2016 IEEE 29th International Symposium on Computer-Based Medical Systems

Resource Frequency Prediction in Healthcare: machine learning approach

Daniel Vieira
X-akseli Oy
Espoo, Finland
daniel.vieira@x-akseli.fi

Jaakko Hollmén
Aalto University School of Science
Department of Information and Computer Science
Espoo, Finland
Jaakko.Hollmen@aalto.fi

***Abstract*—Determining the minimal amount of resources needed to ensure minimal number of bottlenecks in the patient flow not only promotes patient satisfaction but also provides financial benefits to hospitals. The increase of data gathering by healthcare facilities in the last years have brought new opportunities to apply machine learning techniques to tackle this problem. This work makes use of data gathered from the Oulu University Hospital in Finland between 2011 and 2014 to study the effectiveness of machine learning techniques to predict resources usage.**

This work investigates the problem of resource frequency prediction and compares the performance of Nearest Neighbours and Random Forest. The application of data clustering as a preprocessing step is also explored as a way to improve the prediction accuracy of resources whose behavior change over time. The results indicate that 1) highly frequented resources can be predicted with higher accuracy than the lowly frequented resources; 2) the Random Forest have similar performance to the Nearest Neighbours although Random Forest performs better; 3) clustering improves the performance of the Nearest Neighbours but not of Random Forest; and 4) if averages are used to determine the resource frequency then cluster averages yields higher accuracy than all data averages.

***Keywords*-supervised learning; regression; time series prediction; hierarchical clustering; healthcare modelling**

I. Introduction

Hospital simulation and optimization has been a problem of interest since 1960s [1] and for understandable reasons. There is a common struggle in healthcare centers to efficiently determine how many and which kind of resources are needed while improving and keeping a desirable patient flow. Although it's feasible to manually tackle this problem for small clinics, the task becomes considerably harder as they grow. Not only does the increase in the number of resources bring the problem of individual resource optimization but also the interactions that may occur within them which will need to be taken into consideration. The consequences of failing to address these concerns has been shown to lead to financial problems for the institution as well as an overall increasing of waiting times and, consequently, patients dissatisfaction [2].

Operations research methods have been the primary approach to address the problem of hospital simulation and optimization. The growing interest on the application of these methods started mostly after 1990s with the appearance of many publications attempting to provide a solution for different hospital settings [3]. However a few authors have expressed concerns on how to validate many of the published results and whether such implementations could bring a positive impact to the hospitals [4].

Over the last decade many healthcare centers, especially in Finland, have started to adopt patient data system which have led to the gathering of a considerable amount of patient data. This data includes information regarding the patients schedules, treatments and, more recently, out-patient flow. The large gathering of these data has brought new opportunities for applying Machine Learning techniques to address the previous mentioned issues.

The propose of this work is to provide a solution to a main component [5] needed to build an hospital simulation and optimization tool: the prediction of resources usage. A successful prediction of the resources frequency allows healthcare facilities to anticipate months of high patient flow as well as identify resources that will have low and high usage. This is a time series problem that will be addressed with regression methods in Section III, specifically the Nearest Neighbours and Random Forest algorithms. The proposed approaches were experimented on historical data of Oulu Hospital in Finland, presented in Section II. The experiment procedure and the results with discussion are given in Section IV and V, respectively.

II. Healthcare Patient Flow Data

The data examined in this work is the patient flow data from Oulu University Hospital of Finland that was provided by X-akseli Oy in agreement with the customer. The data contains information related with the hospital organization, such as locations and resources, the anonymous patients id and birthday, their respective visit schedules and the event states that define the patient flow. The schedules have four main entities: visits, reservations, patients and resources. Hierarchically a schedule is a set of visits that consist of one of more reservations which per se are defined as the appointment of a patient to a resource. A resource may be a doctor, machinery such as X-Ray or a set of two or more resources, however, their definition is not explicitly specified in the data.

2372-9198/16 $31.00 © 2016 IEEE
DOI 10.1109/CBMS.2016.59

Table I
OVERVIEW OF THE PATIENT FLOW DATA FROM OULU UNIVERSITY HOSPITAL FOR THE AVOHOITOTALO BUILDING, FLOOR R & 1.

	Av. Floor R & 1
Visits	276 696
Reservations	366 529
Resources	131
Patients	101 523
Reservations per Visit	1.325

The provided data was gathered between August 2011 to November 2014. In total there are 1 102 886 visits and 1 453 398 reservations, with a ratio of 1.319 reservations per visit. There are 255 884 patients and 986 resources many of which are no longer available in hospital due to being removed or replaced by another.

Since this is an initial study on this data, this work scope has been reduced to the reservations made on the resources in Avohoitotalo building of floor R and 1. Floor R contains the resources used for diagnostics, such as X-Ray and MRI, while resources in floor 1 are related to surgery and neurosurgery. This data subset corresponds to about 25% of the total reservations and to 13% of the all resources. In the case of floor R and 1, of the 131 resources only 90 have been used in 2014. A summary of these statistics for the specified subset are presented in Table I.

III. METHODS

The daily prediction of resource usage can be interpreted as a problem of time series prediction. This is considered a type of quantitative forecasting that applies models created from successive points in time to predict future values. Originally time series methods were based on mathematical modeling [6] and time-frequency analysis [7] but have started recently to use machine learning methods, primarily artificial neural networks [8] and local learning techniques [9]. This section focus on the application of the machine learning techniques for time series prediction, specifically by using the nearest neighbour algorithm and the random forest regression algorithm.

The machine learning strategies follow a divergent interpretation of the forecasting problem compared to the classical approach [10]. Rather than the behaviour being random the view is that it may be deterministic, with not many but few number of degrees of freedom that in fact interact non-linearly. The mathematical formulation of this approach can be derived from a state space reconstruction problem and by using the Takens theorem [11] to reach the following representation:

$$y_t = f(y_{t-d}, y_{t-d-1}, ..., y_{t-d-n+1}) + w(t) \tag{1}$$

where f is a deterministic process, d is the *lag time*, order n is the number previously used observations and w is a noisy term that accommodates for missing data. This noisy term was included in case the deterministic process f is unable to precisely outline the time series.

When the past data of time series is available the Equation 1 implies that forecasting can be interpreted as supervised learning problem [10]. This means that once the model f is created according to the historical data then it will be possible to predict a future outputs by providing a new input vector.

A. Supervised Learning Setting

Upon finding the modelling function f the forecasting problem can be interpreted as a generic regression problem by considering as inputs the last n observations [10]. This means the data set is composed by a an input matrix X of size $[(N-d-n-1) \times n]$ and a respective output vector Y of size $[(N-d-n-1) \times 1]$ so that

$$X = \begin{bmatrix} y_{N-1} & y_{N-2} & \cdots & y_{N-n-1} \\ y_{N-2} & y_{N-2} & \cdots & y_{N-n-2} \\ \vdots & \vdots & \vdots & \\ y_n & y_{n-1} & \cdots & y_1 \end{bmatrix} \qquad Y = \begin{bmatrix} y_N \\ y_{N-1} \\ \vdots \\ y_{n+1} \end{bmatrix} \tag{2}$$

where N is the total number of observations. This translates to the problem of one-step forecasting considering a single output value is predicted individually.

Local learning techniques are considered since these methods have very few assumptions and can deal with non-stationary processes by interpreting neighbourhoods spatially and temporally. The Random Forest method is also taken in to account as it has been empirically demonstrated [12] to have a state of the art performance on a variety of real-world problems.

B. Nearest Neighbours Algorithm

Nearest Neighbors [13] is among the simplest methods in machine learning that uses local approximations. The algorithm consists in finding the K closest neighbours for a given input vector and use the output of those neighbours to compute the next observation. With respect to the formulation 2, let $X[i]$ be the i row in X and $Y[i]$ the according output value. This can also be written as x_i and y_i, respectively. In order to predict y_t the nearest neighbours algorithm computes the distance of each row in X with $X[t]$, chooses the K less distant neighbours $X_n = \{x_{k1}, x_{k2}, \cdots, x_{kK}\}$ and computes an approximation of y_t by taking the average, or another alternative statistic, of $Y_n = \{y_{k1}, y_{k2}, \cdots, y_{kK}\}$.

Knowing the optimal number of neighbours K is a key decisive element that underline this method effectiveness. The simplest solution to computing K, besides trivially choosing a fixed value beforehand, is by applying Cross-validation

(CV). This technique has been successfully applied in the context of time series prediction [10] primary using the Leave-one-out (LOO) criterion.

C. Leave-one-out Cross-validation Criterion

Leave-one-out Cross-validation (LOOCV) is a type of CV that consists in removing one observation of the data, in order to use it as validation data, and build the model on the remaining samples, known as training data. Let $\hat{y}_t^{(k)}$ be the predicted value at time t when k neighbours are chosen. If the average is applied to compute the approximation then,

$$\hat{y}_t^{(k)} = \frac{1}{k}\sum_{i=1}^{k} Y_n[i] \tag{3}$$

where Y_n is the vector of outputs sorted according to the distance metric applied between the x_t and the previous observations. The objective of LOOCV is to find k which minimizes an error function $e_{LOO}(k)$. As suggested by Bontempi et al. (2013) [10] this computation can efficiently be achieved by using the PRESS statistic [14], meaning the $e_{LOO}(k)$ is the adjusted sum of squared error defined as:

$$e_{LOO}(k) = \frac{1}{k}\sum_{i=1}^{k}\left(Y_n[i] - \frac{1}{k-1}\sum_{j=1(j\neq i)}^{k} Y_n[j]\right). \tag{4}$$

Hence the optimal number of neighbours $\hat{k}$ is calculated as $\hat{k} = \underset{k\in\{2,\ldots,N\}}{argmin}\ e_{LOO}(k)$.

D. Random Forest

Random Forest (RF) [15] is a supervised ensemble learning method that makes use of bootstrap aggregating and random feature selection. In the context of equation 2, let D be the data set in which X is the input values and Y the respective outputs. Random forest constructs B decision trees so that each tree is built in a two-step process. First a bootstrap sample D_i is created from D. Second, the tree T_i is constructed on D_i with the condition that each node chooses, and only splits, a random subset of m features for $m <= n$. For new input samples Random Forest predicts the respective output by assembling the results of the each tree. With regards to regression the average is taken of the trees prediction which is mathematically formulated as,

$$\hat{y}_t = \frac{1}{B}\sum_{i=1}^{B} \hat{y}_t^{(T_i)} \tag{5}$$

where $\hat{y}_t^{(T_j)}$ is the approximation of y_t according to tree T_i.

An individual tree is subjected to high variance since it's particularly sensitive to the arrangement of data samples. Random Forests tackles this issue by averaging an assemble of trees that per se are built on random subspaces of the data. This concept is based on the bias-variance tradeoff [16] which asserts the accuracy of a model is increased by minimizing at the same time the bias and variance error.

E. Clustering

Cluster analysis is a type of unsupervised learning whose purpose is to partition the data into groups, known as clusters. The partition process determines the similarity between data points by applying a distance metric or an alternative measure of closeness. Hierarchical clustering is a type of clustering based on either the iterative partition of the data into different set of clusters, or the recursive agglomeration of smaller clusters. Nevertheless both approaches require a linkage criterion [17] in order to interpret the distances between clusters. For example the $complete$ linkage regards the distance between two clusters as the distance between the two furthest points of each cluster.

Clustering can be useful not only to handle and analyse data relationships but to also to serve as an auxiliary tool in the supervised learning process [18]. Regarding the data in study, there are resources whose frequency vary considerably over the year and during days of the week. For instance, consider the hierarchical clustering result in Figure 1 of clustering the monthly usage of an X-Ray Lab resource named RNAT13. The resource frequencies were summed by month and normalized so that the sum of the months frequencies is 12. The dendrogram of RNAT13 presents three obvious clusters: $July$, $[Nov, May, Oct]$ and remaining months. This separation is understandable if the amounts per cluster are inspected: $July$, an holiday month, only had 13 reservations while $[Nov, May, Oct]$ had between 742 and 861 reservations and the last cluster of months had at least 389 and at most 654.

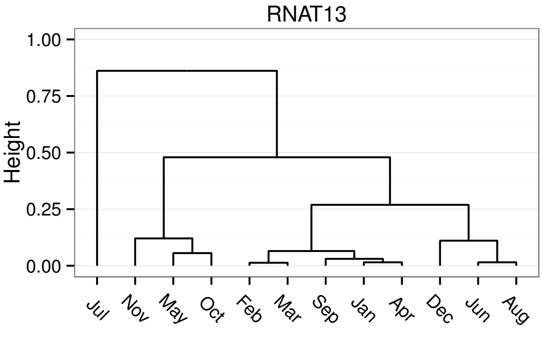

Figure 1. Hierarchical Clustering with Euclidean distance and complete-linkage of resource RNAT13 frequencies, grouped by month. The amounts were normalized so that the sum of the months frequencies is 12.

Consequently if these variations are not taken into account then it may have a negative affected on the supervised learning performance. A solution is to effectively split the data according to the clustering result which means a model is built for each cluster and used accordingly for new samples.

F. Evaluation Metric

The time series prediction will be executed on each resource independently and therefore, in order to compare the performance across different series, the evaluation metric needs to be scale independent. Mean Absolute Scaled Error (MASE) [19] is an scale-free error metric which handles the problems found in percentage errors. This metric performs the evaluation by comparing the provided prediction against the naive approach of considering the previous observation as the new prediction, mathematically written as,

$$MASE(Y_t, \hat{Y_t}) = \frac{1}{N} \left| \frac{\sum_{i=1}^{N} Y_i - \hat{Y_i}}{\frac{1}{N-1}\sum_{i=2}^{N} |Y_i - Y_{i-1}|} \right| \tag{6}$$

This means a result higher or equal to 1 implies that the naïve method is better or the same as the applied prediction method. Consequently a result lower than 1 indicates the used prediction method outperforms the näive method and thus the lower the result the better the prediction.

IV. Experiment

The goal of this experiment is to evaluate how effective a resource usage can be predicted in a real-world case scenario. In order to accomplish this the experiment was conducted using a common supervised learning setup on Oulu data. The experiment was executed in two phases: training and validation phase for which the data was split into three sets: training set, validation set and test set. In the training phase models were built with different parameters using the training set. The created models were then passed on to the validation phase where the best performing model was chosen according to the validation set. Lastly the test set was used to estimate the error of the chosen model.

A fact noted by the hospital staff was that resources perform differently during the year. Thus, in order to have a setup close to reality, the experiment was split in five trials for different times during the year. For each trial, two weeks of data was used for validation and the following 30 days were used for the test set. Both validation and test set were made with data from 2014. The training set comprises of data from January 2012 up until the previous day before the start of the validation set. Overall there was one trial for the Spring season, two for the Summer and another two for the Winter.

The experiment had a preprocessing step in which the data was re-arranged according to the supervised learning setting presented in Section III-A. The last 14 days were used as features for this setting. Additionally the data was clustered by month and weekday in which a model was created for each cluster. The clustering was accomplished with hierarchical clustering using the euclidean distance and the *complete*-linkage method. All methods were run with and without clustering in order to analyse impact of this preprocessing step on the results. The models were built using two different approaches: the Random Forests, introduced in Section III-D, and the Nearest Neighbours algorithm, presented in Section III-B. The average was selected as the baseline method. The models were evaluated with the MASE metric formulated in equation 6.

The experiment was conducted in R in which the LOOCV was applied using the *lazy* [20] package and the Random Forest models were built with the package *randomForest* [21].

V. Results and Discussion

Due to the high number of resources the results were split into two groups. The first group corresponded to the top 12 most frequent resources which covers 67% of total number of reservations made. The second groups contained the remaining 78 resources which included the last 33% of the reservations.

The results were plotted in box plots in order to have a better overall understanding of the performance across the five trials. Additionally, the results obtained with and without model clustering were presented side by side for each method. A dotted line was drawn at value 1 for the MASE result figures to emphasize the characteristic of this metric that results higher or equal to 1 would mean a naive approach would be better than the method being evaluated.

A. Frequently Used Resources

The MASE results for the top 12 most frequented resources are presented in Figure 2. With respect to the results *With Clustering*, its application had a major improvement on the Average method for it consistently lowered the interquartile range of the box plots to below 1 in every trial. This confirms the expectation that averages of data clusters is superior to a single average over the whole training set. However the impact of clustering wasn't as noticeable on the other methods. In the case of NN clustering had a negative effect on trial 1 but otherwise had a positive one on the remaining trials. An interesting observation taken from the *All Trials* plot is that clustering showed to reduce the interquartile range as well as the upper and lower quartiles for the NN. This suggests results from NN models are more consistent when clustering is applied. Contrary to the previous methods, RF doesn't appear to benefit from clustering. In fact, only in trial 2 and 5 did clustering had a small positive impact.

Overall the three methods with clustering were able to achieve better performance than the naive approach expect trial 3, in which the interquartile range of all methods intercepted on MASE = 1. The most likely reason for this is the time of the year of which the validation and test set were chosen for this trial. The validation set of this trial is the first two weeks of June, a time of the year where resources are usually following normal timetables. The test set, however, is the last two weeks of June and the first two weeks of July. Some resources might have started to lower the number of reservations on July which may have lead to a gradual change of frequencies on the test set. Due to this the validation set might not have been the best choice used to select the methods parameters. Another explanation could be related to a change of behaviour observed on this period that didn't occur in previous years.

Regarding the remaining trials, the methods results for the first two trials had fairly similar performances but the Average with clustering didn't performed equivalently to the other three methods for the last two trials. Overall RF had identical or better performance on all trials. The Average with clustering proved to be an alternative in terms of computation speed but it didn't provide as accurate and consistent results as the other methods.

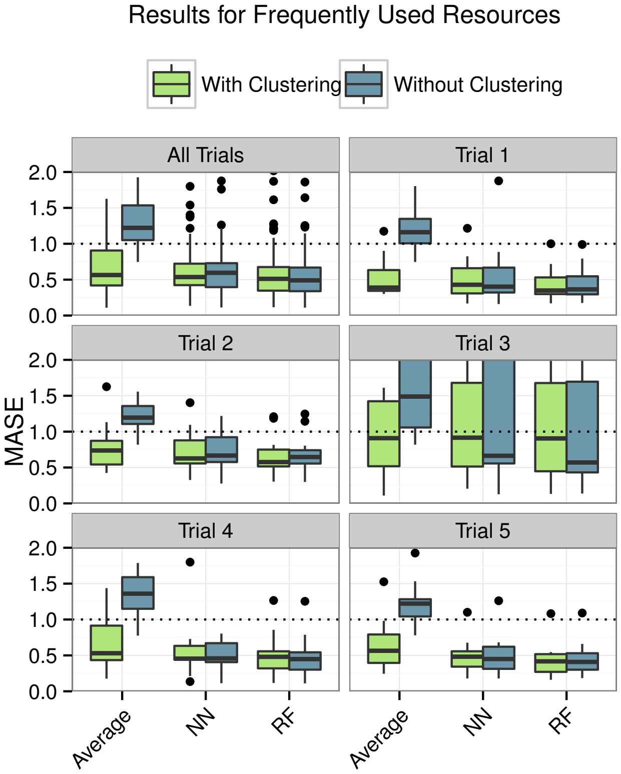

Figure 2. MASE Results for the top 12 most frequented resources. The methods applied were the Average, the Random Forest (RF) and the Nearest Neighbours with the Leave-one Out Cross Validation (NN) criteria. The results are displayed with two box plots per method, one with and another without the usage of clustering. There is a plot that groups the results of all trials as well as an individual plot for each of the five trials.

B. *Less Frequently Used Resources*

The MASE results for the remaining 78 resources are presented in Figure 3. The lowly frequented resources had similar outcomes compared to the previous results. It's evident again that Average benefited the most with the usage of clustering and other methods had a slight improvement on the accuracy. The observation that clustering can reduce the interquartile range on NN is again apparent by inspecting the *All Trials* plot. Contrary to the previous results the RF appears to now benefit more of clustering. However the trials results show that the usage of clustering only benefited RF on trial 1 and 2, although the improvement on trial 1 was significant.

The accuracy of all three methods decreased compared to the highly frequented resources results, as it can be seen by the increase of the interquartile range.The decline of accuracy could be explained by the lack of data for each individually resource and by the randomness factor that might exist on less frequented resources. In the previous results over half of total number of reservations were split between 12 resources. This means that most of the time the resources were in full capacity during the working days. However, in the new results, fewer reservations were distributed for a much higher number of resources. Hence either the variation of the number of reservations is higher through out the year or resources are only active during certain seasons. The latter explanation is likely the cause to why clustering had a higher improvement on the new results.

Although there was an overall decrease of performance the methods with clustering still had, in most part, better performance than a naive approach. The results across the trials had fairly similar results expect for trial 3 as it was also previously observed for the highly frequented resources. As in the previous results, RF showed again to be the best performing method.

VI. Conclusion

This works demonstrated how resource frequency prediction can effectively be predicted using different machine learning approaches. The application of clustering as preprocessing step significantly improved the prediction based on average and added consistency to the NN predictions. Random Forest didn't benefited from clustering but was overall the best performing method. The results of these methods has shown to be more effective on frequently used resources.

Resource frequency prediction is one of the main components needed to build an hospital simulation and optimization tool. Such tool wouldn't only promote patient satisfaction by reducing the patient flow bottlenecks but would also provide financial benefits to hospitals from better resource management. The next steps to building an hospital simulator are the generative modelling of patient visits and patient flow simulation. The former will define how different

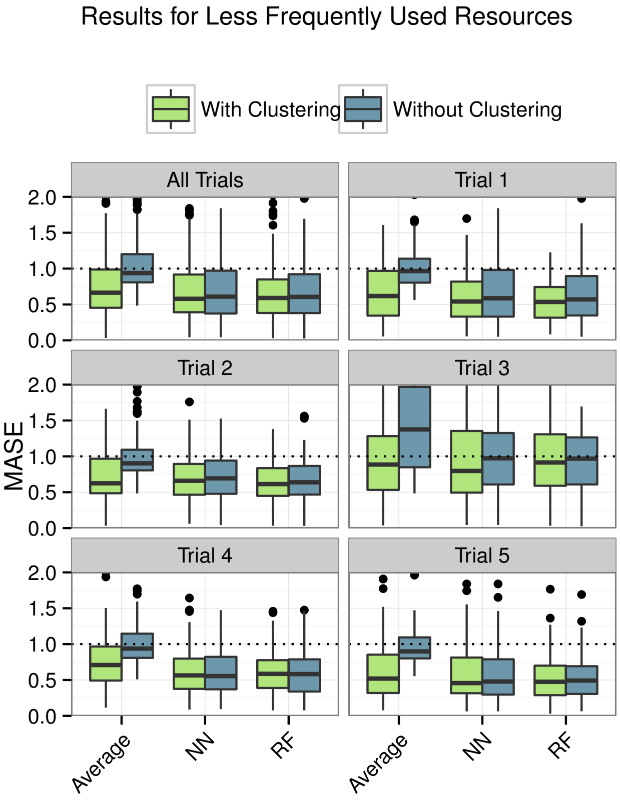

Figure 3. MASE Results for the remaining 78 resources. The results are presented in the same manner as in Figure 2.

reservations are interconnect and the latter will model patient and resource behaviour to generate realistic patient flow that can be used for optimization.

References

[1] R. B. Fetter and J. D. Thompson, "The simulation of hospital systems," *Operations Research*, vol. 13, no. 5, pp. 689–711, 1965.

[2] N. L. McKay and M. E. Deily, "Cost inefficiency and hospital health outcomes," *Health economics*, vol. 17, no. 7, pp. 833–848, 2008.

[3] G. Royston, "One hundred years of operational research in health," *Journal of the Operational Research Society, pages S169–S179*, p. 17, 2009.

[4] D. Fone, S. Hollinghurst, M. Temple, A. Round, N. Lester, A. Weightman, K. Roberts, E. Coyle, G. Bevan, and S. Palmer, "Systematic review of the use and value of computer simulation modelling in population health and health care delivery," *Journal of Public Health*, vol. 25, no. 4, pp. 325–335, 2003.

[5] D. Vieira, "A machine learning approach for resource frequency prediction and generative modeling of visits," Master's thesis, Aalto University School of Science, 2015.

[6] D. H. Hathaway, R. M. Wilson, and E. J. Reichmann, "The shape of the sunspot cycle," *Solar Physics*, vol. 151, no. 1, pp. 177–190, 1994.

[7] L. Cohen, *Time-frequency analysis*. Prentice Hall PTR Englewood Cliffs, NJ:, 1995, vol. 1406.

[8] C. L. Giles, S. Lawrence, and A. C. Tsoi, "Noisy time series prediction using recurrent neural networks and grammatical inference," *Machine learning*, vol. 44, no. 1-2, pp. 161–183, 2001.

[9] E. N. Lorenz, "Atmospheric predictability as revealed by naturally occurring analogues," *Journal of the Atmospheric sciences*, vol. 26, no. 4, pp. 636–646, 1969.

[10] G. Bontempi, S. B. Taieb, and Y.-A. Le Borgne, "Machine learning strategies for time series forecasting," in *Business Intelligence*. Springer, 2013, pp. 62–77.

[11] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical systems and turbulence, Warwick 1980*. Springer, 1981, pp. 366–381.

[12] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 161–168.

[13] T. Cover and P. Hart, "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on*, vol. 13, no. 1, pp. 21–27, 1967.

[14] D. M. Allen, "The relationship between variable selection and data agumentation and a method for prediction," *Technometrics*, vol. 16, no. 1, pp. 125–127, 1974.

[15] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[16] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural computation*, vol. 4, no. 1, pp. 1–58, 1992.

[17] L. Rokach and O. Maimon, "Clustering methods," in *Data mining and knowledge discovery handbook*. Springer, 2005, pp. 321–352.

[18] A. Kalton, P. Langley, K. Wagstaff, and J. Yoo, "Generalized clustering, supervised learning, and data assignment," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001, pp. 299–304.

[19] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International journal of forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[20] M. Birattari and G. Bontempi, *lazy: Lazy Learning for Local Regression*, 2013, r package version 1.2-15. [Online]. Available: http://CRAN.R-project.org/package=lazy

[21] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R News*, vol. 2, no. 3, pp. 18–22, 2002. [Online]. Available: http://CRAN.R-project.org/doc/Rnews/