LENDING CLUB DATA ANALYSIS
(Assignment-2 Report)

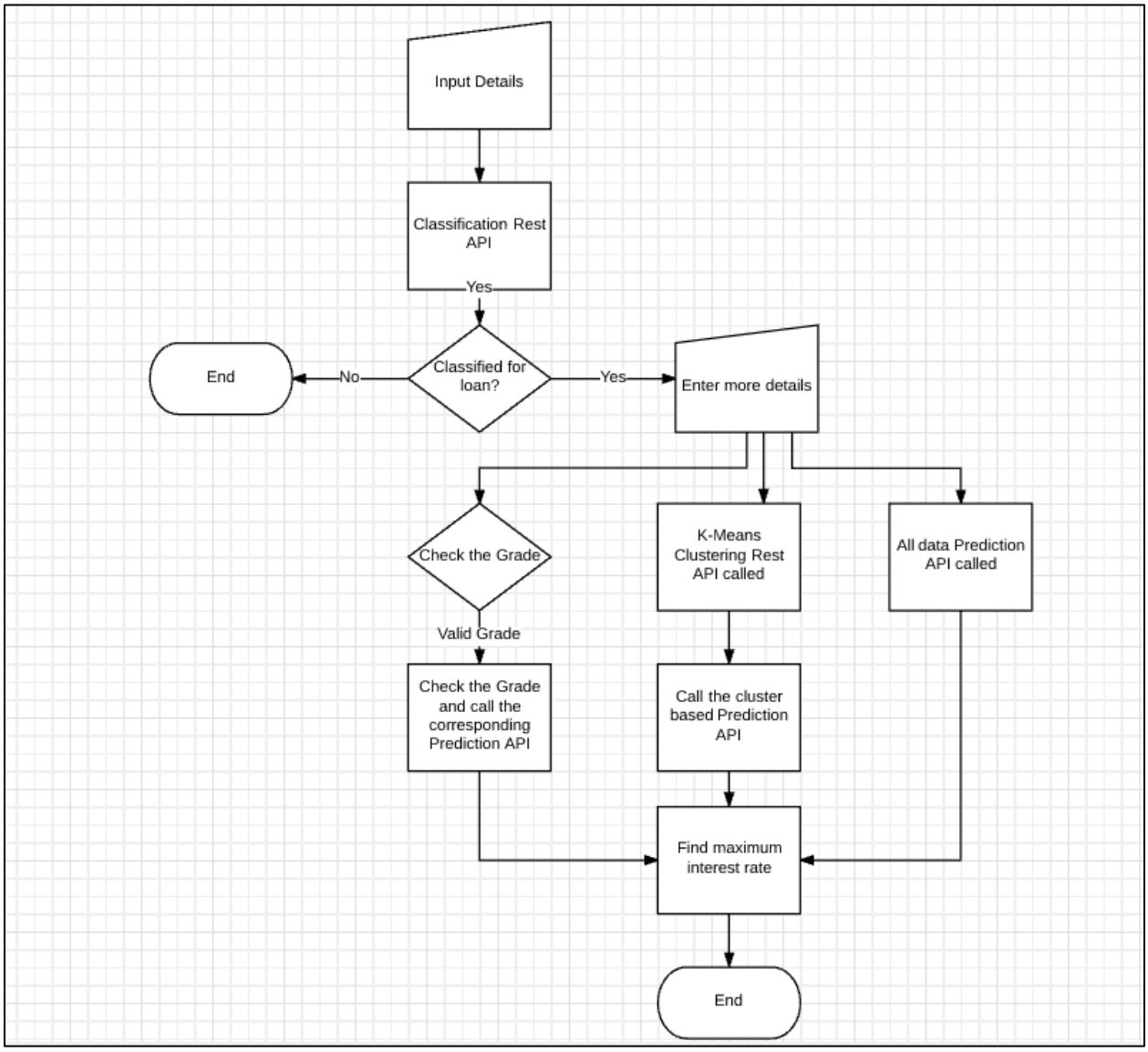
Advance in Data Sciences and Architecture
INFO 7390 - SPRING 2017

PROFESSOR:
SRIKANTH KRISHNAMURTHY

TEAM MEMBERS (Team 9):
AASHRI TANDON
PRAGATI SHAW
SARTHAK AGARWAL

Table of Contents

Flow of the project:



Classification

The task was to classify a new record whether loan will be approved or not. We have created a combined csv file for both accepted loan data and declined loan data with the following columns:

DTI ratio

Fico score Low (or Risk Score): Note- vantage score is assumed to be fico score

State

Zip Code

Loan Amount

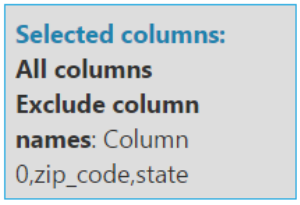
Policy Code

Employment Length

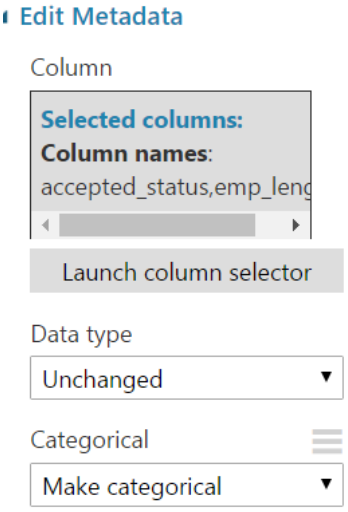
Accepted Status (New Column generated for both files)

We have trained all our models on Microsoft Azure Machine Learning Studio.

Columns excluded: State and Zip Code

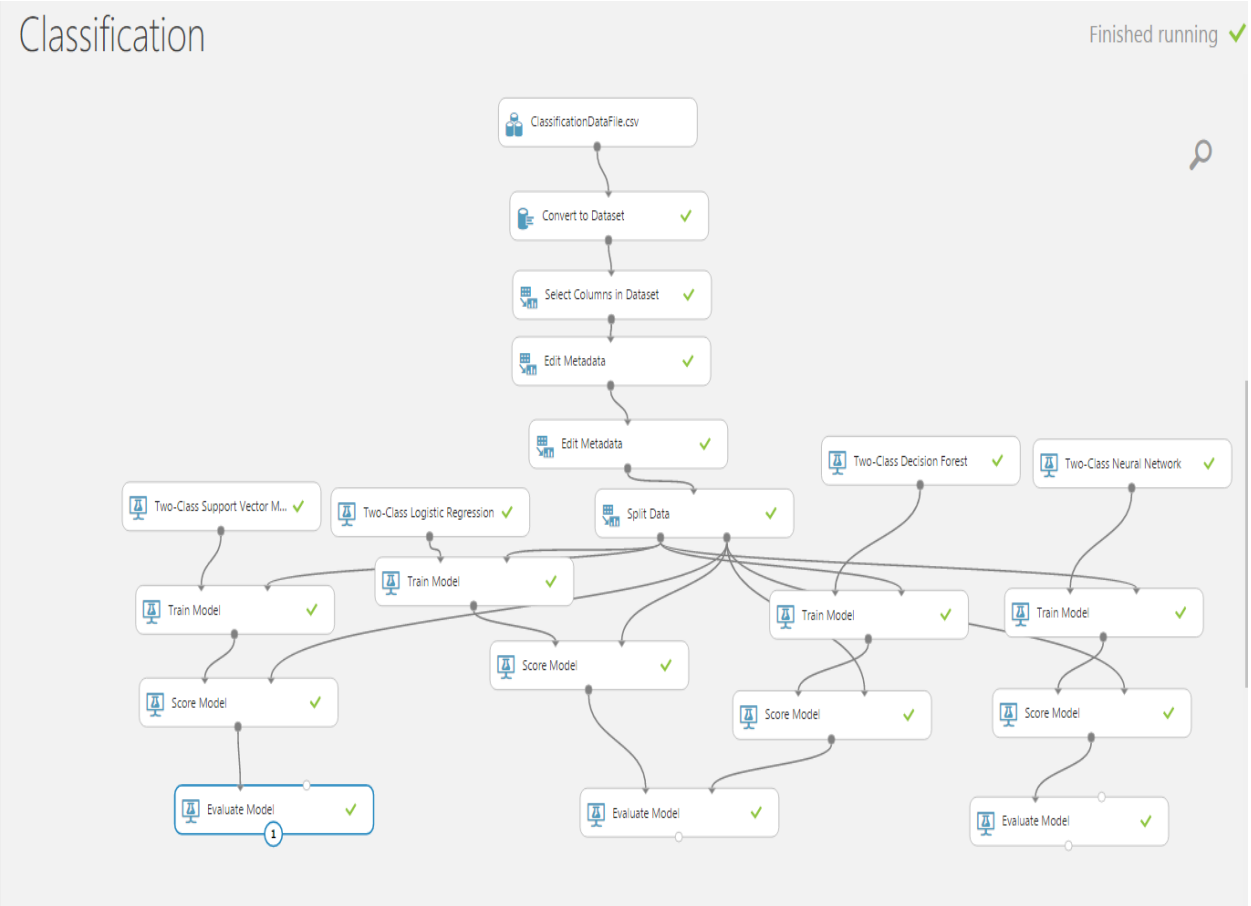


Metadata Editor: Converted Employment Length, Policy Code, Accepted Status to categorical fields



Training was done on 'Accepted Status' column.

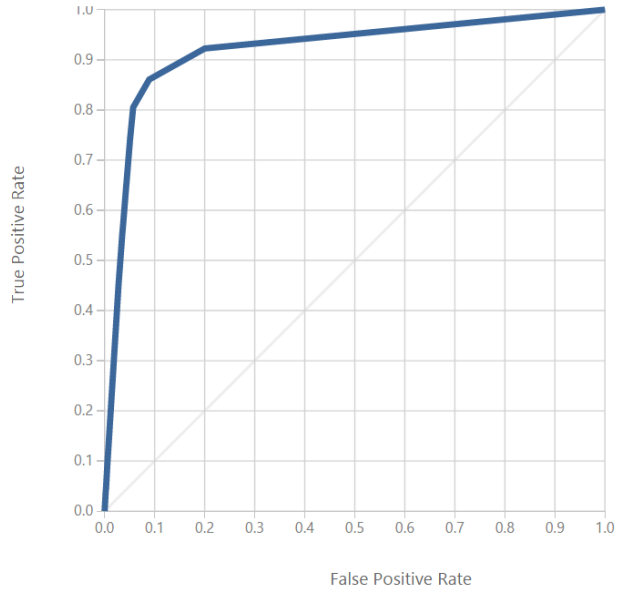
Below is the screenshot of all the algorithms that were used to train the model. ROC curve value was used to determine the best model out of four (Two class support vector machine, Logistic Regression, Two class decision forest, Two class Neural Network)



Evaluation Results of various classification algorithms:

Two class Logistic Regression

- ROC Curve

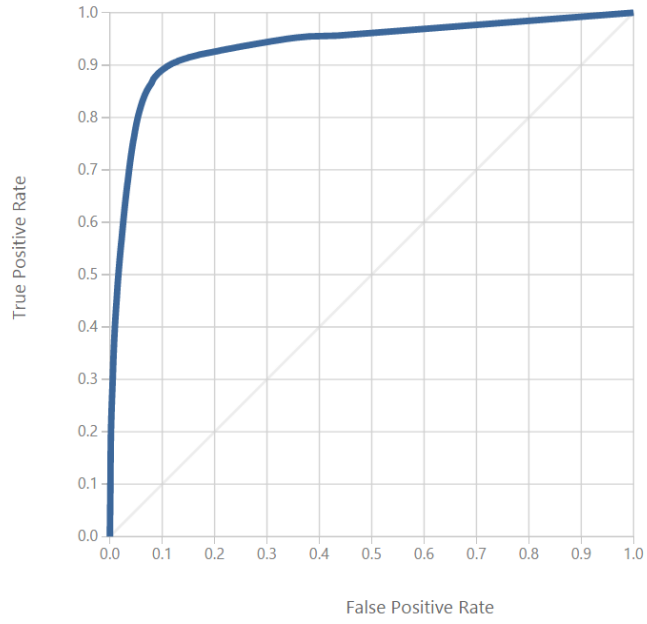


- Confusion Matrix

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
531284	128559	0.928	0.627	0.5	0.920
False Positive	True Negative	Recall	F1 Score		
316194	5223566	0.805	0.705		
Positive Label	Negative Label				
Y	N				

Two class decision forest

- ROC Curve



- Confusion Matrix

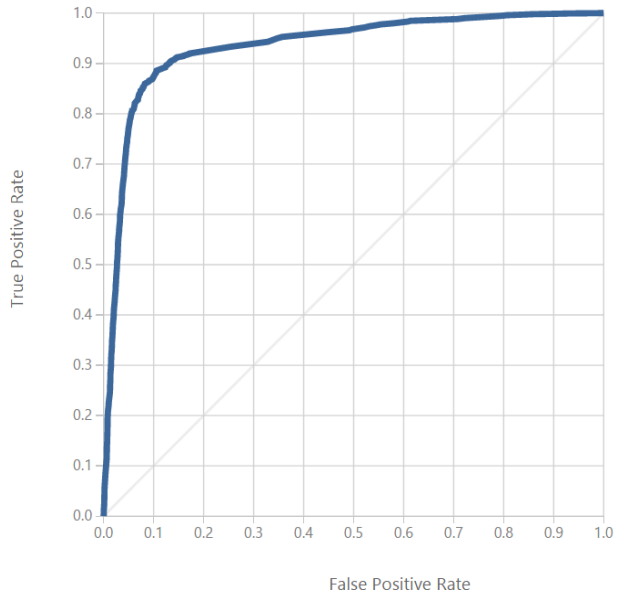
True Positive	False Negative	Accuracy	Precision	Threshold	AUC
443422	216421	0.935	0.706	0.5	0.937

False Positive	True Negative	Recall	F1 Score
184429	5355331	0.672	0.689

Positive Label	Negative Label
Y	N

Two class Neural Network

- ROC Curve



- Confusion Matrix

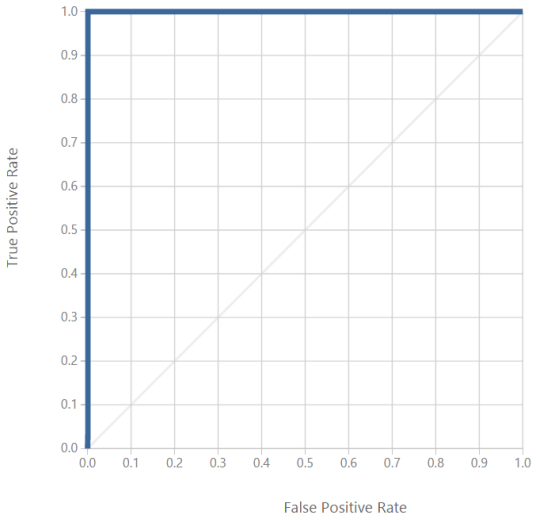
True Positive	False Negative	Accuracy	Precision	Threshold	AUC
517395	142448	0.931	0.642	0.5	0.935

False Positive	True Negative	Recall	F1 Score
288281	5251479	0.784	0.706

Positive Label	Negative Label
Y	N

Two class support vector machine

- ROC Curve



- Confusion Matrix

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
623758	0	0.999	0.992	0.5	0.999
False Positive	True Negative	Recall	F1 Score		
4761	5357201	1.000	0.996		
Positive Label	Negative Label				
Y	N				

Chosen Model: Two class SVM as AUC is 0.999

Clustering

Manual Clustering

Based on each loan application and credit report, every loan is assigned a grade ranging from A to G with a corresponding interest rate. We have considered 'grades' of every loan record to cluster the entire dataset.

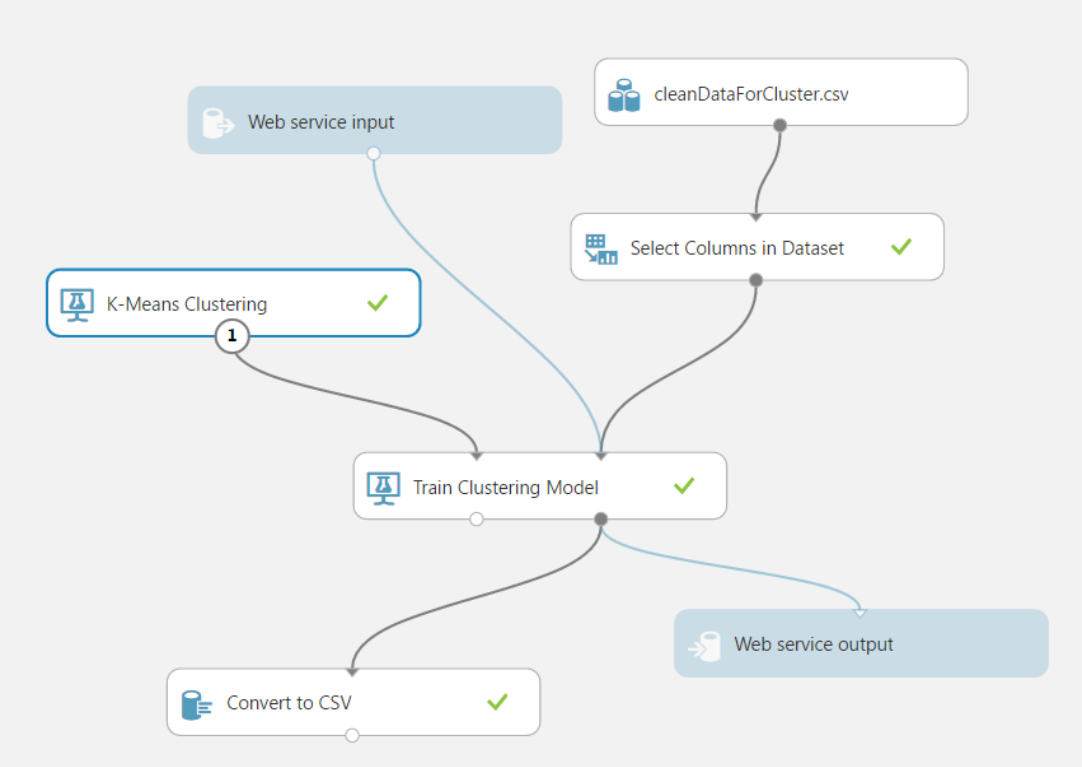
```
import pandas as pd
import numpy as np
pd.set_option('display.max_rows', 200)
pd.set_option('display.max_columns', 200)

# In[2]:
loanData = pd.read_csv('D:\\ADS\\Assignments\\Assignment2\\cleanDataForCluster.csv',encoding = 'iso-8859-1',index_col=0)
loanData
# In[3]:
groupdedDAta=list(loanData.groupby(loanData['grade']))
groupdedDAta
# In[4]:
gradeA = groupdedDAta[0][1]
gradeA.to_csv("gradeA.csv")
# In[5]:
gradeB = groupdedDAta[1][1]
gradeB.to_csv("gradeB.csv")
# In[6]:
gradeC = groupdedDAta[2][1]
gradeC.to_csv("gradeC.csv")
# In[7]:
gradeD = groupdedDAta[3][1]
gradeD.to_csv("gradeD.csv")
# In[8]:
gradeE = groupdedDAta[4][1]
gradeE.to_csv("gradeE.csv")
# In[9]:
gradeF = groupdedDAta[5][1]
gradeF.to_csv("gradeF.csv")
# In[10]:
gradeG = groupdedDAta[6][1]
gradeF.to_csv("gradeG.csv")
```

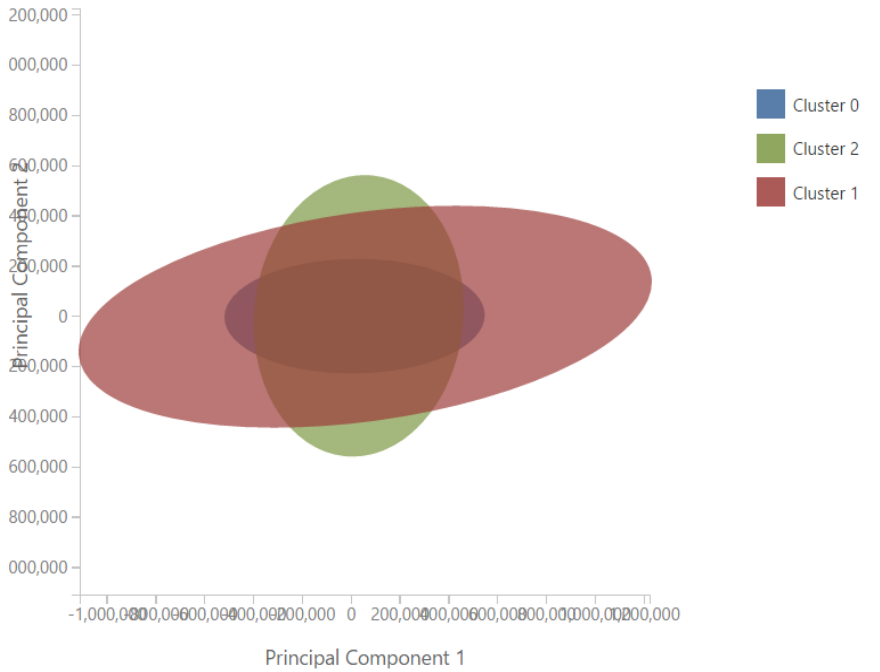
Based on Clustering Algorithm(k-means)

k-means clustering aims to partition dataset into **k clusters** in which record belongs to the **cluster** with the nearest **mean**, serving as a prototype of the **cluster**.

This algorithm is an iterative process.

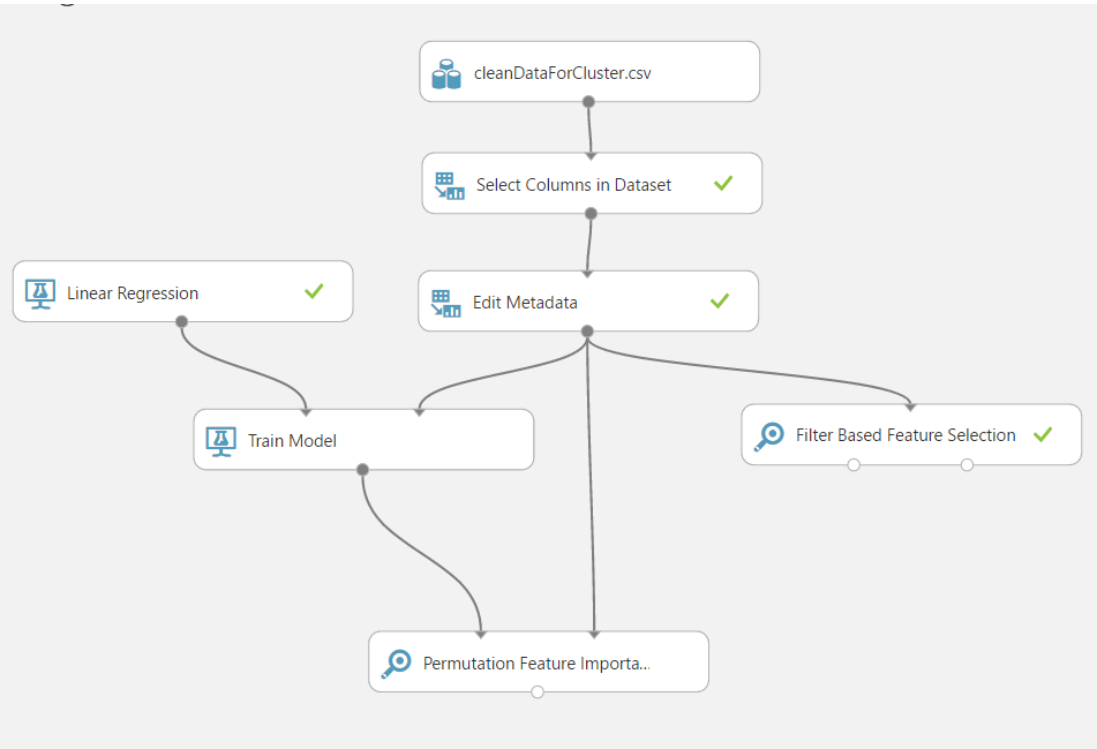
We have made 3 clusters (No of centroids = 3)


Web service input
cleanDataForCluster.csv
Select Columns in Dataset
K-Means Clustering
1
Train Clustering Model
Web service output
Convert to CSV


Cluster 0
Cluster 2
Cluster 1
Principal Component 2
Principal Component 1

Prediction

Variable Selection



Selected 17 variables out of 90:

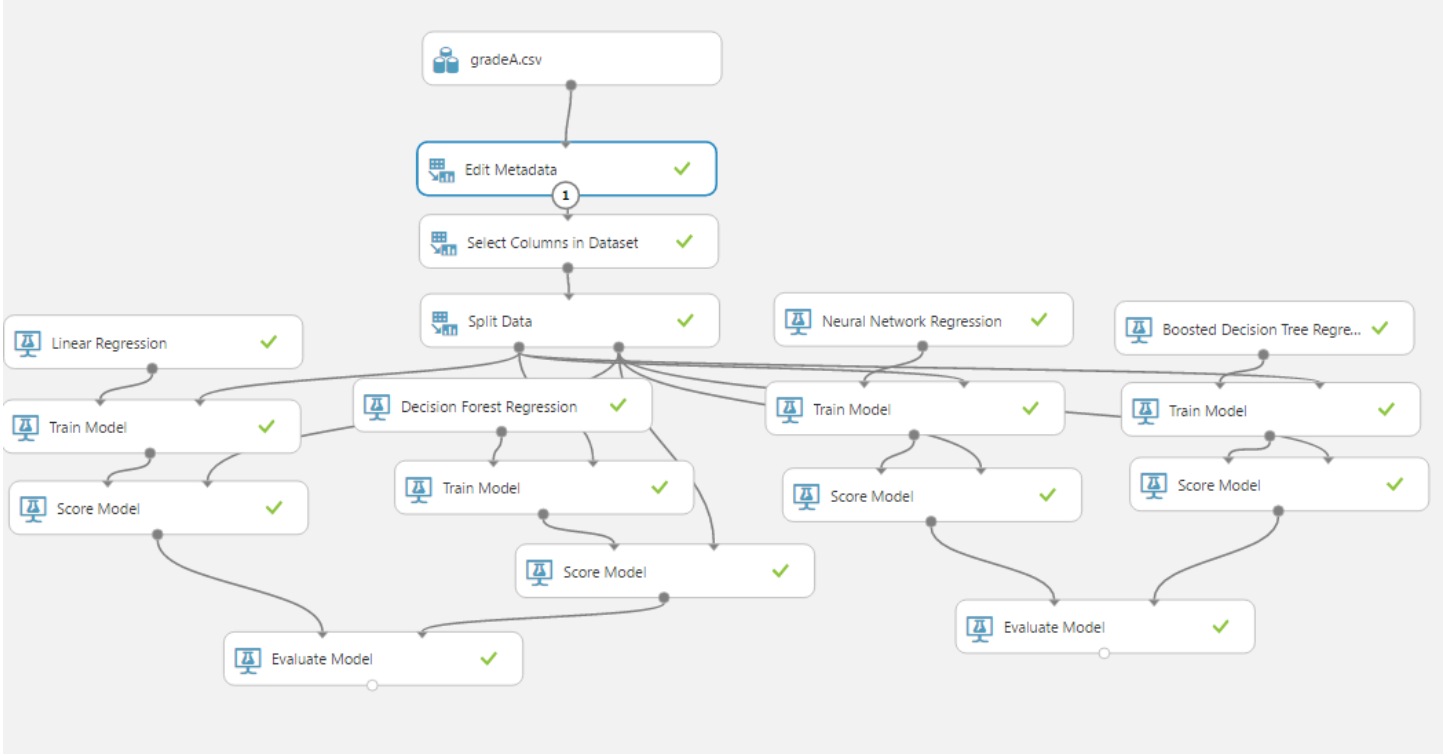
int_rate,
sub_grade,
grade,
dti,
fico_range_low,
fico_range_high,
term,
bc_util,
total_pymnt,
loan_amnt,
funded_amnt,
total_pymnt_inv,
home_ownership,
loan_status,purpose,
application_type,
emp_length,
policy_code

Prediction for manual clusters

- term, grade, sub_grade, home_ownership, loan_status, purpose, application_type, emp_length is converted to categorical

- Training was done on 'int_rate' column.

- Below is the screenshot of all the algorithms that were used to train the model.

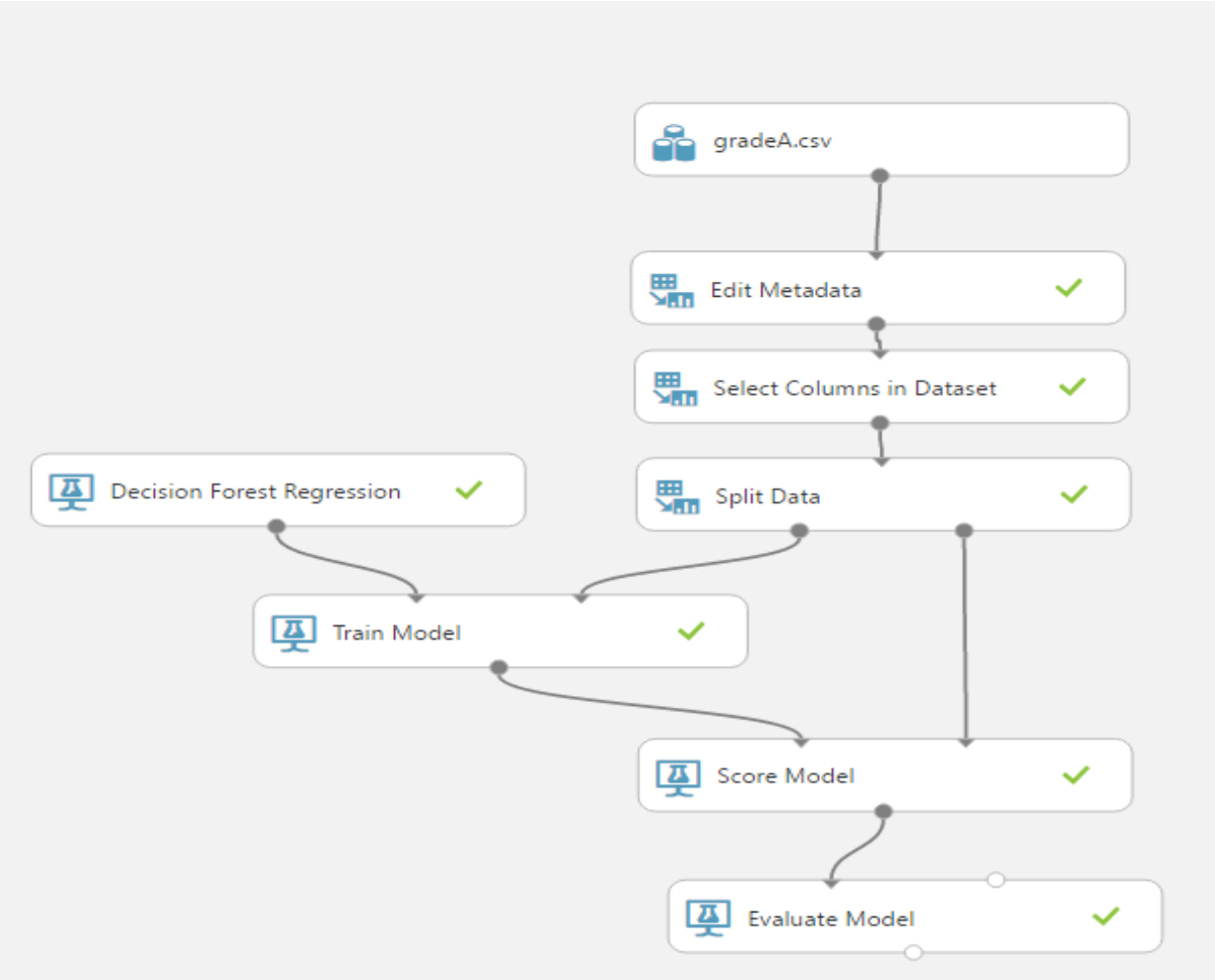


The following metrics are reported for evaluating regression models. All metrics are reported but the models are ranked by the metric you select for evaluation.

- **Mean absolute error (MAE)** measures how close the predictions are to the actual outcomes; thus, a lower score is better.
- **Root mean squared error (RMSE)** creates a single value that summarizes the error in the model. By squaring the difference, the metric disregards the difference between over-prediction and under-prediction.
- **Relative absolute error (RAE)** is the relative absolute difference between expected and actual values; relative because the mean difference is divided by the arithmetic mean.
- **Relative squared error (RSE)** similarly normalizes the total squared error of the predicted values by dividing by the total squared error of the actual values.
- **Mean Zero One Error (MZOE)** indicates whether the prediction was correct or not. In other words: *ZeroOneLoss(x,y) = 1 when x!=y; otherwise 0*
- **Coefficient of determination**, often referred to as R^2, represents the predictive power of the model as a value between 0 and 1. Zero means the model is random (explains nothing); 1 means there is a perfect fit. However, caution should be used in interpreting R^2 values, as low values can be entirely normal and high values can be suspect.

- Coefficient of Determination was used to determine the best model

Grade A	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.255555	0.324468	0.303447	0.102186	0.897814
Decision Forest	0.135789	0.242014	0.161236	0.05685	0.94315
Neural Network Regression	0.176836	0.256185	0.209977	0.063702	0.936298
Boosted Decision Tree Regression	0.169064	0.246523	0.200747	0.058988	0.941012



-The same steps were repeated for all the grades from A – G

Grade B	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.502831	0.644016	0.44532	0.222228	0.777772
Decision Forest	0.272973	0.48012	0.241752	0.12351	0.87649
Neural Network Regression	0.322249	0.472912	0.285393	0.11983	0.88017

Grade C	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.559709	0.695042	0.577957	0.33929	0.66071
Decision Forest	0.289796	0.527047	0.299244	0.195096	0.804904
Neural Network Regression	0.380037	0.545436	0.392427	0.208947	0.791053

Grade D	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.816289	0.977778	0.731532	0.473531	0.526469
Decision Forest	0.408611	0.675745	0.366184	0.226169	0.773831
Neural Network Regression	0.497176	0.718605	0.445553	0.25577	0.74423

Grade E	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination

Linear Regression	1.284342	1.601835	0.81075	0.639658	0.360342
Decision Forest	0.597067	0.972431	0.376903	0.235739	0.764261
Neural Network Regression	0.731612	1.050325	0.461835	0.275017	0.724983

Grade F	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.967413	1.437962	0.697032	0.528946	0.471054
Decision Forest	0.546625	0.946123	0.39385	0.228988	0.771012
Neural Network Regression	0.704689	1.078992	0.507737	0.29782	0.70218

Grade G	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.967413	1.437962	0.697032	0.528946	0.471054
Decision Forest	0.546625	0.946123	0.39385	0.228988	0.771012
Neural Network Regression	0.687735	1.067815	0.495521	0.291681	0.708319

Prediction for algo-based clusters

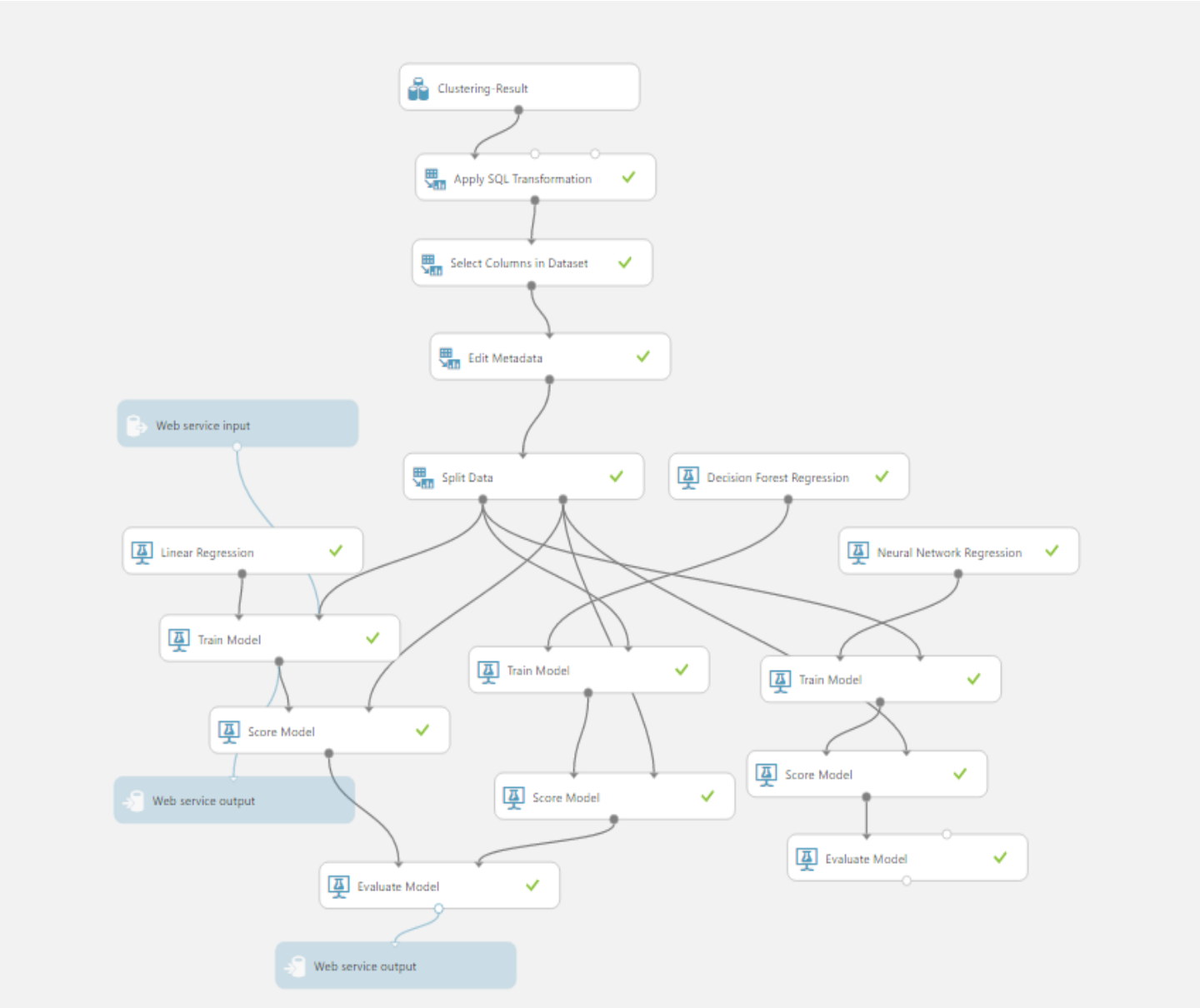
- term, grade, sub_grade, home_ownership, loan_status, purpose, application_type, emp_length is converted to categorical

- Training was done on 'int_rate' column.

- Below is the screenshot of all the algorithms that were used to train the model.

- Coefficient of Determination was used to determine the best model

- The steps were repeated for all the 3 clusters



Cluster 0	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.630395	0.928021	0.170642	0.039107	0.960893
Decision Forest	0.290409	0.560009	0.078611	0.014241	0.985759
Neural Network Regression	0.371853	0.594041	0.100657	0.016024	0.983976

Cluster 1	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Decision Forest	10.319968	12.61246	0.953268	0.883275	0.116725
Neural Network Regression	0.456668	0.791767	0.042183	0.003481	0.996519

Cluster 2	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.535601	0.683568	0.152743	0.02443	0.97557
Decision Forest	0.364592	0.57985	0.103975	0.017579	0.982421
Neural Network Regression	0.363844	0.533762	0.103761	0.014895	0.985105

Prediction for all data cluster

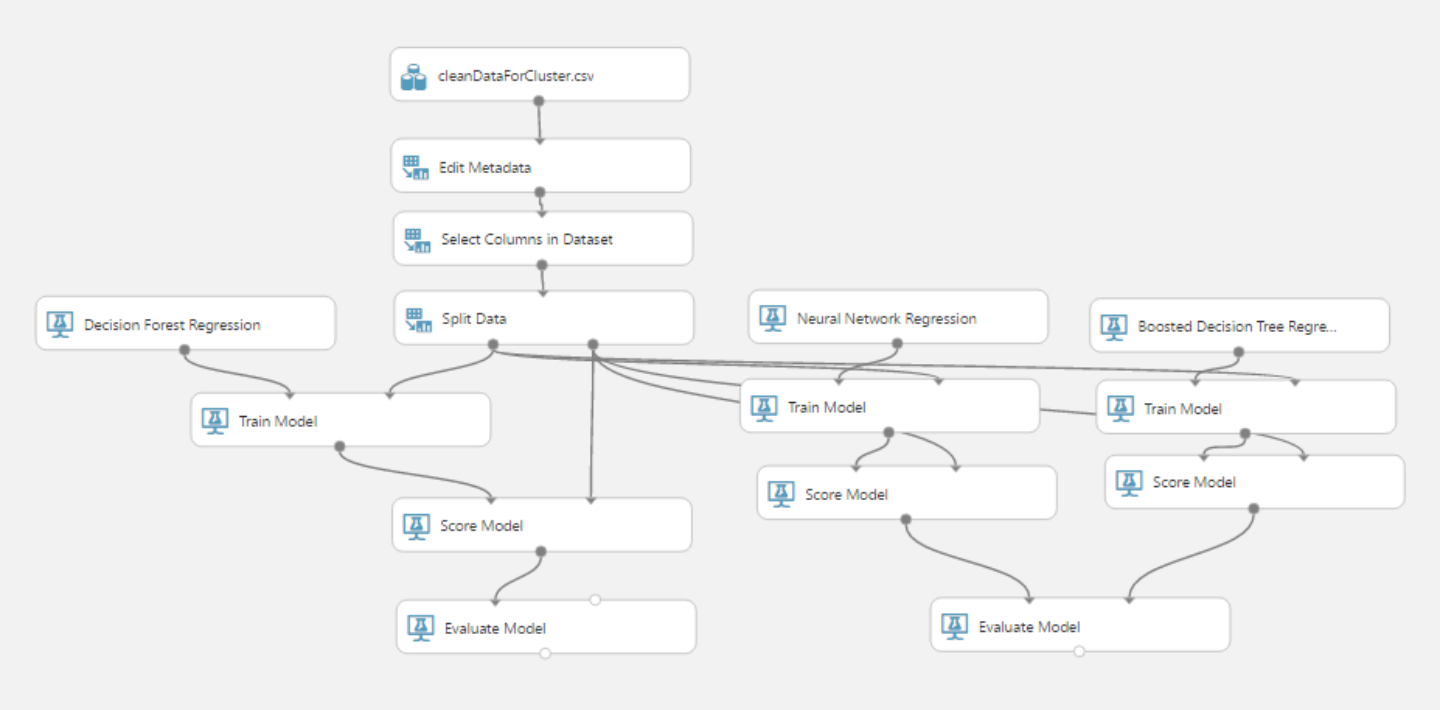
term, grade, sub_grade, home_ownership, loan_status, purpose, application_type, emp_length is converted to categorical

- Training was done on 'int_rate' column.

- Below is the screenshot of all the algorithms that were used to train the model.

- Coefficient of Determination was used to determine the best model

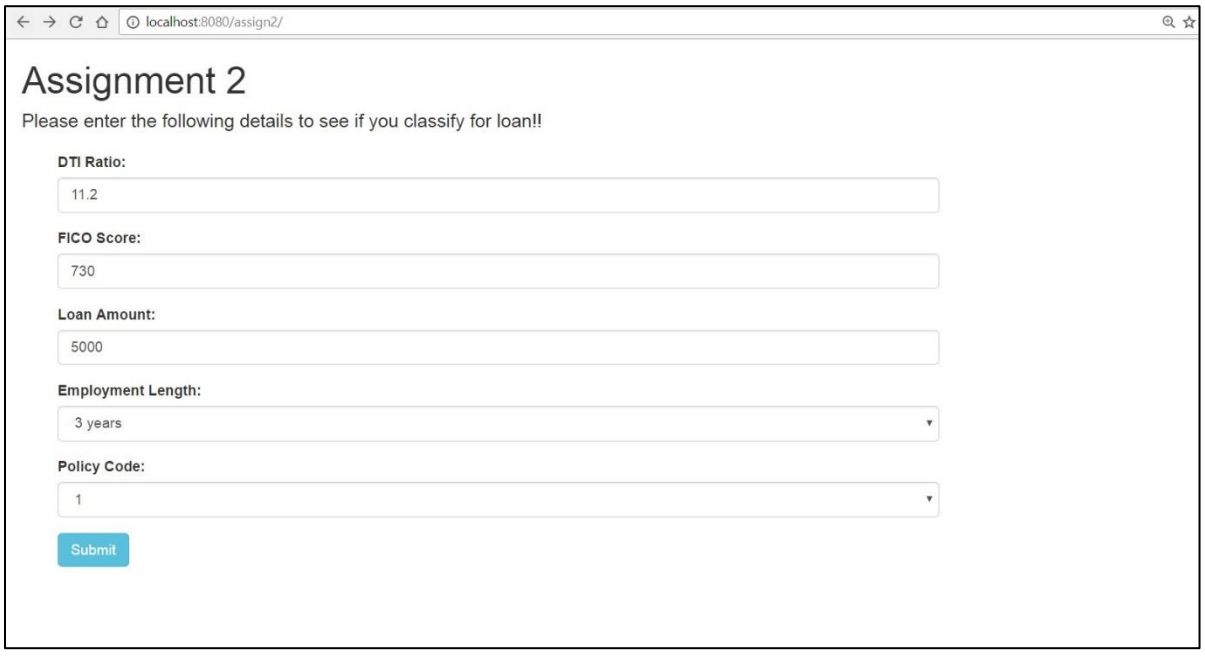
Cluster 0	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Decision Forest	0.312521	0.570922	0.086188	0.015582	0.984418
Neural Network Regression	0.366483	0.586578	0.10107	0.016448	0.983552
Boosted Decision Tree Regression	0.446942	0.633206	0.12326	0.019167	0.980833



Deployment

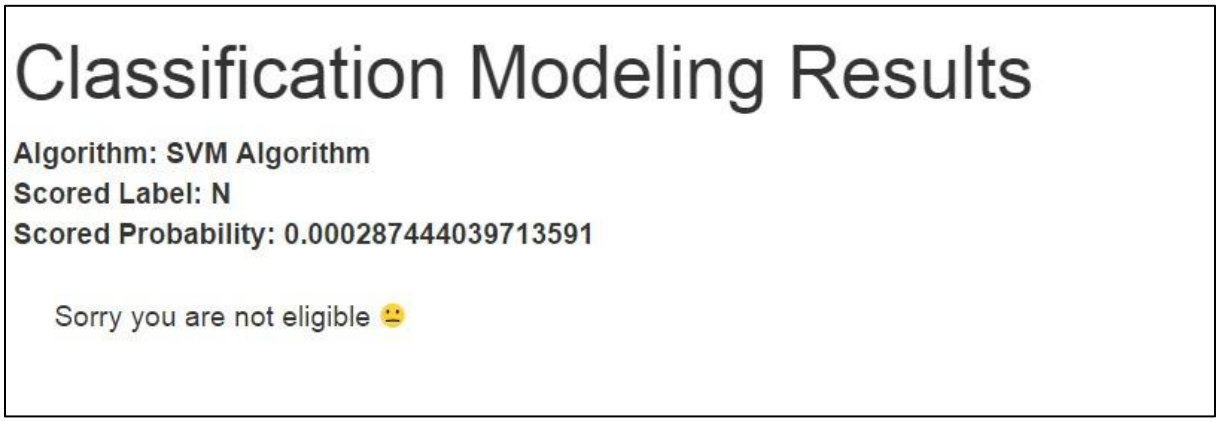
We have created a Spring MVC application that ask the user to input details and calls the Classification/Clustering/Prediction Rest APIs to provide the predicted interest rate. Following steps define the workflow of the application.

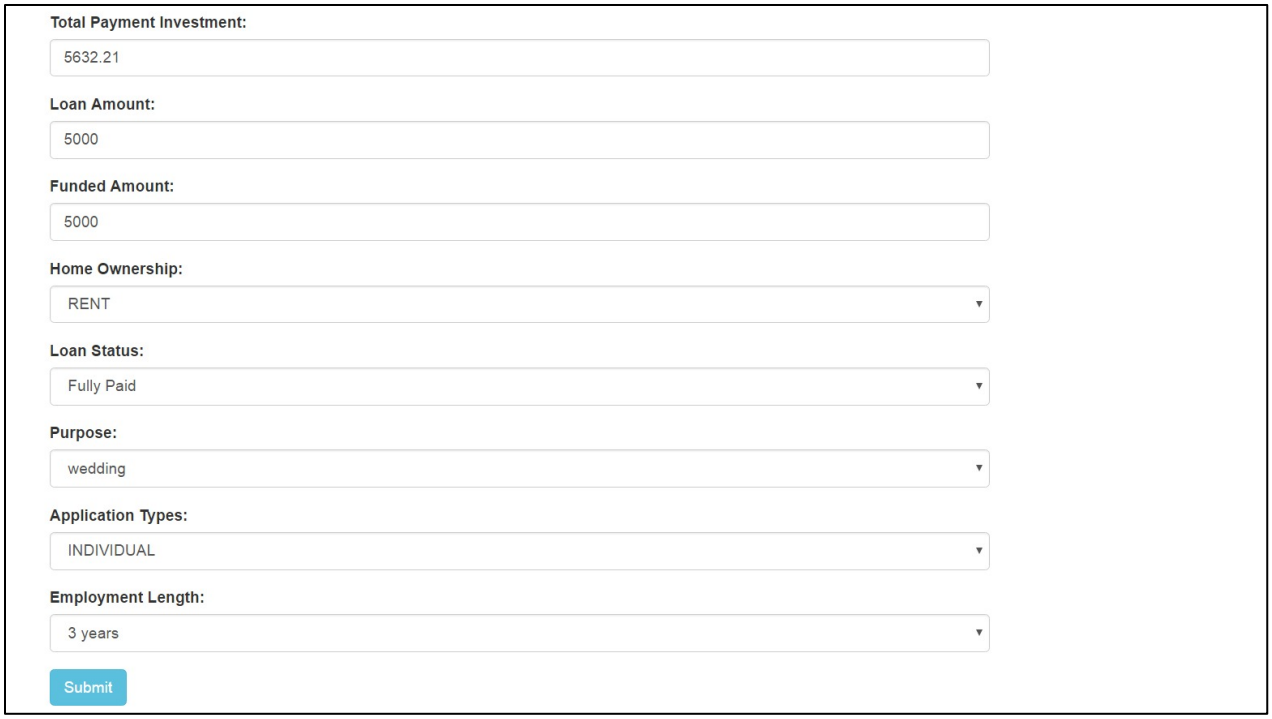
1. Input the dti, fico_score, employment_length, policy_code, loan_amount



2. Call the classification API and score the result in acceptedStatus variable

3. If acceptedStatus variable is 'N', there is no flow further, else we ask the user to enter more details for clustering



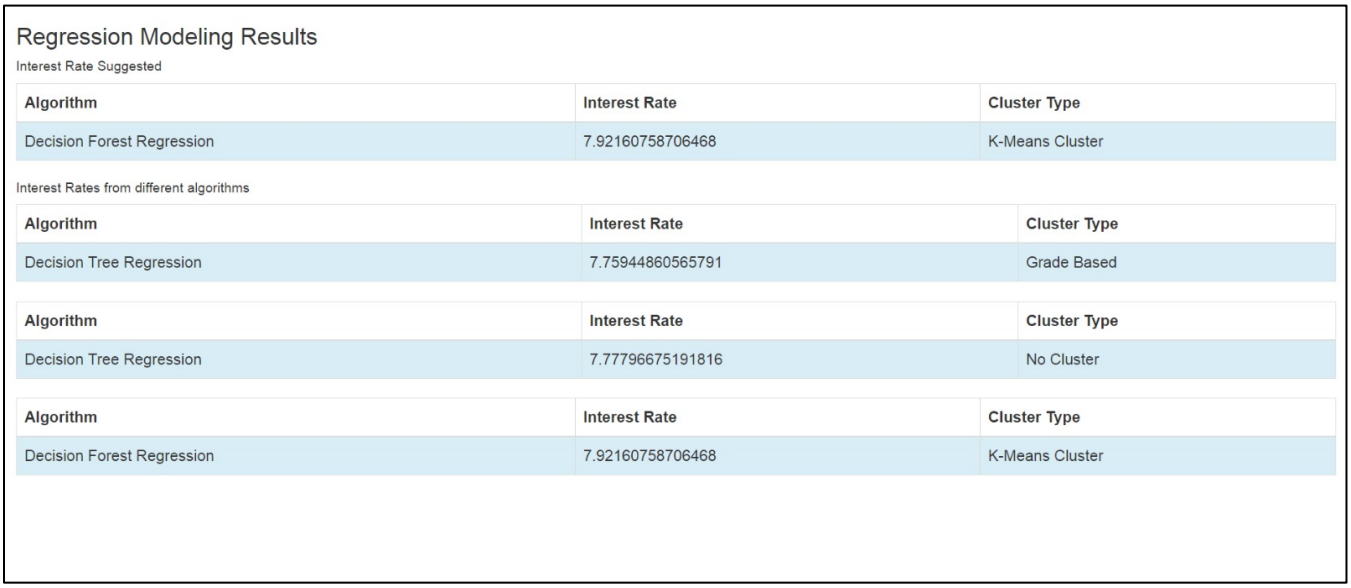


4. We check the grade, and call the grade based Prediction API

5. We call the clustering API, that returns us the cluster value. Then based on cluster value, we call the Prediction API

6. We call the prediction API where no clustering was done.

7. Lastly, we display all the results from all the algorithms and the maximum of all.



Regression Modeling Results

Interest Rate Suggested

Algorithm	Interest Rate	Cluster Type
Decision Forest Regression	7.92160758706468	K-Means Cluster

Interest Rates from different algorithms

Algorithm	Interest Rate	Cluster Type
Decision Tree Regression	7.75944860565791	Grade Based

Algorithm	Interest Rate	Cluster Type
Decision Tree Regression	7.77796675191816	No Cluster

Algorithm	Interest Rate	Cluster Type
Decision Forest Regression	7.92160758706468	K-Means Cluster

Contribution