Phishing Websites classification

Abstract

In today's digital age, and with the digital transformation that we are witnessing nowadays, not locally in our country but in the whole world, cybercrime affects all of us directly or indirectly because as individuals and companies, we all have information that is worth something to cybercriminals. The most common way in cybercrime is to attack an induvial by phishing websites, especially lately due to COVID-19, the use of the internet and E-commerce, E-governance sites has increased. Therefore, in this project we will determine whether this link is phishing or not, by applying multi classifier methods in three approaches, the first approach is by applying classifiers without scaling and modifying data, second approach is applying smote to balnce the data, third approach is by applying smote to balnce the data and scale.

Data

- Dataset source: https://www.sciencedirect.com/science/article/pii/S2352340920313202
- Data Size: It has 112 columns (Features) and 88,648 rows (Records), published date: 2019.
- Data type: (Integer, Float)

Data cleaning

- Replace all negative values with zeros.
- Divide data frame into features and target.
- Check if the data contains missing value.
- Drop duplicated rows or records.

Design

In this project, we used multi classifiers models such as:

1. Decision tree.
2. Logistic regression.
3. Support vector machine.
4. Naïve bayes.
5. Random forest.
6. K-Nearest Neighbor.

Applying the above models to classify data, according to the following three approaches:

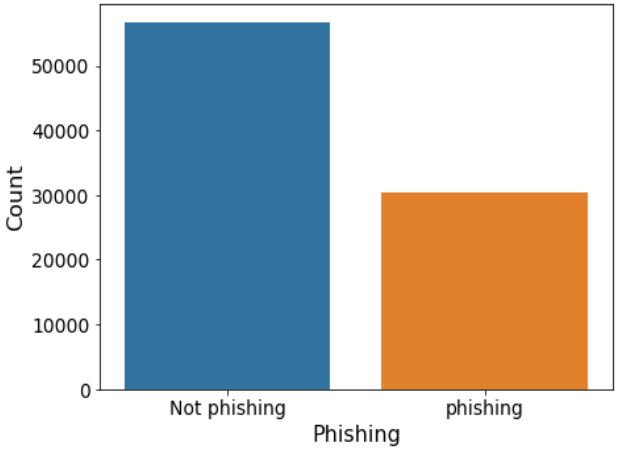
1. Applying classifiers without scaling and modifying the data.
2. Applying smote to balnce the data.
3. Balance the data and then scale.

Finally, we evaluate these models with:

1. Accuracy.
2. Precision.
3. Recall.
4. Roc_auc.
5. F1 scores.

Model

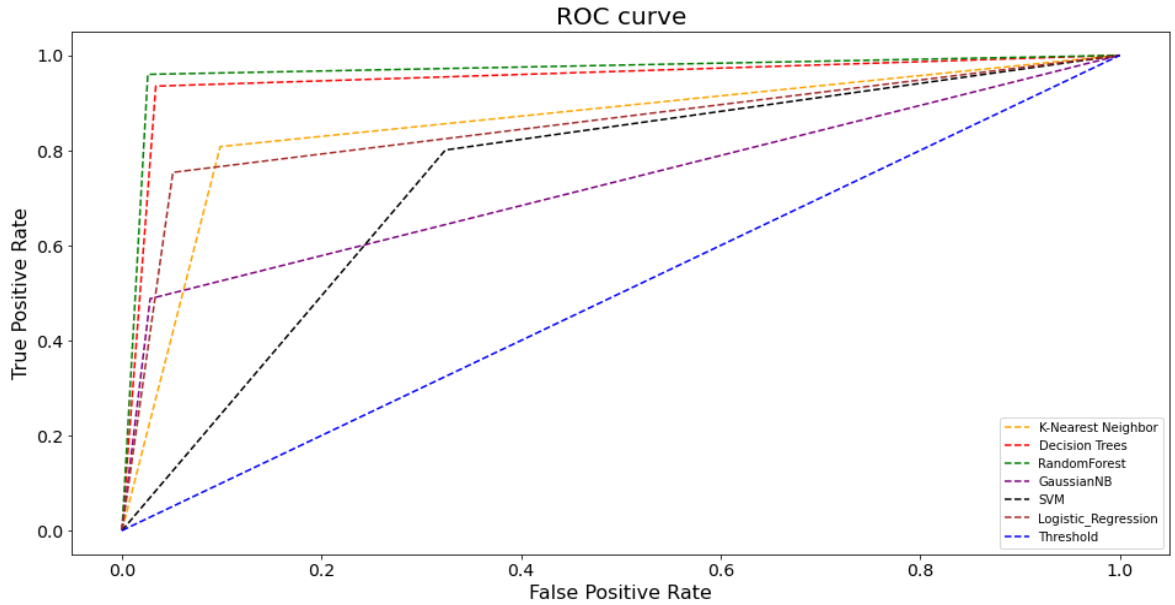
1. Applying classifiers without scaling and modifying the data



The figure above demonstrates the unbalancing in the data between phishing and not phishing classes.

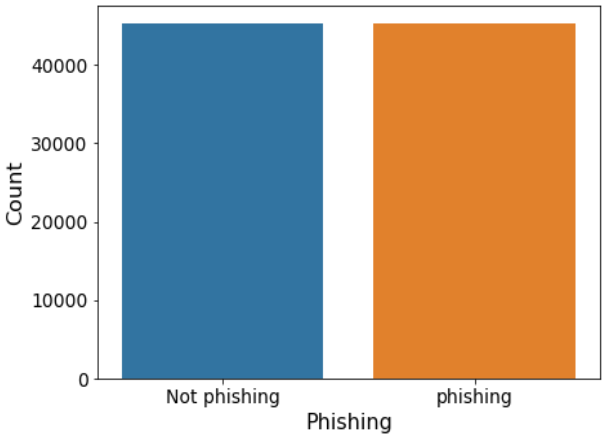
	Accuracy	Precision	Recall	ROC_AUC	F1
Logistic Regression	0.881193	0.753673	0.887269	0.882960	0.815033
Support Vector Machines	0.765596	0.949645	0.603251	0.782334	0.737814
Decision Trees	0.954014	0.934126	0.933509	0.949222	0.933817
Random Forest	0.969209	0.961532	0.950392	0.964897	0.955929
Naive Bayes	0.803727	0.487865	0.902015	0.841504	0.633237
K-Nearest Neighbor	0.869037	0.807826	0.813737	0.855936	0.810771

The table above demonstrate scores of the most important metrics in classification, and Random Forest is the highest score.



ROC curve is a common method for evaluating the equality of a binary classifier, it compares the presence of true positives and false positive at every probability threshold. According to above figure, Random Forest and decision tree predicted the most observations correctly compared with the other classifiers.

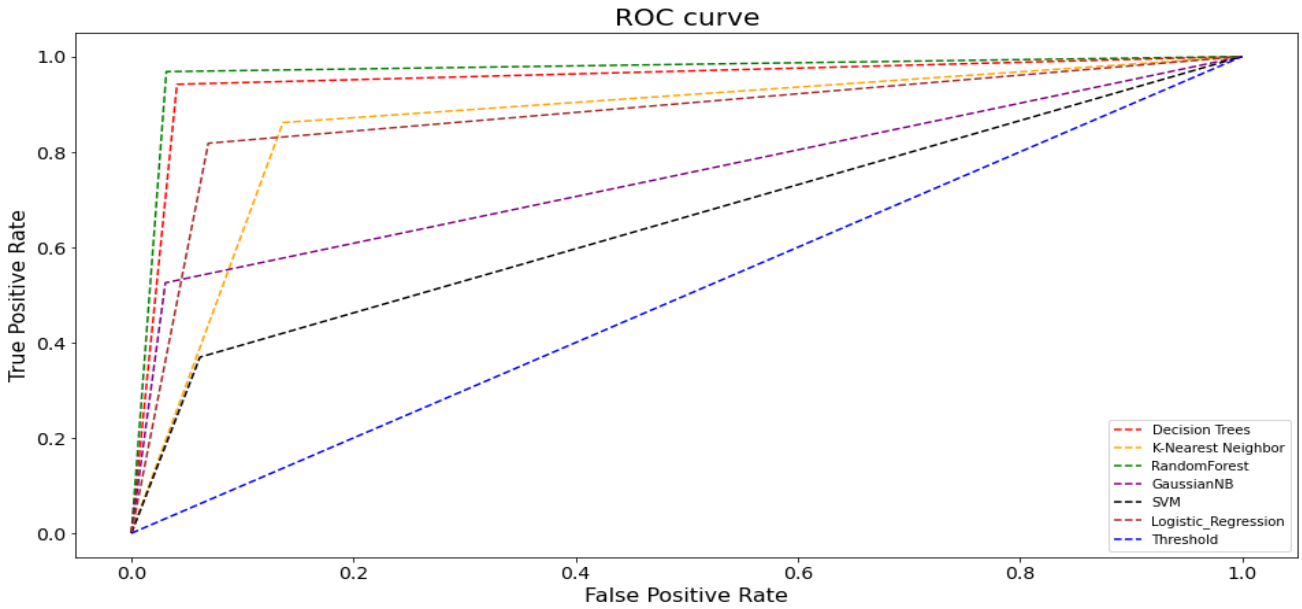
2. Applying smote to balnce the data



The figure above demonstrates the data after balancing it.

	Accuracy	Precision	Recall	ROC_AUC	F1
Logistic Regression	0.891628	0.818062	0.862789	0.884289	0.839831
Support Vector Machines	0.759690	0.703318	0.640216	0.736806	0.670286
Decision Trees	0.953727	0.945022	0.923524	0.946951	0.934149
Random Forest	0.968291	0.969292	0.941167	0.962281	0.955022
Naive Bayes	0.815195	0.525343	0.901416	0.847365	0.663816
K-Nearest Neighbor	0.862844	0.861483	0.770639	0.846000	0.813533

As the first approach, Random Forest was the highest score.



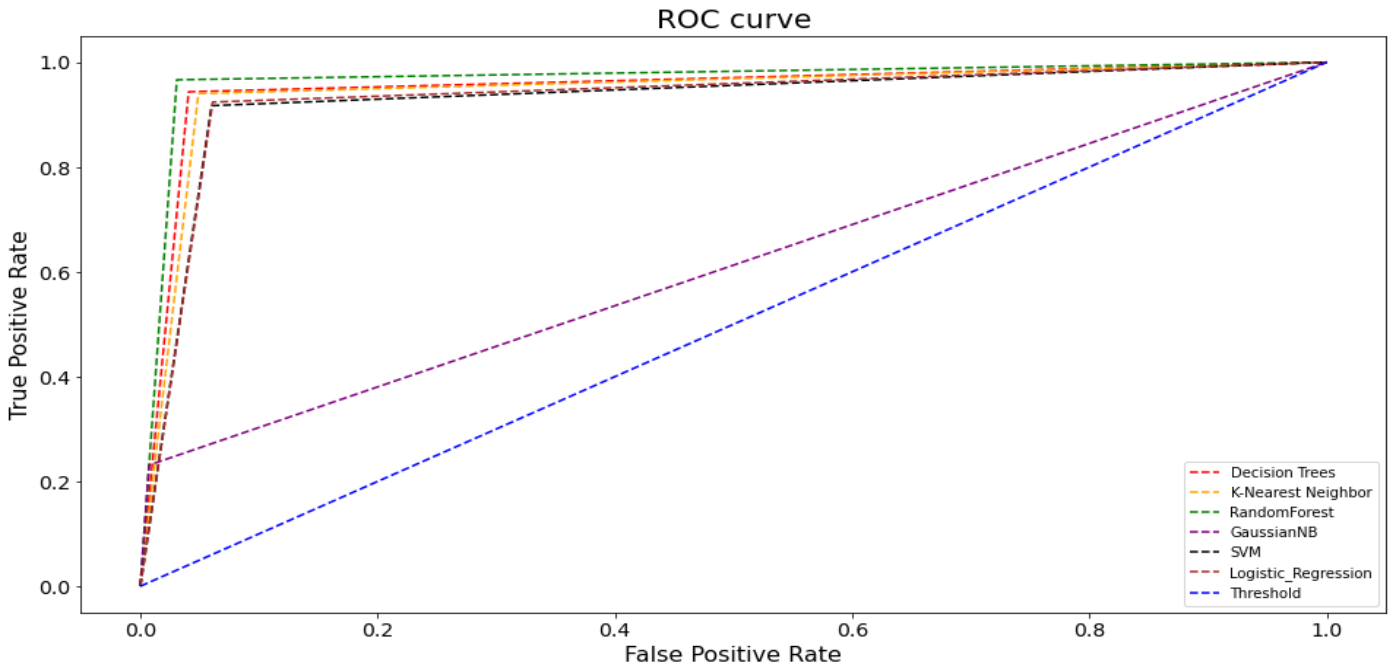
In this figure, Random Forest predicted the most observations correctly compared with other classifiers from the curves as the first approach, decision tree was very close of Random Forest.

3. Balance the data and then scale.

In this approach, the data was first balanced, and then scaled.

	Accuracy	Precision	Recall	ROC_AUC	F1
Logistic Regression	0.933773	0.923890	0.889666	0.924160	0.906455
Support Vector Machines	0.932569	0.916460	0.892300	0.923599	0.904219
Decision Trees	0.952867	0.939739	0.925679	0.946676	0.932656
Random Forest	0.968922	0.966815	0.944973	0.963548	0.955770
Naive Bayes	0.728211	0.231138	0.944032	0.826093	0.371353
K-Nearest Neighbor	0.947248	0.940069	0.910894	0.939226	0.925252

The logistic regression, SVM and KNN got better scores than the second approach. However, the Random Forest was the highest score.



We applied the scaling on the data, due to the fact that some models get affected by the scaling process, such as: KNN, SVM and the logistic regression. Almost all classifiers have a good curve except naïve bayes.

Tools

- Basic tools: anaconda, python and jyputer notebook.
- Libraries: Seaborn, numpy, pandas, matplotlib and sklearn.
- Models function: DecisionTreeClassifier, KNeighborsClassifier, RandomForestClassifier, GaussianNB, LinearSVC and LogisticRegression.
- Metrics function: Accuracy, precision, recall roc_auc and f1.

Conclusion

After applying the three approaches as mentioned in design section, the best classifier was Random Forest based on recall score, that represents a measure of phishing that did occurred, and the predicted value is there is no phishing. There is no need to scale to get good scores when using Random Forest and decision tree. However, balancing the data is important, so that the results are accurate.

The results shows that there is no big change in scores of Random Forest and decision tree, they both were the highest scores. However, there was a slight enhancement in the ROC curves after applying different approaches.