DATA MINING AND MACHINE LEARNING APPLICATIONS

Trinita John Peter
Msc Data Analytics
National College of Ireland, NCI
Dublin, Ireland
x23266848@student.ncirl.ie

***Abstract*—This report produces a portfolio of analyses on two numeric datasets and one textual dataset, exploring the application of Machine Learning and Statistical methods. The datasets comprise of domains from Apple TV streaming platform, Phishing information and Twitter entity sentiment, allowing to explore and implement most efficient methods. Relevant techniques are applied to each of data to study the performance of models and its predictions. Post fitting of models, error metrics are evaluated and visual comparison is made between one another to understand the outcome better and form conclusions. Statistical summary of evaluation on RMSE, MAPE, RSS, F1 score, Precision, Recall and residual visualizations are done to highlight the metrics briefly.**

I. INTRODUCTION

The project aligns with the framework **CRISP-DM** (Cross-Industry Standard Process for Data Mining) to provide a structured approach to build solutions. The objectives from business perspectives are discussed under each subsection below for each of the datasets.

A. DATASET 1 - APPLE STREAMING PLATFORM

The dataset I have chosen for study 1, is an Apple TV streaming entertainment data of shape [14568 rows × 8 columns]. This dataset contains significant columns that are highly useful for selecting the feature and target variable, which is IMDB rating in this case. The aim of this work is to predict the IMDB rating based on feature variables using Regressor models: Decision Tree with scaled input data and Random Forest. **Project objective** from business point of view is to improve the user engagement and predict viewer preferences by providing key insights on the factors that influence the APPLE TV subscribers. Success criteria are measured by the prediction accuracy. The predicted output is further compared with the test data to evaluate error metrics such as Root Mean Squared Error which calculates magnitude of error between actual and predicted values, Residual Sum of Squares which produces sum of squared residuals, Mean Absolute Percentage Error to measure accuracy by calculating the percentage error between predicted and actual values.

B. DATASET 2 - PHISHING INFORMATION

Phishing is a cybercrime where attackers pose as known or trusted entities and contact individuals through email, text or telephone and ask them to share sensitive information. This dataset contains phishing related information such as URL length, hostname, domain level and is of shape [10000 rows × 50 columns]. The goal of this work is to build a predictive model using Logistic Regression to classify websites as legitimate or phishing. **Business objective** is to develop a logistic regression model to detect fraudulent attacks. As part of data preparation, KNN, Baiyes and PCA are applied to enhance the model. Metrics such as Accuracy, Precision Recall, F1 score are evaluated.

C. DATASET 3 - TWITTER ENTITY SENTIMENT

This is an entity-level sentiment analysis dataset of twitter. Given a message and an entity, the task is to judge the sentiment of the message about the entity. There are three classes in this dataset: Positive, Negative and Neutral. We regard the irrelevant messages as Neutral. The **Business Success criteria** is measured on how well the model can determine the sentiment of tweets to improve customer feedback. This dataset is of shape [74682 rows × 4 columns]. Support Vector Machine Algorithm is implemented for the sentiment classification. Model performance is evaluated based on error metrics like F1 score, accuracy and precision.

II. RELATED WORK

A. DATASET 1 - APPLE STREAMING PLATFORM

The partitioning strategy in Decision trees are well able to handle input unscaled data. However quite a few studies reveal the impact of scaling on decision tree performance. Random Forest Regressors create a group of trees averaging their output to improve robustness and reduce overfitting.

Decision tree with scaled input data

1) Advantages: Scaling enhances interpretability in pipelines involving multiple models, in this case decision tree and linear regressor. Supporting works by [1] shows that scaling inputs improves in model performance.
2) Limitations: Empirical research shows limited benefits for decision trees as the scaling possibly does not affect split calculations [2].

Random Forest Regressor

1) Advantages : High robustness to overfitting as highlighted by Breiman's foundational work [3]. Effective in

handling high-dimensional datat with reduction techniques like PCA, per [4]

2) Limitations: Computation grows with the number of trees. Sensitivity to hyperparamter settinfs require careful tuning by [Hastie et al] [5].

B. DATASET 2 - PHISHING INFORMATION

Logistic Regression is widely used in binary classification tasks, including spam detection and medical diagnostics. Commonly applied in phishing detecition due to its interpretability.

KNN

1) Advantages : Common in image classification (Zhang et al., 2006z0) [6] and recommendation systems (Sarwar et al., 2001) [7]. Often used as a baseline for pattern recognition tasks, like handwriting recognition (Duda et al., 2000) [8].
2) Limitations: Sensitive to feature scaling and the choice of k (number of neighbors)

Naive Bayes and PCA

1) Advantages : Popular in text classification tasks such as sentiment analysis and spam detection. Effective for medical diagnosis tasks with categorical data [9]. The paper for PCA on Dimensionality reduction for visualization is discussed in [9].
2) Limitations: Struggles with imbalanced datasets unless specific techniques are applied [11]. The paper that discusses on limitation is on Turk & Pentland, 1991.

C. DATASET 3 - TWITTER SENTIMENT

Support Vector Machines have been extensively applied across different domains for classification and regression tasks.

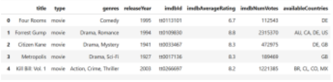
Applications of SVM

1) Advantages : SVMs are used in Natural Language Processing (NLP) like spam detection, sentiment analysis and topic classification. Studies such as JOACHIMS (1998) [12] highlight SVMs excel in binary classification problems, making them suitable for categorization by [Cortes and Vapnik (1995)] [13].
2) Limitations: Burges (1998) discussed the scalibility issues of SVMs and proposed methods. [Guyon, I., Weston, J., Barnhill, S., Vapnik, V. (2002) [14] [15]].

III. DATA MINING METHODOLOGIES APPLIED

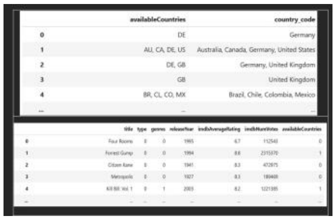
Data Mining is a technique widely used in businesses by analysts to extract hidden patters from large datasets using computational/statistical methods.

Best practices like copying the loaded CSV file into another dataframe object, checking for null values, discarding duplicate rows and dropping columns of not much value in the model prediction are adopted throughout. Dataframe object is then inspected using the describe() method which fetches total row count, top and frequent values from each column.



	title	type	genres	releaseYear	imdbId	imdbAverageRating	imdbNumVotes	availableCountries
0	Four Rooms	movie	Comedy	1995	tt0113101	6.7	112543	DE
1	Forrest Gump	movie	Drama, Romance	1994	tt0109830	8.8	2315370	AU, CA, DE, US
2	Citizen Kane	movie	Drama, Mystery	1941	tt0033467	8.3	472975	DE, GB
3	Metropolis	movie	Drama, Sci-Fi	1927	tt0017136	8.3	189469	GB
4	Kill Bill: Vol. 1	movie	Action, Crime, Thriller	2003	tt0266697	8.2	1221385	BR, CL, CO, MX

Fig. 1. Original dataset



	availableCountries	country_code
0	DE	Germany
1	AU, CA, DE, US	Australia, Canada, Germany, United States
2	DE, GB	Germany, United Kingdom
3	GB	United Kingdom
4	BR, CL, CO, MX	Brazil, Chile, Colombia, Mexico
...	...	...

	title	type	genres	releaseYear	imdbAverageRating	imdbNumVotes	availableCountries
0	Four Rooms	0	0	1995	6.7	112543	0
1	Forrest Gump	0	0	1994	8.8	2315370	1
2	Citizen Kane	0	0	1941	8.3	472975	0
3	Metropolis	0	0	1927	8.3	189469	0
4	Kill Bill: Vol. 1	0	1	2003	8.2	1221385	1
...	...	...	...	...	...	...	...

Fig. 2. Processed dataset

A. DATASET 1 - APPLE STREAMING PLATFORM

- Column ‘availableCountries‘ contains country code which would be more useful in interpretation when converted into respective country names using the python library ‘pycountry‘. This categorical column is then transformed into numerical boolean values based on the condition - fill up with 0 if number of countries is more than 3 in a cell, else 0. Similarly ‘genres‘ column is filled with 0s and 1s if the number of genres is more than 2, otherwise 0. ‘unique()‘ method is applied to column ‘type‘ to identify how many entertainment types are recorded. The results produced the following output ’movie’, ’tv’. Hence using the ‘map()‘ function, values of 0 and 1 are passed into the dictionary for each of type keys. Fig 1 shows original dataset. Fig 2 shows processed dataset.
- A histogram is plotted to analyze the distribution of IMDB votes in the dataset. Fig 3 shows that movie and tv receive moderate number of votes and the distribution is symmetrical. Bar chart is plotted to understand the votes received for entertainment type. In this plot, the y-axis values are logarithmically scaled to manage wide range of vote counts, shown in Fig 4. The plot heights of both the bars are similar indicating that both types received comparable number of votes.
- RandomForestRegressor model is applied to the numerical and categorical columns which are passed as parameters to ColumnTransformer. This tool is used when

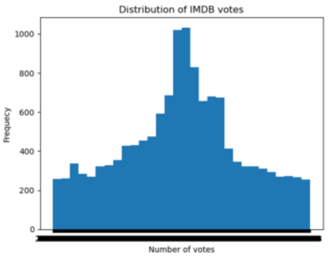


Fig. 3. Distribution of IMDB votes

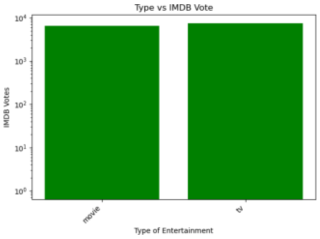
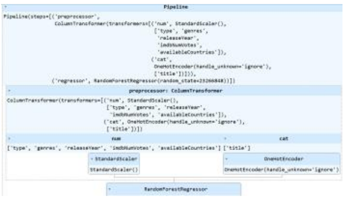


Fig. 4. Type vs IMBD votes



```
Pipeline
Pipeline(steps=[('preprocessor',
                 ColumnTransformer(transformers=[('num', StandardScaler(),
                                                  ['type', 'genres',
                                                   'releaseYear',
                                                   'imdbNumVotes',
                                                   'availableCountries']),
                                                 ('cat',
                                                  OneHotEncoder(handle_unknown='ignore'),
                                                  ['title'])])),
                ('regressor', RandomForestRegressor(random_state=23266848))])
preprocessor: ColumnTransformer
ColumnTransformer(transformers=[('num', StandardScaler(),
                                 ['type', 'genres', 'releaseYear',
                                  'imdbNumVotes', 'availableCountries']),
                                ('cat', OneHotEncoder(handle_unknown='ignore'),
                                 ['title'])])
num                                                                    cat
['type', 'genres', 'releaseYear', 'imdbNumVotes', 'availableCountries'] ['title']
StandardScaler                         OneHotEncoder
StandardScaler()                       OneHotEncoder(handle_unknown='ignore')
RandomForestRegressor
```

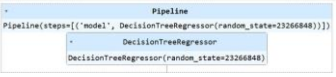
Fig. 5. Pipeline workflow - RF

the dataset contains heterogenous data to apply nuerical and categorical transformations. The pipeline method encapsulates multiple steps as shown in Fig 5.

- DecisionTreeRegressor model is applied to the scaled dataset using StandardScaler class. Train and test split method has parameters of this scaled data. The instance of Pipeline class is instantiated that contains pre-processing and model prediction steps. The fit method is then applied to the training dataset as shown in Fig 6.

B. DATASET 2 - PHISHING INFORMATION

- Histograms and count plots are plotted to visualize the distribution of values in columns UrlLength, FrequentDomainNameMismatch, PathLevel and NumSensitiveWords. Class label 0 and 1 indicates non-phishing URLs and phish URLS repectively. Fig 7, represents X-axis with length of URLs ranging from 0-250 characters. The histogram shows a peak around 50 characters. Hence the attackers attempt to use shorter URL lengths to make the information seem less suspicious.
- The bar chart titled FrequentDomainNameMismatch shows the distribution of mismatch names across labels.



```
Pipeline
Pipeline(steps=[('model', DecisionTreeRegressor(random_state=23266848))])
DecisionTreeRegressor
DecisionTreeRegressor(random_state=23266848)
```

Fig. 6. Pipeline workflow - DT

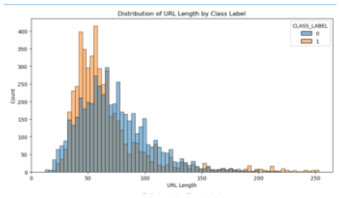


Fig. 7. Distribution of URL Length by Class Label

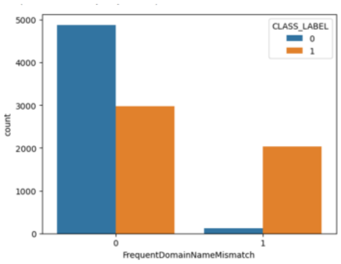


Fig. 8. FrequentDomainNameMismatch

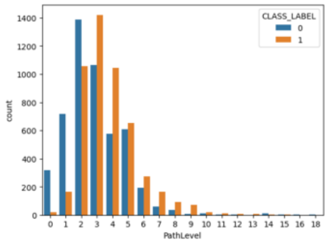


Fig. 9. PathLevel

Legitimate URLs (class label 0) rarely have domain mismatches while Phishing URLs often do (class label 1). This visualization in Fig 8 shows the importance of domain name consistencies in detecting phishing name URLs. The bar chart titled PathLevel shows the distribution of URL path levels across 2 levels. Phishing URLs has multiple count with path levels 2 or more being common in class 1 shown in Fig 9. The bar chart titled NumSensitiveWords compares the distribution of sensitive words across class lables 0 and 1. This visualization highlights that majority of URLs do not contain sensitive words as shown in Fig 10.

- The bar chart titled ”phishing vs non-phishing” shows equal number of phishing and legitimate URLs present in the dataset. This balance is crucial for training machine learning models, ensuring that the models learn to distinguish between the 2 classes without bias in Fig 11.

- Dimensionality reduction - PCA reduces the dimensioanlity into 3 components that capture the most variance in the data by visualizing the plot in 3-D structure. The clusters in Fig 12 are well separated, indicating good

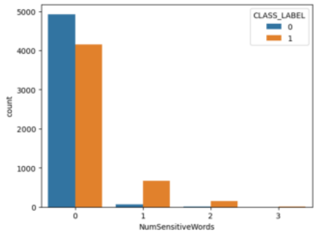


Fig. 10. NumSensitiveWords

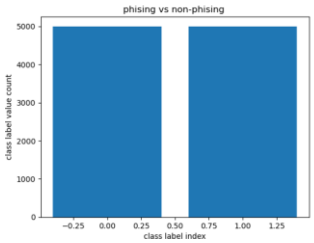


Fig. 11. Phishing vs non-phishing

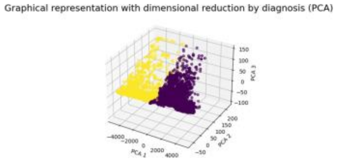


Fig. 12. Graphical representation with dimensional reduction by diagnosis (PCA)

class separability in reduced dimensionality. If the points from different class overlap significantly, it suggests that classes are not distinguished based on 2 components alone. The scatter plot in Fig 13 is a visual representation of PCA with 2 components, PCA 1 and PCA 2. The blue points represent phishing and orange points represent non-phishing. The distinct clustering thus shows that the implementation of PCA has successfully reduced the dimensionality while preserving class distinction, highlighting the effectiveness of PCA.

- The visualization in Fig 14 shows the correlation matrix between different feature variables related to URLs. The white line that runs diagonally represent perfect correlation between each feature. Strong positive correlation between variables like PctExtResourceUrls and PctExtNullSelfRedirectHyperlinksRT. Negative correlation exists between variables NumAmpersand and UrlLengthRT. The graph in Fig 15 illustrates positive correlation between features in the dataset. Features like PctExtResourceUrlsRT, NumNumericChars, SubdomainLevel, DoubleSlashInPath, and NumHash have lighter colors, indicating they are highly important in determining the class labels. The heatmap is shown in Fig 16.
- The graphs in Fig 17 show the confusion matrix for a binary classification problem indicating that the model shows high accuracy for True Positives and True Negatives with low misclassification rates.

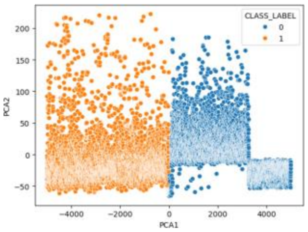


Fig. 13. PCA with n-components=2

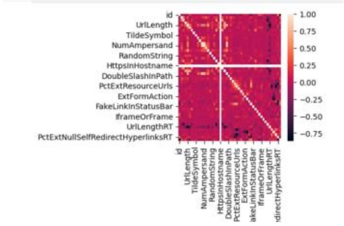


Fig. 14. Correlation Matrix

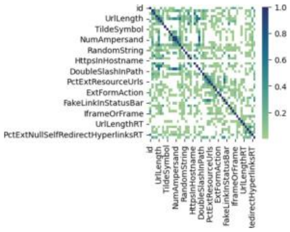


Fig. 15. Positive Correlation Matrix

- Various model objects like Decision Tree Classifier, Logistic Regression, KNN and SVC are created for which the model is trained, fitted and values are predicted.

C. *DATASET 3 - TWITTER SENTIMENT ANALYSIS*

- A bar chart titled ”Countplot for Target” is potted that displays the count of different sentiment as shown in Fig 18. There are 20,000 positive tweets, 22,500 negative tweets, around 17,500 neutral tweets and so on.
- A histogram is plotted to represent number of occurences in each of sentiment category. Similar to the previous chart, the negative sentiment counts are the highest. A countplot is plotted to understand the Entity 2298 and 2304 that refer to Nvidia and CS-GO occurrence. The pie chart in Fig 19 shows 30.2 percentage of the tweets

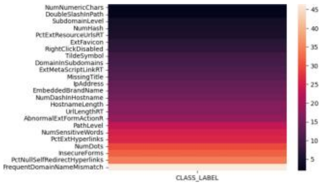


Fig. 16. Feature Importance

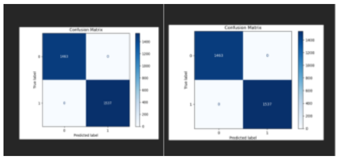


Fig. 17. Confusion Matrix

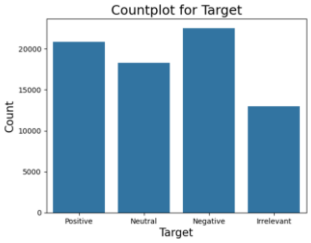


Fig. 18. Countplot for Target

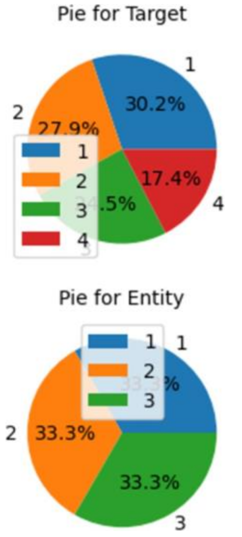


Fig. 19. Pie Chart

are classified positive. distribution of sentiment categories with target variable.

- The scatterplot in Fig 20 titled ”Scatterplot between ID and Entity” provides visual distribution of ID values across various entities. This is helpful in understanding the relationship between ID values and entities and can help in identifying any patterns or anomalies within data. Entities are spread across the entire range of ID values (0 to 12000) indicating a diverse distribution of data points.A visual representation of the most frequent words in the text data shown in Fig 21 shows is word cloud generated from processed text data where words like ”new”, ”games” point to topics of interest confirming that this dataset focuses on text analytics related to a game ”Borderlands”
- The dataset is then split into train and test to apply train test split method. 3 different models like Decision Tree Classifier, Naive Baiyes and Random Forest Classifier are applied to train the model. Heatmap is then generated of a confusion matrix for Decsision Tree Classifier. The high values in the diagonal indicates that this model performs well for most class labels. Heat map for Naive Baiyes indicate that this classifier performs well for classes 1 and 3 with high number of correct classifications. Heat map for Random Forest Classifier shows performance of this model by comparing actual versus predicted classifications. High numbers displayed along the diagonal indicates strong performance.

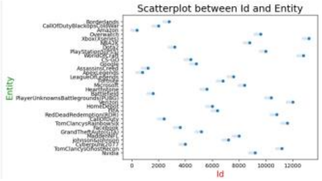


Fig. 20. Scatterplot between ID and Entity

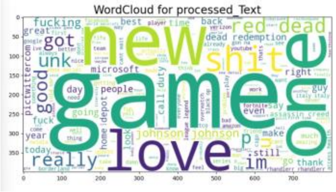


Fig. 21. Word Cloud

IV. EVALUATION OF ERROR METRICS

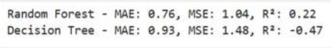
Visualizing errors can provide valuable insights into model performance and help diagnose issues like overfitting, underfitting, or systematic biases. Statistical comparison of errors are implemented in 2 functions for each of the Regressor models.

A. DATASET 1 APPLE TV STREAMING PLATFORM

- The errors like MSE, MAE, R² are computed for both the models in 2 functions separately. The results are shown in Fig 22. In the bar chart, MAPE error is magnified for the Decision Tree Regressor while that Random Tree Regressor performs well implying more accurate and reliable prediction shown in Fig 23.

B. DATASET 2 PHISHING INFORMATION

- Accuracies calculated for models SVC, logistic regression and Decision Tree are shown in Fig 24 below. The ROC curve in Fig 25 shows that Support Vector



```
Random Forest - MAE: 0.76, MSE: 1.04, R²: 0.22
Decision Tree - MAE: 0.93, MSE: 1.48, R²: -0.47
```

Fig. 22. Statistical Errors

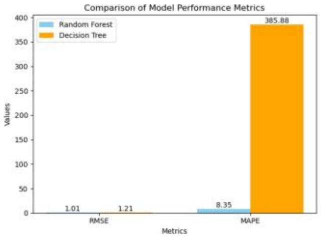
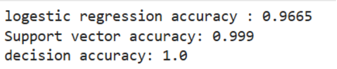
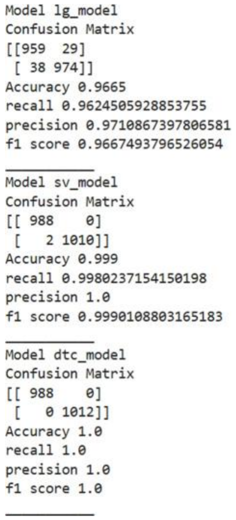


Fig. 23. Error Visualization



```
logestic regression accuracy : 0.9665
Support vector accuracy: 0.999
decision accuracy: 1.0
```

Fig. 24. Accuracy of Models



```
Model lg_model
Confusion Matrix
[[959  29]
 [ 38 974]]
Accuracy 0.9665
recall 0.9624505928853755
precision 0.9710867397806581
f1 score 0.9667493796526054
__________
Model sv_model
Confusion Matrix
[[ 988    0]
 [   2 1010]]
Accuracy 0.999
recall 0.9980237154150198
precision 1.0
f1 score 0.9990108803165183
__________
Model dtc_model
Confusion Matrix
[[ 988    0]
 [   0 1012]]
Accuracy 1.0
recall 1.0
precision 1.0
f1 score 1.0
__________
```

Fig. 25. Overall Error Metrics

Machine and Decision Tree models outperform Logistic Regression with high AUC values, perfect classification and effectiveness. The image in Fig 26 shows statistical performance metrics for 3 models that are evaluated using Confusion Matrix, F1 score, accuracy, recall, precision. LG model shows the model classifies correctly in most instances. SVM also indicates similar performance, while Decision Tree classifer shows the model accuracy to be comparatively higher. Hence concluding that DTC is the best choice for this problem.

C. DATASET 3 TWITTER ENTITY SENTIMENT

- For the 3 models Decision Tree Classifier, DTC Naive Bayes and Random Forest Classifier - error metrics like score, confusion matrix, accuracy and weighted average like precision, recall, f1-score, support factors are calculated to identify best performing training models. The image Fig 28 shows performance metrics for DTC. Score of 0.975 suggests that decision tree model performs well on training set. ”accuracy-dt” on the test set is 0.813 correctly predicting samples. Classification report overall suggests 81 percent model performance with confusion matrix in Fig 27, of 75 percent instances correctly classified as class 0.

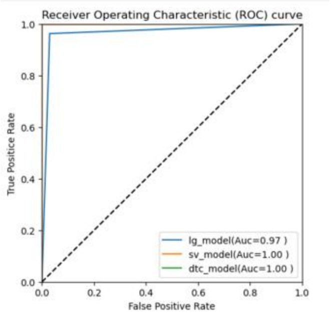


Fig. 26. ROC curve

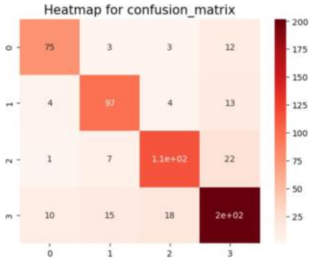
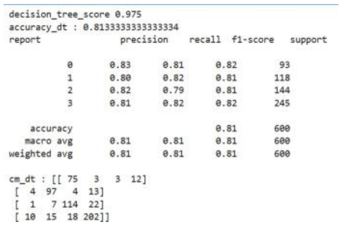


Fig. 27. Heatmap - DTC



```
decision_tree_score 0.975
accuracy_dt : 0.8133333333333334
report                precision    recall  f1-score   support

           0       0.83      0.81      0.82        93
           1       0.80      0.82      0.81       118
           2       0.82      0.79      0.81       144
           3       0.81      0.82      0.82       245

    accuracy                           0.81       600
   macro avg       0.81      0.81      0.81       600
weighted avg       0.81      0.81      0.81       600

cm_dt : [[ 75   3   3  12]
 [  4  97   4  13]
 [  1   7 114  22]
 [ 10  15  18 202]]
```

Fig. 28. Error Metrics - DTC

- The image in Fig 30 highlights performance metrics for a Naive Bayes Classifier, NBC . Training score of 0.85 shows model performing reasonably well and accuracy-dt on test set is 80 percent. Classification report provides metrics for class. Overall the weighted average for precision, recall and F1-score ranges in the 80s as shown in Confusion Matrix as well in Fig 29.
- The error metrics in Fig 32 presents the performance of Random Forest Classifier, RFC. Training score obtained for this model is 0.97 and accuracy-dt is 91 percent, indicating highest performance in classifying of the 3 models implemented. Confusion matrix compares the actual and predicted labels for all classes. Diagonal entries represent correct predictions showing 81 instances of Class 0 were

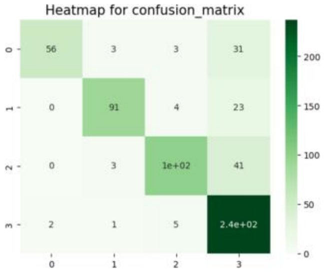
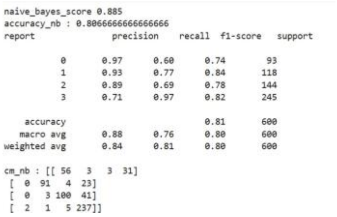


Fig. 29. Heatmap - NBC



```
naive_bayes_score 0.885
accuracy_nb : 0.8066666666666666
report              precision    recall  f1-score   support

           0       0.97      0.60      0.74        93
           1       0.93      0.77      0.84       118
           2       0.89      0.69      0.78       144
           3       0.71      0.97      0.82       245

    accuracy                           0.81       600
   macro avg       0.88      0.76      0.80       600
weighted avg       0.84      0.81      0.80       600

cm_nb : [[ 56   3   3  31]
 [  0  91   4  23]
 [  0   3 100  41]
 [  2   1   5 237]]
```

Fig. 30. Error Metrics - NBC

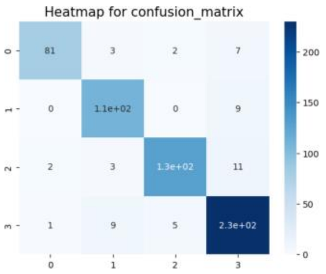


Fig. 31. Heatmap - RFC

correctly classified as class 0. Heat map for this is shown in Fig 31.

V. CONCLUSIONS AND FUTURE WORKS

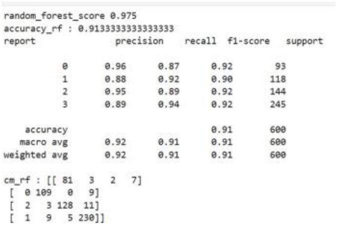
A. SUMMARY OF FINDINGS ON MODELS

Random Forest Regressor employed for Dataset 1, demonstrated excellent performance with high accuracy and very error rates. Decision Tree Classifier employed for Dataset 2, worked relatively well outperforming logistic regression model with high accuracy scores. The model Decision Tree that shows comparatively good overall performance on less misclassification error and computation against Naive Baiyes and KNN.

B. LIMITATIONS AND FUTURE WORK

Data Quality - Accuracy of these models are dependent on quality and quantity of the data. Noisy data when applied to training models can disrupt the performance leading to significant impact on results.

Feature-Engineering - Effectiveness of these models are limited by features used. Sometimes, poor choice of features can lead to performance rates below par.



```
random_forest_score 0.975
accuracy_rf : 0.9133333333333333
report              precision    recall  f1-score   support

           0       0.96      0.87      0.92        93
           1       0.88      0.92      0.90       118
           2       0.95      0.89      0.92       144
           3       0.89      0.94      0.92       245

    accuracy                           0.91       600
   macro avg       0.92      0.91      0.91       600
weighted avg       0.92      0.91      0.91       600

cm_rf : [[ 81   3   2   7]
 [  0 109   0   9]
 [  2   3 128  11]
 [  1   9   5 230]]
```

Fig. 32. Error Metrics - RFC

C. EXTENSIONS

Hyperparameter Tuning - Conducting this tuning method for each model improves the model performance.

Ensemble Methods - This method was applied in this project to observe the performance variation between models.

D. RESEARCH QUESTIONS

Model Selection - Random Forest Regressor, Decision Tree Classifier and Decision Tree Classifier emerged strong due to their high accuracy and robustness.

Computation - As dataset increases, so does the intensity of computation. KNN and Gradient boosting and PCA helped in enhancing the models.

REFERENCES

[1] Kotsiantis SB (2013) Decision trees: a recent overview. Artif Intell Rev 39:261–283

[2] Limitations of Decision Tree and Automatic Learning in Real World Scenario, December 1997.

[3] Bridging_Breiman's_Brook_From_Algorithmic_Modeling_to_Statistical, 23 Feb, 2021.

[4] Understanding_Random_Forests_From_Theory_to_Practice, October 2014.

[5] Zhang et al., 2006z0 for Hyperparametr settings, 2006. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[6] Efficient_kNN_Classification_With_Different_Numbers_of_Nearest_Neighbors, April 12, 2017.

[7] Itembased_Collaborative_Filtering_Recommendation_Algorithms, August 2001.

[8] Duda et al., 2000 – Appying K-Nearest Technique.

[9] Pang et al., 2002) and spam detection (Metsis et al., 2006) An Approach to Spam Detection by Naive Bayes Ensemble Based on Decision Induction

[10] Kononenko et al., 2001 Deep Study on Naïve Bayes and Principal Component Analysis.

[11] Jolliffe, (2002) Principal Component Analysis, PCA

[12] Turk & Pentland, 1991 Discusses on limitations of Baiyes and PCA

[13] Cortes and Vapnik (1995), Support Vector Machines

[14] Huang et al. (2002), Support Vector Machines for Classification Modelling.

[15] Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002), Limitations of Support Vector Machines in Classifications.