CO544
MACHINE LEARNING & DATA MINING PROJECT

Group No 7

E/15/123: Wishma Herath
E/15/280: Pubudu Prematilaka
E/15/316: Suneth Samarasinghe

Deaprtment of Computer Engineering
Unviersity of Peradeniya

Contents

List of Figures

1 Introduction

1.1 Abstract

The purpose of this project was to test, analyze, and experiment with a defined dataset in the real world and to discover the use of machine learning and data mining to solve a classification problem. A data collection has been given to continue with the project. From this dataset, models have been developed to test the available machine learning algorithms. After a detailed analysis of the available algorithms and data examination, it was obvious that the challenge comes under both the linear model solver and the tree learning algorithms.Weka was used for visualization purposes because it offers a general-purpose framework for automated classification, clustering, and function selection – typical problems with data mining. The primary purpose of this project is to build and evaluate a data driven model utilizing appropriate tools and libraries such that it behaves well for unseen data.

1.2 Background

Process model development and optimization are some of the focus fields for machine learning implementations. The design of the process model consists of both the design and testing of the model. Validation is carried out by matching the data collected by the experiment with the data obtained from testing the models in actual conditions.

The validation is based on a review of the key characteristics obtained by the model based on the first criteria and the outcomes of the experiments. The results of this testing will indicate the consistency of the model. The model for this project is designed under consideration of the given number of attributes. After creating a basic model, variety of experiments and validations are done on different models to increase their performance.

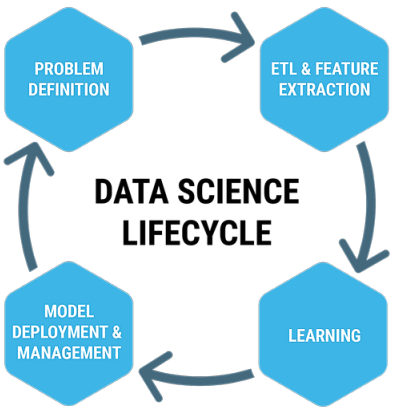

Figure 1: Life cycle of a ML model

2 Feature exploration

Features on the data set can be categorized as qualitative data and quantitative data.
In the given data set quantitative features are A2, A5, A7, A10, A12 and A14. Furthermore, these quantitative features can be categorized as discrete quantitative and continuous quantitative.

- Discrete quantitative - A7, A12, A14
- Continuous quantitative - A2, A5, A10

Since quantitative features are numeric in order to get get a general idea of the data distribution, several numerical estimations such as mean, standard deviation, percentiles are obtained

	A2	A5	A7	A10	A12	A14
count	542.000000	552.000000	552.000000	552.000000	552.000000	542.000000
mean	31.976661	4.884384	1100.827899	2.398678	2.614130	186.928044
std	12.211810	5.086809	5628.306468	3.551266	5.161073	182.590004
min	15.170000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	22.690000	1.083750	0.000000	0.165000	0.000000	70.750000
50%	28.540000	2.750000	5.000000	1.000000	0.000000	160.000000
75%	39.147500	7.551250	456.500000	3.000000	3.000000	280.000000
max	80.250000	28.000000	100000.000000	28.500000	67.000000	2000.000000

Figure 2: Numerical Statics of Training Data

Qualitative data can be categorized as nominal, ordinal and binary. In the given data set, since the feature names are not given, it is not fair to differentiate between nominal and ordinal data.

- Nominal/Ordinal qualitative - A1, A3, A4, A6, A9, A15
- Binary qualitative - A8, A11, A13

Following table shows some estimations based on qualitative features.

As it can be seen, there is a similarity between A3 & A4. This is described under the section of feature correlation.

Feature	No of unique entries	Unique entries	Top entry	Frequency
A1	2	a,b	b	379
A3	3	u,y,l	u	416
A4	3	g,p,gg	g	416
A6	14	w,q,c,x,i,d,e,aa,cc,ff,m,k,j,r	c	104
A8	2	True,False	False	306
A9	9	v,h,bb,ff,j,z,o,dd,n	v	310
A11	2	True,False	True	314
A13	2	True,False	False	305
A15	3	g,s,p	g	496

Table 1: Qualitative Features

2.1 Visualization

It is focused on individual qualitative features and evaluated its impact on the A16 (Success)

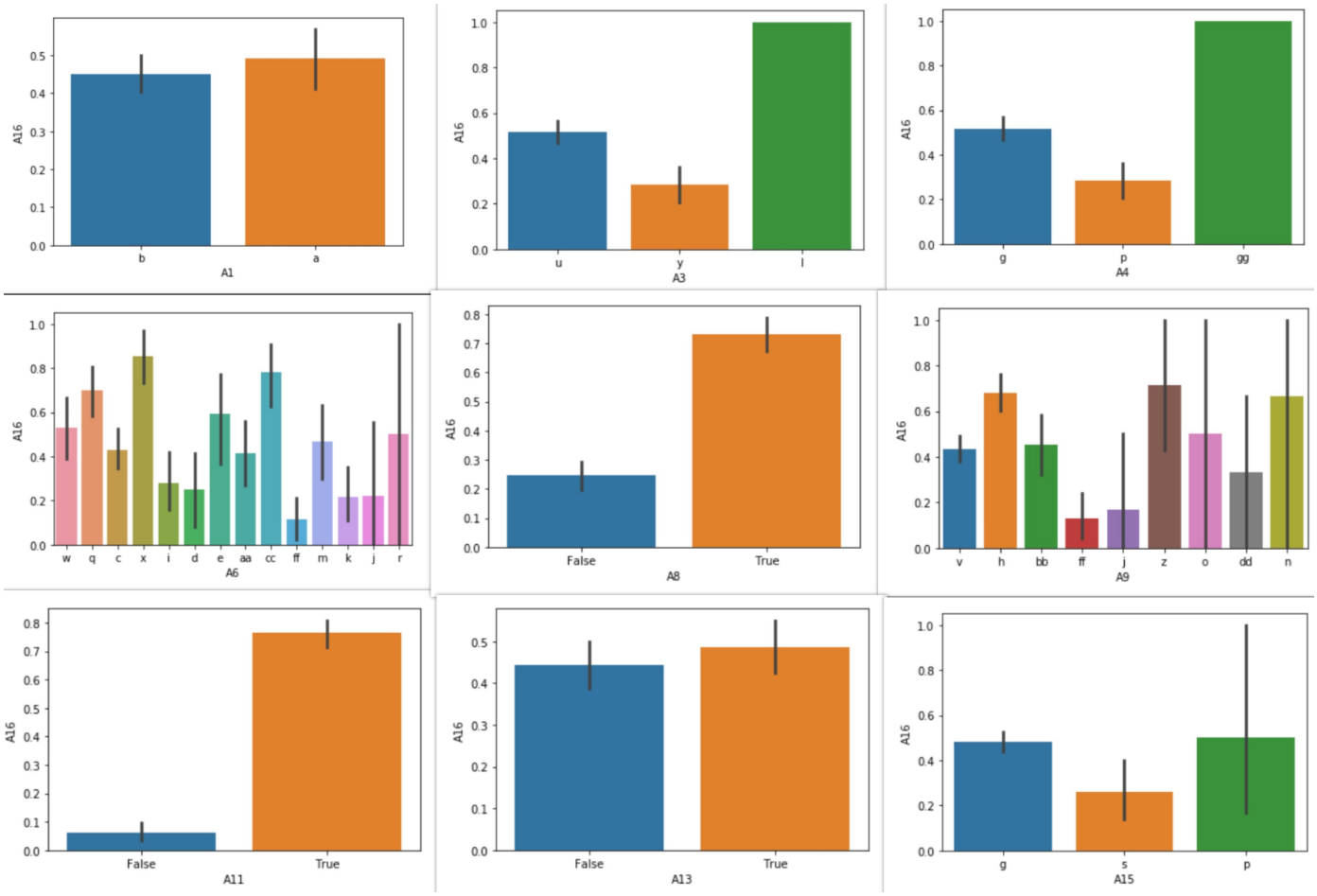

Figure 3: Individual feature impact on the Success

3 Pre-Processing

3.1 Missing Values

Missing(NULL) values play an important role in training data as well as new(unseen) data. Missing data could be included in some dominant features of the training data set. This could lead to a biased estimation model if they don't handle appropriately.

There is a important concept we followed on this project when handling the missing data. As the suitable method for handling missing data on the new data is to redo whatever methodologies followed on the test data. When handling missing data on the training data we followed some methodologies. In numeric data missing values filled using the mean value of that feature. In categorical data it is done using the most frequently appeared instance of that particular feature.

Whenever new data is predicted using the model, the model is designed such that, new data imputation is done by redoing what ever methodologies followed on the training data. Therefore, numerical missing values on training set imputed using corresponding mean values calculated on training set and categorical values imputed by selecting most frequent character on training set.

The reason to use training set to impute new data is to follow the same imputation procedure which was used to build the machine learning model which is already deployed.

Feature	**Missing Count Training**	**Missing Count New data**	**Imputed Value**
A1	8	4	b
A2	10	2	31.976
A3	4	2	u
A4	4	2	g
A5	0	0	-
A6	6	3	c
A7	0	0	-
A8	0	0	-
A9	6	3	v
A10	0	0	-
A11	0	0	-
A12	0	0	-
A13	0	0	-
A14	0	3	186.928
A15	0	0	-
A16	0	-	-

Table 2: Missing Values

3.2 Data Encoding

Since scikit-learn library is used to evaluate machine learning models every categorical variable encoded to numeric. Binary True, False features encoded using boolean (0=False, 1=True) values.

Since all the feature names are unknown(A1,A2,.....A16), it is not possible to make any conclusions on any feature at the training stage of the model. Therefore, all remaining categorical variables are encoded using **one-hot encoding** methodology in order to get unique binary variable for each category.

After using one-hot encoding on the training data, the same applied to new data as well. In this case column count inconsistency occurred between the training data and the testing data. Because some category instances on the training data has not appeared on new data. For example instances 'dd' and 'o' in A9 feature appears on training data but not on the new data. Therefore, to make the column count equal, zero columns added to the new data frame to equalize missing encodes.

3.3 Feature Scaling

Feature scaling is a technique to standardize the independent features present in the data to a fixed range such that they do not add extra weight to the model.
In this project sklearn MinMaxScaler and StandardScaler with is used alternatively by spectating to get the best performing model. Thereafter, different machine lea ring algorithms applied on the model and evaluated

3.4 Feature Correlation

There is no significant correlation between features. Following figure shows the heat map for correlation threshold of 0.0000001.

Though there is no correlation indicated on the heat map, by the inspection of the data columns it could be seen that features A3 and A4 are highly correlated.

A3		**A4**	
Feature	**Frequency**	**Feature**	**Frequency**
u	430	g	430
y	130	p	130
l	2	gg	2

As it can be seen, frequency distribution is same. Feature A3 and A4 has the same individual impact on the training model. Therefore, it is possible to

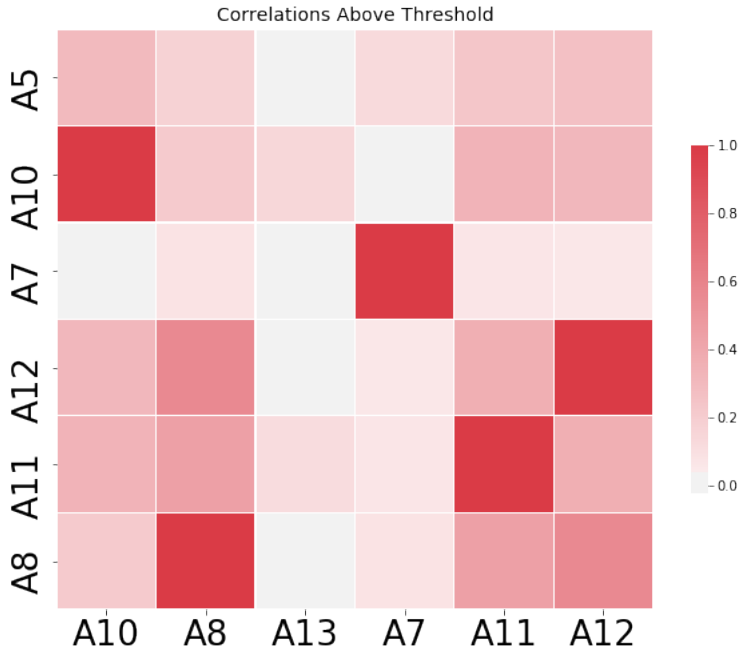

Figure 4: Correlation heat map for 0.0000001 threshold

neglect either A3 or A4.

4 Performance

4.1 Models

Analysis algorithms used in the project basically divided as regression and tree models in order to succeed the classification goal. Following Machine learning models are used in analysis.

Model Name	**Algorithm**
model_1	Gaussian Naive Bayes Classifier
model_2	K-Nearest Neighbors Classifier
model_3	Random Forest Classifier
model_4	Support Vector Classifier
model_5	Logistic Regression Classifier
model_6	XGBoost Classifier
model_7	Gradient Boosting Classifier
model_8	Extra Trees Classifier
model_9	AdaBoost Classifier

Table 3: Models

4.2 Metrics

Multiple methodologies used to measure the performance of the final model. The most common techniques are measuring accuracy, precision and recall.
In order to measure these parameters, training data set is divided randomly in to two sets as training set and testing set. Training set includes 80% and testing set included 20% of the data.

In addition to that, cross validation using 10 folds. In this method, model trained using 9 folds of the training data. The resulting model is validated on the remaining part of the data.The performance measure reported by k-fold cross-validation is then the average of the values computed in the loop.

Accuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

F1-Score

Precision is a good measure to determine when the costs of False Positive is high. Recall actually calculates how many of the Actual Positives our model capture through labeling it as Positive (True Positive). Since it is hard to compare models looking at two metrics we used the F1 score for comparisons.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{Precision * Recall}{Precision * Recall}$$

MAE

Mean Absolute Error, also known as MAE, is one of the many metrics for summarizing and assessing the quality of a machine learning model.

$$MAE = \frac{\sum_{i=1}^{n} abs(y_i - \lambda(x_i))}{n}$$

Cross-validation score

The above metrics alone are typically not enough information to make this decision. Because accuracy they do not represent models overfit to the data set. In order to limit problems like overfitting, underfitting and get an insight into how the model will generalize to an independent data set, we used a cross-validation score and it estimates how accurately a predictive model will perform in practice. Another reason is the limitation of the number of training examples used for training. So, we use 10-fold cross-validation to evaluate machine learning models.

4.3 Results

Model	Accuracy	Fscore	MAE	CorssValid
model_1	0.825301	0.815287	0.174699	0.78 (+/- 0.18)
model_2	0.801205	0.784314	0.198795	0.83 (+/- 0.14)
model_3	0.855422	0.851852	0.144578	0.86 (+/- 0.14)
model_4	0.843373	0.843373	0.156627	0.84 (+/- 0.09)
model_5	0.843373	0.845238	0.156627	0.87 (+/- 0.10)
model_6	0.855422	0.846154	0.144578	0.85 (+/- 0.12)
model_7	0.849398	0.838710	0.150602	0.85 (+/- 0.18)
model_8	0.825301	0.807947	0.174699	0.83 (+/- 0.13)
model_9	0.819277	0.810127	0.180723	0.82 (+/- 0.16)

Table 4: Results

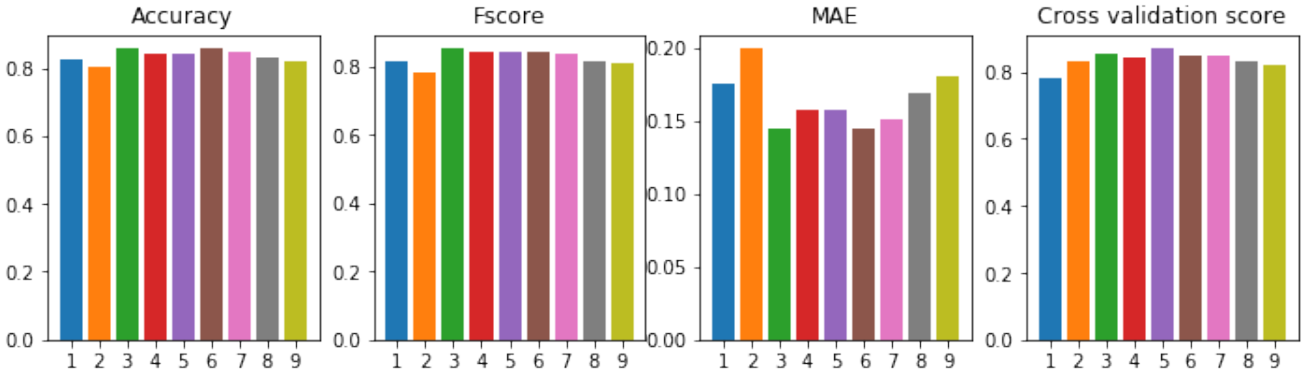

Figure 5: Performance Measure

	Predicted Negative	Predicted Positive
Actual Negative	69	19
Actual Positive	7	17

Table 5: Confusion matrix for Logistic Regression model

4.4 Final model with Conclusion

According to the accuracy results, models 3, 4, 5, and 6 have performed well. Among them, the best accurate models are model 3: Random Forest Classifier and model 6: XGBoost Classifier.
According to the f1 score results, models 3, 4, 5, 6, and 7 have performed equally well.
According to the Mean Absolute Error results, models 3, 4, 5, 6, and 7 have less MEA value showing that the overall predictions are much closer to the given actual value.
According to the cross-validation scores, some models that were good in previous metrics have not reduced their performance here. That is a sign of models overfits to the dataset. But, model 5 shows a better cross-validation score which means a good generalization to the examples.

After exploring different feature selection and parameter tunings for the above 9 models we managed to plot above 4 metrics for each. Then, considering all of their performances while majorly concerning the cross-validation scores, we decided to take Logistic Regression Classifier as our final model.