DICE: Deep Significance Clustering for Outcome-Driven Stratification

Yufang Huang
Cornell University
yfhuang1992new@gmail.com

Joel C. Park
NewYork-Presbyterian Hospital Weill Cornell Medical Center
jcp9010@med.cornell.edu

Kelly M. Axsom
Columbia University Irving Medical Center
kma2161@cumc.columbia.edu

Lakshminarayanan Subramanian
New York University
lakshmi@nyu.edu

Yiye Zhang
Cornell University
yiz2014@med.cornell.edu

Abstract

We present deep significance clustering (DICE), a framework for jointly performing representation learning and clustering for "outcome-driven" stratification. Motivated by practical needs in medicine to risk-stratify patients into subgroups, DICE brings self-supervision to unsupervised tasks to generate cluster membership that may be used to categorize unseen patients by risk levels. DICE is driven by a combined objective function and constraint which require a statistically significant association between the outcome and cluster membership of learned representations. DICE also performs a neural architecture search to optimize cluster membership and hyper-parameters for model likelihood and classification accuracy. The performance of DICE was evaluated using two datasets with different outcome ratios extracted from real-world electronic health records of patients who were treated for coronavirus disease 2019 and heart failure. Outcomes are defined as in-hospital mortality (15.9%) and discharge home (36.8%), respectively. Results show that DICE has superior performance as measured by the difference in outcome distribution across clusters, Silhouette score, Calinski-Harabasz index, and Davies-Bouldin index for clustering, and Area under the ROC Curve for outcome classification compared to baseline approaches.

1 Introduction

Representation learning [1, 2] and clustering [3] are unsupervised algorithms whose results are driven by input features and priors. They are often exploratory in nature, but in certain use cases users have *a priori* expectations for the outputs from representation learning and clustering. In the latter case, having targeted self-supervision in the learning process so as to meet the expectation of the users brings practical value for representation learning and clustering algorithms. This paper proposes deep significance clustering (DICE), an algorithm for self-supervised, interpretable representation learning

and clustering that targets features that best stratify a population concerning specific outcomes of interest.

We are motivated by a practical need in medicine to identify and understand characteristics of subgroups of patients [4, 5, 6]. Patient health information is extremely complex, capturing high-dimensional, temporal, and heterogeneous data on diseases, biomarkers, medications, procedures, among other health indicators. For instance, heart failure (HF) is a syndrome that impacts nearly 6 million Americans and is associated with a 50% 5-year mortality [7]. More than 80% of individuals suffer from three or more comorbidities [8]. There are proven HF therapies that prolong survival. For patients in a moderately sick cohort, known as Stage C, therapies are oral medications, known as neurohormonal blockade. This cohort of patients have a 5-year survival of 75%. For the sickest cohort of heart failure patients, known as Stage D or end stage HF, therapies are focused on heart replacement like heart transplant or left ventricular assist devices. This population has a 20% 5-year survival [7]. On the contrary, therapeutic guidelines for novel coronavirus disease 2019 (COVID-19) are not as clearly delineated as HF treatment. In both scenarios, the complexity due to frequent comorbidity or the lack of clear guidelines warrant the discovery of patient subtypes to assist with clinical decision making. While clustering is an obvious choice for patient subtyping, the heterogeneous health data present a challenge to such unsupervised algorithm for its incapability to elicit cluster membership that is targeted for an outcome of interest, such as whether a HF patient can be safely discharged home and the mortality risk of COVID-19 patients. This challenge arises because a regular clustering algorithm may find clusters of patients who differ with respect to factors that are not related to meaningful clinical endpoints.

Addressing such clinical needs, DICE, a framework to learn a deep representation and cluster memberships from heterogeneous data was developed in an effort to bridge representation learning, clustering, and targeted outcome separation. Its architecture is illustrated in Fig. 1. Representation learning allows us to discover a concise representation from the heterogeneous and sparse health data, which we use to discover latent clusters within a patient population with clustering algorithms. As a way to provide more interpretability to the representation learning and clustering, DICE uses a combined objective function and a constraint that requires statistically different outcome distribution across clusters. The statistical significance is determined using models that are well-understood by clinicians such as regression while adjusting for patient demographics. The combined objective function and constraint serve to force DICE to learn representations that lead to clusters that are discriminative to the outcome of interest. Furthermore, a neural architecture search (NAS) is designed with an alternative grid search for the number of clusters and hyper-parameters in the representation learning. The finalized representation and cluster memberships, which represent significantly different outcome levels, are then used as the class labels for a multi-class classification. This is intended to allow new patients to be categorized according to risk-level specific subgroups learned from historic data.

Previous studies [9] that incorporated statistical significance analyzed it separately after the representation learning process. Our paper considers the statistical significance while performing deep clustering as a constraint. To summarize, our approach makes the following key contributions:

- We propose a combined objective function to achieve the joint optimization for outcome-driven representation and clustering membership from heterogeneous health data.
- We propose an explicit constraint that forces statistical significance of the association between the cluster membership with the outcome to drive learning.

We evaluated DICE on two real-world datasets collected from electronic health records (EHR) data at an academic medical center. Extensive experiments and analyses demonstrate that the DICE obtains better performance than several baseline approaches in outcome discrimination, Area under ROC Curve (AUC) for prediction, and clustering performance metrics including Silhouette score, Calinski-Harabasz index and Davies-Bouldin index.

2 Related Work

Clustering is a fundamental topic in the exploratory data mining which can be applied to many fields, including bioinformatics [10], marketing [11], computer vision [12] and natural language processing [13]. Due to the inefficiency of similarity measures with high-dimensional big data, traditional

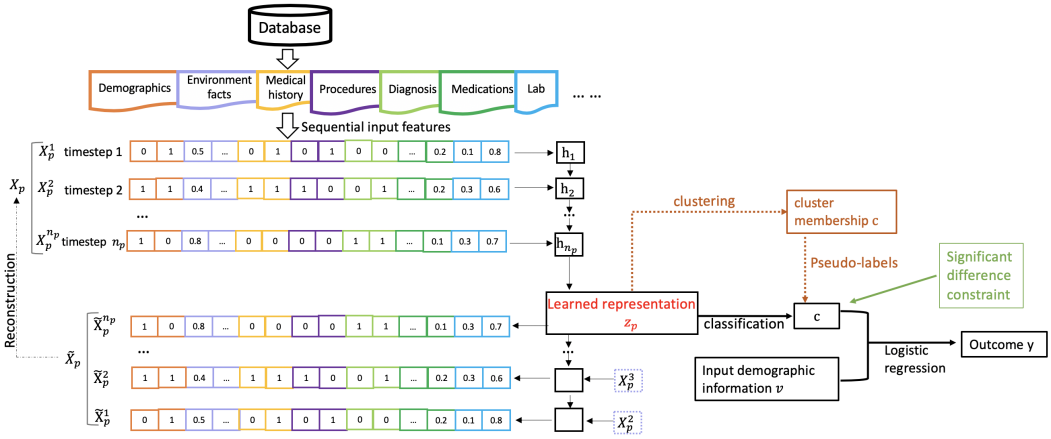


Figure 1: The framework of the proposed deep significance clustering (DICE). Clustering is applied to the representation $\mathbf{z}_p$. A statistical significance constraint is explicitly added to ensure the association of the clustering membership $\mathbf{c}$ and outcome y to facilitate the learning of discriminative representations $\mathbf{z}_p$.

clustering approaches, e.g., k-means [14], finite mixture model [15, 16] and Gaussian Mixture Models (GMM) [17], generally suffer from high computational complexity on large-scale datasets [18]. Also, the mixture models have distribution assumptions on observations [19]. Jagabathula [20] proposed a conditional gradient approach for nonparametric estimation of mixing distributions. Data transform approaches which map the raw data into a new feature space have been studied, including principal component analysis (PCA) [21], kernel methods [22], model-based clustering [23, 19] and spectral methods [24, 25]. However, clustering of high-dimensional heterogeneous data is still challenging for these approaches because of inefficient data representation. Deep representation learning can be used to transform the data into clustering-friendly representation [26, 27, 28]. Parametric t-SNE [29] uses deep neural network to parametrize the embedding of t-SNE [30] with the same time complexity of $O(n^2)$, where n is the number of data points. DEC [27] further relaxes parametric t-SNE with a centroid-based probability distribution which reduces complexity to $O(nk)$ from tree-based t-SNE of $O(n\log(n))$, where k is the number of centroids. Some approaches learn self-supervised representation [31, 32, 33]. Recent deep clustering approaches are learning-based and conduct inference in one shot, consisting of two stages, i.e., deep representation learning followed by various clustering models. Caron et al. [33] jointly learned the parameters of a deep network and the cluster assignments of the resulting representation. DGG [12] further uses gaussian mixture variational autoencoders and graph embedding to improve the clustering and data representation abilities. DICE considers statistical significance and proposes a novel constraint to obtain statistical significant clustering results.

3 Method

Given a dataset $\mathbb{X} = \{\mathbf{X}_1, ..., \mathbf{X}_P\}$ with P subjects, we denote each subject as a sequence of events $\mathbf{X}_p = [\mathbf{x}_p^1, \mathbf{x}_p^2, ..., \mathbf{x}_p^{n_p}]$ of length n_p. A multivariate feature vector $\mathbf{x}_p^t = [x_{p,1}^t, x_{p,2}^t, ..., x_{p,F}^t] \in \mathbb{R}^F$ is the t-th instance of subject p in sequence $\mathbf{X}_p$, where F is the number of features at each timestamp. We have an outcome y_p for each subject p. Our goal is to stratify $\mathbb{X}$ of P subjects into K clusters while enforcing statistical significance in the association of the cluster membership and the outcome while adjusting for relevant covariates.

3.1 Learning representation

The first step is to transform discrete sequences into latent continuous representations, followed by clustering and classification. The latent representation learning for each subject is performed by an LSTM autoencoder (AE) [34]. The AE consists of two parts, the encoder and the decoder, denoted as $\mathcal{E}$ and $\mathcal{F}$, respectively. Given the p-th input sequence $\mathbf{X}_p = (\mathbf{x}_p^1, \mathbf{x}_p^2, ..., \mathbf{x}_p^{n_p})$, the encoder can be formulated as $\mathbf{z}_p = \mathcal{E}(\mathbf{X}_p; \theta_\mathcal{E})$, where $\mathbf{z}_p \in \mathbb{R}^d$ is the representation, and $\mathcal{E}$ is a LSTM network with

parameter $\theta_{\mathcal{E}}$ [35]. We choose the last hidden state $\mathbf{z}_p$ of LSTM to be the representation of the input $\mathbf{X}_p$. The decoder can be formulated as $\tilde{\mathbf{X}}_p = \mathcal{F}(\mathbf{z}_p; \theta_{\mathcal{F}})$, and $\mathcal{F}$ is the other LSTM network with parameter $\theta_{\mathcal{F}}$. The representation learning is achieved by minimizing the reconstruction error

$$\min_{\theta_{\mathcal{E}}, \theta_{\mathcal{F}}} \mathcal{L}_{AE} = \frac{1}{P} \sum_{p=1}^{P} \|\mathcal{F}(\mathcal{E}(\mathbf{X}_p; \theta_{\mathcal{E}}); \theta_{\mathcal{F}}) - \mathbf{X}_p\|_{L_2}^2, \tag{1}$$

where we use L_2 norm in the loss.

3.2 Self-supervised learning by clustering

The obtained representations $\mathbb{Z} = \{\mathbf{z}_p\}_{p=1}^{P}$ can be employed for clustering with K clusters,

$$\begin{aligned} &\min_{\mathbf{M}, \{\mathbf{c}_p\}_{p=1}^{P}} \mathcal{L}_{clustering} = \sum_{p=1}^{P} \|\mathbf{z}_p - \mathbf{M}\mathbf{c}_p\|_2^2 \\ &\text{s.t.} \quad \mathbf{1}^T \mathbf{c}_p = 1,\ c_p^k \in \{0, 1\},\ \forall\, p \in \{1, 2, ..., P\},\ k \in \{1, 2, ..., K\}, \end{aligned} \tag{2}$$

where K is a hyper-parameter to tune, $\mathbf{c}_p = [c_p^1, ..., c_p^K]$, c_p^k is cluster membership of cluster k, $\mathbf{M} \in \mathbb{R}^{d \times K}$ and the k-th columns of $\mathbf{M}$ is the centroid of the k-th cluster.

To enable fast inference, we build a learning-based deep clustering based on self-supervision from $\mathbf{c}_p$ in equation (2). Please note that we can utilize other priors of equation (2) in the DICE. We employ the clustering results $\{\mathbf{c}_p\}_{p=1}^{P}$ from *a priori* in equation (2) as pseudo-labels, and update the parameters of the encoder $\mathcal{E}$ and $\mathcal{F}$. The cluster membership assignment can be formulated as a classification network.

$$\hat{\mathbf{c}}_p = g(\mathbf{z}_p; \theta_1),\ \min_{\theta_1} L_1 = -\sum_{p=1}^{P} \sum_{k=1}^{K} c_p^k \log(\hat{c}_p^k), \tag{3}$$

where $\hat{\mathbf{c}}_p = [\hat{c}_p^1, ..., \hat{c}_p^K]$ is the predicted cluster membership from the classification network $g(\cdot; \theta_1)$, θ_1 is the parameter in the classification network, L_1 is the negative log-likelihood loss for multi-class classification.

The deep clustering bridges the representation learning with the following statistical significance constraint related to the outcome.

For inference, we assign the cluster membership through equation (2) with fixed $\mathbf{M}$ from training.

3.3 Statistical significance constraint

We propose a novel statistical significance constraint to the clustering membership w.r.t. the outcome distribution while adjusting for relevant covariates in the DICE. After obtaining cluster memberships $\{\mathbf{c}_p\}_{p=1}^{P}$ for K clusters, we impose a statistical significance constraint on the cluster membership to drive the representation learning. We fuse the statistical significance constraint into our neural network.

First, in our neural network, we use the cluster memberships and non-clinical events like demographic features to predict outcome, formulated as:

$$\hat{\mathbf{y}}_p = g(\mathbf{c}_p, \mathbf{v}_p; \theta_2),\ \min_{\theta_2} L_2 = -\sum_{p=1}^{P} \big(y_p \log(\hat{y}_p) + (1 - y_p) \log(1 - \hat{y}_p)\big), \tag{4}$$

where $\mathbf{v}_p$ represents non-clinical events, $g(\cdot; \theta_2)$ is the logistic regression, L_2 is the negative log-likelihood loss for binary classification problem.

Second, to quantify the significant difference of cluster k_1 and cluster k_2 ($k_1 \neq k_2$), we use likelihood-ratio test [36] to calculate the p-value of variable c^{k_2} when considering cluster c^{k_1} as the baseline, where c^k refers to the cluster membership belonging to cluster k. We denote the likelihood-ratio as G_{k_1,k_2}, then obtain the p-value from Chi-square distribution, denoted as S_{k_1,k_2}. Finally, we have a matrix $\mathbf{S} \in \mathbb{R}^{K \times K}$ with 0 as diagonal elements, and $S_{k1,k2}(k_1 \neq k_2)$ is the p-value represent the

significance difference of cluster k_2 corresponding to baseline cluster k_1. If all the elements in $\mathbf{S}$ are below a predefined threshold of significance α (equivalently, $G_{k_1,k_2} > \alpha_G$), we conclude that all the clusters are significantly different with each other related to outcome y.

For implementation, we use mask technique to mask each variable of input $\mathbf{c}_p$, then calculate the likelihood ratio G_{k_1,k_2} and add significance constraint on G_{k_1,k_2}, that is $G_{k1,k_2} > \alpha_G, \forall k_1 \neq k_2$.

3.4 Objective function

We use NAS approaches to obtain our final model. The first is weight optimization of a given network architecture, which in our method is the network architecture with fixed clusters number K and hidden state dimension d. The second is the neural architecture search process.

3.4.1 Optimization of a given network architecture

We denote our network architecture as $\mathcal{N}(K, d, \theta)$, where $\theta = \{\theta_{\mathcal{E}}, \theta_{\mathcal{F}}, \mathbf{M}, \theta_1, \theta_2\}$ are the weights of network. The optimization problems is

$$\begin{aligned} &\min_{\theta} \mathcal{L}(\mathcal{N}(K, d, \theta)) \\ =& \min_{\theta} \sum_{i=1}^{P} \|\mathcal{F}(\mathcal{E}(\mathbf{X}_p) - \mathbf{X}_p\|_2^2 + \lambda_1 \|\mathcal{E}(\mathbf{X}_p) - \mathbf{M}\mathbf{c}_p\|^2 + \lambda_2 L_1(g(\mathcal{E}(\mathbf{X}_p); \theta_1), \mathbf{c}_p) \\ &\quad + \lambda_3 L_2(g(\mathbf{c}_p, \mathbf{v}_p; \theta_2), y_p), \\ &\text{s.t.} \quad \mathbf{1}^T \mathbf{c}_p = 1 \quad c_{p,j} \in \{0, 1\}, \quad \forall p \in \{1, 2, ..., P\}, j \in \{1, 2, ..., K\}, \\ &\quad\quad S_{i,j} < \alpha \quad \forall i \in 1, 2, \cdots, K, j \in 1, 2, \cdots, K \end{aligned} \tag{5}$$

To implement, we iteratively optimize the clustering and the other components with the statistical significance constraint. That is,

$$\begin{aligned} &\min_{\theta_{\mathcal{E}}, \mathbf{M}, \mathbf{c}_p} \sum_{i=1}^{P} \|\mathcal{E}(\mathbf{X}_p) - \mathbf{M}\mathbf{c}_p\|^2 \\ &\quad \text{s.t.} \quad \mathbf{1}^T \mathbf{c}_p = 1 \quad c_{p,j} \in \{0, 1\}, \quad \forall p \in \{1, 2, ..., P\}, j \in \{1, 2, ..., K\}, \end{aligned} \tag{6}$$

and

$$\begin{aligned} &\min_{\theta_{\mathcal{E}}, \theta_{\mathcal{F}}} \sum_{i=1}^{P} \|\mathcal{F}(\mathcal{E}(\mathbf{X}_p)) - \mathbf{X}_p\|_2^2 + \lambda_2 L_1(g(\mathcal{E}(\mathbf{X}_p); \theta_1), \mathbf{c}_p) \\ &\quad + \lambda_3 L_2(g(\mathbf{c}_p, \mathbf{v}_p; \theta_2), y_p) + \lambda_4(\alpha_G - G_{k_1,k_2}), \\ &\quad \text{s.t.} \quad k_1 \neq k_2. \end{aligned} \tag{7}$$

The algorithm is elaborated in Algorithm 1.

Algorithm 1: Deep significance clustering

Input: $\mathbb{X}, \{\mathbf{v}\}, K, d$
Output: $\{\mathbf{z}_p\}_{p=1}^{P}, \{\mathbf{c}_p\}_{p=1}^{P}$
Initialization model parameters (Specially, initialization the AE parameters through equation (1);
for *i=1:*n_{iter} **do**
  Extract representations $\{\mathbf{z}\}$, use k-means to do clustering through (2). Obtain cluster membership;
  Consider cluster assignments as "pseudo-labels";
  for *j=1:*n_{epoch} **do**
    Optimize (7);
  end
end
return $\{\mathbf{z}_p\}_{p=1}^{P}, \{\mathbf{c}_p\}_{p=1}^{P}$

3.4.2 Architecture search

We choose the architecture which is trained on the training set and has the best evaluation performance on validation set, that is

$$(K^\star, d^\star) = \underset{K,d}{\operatorname{argmax}}\, AUC_{val}(\mathcal{N}(K, d, \theta)), \tag{8}$$

where $AUC_{val}(\cdot)$ is the AUC score on the validation set.

4 Experiments

We conducted experiments on two datasets and compared against 3 baseline methods. We also carried out ablation experiments to study the impact of statistical significance constraint of DICE.

4.1 Experimental setting

Data We used datasets on two patient populations: HF and COVID-19, extracted from electronic health records (EHRs) at an urban academic medical center. The datasets were split into training, validation, and test sets in a $4:1:1$ ratio.

- **HF**: We included HF patients ($n = 1585$) aged 18 to 89 from years 2014 to 2018 who were treated on the Medicine service. HF was defined by ICD-9/10-CM. Only patients whose initial and final diagnoses both contained HF were included to ensure that HF was treated during the hospital stay. The outcome is defined as discharged to home (36.8%). Sequential medical orders were included in the data.
- **COVID-19**: We included patients aged 18 to 101 who were presented to the emergency department and admitted for COVID-19 ($n = 968$) in 2020. COVID-19 was defined by a positive polymerase chain reaction test. The outcome is in-hospital mortality (15.9%). Age, race, and sequential laboratory values were included in the data.

Baselines We compared our method with baseline method including (1) principal component analysis (PCA) [21], (2) AE, and (3) AE + classification. In (2) AE, clustering was applied directly to representations learned from AE [34]. In (3) AE + classification, first we jointly trained AE and classification with representation learned from AE as the input for classification, then applied clustering to the final learned representation. No statistical constraint or NAS was used in training the baselines, but we report below the results with the same dimension of representation and cluster number with DICE. For AE, we chose the minimum reconstruction error on validation set. For AE + classification, we chose the results with the maximum AUC on the validation set.

Training We conducted experiments in PyTorch framework on NVIDIA GeForce RTX 2070. We initialized the model parameters of AE by 1 epoch training. We set $\alpha = 0.05, \alpha_G = 3.841, n_{iter} = 60, n_{epoch} = 1$. The $\lambda_1, \lambda_2, \lambda_3$ were set as 0.1, 10, 1.0 respectively based on the performance on the validation set. It took approximately 7 minutes for each result with fixed neural architecture.

4.2 Results

We used NAS to choose the best model, then qualitatively assessed our method with baselines using clustering and classification metrics. Ablation studies were also conducted to compare performance absent the statistical significance constraint.

Neural network architecture search Our search spaces were $\{(K, d)|K \in \{2, 3, 4, 5\}, d \in \{20, 25, ..., 100\}\}$ for the HF dataset and $\{(K, d)|K \in \{2, 3, 4, 5\}, d \in \{10, 11, ..., 30\}\}$ for the COVID-19 dataset, which are set according to the number of features and size of datasets. Figure 2 demonstrates the NAS process, with AUC values from the validation set of different neural network architecture on the Y-axis and d on the X-axis. From Figure 2, we can see that the statistical significance constraint can drive the model towards higher AUC, as also demonstrated in the ablation study described below. Maximizing the model AUC, $K = 4, d = 35$ for the HF dataset, and $K = 3, d = 21$ for the COVID-19 dataset, were chosen as the optimal parameters.

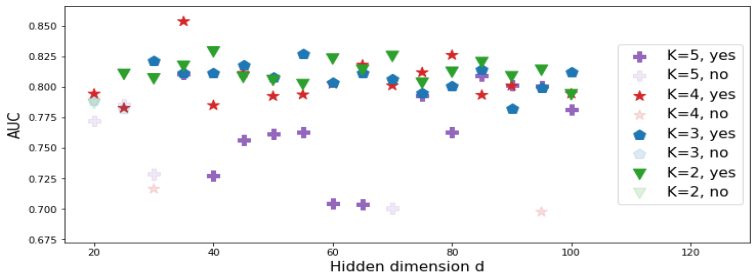
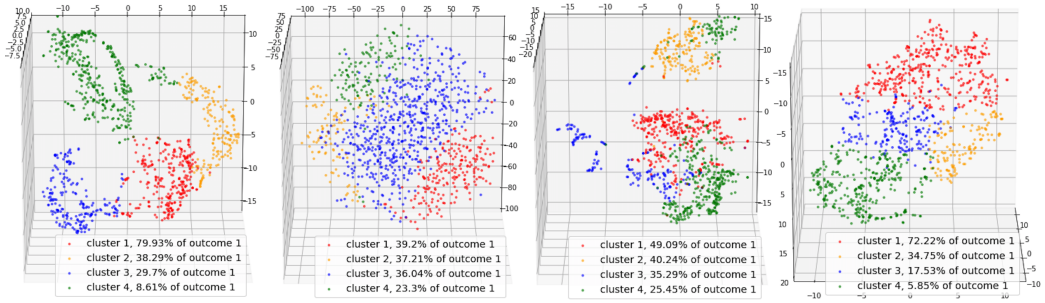


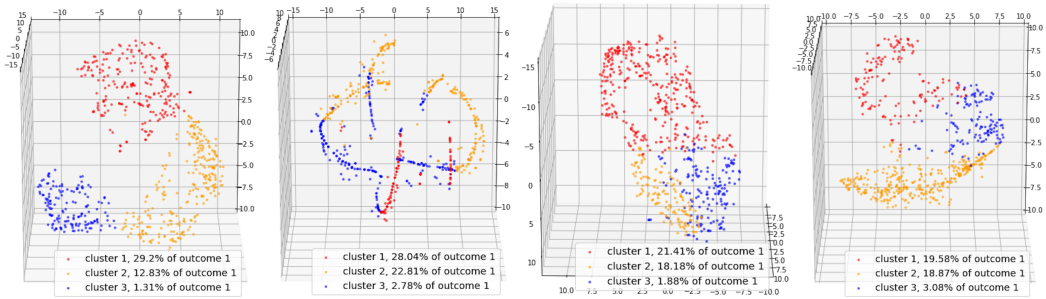
Figure 2: The model selection on HF dataset. "yes" represents that the architecture network met the significance constraint, and "no" otherwise.

Visualization of representation For the HF dataset, we demonstrate the clustering results through the visualization of representation in Figure 3. Compared with Figure 3(b), Figure 3(c) and Figure 3(d), the 4 clusters in Figure 3(a) discovered by DICE displayed tighter separation, with the highest outcome ratio 79.93% in cluster 1 to the lowest outcome ratio 8.61% in cluster 4. The baseline AE+classification also discovered 4 clusters with the outcome ratio in each cluster ranging from 72.22% to 5.85%, but the clusters are not well separated. PCA and AE did not discover clusters with outcomes as clearly separated as DICE, likely because the baseline from those two baselines are not outcome-driven. Our DICE learns representation through outcome-driven and self-supervised learning from pseudo-labels, therefore we can obtain clear outcome risk stratification and well separated clusters at the same time. Results for the COVID-19 dataset are shown in Figure 4. DICE again obtained clearer separation between clusters and outcome stratification as measured by the difference in outcome ratio within each cluster.



(a) DICE. (b) PCA. (c) AE. (d) AE + classification.

Figure 3: Visualization of patient subtyping results by various methods on HF dataset.



(a) DICE. (b) PCA. (c) AE. (d) AE + classification.

Figure 4: Visualization of patient subtyping results by various methods on COVID-19 dataset.

Table 1: Clustering performance evaluation on test set. Upper: HF dataset. Lower: Covid-19 dataset.

	Silhouette score↑	**Calinski-Harabasz index ↑**	**Davies-Bouldin index ↓**
PCA	0.0973	16.0928	2.6093
AE	0.2811	68.0664	1.7438
AE + classification	0.3458	200.0490	1.3043
DICE	**0.4838**	**212.1706**	**0.8637**
PCA	0.2478	66.6715	1.5121
AE	0.4019	150.8834	1.0514
AE + classification	0.4763	**250.4731**	0.8914
DICE	**0.4965**	182.8546	**0.7498**

Table 2: Outcome prediction comparison. Upper: HF dataset, Lower: COVID-19 dataset.

	AUC↑	**ACC↑**	**FPR↓**	**TPR↑**	**FNR↓**	**TNR↑**	**PPV↑**	**NPV ↑**
PCA	0.773	0.712	0.222	0.598	0.402	0.778	0.611	0.769
AE	0.712	0.697	**0.150**	0.433	0.567	**0.850**	0.627	0.721
AE + classification	0.818	0.765	0.251	0.794	0.206	0.746	0.647	0.862
DICE	**0.834**	**0.780**	0.257	**0.845**	**0.155**	0.743	**0.656**	**0.892**
PCA	0.757	0.772	0.221	0.731	0.269	0.779	0.387	0.938
AE	0.855	0.840	0.147	0.769	0.231	0.853	0.5	0.951
AE + classification	0.889	0.877	**0.051**	0.5	0.5	**0.949**	**0.65**	0.908
DICE	**0.907**	**0.889**	0.096	**0.808**	**0.192**	0.904	0.618	**0.961**

Clustering performance on unseen data The learned cluster membership from historic data can serve as a pseudo-label for unseen data, such that new patients may be classified into one of the risk levels. The clustering performance on the test set is shown in Table 1. Since the ground truth labels of stratification are unknown, we used Silhouette score [37], Calinski-Harabasz index [38], and Davies-Bouldin index [39] to evaluate the clustering performance. DICE achieved the best separation across the three metrics in HF dataset, and outperforms two out of three metrics in the COVID-19 dataset. DICE underperformed to AE + classification in Calinski-Harabasz index in the COVID-19 dataset. Compared to HF, the COVID-19 population has a much diverse health conditions, which may have presented a challenge to minimize within-cluster variance.

Outcome classification via learned representation We used the learned representation from DICE for outcome classification using logistic regression, as shown in Table 2. DICE outperformed the baselines in AUC, accuracy, true positive rate, false negative rate, and negative predictive value. The reason DICE had high FPR and low TNR and PPV compared to baselines may be explained by the high negative case ratio in both datasets.

Fairness on race To ensure fairness of the algorithm, we tested DICE within each demographic patient subgroups in the HF dataset. The AUC for Unknown, Asian, Other, Black, and White are 0.9053, 0.8824, 0.8563, 0.8321, 0.8470, respectively, when cluster membership is used as the predictor. The AUC for Unknown, Asian, Other, Black, and White are 0.8632, 0.8289, 0.7816, 0.8535, 0.8525, respectively, when learned representation is used as the predictor.

Ablation study We conducted an ablation experiment on the HF dataset to gauge the effect of the statistical significance constraint. When we disabled the statistical significance constraint, 2 clusters were chosen based on AUC in NAS. The distribution of outcome is 80.1% and 9.01% within the two clusters, compared to the 4-level separation in Figure 3(a). The maximum AUC score is 0.8427 in the ablation study compared to the maximum AUC score 0.8539 with the statistical significance constraint. In addition, the number of neural network which met the significance constraint significantly drops from 82.4% to 64.7% for $K = 5$. These three phenomenons indicate that statistical significance constraint contributes to clearer outcome stratification especially for bigger K.

5 Discussion

An important distinction between DICE and purely unsupervised, or supervised, tasks is that DICE learns outcome-driven clusters in an un-labeled population, where the outcome-driven clusters can later be used to assign risk-levels for future unseen cohort. When there is a lack of precise treatment protocol that applies to individual patients, such as when management plans for COVID-19 were still being developed, the capability of classifying a new patient according to the learned risk-level in the seen population can provide data-driven insights in guiding subsequent management.

In this paper, We demonstrated DICE using AE for representation learning, followed by clustering of the representation using K-means, and an alternative grid search for NAS, to discover subgroups of patients in two disease populations: HF and COVID-19. We discovered that compared to baseline, DICE largely better separated the population as measured by evaluation indices for clustering. The learned representation from DICE lead to higher AUC in classifying individual outcomes, and was further used to assign unseen data into risk-levels. This technique may be used for early identification of practice and patient patterns that suggest risks, patients who may benefit from specialized care, and patients who are on trajectory for quick recovery and early discharge pathways as a form of clinical decision support.

Broader Impact

DICE was proposed to join concepts of deep learning and statistics in healthcare to promote better acceptance of deep learning results. One of the biggest challenges to the successful application of machine learning, and especially deep learning, algorithms in healthcare is its acceptance by clinicians as an interpretable models. Traditionally, biostatistical models and concepts such as statistical significance have been better understood and accepted by clinicians and continue to be so. Thus, one implication of this work is to bridge the gap between deep learning and statistics in the context of healthcare by driving the unsupervised tasks towards statistically significant results. Aside from heart failure and COVID-19, an additional example is sepsis, which afflicts nearly 1.7 million Americans each year with a mortality rate of 270,000 patients per year. The current sepsis treatment guidelines have provided a standardized approach, including aggressive intravenous hydration and early administration of antibiotics. Though significant research has been performed with resulting improvements in overall patient morbidity and mortality, much dispute still exists regarding the indiscriminate application of this protocol to all sepsis-suspected patients. In 2018, the Infectious Diseases Society of America released a statement of non-endorsement as they felt these guidelines generated more harm, particularly for those with less severe disease. Furthermore, recent evidence suggests that multiple subtypes of sepsis may exist, suggesting that this "one-size-fits-all" solution may need to reconsidered. For clinical needs such as sepsis and beyond, DICE may provide opportunities for discovery in the nuances in diagnostics and therapy while ensuring a targeted outcome.

Acknowledgments and Disclosure of Funding

References

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[2] Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural networks*, 2(1):53–58, 1989.

[3] Rui Xu and Donald Wunsch. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.

[4] Inci M Baytas, Cao Xiao, Xi Zhang, Fei Wang, Anil K Jain, and Jiayu Zhou. Patient subtyping via time-aware lstm networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 65–74, 2017.

[5] Yiye Zhang, Rema Padman, and Larry Wasserman. On learning and visualizing practice-based clinical pathways for chronic kidney disease. In *AMIA Annual Symposium Proceedings*, volume 2014, page 1980. American Medical Informatics Association, 2014.

[6] Yiye Zhang, Rema Padman, and Nirav Patel. Paving the cowpath: Learning and visualizing clinical pathways from electronic health record data. *Journal of biomedical informatics*, 58:186–197, 2015.

[7] Boback Ziaeian and Gregg C Fonarow. Epidemiology and aetiology of heart failure. *Nature Reviews Cardiology*, 13(6):368–378, 2016.

[8] Vincent M van Deursen, Renato Urso, Cecile Laroche, Kevin Damman, Ulf Dahlström, Luigi Tavazzi, Aldo P Maggioni, and Adriaan A Voors. Co-morbidities in patients with heart failure: an analysis of the european heart failure pilot survey. *European journal of heart failure*, 16(1):103–111, 2014.

[9] Xi Zhang, Jingyuan Chou, Jian Liang, Cao Xiao, Yize Zhao, Harini Sarva, Claire Henchcliffe, and Fei Wang. Data-driven subtyping of parkinson's disease using longitudinal clinical records: a cohort study. *Scientific reports*, 9(1):1–12, 2019.

[10] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

[11] Srikanth Jagabathula, Lakshminarayanan Subramanian, and Ashwin Venkataraman. A model-based embedding technique for segmenting customers. *Operations Research*, 66(5):1247–1267, 2018.

[12] Linxiao Yang, Ngai-Man Cheung, Jiaying Li, and Jun Fang. Deep clustering by gaussian mixture variational autoencoders with graph embedding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6440–6449, 2019.

[13] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[14] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[15] Geoffrey J McLachlan and David Peel. *Finite mixture models*. John Wiley & Sons, 2004.

[16] Michel Wedel and Wayne S DeSarbo. A review of recent developments in latent class regression models. *Advanced Methods of Marketing Research, R. Bagozzi (Ed.), Blackwell Pub*, pages 352–388, 1994.

[17] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[18] Erxue Min, Xifeng Guo, Qiang Liu, Gen Zhang, Jianjing Cui, and Jun Long. A survey of clustering with deep learning: From the perspective of network architecture. *IEEE Access*, 6:39501–39514, 2018.

[19] Shi Zhong and Joydeep Ghosh. A unified framework for model-based clustering. *Journal of machine learning research*, 4(Nov):1001–1037, 2003.

[20] Srikanth Jagabathula, Lakshminarayanan Subramanian, and Ashwin Venkataraman. A conditional gradient approach for nonparametric estimation of mixing distributions. *Management Science*, 2020.

[21] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[22] Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. *The annals of statistics*, pages 1171–1220, 2008.

[23] Chris Fraley and Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, 97(458):611–631, 2002.

[24] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

[25] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[26] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 31–35. IEEE, 2016.

[27] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016.

[28] Fengfu Li, Hong Qiao, and Bo Zhang. Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognition*, 83:161–173, 2018.

[29] Laurens Van Der Maaten. Learning a parametric embedding by preserving local structure. In *Artificial Intelligence and Statistics*, pages 384–391, 2009.

[30] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[31] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[32] Wenqing Chu and Deng Cai. Stacked similarity-aware autoencoders. In *IJCAI*, pages 1561–1567, 2017.

[33] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[34] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[35] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[36] David W.. Hosmer and Stanley Lemeshow. *Applied logistic regression*. Wiley New York, 2000.

[37] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[38] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

[39] David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 1979.