A Tale of Two Time Series Methods: Representation Learning for Improved Distance and Risk Metrics

by

Divya Shanmugam

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

Author .....
Department of Electrical Engineering and Computer Science
May 25, 2018

Certified by .....
John Guttag
Professor
Thesis Supervisor

Accepted by .....
Katrina LaCurts
Chair, Master of Engineering Thesis Committee

A Tale of Two Time Series Methods: Representation Learning for Improved Distance and Risk Metrics

by

Divya Shanmugam

Submitted to the Department of Electrical Engineering and Computer Science
on May 25, 2018, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Computer Science and Engineering

Abstract

In this thesis, we present methods in representation learning for time series in two areas: metric learning and risk stratification. We focus on metric learning due to the importance of computing distances between examples in learning algorithms and present Jiffy, a simple and scalable distance metric learning method for multivariate time series. Our approach is to reframe the task as a representation learning problem — rather than design an elaborate distance function, we use a CNN to learn an embedding such that the Euclidean distance is effective. Experiments on a diverse set of multivariate time series datasets show that our approach consistently outperforms existing methods. We then focus on risk stratification because of its clinical importance in identifying patients at high risk for an adverse outcome. We use segments of a patient's ECG signal to predict that patient's risk of cardiovascular death within 90 days. In contrast to other work, we work directly with the raw ECG signal to learn a representation with predictive power. Our method produces a risk metric for cardiovascular death with state-of-the-art performance when compared to methods that rely on expert-designed representations.

Thesis Supervisor: John Guttag
Title: Professor

Acknowledgments

It was the best of time series, and it was the worst of time series - but mostly the best because of my friends, family and advisors.

I'd first like to thank John for his support this year. He weathered through many of my panicked moments this year and that alone deserves commendation. A single year of John's advisorship has improved my writing, research and balance. I'm excited to see what the coming years bring!

I'm also thankful for Davis's help. He routinely served as a sounding board for my loosely organized ideas and helped me crystallize them into something worth reading.

And thank you to my undergraduate research advisor, Bonnie Berger. Her enthusiasm for my career in academia undermined any doubts I had at the time and pushed me to succeed this year.

I'm grateful for my friends, for their ability to be both ambitious and lighthearted. Nikita's support qualifies her as a silent co-author on everything I produce. I wrote most of this thesis running on Nikita-provided calories and coffee.

I'd also like to thank my labmates - Harini, Jen, Guha, Joel, Amy, Jose, Maggie, Katie, Adrian, Wayne, Akhil, and Sheila. Thank you for welcoming me into the lab and many more thank yous for reading my writing pre-deadlines.

And thank you to my family! Our group chat makes me laugh enough that everyone around me has to hear about it.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The demonstrated success of machine learning models across disciplines is heavily dependent on data representation. A great deal of work has gone into feature-engineering, where data representation is hand-optimized for a given task [56]. This is time-consuming, dependent on expert knowledge, and often, sub-optimal [7]. By instead *learning* a representation, we identify the most relevant features for a given task in the absence of expert knowledge. The success of representation learning methods over hand-crafted features has been demonstrated across several areas including computer vision [45], natural language processing [72], and audio analysis [52].

We use principles of representation learning to contribute to two areas of research: metric learning for multivariate time series and multiple instance learning for risk stratification. We motivate each area below and deliver:

1. A distance metric learning method for multivariate time series that achieves state-of-the-art performance on the nearest neighbor classification task across six domains.
2. A multiple instance learning method that outperforms existing approaches in identifying patients at high risk for cardiovascular death within 90 days of hospital admission.

1.1 Metric Learning for Time Series

In the first section of the thesis, we explore the application of representation learning to metric learning for multivariate time series. Metric learning aims to learn a distance function between examples. Multiple machine learning tasks rest on the notion of distance between examples. In many domains, Euclidean distance is an effective way to measure similarity. Sequential data, however, elude a straightforward definition of similarity because of the necessity of alignment between examples. Time series similarity is also dependent on the task at hand, further complicating the matter.

Numerous methods have been developed to address measuring distance between time series. Current work falls into one of two broad categories: hand-designed approaches and metric learning methods. We approach the problem from a representation learning perspective and show that measuring similarity using Euclidean distance in a learned, embedded space outperforms existing methods. Ultimately, we present Jiffy, a simple, scalable, task-dependent distance metric for multivarate time series. Experiments demonstrate the strength of the method across six domains on the nearest-neighbor classification task.

1.2 Multiple Instance Learning for Risk Stratification

In the second section of this thesis, we demonstrate the success of representation learning in the context of patient risk stratification. We aim to identify patients at high risk for cardiovascular death (CVD) within 90 days of hospital admission. Existing approaches to this task rely on expert-designed representations of a patient's ECG signal to estimate this risk. In contrast to existing metrics, we operate on the raw ECG signal to learn a representation with predictive power. We accomplish this by reframing the problem as a multiple instance learning task.

Multiple instance learning (MIL) is a form of weakly supervised learning in which labels for collections of examples are provided, but labels for examples are not. In our

work, we treat each patient as a collection of adjacent beat pairs (ABPs). Although we know whether or not a patient has died within 90 days, we do not know what each ABP represents. In other words, we have no intuition as to which ABPs are indicative of CVD risk and which are not. We present the resulting representation-based method and demonstrate its success compared to existing ECG-based risk metrics.

Chapter 2

Metric Learning for Multivariate Time Series

Measuring distances between examples is a fundamental component of many classification, clustering, segmentation and anomaly detection algorithms for time series [57, 63, 6, 22]. Because the distance measure used can have a significant effect on the quality of the results, there has been a great deal of work developing effective time series distance measures [29, 39, 3, 6, 25]. Historically, most of these measures have been hand-crafted. However, recent work has shown that a learning approach can often perform better than traditional techniques [26, 48, 16].

We introduce a metric learning model for multivariate time series. Specifically, by learning to embed time series in Euclidean space, we obtain a metric that is both highly effective and simple to implement using modern machine learning libraries. Unlike many other deep metric learning approaches for time series, we use a convolutional, rather than a recurrent, neural network, to construct the embedding. This choice, in combination with aggressive maxpooling and downsampling, results in a compact, accurate network.

Using a convolutional neural network for metric learning *per se* is not a novel idea [53, 65]; however, time series present a set of challenges not seen in other domains, and how best to embed them is far from obvious. In particular, time series suffer from:

1. *A lack of labeled data.* Unlike text or images, time series cannot typically be

annotated post-hoc by humans. This has given rise to efforts at unsupervised labeling [9], and is evidenced by the small size of most labeled time series datasets. Of the 85 datasets in the UCR archive [17], for example, the largest dataset has fewer than 17000 examples, and many have only a few hundred. Weakly annotated datasets, where each signal carries a label but its components do not, are more common. We discuss methods for this modality of time series data in the next chapter.

2. *A lack of large corpora.* In addition to the difficulty of obtaining labels, most researchers have no means of gathering even *unlabeled* time series at the same scale as images, videos, or text. Even the largest time series corpora, such as those on Physiobank [30], are tiny compared to the virtually limitless text, image, and video data available on the web.
3. *Extraneous data.* There is no guarantee that the beginning and end of a time series correspond to the beginning and end of any meaningful phenomenon. I.e., examples of the class or pattern of interest may take place in only a small interval within a much longer time series. The rest of the time series may be noise or transient phenomena between meaningful events [58, 33].
4. *Need for high speed.* One consequence of the presence of extraneous data is that many time series algorithms compute distances using every window of data within a time series [50, 9, 58]. A time series of length T has $O(T)$ windows of a given length, so it is essential that the operations done at each window be efficient.

As a result of these challenges, an effective time series distance metric must exhibit the following properties:

- Efficiency: Distance measurement must be fast, in terms of both training time and inference time.
- Simplicity: As evidenced by the continued dominance of the Dynamic Time Warping (DTW) distance [62] in the presence of more accurate but more complicated rivals, a distance measure must be simple to understand and implement.

- Accuracy: Given a labeled dataset, the metric should yield a smaller distance between similarly labeled time series. This behavior should hold even for small training sets.

Our primary contribution is a time series metric learning method, *Jiffy*, that exhibits all of these properties: it is fast at both training and inference time, simple to understand and implement, and consistently outperforms existing methods across a variety of datasets.

We introduce the problem statement and the requisite definitions in Section 2. We summarize existing state-of-the-art approaches (both neural and non-neural) in Section 3 and go on to detail our own approach in Section 4. We then present our results in Section 5. The paper concludes with implications of our work and avenues for further research.

2.1 Problem Definition

We first define relevant terms, frame the problem, and state our assumptions.

Definition 2.1.1. ***Time Series*** *A D-variable time series X of length T is a sequence of real-valued vectors $\mathbf{x}_1, \ldots, \mathbf{x}_T, \mathbf{x}_i \in \mathbb{R}^D$. If $D = 1$, we call X "univariate", and if $D > 1$, we call X "multivariate." We denote the space of possible D-variable time series as $\mathcal{T}^D$.*

Definition 2.1.2. ***Distance Metric*** *A distance metric is a distance function $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ over a set of objects $\mathcal{S}$ such that, for any $x, y \in \mathcal{S}$, the following properties hold:*

- *Symmetry:* $d(x, y) = d(y, x)$
- *Non-negativity:* $d(x, y) \geq 0$
- *Triangle Inequality:* $d(x, z) + d(y, z) \geq d(x, z)$
- *Identity of Indiscernibles:* $x = y \Leftrightarrow d(x, y) = 0$

Our approach to learning a metric is to first learn an embedding into a fixed-size vector space, and then use the Euclidean distance on the embedded vectors to measure similarity. Formally, we learn a function $f : \mathcal{T}^D \to \mathbb{R}^N$ and compute the distance between time series $X, Y \in \mathcal{T}^D$ as:

$$d(X, Y) \triangleq \|f(X) - f(Y)\|_2 \tag{2.1}$$

The choice of Euclidean distance in the embedded space is arbitrary - we choose Euclidean distance for its prevalence across literature.

2.1.1 Assumptions

Jiffy depends on two assumptions about the time series being embedded. First, we assume that the time series we deal with represent a single class - in other words, the majority of each input time series is representative of its class. This means that we do not consider multi-label tasks or tasks wherein only a small subsequence within each time series is associated with a particular label, while the rest is noise or phenomena for which we have no class label. This assumption is implicitly made by most existing work [35] and is satisfied whenever one has recordings of individual phenomena, such as gestures, heartbeats, or actions.

The second assumption is that the time series dataset is not too small, in terms of either number of time series or their lengths. Specifically, we do not consider datasets in which the longest time series is of length $T < 40$ or the number of examples per class is less than 25. The former number is the smallest number such that our embedding will not be longer than the input in the univariate case, while the latter is the smallest number found in any of our experimental datasets (and therefore the smallest on which we can claim reasonable performance).

For datasets too small to satisfy these constraints, we recommend using a traditional distance measure, such as Dynamic Time Warping, that does not rely on a learning phase.

2.2 Related Work

2.2.1 Hand-Crafted Distance Measures

Historically, most work on distance measures between time series has consisted of hand-crafted algorithms designed to reflect prior knowledge about the nature of time series. By far the most prevalent is the Dynamic Time Warping (DTW) distance [62]. This is obtained by first aligning two time series using dynamic programming, and then computing the Euclidean distance between them. DTW requires time quadratic in the time series' length in the worst case, but is effectively linear time when used for similarity search; this is thanks to numerous lower bounds that allow early abandoning of the computation in almost all cases [57].

Other handcrafted measures include the Uniform Scaling Distance [36], the Scaled Warped Matching Distance [28], the Complexity-Invariant Distance [4], the Shotgun Distance [63], and many variants of DTW, such as weighted DTW [29], DTW-A [68], and global alignment kernels [21]. However, nearly all of these measures are defined only for univariate time series, and generalizing them to multivariate time series is not trivial [68].

2.2.2 Hand-Crafted Representations

In addition to hand-crafted functions of raw time series, there are numerous hand-crafted representations of time series. Perhaps the most common are Symbolic Aggregate Approximation (SAX) [46] and its derivatives [13, 66]. These are discretization techniques that low-pass filter, downsample, and quantize the time series so that they can be treated as strings. Slightly less lossy are Adaptive Piecewise Constant Approximation [38], Piecewise Aggregate Approximation [37], and related methods, which approximate time series as sequences of low-order polynomials.

The most effective of these representations tend to be extremely complicated; the current state-of-the-art [64], for example, entails windowing, Fourier transformation, quantization, bigram extraction, and ANOVA F-tests, among other steps. Moreover,

it is not obvious how to generalize them to multivariate time series.

2.2.3 Metric Learning for Time Series

A promising alternative to hand-crafted representations and distance functions for time series is metric learning. This can take the form of either learning a distance function directly or learning a representation that can be used with an existing distance function.

Among the most well-known methods in the former category is that of [61], which uses an iterative search to learn data-dependent constraints on DTW alignments. More recently, [48] use a learned Mahalanobis distance to improve the accuracy of DTW. Both of these approaches yield only a pseudometric, which does not obey the triangle inequality. To come closer to a true metric, [16] combined a large-margin classification objective with a sampling step (even at test time) to create a DTW-like distance that obeys the triangle inequality with high probability as the sample size increases.

In the second category are various works that learn to embed time series into Euclidean space. [54] use recurrent neural networks in a Siamese architecture [12] to learn an embedding; they optimize the embeddings to have positive inner products for time series of the same class but negative inner products for those of different classes. A similar approach that does not require class labels is that of [2]. This method trains a Siamese, single-layer CNN to embed time series in a space such that the pairwise Euclidean distances approximate the pairwise DTW distances. [43] optimize a similar objective, but does so by sampling the pairwise distances and using matrix factorization to directly construct feature representations for the training set (i.e., with no model that could be applied to a separate test set).

These methods seek to solve much the same problem as Jiffy, but as we show experimentally, produce metrics of lower quality.

2.3 Method

We learn a metric by learning to embed time series into a vector space and comparing the resulting vectors with the Euclidean distance. Our embedding function is takes the form of a convolutional neural network, shown in Figure 2-1. The architecture rests on three basic layers: a convolutional layer, maxpooling layer, and a fully connected layer.

The convolutional layer is included to learn the appropriate subsequences from the input. The network employs one-dimensional filters convolved over all time steps, in contrast to the traditional two-dimensional filters used with images. We opt for one-dimensional filters because time series data is characterized by infrequent sampling. Convolving over each of the variables at a given timestep has little intuitive meaning in developing an embedding when each step measurement has no coherent connection to time. For discussion regarding the mathematical connection between a learned convolutional filter and traditional subsequence-based analysis of time series, we direct the reader to [20].

The maxpooling layer allows the network to be resilient to translational noise in the input time series. Unlike most existing neural network architectures, the windows over which we max pool are defined as percentages of the input length, not as constants. By aggressively pooling, we are able to heavily downsample and denoise the input signal. The output of the pooling layer is fed into the final fully connected layer.

We downsample heavily after the filters are applied such that each time series is reduced to a fixed size. We do so primarily for efficiency—further discussion on parameter choice for Jiffy may be found in Section 2.5.

We then train the network by appending a softmax layer and using cross-entropy loss with the ADAM [40] optimizer. We experimented with more traditional metric learning loss functions, rather than a classification objective, but found that they made little or no difference while adding to the complexity of the training procedure; specific loss functions tested include several variations of Siamese networks [12, 54] and the triplet loss [34].

2.3.1 Complexity analysis

Let T be the length of the D-variable time series being embedded, let F be the number of length K filters used in the convolutional layer, and Let L be the size of the final embedding.

The time to apply the convolution and ReLU operations is $\Theta(TDFK)$. Following the convolutional layer, the maxpooling and downsampling require $\Theta(T2DF)$ time if implemented naively, but $\Theta(TDF)$ if an intelligent sliding max function is used, such as that of [44]. Finally, the fully connected layer, which constitutes the embedding, requires $\Theta(TDFL)$ time.

The total time to generate the embedding is therefore $\Theta(TDF(K+L))$. Given the embeddings, computing the distance between two time series requires $\Theta(L)$ time. Note that T no longer appears in the latter expression thanks to the max pooling.

With $F = 16$, $K = 5$, $L = 40$, this computation is dominated by the fully connected layer. Consequently, when $L \ll T$ and embeddings can be generated ahead of time, which enables a significant speedup compared to operating on the original data. Such a situation would arise, e.g., when performing a similarity search between a new query and a fixed or slow-changing database [10]. When both embeddings must be computed on-the-fly, our method is likely to be slower than DTW and other traditional approaches.

2.4 Experiments

Before describing our experiments, we first note that, to ensure easy reproduction and extension of our work, all of our code is freely available.[1] All of the datasets used are public, and we provide code to clean and operate on them.

[1] http://smarturl.it/jiffy

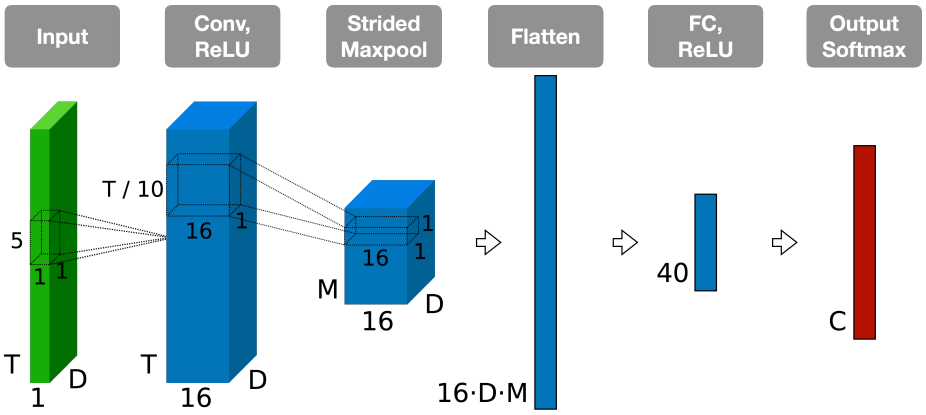

Figure 2-1: Architecture of the proposed model. A single convolutional layer extracts local features from the input, which a strided maxpool layer reduces to a fixed-size vector. A fully connected layer with ReLU activation carries out further, nonlinear dimensionality reduction to yield the embedding. A softmax layer is added at training time.

2.4.1 Experimental Setup

We evaluate Jiffy-produced embeddings through the task of 1-nearest-neighbor classification, which assesses the extent to which time series sharing the same label tend to be nearby in the embedded space. We choose this task because it is the most widely used benchmark for time series distance and similarity measures [25, 3].

To enable direct comparison to existing methods, we benchmark Jiffy using datasets employed by [48]. These datasets are taken from various domains and exhibit high variability in the numbers of classes, examples, and variables. We briefly describe each dataset below, and summarize statistics about each in Table 2.1.

Table 2.1: Summary of Multivariate Time Series Datasets.

Dataset	# Variables	# Classes	Length	# Time Series
Libras	2	15	45	360
AUSLAN	22	25	47-95	675
CharacterTrajectories	3	20	109-205	2858
ArabicDigits	13	10	4 - 93	8800
ECG	2	2	39 - 152	200
Wafer	6	2	104 - 198	1194

- **ECG**: Electrical recordings of normal and abnormal heartbeats, as measured by two electrodes on the patients' chests.
- **Wafer**: Sensor data collected during the manufacture of semiconductor microelectronics, where the time series are labeled as normal or abnormal.
- **AUSLAN**: Hand and finger positions during the performance of various signs in Australian Sign Language, measured via instrumented gloves.
- **Trajectories**: Recordings of pen (x,y) position and force application as different English characters are written with a pen.
- **Libras**: Hand and arm positions during the performance of various signs in Brazilian Sign Language, extracted from videos.
- **ArabicDigits**: Audio signals produced by utterances of Arabic digits, represented by Mel-Frequency Cepstral Coefficients.

2.4.2 Comparison Approaches

We compare to recent approaches to time series metric learning, as well as popular means of generalizing DTW to the multivariate case: we restrict our evaluation to approaches with published results on the tested datasets, source code provided by the authors, or sufficient simplicity that we could confidently reimplement them.

1. **MDDTW** [48] - MDDTW compares time series using a combination of DTW and the Mahalanobis distance. It learns the precision matrix for the latter using a triplet loss.
2. **Siamese RNN** [54] - The Siamese RNN feeds each time series through a recurrent neural network and uses the hidden unit activations as the embedding. It trains by feeding pairs of time series through two copies of the network and computing errors based on their inner products in the embedded space.
3. **Siamese CNN** The Siamese CNN is similar to the Siamese RNN, but uses convolutional, rather than recurrent, neural networks. This approach has proven successful across several computer vision tasks [12, 70].

4. **DTW-I**, **DTW-D** - As pointed out by [68], there are two straightforward ways to generalize DTW to multivariate time series. The first is to treat the time series as D independent sequences of scalars (DTW-I). In this case, one computes the DTW distance for each sequence separately, then sums the results. The second option is to treat the time series as one sequence of vectors (DTW-D). In this case, one runs DTW a single time, with elementwise distances equal to the squared Euclidean distances between the D-dimensional elements.
5. **Zero Padding** - One means of obtaining a fixed-size vector representation of a multivariate time series is to zero-pad such that all time series are the same length, and then treat the "flattened" representation as a vector.
6. **Upsampling** - Like Zero Padding, but upsamples to the length of the longest time series rather than appending zeros. This approach is known to be effective for univariate time series [60].

2.4.3 Evaluation

As shown in Table 2.2, we match or exceed the performance of all comparison methods on each of the six datasets. Although it is not possible to claim statistical significance in the absence of more datasets (see [23]), the average rank of our method compared to others is higher than its closest competitors at 1.16. The closest second, DTW-I, has an average rank of 3.33 over these six datasets.

Not only does Jiffy attain higher classification accuracies than competing methods, but the method also remains consistent in its performance across datasets. This can most easily be seen through the standard deviation in classification accuracies across datasets for each method. Jiffy's standard deviation in accuracy (0.026) is approximately a third of DTWI's (0.071). The closest method in terms of variance is MDDTW with a standard deviation of 0.042 , but MDDTW exhibits a much lower rank than our method. This consistency suggests that Jiffy generalizes well across domains, and would likely remain effective on other datasets not tested here.

Table 2.2: 1NN Classification Accuracy. The proposed method equals or exceeds the accuracies of all others on every dataset.

Dataset	Jiffy	MDDTW	DTW-D	DTW-I	Siamese CNN	Siamese RNN	Zero Pad	Upsample
ArabicDigits	**0.974**	0.969	0.963	**0.974**	0.851	0.375	0.967	0.898
AUSLAN	**1.000**	0.959	0.900	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
ECG	**0.925**	0.865	0.825	0.810	0.756	0.659	0.820	0.820
Libras	**1.000**	0.908	0.905	0.979	0.280	0.320	0.534	0.534
Trajectories	**0.979**	0.961	0.956	0.972	0.933	0.816	0.936	0.948
Wafer	**0.992**	0.988	0.984	0.861	0.968	0.954	0.945	0.936
Mean Rank	**1.67**	3.67	4.67	3.33	6.0	6.5	4.17	4.5

2.5 Method Parameter Effects

A natural question when considering the performance of a neural network is whether, or to what extent, the hyperparameters must be modified to achieve good performance on a new dataset. In this section, we explore the robustness of our approach with respect to the values of the two key parameters: embedding size and pooling percentage. We do this by learning metrics for a variety of parameter values for ten data sets from the UCR Time Series Archive [17], and evaluating how classification accuracy varies.

2.5.1 Embedding Size

Figure 2-2.*left* shows that even a few dozen neurons are sufficient to achieve peak accuracy. As a result, an embedding layer of 40 neurons is sufficient and leads to an architecture that is compact enough to train on a personal laptop.

2.5.2 Pooling percentage

The typical assumption in machine learning literature is that max pooling windows in convolutional architectures should be small to limit information loss. In contrast, time series algorithms often max pool globally across each example (e.g. [32]). Contrary to the implicit assumptions of both, we find that the level of pooling that results in the highest classification often falls in the 10-25% range, as shown by Figure 2-2.*right*

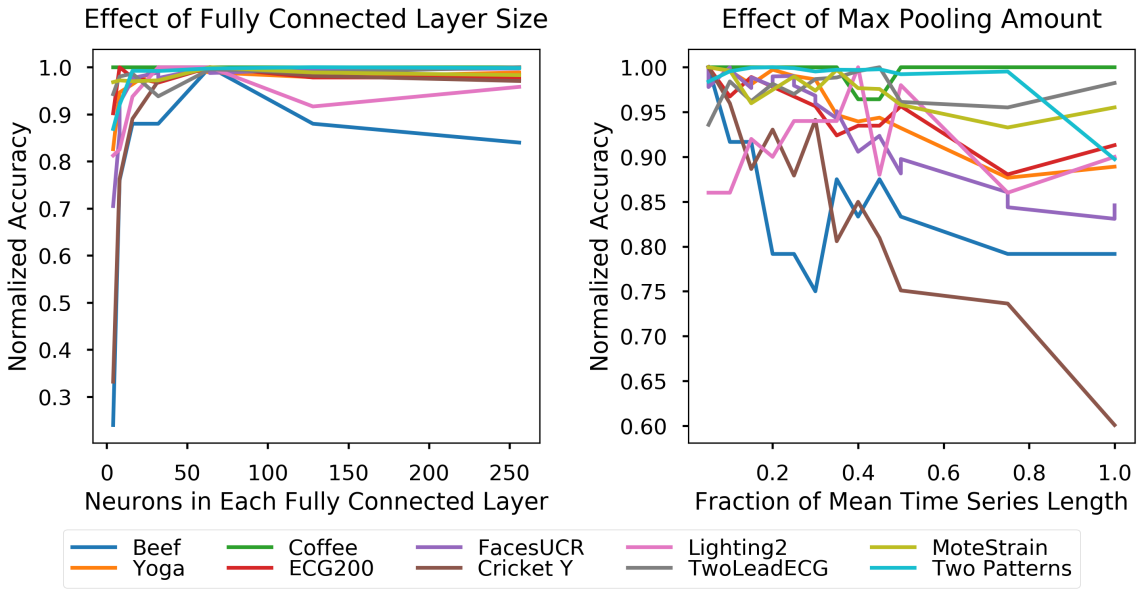

Figure 2-2: Effect of fully connected layer size and degree of max pooling on model accuracy using held-out datasets. Even small fully connected layers and large amounts of max pooling—up to half of the length of the time series in some cases—have little or no effect on accuracy. For ease of visualization, each dataset's accuracies are scaled such that the largest value is 1.0.

Chapter 3

Multiple Instance Learning for Risk Stratification

Supervise machine learning models rely on the availability of labeled data. Multiple instance learning (MIL) research relaxes this assumption by assuming that labels for collections of examples are available, but labels for each example are not. This is applicable in a number of scenarios, including drug discovery [24], image analysis [18], and document classification [73].

In this chapter, we introduce the multiple instance learning paradigm to risk stratification. Risk stratification models aim to identify patients at high risk for a given outcome so that doctors may intervene, with the attempt of avoiding that outcome. Machine learning has led to improved risk stratification models for a number of outcomes, including stroke [67], cancer [41] and treatment resistance [55]. To the best of our knowledge, this is the first application of multiple instance learning to risk stratification.

We treat each patient as a labeled collection of unlabeled medical attributes. The patient labels correspond to adverse outcomes and the unlabeled medical attributes correspond to potential risk indicators. This maps cleanly to the MIL framework in how we are able to differentiate patients based on outcome, but have no further intuition as to what medical factors contribute to this difference. In this chapter, we focus on predicting cardiovascular death (CVD) within 90 days of hospital admission

from a patient's ECG signal.

Existing CVD risk metrics rest on the underlying hypothesis that a patient's electrocardiogram (ECG) signal is crucial to identify patients at high risk for a cardiac-related adverse outcome. In particular, literature suggests that the relationship between adjacent beats can indicate risk for cardiovascular complications [14]. CVD risk metrics including morphological variability (MV)[69] and morphological variability in beat-space (MVB) [47], use this principle and choose to transform the ECG signal in domain-specific ways.

In contrast to these existing approaches, we learn a transformation of the signal using a compact neural network. We treat each patient as a collection of adjacent beat pairs (ABPs) and aim learn the relationship between adjacent beats.

This is challenging because we do not have access labels at the ABP level, but rather at the patient level. Different classes at the patient level do not necessarily correspond to clear class differences between constituent ABPs. In other words, it is unlikely that every ABP, or even most, in a high risk patient is distinguishable from every ABP in a low risk patient. Instead, differing proportions of latent classes of ABPs may be more clear differentiators of a high risk patient from a low risk one.

We address this issue using the multiple instance learning framework. Each patient is a collection of instances, and each ABP is an instance. In our training procedure, we assume every ABP inherits the label of the patient it came from. Despite the fact that certain ABPs are trained on $soft\ labels$, or labels that do not necessarily correspond to their class, we achieve a higher performance on the task of predicting 90-day CVD on a held-out test set than existing methods.

In this chapter, we present a novel application of a simple MIL model to cardiovascular death risk stratification, evaluate its performance and explore its guarantees. Section 3.1 formalizes our problem statement. Section 3.2 provides a brief overview of related work and Section 3.3 details our method. Section 3.4 presents our results applied to ECG data and Section 6 demonstrates performance in simulated settings. Section 3.5 describes theory about method performance in adverse conditions.

3.1 Problem Definition

In traditional supervised learning, we are given a single collection of instances $\{\mathbf{x}^i\}_{i=1}^M$ with associated labels $\{y_i\}_{i=1}^M$. In Multi-Instance Learning, these individual instances are replaced with *bags* of examples $\{B^i\}_{i=1}^M$. Each bag B^i is a set of N_i instances $\{\mathbf{x}^{ij}\}_{j=1}^{N_i}$ and is associated with a single label y_i. Each $\mathbf{x}^{ij}$ is drawn independently conditioned on y_i.

A common modeling assumption is that each instance $\mathbf{x}_{ij}$ is associated with a latent class label c_{ij}, drawn from the set of possible bag labels $\mathcal{Y}$. The set of (estimated) labels is then used to classify the bag, typically using a voting scheme.

In the binary classification case, instances of one of the two classes are termed *witnesses* [15]. Without loss of generality, this is taken to be the positive class. The previous modeling assumption then reduces to claiming that the concentration of witnesses (the *witness rate*) is higher for the positive class. We will refer to the positive bag witness rate as p_1 and the negative bag witness rate as p_2.

The task then is to construct two functions: $F : \mathcal{X} \to [0, 1]$ and $G : \mathbb{R}^{N_i} \to \mathbb{R}^+$. F returns larger values for instances that are more likely to be witnesses, and is often a probabilistic binary classifer. G is an aggregation function that returns a score for a bag that is larger for bags that are more likely to belong to the positive class. Based on both prior work [27] and our analysis below (c.f. Section 3.6), we take G to be the mean, $\frac{1}{N_i}\sum_{j=1}^{N_i} F(x_{ij})$ for bag B_i.

3.2 Related Work

Approaches to predicting patient risk from ECG signals fall into two major categories: hand-crafted and learned.

3.2.1 Hand-Crafted CVD Risk Metrics

Common unlearned risk metrics include the TIMI Risk Score (TRS) [1], Deceleration Capacity (DC) [5], Heart Rate Variability (HRV) [14], and morphological variability

(MV) [69]. TRS is a point-based system for indicating risk based on categorical variables [1]. Risk factors that merit a point include being older than 65 and having taken Aspirin within the past week. A patient with a TRS score greater than 4 is considered to be at high risk for CVD. DC is a metric based on the deceleration and acceleration of a patient's heart rate, operating on the theory that slower heart rate deceleration indicates a higher risk of death. HRV measures the variability in time between successive heartbeats. MV measures the DTW alignment cost between adjacent beats and averages this cost over the first 24 hours of a patient's ECG signal. This averaged alignment cost is the patient's risk score and is then used to stratify patients into high risk and low risk groups. Patients landing in the upper quartile of the MV metric are considered high risk.

3.2.2 Learned CVD Risk Metrics

Recent work in learned approaches to CVD risk stratification include morphological variability in beat-space (MVB) [47] and an approach that combines an RNN and a logistic regression (LR-RNN) [51]. HRV, MV, and MVB each operate on the principle that beat-to-beat variation is indicative of cardiovascular risk. This finding is supported in multiple studies [42] [8]. Heart Rate Variability measures the variance in heart rate over the course of a patient's signal and uses a logistic regression. MVB improves upon MV by learning the optimal frequency band over which to average DTW variability. The LR-RNN approach combines the output of an RNN on a particular segment of the signal with a logistic regression model over seven patient features, including HRV.

Each of these approaches leverages expert information to produce features indicative of CVD risk. MV and MVB approximate variability using the DTW alignment cost, while LR-RNN calculates the Lagrange coefficients of a set of annotated ECG signal segments. Our work assumes nothing about the relationship between adjacent beats and instead learns a representation for an adjacent beat pair that discriminates high risk patients from low risk ones.

3.3 Method

Our method aims to distinguish bags from one another by building an instance classifier trained on bag labels. This mirrors the single instance learning approach proposed by Frank and Xu [27]. A recent survey comparing multiple instance learning methods demonstrated this approach's competitive performance under the standard MI assumption, where $p_1 > 0$ and $p_2 = 0$ [15]. One can view the problem posed here regarding relative witness rates as a generalization of this, where negative bags typically have relatively *fewer* witnesses than positive bags ($p_1 > p_2$).

At the crux of our method is an instance classifier to map instances to a predicted bag label. In this approach, we are intentionally training non-witness instances on a false label. To correct for this error, we average the bag predictions over all instances. We aim to learn a instance-based score that separates one bag from the others. In this scenario, this score is the likelihood of the class label. We focus on scenarios with two bag types, but this approach extends naturally to the separation of one bag class from many.

Assume each bag B_i, with binary label $y_i \in \{0, 1\}$, consists of n instances $X = \{x_{i1}, x_{i2}, ...x_{in}\}$. Instance x_{ij} is a j^{th} member of bag B_i We train a neural network to approximate the function mapping instance x_{ij} to bag label y_i. We arrive at the label of a new bag by averaging the labels contributed by each instance within the bag. Functionally, this means that each instance contributes equally to its bag's label.

We use a shallow neural network to learn the classifier. The network consists of one fully connected layer with two ReLU-activated neurons and a sigmoid-activated output. We choose two neurons to learn a compact ABP representation. Similar to our work in the previous work, we provide the code for the method and simulated experiments available in the interest of reproducibility.

3.4 Experiments

In this section, we present the results of applying ourmethod to the 90-day CVD risk stratification task. We first describe the experimental setup and evaluation metrics and follow with a discussion of performance. We show that a formulation of the problem under the multiple instance learning framework outperforms existing risk metrics.

3.4.1 Experimental Setup

We use 5396 patients from the MERLIN-TIMI dataset [49]. Our goal is to predict a patient's risk of cardiovascular death within 90 days of admission (90-Day CVD risk). Of the 5396 patients in this dataset, 107 patients satisfy this criteria. The incidence rate of 90-day CVD in this population is 1.9%.

We perform the pre-processing steps of competing methods to ensure that differences in performance are a result of the method itself. This includes the removal of baseline wander, high frequency noise, and ectopic beats using the Physionet SQI package [31]. Each patient's ECG signal is sampled at a rate of 128Hz. Each patient is represented by the first 1000 adjacent beat pairs of their ECG signal.

During our training procedure, we enforce a class balance between positive and negative bags. This is important because the witness instance class is shared between bags. A balanced training set of bags guides the network towards most discriminate instance features between the two. We trained on 160 patients and tested on 600. The training set consisted of 80 90-day CVD patients and 80 non-CVD patients. The test set was sampled randomly from the pool of 5216 patients outside of the training set and consists of 12 90-day CVD patients and 588 non-CVD patients. This test bag label distribution was chosen to mirror the incidence rate of CVD within 90 days of admission.

3.4.2 Evaluation Metrics

We evaluate our model along two metrics: the area under the Receiver Operating Curve (AUC) and the Hazard Ratio (HR). The AUC measures how well our algorithm ranks patients in terms of the binary label of cardiovascular death within 90 days of hospital admission. The HR is calculated using the Cox Proportional Hazards Model [19] and measures the dependency of an outcome over time on predictor variables. The hazard ratio reflects relative risk of being in a high risk group versus the low risk group and is a common measure of performance for risk stratification models [71].

3.4.3 Evaluation

Hazard Ratio We rank patients based on our risk score and calculate the HR of a patient in the upper quartile. We choose the upper quartile in accordance with existing literature. Looking at Figure 3-1, we see the Kaplan-Meier mortality curve of the risk metric learned by our algorithm. If a patient's risk score lands in the top quartile of our risk score, the patient is approximately 17 times more likely to die of cardiovascular death than low risk patients. This is drastically more informative than existing metrics. A detailed comparison of the hazard ratios across different time scales - including death within 60 days and 30 days - can be found in Table 3.1. It is worth noting that our metric excels in predicting adverse events at a longer time scale than competing methods, but lands in the middle of existing metrics in terms of 30-Day HR.

AUC Our method outperforms existing metrics in terms of AUC, suggesting an improved ability to rank patients in terms of likelihood of cardiovascular death. This suggests that our method calculates a score that has higher discriminative power between patients who die within 90 days of hospital admission and those who don't than existing approaches. Figure 3-2 demonstrates the AUC values on two tasks: 1) Averaged instance-level prediction to bag label (left) and 2) Instance prediction to bag label (right). Recall that each instance does not necessarily correspond to its bag label.

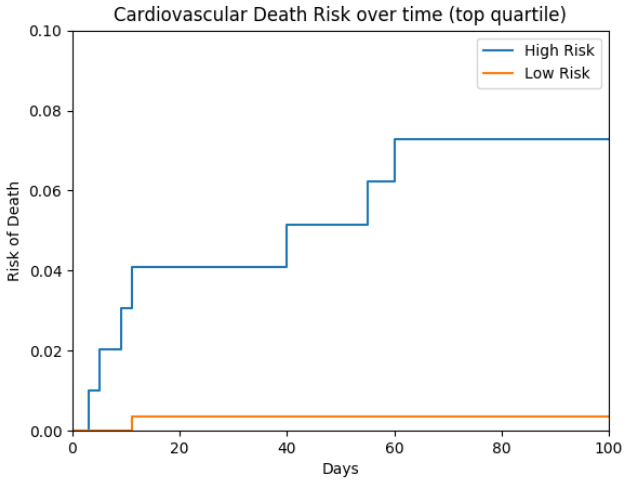

Figure 3-1: Kaplan-Meier curve illustrating the mortality rates in high risk cohort (blue) and the low risk cohort (orange) over time. Of the 12 patients who die within 90 days of hospital admission, 11 are identified by our model as high risk.

Table 3.1: AUC and Hazard Ratio results for existing ECG risk metrics and our model. Our model achieves a higher AUC than existing methods and exceeds the HR values at the 60 and 90 day time-scales. Empty entries in the table correspond to unavailable results.

Name	AUC	90-Day HR	60-Day HR	30-Day HR
Our Method	.81	17.38	17.37	9.55
MV	.72	8.45	–	12.30
MVB	.72	8.81	–	–
ANN	.743	4.94	4.94	5.26
TRS	.67	–	3.82	3.31

Thus, it is unsurprising that the AUC at the instance level is lower than the AUC averaged across all instances in a bag. Interestingly, however, there is a significant amount of discriminative power in the instances themselves. The relationship between the bag label from one instance results in an AUC of .78, while the relationship between the true bag label and the aggregation of instance predictions is .81. Table 3.1 summarizes AUCs achieved by existing methods.

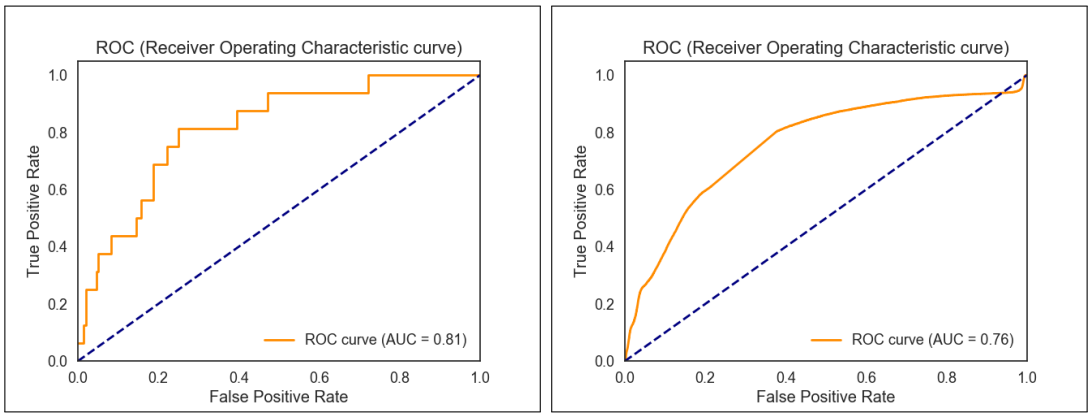

Figure 3-2: Visualization of AUC for two tasks: 1) prediction of patient outcome aggregated over a patient's ABPs (left) and 2) prediction of patient outcome from a single ABP (right). We include the second plot to demonstrate the cost of training instances on a soft label from the bag - as expected, the AUC of prediction based on an instance is lower than the AUC of prediction based on a set of instances.

3.5 Problem Parameter Effects

The ECG risk stratification task is a single instance of the broader problem posed in this chapter. We include simulations for method performance under different parameters to show two points: 1) the witness rate differences under which this method will or won't work and 2) broader relationships between the witness rate and bag size.

We focus on three parameters: the witness rate in the positive bags (p_1), the witness rate in the negative bags (p_2), and the number of samples per bag (N).

3.5.1 Experimental Setup

For these simulations, instances from the witness class are drawn from a 2D Gaussian centered on (1, 1). Non-witness instances are drawn from a 2D Gaussian centered on (1, -1). The identity matrix parametrizes the covariance for both distributions. Similar to the ECG setting, we train the model on 200 bags evenly split over the positive and negative bag classes. We measure the effects of these parameters in two ways: instance AUC (corresponding to the instance to bag label task) and bag AUC

(corresponding to the bag prediction to bag label task).

3.5.2 Relative Witness Rates

In this simulation, we explore the relationship between witness rates in the positive and negative bags. We measure the model's ability to distinguish bags from one another by its AUC - as in the ECG setting, we observe both the instance AUC and the bag AUC. Figure 3-3 visualizes our experiments in a heat map. The darker the square, the worse the AUC. We plot combinations of p_1 and p_2 such that $p_1 > p_2$. Each row represents model behavior as the negative bag witness rate is held constant and the positive bag witness rate is increased. Intuitively, when we fix p_2 and increase p_1, the model achieves higher AUCs. Interestingly, the model performance in terms of instance AUC and bag AUC is dependent on the *difference* between the witness rates rather than the magnitudes of the witness rates themselves. Previous work suggests that the application of SIL does not perform well in scenarios in which the witness rate is low [15]. These experiments also show that an increase in bag size - from 10 to 50 - results in a drastic improvement in bag-level AUC. This demonstrates that SIL performs well in this setting given a large enough bag size. Note that the higher rows of each graph corresponds more closely to the model's simulated performance under the standard MI assumption ($p_1 > 0, p_2 = 0$).

3.5.3 Bag Size

Since bag predictions are simply an aggregate of instance predictions, more instances in each bag results in a lower variance in bag prediction, results in a higher AUC. Our simulations demonstrate this fact and also demonstrate the model's ability to disentagle bags in which the witness rate differs by as little as .05. Figure 3-4 shows the AUC response to bag size in the context of 8 witness rate differences. With a witness rate difference of .35, we need only 10 samples per bag to achieve peak global accuracy.

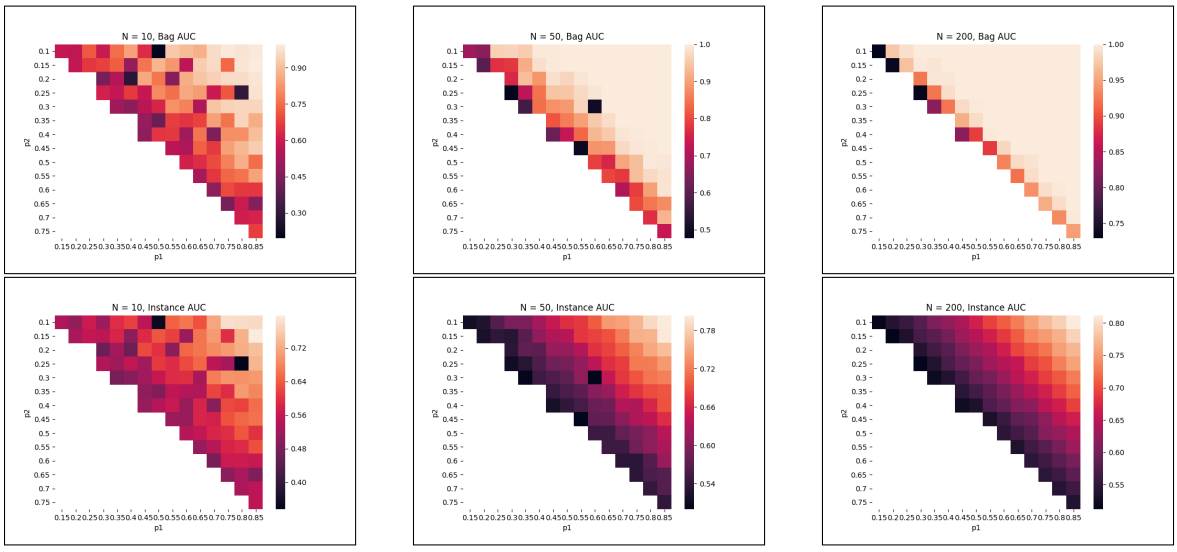

Figure 3-3: Plots in the top row represent the Bag AUC, which represents the model's correctness going from a set of instances to a bag label. Plots in the bottom row display the model's accuracy going from a single instance to a bag label. Each column represents experiments run with separate setting for the number of instances in each bag. The gradient at each square represents the AUC, where lighter squares represent a higher AUC than darker squares.

3.6 Theory

In addition to offering good simulation and empirical results, labeling of bags based on estimated instance labels has strong mathematical grounding. In particular, we show that:

1. There is (for sufficiently large bags) a closed-form expression for the distribution of predicted witnesses as a function of bag size, true number of witnesses, and instance classifier characteristics.
2. Assuming a Normal approximation to a Beta and/or Truncated Normal distribution, there is a closed-form expression for the distribution of AUC values given priors on the numbers of witnesses in bags from each class.

In what follows, let ω denote the positive class, and let $\hat{c}_{ij}$ denote the prediction of the instance classifier f. Further define $q_+ \triangleq E[\hat{c}_{ij}|c_{ij} = \omega]$ and $q_- \triangleq E[\hat{c}_{ij}|c_{ij} \neq \omega]$. Similarly let $\sigma_+ \triangleq Var[f|c_{ij} = \omega]$ and $\sigma_- \triangleq Var[f|c_{ij} \neq \omega]$. Finally $s_i(B) \triangleq$

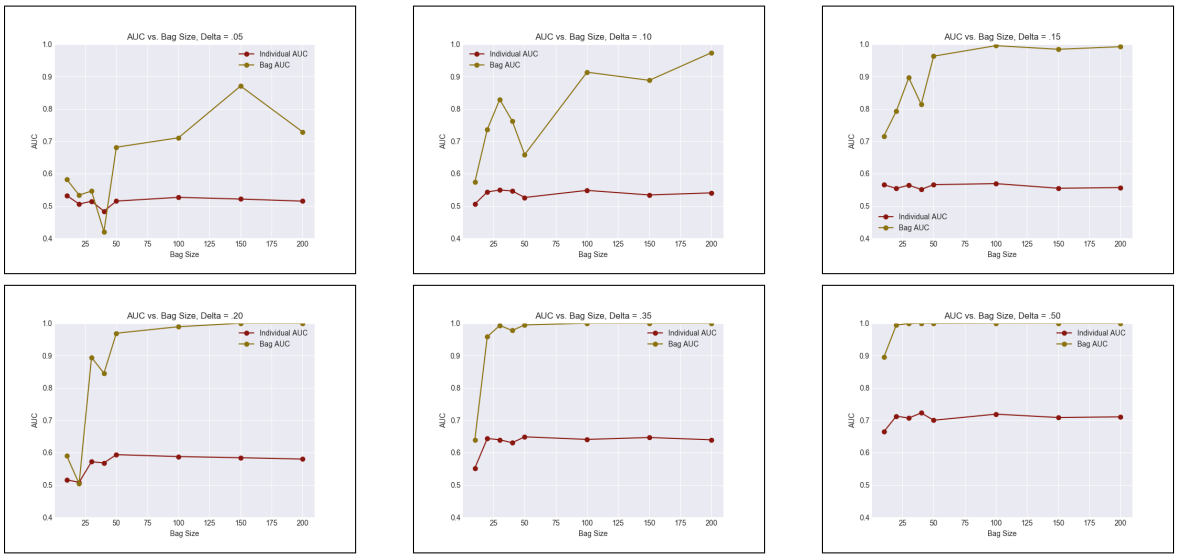

Figure 3-4: Plots demonstrating the relationship between instance AUC, bag AUC, and bag size. Each plot represents a certain difference in witness rate. As the bag size increases, there is a slight increase in instance AUC but an exponential increase for bag AUC.

$N_i * g(f(B))$ be the score of a bag. Unless otherwise stated, all expectations are with respect to the true distributions of instances and/or bags as applicable.

Lemma 3.6.1 (Gaussian Witness Count)**.** *Let B^i be a bag of size N_i with n_i witnesses. Then, for large N_i, n_i, $E[s(B^i)] \sim \mathcal{N}(\mu_i, \sigma_i^2)$, where:*

$$\mu_i \triangleq n_i q_+ + (N_i - n_i) q_- \tag{3.1}$$

$$\sigma_i^2 \triangleq n_i \sigma_+ + (N_i - n_i) \sigma_- \tag{3.2}$$

Proof. Let B^i_+ denote the set of witnesses in bag B^i and let B^i_- denote its complement. By linearity of expectation, $E[s(B^i)]$ can be decomposed into $E[\sum_{\mathbf{x}_{ij} \in B^i_+} f(\mathbf{x}_{ij})] + E[\sum_{\mathbf{x}_{ij} \in B^i_-} f(\mathbf{x}_{ij})]$. Because f is bounded and a deterministic function of i.i.d. examples, the central limit theorem implies that these two expectations are distributed according to $\mathcal{N}(n_i q_+, n_i \sigma_+)$ and $\mathcal{N}((N_i - n_i) q_-, (N_i - n_i) \sigma_-)$. Summing these Gaussians completes the proof. □

Lemma 3.6.2 (Witness Counts with Prior)**.** . *Let B^i be a bag of size N_i with n_i*

witnesses, with $n_i \sim \mathcal{N}(\mu_0, \sigma_0^2)$*, and suppose that* $\sigma_+ = \sigma_-$*. Then, for large* N_i*,* n_i*,* $E[s(B^i)] \sim \mathcal{N}(\tilde{\mu}_i, \tilde{\sigma}_i^2)$*, where:*

$$\tilde{\sigma}_i \triangleq (q_+ - q_-)^2 \left(\frac{1}{\sigma_0^2} + \frac{N_i}{\zeta_i^2} \right)^{-1} \tag{3.3}$$

$$\tilde{\mu}_i \triangleq N_i q_- + \tilde{\sigma}_i \left(\frac{\mu_0}{\sigma_0^2} + N_i \frac{\mu_i}{\sigma_i^2} \right) \tag{3.4}$$

where $\zeta_i^2 \triangleq (q_+ - q_-)^{-2} N_i \sigma_-$

Proof. Observe that equations (3.1) and (3.2) can be rewritten as:

$$\mu_i = N_i q_- + n_i \delta_q \tag{3.5}$$

$$\sigma_i^2 = N_i \sigma_- \tag{3.6}$$

where $\delta_q \triangleq q_+ - q_i$ and the second line follows from the assumption that $\sigma_+ = \sigma_-$. This allows us to rewrite the resulting distribution as:

$$N_i q_- + \delta_q \mathcal{N}(n_i, \zeta_i^2) \tag{3.7}$$

We now have a conjugate prior for the mean for this distribution. Applying well-known formulas gives the posterior variance and mean:

$$\sigma^2 = \left(\frac{1}{\sigma_0^2} + \frac{N_i}{\zeta_i^2} \right)^{-1} \tag{3.8}$$

$$\mu = \tilde{\sigma}_i^2 \left(\frac{\mu_0}{\sigma_0^2} + N_i \frac{\mu_i}{\zeta_i^2} \right) \tag{3.9}$$

Inverting the offset and scaling completes the proof. □

Lemma 3.6.2 shows that, given a normal prior over the number of witnesses in a bag, there is a normal posterior over the predicted number of witnesses (assuming that $\sigma_+ = \sigma_-$). Of course, an exactly Normal prior may not be ideal, since it does not account for the fact that the number of witnesses cannot be less than 0. However, it can arise as a close approximation to other priors that would account for the limited

support—in particular, a truncated normal prior or a Beta prior over the fraction of instances that are witnesses.

Building on Lemma 3.6.2 allows us to show that the expected AUC also has a closed-form expression.

Lemma 3.6.3 (Gaussian Score Differentials)**.** *Suppose that* $n_i \sim \mathcal{N}(\mu_\omega, \sigma_\omega^2)$ *when* $y_i = \omega$ *and* $n_i \sim \mathcal{N}(\mu_{\bar{\omega}}, \sigma_{\bar{\omega}}^2)$ *otherwise. Let* B_i *and* B_k *be bags such that* $y_i = \omega$ *and* $y_k \neq \omega$*. Then, using the assumptions and definitions of Lemma 3.6.2:*

$$s(B_i) - s(B_k) \sim \mathcal{N}(\mu_\Delta, \sigma_\Delta^2) \tag{3.10}$$

where

$$\sigma_\Delta^2 \triangleq (q_+ - q_-)^2 \left[\left(\frac{1}{\sigma_\omega^2} + \frac{N_i}{\zeta_i^2} \right)^{-1} + \left(\frac{1}{\sigma_{\bar{\omega}}^2} + \frac{N_k}{\zeta_k^2} \right)^{-1} \right] \tag{3.11}$$

$$\mu_\Delta \triangleq (N_i - N_k) q_- \left[+\tilde{\sigma}_i \left(\frac{\mu_\omega}{\sigma_\omega^2} + N_i \frac{\mu_i}{\sigma_i^2} \right) - \tilde{\sigma}_i \left(\frac{\mu_\omega}{\sigma_\omega^2} + N_k \frac{\mu_k}{\sigma_k^2} \right) \right] \tag{3.12}$$

Proof. Because the posterior distributions of $s(B_i)$ and $s(B_k)$ are normal, their difference is also normal with mean equal to the difference of the means and variance equal to the sum of the variances. □

Theorem 3.6.1 (AUC expression)**.** *. Let* B_i*,* B_k*,* μ_Δ*,* σ_Δ^2*, and the distribution of* n_i *be defined as in Lemma 3.6.3. Then the expected AUC of the classifier* f *is given by* $1 - \Phi_\Delta(0)$*, where* $\Phi_\Delta(\cdot)$ *is the cumulative distribution function of* $\mathcal{N}(\mu_\Delta, \sigma_\Delta^2)$*.*

Proof. Recall that AUC can be defined as the fraction of the possible (positive, negative) bag pairs in which the positive bag is scored above the negative bag. In expectation, this is equal to the probability of a positive bag being scored above a negative bag. By Lemma 3.6.3, this difference is normally distributed, and > 0 exactly when $s(B_i) > s(B_k)$. The probability of the score being higher for the positive bag is therefore the portion of this distribution greater than 0, which can be expressed as $1 - \Phi_\Delta(0)$. □

Chapter 4

Summary and Conclusion

In this thesis, we presented the success of representation learning in two areas: metric learning for multivariate time series and multiple instance learning for patient risk stratification.

In the first section of this work, we presented Jiffy, a simple and efficient metric learning approach to measuring multivariate time series similarity. We show that our method learns a metric that leads to consistent and accurate classification across a diverse range of multivariate time series. Jiffy's resilience to hyperparameter choices and consistent performance across domains provide strong evidence for its utility on a wide range of time series datasets.

In the second section of this work, we presented the application of multiple instance learning to an important risk stratification problem. Despite the simplicity of the proposed model, we achieve state-of-the-art results compared to existing risk metrics. This leads us to believe that approaching the risk stratification problem as one of multiple instance learning can generalize to scenarios outside of 90-day cardiovascular risk. We also perform empirical demonstrations over different class witness rates to demonstrate this method's ability to rank bags based on presence of a witness instance and outline method behavior in response to different bag sizes. This exploration also justifies the ability of the model to generalize to alternate risk scenarios. To supplement our experiments, we provide a theoretical basis for AUC guarantees of the model in problems with varying witness rates.

The simplicity of this model and its success at classifying an important adverse outcome demonstrate the potential for multiple instance learning on this class of problems. It also implies that although patient labels may only be weakly applicable to constituent heart beats, this level of model supervision enables learning meaningful relationships between adjacent beats.

4.1 Future Work

Both areas discussed present interesting directions for future work. We enumerate some opportunities for further research below.

The extension of Jiffy to multi-label classification and unsupervised learning poses a challenging but necessary task. The availability of unlabeled time series data eclipses the availability of its annotated counterpart. Thus, a simple network-based method for representation learning on multivariate time series in the absence of labels is an important line of work. There is also potential to further increase Jiffy's speed by replacing the fully connected layer with a structured [11] or binarized [59] matrix.

The proposed risk stratification model extends naturally to a range of adverse outcomes. The model is not limited to operating on ECG signals - it is worth exploring whether the multiple instance learning approach may be successful in other modalities of medical data, including voice. On a theoretical level, strong generalization guarantees for distinguishing bags with relative witness rates do not exist and are worth exploring as these models are applied in the real world.

Bibliography

[1] Elliott M Antman, Marc Cohen, Peter JLM Bernink, Carolyn H McCabe, Thomas Horacek, Gary Papuchis, Branco Mautner, Ramon Corbalan, David Radley, and Eugene Braunwald. The timi risk score for unstable angina/non–st elevation mi: a method for prognostication and therapeutic decision making. *Jama*, 284(7):835–842, 2000.

[2] Lods Arnaud, Simon Malinowski, Romain Tavenard, and Laurent Amsaleg. Learning dtw-preserving shapelets. In *Sixteenth International Symposium on Intelligent Data Analysis (IDA 2017)*. Springer International Publishing, 2017.

[3] Anthony Bagnall, Aaron Bostrom, James Large, and Jason Lines. The great time series classification bake off: An experimental evaluation of recently proposed algorithms. extended version. *arXiv preprint arXiv:1602.01711*, 2016.

[4] Gustavo EAPA Batista, Xiaoyue Wang, and Eamonn J Keogh. A complexity-invariant distance measure for time series. In *Proceedings of the 2011 SIAM International Conference on Data Mining*, pages 699–710. SIAM, 2011.

[5] Axel Bauer, Jan W Kantelhardt, Petra Barthel, Raphael Schneider, Timo Mäkikallio, Kurt Ulm, Katerina Hnatkova, Albert Schömig, Heikki Huikuri, Armin Bunde, et al. Deceleration capacity of heart rate as a predictor of mortality after myocardial infarction: cohort study. *The lancet*, 367(9523):1674–1681, 2006.

[6] Nurjahan Begum, Liudmila Ulanova, Jun Wang, and Eamonn Keogh. Accelerating dynamic time warping clustering with a novel admissible pruning strategy. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 49–58. ACM, 2015.

[7] 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.

[8] J Thomas Bigger, Joseph L Fleiss, Richard C Steinman, Linda M Rolnitzky, Robert E Kleiger, and Jeffrey N Rottman. Frequency domain measures of heart period variability and mortality after myocardial infarction. *Circulation*, 85(1):164–171, 1992.

[9] Davis W Blalock and John V Guttag. Extract: Strong examples from weakly-labeled sensor data. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 799–804. IEEE, 2016.

[10] Davis W Blalock and John V Guttag. Bolt: Accelerated data mining with fast vector compression. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 727–735. ACM, 2017.

[11] Mariusz Bojarski, Anna Choromanska, Krzysztof Choromanski, Francois Fagan, Cedric Gouy-Pailler, Anne Morvan, Nouri Sakr, Tamas Sarlos, and Jamal Atif. Structured adaptive and random spinners for fast machine learning computations. *arXiv preprint arXiv:1610.06209*, 2016.

[12] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. In *Advances in Neural Information Processing Systems*, pages 737–744, 1994.

[13] Alessandro Camerra, Themis Palpanas, Jin Shieh, and Eamonn Keogh. isax 2.0: Indexing and mining one billion time series. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 58–67. IEEE, 2010.

[14] A John Camm, Marek Malik, J Thomas Bigger, Günter Breithardt, Sergio Cerutti, Richard J Cohen, Philippe Coumel, Ernest L Fallen, Harold L Kennedy, RE Kleiger, et al. Heart rate variability. standards of measurement, physiological interpretation, and clinical use. *European heart journal*, 17(3):354–381, 1996.

[15] Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslain Gagnon. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition*, 2017.

[16] Zhengping Che, Xinran He, Ke Xu, and Yan Liu. Decade: A deep metric learning model for multivariate time series. 2017.

[17] Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The ucr time series classification archive, July 2015. `www.cs.ucr.edu/~eamonn/time_series_data/`.

[18] Yixin Chen, Jinbo Bi, and James Ze Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1931–1947, 2006.

[19] D Cox. Regression models and life-tables jr statist soc b 34: 187–220. *Find this article online*, 1972.

[20] Zhicheng Cui, Wenlin Chen, and Yixin Chen. Multi-scale convolutional neural networks for time series classification. *arXiv preprint arXiv:1603.06995*, 2016.

[21] Marco Cuturi. Fast global alignment kernels. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 929–936, 2011.

[22] Hoang Anh Dau, Nurjahan Begum, and Eamonn Keogh. Semi-supervision dramatically improves time series clustering under dynamic time warping. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 999–1008. ACM, 2016.

[23] Janez Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30, 2006.

[24] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1-2):31–71, 1997.

[25] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, 2008.

[26] Cao-Tri Do, Ahlame Douzal-Chouakria, Sylvain Marié, Michèle Rombaut, and Saeed Varasteh. Multi-modal and multi-scale temporal metric learning for a robust time series nearest neighbors classification. *Information Sciences*, 418:272–285, 2017.

[27] Eibe Frank and Xin Xu. Applying propositional learning algorithms to multi-instance data. 2003.

[28] Ada Wai-Chee Fu, Eamonn Keogh, Leo Yung Lau, Chotirat Ann Ratanamahatana, and Raymond Chi-Wing Wong. Scaling and time warping in time series querying. *The VLDB Journal—The International Journal on Very Large Data Bases*, 17(4):899–921, 2008.

[29] Gartheeban Ganeshapillai and John V Guttag. Weighted time warping for temporal segmentation of multi-parameter physiological signals. 2011.

[30] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet. *Circulation*, 101(23):e215–e220, 2000.

[31] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet. *Circulation*, 101(23):e215–e220, 2000.

[32] Josif Grabocka, Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–401. ACM, 2014.

[33] Yuan Hao, Yanping Chen, Jesin Zakaria, Bing Hu, Thanawin Rakthanmanon, and Eamonn Keogh. Towards never-ending learning from time series streams. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 874–882. ACM, 2013.

[34] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer, 2015.

[35] Bing Hu, Yanping Chen, and Eamonn Keogh. Time series classification under more realistic assumptions. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 578–586. SIAM, 2013.

[36] Eamonn Keogh. Efficiently finding arbitrarily scaled patterns in massive time series databases. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 253–265. Springer, 2003.

[37] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and information Systems*, 3(3):263–286, 2001.

[38] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM Sigmod Record*, 30(2):151–162, 2001.

[39] Eamonn Keogh, Jessica Lin, and Ada Fu. Hot sax: Efficiently finding the most unusual time series subsequence. In *Data mining, fifth IEEE international conference on*, pages 8–pp. Ieee, 2005.

[40] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[41] Konstantina Kourou, Themis P Exarchos, Konstantinos P Exarchos, Michalis V Karamouzis, and Dimitrios I Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and structural biotechnology journal*, 13:8–17, 2015.

[42] Maria Teresa La Rovere, J Thomas Bigger Jr, Frank I Marcus, Andrea Mortara, Peter J Schwartz, ATRAMI (Autonomic Tone, Reflexes After Myocardial Infarction) Investigators, et al. Baroreflex sensitivity and heart-rate variability in prediction of total cardiac mortality after myocardial infarction. *The Lancet*, 351(9101):478–484, 1998.

[43] Qi Lei, Jinfeng Yi, Roman Vaculin, Lingfei Wu, and Inderjit S Dhillon. Similarity preserving representation learning for time series analysis. *arXiv preprint arXiv:1702.03584*, 2017.

[44] Daniel Lemire. Streaming maximum-minimum filter using no more than three comparisons per element. *arXiv preprint cs/0610046*, 2006.

[45] Shu Liao, Yaozong Gao, Aytekin Oto, and Dinggang Shen. Representation learning: a unified deep learning framework for automatic prostate mr segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 254–261. Springer, 2013.

[46] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11. ACM, 2003.

[47] Yun Liu, Zeeshan Syed, Benjamin M Scirica, David A Morrow, John V Guttag, and Collin M Stultz. Ecg morphological variability in beat space for risk stratification after acute coronary syndrome. *Journal of the American Heart Association*, 3(3):e000981, 2014.

[48] Jiangyuan Mei, Meizhu Liu, Yuan-Fang Wang, and Huijun Gao. Learning a mahalanobis distance-based dynamic time warping measure for multivariate time series classification. *IEEE transactions on Cybernetics*, 46(6):1363–1374, 2016.

[49] David A Morrow, Benjamin M Scirica, Ewa Karwatowska-Prokopczuk, Sabina A Murphy, Andrzej Budaj, Sergei Varshavsky, Andrew A Wolff, Allan Skene, Carolyn H McCabe, Eugene Braunwald, et al. Effects of ranolazine on recurrent cardiovascular events in patients with non–st-elevation acute coronary syndromes: the merlin-timi 36 randomized trial. *Jama*, 297(16):1775–1783, 2007.

[50] Abdullah Mueen, Eamonn Keogh, Qiang Zhu, Sydney Cash, and Brandon Westover. Exact discovery of time series motifs. In *Proceedings of the 2009 SIAM International Conference on Data Mining*, pages 473–484. SIAM, 2009.

[51] Paul D Myers, Benjamin M Scirica, and Collin M Stultz. Machine learning improves risk stratification after acute coronary syndrome. *Scientific reports*, 7(1):12692, 2017.

[52] Juhan Nam, Jorge Herrera, and Kyogu Lee. A deep bag-of-features model for music auto-tagging. *arXiv preprint arXiv:1508.04999*, 2015.

[53] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4004–4012, 2016.

[54] Wenjie Pei, David MJ Tax, and Laurens van der Maaten. Modeling time series similarity with siamese recurrent networks. *arXiv preprint arXiv:1603.04713*, 2016.

[55] Roy H Perlis. A clinical risk stratification tool for predicting treatment resistance in major depressive disorder. *Biological psychiatry*, 74(1):7–14, 2013.

[56] Tina Raissi, Alessandro Tibo, and Paolo Bientinesi. Extended pipeline for content-based feature engineering in music genre recognition. *arXiv preprint arXiv:1805.05324*, 2018.

[57] Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. Searching and mining trillions of time series subsequences under dynamic time warping. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 262–270. ACM, 2012.

[58] Thanawin Rakthanmanon, Eamonn J Keogh, Stefano Lonardi, and Scott Evans. Time series epenthesis: Clustering time series streams requires ignoring some data. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 547–556. IEEE, 2011.

[59] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*, pages 525–542. Springer, 2016.

[60] Chotirat Ann Ratanamahatana and Eamonn Keogh. Everything you know about dynamic time warping is wrong. In *Third Workshop on Mining Temporal and Sequential Data*, pages 22–25. Citeseer, 2004.

[61] Chotirat Ann Ratanamahatana and Eamonn Keogh. Making time-series classification more accurate using learned constraints. In *Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 11–22. SIAM, 2004.

[62] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978.

[63] Patrick Schäfer. Towards time series classification without human preprocessing. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pages 228–242. Springer, 2014.

[64] Patrick Schäfer and Ulf Leser. Fast and accurate time series classification with weasel. *arXiv preprint arXiv:1701.07681*, 2017.

[65] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[66] Pavel Senin and Sergey Malinchik. Sax-vsm: Interpretable time series classification using sax and vector space model. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1175–1180. IEEE, 2013.

[67] Aditya M Sharma, Ajay Gupta, P Krishna Kumar, Jeny Rajan, Luca Saba, Ikeda Nobutaka, John R Laird, Andrew Nicolades, and Jasjit S Suri. A review on carotid ultrasound atherosclerotic tissue characterization and stroke risk stratification in machine learning framework. *Current atherosclerosis reports*, 17(9):55, 2015.

[68] Mohammad Shokoohi-Yekta, Jun Wang, and Eamonn Keogh. On the non-trivial generalization of dynamic time warping to the multi-dimensional case. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 289–297. SIAM, 2015.

[69] Zeeshan Syed, Benjamin M Scirica, Satishkumar Mohanavelu, Phil Sung, Eric L Michelson, Christopher P Cannon, Peter H Stone, Collin M Stultz, and John V Guttag. Relation of death within 90 days of non-st-elevation acute coronary syndromes to variability in electrocardiographic morphology. *American Journal of Cardiology*, 103(3):307–311, 2009.

[70] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[71] PS Wasan, M Uttamchandani, S Moochhala, VB Yap, and PH Yap. Application of statistics and machine learning for risk stratification of heritable cardiac arrhythmias. *Expert Systems with Applications*, 40(7):2476–2486, 2013.

[72] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *arXiv preprint arXiv:1708.02709*, 2017.

[73] Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. Multi-instance learning by treating instances as non-iid samples. In *Proceedings of the 26th annual international conference on machine learning*, pages 1249–1256. ACM, 2009.