $\mathcal{D} = \{\mathcal{X}^{(k)}\}_{k=1}^N = \{\{x^{(k,i)}\}_{i=1}^{n_k}\}_{k=1}^N$, where $\mathcal{X}^{(k)} = \{x^{(k,i)}\}_{i=1}^{n_k}$ is the sample-set for the k -th profiled biological sample. Sample-sets often have an associated labels of interest, y , such as their clinical phenotype or conditions. In this case, our dataset consists of sample-set and label tuples: $\mathcal{D} = \{(\mathcal{X}^{(k)}, y^{(k)})\}_{k=1}^N$.

¹Gating refers to partitioning cells into populations

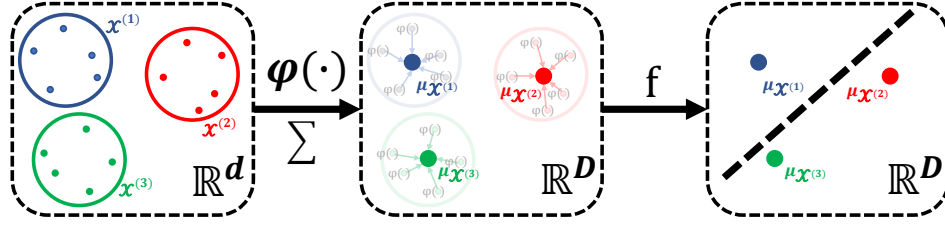


Figure 2: The pipeline of CKME to process the dataset and train the classifier. The left rectangle shows a dataset with 3 sample-sets and each sample-set contains 5 features vectors in the original feature space $\mathbb{R}^d$. In the middle rectangle, we use $\varphi(\cdot)$ to transform the data into random Fourier feature space $\mathbb{R}^D$ and compute the mean embedding $\mu_{\mathcal{X}^{(1)}}$, $\mu_{\mathcal{X}^{(2)}}$, $\mu_{\mathcal{X}^{(3)}}$ for every sample-set (eq. 4). Finally, in the right rectangle, we train a linear discriminative model f (eq. 6) upon the mean embeddings to predict the label of each sample-set.

Problem Formulation Given a dataset of multiple sample-sets as discussed above, we wish to build a discriminative model to predict patient-level phenotypes or conditions based on the cellular composition that is found within a sample-set, $\mathcal{X} \rightarrow y$. If the model is human interpretable, it will help researchers make sense of how the cell feature vectors in a sample-set influence the predicted phenotype or conditions of a patient, which eventually will lead to an improved understanding of biological phenomena and enable better diagnoses and treatment for patients. Below we propose a methodology to featurize and classify input sample-sets in a way that is more transparent and human-understandable than comparably performant models (e.g. [39]).

Kernel Mean Embedding To classify labels of interest, y , given an input sample-set, $\mathcal{X}$, we featurize $\mathcal{X}$ so that we may learn an estimator over those features. However, unlike traditional data-analysis, which featurizes a single vector instance $x \in \mathbb{R}^d$ with features $\phi(x) : \mathbb{R}^d \mapsto \mathbb{R}^q$, here we featurize a *set of multiple* vectors (one vector for each cell in a sample-set) $\mathcal{X} = \{x^{(i)}\}_{i=1}^n$, $\phi(\mathcal{X}) \in \mathbb{R}^q$. Featurizing a set presents a myriad of challenges since typical machine learning approaches are constructed for *statically-sized, ordered* inputs. In contrast, sets are of *varying cardinalities* and are *unordered*. Hence, straight-forward approaches, such as concatenating the sample-set elements into a single vector $(x^{(1)}, \dots, x^{(n)}) \in \mathbb{R}^{nd}$ shall fail to provide mappings that do not depend on the order that elements appear in. To respect the *unordered* property of sample-sets one must carefully featurize $\mathcal{X}$ in a way that is *permutation-invariant*. That is, the features $\phi(\mathcal{X})$ should be unchanged regardless of what order that the elements of $\mathcal{X}$ are processed. Recently, there have been multiple efforts to create methods based on neural networks to featurize sets in a permutation invariant manner [19, 25, 30, 31, 40]. Although these approaches provide expressive, non-linear, discriminative features, they are often opaque and difficult to understand in how they lead to their ultimate predictions. In contrast, we propose an approach based on kernels and random features that is more transparent and understandable whilst being comparably accurate.

Kernel methods have achieved great success in many distinct machine learning tasks, including: classification [10], regression [37], and dimensionality reduction [21]. They utilize a positive definite

kernel function $k : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$, which induces a reproducing kernel Hilbert space (RKHS) (e.g. see [4] for further details). Kernels have also been deployed for representing a distribution, p , with the *kernel mean embedding* $\mu_p : \mathbb{R}^d \mapsto \mathbb{R}$:

$$\mu_p(\cdot) \equiv \mathbb{E}_{x \sim p}[k(x, \cdot)]. \quad (1)$$

Note that μ_p is itself a function (see e.g. [23] for more details). For “characteristic” kernels, k , such as the common radial-basis function (RBF) kernel $k(x, x') = \exp(-\frac{1}{2\gamma}\|x - x'\|^2)$, the kernel mean embedding will be unique to its distribution; i.e., for characteristic kernels, $\|\mu_p - \mu_q\| = 0$ if and only if $p = q$. In general, the distance³ $\|\mu_p - \mu_q\|$ induces a divergence, the maximum mean discrepancy (MMD) [15], between distributions.

For our purposes, we propose to use kernel mean embeddings to featurize sample-sets:

$$\mu_{\mathcal{X}}(\cdot) \equiv \frac{1}{n} \sum_{i=1}^n k(x^{(i)}, \cdot) \approx \mu_p(\cdot), \quad (2)$$

where p is the underlying distribution (of cell features) that $\mathcal{X}$ was sampled from. That is, the set embedding $\mu_{\mathcal{X}}$ (eq. 2) also approximately embeds the underlying distribution of cells that the sample-set was derived from. To produce a real-valued output from the mean embedding, one would take the (RKHS) inner product with a learned function f :

$$\langle \mu_{\mathcal{X}}, f \rangle = \frac{1}{n} \sum_{i=1}^n \langle k(x^{(i)}, \cdot), f(\cdot) \rangle = \frac{1}{n} \sum_{i=1}^n f(x^{(i)}), \quad (3)$$

where the last term follows from the reproducing property of the RKHS. For example, eq. 3 can be used to output the log-odds for a target y given $\mathcal{X}$: $p(y = 1|\mathcal{X}) = (1 + \exp(-\langle \mu_{\mathcal{X}}, f \rangle))^{-1}$. Using the representer theorem [29] it can be shown that f may be learned and represented using a “Gram” matrix of pairwise kernel evaluations, $k(x, x')$. This, however, will be prohibitive in larger datasets. Instead of working directly with a kernel k , we propose to leverage random Fourier features for computational efficiency and simplicity.

Random Fourier Features We propose to use random Fourier frequency features [27] to build our mean embedding [23]. For a shift-invariant kernel (such as the RBF kernel), random Fourier

²Note that kernels may be defined over non-real domains, this is omitted for simplicity.

³In the RKHS norm.

features provide a feature map $\varphi(x) \in \mathbb{R}^D$ such that the dot product in feature space approximates the kernel evaluation, $\varphi(x)^T \varphi(x') \approx k(x, x')$. I.e. $\varphi(x)$ acts as an approximate *primal space* for the kernel k (please see [24, 27, 33, 34] for further details). Using the dot product of $\varphi(x)$, our mean embedding becomes:

$$\mu_X = \frac{1}{n} \sum_{i=1}^n \varphi(x^{(i)}) \in \mathbb{R}^D, \quad \mu_{X'} = \frac{1}{n} \sum_{i=1}^n \varphi(x^{(i)})^T \varphi(x') \quad (4)$$

where $\varphi(x) = (\sin(\omega_1^T x), \dots, \sin(\omega_{D/2}^T x), \cos(\omega_1^T x), \dots, \cos(\omega_{D/2}^T x))$ with random frequencies $\omega_j \sim \rho$ drawn once (and subsequently held fixed) from a distribution ρ that depends on the kernel k . For instance, for the RBF kernel, ρ is an *iid* multivariate independent normal with mean 0 and variance that depends (inversely) on the bandwidth of the kernel, γ .

When computing the mean embedding in the $\varphi(\cdot)$ feature space (eq. 4), one may directly map μ_X to a real value with a dot product with learned coefficient $\beta \in \mathbb{R}^D$:

$$\mu_X^T \beta = \left(\frac{1}{n} \sum_{i=1}^n \varphi(x^{(i)}) \right)^T \beta = \frac{1}{n} \sum_{i=1}^n \varphi(x^{(i)})^T \beta. \quad (5)$$

That is, we may learn a linear model directly operating over the D dimensional feature vectors μ_X , which are composed of the average random features found in a sample-set. Below we expound on how to build a discriminative model based on μ_X .

Linear Classifier with Interpretable Scores With the mean embedding of sample-sets $\{\mu_{X^{(k)}}\}_{k=1}^N$, $\mu_{X^{(k)}} \in \mathbb{R}^D$, we can build a discriminative model $f: \mathbb{R}^D \rightarrow \mathbb{R}$ to predict their labels. If we choose f as a linear model (e.g. linear SVM), then $f(\mu_X)$ can be generally expressed as

$$f(\mu_{X^{(k)}}) = \mu_{X^{(k)}}^T \beta + b = \frac{1}{n_k} \sum_{i=1}^{n_k} \underbrace{\varphi(x^{(k,i)})^T \beta}_{s^{(k,i)}} + b = \frac{1}{n_k} \sum_{i=1}^{n_k} s^{(k,i)}, \quad (6)$$

where $\beta \in \mathbb{R}^D$ and $b \in \mathbb{R}$ are the weight and the bias of the linear model (see Fig.2 for illustration). From eq. 6, we can express the output response of f as the mean of all $s^{(k,i)}$, which we denote as *the score* of the i -th cell in the k -th sample-set $X^{(k)}$. This formulation naturally allows to quantify and interpret the contribution of every cell to the final prediction, which can potentially lead to an improved understanding of biological phenomena and enable better diagnoses and treatment for patients.

Kernel Herding When using random features (eq. 4), μ_X may be understood as the average of random features for cells found in a sample-set. Predicted outputs based on μ_X (eq. 6) may be further interpreted as the average “score” of cells, $s^{(i)}$, in the respective sample-set. However, sample-sets may contain many (hundreds of) thousands of cells, making it cumbersome to analyze and synthesize the cell scores in a sample-set. To ease interpretability, we propose to sub-select cells in the sample-set in a way that yields a similar embedding to the original sample-set. That is, we wish to find a subset $\hat{X} \subset X$ such that $\mu_{\hat{X}} \approx \mu_X$, which implies that one may make similar inferences using a smaller (easier to interpret) subset of cells as with the original sample-set. Although a uniformly

Algorithm 1 COMPUTE THE MEAN EMBEDDING AFTER SUB-SELECTION VIA KERNEL HERDING

Require: A sample-set X , number of cells kept after sub-selection m , dimensionality of the random feature space D , kernel hyperparameter γ .

- 1: # Compute Random Fourier Frequency Features
 - 2: Compute $\mathbf{W} \in \mathbb{R}^{D \times \frac{D}{2}}$ by sampling its elements independently $\mathbf{w}_{i,j} \sim \mathcal{N}(0, \frac{1}{\gamma})$
 - 3: **for each** $x^{(i)} \in X$ **do**
 - 4: $\varphi(x^{(i)}) \leftarrow [\sin(\mathbf{W}^T x^{(i)}), \cos(\mathbf{W}^T x^{(i)})] \in \mathbb{R}^D$, where $[\cdot, \cdot]$ denotes concatenation.
 - 5: **end for**
 - 6: # Sub-selection with Kernel Herding
 - 7: Initialize $j \leftarrow 1$, $\hat{X} \leftarrow \emptyset$, $\theta_0 \leftarrow \frac{1}{n} \sum_{i=1}^n \varphi(x^{(i)})$, $\theta \leftarrow \theta_0$
 - 8: **while** $j \leq m$ **do**
 - 9: $i^* \leftarrow \arg \max_i \theta^T \varphi(x^{(i)})$
 - 10: $\hat{X} \leftarrow \hat{X} \cup \{\varphi(x^{(i^*)})\}$,
 - 11: $\theta \leftarrow \theta + \theta_0 - \varphi(x^{(i^*)})$
 - 12: $j \leftarrow j + 1$
 - 13: **end while**
 - 14: # Compute the mean embedding of the Sub-selected Sample-set
 - 15: $\mu_{\hat{X}} = \frac{1}{m} \sum_{z \in \hat{X}} z$
 - 16: **return** $\mu_{\hat{X}}$
-

random subsample of X would provide a decent approximation $\mu_{\hat{X}}$ for large enough cardinality ($|\hat{X}|$), it is actually a suboptimal way of constructing an approximating subset [9]. Instead, we propose to better construct synthesized subsets (especially for small cardinalities) using kernel herding (KH) [9]. KH can provide a subset of m points that approximates the mean embedding as well as m^2 uniformly sub-sampled points. We expound on KH for producing subsets of key predictive cells in Algorithm 1. We denote the dataset with the sub-selected sets as $\hat{\mathcal{D}} = \{(\hat{X}^{(k)}, y^{(k)})\}_{k=1}^N$.

3 RESULTS

3.1 Datasets

We used publicly available, multi-sample-set flow and mass cytometry (CyTOF) datasets. Each sample-set consists of several protein markers measured across individual cells.

Preeclampsia. The preeclampsia CyTOF dataset profiles 11 women with preeclampsia and 12 healthy women throughout their pregnancies. Sample-sets corresponding to profiled samples from women are publicly available⁴. Our experiments focused on uncovering differences between healthy and preeclamptic women.

HVTN. The HVTN (HIV Vaccine Trials Network) is a Flow Cytometry dataset that profiled T-cells across 96 samples that were each subjected to stimulation with either Gag or Env proteins⁵

⁴<http://flowrepository.org/id/FR-FCM-ZYRQ>

⁵Note these are proteins meant to illicit functional responses in immune cell-types.

Table 1: Classification accuracies on the three datasets. Standard deviations are computed from 5 independent runs.

	HVTN	Preeclampsia	COVID-19
CytoDx [17]	65.24 $\pm$ 1.90	56.17 $\pm$ 3.95	68.82 $\pm$ 3.15
CellCNN [2]	81.87 $\pm$ 1.77	58.51 $\pm$ 3.25	75.93 $\pm$ 2.62
CytoSet [39]	90.46 $\pm$ 2.20	62.85 $\pm$ 3.42	86.55 $\pm$ 1.76
CKME	90.68 $\pm$ 1.69	63.52 $\pm$ 2.22	86.38 $\pm$ 1.92

Table 2: Number of parameters ($\times 10^3$) for different methods on different datasets.

	HVTN	Preeclampsia	COVID-19
CytoDx [17]	0.01	0.03	0.03
CellCNN [2]	40.7	20.2	40.5
CytoSet [39]	330.2	80.0	330.0
CKME	2.0	2.0	2.0

[1]. The data are publicly available⁶. Our experiments focused on uncovering differences between Gag and Env stimulated samples.

COVID-19. The COVID-19 dataset analyzes cytokine production by PBMCs derived from COVID-19 patients. The dataset profiles healthy patients, as well patients with moderate and severe covid cases. Specifically, the dataset consists of samples from 49 total individuals and is comprised of 6 healthy, 23 labeled intensive-care-unit (ICU) with moderately severe COVID, and 20 Ward (non-ICU, but covid-severe) labeled individuals with severe COVID cases, respectively. The data are publicly available⁷. Our experiments focused on uncovering differences between sample-sets from ICU and Ward patients.

3.2 Baselines

We focus our comparison to current state-of-the-art gating-free methods CytoDx [17], CellCNN [2], CytoSet [39], which look to assuage the drawbacks of gating-based methods. Please refer to Sec. 1 “Related Work” for a detailed discussion about these methods.

3.3 Implementation Details

Given the limited number of sample-sets, we used 5-fold cross-validation to report the classification accuracies on the three datasets. We tuned hyperparameters (e.g. the bandwidth parameter γ) on the validation set. On the HVTN and the Preeclampsia datasets, γ was set to 1 while on the COVID-19 dataset γ was set to 8. The dimensionality D was set to 2000 for all datasets. The linear classifier in eq. 6 is implemented by a linear SVM. For all the methods, $m = 200$ cells are sub-selected for all datasets. The best hyperparameters of CytoSet, CellCNN and CytoDx are selected based on the validation performance. CytoSet, CellCNN, CytoDx was run using their public implementation⁸.

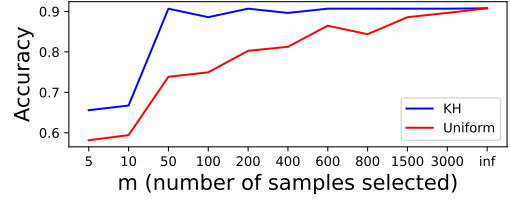
3.4 Classification Accuracies

We report accuracies in Table 1 and numbers of model parameters in Table 2. Although our foremost goal is to make a more

⁶<http://flowrepository.org/id/FR-FCM-ZZZV>

⁷<http://flowrepository.org/id/FR-FCM-Z2KP>

⁸CytoSet: <https://github.com/CompCy-lab/cytoSet>, CellCNN: <http://eiriniar.github.io/CellCnn/code.html>, CytoDx: <http://bioconductor.org/packages/CytoDx>

**Figure 3: The influence of the number of cells selected by KH and uniform sub-sampling, m , on the classification accuracy. $m = \text{“inf”}$ means no sub-selection.****Table 3: Ablation Studies.**

	HVTN	Preeclampsia	COVID-19
CKME w/ Unif	80.26 $\pm$ 1.53	55.52 $\pm$ 4.33	84.28 $\pm$ 1.74
Naive Mean	64.24 $\pm$ 3.17	57.60 $\pm$ 4.20	83.07 $\pm$ 2.29
CKME	90.68 $\pm$ 1.69	63.52 $\pm$ 2.22	86.38 $\pm$ 1.92

transparent model, we find that CKME achieves comparable or better accuracies than the state-of-the-art gating-free methods. The strength of CKME’s performance is impressive when taking into account that: 1) CKME contains on average two orders of magnitude fewer parameters than CytoSet; 2) unlike CytoSet, which has an opaque, uninterpretable set-featurization and prediction model, CKME uses a simple average score over cells; 3) comparably simple models, such as CytoDX and CellCNN, have significantly poorer accuracies than CKME. We believe that CKME’s success is due to the expressive ability of random Fourier features, which provide flexible non-linear mappings (to cell scores) over input cell features without the need to learn additional featurization (e.g. from a neural network). Moreover, the average of the random Fourier features, the mean map embedding (Sec. 2 “Random Fourier Features”), is descriptive of the overall distribution and composition of respective sample-sets; since the discriminator is trained directly on the mean map embeddings, CKME is able to classify based on the cell composition of sample-sets. Lastly, when a sample-set is sub-selected with kernel herding (Sec. 2 “Kernel Herding”), the selected cells are explicitly a salient, descriptive subset that distills the sample-set whilst preserving overall characteristics.

3.5 Ablation Studies

Ablation Studies on Sub-sampling We first ablate the number of sub-samples to study the impact of cell sub-selection on accuracy. Fig. 3 shows the classification accuracies of CKME on the HVTN dataset with different numbers of cells selected by Kernel Herding (KH) and uniform sub-selection. We found that KH consistently outperforms uniform sub-selection and the accuracy of KH quickly saturates with as few as 50 cells sub-selected for every sample-set, confirming the strong capacity of Kernel Herding to maintain the distribution of the original sample-sets after sub-selection. Moreover, in Table 3, we show the performance of CKME with 200 cells sub-selected by uniform sub-sampling (CKME w/ Unif). Compared to sub-selection with Kernel Herding, we find that using uniform sub-sampling leads to a worse result. This confirms that Kernel Herding is more effective than uniform sub-sampling.

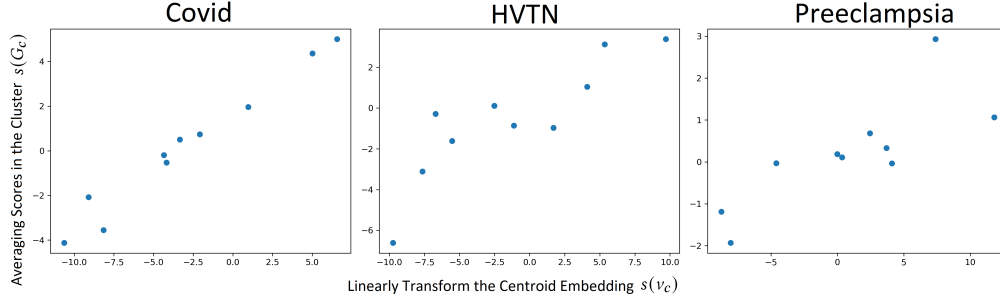


Figure 4: Cluster scores computed by the two methods are positively correlated.

Ablation Studies on Naive Mean Embedding Here, we investigate the performance of mean embeddings without using random Fourier features. In this case, we still used Kernel Herding to sub-select m cells but represented the summarized sample-set in the original feature space as $\bar{x}^{(k)} = \frac{1}{m} \sum_{j=1}^m \hat{x}^{(k,j)} \in \mathbb{R}^d$. We call this approach “Naive Mean”. As shown in Table 3, “Naive Mean” is worse than CKME, which indicates that simple summary statistics ($\bar{x}^{(k)}$, the mean of input features) does not suffice for classification; in contrast, the random feature kernel mean embedding is able to provide an expressive enough summary of sample-sets.

4 ANALYSIS

4.1 Semantic Analysis of Cell Scores

Recall that CKME can compute a score, $s^{(k,i)}$ (eq. 6), for every cell feature vector $\hat{x}^{(k,i)}$ in the KH sub-selected sample-set $\hat{X}^{(k)}$. Although this already enables our model to be more transparent, we hope that these scores are semantically meaningful and consistent with intuitive properties that one would want. We draw an analogy to natural language processing word-embeddings [22], where one constructs word-level featurizations (embeddings) that one hopes are semantically meaningful. For example, the sum of a group of consecutive words-embeddings in a sentence should yield a sentence-embedding that maintains the characteristics of that sentence as a whole. Here, we study the semantics of scores of regions (groups) of cells induced by our cell scores.

We may assign scores to regions in two ways: 1) averaging scores of cells in a nearby region; 2) directly compute the score of the centroid of the respective region. That is, for cells in a nearby region, $G = \{x^{(i)}\}_{i=1}^m$, their average score is $s(G) = \frac{1}{m} \sum_{i=1}^m \varphi(x^{(i)})^T \beta + b$, where $\varphi : \mathbb{R}^d \mapsto \mathbb{R}^D$ are the Fourier features, and β, b are the parameters of the learned linear model. In contrast, one may *directly* compute the score of the centroid of this region, $v = \frac{1}{m} \sum_{i=1}^m x^{(i)}$, as $s(v) = \varphi(v)^T \beta + b$ (note the abuse of notation on s). Both $s(G)$ and $s(v)$ score a region. Analogously to word-embeddings we hope that the score of cells (“words”) retains the semantics of the regions (“sentences”). Below we study the semantic retention of cell scores through a comparative analysis between $s(G)$ and $s(v)$, and a predictive analysis.

Regions We construct nearby regions with a k-means cluster analysis of cells in each dataset. I.e., we obtain C cluster centroids, $v_1, \dots, v_C$, and respective partitions $G_c = \{x \in \bigcup_{k=1}^N \hat{X}^{(k)} \mid c =$

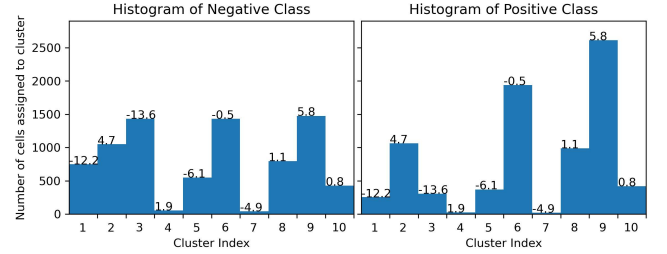


Figure 5: The histogram of the clustering assignments on the HVTN dataset. The scores of all clusters computed by linearly transforming the corresponding centroid embedding, $s(v_c)$, is shown on the top of each corresponding bar.

$\arg\min_l \|v_l - x\|$. Below we set $C = 10$. The distribution of cluster assignments among cells from positive and negative classes on HVTN is shown in Fig. 5. One can see that clusters with a negative score (e.g. Clusters 1 & 3) tend to be more heavily represented in negative sample sets than in positive sample sets (and vice-versa for positive clusters such as Cluster 9).

Comparative Analysis First, we analyze the semantic retention of scores by studying the correlation between the direct scores of cluster centroids $s(v_c) = \varphi(v_c)^T \beta + b$, to the average score of cells in respective clusters $s(G_c) = \frac{1}{|G_c|} \sum_{x \in G_c} \varphi(x)^T \beta + b$. Both $s(v_c)$ and $s(G_c)$ assign scores to the same region, however, due to the nonlinearity of Fourier features there is no guarantee that $s(G)$ will match $s(v)$ as generally $\frac{1}{m} \sum_{i=1}^m \varphi(x^{(i)})^T \beta + b \neq \varphi\left(\frac{1}{m} \sum_{i=1}^m x^{(i)}\right)^T \beta + b$. Notwithstanding, if cell scores are semantically meaningful, then $s(v_c)$ and $s(G_c)$ should relate to each other as they both score regions. When observing the scatter plot of $s(v_c)$ vs. $s(G_c)$ across our datasets (see Fig. 4), we see that both scores on regions are heavily correlated; this shows that our scores are capable of semantically retaining the characteristics of regions. The respective Pearson correlation coefficients for COVID-19, HVTN, and Preeclampsia are 0.9785, 0.8234, and 0.7654.

Predictive Analysis Next, we further study the semantic retention of cell scores with a predictive analysis. Traditionally [5, 26, 32] one may predict a label y using a cluster analysis through a frequency-featurization. E.g. with a linear model $\hat{f}(\hat{X}^{(k)}) = \sum_{c=1}^C \hat{r}_c^{(k)} \alpha_c + a$, where $\hat{r}_c^{(k)}$ is the proportion of cells in $\hat{X}^{(k)}$ assigned to cluster c ,

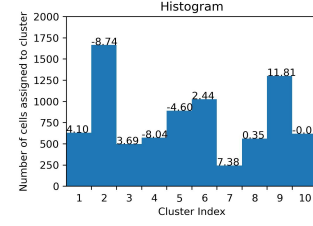
Table 4: Classification accuracies on the three datasets. Standard deviations are computed from 5 independent runs.

	HVTN	Preeclampsia	COVID-19
<i>Linear</i>	78.66 ± 2.86	56.64 ± 4.16	76.92 ± 2.05
<i>Centroid CKME</i>	76.48 ± 2.62	60.57 ± 3.19	77.88 ± 2.11
<i>Average CKME</i>	74.51 ± 2.46	60.80 ± 3.03	76.17 ± 2.13
<i>Average MELD</i>	64.52 ± 0.92	59.64 ± 3.47	59.16 ± 2.72

and α , a are parameters of a learned linear model. In contrast to learning a linear model, one may directly build a predictor using scores for clusters. That is, it is intuitive to combine the scores of clusters using a convex combination according to the frequency of clusters in a sample-set: $\hat{f}_s(\hat{X}^{(k)}) = \sum_{c=1}^C \hat{f}_c^{(k)} s_c$, where s_c is the (pre-trained) score of the c -th region as described above, which is acting as a linear coefficient in the predictor $\hat{f}_s$. I.e. if we have scores for each cluster (coming from a pre-trained CKME model), then weighing these scores according to their respective prevalence in sample-sets should be predictive. Note that although this line of reasoning is *semantically intuitive*, the CKME cell scores were not trained to be used in this fashion, hence there is no guarantee that $\hat{f}_s$ will be predictive. We compare the accuracy of learning a linear model over cluster frequency features (denoted as *Linear*), $\hat{f}_s$, to directly predicting using fixed cluster scores, f_s , in Table 4. We consider cluster scores coming directly from respective centroids $s_c = s(v_c) = \varphi(v_c)^T \beta + b$ as *Centroid CKME*, and cluster scores coming from averages $s_c = s(G_c) = \frac{1}{|G_c|} \sum_{x \in G_c} \varphi(x)^T \beta + b$ as *Average CKME*. As an additional baseline, we also considered building a predictor through cell scores derived from MELD [6]. MELD is an alternative method that computes a score for every cell by first estimating the probability density of each sample along a k NN graph and then computes the relative likelihood of observing a cell in one experimental condition relative to the rest. As MELD cannot directly assign scores to post-hoc centroids, we utilize the average MELD scores for prediction $s_c = \frac{1}{|G_c|} \sum_{x \in G_c} s_x^{\text{MELD}}$, denoted as *Average MELD*. As shown in Table 4, predictors based on both CKME cluster scores (*Centroid CKME* and *Average CKME*) perform comparably to learning a linear model (*Linear*). Here we see that these scores are semantically meaningful since they retain intuitive predictive properties after manipulations. Again, this is surprising given that our CKME cell scores were not trained for this purpose (for prediction with cluster frequencies). This point is further emphasized by the lesser predictive performance of scores from MELD (*Average MELD*), which also seeks to provide semantically meaningful scores and was not trained for prediction in this fashion.

4.2 Biological Validation

An important advantage of interpretable models is their capacity to uncover scientific knowledge [7]. Therefore, we would like to know whether the patterns automatically learned by CKME are consistent with existing knowledge discovered independently via manual analysis. To do so, we focus our analysis on a salient cluster in the Preeclampsia dataset. More specifically, we used our scores to prioritize cells that were different between control (healthy) and preeclamptic women. Cells across sample-sets were clustered into

**Figure 6: We computed the frequency of (e.g. number of) cells assigned to each of 10 clusters, or cell-populations. We further computed the score of corresponding centroids (value shown above each bar).**

one of 10 clusters. We then computed the average score of the cells assigned to each cluster. We first identified cell-populations associated with patient phenotype using CKME scores for each cluster, followed by a finer per-cell analysis highlighting clinically-predictive individual cells and lastly identifying their defining features.

4.2.1 Identifying Phenotype-Associated Cell-Populations. To link particular clusters (e.g. cell-populations) to clinical phenotype, we first prioritized cluster 2, based on its highly negative score according to CKME (-8.74 , see Fig. 6). A highly negative score implies that it likely contained a large number of cells predicted as “control”. The prominent protein markers expressed in cluster 2 were CD3, CD4, CD45RA, and MAPKAPK2 and indicated this is a cell-population of naive CD4⁺ T cells expressing MAPKAPK2 (Fig. 7a). The negative score of cluster 2 aligns with previous work, which showed that women with preeclampsia exhibit a decrease in MAPKAPK2⁺ naive CD4⁺ T-cells during the course of pregnancy, while healthy, women exhibit an increase [16].

We first compared the distributions of frequencies of cells assigned to cluster 2 (i.e. this population of naive CD4⁺ T cells expressing MAPKAPK2) between sample-sets from preeclamptic and healthy control women (Fig. 7c). Consistent with our previous observations, sample-sets from control women had a statistically significantly higher proportion of cells assigned to cluster 2 in comparison to preeclamptic women (see Fig. 7c with a p-value of $p = 0.038$ under a Wilcoxon Rank Sum Test). As a complementary visualization, we constructed a k -nearest neighbor graph between sample-sets according to the computed frequencies across all cell-types (Fig. 7d-e). Here, each node represents a sample-set and an edge represents sufficient similarity between a pair of sample-sets according to the frequencies of cells across cell-populations. In Fig. 7d, sample-sets (nodes) are colored by the probability of their cells belonging to cluster 2. In comparison to sample-sets (nodes) colored by their ground-truth labels (Fig. 7e), we observed that control, healthy sample-sets such as the densely connected set of blue nodes in the bottom of Fig. 7e tend to have high frequencies of cells assigned to cluster 2.

4.2.2 tSNE Visualizations of Per-Cell CKME Scores in Patient Samples in the Preeclampsia CyTOF Dataset. Next, we perform a fine-grained analysis of individual cell scores. In contrast to previous approaches to identifying phenotype-associated cell-populations on a population level [5, 32], CKME provides much finer resolution and instead is able to highlight individual cells (in a digestible, synthesized KH subset) that are likely driving particular clinical

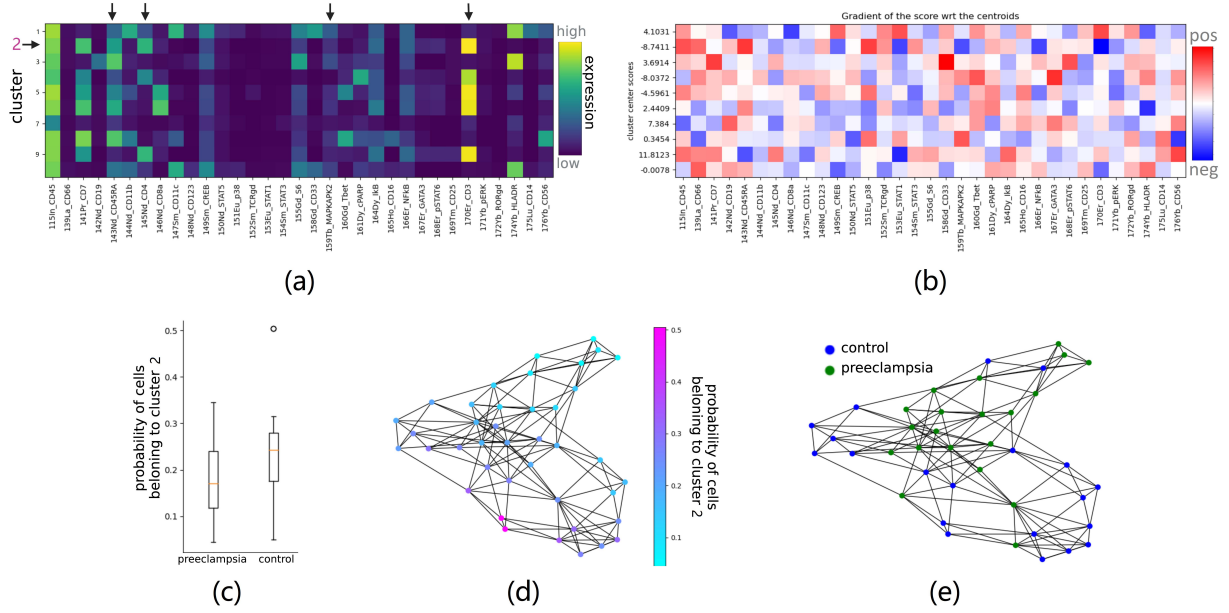


Figure 7: Our predicted scores prioritized cluster 2 ($CD4^+$ naive T-cells) as cell-population likely to have frequency differences between control and preeclamptic samples. (a) Cluster 2 was identified to correspond to a population of $MAPKAPK2^+$ $CD4^+$ naive T-cells according to protein markers (denoted with arrows). (b) Gradients of CKME scores for cluster centroids v_c , $\nabla_x(\varphi(x)^T \beta)|_{x=v_c}$, where $\varphi(x)$ are random Fourier features and β are the learned model weights (eq. 5). (c) The distributions of frequencies of cells assigned to cluster 2 in sample-sets from preeclamptic and control women. (d) A k -NN graph connecting samples-sets (nodes) according to computed frequencies across cell-populations reveals a higher frequency of cells assigned to cluster 2 in control samples-sets. (e) The k -NN graph from (d), with each sample-set (node) colored by its ground-truth label.

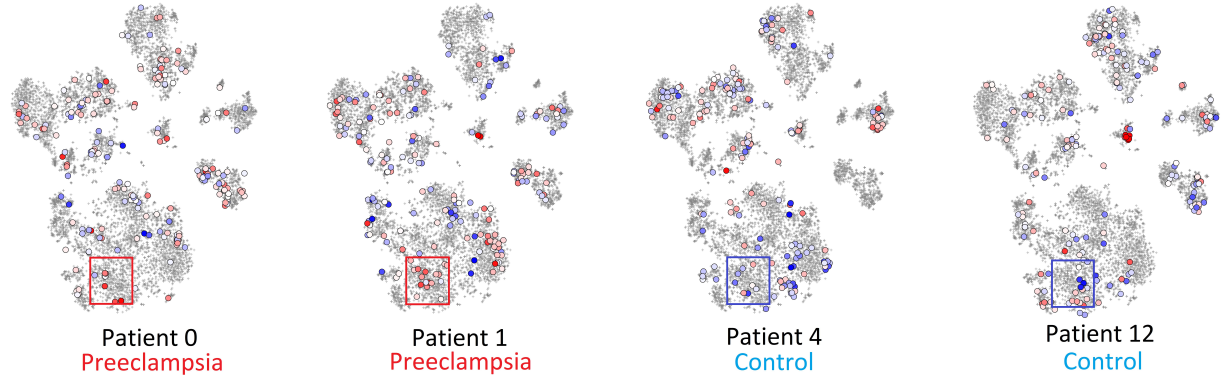


Figure 8: Visualizing computed per-cell CKME scores via tSNE. Cells in gray represent a general, patient-wide cellular landscape. Cells colored red (blue) imply high (low) associations with the preeclampsia phenotype.

phenotypes. For example, by combining cells from samples collected from healthy and preeclamptic women in the preeclampsia CyTOF dataset, we closely examined the patterns in the computed per-cell CKME scores and how they related to patient phenotypes. Briefly, 200 cells were sub-selected with Kernel Herding from each patient and the sub-selected cells from all patients are together projected into two dimensions with tSNE [35] to establish a general, patient-wide cellular landscape (gray cells in Fig. 8). For each patient, their sub-selected cells in the two-dimensional space are colored by their computed CKME score (Fig. 8). In particular, a cell colored

red (blue) implies a high (low) association with the preeclampsia phenotype. In Fig. 8, we visualized scores for patients sampled from two preeclamptic patients (Patients 0 and 1) and two control patients (Patients 12 and 4)⁹. Remarkably, in the two preeclamptic women, we identified prominent subsets of red-colored nodes (outlined in red rectangles), implying a strong association with the preeclampsia phenotype. Based on the expression of phenotypic markers, these cells were identified as memory $CD4^+$ T-cells (based on the expression of CD3, CD4, and CD45RA). In contrast,

⁹These patients were from a testing set and were not used to learn the CKME scores.

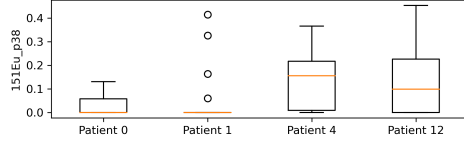


Figure 9: Distributions of features p38 for the cells inside the rectangles of Fig. 8.

the memory CD4⁺ T-cells in the healthy patients exhibited low scores (blue-colored cells outlined in blue rectangles), indicating their negative association with the healthy phenotype. Taken together, the CKME scores highlighted the memory CD4⁺ T-cells as a key predictive population (also shown in previous work [16]). To further unravel the association between specialized immune cells and the preeclamptic and control patient phenotypes according to CKME scores, we investigated the prominent protein feature co-expression patterns of the subset of cells outlined in the rectangles in Fig. 8. Comparing the two preeclamptic (patients 0 and 1) to the two control (patients 4 and 12) sample-sets (Fig. 8), we quickly identified further nuanced cellular subsets associated with the cells with high (red) and low (blue) scoring CKME scores. A particular differentiating protein marker between the preeclamptic and control patients in this memory CD4⁺ T-cell population was p38. The distribution of its expression in cells in the rectangular regions is shown in Fig. 9 and reveals prominent differences between the preeclamptic and control patients. p38 is a functional protein marker, indicating likely differences in signaling responses between control and preeclamptic samples. Interestingly, previous work [16] also independently showed that the expression of p38 in memory CD4⁺ T-cells was an important feature for predicting preeclampsia status.

4.2.3 Assessing Protein Features in Preeclamptic and Healthy Patients. Lastly, we investigated the protein feature coexpression patterns of immune cell types from preeclamptic and healthy patients to further determine whether CKME scores are consistent with existing knowledge and could be used for hypothesis generation. To identify the cell populations associated with extremely positive or negative average CKME scores, we further annotated the clusters according to the expression of known protein markers (Fig. 7a). We found that the cell types that were strongly associated with the preeclampsia phenotype (positive average CKME score) were clusters 1 and 9, corresponding to classical monocytes and memory CD4⁺ T cells (Table 5). In contrast, the cell types associated with the control phenotype (negative average CKME score) were clusters 2, 4, and 5 corresponding to naive CD4⁺ T cells expressing MAPKAPK2, naive CD4⁺ T cells, and naive CD8⁺ T cells (Table 5). Notably, this is consistent with previous work that found that the absolute monocyte count was significantly higher in preeclampsia patients than in controls [36, 38]. Moreover as highlighted previously, an abundance of naive CD4⁺ T cells (Fig. 7c-e) or memory CD4⁺ T cells (Fig. 8) are strong indicators of healthy or preeclampsia status, respectively [8, 11].

We also provide an alternative investigation of important protein markers in CKME scores. In particular, we study how minuscule changes in the expression of the protein markers defining the cluster (cell type) would increase the CKME score and thus have a stronger

Table 5: Gradient analysis of each cell-population highlights prominent features in clinically-associated cell-populations.

ID/CKME score	cell type (expression)	gradient	cell type (gradient)
1/4.1031 9/11.8123	classical monocytes memory CD4 ⁺ T cells	↓ CD14 ↑ CD16 ↑ STAT1 ↑ CD4 ↓ CD45RA ↑ STAT3	nonclassical monocytes STAT1 ⁺ memory CD4 ⁺ STAT3 ⁺ T cells
2/-8.7411 4/-8.0372 5/-4.5961	naive CD4 ⁺ MAPKAPK2 ⁺ T cells naive CD4 ⁺ T cells naive CD8 ⁺ Tbet ⁺ T cells	↑ CD45RA ↓ MAPKAPK2 ↑ p38 ↑ Tbet ↑ GATA3 ↑ Tbet ↑ p38	naive CD4 ⁺ p38 ⁺ T cells CD4 ⁺ Tbet ⁺ GATA3 ⁺ T cells naive CD8 ⁺ Tbet ⁺ p38 ⁺ T cells

association with the clinical phenotype. To do so, we computed the gradient of the CKME score with respect to the cluster centroid. I.e. we compute $\nabla_x(\varphi(x)^T \beta) |_{x=v_c}$ for cluster centroids v_c , where $\varphi(x)$ are random Fourier features and β are the learned model weights (eq. 5). Doing so will uncover what small changes in protein markers alter the CKME of cluster centers, which serves as one proxy for feature importance.

The gradient heatmap (as shown in Fig. 7b) provides a value for each feature within a centroid, where the corresponding direction of change specifies whether an increase (positive) or decrease (negative) in protein expression would increase the association with the preeclampsia phenotype. By prioritizing large magnitude changes in particular features that define cell types, in addition to signaling markers, we observed a shift in the expression of prioritized proteins. When examining the gradient feature changes in cluster 1 (generally defined as classical monocytes by the expression of CD14⁺ CD16⁺), we observed a decrease in CD14 and an increase in CD16 expression (Fig. 7a-b, Table 5). This indicates that a transition to a nonclassical monocyte phenotype (CD14^{med} CD16^{med}) may cause a sample-set to appear more preeclamptic. Interestingly, previous work has indicated that the subpopulation composition of monocytes (classical, intermediate, nonclassical) significantly varies between preeclamptic and control patients [36]. With respect to cluster 2 (naive CD4⁺ T cell population expressing MAPKAPK2), we observed a decrease in MAPKAPK2 and an increase in p38 expression (Fig. 7a-b, Table 5). This further highlights that p38 may be a good functional marker for distinguishing between healthy and preeclamptic patients in both naive and memory CD4⁺ T cells.

5 DISCUSSION AND CONCLUSION

In this work, we introduced CKME (Cell Kernel Mean Embedding) as a method to link cellular heterogeneity in the immune system to clinical or external variables of interest, while simultaneously facilitating biological interpretability. As high-throughput single-cell immune profiling techniques are being readily applied in clinical settings [13, 14, 16], and there are critical needs to 1) accurately diagnose or predict a patient’s future clinical outcome and 2) to explain the particular cell-types driving these predictions. Recent bioinformatic approaches have uncovered [5, 17, 32] or learned [2, 39] immunological features that can accurately predict a patient’s clinical outcome. Unfortunately, these existing methods must often make a compromise between interpretability (e.g. simpler, more transparent methods, such as [2, 17] and accuracy (e.g. more complicated, opaque methods, such as [39]). Our experimental results show that CKME allows for the best of both worlds, yielding state-of-the-art accuracies, with a simple, more transparent model.

We leveraged the transparency of CKME with a thorough analysis. First, we showed that cell scores stemming from CKME retain intuitive desirable properties for cell contributions. We studied the

semantics of CKME cell scores in tasks that CKME *was not explicitly trained for* and found that CKME cell scores were useful for those scenarios. This suggests that, through simple supervised training, CKME can be used to obtain insights in a myriad of analyses that it was not trained for. Furthermore, cell-population, individual-cell, and protein feature coexpression analyses yielded insights that aligned with previous literature (e.g. [16]). This suggests that CKME is a robust approach for uncovering scientific knowledge.

In summary, CKME enables a more comprehensive, automated analysis and interpretation of multi-patient flow and mass cytometry datasets and will accelerate the understanding of how immunological dysregulation affects particular clinical phenotypes. Future directions may include alternate synthesizing approaches to Kernel Herding and variants that weigh cells differently. Furthermore, in addition to the gradient-based analysis presented in Sec. 4.2.3, we shall explore other methodologies for assessing feature importance in CKME scores such as fitting local, sparse models [28].

ACKNOWLEDGMENTS

This research was partly funded by NSF grant IIS2133595 and by NIH 1R01AA02687901A1.

REFERENCES

- [1] Nima Aghaeepour, Greg Finak, Holger Hoos, Tim R Mosmann, Ryan Brinkman, Raphael Gottardo, and Richard H Scheuermann. 2013. Critical assessment of automated flow cytometry data analysis techniques. *Nature methods* 10, 3 (2013), 228–238.
- [2] Eirini Arvaniti and Manfred Claassen. 2017. Sensitive detection of rare disease-associated cell subsets via representation learning. *Nature communications* 8, 1 (2017), 1–10.
- [3] Sean C Bendall et al. 2011. Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science* 332, 6030 (2011), 687–696.
- [4] Alain Berline and Christine Thomas-Agnan. 2011. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media.
- [5] Robert V Bruggner, Bernd Bodenmiller, David L Dill, Robert J Tibshirani, and Garry P Nolan. 2014. Automated identification of stratifying signatures in cellular subpopulations. *Proceedings of the National Academy of Sciences* 111, 26 (2014), E2770–E2777.
- [6] Daniel B Burkhardt, Jay S Stanley, Alexander Tong, Ana Luisa Perdigoto, Scott A Gigante, Kevan C Herold, Guy Wolf, Antonio J Giraldez, David van Dijk, and Smita Krishnaswamy. 2021. Quantifying the effect of experimental perturbations at single-cell resolution. *Nature Biotechnology* (2021), 1–11.
- [7] Diogo V Carvalho, Eduardo M Pereira, and Jaime S Cardoso. 2019. Machine learning interpretability: A survey on methods and metrics. *Electronics* 8, 8 (2019), 832.
- [8] Tinnakorn Chaiworapongsa, Maria-Teresa Gervasi, Jerrie Refuerzo, Jimmy Espinoza, Jun Yoshimatsu, Susan Berman, and Roberto Romero. 2002. Maternal lymphocyte subpopulations (CD45RA+ and CD45RO+) in preeclampsia. *Am. J. Obstet. Gynecol.* 187, 4 (oct 2002), 889–893.
- [9] Yutian Chen, Max Welling, and Alex Smola. 2010. Super-samples from kernel herding. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*. 109–116.
- [10] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.
- [11] D Darmochwal-Kolarz, S Saito, J Rolinski, J Tabarkiewicz, B Kolarz, B Leszczynska-Gorzela, and J Oleszczuk. 2007. Activated T lymphocytes in pre-eclampsia. *Am. J. Reprod. Immunol.* 58, 1 (jul 2007), 39–45.
- [12] Mark M Davis, Cristina M Tato, and David Furman. 2017. Systems immunology: just getting started. *Nature immunology* 18, 7 (2017), 725.
- [13] Ramin Fallahzadeh, Franck Verdonk, et al. 2021. Objective Activity Parameters Track Patient-Specific Physical Recovery Trajectories After Surgery and Link With Individual Preoperative Immune States. *Annals of Surgery* (2021).
- [14] Edward A Ganio, Natalie Stanley, Viktoria Lindberg-Larsen, et al. 2020. Prefential inhibition of adaptive immune system dynamics by glucocorticoids in patients after acute surgical trauma. *Nature communications* 11, 1 (2020), 1–12.
- [15] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. 2008. A Kernel Method for the Two-Sample Problem. *Journal of Machine Learning Research* 1 (2008), 1–10.
- [16] Xiaoyuan Han, Mohammad S Ghaemi, et al. 2019. Differential dynamics of the maternal immune system in healthy pregnancy and preeclampsia. *Frontiers in immunology* 10 (2019), 1305.
- [17] Zicheng Hu, Benjamin S Glicksberg, and Atul J Butte. 2019. Robust prediction of clinical outcomes using cytometry data. *Bioinformatics* 35, 7 (2019), 1197–1203.
- [18] Vladimir Yu Kiselev, Tallulah S Andrews, and Martin Hemberg. 2019. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat. Rev. Genet.* 20, 5 (may 2019), 273–282.
- [19] Yang Li, Haidong Yi, Christopher Bender, Siyuan Shan, and Junier B Oliva. 2020. Exchangeable neural ode for set modeling. *Advances in Neural Information Processing Systems* 33 (2020), 6936–6946.
- [20] Thomas Liechti, Lukas M Weber, Thomas M Ashhurst, Natalie Stanley, Martin Prlic, Sofie Van Gassen, and Florian Mair. 2021. An updated guide for the perplexed: cytometry in the high-dimensional era. *Nature Immunology* 22, 10 (2021), 1190–1197.
- [21] Sebastian Mika, Bernhard Schölkopf, Alexander J Smola, Klaus-Robert Müller, Matthias Scholz, and Gunnar Rätsch. 1998. Kernel PCA and De-noising in feature spaces.. In *NIPS*, Vol. 11. 536–542.
- [22] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* 26 (2013).
- [23] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. 2017. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends in Machine Learning* 10, 1-2 (2017), 1–141.
- [24] Junier Oliva, Willie Neiswanger, Barnabás Póczos, Jeff Schneider, and Eric Xing. 2014. Fast distribution to real regression. In *Artificial Intelligence and Statistics*. PMLR, 706–714.
- [25] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 652–660.
- [26] Peng Qiu, Erin F Simonds, Sean C Bendall, Kenneth D Gibbs, Robert V Brugner, Michael D Linderman, Karen Sachs, Garry P Nolan, and Sylvia K Plevritis. 2011. Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nature biotechnology* 29, 10 (2011), 886–891.
- [27] Ali Rahimi, Benjamin Recht, et al. 2007. Random Features for Large-Scale Kernel Machines.. In *NIPS*, Vol. 3. Citeseer, 5.
- [28] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*. 1135–1144.
- [29] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola. 2001. A generalized representer theorem. In *International conference on computational learning theory*. Springer, 416–426.
- [30] Siyuan Shan, Yang Li, and Junier B Oliva. 2021. NRTSI: Non-Recurrent Time Series Imputation. *arXiv preprint arXiv:2102.03340* (2021).
- [31] Yifeng Shi, Junier Oliva, and Marc Niethammer. 2020. Deep Message Passing on Sets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 5750–5757.
- [32] Natalie Stanley, Ina A Stelzer, et al. 2020. VoPo leverages cellular heterogeneity for predictive modeling of single-cell data. *Nature communications* 11, 1 (2020), 1–9.
- [33] Dougal Sutherland, Junier Oliva, Barnabás Póczos, and Jeff Schneider. 2016. Linear-time learning on distributions with approximate kernel embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.
- [34] Danica J Sutherland and Jeff Schneider. 2015. On the error of random Fourier features. *arXiv preprint arXiv:1506.02785* (2015).
- [35] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
- [36] Polina Vishnyakova, Andrey Elchaninov, Timur Fatkhudinov, and Gennady Sukhikh. 2019. Role of the Monocyte-Macrophage System in Normal Pregnancy and Preeclampsia. *Int. J. Mol. Sci.* 20, 15 (jul 2019).
- [37] Vladimir Vovk. 2013. Kernel ridge regression. In *Empirical inference*. Springer, 105–116.
- [38] Jing Wang et al. 2019. Assessment efficacy of neutrophil-lymphocyte ratio and monocyte-lymphocyte ratio in preeclampsia. *J. Reprod. Immunol.* 132 (apr 2019), 29–34.
- [39] Haidong Yi and Natalie Stanley. 2021. CytoSet: Predicting Clinical Outcomes via Set-Modeling of Cytometry Data. In *Proceedings of the 12th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics* (Gainesville, Florida) (BCB '21). Association for Computing Machinery, Article 47, 8 pages.
- [40] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. 2017. Deep Sets. *Advances in Neural Information Processing Systems* 30 (2017).