**Dataset: Loan data**

**Target**: built supervised(labeled) learning model, applied algorithm to define people should be rejected or accepted.

**Overview**: 13 columns, categories and continuous, numeric and text

* Data exploring
* Data preprocessing
* Data algorithm modeling + optimization
* Data visualization

Algorithms applied: Neural Network, Text Mining, NBC

**Neural Network:**

* **Data pre-processing and cleaning**

Change to 0,1 or categories to 1-n transmission, normalization,

* **Splitting the cleaned dataset with 80/20% proportion**
* **Built first model, get the accuracy and plot the neural network**
* **Optimization of model:**

1. Choose the columns to get the better accuracy using RandomForest
2. Write a function to get the best number of hidden layer.

* **Build model computing accuracy.**

Using package: neuralnet, nnet, devtools.

source\_url('https://gist.githubusercontent.com/fawda123/7471137/raw/466c1474d0a505ff044412703516c34f1a4684a5/nnet\_plot\_update.r')-----for plot.nnet

**Text Mining using Rvest (external p&n)**

* Web scraping live reviews about JW Marriott hotel from Trip-Advisor, northeasternreviews

write.csv(df, "C:/Users/agraw/Desktop/ADS/Final Project/tripadvisor\_reviews.csv")

* sentiment analysis

1. # Load the tripadvisor\_reviews.csv review data and explore
2. # Load the positive and negative lexicon data and explore
3. # Making a corpus out of the hotel reviews
4. # Remove stop words, punctuations, numbers from all the reviews and inspecting it
5. # Load the stop words text file and explore
6. # Remove stop words of the external file from the corpus and whitespaces again and inspect
7. # Character representation of the corpus of first review
8. # Stem the words to their root of all reviews present in the corpus
9. # Building a term document matrix of the stemmed corpus (terms occurring with a minimum frequency of 5)
10. # Calculating the count and percentage of total positive and negative words in each review and
11. Labeling each review as either negative or positive
12. # Sentiment score of each review and visualizing using boxplot
13. # Calculating & Visualiztion of positive and negative count of single review
14. # Calculating & Visualiztion of overall percentage of positive and negative words of all the reviews

Package used: tm, stringr, rvest.

**Text Mining using devtools, pander(includes dictionary).**

* Tokenizing the data without stop words
* Single Sentiment analysis:

1. First set the positive in sentiment columns
2. Loop all the rows to match the dictionary using pander package(compare with dictionary to define negative)
3. Write out the result table
4. Visualization of the individual review (which you can enter which row you want to see)

nrc\_data <- get\_nrc\_sentiment(as.String(product\_review$Body[n]))

pander::pandoc.table(nrc\_data[, 1:8], split.table = Inf)

* overall sentiment analysis.
* Show the actual word using function, like which words comes more

**NBC analysis**

Packages: Caret, e1071

* Splitting data
* Make naïve Bayesian model
* Predict output variables
* Build confusion matrix to compare actual and predict
* Accuracy of the model