Predicting sales through ads delivered on social networking sites

Prepared by,
Nipen Khirsaria

August 17, 2021
Indian Institute of Technology, Bombay
Department of Aerospace Engineering

Contents

1. About Data Set

- Dataset for social networking ads is stored in csv file with 5 columns namely, User ID, Gender, Age, EstimatedSalary and Purchased. It has total 400 datapoints. First 10 of them are as shown below:

Table 1: First 10 entries of data set

User ID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0
15728773	Male	27	58000	0
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	0
15727311	Female	35	65000	0

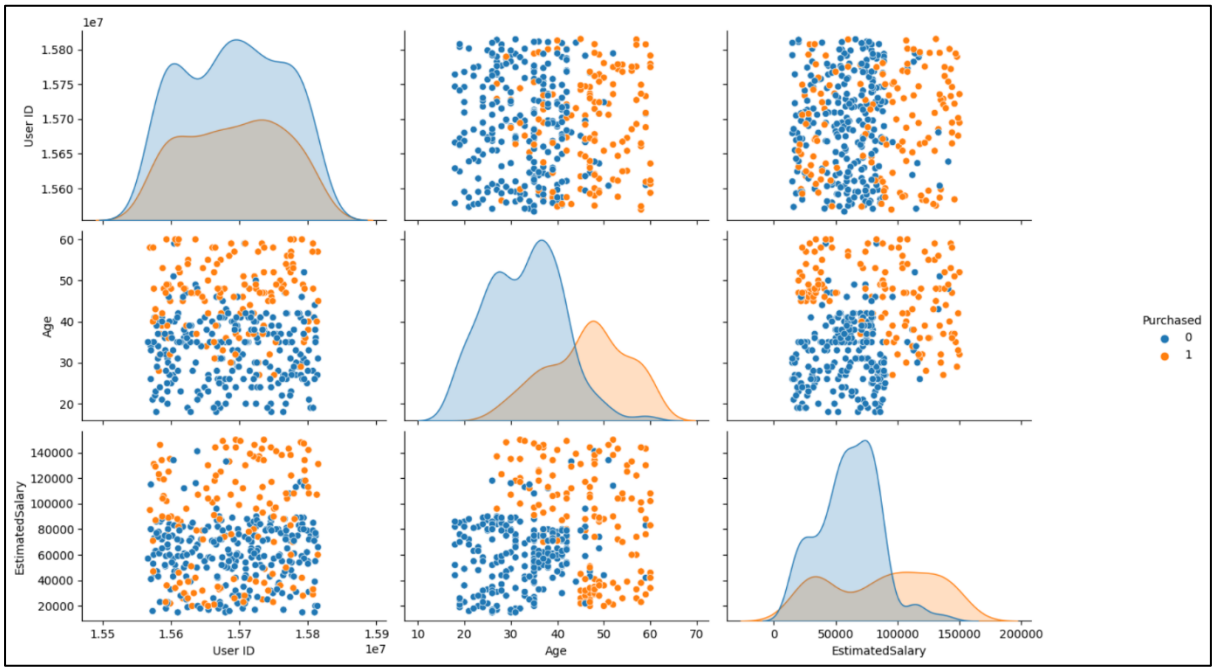


Figure 1: Data Visualization

- From Figure 1 it is clear that given data is binary type classification (0 and 1) with Age and EstimatedSalary as variables.
- Here under Purchased column 0 means a person does not buy and 1 means a person buys product.
- Third and fourth columns is taken as input vector X whose dimension will be (400,2).

- Fifth column (Purchased) will be output vector Y whose dimension will be (400,1).
- 400 data points is divided into training (300) and test sets (100). Hence dimensions of training and test sets will be as follows:

$$X_{train}: (300,2) \quad Y_{train}: (300,1)$$

$$X_{test}: (100,2) \quad Y_{test}: (100,1)$$

- Note: Dividing 400 data points into training and test sets is random.
- From Table 1 it is clear that values in two columns of input vector X (i.e Age & EstimatedSalary) are not within same range. Hence feature normalization is done as shown below:

$$X_{new}(i,j) = \frac{X(i,j) - \mu(j)}{\sigma(j)} \qquad i = 1,2,\dots,400 \; and \; j = 1,2$$

Where $\mu(j)$ is the mean of values in j^{th} column of X

$\sigma(j)$ is the standard deviation of values in j^{th} column of X

2. Training using Logistic Regression

2.1 Forward Propagation

Figure 2 shows forward propagation in logistic regression where input features x_1, x_2 and parameters w_1, w_2, b are used to formulate z and then sigmoid function is used as activation which then finally is used to evaluate cost function J.

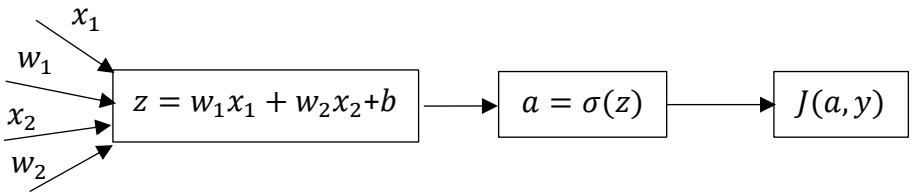


Figure 2: Logistic Regression

x_1, x_2 : Features

w_1, w_2, b : Parameters

$a = \frac{1}{1 + e^{-z}}$ is sigmoid function

$$J(a, y) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)} \log(a^{(i)}) + (1 - y^{(i)}) \log(1 - a^{(i)})]$$

is cost function to be minimized

2.2 Backward Propagation and Parameter update

Now in order to optimize parameters w_1, w_2, b, derivatives of J is taken with respect to parameters w_1, w_2, b and these parameters are updated with learning rate α as follows:

$$w_1 : w_1 - \alpha \frac{\partial J}{\partial w_1}$$

$$w_2 : w_2 - \alpha \frac{\partial J}{\partial w_2}$$

$$b : b - \alpha \frac{\partial J}{\partial b}$$

3. Training using KNN (K Nearest Neighbour)

3.1 Algorithm

- K Nearest Neighbour is an algorithm that stores all the available cases and classifies the new data or case based on a similarity measure. Here K is no. of nearest neighbour.
- Let us take few of the training datapoints which is plotted as shown in Figure 3, where blue colour is for positive (1) and red for negative (0).
- Now the new data point shown by star is to be categorised into either 0 or 1. For this Euclidean distance is evaluated between new data point (star) and training data points. Euclidean distance is expressed as shown below:

$$r = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

- Now value of K is selected. For simplicity let value of K = 3. So, among all the Euclidean distances evaluated 3 least distances are chosen. If there are 2 blues (1) and 1 red (0) then new data point is allotted as blue (1) since it has majority.

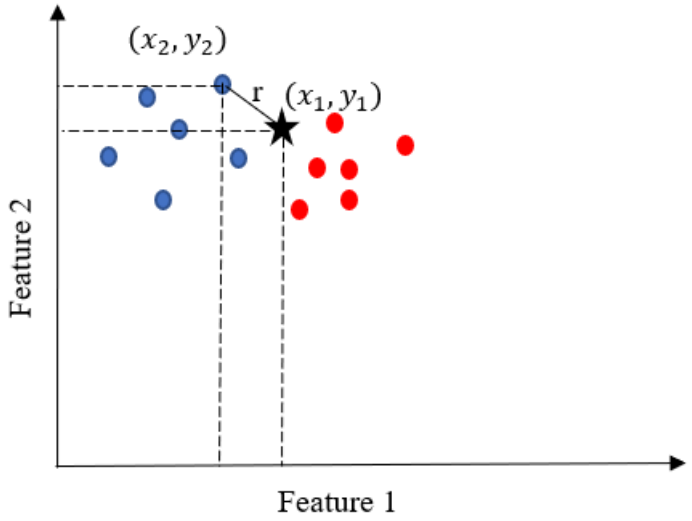


Figure 3: KNN

3.2 Selection of value of K

- Having studied about KNN as algorithm, it is important to find the optimized value of K.
- For different values of K sum of squared error between actual and prediction is calculated as shown below:

$$error = \sum_{i=1}^{i=m} \left(y_{pred}(i) - y_{actual}(i)\right)^2$$

- From Figure 4 it is clear that K=5 is the optimum value for which error is minimum and F1 score is maximum. K=3 also gives same values of error and F1score. The reason for

choosing K=5 over K=3 is that, the boundary curve separating two classes is smoother for K=5 then K=3 which is clear from Figure 5.

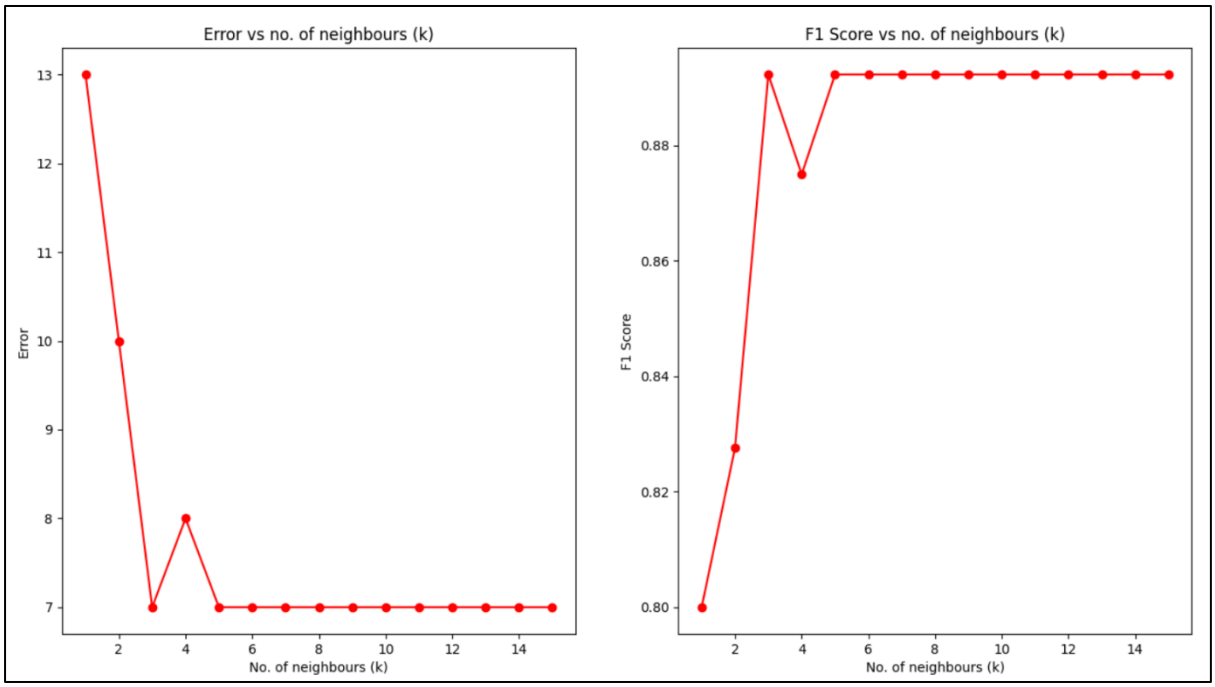
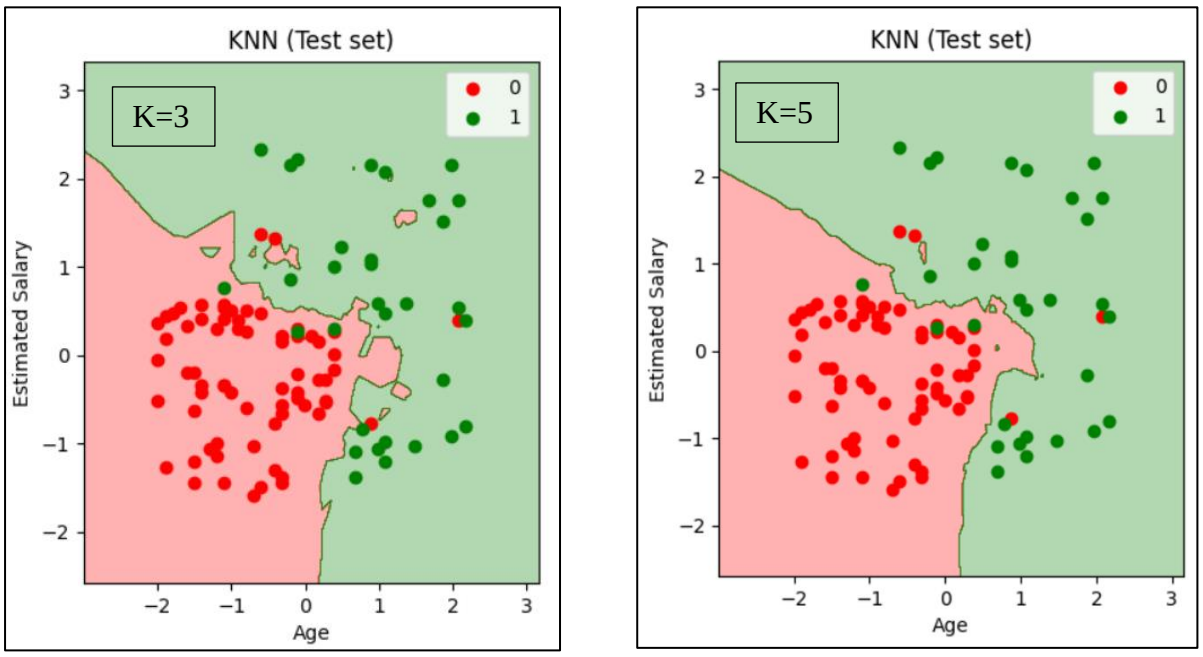


Figure 4: Optimization of K



(a) (b)

Figure 5: KNN Classification

4. Performance Evaluation

Table 2: Confusion Matrix

	Predicted 0	Predicted 1
Actual 0	True (-ve)	False (+ve)
Actual 1	False (-ve)	True (+ve)

Table 3: Confusion Matrix for Logistic Regression (LR)

	Predicted 0	Predicted 1
Actual 0	65	3
Actual 1	8	24

Table 4: Confusion Matrix for KNN

	Predicted 0	Predicted 1
Actual 0	64	4
Actual 1	3	29

Precision: Precision is defined as of all the person who made purchase from prediction what fraction of them did really made actual purchase. It is expressed as follows with help of Confusion Matrix (Table 2):

$$Precision = P = \frac{True(+ve)}{True(+ve) + False(+ve)}$$

$$P_{LR} = \frac{24}{24 + 3} = 0.8889$$
$$P_{KNN} = \frac{29}{29 + 4} = 0.8788$$

Recall: Recall is defined as of all the persons that actually made purchase what fraction of them is correctly predicted as purchased. It is expressed as follows with help of Confusion Matrix (Table 2):

$$Recall = R = \frac{True(+ve)}{True(+ve) + False(-ve)}$$

$$R_{LR} = \frac{24}{24 + 8} = 0.75$$
$$R_{KNN} = \frac{29}{29 + 3} = 0.90625$$

F1 Score: It is often difficult to measure performance based on Precision and Recall, so single performance matric F1 Score is formulated as below using Precision and Recall:

$$F1\ Score = \frac{2PR}{P+R}$$

$$F1_{LR} = 0.8136$$
$$F1_{KNN} = 0.8923$$

Hence it is clear that performance of KNN model is better than Logistic Regression.

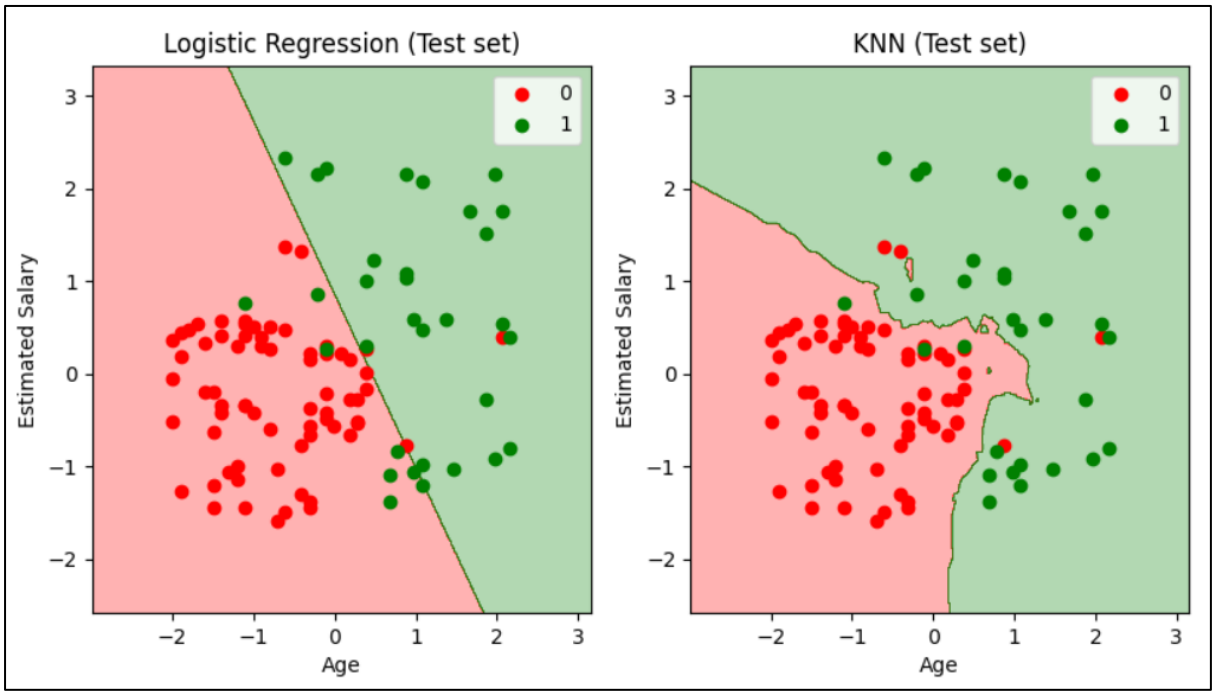


Figure 6: Performance Comparison