BIRZEIT UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

COMP4388: Machine learning

Assignment#2

Prepared by:

Aws Ayyash 1190680

Salah AlDin Dar AlDeek 1192404

Instructor: **Radi Jarrar**

13-jan-2022

Describe Data set

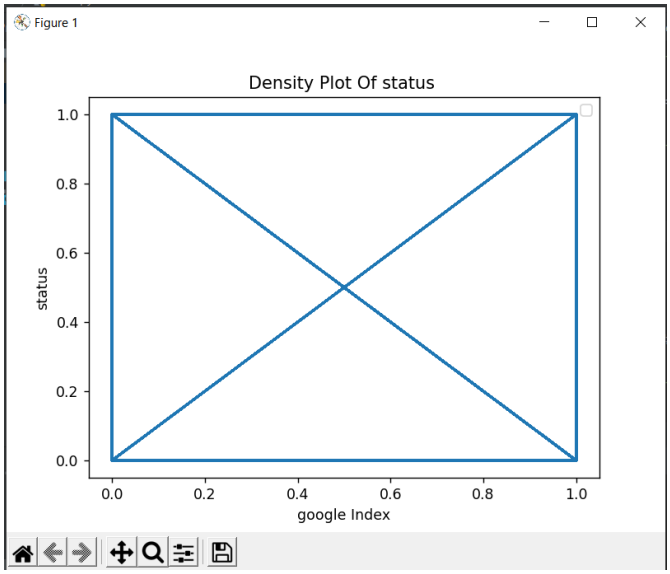
describe() is used to view some basic statistical details like count, mean, standard Deviation, min,max value and 1st,2nd,3rd quartile of all features in data set, These values help us to know the Range of False for every feature,And it reveals if there are some missing values in the file By looking at the numbers of values in each feature, Knowing the features that have no benefit from using them in the model, which can negatively affect the model.

We are just showing the first 9 values of the output, because the data set has a large number of features(76 features).

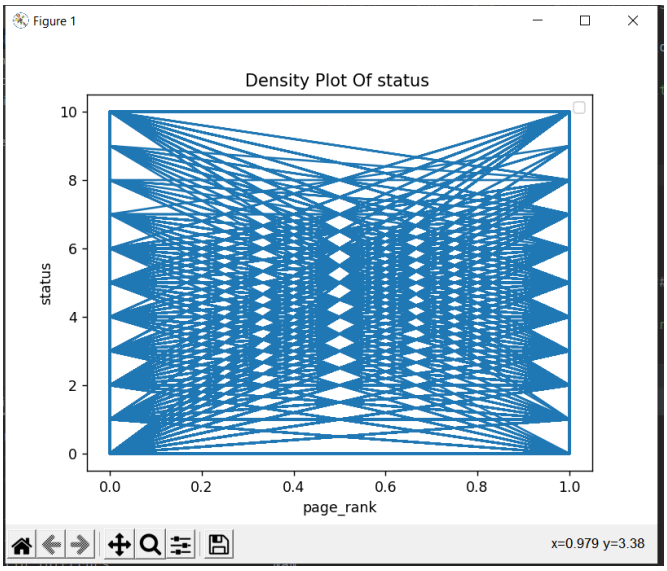
	l_url	l_hostname	ip	dots	hyphens	at	qm	and	or
count	8002.000000	8002.000000	8002.000000	8002.000000	8002.000000	8002.000000	8002.000000	8002.000000	8002.0
mean	60.832417	21.008873	0.147713	2.477381	0.987878	0.022744	0.138340	0.156086	0.0
std	53.258795	10.997099	0.354838	1.394384	2.092052	0.160404	0.358073	0.784652	0.0
min	12.000000	4.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.0
25%	33.000000	15.000000	0.000000	2.000000	0.000000	0.000000	0.000000	0.000000	0.0
50%	47.000000	19.000000	0.000000	2.000000	0.000000	0.000000	0.000000	0.000000	0.0
75%	71.000000	24.000000	0.000000	3.000000	1.000000	0.000000	0.000000	0.000000	0.0
max	1386.000000	214.000000	1.000000	24.000000	43.000000	4.000000	3.000000	19.000000	0.0

Density plot for status

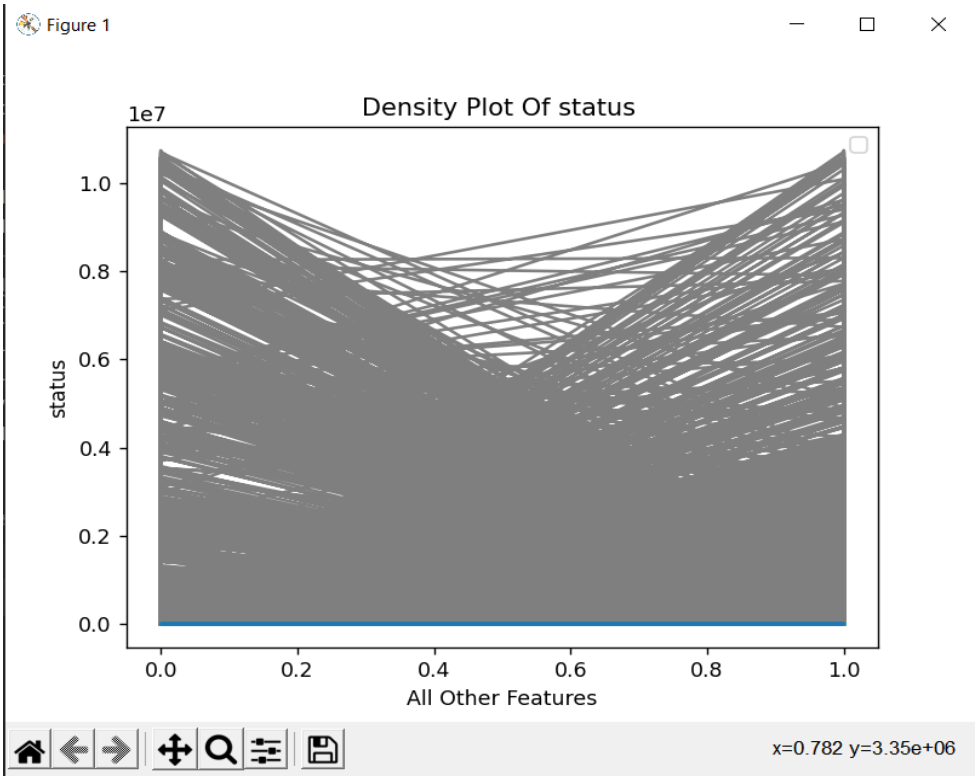
Status VS google Index



Status VS page_rank

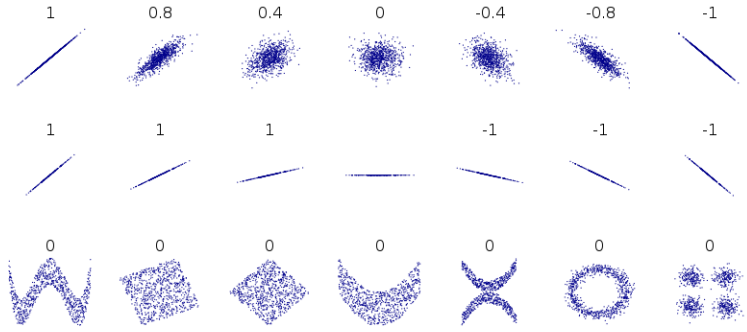


Status VS All Other Features



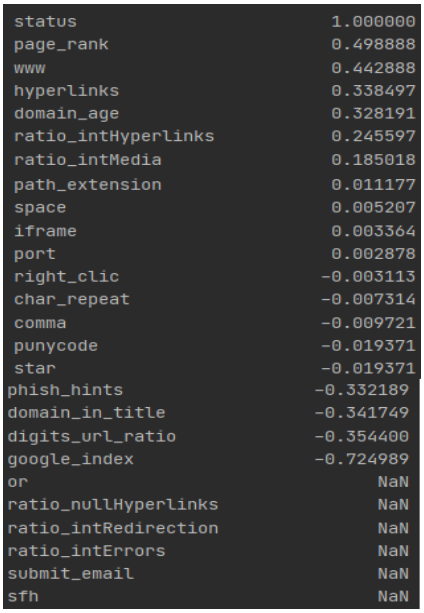
Correlation Coefficient

The correlation coefficient is a statistical measure of the strength of the relationship between the relative movements of two variables,and the values range between -1 and 1,And we can see the impact of the features on the target class that we want to expect,the figure below show the data shape and the value of correlation coefficient.



After we apply the correlation coefficient and sort So we use the features that have a greater correlation with the crown class, and delete the features that have no correlation with the target class.

As shown in the Picture we get 5 features that have no correlations with the target class , so we drop this feature from the data set .



status	1.000000
page_rank	0.498888
www	0.442888
hyperlinks	0.338497
domain_age	0.328191
ratio_intHyperlinks	0.245597
ratio_intMedia	0.185018
path_extension	0.011177
space	0.005207
iframe	0.003364
port	0.002878
right_clic	-0.003113
char_repeat	-0.007314
comma	-0.009721
punycode	-0.019371
star	-0.019371
phish_hints	-0.332189
domain_in_title	-0.341749
digits_url_ratio	-0.354400
google_index	-0.724989
or	NaN
ratio_nullHyperlinks	NaN
ratio_intRedirection	NaN
ratio_intErrors	NaN
submit_email	NaN
sfh	NaN

Linear Regression

Linear regression attempts to model the relationship between two variables (or more) by fitting a linear equation to observed data. The equation is in the form $\boldsymbol{Y = a + bX}$, where $\boldsymbol{X}$ is the explanatory variable and $\boldsymbol{Y}$ is the dependent variable.

After a regression line has been computed for a group of data, a point which lies far from the line (and thus has a large residual value) is known as an ***outlier***. Such points may represent erroneous data, or may indicate a poorly fitting regression line. If a point lies far from the other data in the horizontal direction, it is known as an ***influential observation***. The reason for this distinction is that these points may have a significant impact on the slope of the regression line.

Results

```
----------------------------------------Linear Regression----------------------------------------
-Performance measure (Accuracy)(R^2) for predictions using linearRegression: 0.738821202369177
```

Logistic Regression

Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).

When selecting the model for the logistic regression analysis, another important consideration is the model fit. Adding independent variables to a logistic regression model will always increase the amount of variance explained in the log odds (typically expressed as R^2). However, adding more and more variables to the model can result in overfitting, which reduces the generalizability of the model beyond the data on which the model is fit.

Results

```
----------------------------------------Logistic Regression----
Accuracy:  0.7847141190198367
Precision:  0.7963525835866262
Recall:  0.7647402218330415
Error Rate= 0.21528588098016332
*LogisticReg_classification_report:
              precision    recall  f1-score   support

           0       0.77      0.80      0.79      1715
           1       0.80      0.76      0.78      1713

    accuracy                           0.78      3428
   macro avg       0.79      0.78      0.78      3428
weighted avg       0.79      0.78      0.78      3428

*LogisticReg_confusion_matrix:
[[1380  335]
 [ 403 1310]]
```

KNN

- Simply kNN is an algorithm learning algorithm that calculates the k nearest neighbors of learning data for the data we want to expect.
- Choosing a value for k depends on the number of records in the training dataset. But probably we set the value of k as the square root of n, and k somewhere will be between 3 and 10 .
- Distance is computed by similarity measures such as Chi2, Minkowski derivatives (e.g., Euclidean distance, Manhattan distance, …), cosine distance, and other similarity measures.
- We can use the KNN algorithm for applications that require high accuracy but that do not require a human-readable model. The quality of the predictions depends on the distance measure. Therefore, the KNN algorithm is suitable for applications for which sufficient domain knowledge is available.

Results

```
-----------------------------------KNN-----------------------------------------
Accuracy:  0.925904317386231
Precision:  0.917095483133219
Recall:  0.9363689433741973
Error Rate= 0.07409568261376898
*KNN_classification_report:
              precision    recall  f1-score   support

           0       0.94      0.92      0.93      1715
           1       0.92      0.94      0.93      1713

    accuracy                           0.93      3428
   macro avg       0.93      0.93      0.93      3428
weighted avg       0.93      0.93      0.93      3428

*KNN_confusion_matrix:
[[1570  145]
 [ 109 1604]]
```

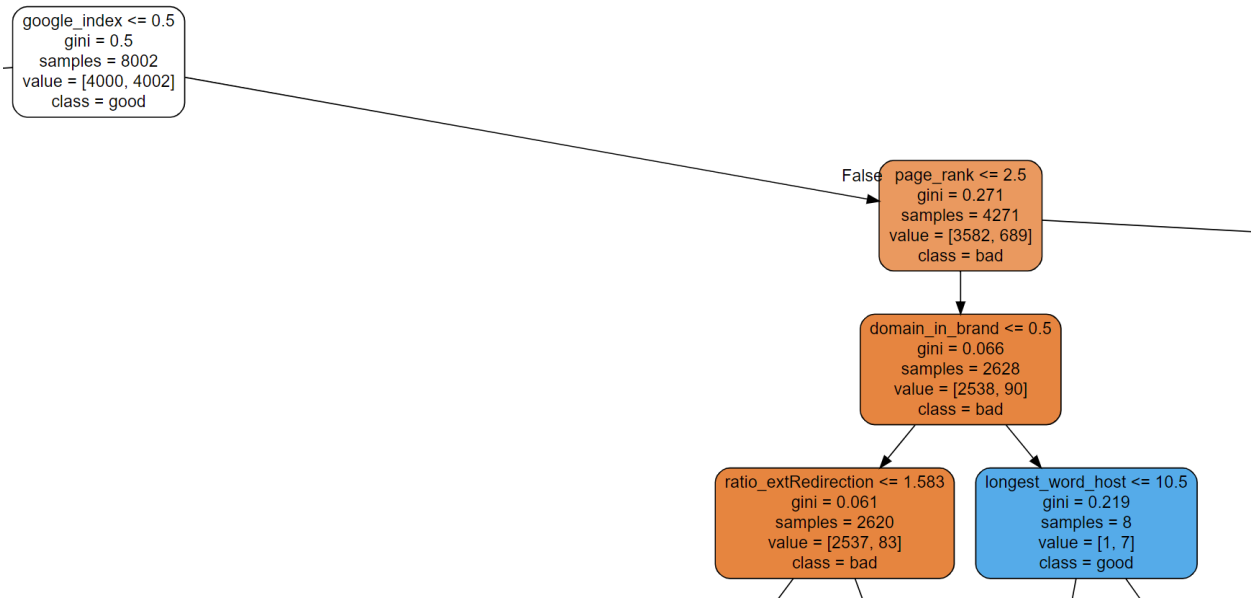
Decision Tree

Decision Tree is a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions.Decision trees can perform both classification and regression tasks, so you'll see authors refer to them as CART algorithm: Classification and Regression Tree.The intuition behind Decision Trees is that you use the dataset features to create yes/no questions and continually split the dataset until you isolate all data points belonging to each class.With this process you're organizing the data in a tree structure.Every time you ask a question you're adding a node to the tree. And the first node is called the root node.The result of asking a question splits the dataset based on the value of a feature, and creates new nodes.If you decide to stop the process after a split, the last nodes created are called leaf nodes.

Results

```
------------------------Decision tree--------------------------
Accuracy:  0.9366977829638273
Precision:  0.9328703703703703
Recall:  0.9410391126678342
Error Rate= 0.06330221703617267
*DecisionTree_classification_report:
              precision    recall  f1-score   support

           0       0.94      0.93      0.94      1715
           1       0.93      0.94      0.94      1713

    accuracy                           0.94      3428
   macro avg       0.94      0.94      0.94      3428
weighted avg       0.94      0.94      0.94      3428

*DecisionTree_confusion_matrix:
[[1599  116]
 [ 101 1612]]
```

Also the tree that has been built is attached, as .dot.html file as it is so large to fit here
But here the root node with some childs:



Naive bayes

- simple technique for constructing classifiers, models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set.
- There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable.
- For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.
- Naive Bayes is suitable for solving multi-class prediction problems.

Results

```
----------------------------------------Naive bayes----------------
Accuracy:  0.9049008168028004
Precision:  0.9039021549213745
Recall:  0.9060128429655575
Error Rate= 0.09509918319719957
*Naive bayes_classification_report:
              precision    recall  f1-score   support

           0       0.91      0.90      0.90      1715
           1       0.90      0.91      0.90      1713

    accuracy                           0.90      3428
   macro avg       0.90      0.90      0.90      3428
weighted avg       0.90      0.90      0.90      3428

*Naive bayes_confusion_matrix:
[[1550  165]
 [ 161 1552]]
```

K-means

- K-means is Unsupervised learning, on the other hand, is the task of inferring/describing hidden structures or patterns from unlabeled data.

- We used K-means to find groups which have not been explicitly labeled in the data.

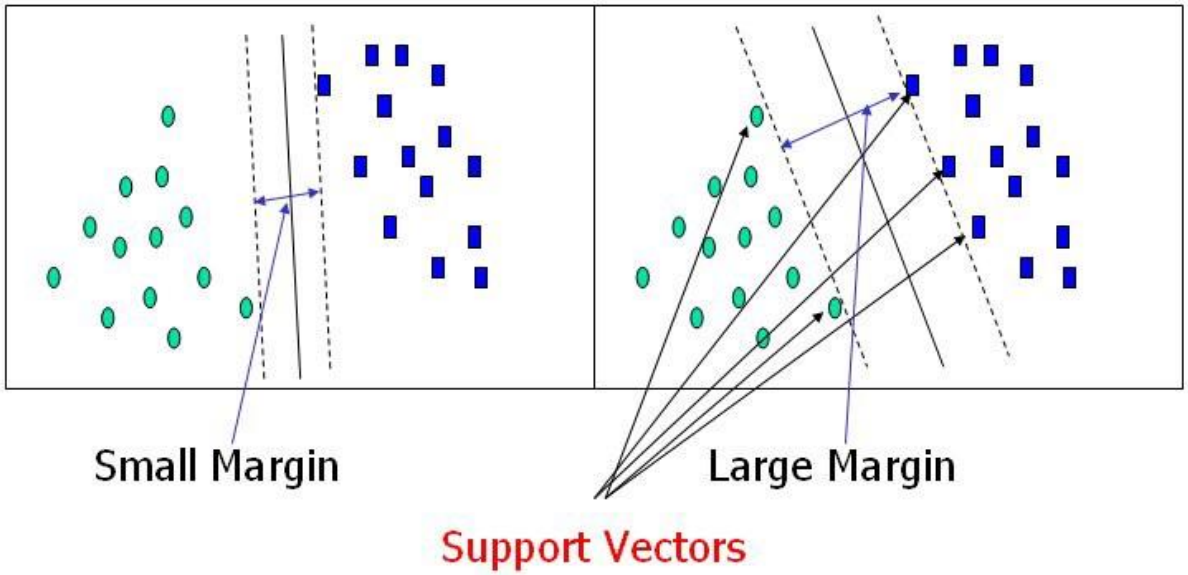
- An example of K-means is the appearance of suggested friends on social media based on several factors, and the classification of videos or movies into labels (educational, action, fantasy, …) on sites that show videos or movies like YouTube and Netflix.

Results

```
---------------------------------------- K-means ------------
Accuracy:  0.8719369894982497
Precision:  0.8719369894982497
Recall:  0.8719369894982497
Error Rate= 0.12806301050175029
*K-means_classification_report:
              precision    recall  f1-score   support

           0       0.85      0.91      0.88      1715
           1       0.90      0.84      0.87      1713

    accuracy                           0.87      3428
   macro avg       0.87      0.87      0.87      3428
weighted avg       0.87      0.87      0.87      3428

*K-means_confusion_matrix:
[[1554  161]
 [ 278 1435]]
```

Support Vector Machine(SVM)

The objective of the support vector machine algorithm is to find a hyperplane that has the maximum margin in an N-dimensional space (N:- the number of features) that distinctly classifies the data points. Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane.



Results

```
-----------------------------------------SVM----------------------
Accuracy:  0.9101516919486581
Precision:  0.9159265837773831
Recall:  0.9030939871570345
Error Rate= 0.08984830805134192
*SVM_classification_report:
              precision    recall  f1-score   support

           0       0.90      0.92      0.91      1715
           1       0.92      0.90      0.91      1713

    accuracy                           0.91      3428
   macro avg       0.91      0.91      0.91      3428
weighted avg       0.91      0.91      0.91      3428

*SVM_confusion_matrix:
[[1573  142]
 [ 166 1547]]
```

Conclusion

To sum thing up, almost all of the tried algorithms are relatively good, but we consider the one to be the Decision Tree classifier (CART algorithm), as maybe there are some redundancies in the features of the dataset or some patterns we couldn't figure out as the datasets are huge and an overfitting or underfitting has happened to some algorithms.