An Ensemble Approach to Time Dependent Classification

2018 17th IEEE International Conference on Machine Learning and Applications

An Ensemble Approach to Time Dependent Classification

Florian van Daalen*, Evgueni Smirnov*, Nasser Davarzani *, Ralf Peeters*, Joël Karel *
and Hans-Peter Brunner-La Rocca†
* DKE, Maastricht University, 6200MD Maastricht, The Netherlands
Email: florian.vandaalen@alumni.maastrichtuniversity.nl, {smirnov,n.davarzani,ralf.peeters,joel.karel}@maastrichtuniversity.nl
†Maastricht UMC, 6202AZ Maastricht, The Netherlands
Email: hp.brunnerlarocca@mumc.nl

***Abstract*—A difficult aspect of a time dependent classification task is that the data are not IID sampled. To model this dependency several approaches in longitudinal analysis were developed. However, these approaches either have trouble estimating their generalization performance or are parametric in a statistical sense. To overcome these problems we propose in this paper a new approach of time dependent ensembles. Our approach decomposes the time dependent classification task into a series of classification tasks with IID sampled data. Each task can be solved by a classifier that is not supposed to model any dependency in the data. This allows for the use of a much broader spectrum of existing approaches than is possible on the original data. The classifiers associated with the tasks form the time dependent ensembles. The ensembles estimates the final class of the objects being classified by using a voting scheme. The experiments show the potential of the time dependent ensembles.**

***Index Terms*—Prediction methods; Time series analysis; Ensembles;**

I. Introduction

Most machine learning algorithms assume that the data is generated under the IID assumption [1], [2]. However, this assumption is often violated for many tasks. An example includes the task of time-dependent classification. In this task we predict a class of a new object at a given point of time given labeled objects measured over time. The object measurements from consecutive time points introduce dependencies in the data. Thus, the data is not IID sampled and cannot be handled by standard machine learning algorithms based on the IID assumption.

The time-dependent classification task is a common task in medical research. Our team is involved in a project for analyzing the data of heart-failure patients. The goal of the project is to identify a combination of biomarkers that predict the survival of those patients. Several characteristics for each patient are recorded at subsequent time points and they introduce dependencies in the data. To tackle these dependencies we first employed several approaches from longitudinal analysis such as: generalized estimating equation (GEE) [3] and generalized linear mixed models (GLMM) [4]. However, these approaches either have difficulties with estimating generalization performance or are parametric in a statistical sense.

To address these problems we propose a new approach to time dependent classification: time dependent ensembles. The key idea is to decompose the time dependent classification task into a series of classification tasks with IID sampled data. The data for each of these tasks consists of the data of the objects measured for a concrete time point and labeled according to class distribution observed in the final time point for which classification is required. This makes the data IID sampled and allows for each task to train a classifier. All the classifiers associated with the tasks form time dependent ensembles. The ensembles estimates the final class of the objects being classified by using a voting scheme. By construction the time dependent ensembles do not impose restrictions on the classifiers: they can be trained by any machine learning algorithm. In this way we extend the repertoire of algorithms applicable to time dependent classification beyond the existing methods from longitudinal analysis.

If we continue to contrast the time dependent ensembles with respect to the longitudinal methods we note that:

- the generalization performance of time dependent ensembles can be estimated using any standard validation method such the hold-out method and cross validation;
- the time dependent ensembles can be parametric/non-parametric depending on the machine learning algorithms used;
- the time dependent ensembles allow using any feature selection method available.

The remainder of the paper is organized as follows. Section II formalizes a specific time dependent classification task that we consider. Relevant work is provided in Section III. Section IV introduces the time dependent ensembles. Experiments are provided in Section VI. Section VII concludes the paper.

II. Time Dependent Classification Task

We assume an existence of a set $\mathcal{O}$ of all possible objects and a training set $O \subseteq \mathcal{O}$ of objects under investigation. Each object evolves over time. The time is represented discretely by sequential time points given by integers $t \in \mathbb{N}^0 : 0 \leq t \leq T$ so that $T+1$ is the total number of time points. The set of objects $o \in O$ that exist in time point t is denoted by O_t. We assume that $O_t \supseteq O_{t+1}$ for $t \in \{0, \ldots T\}$.

Any object $o \in O$ can be represented in an instance space X (usually defined by I number of variables X_i with $i \in \{1, \ldots, I\}$). To map the objects to instances we introduce a

978-1-5386-6805-4/18/$31.00 ©2018 IEEE
DOI 10.1109/ICMLA.2018.00164

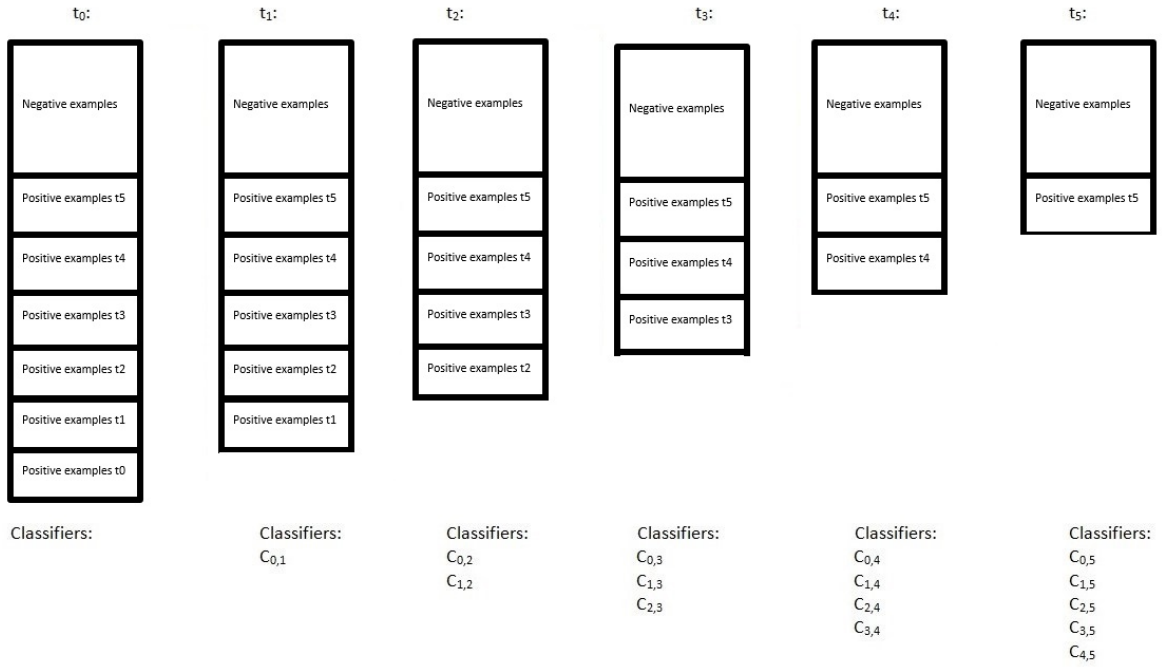

Fig. 1. The data sets and classifiers for sequential time points.

function $x : \mathcal{O} \times \mathbb{N}^0 \to X$. The function maps any object $o \in O$ and time point $t \in \{0, \dots T\}$ to an instance (description) $x \in X$ of the object o.

The class of an object $o \in O$ is determined for time point t depending whether o belongs to the set O_t (of objects in t). If $o \in O_t$ we say that the class is negative; otherwise, the class is positive. To represent the classes we introduce a set Y of class labels. Y consists of labels "$+$" and "$-$" for the positive and negative classes, respectively. To associate an object $o \in O$ with a class label $y \in Y$, we introduce a function $y : \mathcal{O} \times \mathbb{N}^0 \to Y$. The function is defined for any object $o \in O$ and time point $z \in Z$ as:

$$\begin{cases} + & \text{if } o \notin O_t, \\ - & \text{if } x \in O. \end{cases}$$

The time dependent classification task we consider is a binary classification task. Given a set $O \subseteq \mathcal{O}$ of training objects and a test object $o \in \mathcal{O}$ that exists in time given by time points t and $t + k - 1$ for some positive integer k, the task is to predict the class label $y \in Y$ of the object o for time point $t + k$.

Figure 1 provides a vizualization for the object sets O and O_t. We note that in order to operate with these sets we need to label them and then to repesent using the functions x and y. Given time point $t \in \{0, \dots T\}$ and a positive integer k so that $t + k \leq T$ we define data set $D_{t,t+k}$ as:

$$\{(x, y) \in (X \times Y) \mid x = x(o, t) \wedge y = y(o, t) \wedge o \in O_t\}.$$

The set $D_{t,t+k}$ consists of those objects $o \in O$ that exist in time point t. They are labeled as positive if they do not exist in time point $t + k$; otherwise, they are labeled as negative.

III. Relevant Work

The straightforward solution to the task of time dependent classification is to unite the datasets $D_{t,t+k}$ for all $t \in \{0, \dots T\}$. However, this union will contain a lot of instance dependencies due to the fact that by construction many instances represent the same or very similar objects. This implies that the IID assumption is violated.

The existing approaches to time dependent classification can be separated into two groups: approaches for continuous outcome variables and approaches for binary outcome variables. Examples of a continuous outcome variable would be *time of death* or *time between separate hospitalizations*. An example of a binary outcome variable would be *is the patient hospitalized in month X*.

For continuous outcome variables relevant survival analysis methods are a Cox model ([5]), when predicting time of death; and Gap-time analysis methods ([6]) when predicting time between hospitalizations. However, both methods only take into account the value of time dependent variables at the time of the events. As such they may miss vital information.

For binary outcome variables there are a few methods in longitudinal analysis methods such as generalized estimating equation (GEE) ([3]) and generalized linear mixed models (GLMM) ([4]). Although both methods have been applied for time dependent classification, they do suffer from several problems. Estimating the generalization performance of the GEE methods is rather difficult. There exist several goodness-of-fit tests. However, there is no method that allows to estimate the generalization performance beyond the training data. This severely restricts practical usage of the GEE models, since we cannot decide whether they can be employed for classifying new cases. This also makes comparing them to our new approach impossible.

The GLMM methods have an extended repertoire of estimation methods. However, they are not distribution free; i.e., they assume certain distribution on the input variables. This definitely restricts practical usage of the GLMM models to applications where those assumptions (approximately) hold.

From the above we may conclude that we need a method for the time dependent classification task that: (1) can handle the non-IID data, (2) is non-parametric, and (3) can be easily used to estimate the generalization performance beyond training data. In the next section we provide such a method which we call time dependent ensembles.

IV. Time Dependent Ensembles

Time dependent ensembles are proposed for the time dependent classification task introduced in Section II. The starting point to introduce these ensembles is to observe that any data set $D_{t,t+k}$ is IID sampled for $t \in \{0,\ldots,T\}$ and integer k such that $t+k \leq T$. Hence, we can train any machine learning classifier $C_{t,t+k}$ on $D_{t,t+k}$ ($C_{t,t+k} : X \rightarrow Y$) that does not have to model any data dependency. The set of all these classifiers for time point $t+k$ for a time dependent ensemble $E_{t,t+k}$. Formally, the ensemble is defined as:

$$E_{t,t+k} = \{C_{s,t+k} \mid t \leq s \leq t+k-1\}.$$

The classification process of the ensemble $E_{t,t+k}$ is organized as follows. Given a test object $o \in O$ that exists in time between point t and point $t+k-1$, we first create instance (description) x_s for any $s \in \{t,\ldots,t+k-1\}$. Each instance x_s is classified by the corresponding classifier $C_{s,t+k}$. We note that the classifiers $C_{s,t+k}$ predict for the same time point $t+k$. Thus, to estimate the class of the test object o we can employ any voting scheme (we employed probability averaging in our experiments).

Any voting scheme improves the class estimation if the classifiers $C_{s,t+k}$ in the ensemble $E_{t,t+k}$ are rather (error) independent. However, here we have to note that these classifiers can be dependent because they represent the same objects that evolve over time. Thus, classifier selection is an important topic to investigate for future research.

A nice property of the time dependent ensembles is that they employ all the information available through the data sets $D_{s,t+k}$ for $t \leq s \leq t+k-1$ (see Figure 1). The bigger is the horizon k, the bigger is the size of the ensemble $E_{t,t+k}$; i.e. the desired classifier diversity. However, it should be noted that when k is too big the classifiers $C_{s,t+k}$ for s close to the staring time point t can be outdated. This can be easily identified by a validation process and the outdated classifiers can be just simply removed from the ensemble $E_{t,t+k}$. This is a simple and elegant solution to the problem of concept drift [7], [8].

The generalization performance of the time dependent ensembles can estimated using cross validation. However, the cross validation process is a bit more complicated than it normally would be for a single classifier. We first divide the set O_t of objects at time point t into L equally sized folds O_t^l for $l \in \{1,\ldots,L\}$. The class stratification is based on the class distribution of the class label set $\{y = y(o,t+k) \mid o \in O_t\}$. Due to the direct object correspondence the L-fold division of the set O_t imposes a L-fold division of the set O_s of objects at any time point s between t and $t+k-1$; e.g. $O_s^l = O_t^l \cap O_s$. Thus, cross validation folds O_s^l of objects are defined for any s and l.

The cross validation folds O_s^l are represented on the data set level by the data sets $D_{s,t+k}^l$ defined as follows:

$$\{(x,y) \in D_{s,t+k} \mid x = x(o,t) \wedge y = y(o,t) \wedge o \in O_t^l\}.$$

The definition of the data set folds $D_{s,t+k}^l$ allows us to organize a cross validation process for each individual classifier $C_{s,t+k}$. By combining the votes of all the classifiers $C_{s,t+k}$ we implement the L-fold cross validation process of the ensemble $E_{t,t+k}$.

V. Feature Selection

The time dependent ensembles can be applied in a combination with feature selection. Below we consider three possible approaches.

A. Individual Filter

The individual filter (*IF*) approach assumes that feature selection is realized *separately* for each individual classifier $C_{s,t+k}$ within an ensemble $E_{t,t+k}$. It usually results in an individual list of features for each classifier $C_{s,t+k}$. This can be advantageous for individual adaptability of the classifiers $C_{s,t+k}$ in the ensemble $E_{t,t+k}$. However, the approach cannot identify features that are important for the whole ensemble.

B. Ensemble Wrapper Filter

The ensemble wrapper filter (*EWF*) does feature selection directly on the ensemble $E_{t,t+k}$ as a whole [1]. As such it no longer needs to recombine the lists of features selected on an individual classifier level and it is capable of finding synergy among features across the ensemble.

[1]Note that a univariate and a multivariate filter cannot be applied on the ensemble level.

C. Average Ensemble Wrapper Filter

The average ensemble wrapper filter (*AEWF*) averages the generalization performance of all ensembles $E_{t,t+k}$ (for all valid t and k) to determine the best set of features. It provides the most complete feature overview with respect to the entire data set. However, as *AEWF* considers all ensembles at the same time it has the highest time complexity among the three feature selection approaches proposed.

VI. Experiments and Results

A. The Original Data

Our experiments are based on the time dependent classification data of heart-failure patients described in Section I. The data consists of 248 features, out of which approximately one third are time-varying and contains information about 622 patients. Part of the patients are right censored; meaning that we do not have measurements available after a certain time point due to the fact that the patients drop out of the study or die. The class label we are trying to predict corresponds to the death or hospitalization of the patient at a given point in time. The features range from social features, such as marital status and personal features (age & sex), to the so-called biomarkers. An example of a biomarker would be the serum level of certain hormone. These features are a mix of numerical and nominal features. Multiple measurements are done at time points $0, 1, 3, 6, 12$ and 18 which corresponds to the months the patient followed the treatment.

Certain features are uninteresting for classification. Some of these are irrelevant due to the fact that they are not connected to the classification problem at hand. For example a patient's religion does not influence the effectiveness of a treatment. Some are irrelevant due to their unique nature. An example of this would be the feature that is reserved for doctor's comments. These features were manually excluded from the analysis.

B. Decomposing the Data

The data was decomposed for the time dependent ensembles in the manner described in Section II. More precisely, we created the data sets $D_{t,t+k}$ for (month) time points $t \in \{0, 1, 3, 6, 12\}$ and corresponding valid $k \in \{1, 3, 6, 12, 18\}$ so that $t + k \leq 18$. Each data set $D_{t,t+k}$ contained the following: the biomarkers' values at the time t, the general patient info (e.g. sex) which is constant over time, and the class label which indicates whether the patient was hospitalized or died at the end of the time point $t + k$.

C. Set Up

The time dependent ensembles and their classifiers were tested using 10-fold cross validation. An initial short experiment was run with various types of classifiers to determine the best classifier for the ensembles. Based on this logistic regression with a ridge factor of 0.01 was chosen [9]. It allows a user to determine feature p-values as well as provides good results while having acceptable time complexity. The average probability of the positive class label, representing the death or hospitalization of a patient, within an ensemble was used in the voting scheme. The ensembles were tested both with and without the various feature selection approaches described in Section V. The generalization performance of an ensemble was estimated using the ROC area under curve (AUC) [9].

D. Results: No Feature Selection

The results of our experiments with time dependent ensembles that do not employ feature selection can be found in Table I. The table shows that the ensembles outperform the individual classifiers in 7 cases and in 3 cases the individual classifiers perform better. In the remaining cases the ensembles consist of only one classifier and thus perform equally to the individual classifiers. No significant difference between the generalization performances classifiers and ensembles was found using the Wilcoxon test. Nevertheless, the ensembles outperform the worst individual classifiers within the ensembles, indicating that the voting reduces the average error of the individual classifiers.

TABLE I
AUC results of the experiments with time dependent ensembles and individual classifiers when no feature selection was performed. The best classifier/ensemble for each period is given in bold.

	Classifier	Ensemble
0-1	**0,6533**	**0,6533**
0-3	0,7050	**0,7132**
0-6	**0,7663**	0,7318
0-12	0,6433	**0,6648**
0-18	0,7089	**0,7330**
1-3	**0,7254**	**0,7254**
1-6	**0,7297**	0,6876
1-12	0,6619	**0,6839**
1-18	0,7230	**0,7395**
3-6	**0,7142**	**0,7142**
3-12	0,6671	**0,6861**
3-18	0,7224	**0,7340**
6-12	**0,7427**	**0,7427**
6-18	**0,7408**	0,7309
12-18	**0,7438**	**0,7438**
Average	0,7098	**0,7123**

E. Results: Feature Selection

The results of our experiments for the time dependent ensembles applied together with the all three feature selection approaches (from Section V) can be found in Tables II and III. Below we describe them separately.

1) Results: Individual Feature Selection: The individual filter (*IF*) approach was applied together with the correlation-based feature subset selection filter (CFSSF) [10] and the wrapper filter. The method of search in the feature-set space was a genetic algorithm with 30 individuals and 100 generations. The fitness function was AUC. The results are provided in Table II. They show that the CFSSF and wrapper filters improve the generalization performance of the time dependent ensembles with respect to the individual classifiers. The improvement is significant established with the Wilcoxon test.

Table III provides a comparison information. It shows that the *IF* approach with the CFSSF filter and wrappers,

TABLE II
AUC RESULTS OF THE EXPERIMENTS WITH TIME DEPENDENT ENSEMBLES BASED ON THE *IF* APPROACH. THE *IF* APPROACH WAS APPLIED WITH THE CFSSF FILTER AND THE WRAPPER FILTER. THE BEST CLASSIFIER/ENSEMBLE FOR EACH PERIOD IS GIVEN IN BOLD.

	IF CFSSF Filter		*IF* wrapper	
	Classifier	Ensemble	Classifier	Ensemble
0-1	0,7499	0,7499	**0,7528**	**0,7528**
0-3	0,8889	**0,9020**	0,8938	0,9000
0-6	0,7842	0,8285	0,8125	**0,8482**
0-12	0,5691	0,5848	0,7041	**0,7345**
0-18	0,5968	0,6944	0,6356	**0,7299**
1-3	**0,8870**	**0,8870**	0,8859	0,8859
1-6	0,8203	0,8430	0,8367	**0,8448**
1-12	0,6031	0,5900	0,7064	**0,7419**
1-18	0,6602	0,7016	0,6515	**0,7204**
3-6	0,8202	0,8202	**0,8367**	**0,8367**
3-12	0,5038	0,5730	**0,7420**	0,7413
3-18	0,6111	0,6983	0,6033	**0,7113**
6-12	0,6189	0,6189	**0,7959**	**0,7959**
6-18	0,6460	0,6944	0,6460	**0,7062**
12-18	**0,7038**	**0,7038**	0,7020	0,7020
Average	0,6976	0,7260	0,7470	**0,7768**

respectively, outperformed the unfiltered approach showing a significant overall improvement according to the Wilcoxon test.

TABLE III
AUC RESULTS OF THE EXPERIMENTS WITH TIME DEPENDENT ENSEMBLES BASED ON THE *IF* APPROACH, THE *EWF* APPROACH, AND THE *AEWF* APPROACH. THE *IF* APPROACH WAS APPLIED WITH THE CFSSF FILTER AND THE WRAPPER FILTER. THE BEST CLASSIFIER/ENSEMBLE FOR EACH PERIOD IS GIVEN IN BOLD.

	No Filter	*IF* CFSSF	*IF* wrapper	*EWF*	*AEWF*
0-1	0,6533	0,7499	0,7528	0.7528	**0.8060**
0-3	0,705	0,9020	0,9000	**0.9087**	0.8577
0-6	0,7663	0,8285	0,8482	0.8506	**0.8959**
0-12	0,6433	0,5848	0,7345	0.7553	**0.7966**
0-18	0,7089	0,6944	**0,7299**	0.7281	0.7232
1-3	0,7254	**0,8870**	0,8859	0.8859	0.8557
1-6	0,7297	0,8430	0,8448	0.8368	**0.9058**
1-12	0,6619	0,5900	0,7419	0.7647	**0.8100**
1-18	0,723	0,7016	0,7204	**0.7287**	0.7253
3-6	0,7142	0,8202	0,8367	0.8367	**0.9070**
3-12	0,6671	0,5730	0,7413	0.7560	**0.8038**
3-18	0,7224	0,6983	0,7113	**0.7348**	0.7188
6-12	0,7427	0,6189	**0,7959**	**0.7959**	0.7831
6-18	**0,7408**	0,6944	0,7062	0.7181	0.7300
12-18	**0,7438**	0,7038	0,7020	0.7020	0.7238
Average	0,7098	0,7260	0,7768	0.7837	**0.8028**

2) Results: Ensemble Wrapper Feature Selection: The ensemble wrapper filter (*EWF*) was applied on complete time dependent ensembles. The method of search in the feature-set space was a genetic algorithm with 30 individuals and 100 generations. The fitness function was AUC.

Table III shows that the *EWF* approach performs inconsistently. This happens because to the genetic algorithm used got stuck in a local optimum. However, there are several ensembles where the approach provides a significant improvement. Furthermore, it should be noted that the time complexity is considerably worse when compared to the *IF* approach. However, even with the local optima the *EWF* approach still showed a significant improvement compared to the unfiltered approach establsihed with the Wilcoxon test.

3) Average Ensemble Wrapper Feature Selection Experiment Results: The average ensemble wrapper filter (*AEWF*) determined the best set of features based on the averaged generalization performance of all the valid time dependent ensembles. The method of search in the feature-set space was a genetic algorithm with 30 individuals and 100 generations. The fitness function was AUC.

Table III shows that the *AEWF* approach has the best results. Due to the genetic search algorithm it did occasionally get stuck in a local optimum. However, this less problematic, since several time dependent ensembles were used. The Wilcoxon test showed a significant improvement over the unfiltered approach.

We note that the *AEWF* approach is the most computationally intensive approaches to feature selection among those used in our experiments. The approach needs to train all the classifiers for all the ensembles for each individual in each generation of the genetic algorithm.

VII. CONCLUSION

In this paper we introduced ensembles for time dependent classification. These ensembles are based on decomposing any time dependent classification task into a series of classification tasks with IID sampled data. This allows training any type of classifiers for those tasks that form the final ensemble. We showed that the time dependent classification ensembles are non-parametric. Their generalization performance can be estimated using straightforward adaptation of the standard validation methods (e.g. cross validation). The time dependent classification ensembles can be applied together with feature selection and this combination results in the best generalization performance established in our experiments.

REFERENCES

[1] R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye, *Probability and Statistics for Engineers and Scientists.*, 8th ed. Pearson Education International, 2007.

[2] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*, 3rd ed. Morgan Kaufmann, 2012.

[3] K.-Y. Liang and S. Zeger, "Longitudinal data analysis using generalized linear models." *Biometrika*, vol. 73, pp. 13–22, 1986.

[4] N. Breslow and D. Clayton, "Approximate inference in generalized linear mixed models," *Journal of the American Statistical Association*, vol. 88, pp. 9–25, 1993.

[5] D. Cox, "Regression models and life-tables." *Journal of the Royal Statistical Society. Series B (Methodolocial)*, vol. 34, pp. 187–220, 1972.

[6] H. Zhu, "Non-parametric analysis of gap times for multiple event data: An overview," *International Statistical Review*, vol. 82, no. 1, pp. 106–122, 2014. [Online]. Available: http://dx.doi.org/10.1111/insr.12031

[7] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Transactions on Neural Networks*, vol. 22, 2011.

[8] R. Polikar, L. Udpa, S. S. Udpa, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," 2001.

[9] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*. Springer, 2013.

[10] I. H. Witten, E. Frank, M. Hall, and C. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems, 2017.