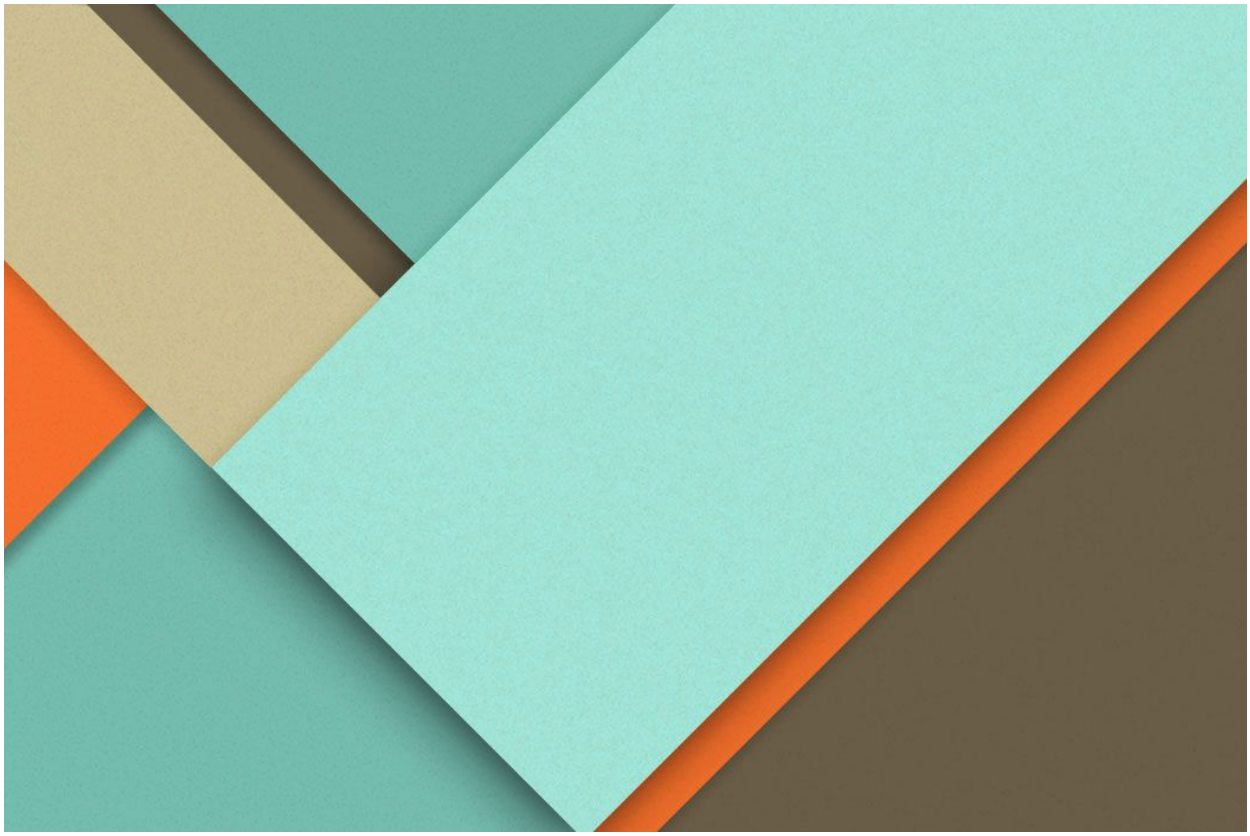


Social Network Ads

Parth Sachdev
1PE16CS108
github.com/parthsachdev
parthsachdev05@gmail.com

Ninad Dighe
1PE16CS100
github.com/ninad7007
dighe.ninad7@gmail.com

Nikhil
1PE16CS094
github.com/nikhilsawlani
nikhisawlani123@gmail.com

Problem Statement

The problem statement of our dataset is to predict whether the users of a social networking site will buy a particular product. We'll determine this by using user's gender, estimated salary and their age. It is a fictional dataset which is meant only for the purpose of practice and understanding.

Stocktaking of the data

1. **Data Source:** Kaggle.
2. **No. of records:** 400. These consist of 75% training data and 25% test data.
3. **No. of attributes:** 5, which are User ID, Gender, Age, Estimated Salary and Purchased
4. **Kind of Attributes:** User Id, discrete attribute, it is a unique identifier for each record. Gender, Binary attribute. Age, numeric attribute. Estimated Salary, Continuous and Purchased, binary telling whether the user purchased it or not.
5. Since it is not a real dataset there aren't any missing values in it.

Data Preprocessing

The gender attribute which was given as male or female in the dataset was encoded into one hot encoding. The whole dataset was then normalized to convert it into a standard range for convenient calculations.

Each of the records of the dataset belong to two classes of customers, those who purchased and those who didn't. Out of 400 records 257 belong to class 0 (who didn't purchase) and 143 belong to class 1. Therefore, the data is partitioned as **64.25% of class 0 and 35.75% of class 1**.

Classification

The task was to classify the users into two groups, one who buys that product and other who doesn't.

We used **K-Nearest Neighbors** algorithm for classification of the data. We tuned the value of k in our algorithm. The value of 5 was the optimal as it gave us maximum accuracy. The distance metric chosen was euclidean distance.

Experimental Results

The results of our experiment are as follows:

The dataset was partitioned as 75% used for training the model and the rest for validation.

The results based on some of the metrics are:

1. **Confusion Matrix:** The confusion matrix of our results is as follows:

CM	No	Yes
No	64	4
Yes	3	29

2. **Classification Accuracy:** It shows that the accuracy of the model is 93% since 93 out of 100 predicted results were correct.
3. **Logarithmic Loss:** Logarithmic loss measures the performance of a classification model where the prediction input is a probability value between 0 and 1.

$$LogarithmicLoss = \frac{-1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} * \log(p_{ij})$$

The logarithmic loss turned out to be **2.418.**

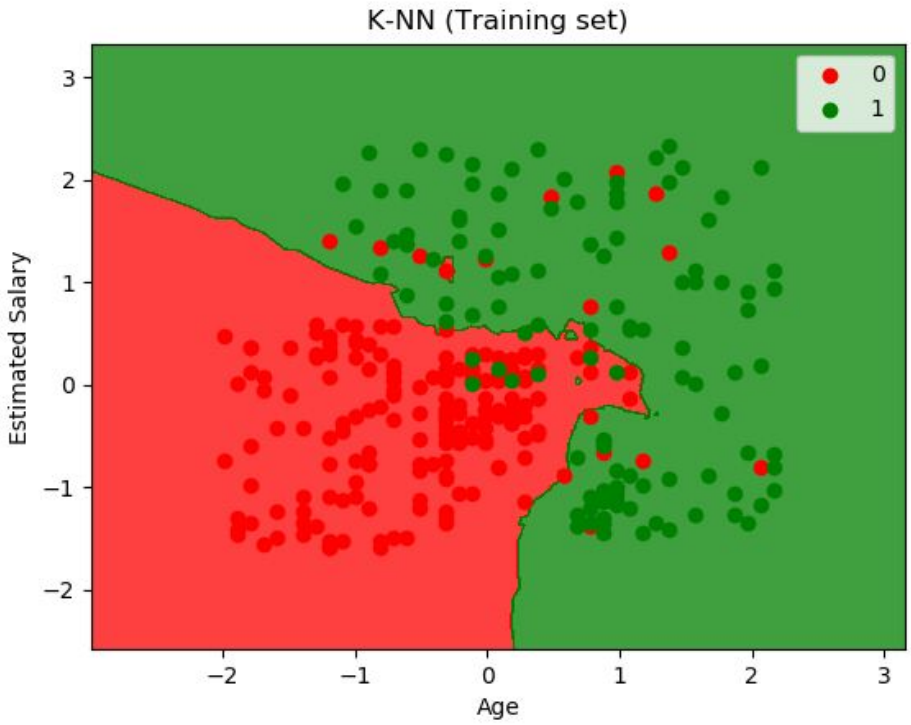
4. **F1 Score**: The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0.

$$F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}}$$

The F1 score of our result is **0.892.**

Discussion

The scatter plot obtained from the results is:

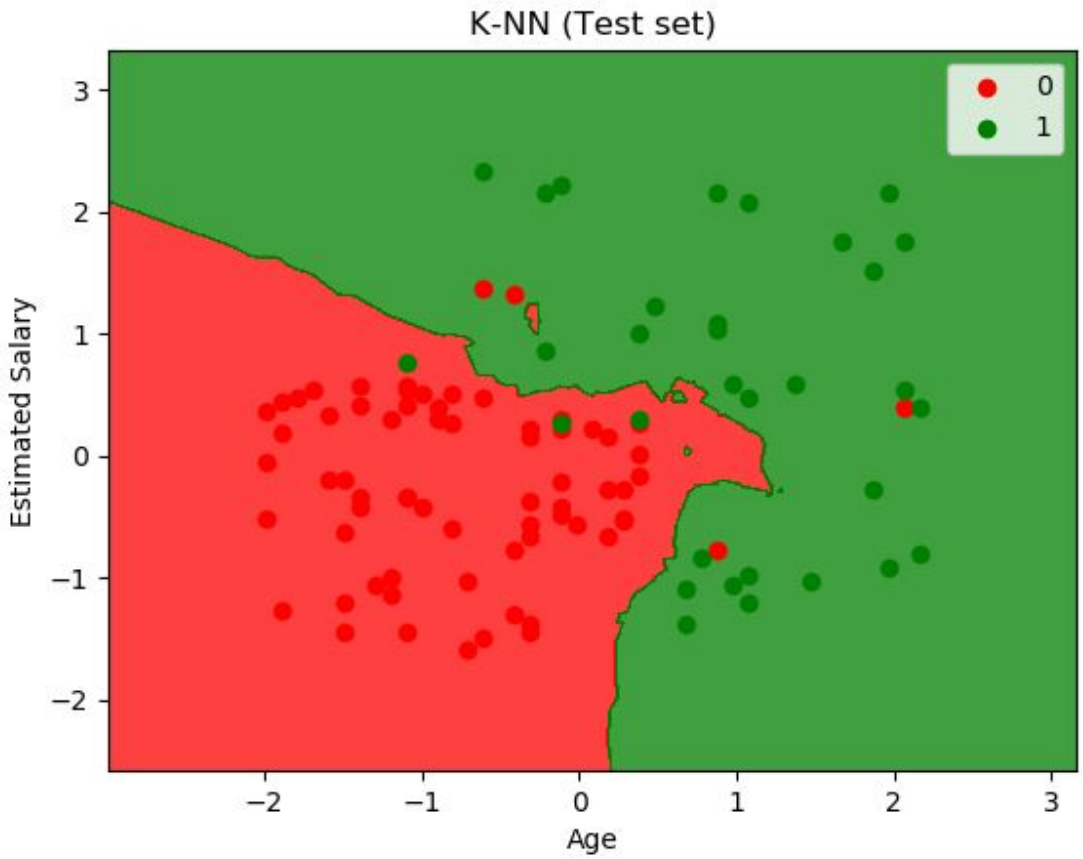


This is a scatter plot for the 300 training records after the KNN classification is done on the data.

The green region above depicts the region where users will buy the product and similarly the red region for the users who don't buy the product.

As we can see there are quite a few outliers indicated by red dots in the green region that are quite far from the red region. There are also a few green dots in the red region as well.

Similarly the scatter plot for the test data is:



Conclusion

We can conclude that the K-Nearest Neighbour Algorithm works good for small datasets (400 records in our dataset) and gives high accuracy.

The results of the model could be improved by using Neural Networks which are proven to give better and more diverse models for all types of datasets.

References

1. Machine Learning Course on Udemy: https://www.udemy.com/machinelearning/
2. Metrics used to evaluate results: https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234
3. Sklearn documentation: https://scikit-learn.org