CYBR 520 Lab 5: : Dimensionality Reduction through Recursive Feature Elimination - Spambase Dataset (100 points)

Due on Sunday 11/27/2022 @11:59PM

Instructions:

The is a group lab, each team is to submit one submission on eCampus. After the lab is submitted, each group member is to submit a [group member evaluation](https://forms.gle/TvsvercxLsb7sgev9) for each group member (this is worth 15 % of the total grade). We will be using R and Rstudio for this lab. Please read the following the document and provide your answer **below** each question. Keep the formatting of the document as is.

**Dataset: Spambase**

**R Packages: caret, e1071, randomForest,** **Hmisc,** **corrplot, ggcorrplot**

Background: The spambase dataset is a classic example dataset for exploring machine learning applications for developing cybersecurity controls (specifically spam filters). The spambase dataset was developed in 1999 by Mark Hopkins, Erik Reeber, George Forman, and Jaap Suermondt of Hewlett-Packard Labs. The spambase dataset is a corpus of spam emails and non-spam (“ham”) emails. 4601 unique instances comprise the dataset. Each instance contains 57 unique attributes which can be used to train machine learning algorithms to automatically classify the data as “ham” or “spam”. It is commonly assumed that this dataset does not contain any error, and all 4601 instances are correctly labeled based upon expert knowledge.

Our objective for this lab will be to try a dimensionality reduction of the spambas dataset and explore its effect on the classification performance. The Recursive Elimination Feature Elimination.

“Recursive Feature Elimination, or RFE Feature Selection, is a feature selection process that reduces a model’s complexity by choosing significant features and removing the weaker ones. The selection process eliminates these less relevant features one by one until it has achieved the optimum number needed to assure peak performance” ~[simplelearn](https://www.simplilearn.com/recursive-feature-elimination-article)

# Get Lab5.r code:

Using GitHub Desktop or PyCharm, pull latest Lab5.r file from the Github repository of the CYBR520 class and the Derby5.csv dataset. The code runs a feature reduction functionality to reduce the number of features used in the DERBY dataset to explore prediction performance (accuracy) enhancement. The idea is to explore whether using a subset of the features (ideally significantly smaller than the original) to obtain a higher accuracy. The code runs for the DERBY dataset to predict whether a file is buggy or not.

# Utilize code for the spambase dataset

Using the provided for Lab5.r on GitHub, modify the code, and utilize it to predict whether a given email is spam or nonspam usin the spambase dataset. [15 points]

Run the model and answer the following questions:

1. **What does correlationMatrix generate?** [5 points)

**Observation:**

**The correlationMatrix generates a table of correlation values (coefficients) for each feature in the dataset compared to every other feature in the dataset. The matrix lists all features in both the x- and y-directions, and provides a correlation coefficient between -1.0 and 1.0 for each feature comparison represented by the intersection of the features on the x-axis and y-axis. Positive correlation coefficients indicate the degree of positive correlation, which means that as one feature increases in value, the other feature does as well. Negative correlation coefficients indicate the degree of negative correlation, which means that as one feature increases in value, the other feature decreases in value. The closer the absolute value of the correlation coefficient is to 1.0, the stronger the correlation is (i.e., features with a correlation coefficient of -0.85 have a relatively strong negative correlation). Correlation coefficients of 1.0 indicate a perfect correlation and occur when a feature is compared with itself.**

**References:**

**How to Create a Correlation Matrix in R. (2022, August 23), “A correlation matrix is a table of correlation coefficients for a set of variables used to determine if a relationship exists between the variables.”**

1. **Name three features with the highest positive correlation with the type feature.**
   1. **How can you tell which features to select?** [5 points)

**Observation:**

**The three features with the highest positive correlation with the “type” feature (not including when compared with itself) are “your”, “num000”, and “remove”. Please refer to the table below for their corresponding correlation coefficients:**

|  |  |
| --- | --- |
| **Feature** | **Value** |
| **type** | **1** |
| **your** | **0.38323382** |
| **num000** | **0.33478704** |
| **remove** | **0.33211742** |

**To determine the features with the highest positive correlation, one simply selects the comparisons with the highest correlation coefficients (closest to 1.0). Of course, “type” has the strongest correlation with itself, as is shown by its correlation coefficient value of 1.0 when compared to “type”.**

1. **How many values are in the correlationMatrix?** [5 points)

**Observation:**

**In total there are 58 x 58 = 3,364 values in the correlation matrix.**

1. **How many of those value’s matter? Do we need all of them to ?** [5 points)

**Observation:**

**At face value one can eliminate correlations of each feature to themselves, of which there are 58 in this correlation matrix. This leaves 3,306 useful correlation coefficients.**

1. **What are the highest correlated features?** [5 points)

**Observation:**

**The following features were identified as having the highest correlation:**

**"num857"**

**"num415"**

**"technology"**

**"labs"**

**"telnet"**

**"num650"**

**"direct"**

**"hp"**

1. **What are the most important features in the dataset?** [5 points)

**Observation:**

**The output from printing the importance variable lists the following top 20 features in regards to importance:**

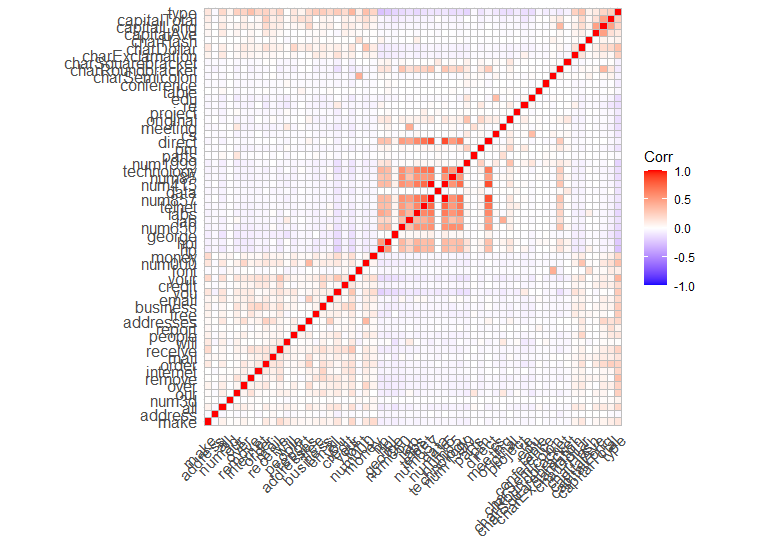
**Only 20 most important variables shown (out of 57)**

|  |  |
| --- | --- |
| **Feature** | **Overall** |
| **charExclamation** | **0.32453** |
| **your** | **0.14687** |
| **num000** | **0.11208** |
| **remove** | **0.1103** |
| **charDollar** | **0.10474** |
| **you** | **0.07489** |
| **free** | **0.06928** |
| **business** | **0.06928** |
| **hp** | **0.06591** |
| **capitalTotal** | **0.06208** |
| **our** | **0.05853** |
| **receive** | **0.055** |
| **hpl** | **0.05427** |
| **over** | **0.0541** |
| **order** | **0.05362** |
| **money** | **0.0467** |
| **capitalLong** | **0.0467** |
| **internet** | **0.04277** |
| **email** | **0.0417** |
| **all** | **0.0388** |

1. **Provide the correlation plot, what do you think the dark areas reflect?** [5 points)

**Observation:**

**The correlation plot we generated is below:**



**The darker areas indicate higher correlations between feature comparisons. Darker red indicates higher positive correlations, whereas, darker blue indicates higher negative correlations. The red diagonal stripe indicates the perfect correlation that occurs when each feature is compared with itself.**

1. **What were the chosen features after running the random forest model?** [5 points)

**Observation:**

**The RFE function pared the dataset down to the following 5 critical features required for optimization:**

**The top 5 variables (out of 42):**

**“capitalLong”**

**“report”**

**“order”**

**“num1999”**

**“charHash”**

1. **Use the code provided in the previous labs and run one classification model (e.g., svm, knn, or decision tree) using the full data, the selected features in step 6, and the selected features in step 8.** [15 points)

**Observation:**

**Our group opted to compare the performance of an SVM classification model for the full dataset in comparison to the two feature-selected datasets from questions 6 and 8. Please see the observations in part a. and b. below:**

* 1. **Report Accuracy and Recall** [10 points)

**Full Dataset: 58 Features**

**Accuracy: 0.934**

**Recall: 0.9653**

**Question 6: 20 Features**

**Accuracy: 0.9246**

**Recall: 0.9474**

**Question 8: 5 Features**

**Accuracy: 0.7578**

**Recall: 0.8864**

* 1. **Are there any differences among the three models in terms of accuracy and recall? [10 points]**

**The differences from the full dataset and Question 6 are marginal. The Accuracy in Question 6 is 0.0094 less than the full dataset and the Recall is 0.0179 less than the full dataset.**

**The differences from the full dataset and Question 8 are significant. The Accuracy in Question 8 is 0.1762 less than Question 6 and the Recall is 0.0789 less than Question 6.**

**The differences from Question 6 and Question 8 are significant. The Accuracy in Question 8 is 0.1668 less than Question 6 and the Recall is 0.061 less than Question 6.**

# Submission:

Submit the source code of your code to your group GitHub Repository, and submit this files along with the answers to eCampus. [10 points]