1. Dataset Description

After importing our data, we noticed that they contained 10,000 rows and 55 variables. Although there were not any duplicate observations in the data, there were 32,813 missing values (i.e. 6% of the dataset). We noticed that these values derived from 8 variables, each one with different percentage of missing values as showcased below.

emp_length	8.17
debt_to_income	0.24
annual_income_joint	85.05
debt_to_income_joint	85.05
months_since_last_delinq	56.58
months_since_90d_late	77.15
months_since_last_credit_inquiry	12.71
num_accounts_120d_past_due	3.18

We decided to follow a simplistic approach when handling the aforementioned missing values in our data. Specifically, we removed the variables with higher than 50% missing values and for the rest we replaced their value with the median of their respective columns. Another approach we could have used to address this problem, especially for the variables with a high percentage of missing values, would be to run a model on top of non-missing values and predict the missing values in that respective column. An exception was made for the "num_accounts_120d_past_due" variable, which apart from "NA" it only had one value (i.e. zero). Thus, we removed this variable from our data as it would not be proved helpful to our analysis.

In addition, we had two variables carrying the same information at the most part, namely "verified_income", "verification_income_joint". Regarding the latter variable, we noticed that 85% of its values were missing, thus we removed it from our data. Furthermore, from the sub-grade variable, one of its values (i.e. "G4") only occurred once, so we decided to remove it in order to avoid facing any problems when splitting are data for predicting the interest rates (e.g. having this value exist only in the test training set). So, we continued our analysis with 49 out of the total 55 variables. From them, 12 were factors, 10 were numeric and 27 were integer type variables. Our goal was to predict the interest rate, which had a mean equal to 12.43 and ranged from 5.31 to 30.79.

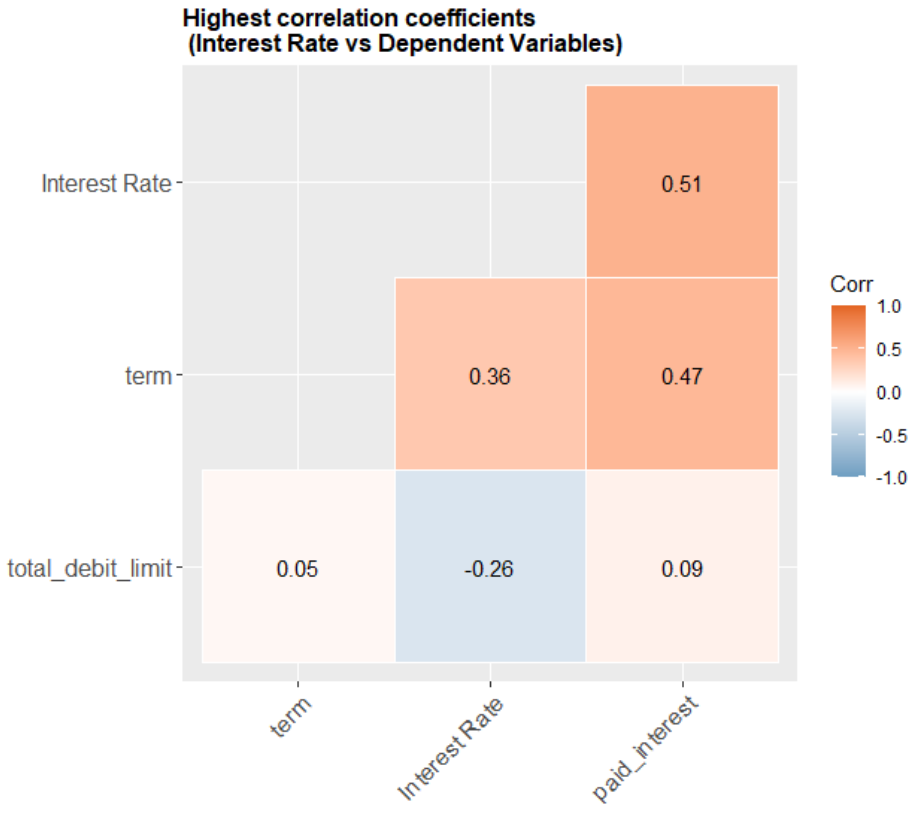
2. Visualizations

Exploratory Data Analysis (EDA) is a method of analyzing datasets to summarize their main characteristics, often using visual methods, used before the modeling task. There are two ways to categorize exploratory data analysis. Firstly, each method can be either non-graphical or graphical and secondly each method can be either univariate or multivariate (usually bivariate analysis).

2.1. Correlations

We began by computing the Pearson's correlation coefficients and we created correlation plots between our dependent variable and the continuous variables to investigate if there were any associations implied by the dataset. The rationale behind exploring the correlations lies in the fact that many methods perform better when highly correlated are removed. For better distinguishing the features that had stronger correlation with the interest rate, we sorted the computed correlations and our assessment of the strength of correlation coefficients was based on empirical correlation values.

In general, we observed that there were some variables that showed higher influence towards the interest rate. In the figure below, we presented the correlation coefficients between our target variable and the attributes that showed the higher Pearson's correlation coefficient values.



2.2. Contingency Tables

For the independent categorical variables we decided to create some contingency tables in order to examine their relationships with our target variable. Our conclusions from these tables were based on chi-square test. Below, we showcase a table containing the ones where we could not reject the null hypothesis along with an example of the contingency table regarding two of the associated variables.

Variable Name	**Associated Variable Name**
issue_month	homeownership, loan_purpose, application_type, grade, sub_grade
initial_listing_status	verified_income, application_type, loan_status
homeownership	disbursement_method
application_type	loan_status

verified_income	*initial_listing_status* fractional	whole	***Total***
Not Verified	644	2950	3594
Source Verified	727	3389	4116
Verified	423	1866	2289
Total	1794	8205	9999

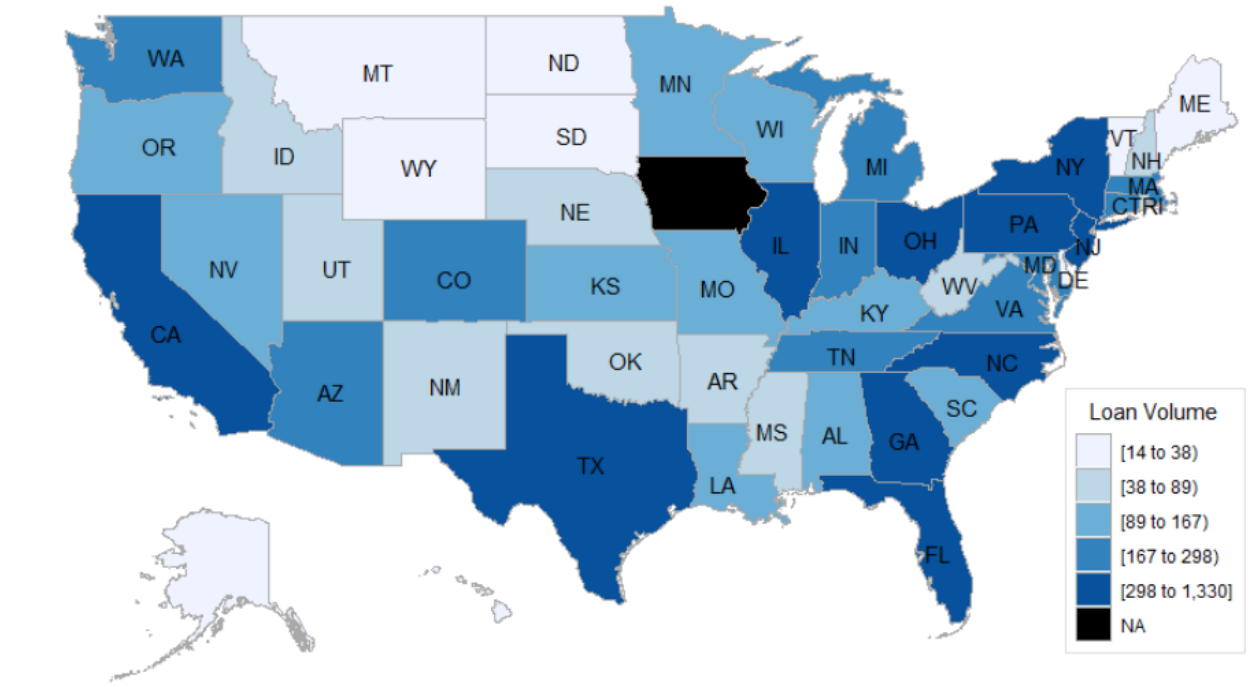
χ^2=0.669 · df=2 · Cramer's V=0.008 · p=0.716

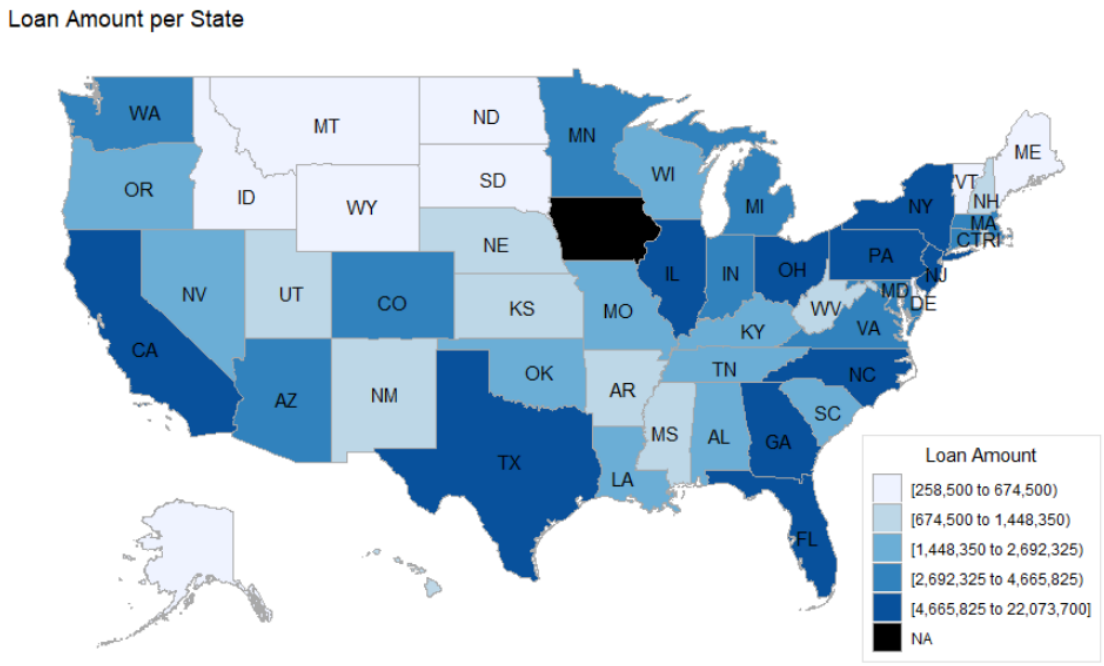
2.3. Other Visualizations

In order to exploit the fact that our data contained information regarding the U.S. states, we decided to illustrate some information using maps. Considering that thanks to our data we only had in our disposal the state abbreviations, we used them to create a variable depicting the actual state names. Our rationale was to create charts based on the geographic distribution of issued loans across various states, as they could be proven of pivotal importance by providing information relative to the markets of Lending Club's current business as well as to the location of potential customers.

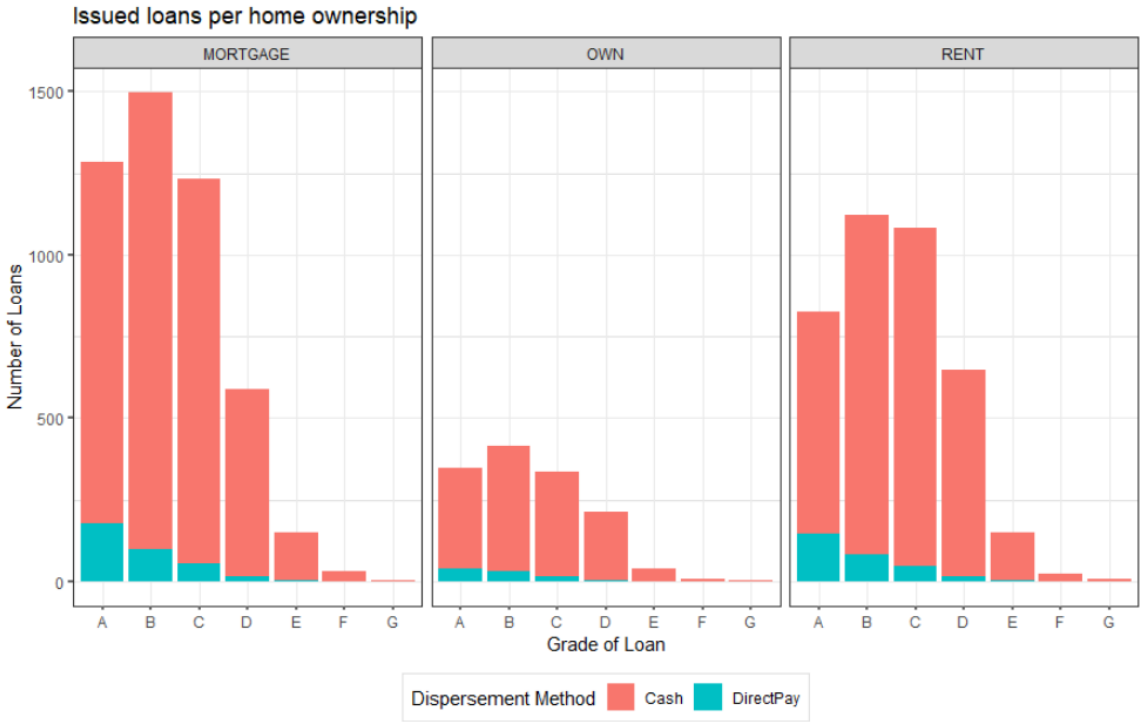
So, we created some U.S. maps illustrating the loan volume per state and the loan amount per state, as showcased below. Unfortunately, from our data information regarding the state of Iowa was missing, which was depicted in the maps as well. We noticed that the states of California, Texas, Florida and New York seemed to be among the highest borrowers, while also having the largest dollar amounts.

Loan Volume per State

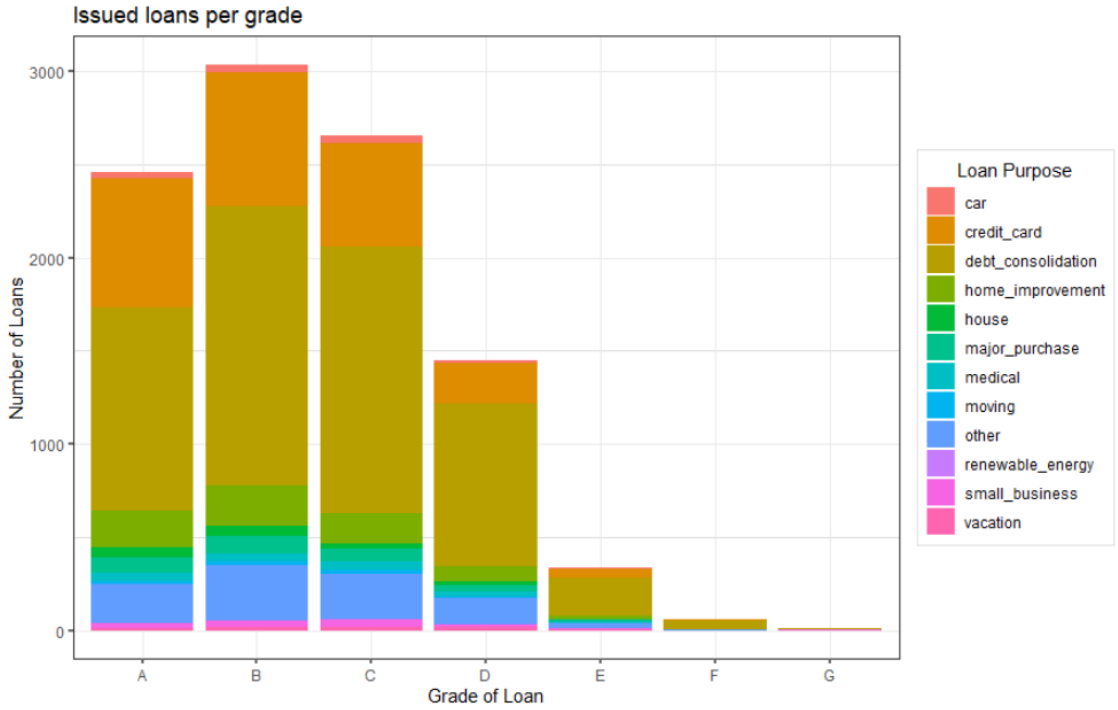




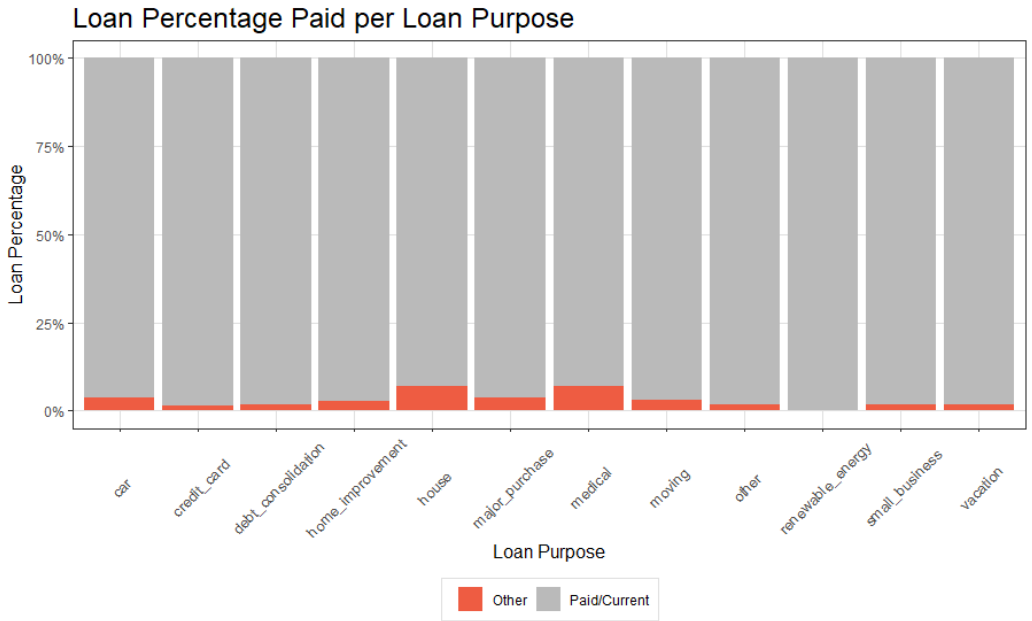
In addition, we analyzed the grade associated with the loan based on the ownership status of the applicant's residence, while also taking into account the disbursement method of the loan. From the figure below, we noticed that people involved in mortage loans or housing loans have much more demands of borrowing money compared to living in their own house. Of course, this was to be expected as typically people who own a house usually have better financial situation than others. Also, we noticed that as we moving from grade A to grade G type of loan, the number of direct payments decreases.



We also analyzed the grade associated with the loan based on the purpose for which the loan was taken. From the figure below, we observed that the most general purpose for which loans were taken was debt consideration, followed by credit card payments.



Furthermore, we decided to examine the purposes for taking a loan once again, but this time to see the percentage of loan paid. This would enable us to see the probability of a loan remain unpaid based on the purpose for which it was taken. From the figure below, we noticed that loans taken for housing and medical purposes had the higher probability of being unpaid.



3. Methodology

The target of our analysis was to predict the interest rate, so we performed a regression analysis. To do so, we used two different algorithms, namely linear regression and regression trees. At first, we begun by spitting our data into training and test subsets using a 70%-30% ratio without replacement. So, we randomly selected 70% (c. 7,000 observations) of the data as the training set in order to fit the model and used the rest 30% of them (3,000 observations) as the test set to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper-parameters.

3.1. Linear Probability model

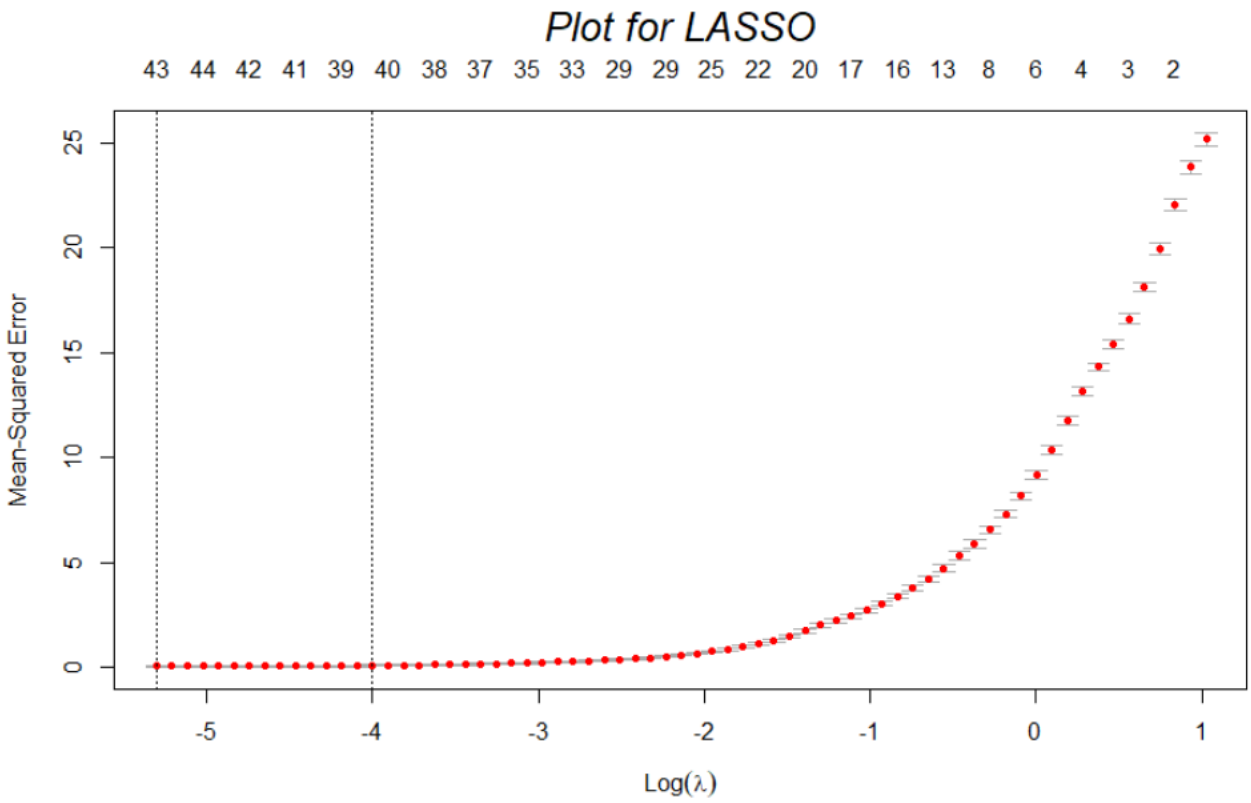
Regarding the linear regression model, after fitting the model in the training set, we did not observed any "NA" values in the variable coefficients, so we proceeded as is. The existence of such values as a coefficient in a regression model would indicate that the variables in question were linearly related to other variables. So, these coefficients would not add any new information to our model, thus we had to remove them.

Considering that we had number of our covariates was slightly high, we decided to use Least Absolute Shrinkage and Selection Operator (or LASSO). This would allow us to remove all of the truly "bad" covariates and to continue our analysis with the reduced space. The rationale behind LASSO, is that since the overall magnitude of the coefficients is constrained, the important predictors would be included in the model and the less important predictors would shrink (potentially to zero). In fact, LASSO for the regression approach tries to minimize a specific quantity.

The tuning parameter lambda (λ) controls the amount of regularization and for a large enough λ, some coefficients become exactly equal to 0. Large enough values of lambda used in lasso will set some coefficients exactly equal to zero. Solutions proposed by k–fold Cross Validation indicate that Lasso suggests over-fitted models. So, lasso does not exactly do variables selection, but rather data screening. This has proven very useful for performing an initial cleaning when there are a lot of variables in the dataset, as we are able to clear all irrelevant variables very fast.

Before continuing with the lasso implementation, we also separated the intercept from our data, beside from the response variable. Then, we used 10-fold cross validation on the training set, where we split the data in 10 groups (aka folds) and fitted them into 9 of these folds, while the remaining fold was used for testing the data. This procedure was repeated for all possible test folds. The reason why we did this, was to identify which was the optimal 'lambda' value for using in lasso. In order to assess the goodness of the chosen model, the MSE (Mean Square Error) for Tk fold was used.

After implementing Lasso, we observed that the proposed non-zero coefficients were 43 based on the minimum lambda and 40 according to lambda with one standard deviation. From the figure below, we noticed that whether we used the value for the minimum lambda or for the lambda with one standard deviation, the results for the Mean Squared Error (MSE) will (approximately) be the same. So, we preferred using the results suggested according to lambda with 1 standard error, because we had essentially the same MSE but with fewer variables, which would help in simplifying our analysis. As a result, after implementing LASSO we kept 12 out of the total 48 independent variables.



Afterwards, in order to identify the most important variables for predicting the interest rates, we used a stepwise procedure, but we could have also used backward or forward procedures. This procedure was used to allow us to remove variables from our model that were not statistically significant. Considering that we wanted to create a model for prediction purposes, we decided to implement the stepwise method according to Akaike's information criterion (AIC).

The goal of these stepwise methods is to find the predictors which would lead to the minimization of AIC (technically, the <none> model should have the minimum AIC value so that no other variables need to be added or removed from the model). So, after implementing this procedure we noticed that 8 more variables were removed from our model. The model selected from the stepwise procedures is not the best, but generally is considered a good model.

Furthermore, we checked whether our regression model faced a multicollinearity problem (i.e. two or more exploratory variables had high correlation). When two variables are highly related, this means that they carry similar information, thus there is no need to include both of them in the model, as they are not adding any further information about the interest rates, when we add them sequentially.

We assessed multicollinearity by calculating the variance inflation factor (VIF) of all predictors in our model. This helped us in identifying the correlation between independent variables and determining the strength of that correlation. The VIF values begin at 1 (indicating that there is no correlation between this independent variable and any others) and have no upper limit. Based on the general rule of thumb we excluded sequentially the variables whose VIFs exceeded 10 (in the case of factors with more than two levels, we could check whether the square root of VIF (GVIF) was higher than $\sqrt{10}$=3.16), starting from the ones with the highest value. This would result in having moderate correlation, but not severe enough

to require corrective measures. In our case, we did not face any multicollinearity problem. Finally, we ended up with a model containing four predictors (out of the total 48), so we could now proceed with making our predictions.

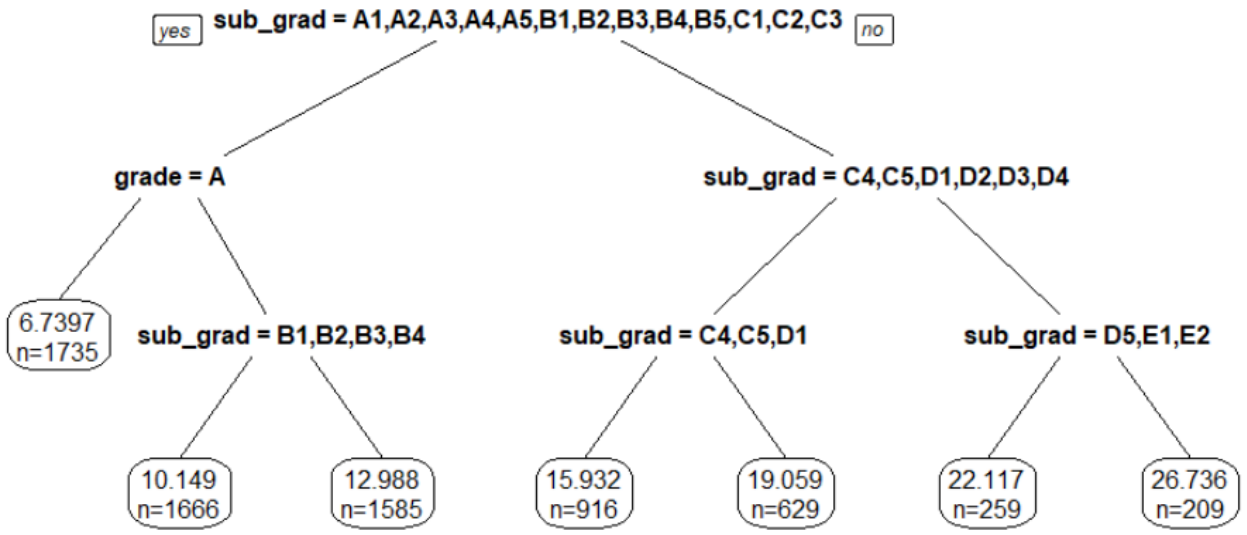
After using the linear regression model for making our predictions on the test set, we noticed that the overall estimate for the interest rate was equal to 12.41 (in contrast the mean value of interest rate was equal to 12.43). We also computed the prediction interval for our fitted values, which gives the uncertainty around a single value, by taking into account the sampling error of the fitted values. In our case, the prediction interval was (12.03, 12.75). In addition, we used the RMSE metric (equal to 0.37) as an extra assessment of the models' forecasting accuracy (the smaller the value, the better). A good model would have similar RMSE score in the test and train sets. A much higher value of the RMSE in the test set would be a sign of overfitting, while a much higher value in the train set would be a sign of underfitting the data.

3.2. Regression Tree

The decision trees can be used both for classification and regression. In our case, we had a continuous dependent variable, thus we used regression trees for predicting the interest rates. We started by creating the largest (full) tree possible, without any restriction. When growing a big bushy tree, it could have too much variance (high node heterogeneity) and there is also the risk of overfitting. This is the reason why we decided to post prune the tree.

Pruning is a technique that reduces the size of decision trees as well as the overall complexity of the tree, and hence improves predictive accuracy by the reduction of overfitting. By overfitting we mean that as the size of the tree is getting bigger, the model will keep getting better and better. The smallest decision tree we can create is one with two nodes. The main techniques used to avoid overfitting when building a decision tree model are pre-pruning and post-pruning. As we have already let the tree grew to its entirety, we decided use the latter technique (i.e. post-pruning).

In our case, in order to (post) prune the tree, we utilized the complexity parameter (cp) which is used to control the size of the decision tree and to select the optimal tree size. If the cost of adding another variable to the decision tree from the current node is above the value of cp, then tree building does not continue. So, we produced a pruned tree based on the best cp value (equal to 0.01). A plot of the (post-) pruned tree was showcased below.

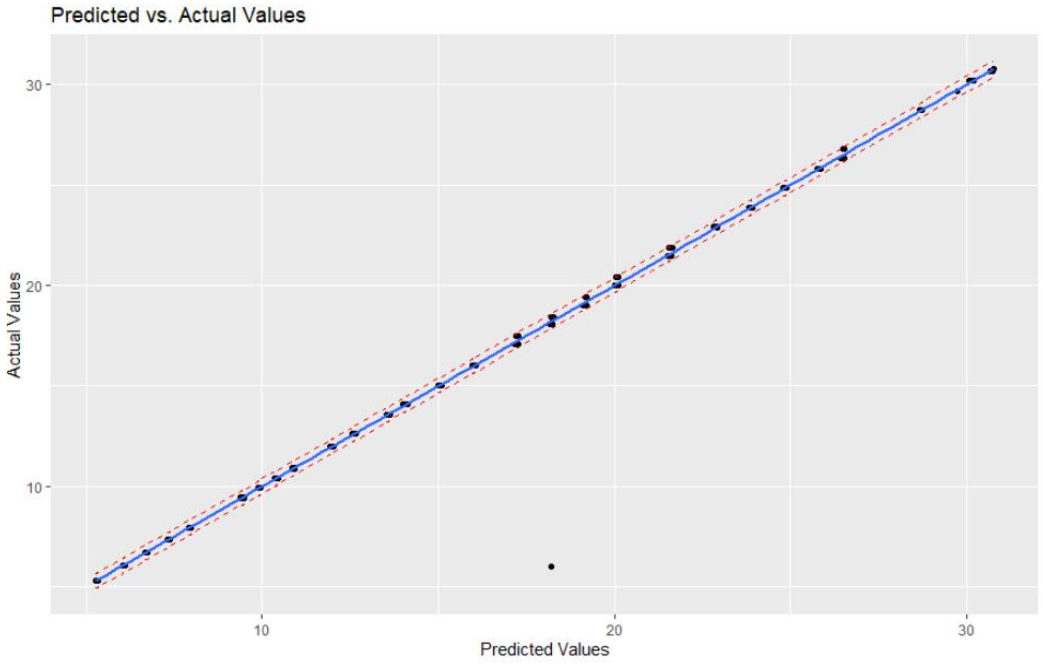


After using the linear regression model for making our predictions on the test set, we noticed that the overall estimate for the interest rate was equal to 12.39 (in contrast the mean value of interest rate was equal to 12.43) with an RMSE value equal to 0.86 9compared to 0.37 when using the linear regression model).

4. Test results visualization

Below, we presented some visualizations of the test results for the algorithms used.

4.1. For Linear Regression Model



4.2. For Regression Tree