Implementation and Evaluation of a Case Study Using Machine Learning Techniques

CMT307: APPLIED MACHINE LEARNING

CARDIFF UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND INFORMATICS

CARDIFF UNIVERSITY

Author:

Georgios K. Psevdiotis

C1841824

Table of Contents

Table of Figures & Contents

Data Exploration

In order to fully grasp the nature of the data's characteristics, data exploration is the first step that must be carried out, which is one of the most crucial processes and requires the most work overload. It involves the use of data visualization tools and techniques to help unravel patterns, features, similarities/differences, and relationships between the data. At first, essential libraries are imported into python that will provide functionality and flexibility.

Moving on, to begin this machine learning project, the dataset CSV file is input and loaded through the Pandas library as a data frame and basic information about the dataset is explored such as its shape, information such as datatypes, column names. It is noticed that the dataset includes two different Data Features which are classified into two types, Categorical and Numerical Features. The Categorical Features can only take nominal, ordinary or binary values while Numerical Features represent quantitative interval or ratio scaled data.

A few columns were dropped from the dataset as they do not contain data that would provide value and would not assist in carrying the case study's purpose. The next step is to find and most importantly, deal with missing data of the dataset. As it's observed, the dataset contained a large amount of missing data and therefore it was decided to preserve all the cases by using imputation to handle the missing data. The method of imputation used was to replace all the missing data for both categorical and numerical features by using the most frequent/mode value for each column. In my opinion this option suits the particular dataset as it works for both features and because it scales well for large datasets such as the one we were provided.

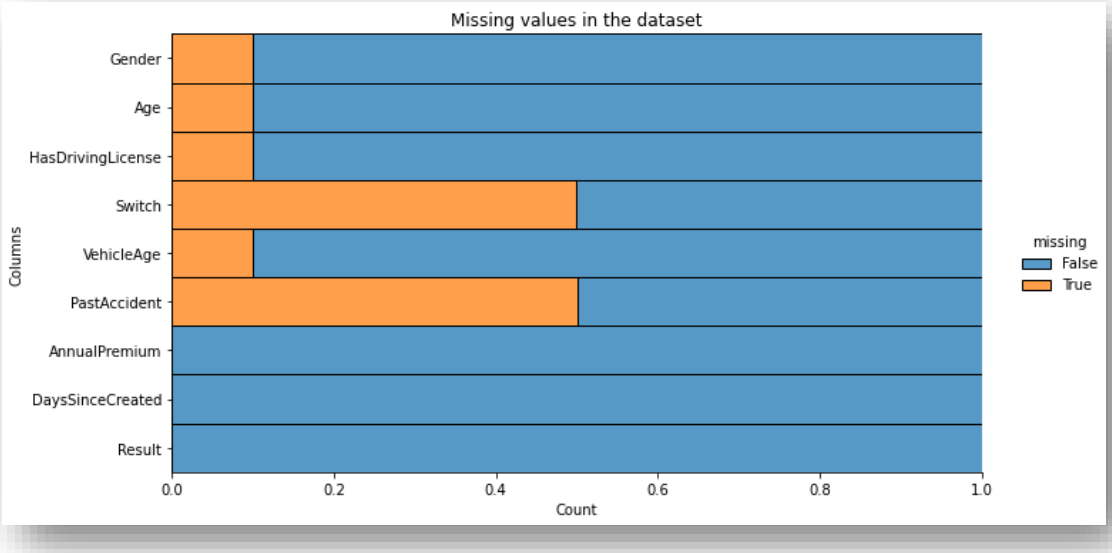

Figure 1: Graph showing missing values per column in the dataset

After the previous, to acquire a clear image of the data, the qualities of it and the link between them, different visualization techniques were used. At first, a histogram was drawn for every value per column in the dataset. This allowed the understanding of how the data is distributed and summarized statistical values such as center-point of data. Additionally, a pair plot was drawn to show how data are scattered against every possible pair of features. From here cluster points were noticeable and outliers were observed. Moreover, to get a better look, a boxplot was created for every numerical feature in order to review the distribution of each attribute. As this is univariate, the dots outside the whiskers have pointed out the outlier values of columns. As the outliers lied above the third quartile, this meant that the outliers need a way to be treated and the decision was to replace them with the median value by calculating the quartiles and replacing values outside them.

Furthermore, a Feature Correlation Map was implemented to indicate how data features are correlated with regards to the "Result" and give a clear idea about the degree of the relationship. The Map was drawn by using a seaborn heatmap and it showed that there was no strong negative correlation in a relationship therefore no columns were dropped. Lastly, Class Imbalance is a common problem in machine learning, especially when it comes to classification problems. When one class' observation is higher than the other classes' observations, class imbalance exists, and this can affect a model's accuracy. After evaluating the model, it was discovered that there is a Class Imbalance in favour of Result's '0' (False) of 87.8% to 12.2%.

Data pre-processing

In our case, the data could not be immediately used to train a machine learning model because they firstly needed to be cleaned and encoded. The cleaning part was carried out in the previous step therefore on this step, the actions needed to be taken were to encode both features and train the machine learning model. As this would improve the model's precision and effectiveness, CountVectoriser was used to encode categorical features while Normalizer was used to encode numerical features. Following, the dataset was split into a test set which is kept aside and would evaluate the predictions at a later stage, and a training set, which is a subset of the dataset that is will be fed into the machine learning model to uncover, learn patterns, and generate predictions.

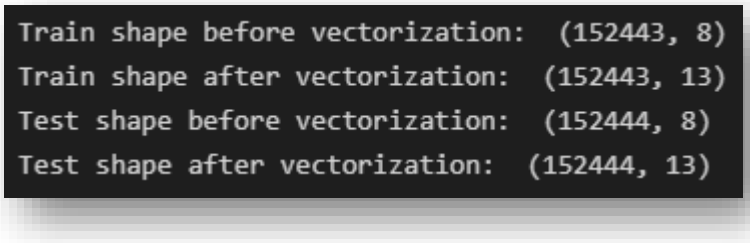

Figure 2: Shapes of the Train and Test model before and after vectorisation

Model implementation

To develop my Machine Learning model, I have decided to use three classification methods, the Decision Tree, the Linear SVC, and the Gradient Boosting algorithms. The Decision Tree algorithm provides a graphical representation for obtaining all feasible answers to a problem. The algorithm categorizes given trained and tested data and creates predictions based on how a previous set of questions were answered. In this way, it follows a tree-like structure, and it replicate human thinking abilities in a logical way as it calculates different possibilities to create predictions. The Linear SVC was used to produce a "best fit" hyperplane by dividing and categorizing the trained and test data. Moreover, the predictions obtained were observed and proved how the particular algorithm was a suitable model. The Gradient Boosting algorithm works by combining previously models to integrate and minimize prediction errors and produce a more precise model. It can optimize on different loss functions and can also provide hyperparameter tuning options thus more reasons to make the function fit in a flexible manner.

Performance evaluation

Performance matrices can be evaluated in a variety of ways. The AUC (Area under the ROC Curve) and ROC (Receiver Operating Characteristic) were implemented for this model as they are two of the most important evaluation metrics to evaluate a model's performance. The AUC was used to assess how well the model performed on the test data by using the previously implemented techniques. It uses a statistic metric for assessing how well a categorization problem performs at various thresholds. The ROC (Receiver Operating Characteristic Curve) was used to evaluate diagnostic tests and predictive models. It generates two curves, the TPR (True Positive Rate) and FPR (False Positive Rate) by choosing a set of threshold values from a list of probability scores of the backward sorted model.

Result analysis and discussion

As a result, in this coursework, a machine learning classification model was created that predicts whether a customer will buy car insurance based on the customer's data value. I have used three different classification strategies to implement the model and also two different measures to evaluate the performance of them. The results have shown that the Gradient Boosting Classifier achieved the most accuracy at an 80% while the Decision Tree was close at 79% and the LinearSVC at 75%.

Table 1: Table of Models, Hyper Parameters and AUC scores

Model	Hyper Parameter	AUC
Decision Tree	10-500	0.79
LinearSVC	5.648	0.75
GradientBoostingClassifier	100-0.2	0.8