BLG 454E Learning From Data Fall 2022 Term Project Report

Eyüpcan Çakır
Department of Computer Engineering
Istanbul Technical University
Istanbul, Turkey
cakirey19@itu.edu.tr

Furkan Salık
Department of Computer Engineering
Istanbul Technical University
Istanbul, Turkey
salik20@itu.edu.tr

Nour Elhouda Znagui
Department of Computer Engineering
Istanbul Technical University
Istanbul, Turkey
znagui19@itu.edu.tr

***Abstract*—As the technological landscape is continuously evolving, it is offering more tools for both industries and consumers to use. In fact in the insurance industry, both the customer and the insurer can exploit data analytics to obtain valuable information and insight. This paper aims to predict the individual price of insurance by using machine learning tools based on a list of factors.**

***Index Terms*—mean square error, preprocess, regression, ridge regression, support vector regression, gradient boosting regression, random forest regression, decision tree regression**

I. Introduction

In this project, our aim was to predict the cost of insurance for an individual based on various features, such as age, body mass index, number of children, and whether the person smokes or not. To make predictions we tried different machine learning models, which were regression algorithms, and we calculated the mean square error for each model and compared them. We pipelined the best-fit models and then we applied data preprocessing to get better results.

A. Kaggle Results

Team Name:	150200056_150190002_150190916
Team Members:	eypcnkr(Eyüpcan Çakır), sfurkan20(Furkan Salık), nourzng(Nour Elhouda Znagui)
Rank:	1 (Date: 19.12.2022-23.10)
Score:	19413178.75567

II. Datasets

Our main approaches in data preprocessing were to fix the outlier values to fit better within the features of the dataset, drop unnecessary features to avoid misleading models, add an extra feature extracted from the dataset, and convert categorical values to numerical values.

A. Fixing the Outliers

To spot the outliers, looking at the scatter plots would be a perfect idea. As can be clearly seen from the figures 2, 3, there are outliers that would affect our model's accuracy negatively 4. We introduced a rule-based solution for this problem to cap the target values (charges) to appropriate values. Our transformation function for the charges column can be seen in the figure 1.

```
for index, row in X.iterrows():
    if row["age"] <= 23 and row["smoker"] == "no":
        y.iloc[index] = min(y.iloc[index], 5000)
    else:
        y.iloc[index] = min(y.iloc[index], 48000)
```

Fig. 1. Outlier transformation function

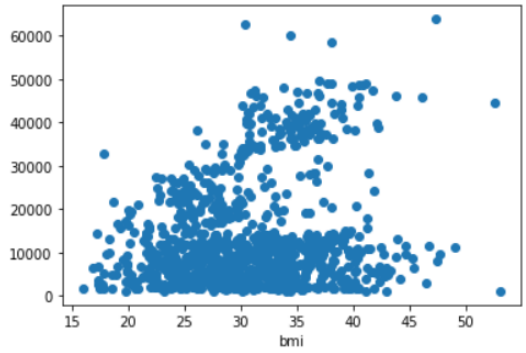

Fig. 2. BMI - charge scatter

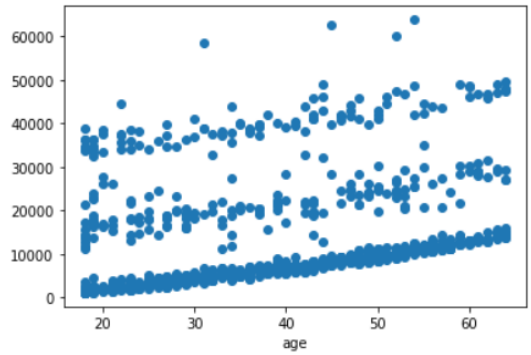

Fig. 3. Age - charge scatter

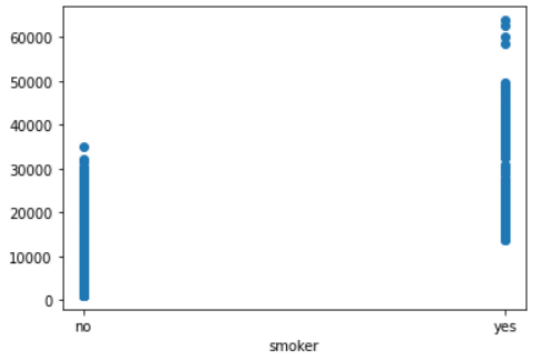

Fig. 4. Smoker - charge scatter

B. Dropping Unnecessary Features

2 of the features (region and sex) had no significance for charge values. Again, we are able to spot this from the scatter plots, which are in the figures 5 and 6. Therefore we dropped them.

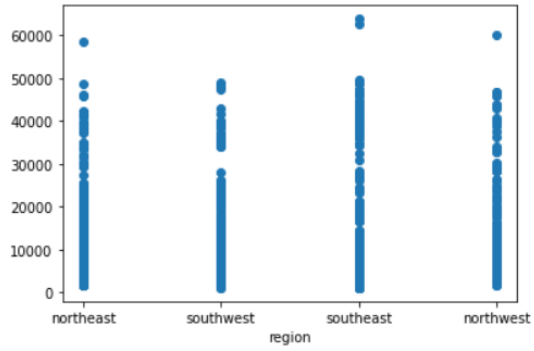

Fig. 5. Region - charge scatter

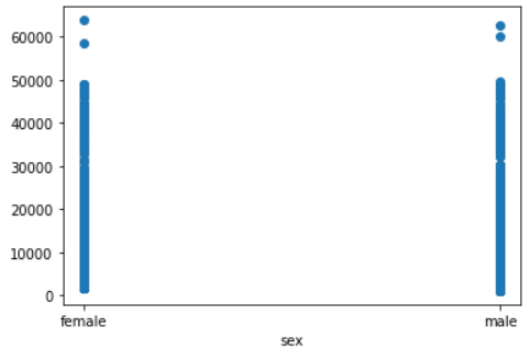

Fig. 6. Sex - charge scatter

C. Adding Obese Label

It can be deduced from figure 2 that the combination of BMI and smoker features creates a huge difference (there are 2 different distributions in the plot). To make our model able to explain this interaction, we should add a label that indicates whether the person is obese ($bmi >= 30$). This is because, as it will be explained shortly, our model's polynomial degree is 2 (BMI and smoker features are multiplied after the polynomial feature transform).

D. Converting Categorical Features to Numerical

Since machine learning algorithms cannot work by categorical values such as 'yes' and 'no', they must be converted into numerical values (1 for 'yes', and 0 for 'no'). This is only necessary for the smoker column. We specifically transformed 'yes' to 1, and 'no' to 0, since the smoker feature being 'yes' would increase the charges.

III. Methods

A. The How

After preprocessing, we raised the degree of the polynomial to 2. This introduced new columns to the data frame, which are the multiplication of the columns (for example, resulting columns are (age, age^2, $age * bmi$, $age * children$, $age * smoker$, bmi, bmi^2, $bmi * children$...). After the polynomial transform, a standard scaler is introduced. For each feature, this component removes the mean and scales to unit variance. After that, a Ridge regressor [1] is used to produce predictions. We developed the model by following a 5-fold cross-validation approach. In each fold, we split the data into training and testing datasets, obtained a mean square error, and took the mean of the errors of the folds, which was so useful while tuning the hyperparameters.

B. The Why

The polynomial degree raised to 2 increased the accuracy of our model drastically since it enabled the model to also learn the interactions between the features ($bmi * smoker$ being the most prominent one). A standard scaler is introduced because Ridge regression utilizes regularization to make the model more general and reduce overfitting. For regularization, scaling is important since units of the features may be different and penalization of coefficients might not be fair for this reason. That being said, Ridge regression yielded the best results among our trials and we chose it to avoid overfitting. For hyperparameter tuning, we used $sklearn.GridSearchCV$ [2] to automatically find the best hyperparameters for the model.

IV. Results and Conclusions

A. 5-fold Cross-validation Results

Ridge Regression	15600478.35673672
Random Forest Regression	16794894.66375887
Support Vector Regression	17268492.78553814
Gradient Boosting Regression	18231966.2776039
Decision Tree Regression	18726295.628325082

B. Kaggle Results

Rank: 1 (Date: 19.12.2022-23.10)
Score: 19413178.75567

REFERENCES

[1] medium.org, 'ridge-regression'
Available: https://medium.com/analytics-vidhya/ridge-regression-regularization-fundamentals-cc631ba37b1a

[2] scikit-learn.org, 'sklearn.GridSearchCV'
Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

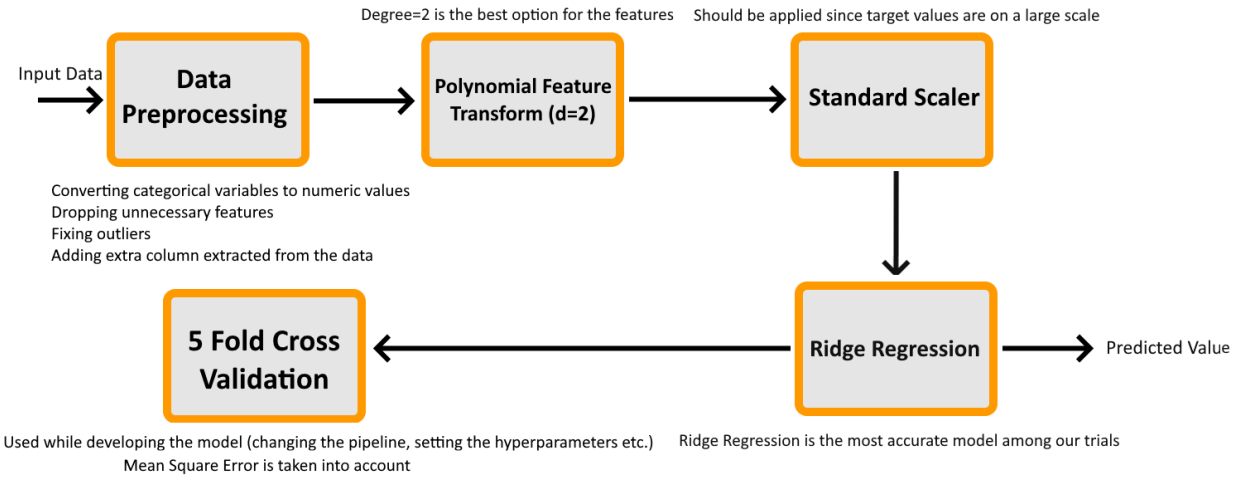

Fig. 7. Learning Diagram