Machine Learning Project
February 2021 Exam session

Medical Cost Personal Dataset: Analysis and prediction

Team 33: Sanja Stanišić[1]

Abstract

One of the most important activities in health care insurance business (the mission critical application for the insurance companies) is determining insurance coverage and payment plans. Most plans share a few basic similarities - they require insurance subscribers to pay premiums, which are essentially subscription fees assigned monthly, semi-annually or annually. In order to do so, insurance companies have to predict health care costs for each individual.

The goal of this research work is to predict the health care expenses of insurance subscribers using properly trained Machine Learning models and to evaluate the quality of the predictions.

The analyzed dataset is available on Kaggle platform and it contains data relating to 1338 individuals.

Keywords

Machine Learning — Health care — Insurance — Expenses

[1] Università degli Studi di Milano Bicocca, CdLM Data Science

Contents

Introduction

In order to determine health insurance premium for a policy holder, an insurer has to predict the amount of money that a person would spend for health care.

Many factors affect health expenses – a person's age, medical record, type of medical coverage. A more competent decision maker would like to make informed decisions and therefore would be interested in learning about insurance subscriber's lifestyle, his good/bad habits (smoking, overweight, etc.). Other factors, not strictly related to a person's health, like living place, marital status, number of kids etc. also have impact on the individual health expenses.

Hence, most factors that affect the decision on the amount of money a person might spend for healthcare are hardly quantifiable.

Despite the inherent difficulty, it is still worth investigating whether it is possible to predict the amount of individual health expenses starting from some objective and measurable features.

The dataset chosen for answering this research question is the Medical Cost Personal Dataset, available on Kaggle Platform [1]. It has 1338 records each with the following seven attributes:

1. *age* (type: numeric, integer): represents the age of a person, in years.
2. *sex* (type: categorical, binary): represents the gender of a person [female, male].
3. *bmi* (type: numeric, ratio): represents the body mass index of a person - objective index of body weight (kg / m^2) using the weight-to-height ratio, ideally 18.5 to 24.9. According to the [2], a BMI of 30 or above is considered clinically obese.
4. *children* (type: numeric, integer): number of children covered by health insurance of a person (number of dependents).
5. *smoker* (type: categorical, binary): indicate if a person is a smoker or not [yes, no].
6. *region* (type: categorical, nominal): denotes US region where a person lives within US [southwest, northwest, southeast, northeast].
7. *charges* (type: numeric, ratio): personal medical cost billed by health insurance.

The goal of this analysis is to predict the medical charges of a given individual using machine learning tools and to evaluate the quality of the predictions obtained.

This report is organized in sections, as follows:

- Data exploration. In this section, attributes from the dataset are analyzed, focusing on charges amount.
- Preprocessing. In order to make the dataset more suitable for analysis, the missing values are checked and treated (if necessary); dataset attributes are analyzed and transformed.
- Models. This section contains description of different models used in the report to predict the value of the expenses per person.
- Implementation. The implementation of the previously selected models is described as well as the implementation of the whole ML workflow.
- Evaluation. Prediction quality measures are described and the obtained experimental results are analyzed.
- Conclusion. At the end, conclusions are drawn as the final section of the report.

Data exploration

Since the research question is to predict the medical charges of an individual, we consider the *charges* to be the dependent variable in this analysis. This variable represents medical costs an individual incur, which are subsequently billed by health insurance. Therefore, since *charges* is a quantitative variable, regression will be used as prediction method.

The values of *charges* in the dataset under study range from $1122 to $63770. The distribution of this variable is shown in Figure 1.

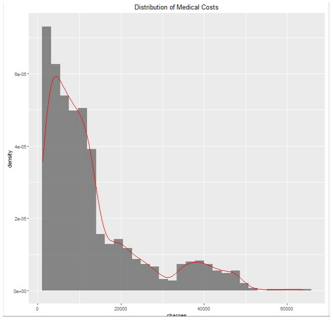


Figure 1

The correlation of each pair of attributes is checked.

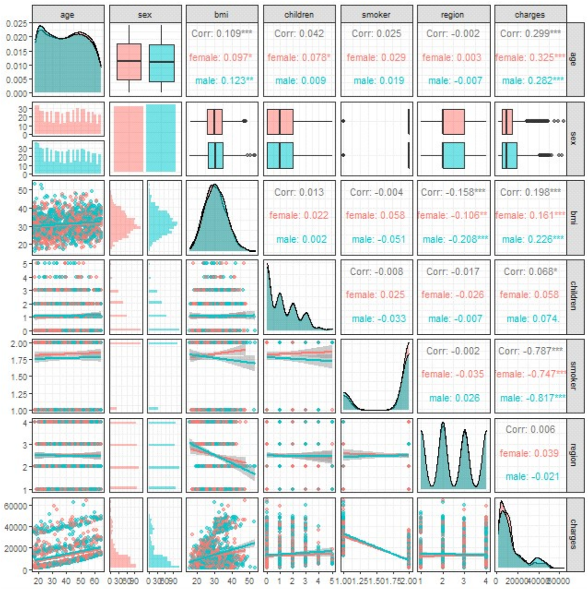


Figure 2

Figure 2 suggests that the attribute *region* is not correlated to any other attribute in this dataset. It can also be seen that the attribute *charges* is highly correlated to the attribute *smoker*. There is a correlation between *charges* and the attributes *age* and *bmi,* but the correlation level significantly lower.

It can be concluded that *charges* depends more on *smoker*, *age* and *bmi* and less on *sex*, *children*, and *region* as shown in scatter plots in Figure 3, Figure 4 and Figure 5.

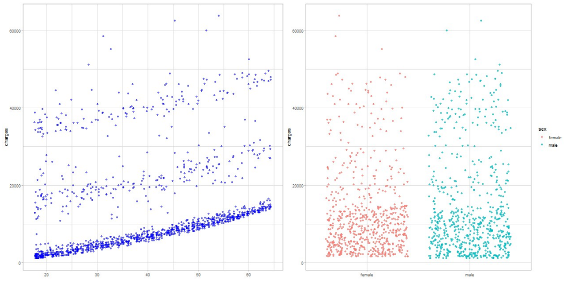


Figure 3

Figure 3 is a graphical representation of the relationship between *charges* and the attributes *age* and *sex*. *Charges* is associated to *age*, and its distribution among male and female individuals is very similar.

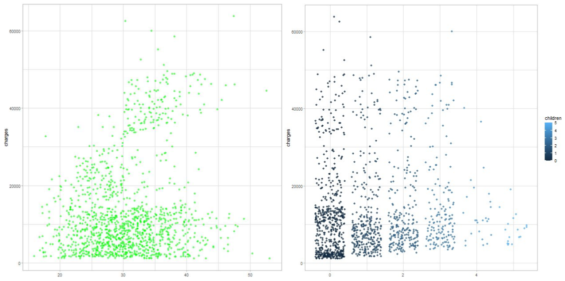


Figure 4

Figure 4 illustrates the relationship between *charges* and the attributes *bmi* and *children*. Intuitively, more dependents i.e. more children should lead to higher charges, but scatter plot in Figure 4 indicate that this proposition doesn't hold completely.

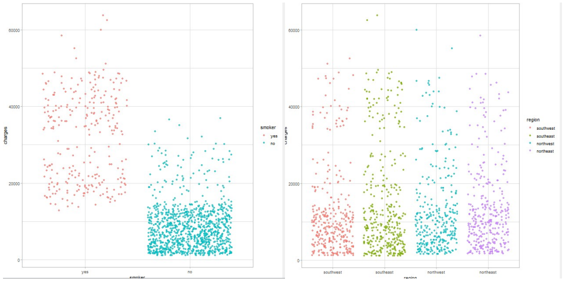


Figure 5

Figure 5 shows the relationship between *charges* and the attributes *smoker* and *region*. It can easily be seen that charges for smokers are higher than charges for non-smokers, while the distribution of charges in various regions looks pretty much alike.

Preprocessing

In order to make the dataset more suitable for prediction (more precisely, for regression), various controls were implemented and several changes were made.

Firstly, the missing values were treated. The analysis has shown that there were no missing values in the Medical Cost Personal Dataset – therefore, there was no need for corrective actions.

Data exploration made evident that the attribute *region* should be removed, and it was done in the preprocessing stage. Therefore, the subsequent analyses and ML models were developed excluding *region*.

Another question has arisen: should the attribute *bmi* be discretized and categorized? There are reports available in open literature, where such transformation (e.g. bucketing *bmi* in categories according to WHO guidelines [3]) is proposed.

However, since the research question is to build a regression model in order to predict the values of a continuous variable *charges*, and since *charges* is correlated to *bmi*, the proposed transformation of *bmi* would cause an information loss. Such a loss might misgovern ML regression process and ultimately lead to worse final result. Therefore it was decided not to perform the proposed bucketing.

Similarly, the explorative analyses from the previous section indicated that *charges* is associated to the attribute *children*, but nature of that dependency is not clear. Bucketing of that attribute (lowering number of categories) would cause information loss, and the importance of the lost information is not known – there is some chance that it could influence the prediction (regression) process. Therefore, it was decided not to modify the attribute *children* during preprocessing.

Models

Various supervised Machine Learning models were chosen for predicting the value of the attribute *charges*. Here is the list of models used in this report:

- Linear Regression (**lm**), described in [4]. In linear regression, the relationships are modeled using linear predictor functions whose unknown model parameters are estimated from the data.
- Polynomial Regression (**poly**), explained in [4]. It is similar to **lm**, but instead of linear it uses of polynomial terms. In this analysis, the polynomials maximum degree to compute was set to 2.
- Simple Regression Tree (**srt**) with structure and characteristics described in [5]. Basically, it is a single decision tree for the regression problem.
- Random Forest (**rf**), also described in [5]. Random forest operates by constructing a multitude of decision trees at training time and outputting the value that is mean/average prediction of the individual trees.
- SVM with Linear Kernel (**svm-l**), described in [6]. Training algorithm of SVM builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.
- SVM with Radial Kernel (**svm-r**), also described in [6]. It is using the kernel trick, which implicitly maps kernel inputs into high-dimensional attribute spaces where attributes are linearly separable. In this case, Gaussian radial basis function $k(\vec{x}_i, \vec{x}_j) = e^{-\gamma\|\vec{x}_i-\vec{x}_j\|^2}$ was used for kernel.
- Bayesian Regular Neural Network (**by-reg-nn**), which follows the algorithms, described in [7]. In this model, a linear combination of Bayesian methods is applied for regularization of the artificial neural network in order to achieve its early stopping.

Some of the models under study are regression models (**lm** and **poly**), some are heuristic models (**srt** and **rf**) and some are separation models (**svm-l**, **svm-r** and **by-reg-nn**).

Implementation

ML process is implemented as Knime Analytics Platform [8] workflow. Surely, some of the models enlisted in the previous section were implemented as Knime nodes.

Due to the fact that the number of available Knime nodes for ML models is much smaller than the number of available ML functions in R, the implemented ML process was not developed solely with Knime.

Therefore R language and environment for statistical computing and graphics [9] was used to implement some parts of the ML process. More precisely, R functions were used for explanatory data analysis, for visualization and for the development of the models that are not executed through Knime nodes.

Both the Machine Learning models implemented using Knime and using R were incorporated within a unique Knime workflow. In order to achieve a successful Knime-R collaboration, extensions had to be installed: “Knime R scripting extension” and ”Knime R statistic integration extension” on the Knime side, and the library ‘Rserve’ [10] within the R system.

Here is the list of the used Knime nodes for ML modelling:

- Knime nodes “Linear Regression Learner” and “Regression Predictor”, based on Weka [11], for implementing lm model.
- Knime nodes “Polynomial Regression Learner” and “Regression Predictor”, based on Weka [11], for implementing poly model.
- Knime nodes “Simple Regression Tree Learner” and “Simple Regression Tree Predictor”, based on Weka [11], for implementing srt model.

Models implemented with Knime were denoted by concatenating name of the model with the string “ k”.

When it comes to the part of the workflow that used R, loading data was implemented via Knime node “R Source Table”, explorative data analysis, data checking and data transformation were implemented with Knime nodes “R Snippet” and “R View”, while R model development was implemented with Knime nodes “R Learner” and “R Predictor”.

The following ML predictor models were developed with R functions:

- Function ‘lm’ [12] in ‘stats’ library was used for lm model implementation.
- Function ‘randomForest’ [13] from the library ‘randomForest’ was used for rf model implementation.
- Function ‘ksvm’ [14] in ‘kernlab’ library [15] with parameter kernel = vanilladot() - which represents linear kernel, was used for svm-l model development.
- Function ‘ksvm’ from the library ‘kernlab’ with parameter kernel = "rbfdot" - which represents radial kernel, was used for svm-r model.
- Function ‘brnn’ [16] in library ‘brnn’ was used for by-reg-nn model.

Models implemented with R were named by concatenating name of the model with string “ r”.

The most important and helpful function in this process was ‘train’ [17] R function from `caret` library [18] – it was used as umbrella function that covered all R functions and libraries needed for the models’ implementation. Thanks to ‘train’ function, it was possible to handle all learning models in a uniform manner. At the moment, the caret library supports “out-of-the box” use of over 230 classification and regression models [19].

During the training phase, parameters for all regression models were determined. For instance, during the training phase, the model “svm-r r” achieved the best results with hyper parameter $\gamma = 0.198110504234301$ and with 368 support vectors, model “svm-l r” achieved the best results with 453 support vectors, model “by-reg-nn r” achieved the best results with Bayesian regularized neural network 5 - 2 - 1 with 14 weights and biases and with normalized connection strengths, inputs and output, etc.

Evaluation

Developed models were compared using k-fold validation [20], with the value of parameter k set to 5. Selected 5-fold validation was realized with Knime nodes “X-Partition” and “X-Aggregator”. In order to achieve exactly the same conditions for comparison among developed ML models, in all 5-fold validation scenarios, random generator was set to a predefined value 1574255294099.

For the purpose of evaluating the quality of the selected ML regression models, various measures [4] were used.

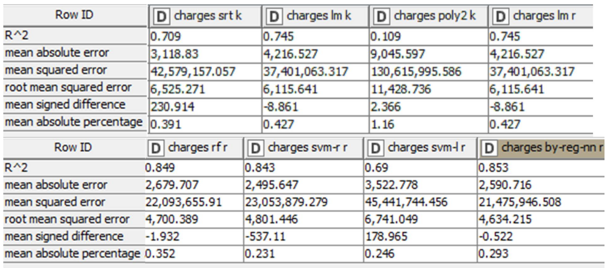
To define those measures, let us denote the actual value with y and the estimated value with $\hat{y}$.

The following overall measures were calculated for all ML models:

1. Mean Absolute Error (MAE), calculated as follows: $MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$
2. Mean Square Error (MSE), calculated as follows: $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
3. Root Mean Square Error (RMSE), calculated as follows: $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
4. Mean Signed Difference (MSD), calculated as follows: $MSD = \frac{1}{n}\sum_{i=1}^{n}\hat{y}_i - y_i$
5. Mean Absolute Percentage Error (MAPE), calculated as follows: $MPAE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$
6. Coefficient of determination (R^2) calculated as follows: $R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$, where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$

Smaller values of measures 1-5 indicate that the prediction model is better. For measure 6, values closer to 1 indicate that prediction model is better.

The following table shows the summary of the results obtained:



Row ID	charges srt k	charges lm k	charges poly2 k	charges lm r
R^2	0.709	0.745	0.109	0.745
mean absolute error	3,118.83	4,216.527	9,045.597	4,216.527
mean squared error	42,579,157.057	37,401,063.317	130,615,995.586	37,401,063.317
root mean squared error	6,525.271	6,115.641	11,428.736	6,115.641
mean signed difference	230.914	-8.861	2.366	-8.861
mean absolute percentage	0.391	0.427	1.16	0.427

Row ID	charges rf r	charges svm-r r	charges svm-l r	charges by-reg-nn r
R^2	0.849	0.843	0.69	0.853
mean absolute error	2,679.707	2,495.647	3,522.778	2,590.716
mean squared error	22,093,655.91	23,053,879.279	45,441,744.456	21,475,946.508
root mean squared error	4,700.389	4,801.446	6,741.049	4,634.215
mean signed difference	-1.932	-537.11	178.965	-0.522
mean absolute percentage	0.352	0.231	0.246	0.293

Table 1

Columns in Table 1 represent the implemented model, while rows indicate the measures which quantify the quality of the specific model.

Firstly, data in Table 1 show that results for implemented models “lm k” and “lm r” are the same – that behavior was expected, because it is basically the same linear regression algorithm implemented in two ways: as the Knime node and as a R script.

More important, data in Table 1 clearly indicate that models “by-reg-nn r”, “rf r” and “svm-r r” are better than other developed models. Therefore, in further discussion, focus will be set on those three predictors and their performances.

With reference to the coefficient of determination, three implemented models “by-reg-nn r”, “rf r” and “svm-r r” have R^2 value 0.853, 0.849 and 0.843 respectively. According to R^2 criteria “by-reg-nn r” is better than the other two models.

On the other hand, in terms of MAE, Table 1 indicate that the mean absolute errors for models “svm-r r”, “by-reg-nn r” and “rf r” are 2495.647, 2590.716 and 2679.707 respectively. According to MAE criteria “svm-r r” is better than two other models.

It is important to calculate the number of best predictions for those three models during cross validation.

Table 2 shows the result of the best prediction count in “head to head” comparison among the selected three predictors.

	rf r	svm-r r	by-reg-nn r
# best predictions	433	651	254

Table 2

It is clear that “svm-r r” model has highest rate of best predictions. In addition, its predictions have the smallest mean absolute error, so it is the candidate for the best prediction model.

However, one issue remains: is the quality of “svm-r r” predictions statistically significant?

To address this issue it is not sufficient to average the absolute error. The count of success in head-to-head comparison can give some insights, but more complete information can be obtained by analyzing the distribution of the absolute errors of prediction.

Therefore the distribution of absolute prediction error for each prediction model is studied and shown in the following figure:

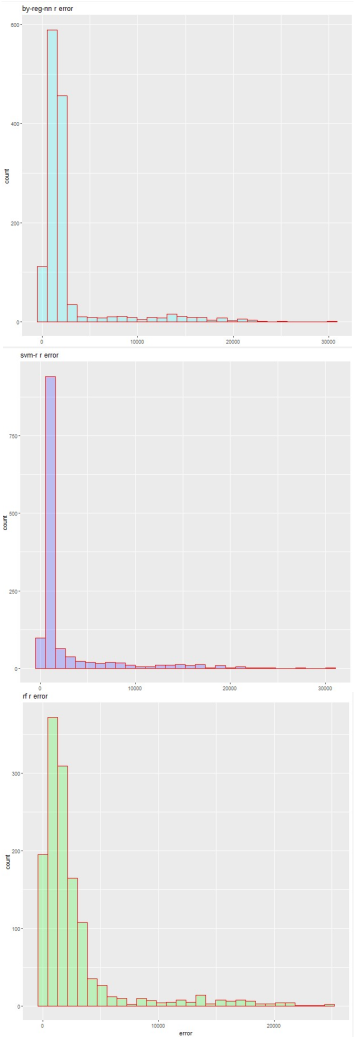


Figure 6

Three diagrams from Figure 6 illustrate the error distribution for models “by-reg-nn r”, “svm-r r” and “rf r” respectively. The shape of the error distribution is similar for all three models.

However, if we align these diagrams on the same scale, as in Figure 7, the shape remains similar, but the values are different. The difference range is high.

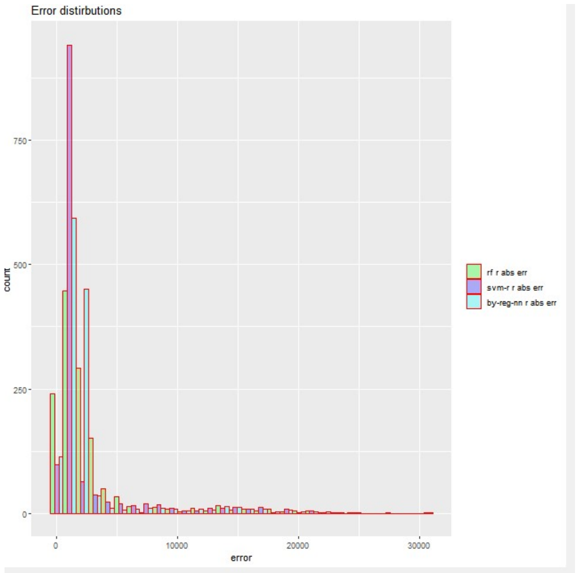


Figure 7.

In spite of the distributions being similar, the definitive conclusions relating to errors cannot be drawn. Therefore, a nonparametric two sided Kolmogorov-Smirnov test [21] is used, with the null hypothesis stating that “data samples representing the absolute prediction errors come from the same distribution”. The obtained test results of all three tests have p-values less than 2.2e-16, which is far below $\alpha = 0.005$. Therefore, it can be concluded that the error distribution is the same for three studied models.

Finally, the difference among those errors was studied. Paired t test [22] was applied, using R function t.test [23] with null hypothesis being that "the paired population means are equal" and the alternate hypothesis being that “true difference in means is not equal to 0”.

The results of all three tests are shown in Table 3.

	95% confidence interval	t	p-value
“svm-r r” vs. “by-reg-nn r”	[-154.94111, -19.79372]	-2.5364	0.01131
“rf r” vs. “by-reg-nn r”	[31.5866, 189.6624]	2.7457	0.006119
“rf r” vs. “svm-r r”	[104.2028, 291.7810]	4.1413	3.67e-05

Table 3

It is clear that in all three tests the confidence interval doesn’t span over 0, so (with 95% certainty) it can be concluded that the null hypothesis cannot be accepted and that the difference between the absolute errors for each pair of models is statistically significant.

The sign of the obtained test statistics and the interval (smaller values indicate lower errors and better model) indicate that the model “svm-r r” is significantly better than both “by-reg-nn r” and “rf r”, and that the model “by-reg-nn r” is better than “rf r”.

However, in practice, any of those three regression models can serve as good medical charges predictor.

Conclusion

The goal of this report was to predict the personal health care charges, starting from some measurable attributes available in the Medical Cost Personal Dataset.

In order to do so, six ML regression models were proposed and its implementations were compared, under the same conditions.

Models based on Bayesian Regular Neural Network, Support Vector Machine with Radial Kernel and Random Forest were significantly better than the others. Among these three models, Support Vector Machine with Radial Kernel had the best prediction results, but with a very small margin.

However, we should take care not to jump too early to general conclusions. In other words, as stated in [24], the results obtained indicate that the proposed ML model outperformed the others for this specific prediction problem, but it is very likely that the same ML model would be inadequate for some other problem. Formal mathematical proof of this statement, widely known as No free lunch theorem for supervised learning, is given in paper [25].

References

[1] Kaggle, "Medical Cost Personal Dataset," 05 January 2021. [Online]. Available: https://www.kaggle.com/mirichoi0218/insurance.

[2] Cenetrs for Discease Control and Prevention, "About Adult BMI," [Online]. Available: https://www.cdc.gov/healthyweight/assessing/bmi/adult_bmi/index.html. [Accessed 5 01 2021].

[3] World Health Organization, "Body mass index - BMI," [Online]. Available: https://www.euro.who.int/en/health-topics/disease-prevention/nutrition/a-healthy-lifestyle/body-mass-index-bmi. [Accessed 5 1 2021].

[4] K. P. Murphy, "Linear regression," in *Machine Learning - A Probabilistic Perspective*, Cambridge,

Massachusetts, MIT Press, 2012, pp. 217-243.

[5] K. P. Murphy, "Classification and regression trees (CART)," in *Machine Learning - A Probabilistic Perspective*, Cambridge, Massachusetts, MIT Press, 2012, pp. 544-552.

[6] K. P. Murphy, "Kernels," in *Machine Learning - A Probabilistic Perspective*, Cambridge, Massachusetts, MIT Press, 2012, pp. 479-513.

[7] F. Burden and D. Winkler, "Bayesian Regularization of Neural Networks," *Artificial Neural Networks. Methods in Molecular Biology ,* vol. 458, pp. 23-42, 2008.

[8] KNIME, "KNIME Analytics Platform," [Online]. Available: https://www.knime.com/knime-analytics-platform. [Accessed 5 1 2021].

[9] R Foundations, "The R Project for Statistical Computing," [Online]. Available: https://www.r-project.org/. [Accessed 5 1 2021].

[10] R Foundations, "Rserve - Binary R server," [Online]. Available: https://www.rforge.net/Rserve/. [Accessed 5 1 2021].

[11] University of Waikato, "WEKA The workbench for machine learning," [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/. [Accessed 5 1 2021].

[12] R Documentation, "lm," [Online]. Available: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/lm. [Accessed 5 1 2021].

[13] R Documentation, "randomForest," [Online]. Available: https://www.rdocumentation.org/packages/randomForest/versions/4.6-14/topics/randomForest. [Accessed 5 1 2021].

[14] R Documentation, "ksvm," [Online]. Available: https://www.rdocumentation.org/packages/kernlab/versions/0.9-29/topics/ksvm. [Accessed 5 1 2021].

[15] A. Karatzoglou, A. Smola, K. Hornik, NICTA, M. A. Maniscalco and C. Hui Teo, "Package ‘kernlab’," 12 11 2019. [Online]. Available: https://cran.r-project.org/web/packages/kernlab/kernlab.pdf. [Accessed 5 1 2021].

[16] P. Perez Rodriguez and D. Gianola, "Package ‘brnn’," [Online]. Available: https://cran.r-project.org/web/packages/brnn/brnn.pdf. [Accessed 5 1 2021].

[17] R Documentation, "train," [Online]. Available: https://www.rdocumentation.org/packages/caret/versions/4.47/topics/train. [Accessed 5 1 2021].

[18] M. Kuhn, "The caret Package," [Online]. Available: https://topepo.github.io/caret/. [Accessed 5 1 2021].

[19] M. Kuhn, "Available Models," [Online]. Available: https://topepo.github.io/caret/available-models.html. [Accessed 5 1 2021].

[20] K. P. Murphy, "Empirical Risk Minimization," in *Machine Learning - A Probabilistic Perspective*, Cambridge, Massachusetts, MIT Press, 2012, pp. 201-210.

[21] A. Garth, "Nonparametric Tests," in *Analysing data using SPSS*, Sheffield Hallam University, 2008, p. 66.

[22] G. Cicchitelli , "Confronti tra due populazioni," in *Statistica: Principe e metodi*, Pearson, 2012, pp. 439-460.

[23] R Documentation, "t.test," [Online]. Available: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/t.test. [Accessed 05 01 2021].

[24] K. P. Murphy, "No free lunch theorem," in *Machine Learning - A Probabilistic Perspective*, Cambridge, Massachusetts, MIT Press, 2012, pp. 24-25.

[25] D. Walpert, "The Lack of A Priori Distinctions Between Learning Algorithms," *Neural COmputation,* vol. 8, no. 7, pp. 1341-1390, 1996.