Catalysis Clustering with GAN by Incorporating Domain Knowledge

Olga Andreeva
Department of Computer Science
University of Massachusetts Boston
Boston, Massachusetts, USA
olga.andreeva001@umb.edu

Wei Li
School of Artificial Intelligence and Computer Science
Jiangnan University
Jiangsu Key Laboratory of Media Design and Software Technology
Wuxi, Jiangsu, P.R.China
umass.weili@gmail.com

Wei Ding
Department of Computer Science
University of Massachusetts Boston
Boston, Massachusetts, USA
wei.ding@umb.edu

Marieke Kuijjer
Centre for Molecular Medicine Norway
University of Oslo Faculty of Medicine
Oslo, Norway
marieke.kuijjer@ncmm.uio.no

John Quackenbush
Department of Biostatistics
Harvard T. H. Chan School of Public Health
Boston, Massachusetts, USA
johnq@hsph.harvard.edu

Ping Chen
Department of Engineering
University of Massachusetts Boston
Boston, Massachusetts, USA
ping.chen@umb.edu

ABSTRACT

Clustering is an important unsupervised learning method with serious challenges when data is sparse and high-dimensional. Generated clusters are often evaluated with general measures, which may not be meaningful or useful for practical applications and domains. Using a distance metric, a clustering algorithm searches through the data space, groups close items into one cluster, and assigns far away samples to different clusters. In many real-world applications, the number of dimensions is high and data space becomes very sparse. Selection of a suitable distance metric is very difficult and becomes even harder when categorical data is involved. Moreover, existing distance metrics are mostly generic, and clusters created based on them will not necessarily make sense to domain-specific applications. One option to address these challenges is to integrate domain-defined rules and guidelines into the clustering process. In this work we propose a GAN-based approach called Catalysis Clustering to incorporate domain knowledge into the clustering process. With GANs we generate catalysts, which are special synthetic points drawn from the original data distribution and verified to improve clustering quality when measured by a domain-specific metric. We then perform clustering analysis using both catalysts and real data. Final clusters are produced after catalyst points are removed. Experiments on two challenging real-world datasets clearly show that our approach is effective and can generate clusters that are meaningful and useful for real-world applications.

KDD '20, August 23–27, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403187

CCS CONCEPTS

• **Computing methodologies** → **Cluster analysis**; *Neural networks.*

KEYWORDS

Domain-informed Clustering; Clustering Evaluation; GAN; Cancer Subtyping

ACM Reference Format:
Olga Andreeva, Wei Li, Wei Ding, Marieke Kuijjer, John Quackenbush, and Ping Chen. 2020. Catalysis Clustering with GAN by Incorporating Domain Knowledge. In *26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394486.3403187

1 INTRODUCTION

Cluster analysis serves as an essential tool for effective exploratory data analysis and knowledge discovery. During this process, similar items are grouped together and distinct samples are separated. Traditional approaches, such as Euclidean or Manhattan distances, rely on distance-based metrics to measure how similar or dissimilar two items are. But challenges arise when we need to produce clusters useful for domain applications. Standard metrics may not necessarily reflect sample similarity from the perspective of a specific domain or application. Hence, it is important to pick a good domain-oriented distance metric. Another challenge involves comparing objects that lack natural ordering, such as categorical data. For instance, how similar are two lung cancer patients with different mutated genes, but with similar responses to the same type of treatment? This question can be easily answered by an oncologist, who will conclude that these two patients should belong to the same cluster. Hence, effective incorporation of domain knowledge to replace generic metrics can generate better clusters more useful for domain applications. Works that tackle these challenges are

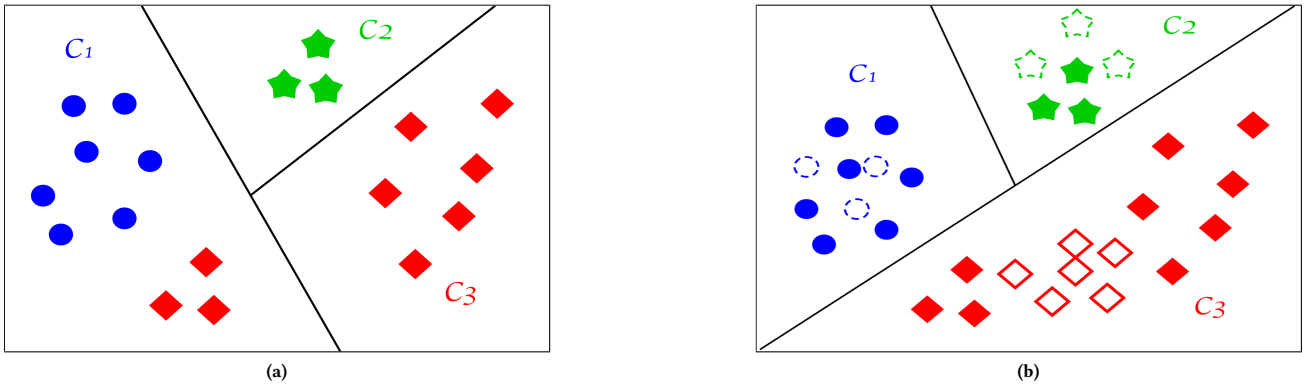


Figure 1: a) Due to insufficient sample collection, a clustering algorithm may group points into three clusters C_1, C_2 and C_3 as depicted above. However, after domain experts analyze these clusters, they may conclude that from the domain knowledge perspective, samples such as the red squares in cluster C_1 have characteristics more similar to those in cluster C_3, so there really should be 3 clusters as coded by blue, green, and red. b) Our approach will keep synthetic points (hollow red squares as shown above) – drawn from the learned data distribution – so that wrongly clustered samples (e.g., three red squares in the left bottom) will be assigned to the correct cluster. Not all synthetic samples are useful, such as the dashed blue circles and dashed green stars that have no effect on clustering. Instead, we focus on sampling useful synthetic points (hollow red squares), which are called catalyst points or catalysts. We name this clustering framework with the help of catalysts as Catalysis Clustering.

not only of high research importance for the machine learning community, but is also practically crucial for domain scientists.

Existing distance-based clustering methods often fail to work with high-dimensional data [29]. The reason behind this is that data in a high-dimensional space tends to become geometrically sparse, and many distance metrics become ineffective and less meaningful in a high-dimensional space. A traditional approach is to reduce the dimensionality of a given feature space. However, generic techniques such as PCA do not take into account a feature's importance based on domain knowledge. If we drop important features, some samples will be wrongly brought closer together in the reduced space. This can negatively impact the quality of final clusters. Both these challenges call for a domain-specific similarity metric to produce clusters useful in practice. For example, we conducted a cancer subtyping case study, in which cancer patients needed to be clustered into subtypes. This clustering procedure becomes clinically meaningful if patients in the same subtype respond similarly to the same option of treatment (e.g., chemotherapy, radiation). Cancers are generally caused by gene mutations, which can range from a few in one patient to a few thousand in another patient, often with very little overlap even for patients of the same cancer type.

To address the aforementioned challenges, we propose a novel clustering framework, which we call Catalysis Clustering in order to incorporate domain knowledge into clustering analysis. To further illustrate our idea, assume there is a group of patients diagnosed with lung cancer. A clustering algorithm divides them into three clusters as shown in Figure 1(a). Results are then analyzed by a physician, who concludes, that from the medical perspective, the three squares assigned to cluster C_1 have characteristics (treatment responses, survival rates, etc.) similar to those in cluster C_3. In Catalysis Clustering, we propose to use Generative Adversarial Networks (GANs) to generate synthetic data points, determine the validity of these synthetic data through domain knowledge, then utilize them in cluster analysis. In Figure 1(b) these synthetic points are represented as dashed and hollow points. Not all of these points turn out to be useful, such as with the blue and green dashed points in Figure 1(b) that have no effect on clustering. On the other hand, hollow red points connect three misclustered red squares with the rest of red squares, changing boundaries of clusters C_1 and C_3. Since not all of the synthetic points turn out to be helpful, we evaluate them through the domain-specific metric. If no metric improvement is reached, unhelpful synthetic points are dropped. By this way, our approach leads to more useful and domain-relevant clusters. One of the advantages of our Catalysis Clustering framework is that it can work with any existing clustering algorithm. To our best knowledge, our method is the first to use GAN-generated data in clustering analysis.

In summary, the major contributions of this paper are:

- We introduce a new clustering analysis framework called Catalysis Clustering, which utilizes domain knowledge to produce clusters useful for domain applications and works with any existing clustering algorithm.
- Our framework can adopt any clustering algorithm, which makes it independent from any domain or clustering approach.
- How to evaluate data generated by GANs remains a serious challenge in machine learning, which is even more difficult for numeric data. With the incorporation of domain knowledge, we develop an approach to assess the quality of GAN-generated numeric data and its usefulness for clustering analysis.
- We chose a critical and challenging real-world application of cancer stratification to validate our idea. Experiment results show both the effectiveness and usefulness of our approach.

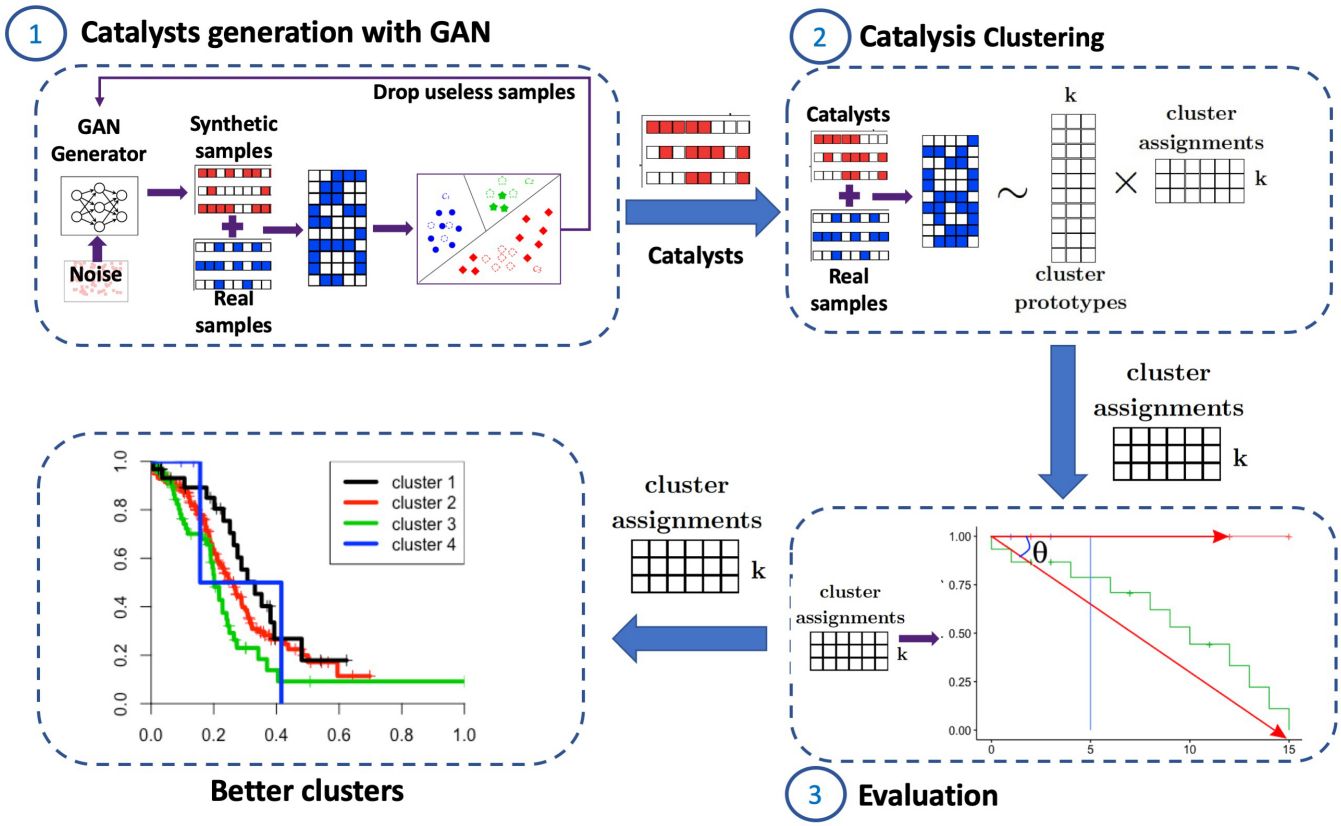


Figure 2: Catalysis Clustering framework. 1) Catalysts generation with GAN: At this step we use trained GANs to generate catalysts (special synthetic points drawn from the learned distribution to improve clustering quality) which we combine with the real data. The quality and usefulness of these points is further evaluated through domain knowledge. 2) Catalysis Clustering: we perform clustering analysis on the combined dataset. The advantage of our method is that any clustering algorithm could be adopted here. For example, a non-negative matrix factorization-based clustering algorithm is illustrated above. 3) Evaluation: Catalysts are removed and resulted clusters are then assessed with domain-specific metric. Catalysis Clustering produces better clusters more useful in practice.

2 RELATED WORK

Clustering analysis is a powerful tool for effective exploratory data analysis, and has been extensively studied in data mining and applied in many real-world applications. For example, k-means clustering [13] was applied in [11] to identify distinct asthma phenotypes. Consensus clustering [22] was adopted for retrospective identification of intensive care unit patients [32]. Unfortunately, existing algorithms highly depend on distance metrics and do not necessarily provide results useful for domain applications. DiSC [3] attempted to incorporate domain-specific usefulness scores, provided for each sample, into the semi-supervised dimension reduction clustering approach. Alas, there are domains where scientists do not have consensus on which scores to use. Recognizing the limitation of generic clustering methods, a wide range of domain specific algorithms were introduced recently, e.g. CoINcIDE for clustering across multiple cancer datasets [27], Network-Based Stratification (NBS) [16] for clustering analysis on gene mutations. In [7] a survival analysis was incorporated into the clustering validation process. Unfortunately, these domain-specific techniques are often hard to generalize.

In our Catalysis Clustering framework, besides clustering algorithm, the other important component is a synthetic data generator, which we adopt from the state-of-the-art Generative Adversarial Networks [10]. GANs have attracted much attention for their ability to capture a data distribution by two neural networks, which can be useful for many fields including clustering. Recent work includes the following. ClusterGAN [8, 23] enables clustering in the latent space. CatGAN [28] trains one of its networks to classify the data into a predefined number of categories. InfoGAN [5] learns disentangled representations used for clustering. Task-oriented GAN [21] tackles difficulties in PolSAR image interpretation, where a special network T-Net is employed to accomplish a certain task. While all these methods incorporate GANs directly into the clustering process, our approach only uses GANs' ability to learn underlying data distribution to further generate synthetic points.

3 CATALYSIS CLUSTERING WITH GAN

In this section, we introduce our Catalysis Clustering framework. Catalysis Clustering framework improves the usefulness and quality of clustering analysis and allows experts to apply domain knowledge for clustering analysis to generate useful clusters for domain-specific applications.

3.1 Problem formulation

The main focus of Catalysis Clustering is to utilize domain knowledge during clustering analysis to improve the quality of produced clusters for domain applications.

Let us assume we are given a numeric dataset X with an unknown distribution P_X, and k for the number of clusters. k is not required by our Catalysis Clustering framework, but may be required by the specific clustering algorithm, for instance the one adopted in Section 4.3. Also, assume that we choose a clustering algorithm *Clust* and a metric $\mathcal{M}$ designed by an expert to assess the performance of *Clust* from the domain perspective. Our goal is to find a set of clusters $\{X_1, ..., X_k\}$, $X_i \subset X$ and $X_1 \cap \cdots \cap X_k = \emptyset$:

$$Clust(X) = \{X_1, ..., X_k\},$$

which maximizes the domain metric $\mathcal{M}$. Thus the optimization problem is defined by:

$$\max_{\substack{X_i \subset X \\ X_1 \cap \cdots \cap X_k = \emptyset}} \mathcal{M}(Clust(X)) \tag{1}$$

3.2 Catalysis Clustering Architecture

The architecture of the Catalysis Clustering framework is shown in Figure 2. Our main idea is to improve the usefulness and quality of resulting clusters by introducing catalysts - synthetic samples generated from the real data distribution. Similar to catalysts in chemistry, these catalyst samples are used to enable and improve clustering quality and they will be removed afterward, meaning they do not participate in a final clustering evaluation. Our Catalysis Clustering framework includes the following three stages:

(1) **Catalyst generation with GAN**: GANs fit data distribution P_X to generate a set of catalysts C. There are two requirements for generated catalyst samples: (1) they must follow the original data distribution ensured by GANs; (2) they must improve clustering quality as specified by the domain knowledge.
(2) **Catalysis Clustering**: Perform clustering over the combined dataset $X \bigcup C$ to obtain $\{(X_1 \bigcup C_1), ..., (X_k \bigcup C_k)\}$.
(3) **Evaluation**: Every C_i is removed from the respective clusters and $\{X_1, ..., X_k\}$ is evaluated with the metric $\mathcal{M}$. It is important to note that the set C is only used during the clustering stage and disregarded afterwards, so only real data is evaluated.

3.3 Stage 1: Catalyst generation with GAN

Synthetic sampling techniques, such as SMOTE [4], Borderline-SMOTE [12], and ADASYN[14], are effectively used in imbalanced learning[15]. Most existing methods attempt to populate under-represented classes within existing data to reduce data imbalance and provide a balanced dataset. Our approach is fundamentally different as we apply synthetic sampling to the whole data and the quality of these synthetic samples is sufficiently assessed through domain knowledge. We only use synthetic samples of high quality, which we name ***catalysts***.

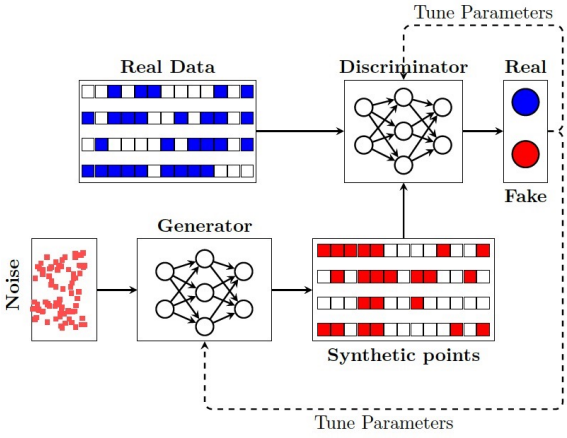


Figure 3: Learning underlying data distribution with GANs to generate catalyst samples.

Definition 1 (Catalyst). *Given a set X, an underlying distribution P_X, clustering algorithm Clust and an evaluation metric $\mathcal{M}$ to maximize, a synthetic point c is a* ***catalyst*** *if it satisfies two properties:*

(1) $c \sim P_X$
(2) $\mathcal{M}(S_1) > \mathcal{M}(S_2)$,
where S_1 is a new cluster assignment on $X \cup \{c\}$,
and $\{c\}$ is excluded from the final assignment, i.e.
$S_1 = Clust(X \cup \{c\}) \setminus \{c\}$
and S_2 is an initial cluster assignment on X, i.e.
$S_2 = Clust(X)$

To satisfy the first requirement, we adopt a variant of Generative Adversarial Networks introduced by Goodfellow and others in [10]. In its original form GANs do not require any prior knowledge about data and learn to map from a latent space to a data distribution of interest, P_X in our case. To achieve that, GANs use two competing adversarial models: a generator G and a discriminator D. The main goal of G is to capture the data distribution P_X and generate data samples similar to original data X. Meanwhile, D estimates the probability that a given sample is derived from P_X. Thus, the generator and discriminator models compete against each other as described by the equation (2).

$$\begin{aligned} \min_G \max_D V(D,G) = & E_{x \sim P_X}[log D(x)] \\ & + E_{z \sim P_z}[log(1 - D(G(z)))] \end{aligned} \tag{2}$$

Here $V(D, G)$ is a value function, $x \in X$ is a data sample, z is a noise sample, and P_z is a prior noise distribution. $D(\cdot)$ and $G(\cdot)$ are discriminator and generator functions respectively. $D(\cdot)$ returns a probability of the sample being real ($D(\cdot)$ =1) or fake ($D(\cdot)$ =0). During the training phase, $G(\cdot)$ seeks to minimize $log(1 - D(G(z)))$, meaning maximizing $D(G(z))$. On the contrary, $D(\cdot)$ seeks to minimize $D(G(z))$, meaning the discriminator should reject a fake sample $G(z)$ with high probability. In the end, G fits the data distribution

P_X to fool D with the generated samples. Figure 3 depicts the aforementioned process. As a result, $G(z) = c \sim P_X$ satisfies the first requirement for a synthetic point to be a catalyst.

The second requirement ensures that a synthetic sample is useful, i.e. improves clustering quality. Not every GAN-generated sample will satisfy this requirement, and a domain-specific metric should be applied to check whether this sample improves clustering quality or not. More formally, our method requires every synthetic sample c to be assessed with the metric $\mathcal{M}$, which is designed by a domain expert. If c introduces an improvement to the quality of produced clusters, i.e. $\mathcal{M}(S_1) > \mathcal{M}(S_2)$, where $S_1 = Clust(X \cup \{c\}) \cap \{c\}$ and $S_2 = Clust(X)$, then c is useful. In summary, a synthetic sample that comes from the original data distribution and is useful will be chosen as a catalyst to participate in the next Catalysis Clustering stage.

Algorithm 1: Catalysis Clustering

Input : a set X, a set of generated catalysts C, evaluation metric $\mathcal{M}$, clustering algorithm *Clust* and, if *Clust* requires, a number of clusters k

Output : cluster assignments $\{X_1, ..., X_k\}$

```
/* combine X and a set of catalysts C          */
```
$\tilde{X} \longleftarrow X \bigcup C$;
```
/* Cluster combined dataset X̃                  */
```
$\{\tilde{X_1}, ..., \tilde{X_k}\} \longleftarrow Clust(\tilde{X})$;
```
/* delete catalysts from each cluster          */
```
$\{X_1, ..., X_k\} \longleftarrow \{\tilde{X_1}, ..., \tilde{X_k}\} \setminus C$;

3.4 Stage 2: Catalysis Clustering

The main objective for this stage is formulated as follows:

(1) **Given:** a dataset X, a set of catalysts C, a domain specific metric $\mathcal{M}$, a clustering algorithm *Clust*, and a number of clusters k, if required by *Clust*
(2) **Find:** cluster assignment $\{X_1, ..., X_k\}$.
(3) **Objective:** optimize equation (1).

Algorithm 1 describes the process of Catalysis Clustering. As an input, Catalysis Clustering takes a dataset X, a set of generated catalysts C, a domain specific evaluation metric $\mathcal{M}$, and a clustering algorithm *Clust*. *Clust* in practice should be substituted with a specific clustering algorithm, such as K-means, NMF, etc. The number of clusters k is not required by our Catalysis Clustering framework, but could be required by the specific clustering algorithm selected for this step. In the beginning, the set of catalysts C is combined with X and *Clust* is applied to the combined dataset. After clustering is finished, all catalysts are removed from each cluster. Resulting cluster assignments $\{X_1, ..., X_k\}$ are then evaluated with $\mathcal{M}$ in the next stage.

3.5 Stage 3: Evaluation

Catalysis Clustering requires specific and quantitative assessment from the domain knowledge perspective, which can be challenging in practice. One of the main issues is the lack of an accurate yet general definition of what a bad/not useful or good/useful "cluster" is. There exist many generic metrics for clustering evaluation, such as Normalized Mutual Information or Rand Index. However, they require a set of "true" clusters and they do not take into consideration whether clusters will be useful for domain applications. On the contrary, Catalysis Clustering incorporates domain knowledge into the quality assessment and does not require the knowledge of the ground truth cluster assignments. In Stage 1, a synthetic sample is constantly evaluated according to a domain-specific metric $\mathcal{M}$ before it can be selected for the forthcoming clustering process. In this case, catalysts serve as an adjusting and enabling mechanism, and with their help we are able to explore and evaluate various cluster boundaries. The fact that catalysts are derived from the original data distribution makes them valid candidates to make up the deficiency in data collection. In Stage 3, the same $\mathcal{M}$ is applied to the final clustering results to give a quantitative measure of the clustering quality.

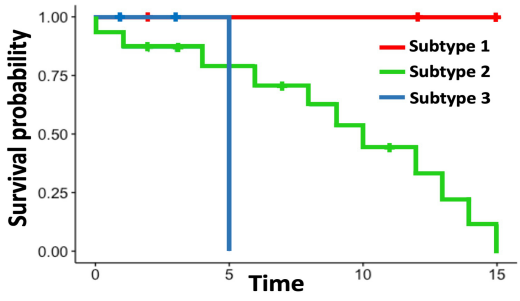


Figure 4: A sample survival plot with three subtypes.

Use of the domain knowledge during the cluster analysis highly benefits real-world applications. In a given domain of interest, the underlying model from which our data came from will likely be known and well-defined. In this case we can use existing rules and constraints to set or design the metric $\mathcal{M}$ for evaluation. As an example, in our case study we assess gene mutation subgroups with the help of survival information from a group of patients. Such a domain-specific approach helps to incorporate more knowledge to build a better picture of the underlying model, yet remain flexible to plug in different metrics to study different aspects even in the same application or dataset.

In the next section, we present a case study to develop a domain-specific metric for Catalysis Clustering. To achieve this, we make use of a Kaplan-Meier estimator [9], which is widely used for time-to-event data evaluation—a common issue in the medical domain. The Kaplan-Meier estimator is used to build a survival plot similar to the one shown in Figure 4.

4 CASE STUDY: CANCER STRATIFICATION USING GENE MUTATION DATA

Cancer stratification aims at grouping cancer patients into clinically meaningful subtypes based on specific characteristics of their cancer type. This problem is of particular interest to machine learning researchers and medical researchers because mutation-based clustering analysis is still in its infancy, and medical science still has no complete theory for the cancer stratification process. Useful mutation-based clustering analysis and correct prediction of clinical

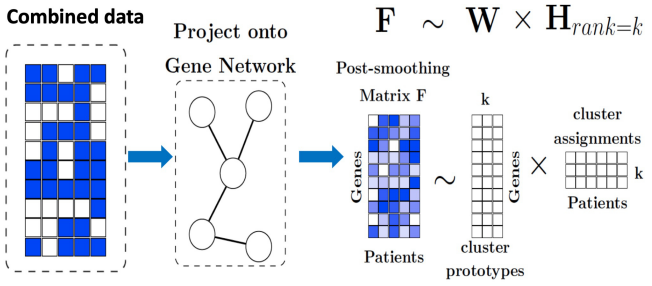


Figure 5: After catalysts are combined with real data, the whole dataset is projected onto a gene interaction network transforming each binary vector into a continuous activation profile on the network. Non-negative matrix factorization is applied to the smoothed matrix F.

outcomes for each subgroup becomes the key to successful patient treatment.

Mathematically, the problem of mutation-based cancer stratification may be formulated as follows. Assume we have a set G of g genes, collected from n patients, and k is the number of subtypes. k exists in several types of cancers developed by medical researchers. Thus, the resulting data is stored in the $n \times g$ matrix

$$P = [p_{i,j}]_{i=1..n}^{j=1..g},$$

where $p_{i,j}$ is either 0 or 1 to indicate whether a gene is mutated or not. In this way, the i^{th} row represents a particular patient, while the j^{th} column represents whether the gene j is mutated(1) or not(0). The task is to assign each mutation profile $p_{i,j}$ to one of k subtypes.

In our Catalysis Clustering framework, we adopt Network-Based Stratification (NBS) [16] as the clustering algorithm. Patients with the same type of cancer may not have common mutations in their genes, and mutated genes in one patient may range from a few to a few thousand [18, 19] which is quite sparse considering there are over 20,000 human genes. To deal with a high level of data sparseness, NBS projects each mutation profile onto a human gene interaction network to spread the influence of each mutation over its network neighborhood and generate a less sparse feature matrix F. A non-negative matrix factorization further approximates F with a low-rank matrix approximation [20] such that $F \approx WH$. Figure 5 depicts this process.

4.1 Development of the Survival Curve Measure as a clustering evaluation metric

In this case study, our goal is to help physicians answer the following questions from cancer patients through cancer stratification:

- How likely am I to survive from cancer? A probability close to 0 (i.e., death) or 1 (i.e., to stay alive) is preferred for a more definitive answer.
- How long will I live? The more specific the period is, the better.

Accurate information to answer these questions is highly desirable and crucial in a clinical setting. Our underlying idea is that a subgroup of patients whose clinical outcomes are similar, and drastically differ from any other subgroup of patients, should belong to the same cluster. To use this notion during Catalysis Clustering, we need to convert it to a quantitative measure, so that cluster quality can be described with a numerical value. Additionally, we need to decide on how edge cases—high clustering quality vs. low clustering quality—would look like. It is important to note that the following process is general and can be used as a template for metric design in any other domain.

In this case study, we adopt the Kaplan-Meier estimator and survival curves for our metric design, similar to [7]. Suppose we have survival data for n cancer patients divided into 3 clusters. Based on this information, the Kaplan-Meier estimator produces a survival plot as shown in Fig. 4. Here each colored line (survival curve) indicates a cluster and marks (plus signs) on each line indicate a living patient. These 3 clusters represent what we consider as three extreme cases in cancer subtyping. The horizontal red line of survival curve 1 indicates that all patients are alive with a 100% survival rate at the end of the monitoring period (15 months in this case). In other words, if a patient is assigned to this cluster, a physician can confidently tell him that with the corresponding treatment he has a great chance to survive for at least 15 months. As for subtype 3, all subjects are unlikely to pass the 5 month threshold. Both subtype 1 and 3 are of high clustering quality from the perspective of clinical findings because they provide more specific information on the survival rate. On the contrary, subtype 2 is not as clear as subtypes 1 and 3 because a patient may pass away anytime between 0 and 15 months. Subtype 2 is not informative for physicians, because it does not provide any definite information for a patient's lifespan.

Taking all the above information into account, we design a domain-specific metric, which we call a **Survival Curve Measure** (**SCM**), to determine the quality of a cluster defined by a survival curve. In the best case scenario, the survival curve should resemble subtypes 1 or 3 in Figure 4. As for the worst case, such as with subtype 2, survival periods vary and are less definite. The more "steps" a survival curve has, the lower its prediction value is. Thus, with $time \rightarrow \infty$, survival curve 2 would look more like a diagonal line. With this insight, we measure the angle between the diagonal line and each survival curve. This angle would represent how far our curve is from the worst-case scenario, which is a diagonal line. The closer the angle is to 45°, meaning the further a survival curve is from the diagonal line, the higher the clustering quality is. For instance, subtypes 1 and 3 in Figure 4 both have a 45° angle from the diagonal line. Equation (3) computes the angle at a point (x,y) and Figure 6 provides a visual explanation for what angle θ_j is.

$$\theta = arctan\frac{k_2 - k_1}{1 + k_1 k_2} \tag{3}$$

Here k_1 is the slope of the diagonal line, i.e. the line $(0,1) - (1,0)$, k_2 is the slope of the line $(0,1) - (x,y)$, and j is the cluster that a point (x,y) belongs to.

The SCM for a survival curve j is described by equation (4), where n_j is the total number of points in cluster j.

$$SCM_j = \frac{\sum_{i=1}^{n_j} \theta_i}{n_j} \tag{4}$$

This measure is used to determine the quality of cluster j.

Equation (5) describes the angle value assigned to each survival plot. SCM_{avg} represents a weighted sum of calculated angles so

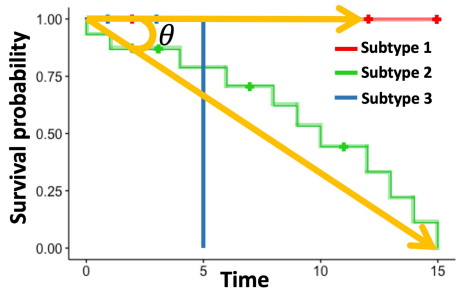


Figure 6: A visual representation of the θ angle. Here θ is computed between the diagonal line and a line (0,1) - (15,1)

Table 1: Statistics of the experimental data.

Cohort	#Patients	#Genes	#Mutations (min - max)
Lung	381	15,967	12 - 2,048
Ovarian	356	9,850	1 - 181

that subtypes of a smaller size (i.e. contains a small number of patients) do not unfairly skew the total outcome. Hence, SCM_{avg} is used to determine the quality of a survival plot, as shown in the equation,

$$SCM_{avg} = \sum_{j=1}^{k} \frac{n_j}{N} SCM_j, \tag{5}$$

where k is the number of clusters, n_j is the cardinality of the cluster j, and N is the total number of points.

Although our evaluation metric is designed for cancer stratification, it can easily be adjusted to other life-threatening diseases. Even for diseases with no fatalities, we can replace the survival time by the recovery time or other clinical markers. Similar metrics can be developed for other fields, such as in social analysis [17, 31], predicting corporate survival [26], or financial analysis [1].

4.2 Data description

To test our approach, we used somatic mutation profiles of ovarian and lung cancer cohorts, collected from major projects such as The Cancer Genome Atlas (TCGA) [6] and the International Cancer Genome Consortium (ICGC) [24, 25]. Table 1 shows the statistics for these two datasets. For each patient, each dataset contains a mutation profile—a binary vector where 0 stands for a non-mutated gene and 1 for a mutated gene. Somatic mutation profiles are extremely challenging to work with for a couple of reasons. First, these profiles are extremely sparse, such as with the ovarian cancer dataset, where on average a patient has fewer than 200 mutations out of 9,850 genes. Another challenge is that somatic mutation profiles are remarkably heterogeneous, and two clinically identical patients typically do not share more than a single mutation[19]. Hence, standard distance-based clustering algorithms fail in the task of mutation-based cancer stratification.

For the evaluation step, we used survival data collected from [16]. These datasets contain information on patients' age, gender, survival information (i.e. the number of days lived during the observation period), etc. Unfortunately, we do not have a full correspondence between mutation and survival profiles. As a result, in the case of ovarian cancer, survival information was only collected for 325 subjects (out of 356), and in the case of lung cancer, information on only 303 patients was present (out of 381).

4.3 Experiment setup

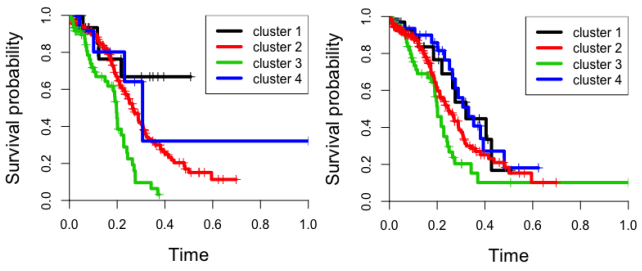
In all of our experiments, we used an improved version of GANs, Wasserstein GAN (WGAN) [2], which takes into account the distances between distributions. Since our data is discrete and very sparse, some data preprocessing was performed for WGAN to work. In our experiments WGAN was trained for 25,000 epochs. Because GANs do not generate discrete data, we setup a threshold of 0.5 to convert generated continuous data into binary form.

We chose Network-Based Stratification (NBS) [16] to compare with Catalysis Clustering (CC) since NBS represents the most advance state-of-the-art in mutation-based cancer stratification. NBS requires the mutation profile of each patient to be projected onto a human gene (protein) interaction network. Then, network propagation is used to spread the influence of each mutation over its network neighborhood to produce a non-sparse feature matrix. Finally, NMF [20] is applied to the smoothed matrix to stratify the gene dataset (Figure 5). This approach was proven to work in [16]. We apply the NBS algorithm to the original dataset and it serves as a baseline. In [16] NBS identified 4 subtypes for the ovarian cancer cohort and 6 subtypes for the lung cancer cohort, which were significant predictors of patient survival time. Thus, for a fair comparison, we chose $k = 4$ for the ovarian case study and $k = 6$ for the lung case study.

Because true clusters are unknown and yet to be discovered, it is impossible to incorporate a standard metric like Normalized Mutual Information or Adjusted Rand Index into the clustering evaluation process, which is often the case in practice. Instead, along with *SCM*, we also compute the log-rank test and its corresponding p-value as an external measure of clustering quality. The log-rank test is a form of a χ^2 test [30] and is often used to compare the survival distributions of two samples. It calculates a statistic to test the null hypothesis H_0, which is that there is no difference in survival between two or more independent groups (i.e. the probability of a death occurring at any time point is the same for each group). The lower the p-value, the greater our confidence is that the survival curves are statistically significantly different. The *SCM* and the log-rank test together provide a comprehensive evaluation of the usefulness and distinctiveness of the resulting clusters.

4.4 Ovarian cancer stratification

Figure 7(a) illustrates a survival plot based on NBS clustering assignments and Figure 7(b) illustrates based on Catalysis Clustering assignments. Table 2 shows that, according to SCM_{avg}, CC achieves better results compared to NBS. Although CC achieved improvements in two out of four *SCM* values, SCM_{avg} clearly shows that the new cluster assignment is better from the domain knowledge perspective.



(a) NBS
p = 0.00002

(b) Catalysis Clustering
p = 0.02

Figure 7: Ovarian cancer case study: Kaplan-Meier survival plots for (a) NBS subtypes and (b) Catalysis Clustering subtypes. Time scale was normalized to have values between 0 and 1. Both p-values are low, which gives us high confidence that in both cases, survival curves are statistically different.

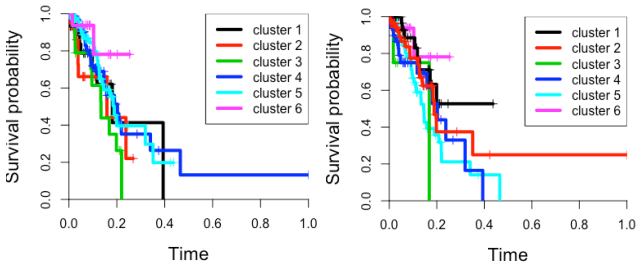
Table 2: Ovarian cancer stratification evaluated with Survival Curve Measure (SCM)

Model	SCM_1	SCM_2	SCM_3	SCM_4	SCM_{avg}
NBS	10.58	13.69	15.92	25.09	15.80
CC	**10.97**	**16.08**	12.74	20.53	**16.13**

Table 3: Ovarian cancer case study: contingency table for cluster assignments

	CC_1	CC_2	CC_3	CC_4	total
NBS_1	**3**	8	3	7	**21**
NBS_2	23	**170**	24	10	227
NBS_3	0	8	**1**	5	**14**
NBS_4	8	5	5	**45**	68
total	**34**	191	**33**	67	325

Table 3 represents the contingency table for cluster assignments. As Table 3 clearly shows, NBS assignments contain two small clusters: NBS_1 of size 21 and NBS_3 of size 14. Catalysis Clustering is able to increase the size of these two clusters by approximately 57% and 142% respectively. The diagonal line in Table 3 shows how many samples are assigned to the same clusters by NBS and CC. These results support our assumption, that Catalysis Clustering is capable of re-assigning relatively far-away samples to more relevant clusters. Table 3 shows that most samples from clusters 2 and 4 are assigned to the same cluster, and more uncertain samples are rearranged among other clusters. Even though these identified improvements may seem marginal, in a critical cancer stratification task, even a small improvement from our algorithm would mean a significant improvement in patient treatment. In our case, **106 re-assigned patients (or 33% of all patients) have a higher chance to receive correct treatment in time**, meaning many more lives could be saved.



(a) NBS
p = 0.2

(b) Catalysis Clustering
p = 0.08

Figure 8: Lung cancer case study: Kaplan-Meier survival plots for (a) NBS subtypes and (b) Catalysis Clustering subtypes. Time scale was normalized to have values between 0 and 1. The NBS p-value is significantly higher than the CC p-value, which indicates a much lower confidence that NBS-produced subtypes are statistically different. On the contrary, the CC p-value is low, so we can say with a high confidence that CC-produced survival curves are statistically different, which means that the probability of a death occurring at any point in time is different for each group.

4.5 Lung cancer stratification

Out of 381 patients within the collected mutation profiles, only 303 have survival information, 22 of which cannot be used due to missing data, so only data from 281 patients is used for the evaluation. Another complication is having a higher number of clusters as $k = 6$.

Figure 8(a) illustrates a survival plot based on NBS clustering assignments and Figure 8(b) illustrates CC clustering assignments. According to the log-rank test and the p-value, Catalysis Clustering produces more distinctive subtypes compared to NBS. In their work, Hofree et al. achieved better results for the log-rank test by removing unstable samples identified after the consensus clustering process. Our approach was able to achieve even better results without the elimination of data. Table 4 shows that CC achieves better SCM_{avg} results compared to NBS. Although CC achieved improvements in three out of six SCM values, SCM_{avg} clearly indicates that the new cluster assignment is better from the domain knowledge perspective. Table 5 is the contingency table for NBS

Table 4: Lung cancer stratification evaluated with SCM

Model	SCM_1	SCM_2	SCM_3	SCM_4	SCM_5	SCM_6	SCM_{avg}
NBS	30.81	34.00	31.61	20.94	19.99	18.73	23.89
CC	23.80	26.22	**31.95**	**29.49**	16.82	**20.76**	**24.05**

and CC cluster assignments. Similar to the previous case study, approximately 28% of patients were re-assigned, meaning **82 people have higher chances to receive proper treatment in time.**

5 CONCLUSION

In this paper, we present a Catalysis Clustering framework which incorporates domain knowledge into our approach. This framework

Table 5: Lung cancer case study: contingency table for cluster assignments

	CC_1	CC_2	CC_3	CC_4	CC_5	CC_6	total
NBS_1	19	8	0	4	3	0	34
NBS_2	5	15	2	9	1	0	32
NBS_3	0	0	19	7	0	0	26
NBS_4	8	19	0	69	2	0	98
NBS_5	7	1	0	5	83	0	96
NBS_6	0	0	0	0	1	16	17
total	39	43	21	94	90	16	303

can accommodate various clustering algorithms and utilize domain knowledge to produce clusters useful for domain applications. With the help of GAN-generated catalysts, which are special synthetic points drawn from the real data distribution, deficiencies in data collection can be overcome and clustering quality can be improved. To our best knowledge, Catalysis Clustering is the first work to utilize GAN-generated numerical samples during the clustering process. We also showed how to evaluate both GAN-generated catalysts and clusters with domain knowledge. Hence, Catalysis Clustering produces not only groupings of similar samples, but clusters that are of higher quality and usefulness to domain scientists. Experiments on two challenging real-world datasets show both the effectiveness and usefulness of our approach. Although our case study focuses on cancer research, Catalysis Clustering is independent of domain, clustering algorithm, or any particular evaluation process. Both Catalysis Clustering framework and the domain knowledge metric design procedure can be easily adapted by various domains.

ACKNOWLEDGMENTS

This work was supported by an NSF award 1743010. Project title: NSF: Advanced Machine Learning Techniques to Discover Disease Subtypes in Cancer. We would like to thank Alexander Choe for his help in proofreading this manuscript. We would also like to thank anonymous reviewers for their insightful comments.

REFERENCES

[1] L. N. Allen and L. C. Rose. 2006. Financial Survival Analysis of Defaulted Debtors. *The Journal of the Operational Research Society* 57, 6 (2006), 630 – 636.
[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein gan. *arXiv preprint arXiv:1701.07875* (2017).
[3] Yale Chang, Junxiang Chen, Michael H Cho, Peter J Castaidi, Edwin K Silverman, and Jennifer G Dy. 2017. Clustering with domain-specific usefulness scores. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 207–215.
[4] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.
[5] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*. 2172–2180.
[6] The International Cancer Genome Consortium. 2010. International network of cancer genome projects. *Nature* 464 (15 04 2010), 993 – 998. http://dx.doi.org/10.1038/nature08987
[7] Pietro Coretto, Angela Serra, Roberto Tagliaferri, and Jonathan Wren. 2018. Robust clustering of noisy high-dimensional gene expression data for patients subtyping. *Bioinformatics* (2018).
[8] Kamran Ghasedi, Xiaoqian Wang, Cheng Deng, and Heng Huang. 2019. Balanced self-paced learning for generative adversarial clustering network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4391–4400.
[9] Manish Kumar Goel, Pardeep Khanna, and Jugal Kishore. 2010. Understanding survival analysis: Kaplan-Meier estimate. *International journal of Ayurveda research* 1, 4 (2010), 274.
[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.
[11] Pranab Haldar, Ian D Pavord, Dominic E Shaw, Michael A Berry, Michael Thomas, Christopher E Brightling, Andrew J Wardlaw, and Ruth H Green. 2008. Cluster analysis and clinical asthma phenotypes. *American journal of respiratory and critical care medicine* 178, 3 (2008), 218–224.
[12] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. 2005. Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing*. Springer, 878–887.
[13] J. A. Hartigan and M. A. Wong. 1979. Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28, 1 (1979), 100–108. https://doi.org/10.2307/2346830
[14] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. IEEE, 1322–1328.
[15] Haibo He and Edwardo A Garcia. 2009. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering* 21, 9 (2009), 1263–1284.
[16] Matan Hofree, John P Shen, Hannah Carter, Andrew Gross, and Trey Ideker. 2013. Network-based stratification of tumor mutations. *Nature Methods* 10 (15 09 2013), 1108 – 1115. http://dx.doi.org/10.1038/nmeth.2651
[17] Christian O Jacke, Iris Reinhard, and Ute S Albert. 2013. Using relative survival measures for cross-sectional and longitudinal benchmarks of countries, states, and districts: the BenchRelSurv-and BenchRelSurvPlot-macros. *BMC public health* 13, 1 (2013), 34.
[18] Michael S Lawrence, Petar Stojanov, Craig H Mermel, James T Robinson, Levi A Garraway, Todd R Golub, Matthew Meyerson, Stacey B Gabriel, Eric S Lander, and Gad Getz. 2014. Discovery and saturation analysis of cancer genes across 21 tumour types. *Nature* 505, 7484 (2014), 495.
[19] Michael S Lawrence, Petar Stojanov, Paz Polak, Gregory V Kryukov, Kristian Cibulskis, Andrey Sivachenko, Scott L Carter, Chip Stewart, Craig H Mermel, Steven A Roberts, et al. 2013. Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature* 499, 7457 (2013), 214.
[20] Daniel D Lee and H Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 6755 (1999), 788–791.
[21] Fang Liu, Licheng Jiao, and Xu Tang. 2019. Task-oriented GAN for PolSAR image classification and clustering. *IEEE transactions on neural networks and learning systems* 30, 9 (2019), 2707–2719.
[22] Stefano Monti, Pablo Tamayo, Jill Mesirov, and Todd Golub. 2003. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine learning* 52, 1-2 (2003), 91–118.
[23] Sudipto Mukherjee, Himanshu Asnani, Eugene Lin, and Sreeram Kannan. 2019. Clustergan: Latent space clustering in generative adversarial networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 4610–4617.
[24] The Cancer Genome Atlas Research Network. 2011. Integrated genomic analyses of ovarian carcinoma. *Nature* 474 (29 06 2011), 609 – 615. http://dx.doi.org/10.1038/nature10166
[25] The Cancer Genome Atlas Research Network. 2013. Integrated genomic characterization of endometrial carcinoma. *Nature* 497 (01 05 2013), 67 – 73. http://dx.doi.org/10.1038/nature12113
[26] José Pereira. 2014. Survival Analysis Employed in Predicting Corporate Failure: A Forecasting Model Proposal. 7 (04 2014).
[27] Catherine R Planey and Olivier Gevaert. 2016. CoINcIDE: A framework for discovery of patient subtypes across multiple datasets. *Genome medicine* 8, 1 (2016), 27.
[28] Jost Tobias Springenberg. 2015. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390* (2015).
[29] Michael Steinbach, Levent Ertöz, and Vipin Kumar. 2004. The challenges of clustering high dimensional data. In *New directions in statistical physics*. Springer, 273–309.
[30] Mark Stevenson and IVABS EpiCentre. 2009. An introduction to survival analysis. *EpiCentre, IVABS, Massey University* (2009).
[31] Mike Stoolmiller and James Snyder. 2013. Embedding multilevel survival analysis of dyadic social interaction in structural equation models: hazard rates as both outcomes and predictors. *Journal of pediatric psychology* 39, 2 (2013), 222–232.
[32] Kelly C Vranas, Jeffrey K Jopling, Timothy E Sweeney, Meghan C Ramsey, Arnold S Milstein, Christopher G Slatore, Gabriel J Escobar, and Vincent X Liu. 2017. Identifying Distinct Subgroups of Intensive Care Unit Patients: a Machine Learning Approach. *Critical care medicine* 45, 10 (2017), 1607.