MIS431 - Final Project Instructions

Jingyuan Yang - George Mason University, School of Business

Introduction

This is an individual assignment and will be a chance for you to perform an applied data science project on a real data set.

As with the mid-term project please submit the following for the final project

- R Markdown file in based on the template shared
- Word document which has knit from the R Markdown file

Loan Dataset

We will be working with the `loans_df` data frame in this project. This data set contains information on over 4,000 individuals who secured a personal loan in 2017 from a national bank. The description of this data and the variables contained in it are provided below.

The objective of this project is to explore the factors that lead to loan default and develop a machine learning algorithm that will predict the likelihood of an applicant defaulting on their loan in the future.

The loans_df data frame contains information on 3 and 5-year loans that were originated in 2017 by a national bank for customers residing in the Middle Atlantic and Northeast regions of the United States.

The company is looking to see if it can determine the factors that lead to loan default and whether it can predict if a customer will eventually default on their loan.

The bank has experienced record levels of customers defaulting on their loans in the past couple of years and this is leading to large financial losses.

The goal is to become better at identifying customers at risk of defaulting on their loans to minimize financial losses.

- What are the factors that are associated with customers defaulting on their loans?
- Is it possible to predict whether a customer will default on their loan? If so, how accurate are the predictions?
- How many costly errors is the model expected to produce (customers classified as not defaulting, but eventually do)?
- Are there any actions or policies the bank can implement to reduce the risk of loan default?

Specifically, the broad questions that the bank is trying to answer include:

The data set contains a mixture of applicant financial information (income, debt ratios, etc..), and applicant behavior (number of open accounts, historical engagement with the bank's products, number of missed payments, etc. . .)

The response variable in this data is `loan_default`. This variable records whether an applicant eventually defaulted on their loan and indicates a financial loss to the bank.

Note: The response variable has been coded as a factor with 'yes' as the first level. This is the format that `tidymodels` expects for calculating model performance metrics. There is no need to recode this variable in your machine learning process.

Variable Information

Variable	Definition	Data Type
loan_default	Did the borrower default on their loan (yes/no)	Factor
loan_amount	Loan amount	Integer
installment	Monthly paymeny amount	Numeric
interest_rate	Interest rate	Numeric
loan_purpose	Purpose of the loan	Factor
application_type	Loan application type (individual or joint)	Factor
term	Loan term (three/five year)	Factor
homeownership	Borrower(s) homeownership status	Factor
annual_income	Annual income	Numeric
current_job_years	Years employed at current job	Numeric
debt_to_income	Debt-to-income ratio at application time	Numeric
total_credit_lines	Total number of open credit lines	Integer
years_credit_history	Years of credit history	Numeric
missed_payment_2_yr	History of missed payments in the last 2 years (yes/no)	Factor
history_bankruptcy	History of bankruptcy (yes/no)	Factor
history_tax_liens	History of tax liens (yes/no)	Factor

Raw Data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
loans_df <- read_rds("E:/RDataFiles/loan_data.rds")
loans_df
```

```
## # A tibble: 4,110 x 16
##    loan_default loan_amount installment interest_rate loan_purpose
##    <fct>              <int>       <dbl>         <dbl> <fct>
##  1 yes                35000        927.         17.2  small_busin~
##  2 yes                10000        260.         11.5  small_busin~
##  3 no                 28800        942.          8.97 debt_consol~
##  4 yes                 4475        165.         10    medical
##  5 no                  3600        111.          9.72 medical
##  6 yes                12800        389.         20    medical
##  7 yes                35000        927.         18.2  debt_consol~
```

```
##  8 no                 26000        619.         12.0  debt_consol~
##  9 no                  5500        176.          7.97 debt_consol~
## 10 no                 40000        952.         11.0  home_improv~
## # ... with 4,100 more rows, and 11 more variables: application_type <fct>,
## #   term <fct>, homeownership <fct>, annual_income <dbl>,
## #   current_job_years <dbl>, debt_to_income <dbl>, total_credit_lines <int>,
## #   years_credit_history <dbl>, missed_payment_2_yr <fct>,
## #   history_bankruptcy <fct>, history_tax_liens <fct>
```

Data Analysis [30 Points]

In this section, you must think of at least 6 relevant questions that explore the relationship between `loan_default` and the other variables in the `loan_df` data set. The goal of your analysis should be discovering which variables drive the differences between customers who do and do not default on their loans.

You must answer each question and provide supporting data summaries with either a summary data frame (using `dplyr`/`tidyr`) or a plot (using `ggplot`) or both.

In total, you must have a minimum of 3 plots (created with `ggplot`) and 3 summary data frames (created with `dplyr`) for the exploratory data analysis section. Among the plots you produce, you must have at least 3 different types (ex. box plot, bar chart, histogram, scatter plot, etc. . .)

See the example question below.

Note: To add an R code chunk to any section of your project, you can use the keyboard shortcut `Ctrl` + `Alt` + `i` or the `insert` button at the top of your R project template notebook file.

Sample Question

Are there differences in loan default rates by loan purpose?

Answer: Yes, the data indicates that credit card and medical loans have significantly larger default rates than any other type of loan. In fact, both of these loan types have default rates at more than 50%. This is nearly two times the average default rate for all other loan types.

Summary Table

```
loans_df %>%
  group_by(loan_purpose) %>%
  summarise(n_customers = n(),
            customers_default = sum(loan_default == 'yes'),
            default_percent = 100 * mean(loan_default == 'yes'))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

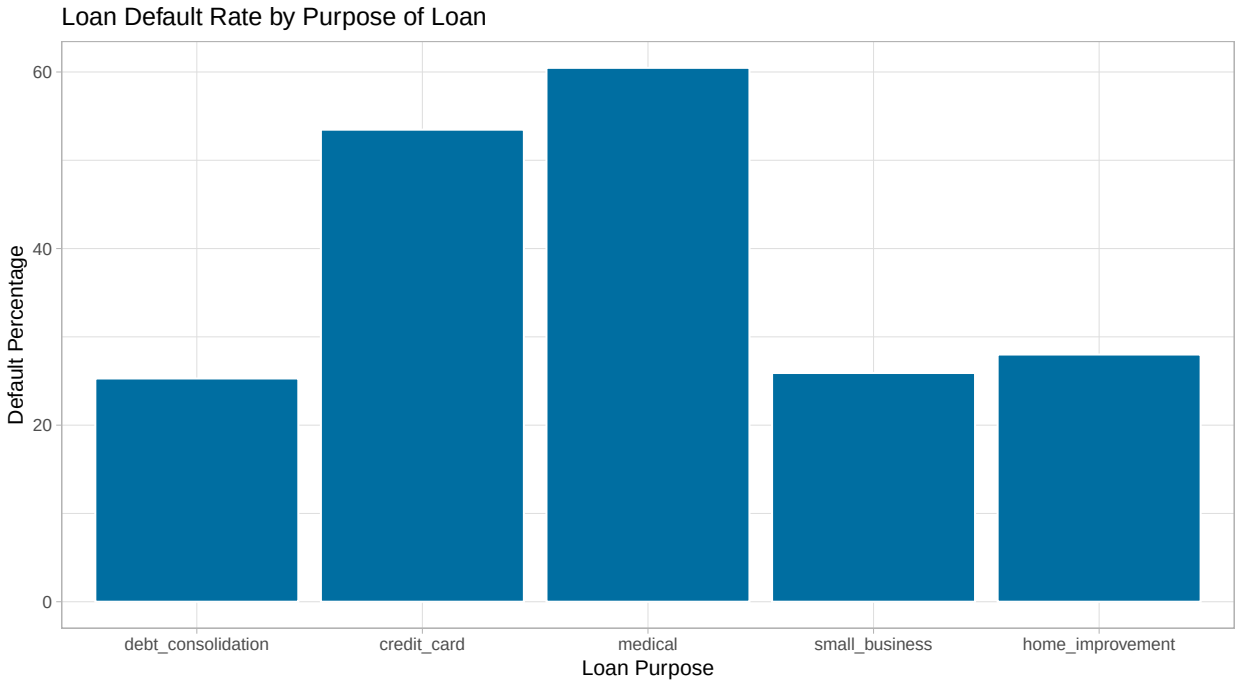
```
## # A tibble: 5 x 4
##   loan_purpose       n_customers customers_default default_percent
##   <fct>                    <int>             <int>           <dbl>
## 1 debt_consolidation        1218               308            25.3
## 2 credit_card                879               470            53.5
## 3 medical                    635               384            60.5
## 4 small_business             853               221            25.9
## 5 home_improvement           525               147            28.
```

Data Visulatization

```
default_rates <- loans_df %>%
                 group_by(loan_purpose) %>%
                 summarise(n_customers = n(),
                 customers_default = sum(loan_default == 'yes'),
                 default_percent = 100 * mean(loan_default == 'yes'))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(data = default_rates, mapping = aes(x = loan_purpose, y = default_percent)) +
    geom_bar(stat = 'identity', fill = '#006EA1', color = 'white') +
    labs(title = 'Loan Default Rate by Purpose of Loan',
         x = 'Loan Purpose',
         y = 'Default Percentage') +
    theme_light()
```



Predictive Modeling [70 Points]

In this section of the project, you will fit **three classification algorithms** to predict the response variable,`loan_default`. You should use all of the other variables in the `loans_df` data as predictor variables for each model.

You must follow the machine learning steps below.

The data splitting and feature engineering steps should only be done once so that your models are using the same data and feature engineering steps for training.

- Split the `loans_df` data into a training and test set (remember to set your seed)
- Specify a feature engineering pipeline with the `recipes` package
  - You can include steps such as skewness transformation, dummy variable encoding or any other steps you find appropriate
- Specify a `parsnip` model object

  - You may choose from the following classification algorithms:
    * Logistic Regression
    * LDA
    * QDA
    * KNN
    * Decision Tree
    * Random Forest
    * Please note that you cannot use linear regression for classification problem
- Package your recipe and model into a workflow
- Fit your workflow to the training data
  - If your model has hyperparameters:
    * Split the training data into 5 folds for 5-fold cross validation using `vfold_cv` (remember to set your seed)
    * Perform hyperparamter tuning with a random grid search using the `grid_random()` function for Random Forest or regular tune_grid for KNN

    * Hyperparameter tuning can take a significant amount of computing time. Be careful not to set the `size` argument of `grid_random()` too large. I recommend `size` = 10 or smaller.
    * Select the best model with `select_best()` and finalize your workflow
- Evaluate model performance on the test set by plotting an ROC curve using `autoplot()` and calculating the area under the ROC curve on your test data

Summary of Results [50 Points]

Write a summary of your overall findings and recommendations to the executives at the bank. Think of this section as your closing remarks of a presentation, where you summarize your key findings, model performance, and make recommendations to improve loan processes at the bank.This needs to be provided in a word document, once you knit the document.

Your executive summary must be written in a business tone, with minimal grammatical errors, and should include the following sections:

1. An introduction where you explain the business problem and goals of your data analysis
   - What problem(s) is this company trying to solve? Why are they important to their future success?
   - What was the goal of your analysis? What questions were you trying to answer and why do they matter?
2. Highlights and key findings from your Exploratory Data Analysis section
   - What were the interesting findings from your analysis and **why are they important for the business**?
   - This section is meant to **establish the need for your recommendations** in the following section
3. Your "best" classification model and an analysis of its performance
   - In this section you should talk about the expected error of your model on future data
     - To estimate future performance, you can use your model performance results on the **test data**
   - You should discuss at least one performance metric, such as an F1 or ROC AUC for your model. However, you must explain the results in an **intuitive, non-technical manner**. Your audience in this case are executives at a bank with limited knowledge of machine learning.
4. Your recommendations to the company on how to reduce loan default rates
   - Each recommendation must be supported by your data analysis results

- You must clearly explain why you are making each recommendation and which results from your data analysis support this recommendation
- You must also describe the potential business impact of your recommendation:
  - Why is this a good recommendation?
  - What benefits will the business achieve?

5. Conclusion

Wrap up the report with concluding remarks by summarizing the results and your recommendations in one or two paragraphs.

6. Appendix/Appendices

Include all the code, tables, and plots in this section.

— End of Project Instructions —