**Project Overview:**

Extraction of interesting (non-trivial, implicit, previously unknown, and potentially helpful) patterns or information from huge amounts of data is what data mining (knowledge discovery from data) is all about. Data mining is used in many different industries, including business and research. Data mining is a comprehensive field in computer science and statistics that aims to extract information from data sets and structure it for subsequent use. KNN, Naive Bayes, and Decision Tree are examples of classification algorithms used in data mining.

Bank Marketing Dataset is the dataset we've picked for our project's classification application. This dataset was obtained from the [www.kaggle.com](http://www.kaggle.com) website. The data is related to direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact with the same client was required, in order to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed.

**Dataset Overview:**

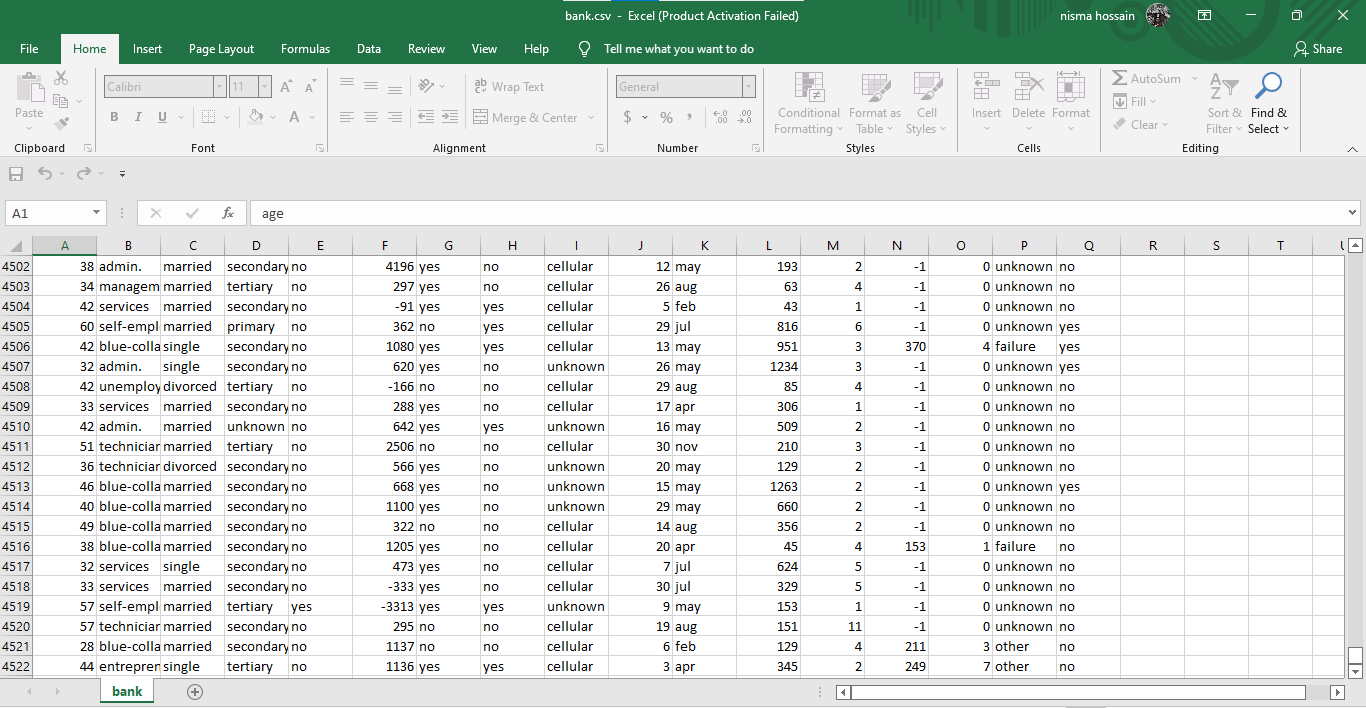


Figure 1: Bank Marketing Dataset

The dataset has a total of 17 attributes and 4522 instances. The sample of the figure has shown above and for the whole dataset please go to <https://www.kaggle.com/datasets/hariharanpavan/bank-marketing-dataset-analysis-classification>

**Attributes:**

**Age** -Age of customer

**Job**- Job of customer

**Martial**- Martial status of customer

**Education**- Customer education level

**Default**- Has credit in default?

**Housing**- If the customer has a housing loan

**Loan** -Has Personal Loan

**Balance**- Customer's individual balance

**Contact**- Communication type

**Month**- Last contact month of the year

**Day**- Last contact day of the week

**Duration**- Last contact duration, in seconds

**Campaign**- Number of contacts performed during this campaign and for this client

**P days** - Number of days that passed by after the client was last contacted from a previous campaign

**Previous**- Number of contacts performed before this campaign and for this client

**Poutcome**- the outcome of the previous marketing campaign

**Y** -has the client subscribed to a term deposit (yes/no)

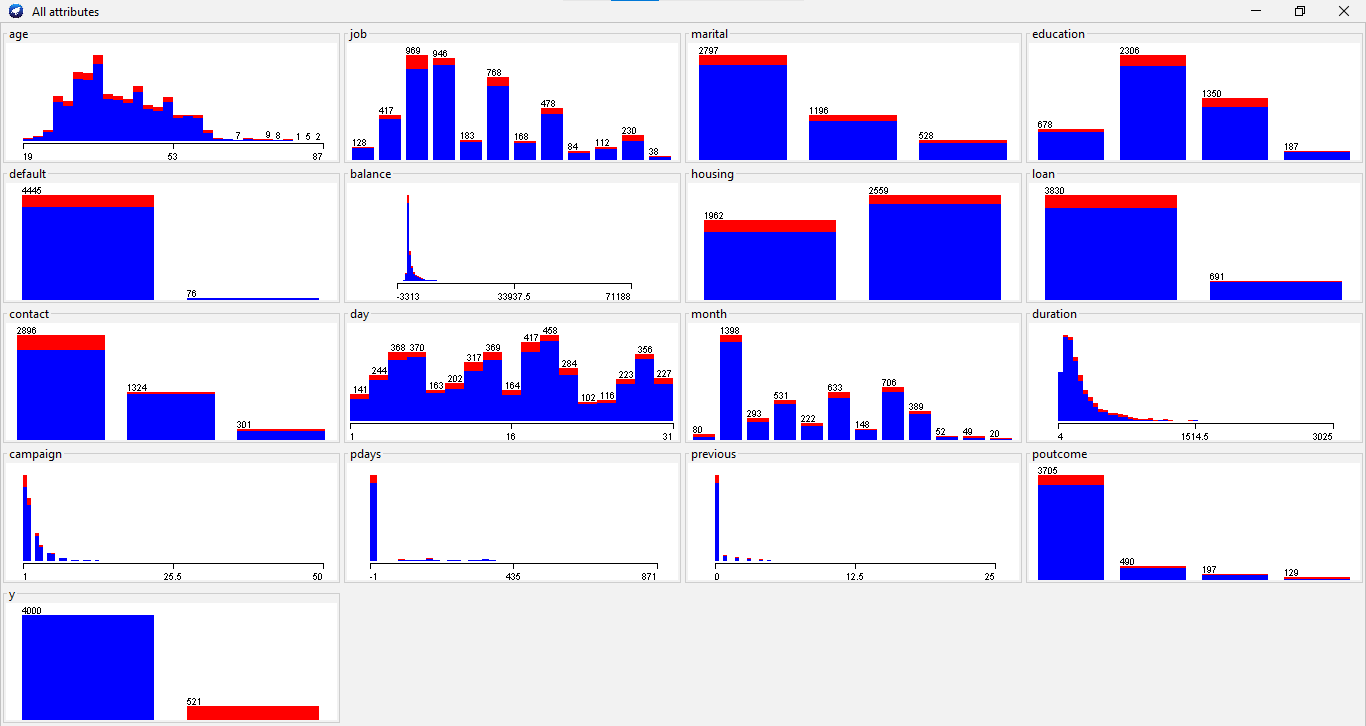


Figure 2 :Visualization of all attributes

**Model Development:**

Classification is one form of prediction which occurs very frequently in everyday life. Essentially it involves dividing up objects so that each is assigned to one of a number of mutually exhaustive and exclusive categories known as classes. If the designated attribute is categorical, the task is called classification. In classification, there are many classifiers.

For our project, we need to develop 3 classifier models and compare them which is better for our dataset. The classifiers are Naive Bayes, Decision tree, and K-nn (nearest neighbor).

**Naive Bayes:**

Naive Bayes is a method of classification that uses probability theory to

find the most likely of the possible classifications. It is dependent on categorical attributes for all attributes. The Naive Bayes algorithm combines the prior probability and conditional probabilities in a single formula. The Naive Bayes algorithm is comparatively easy to build and apply.

***Process:***

For Naive Bayes first task is to covert all the attributes to nominal. then from the **weka** tool, we need to select the naive Bayes from the classifiers.

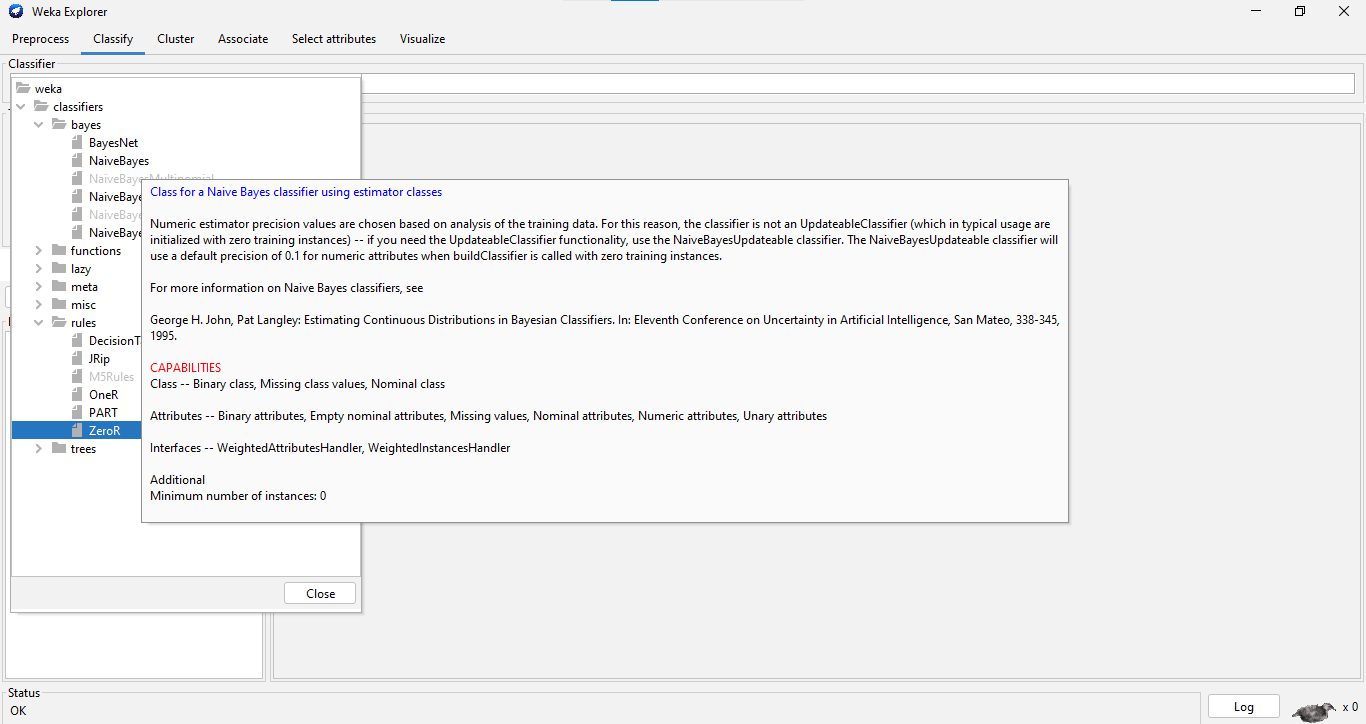


Figure 3: Naive Bayes selection

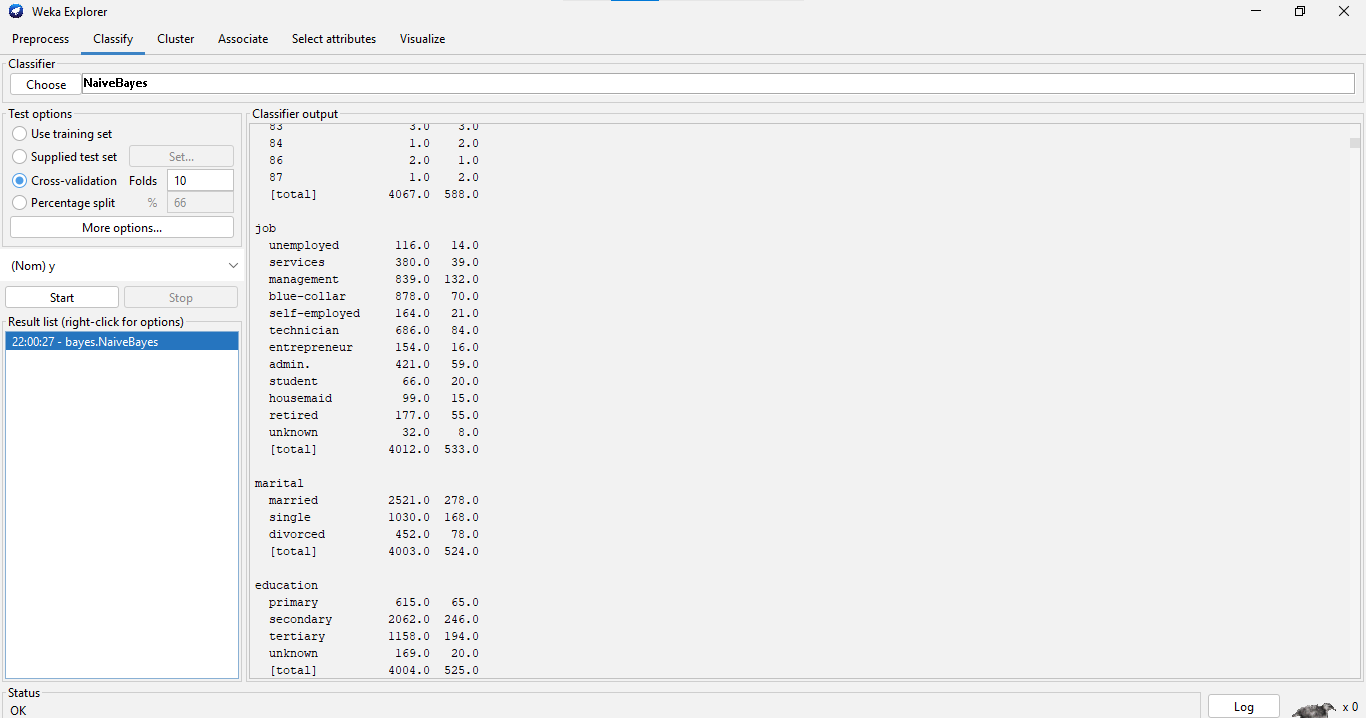


Figure 4: Naive Bayes results

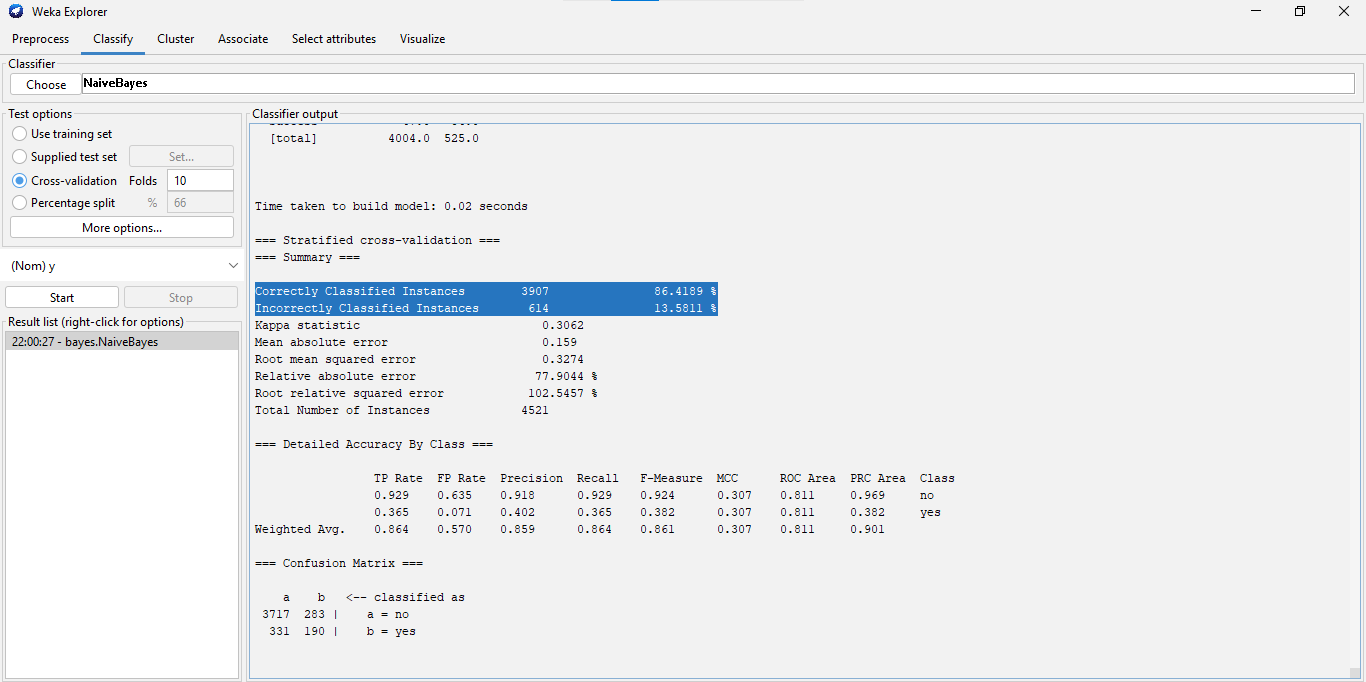


Figure 5: Naive Bayes results(correctly, incorrectly classified instances)

***Results:***

Time is taken to build the model: 0.02 seconds

Correctly Classified Instances 3907 86.4189 %

Incorrectly Classified Instances 614 13.5811 %

=== Confusion Matrix ===

a b <-- classified as

3717 283 | a = no

331 190 | b = yes

**Nearest Neighbour(k-nn):**

Nearest Neighbour classification is mainly used when all attribute values are continuous, although they can be modified to deal with categorical attributes. It is usual to base the classification on those of the k nearest neighbors (where k is a small integer such as 3 or 5), not just the nearest one. The method is then known as k-Nearest Neighbour or just k-NN classification.

**Process:**

For k-NN we need to choose IBk from the weka tool lazy folder. Then apply that

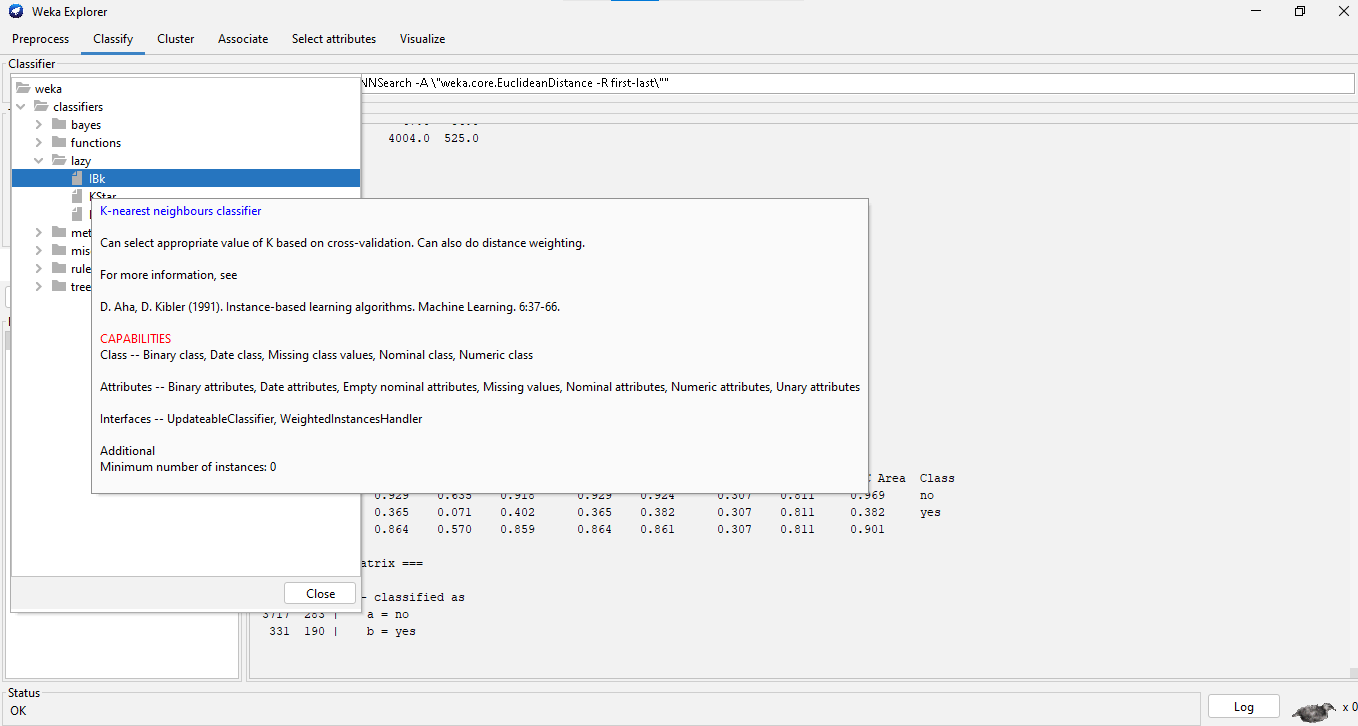


Figure 6: k-NN selection

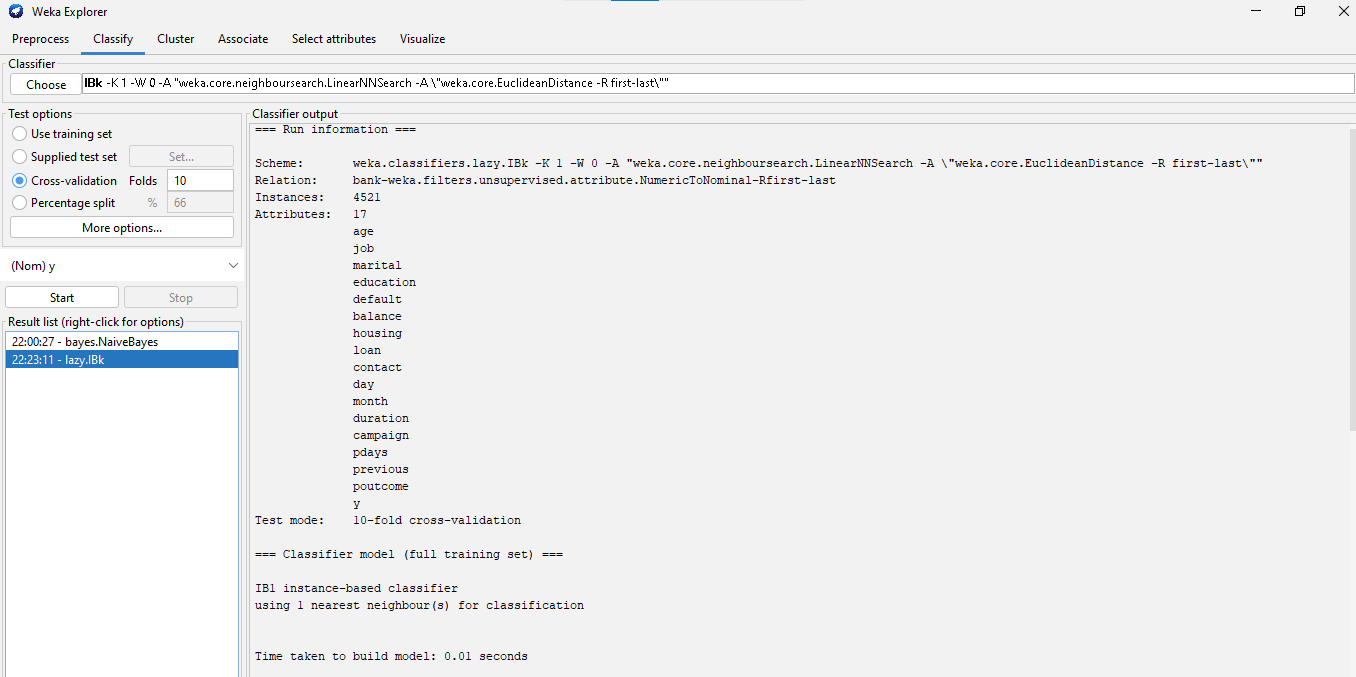


Figure 7: k-NN results

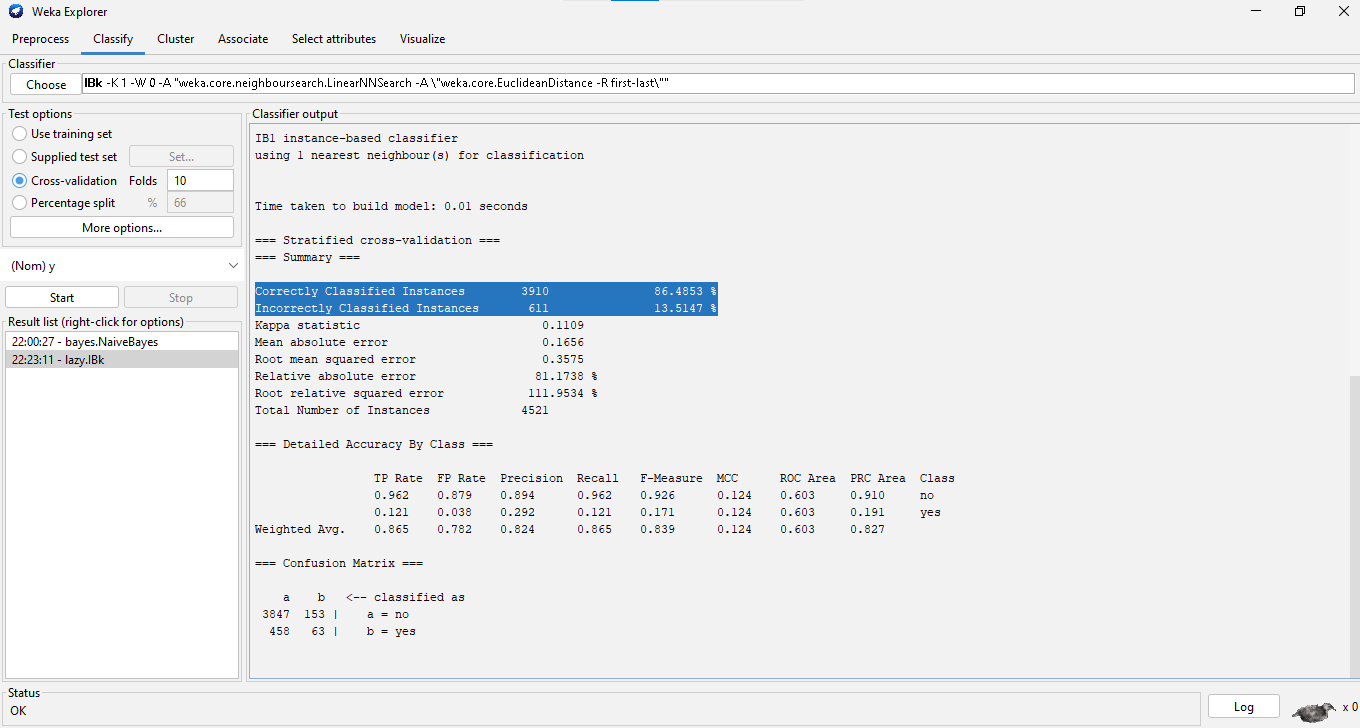


Figure 8: k-NN (correctly, incorrectly classified)

**Results:**

Time is taken to build the model: 0.01 seconds

Correctly Classified Instances 3910 86.4853 %

Incorrectly Classified Instances 611 13.5147 %

=== Confusion Matrix ===

a b <-- classified as

3847 153 | a = no

458 63 | b = yes

**Decision tree:**

The Decision Tree classification approach consists of three components: the root node, the branch (edge or link), and the leaf node. The root node contains the test condition for various attributes, the branch node represents all possible outcomes in the test, and the leaf nodes contain the label of the class to which it belongs. The root node is located at the beginning of the tree, often known as the tree's top.

J48 is an algorithm used by C4.5 to create a decision tree (an extension of ID3). It is often referred to as a statistical classifier. To use the j48 decision tree, we must now perform the following procedures.

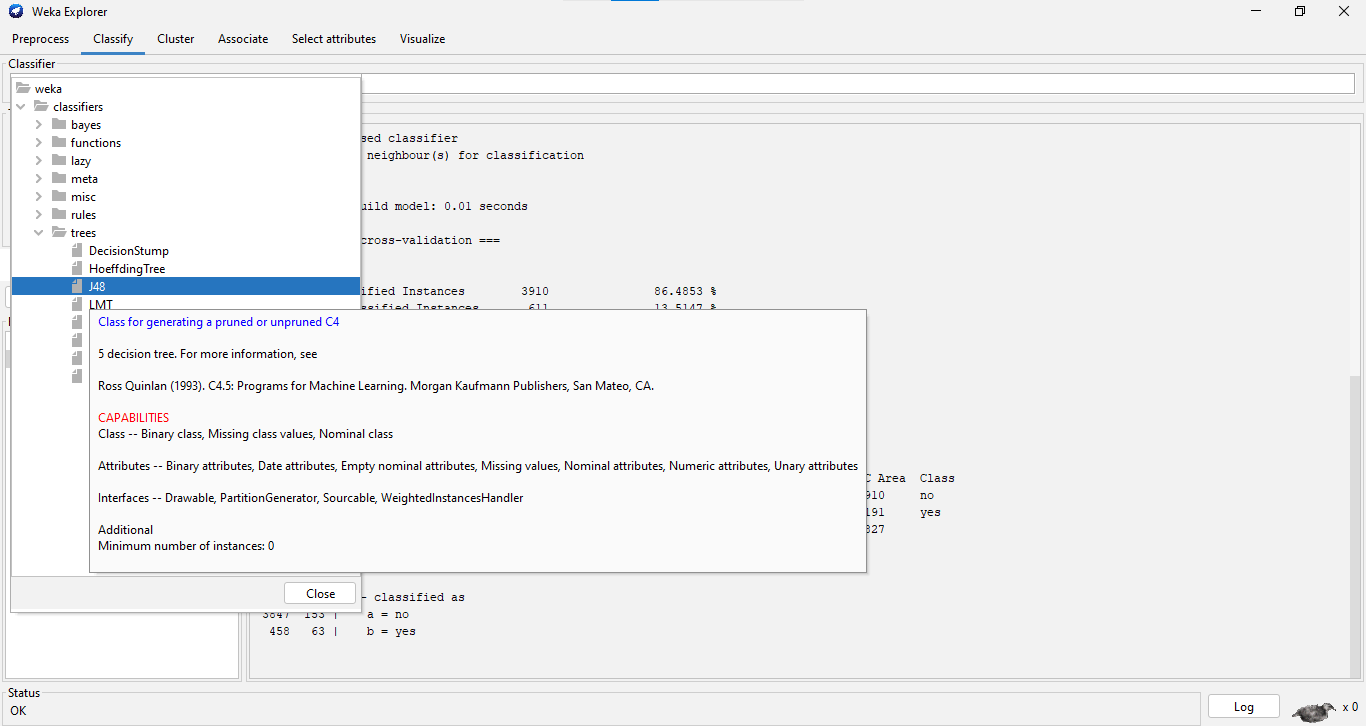


Figure 9: Decision tree selection

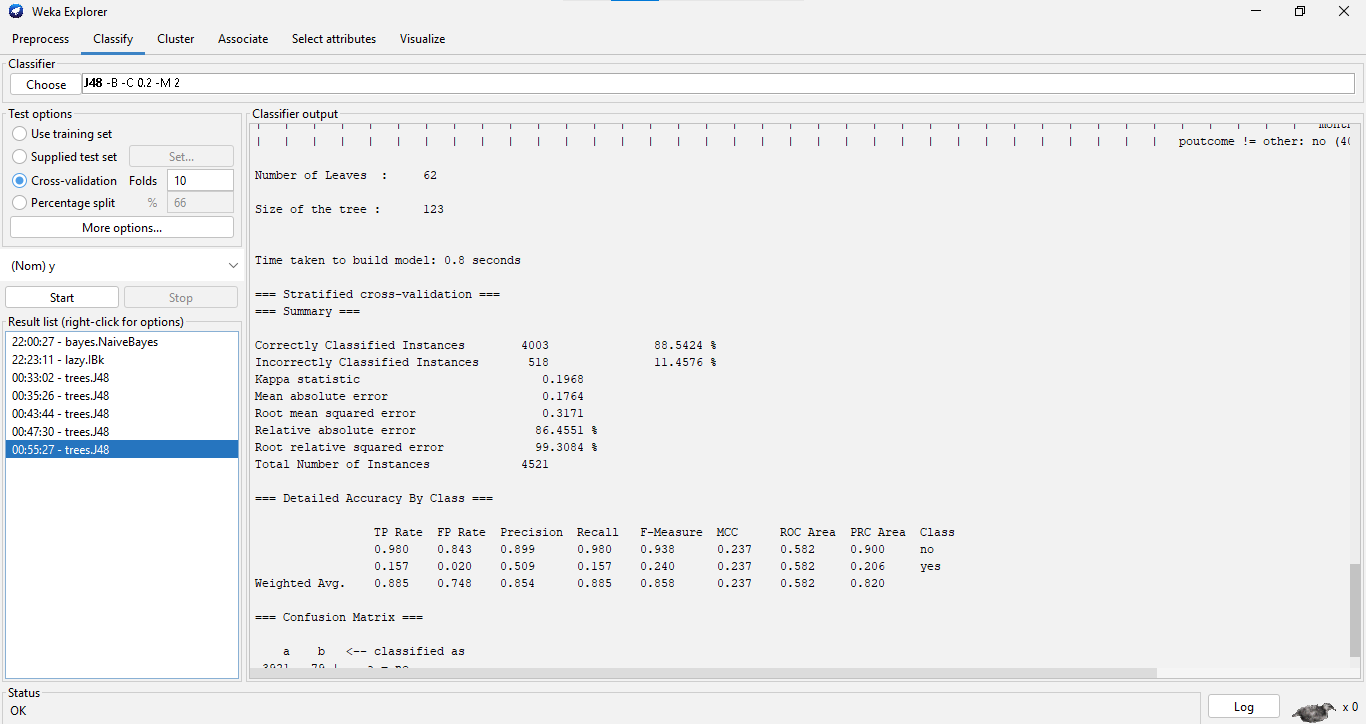


Figure 10: Decision tree results

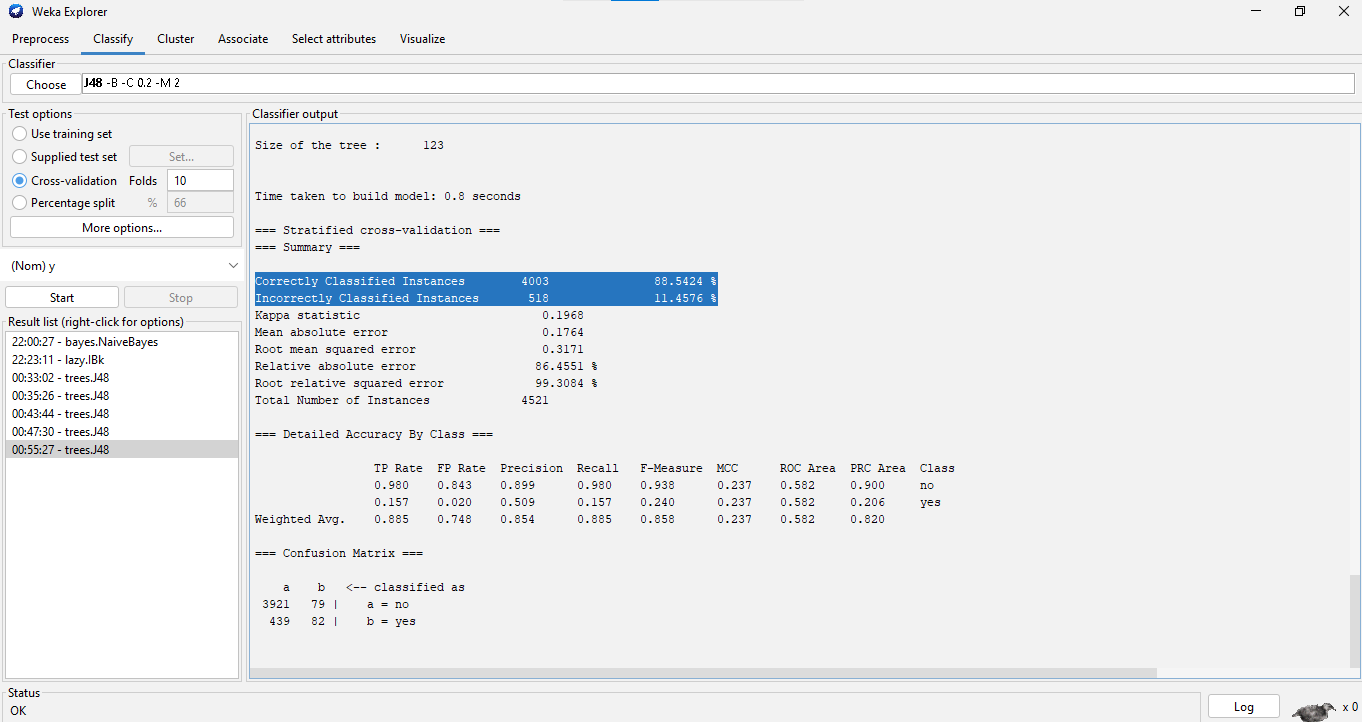


Figure 11: Decision tree(correctly, incorrectly classified)

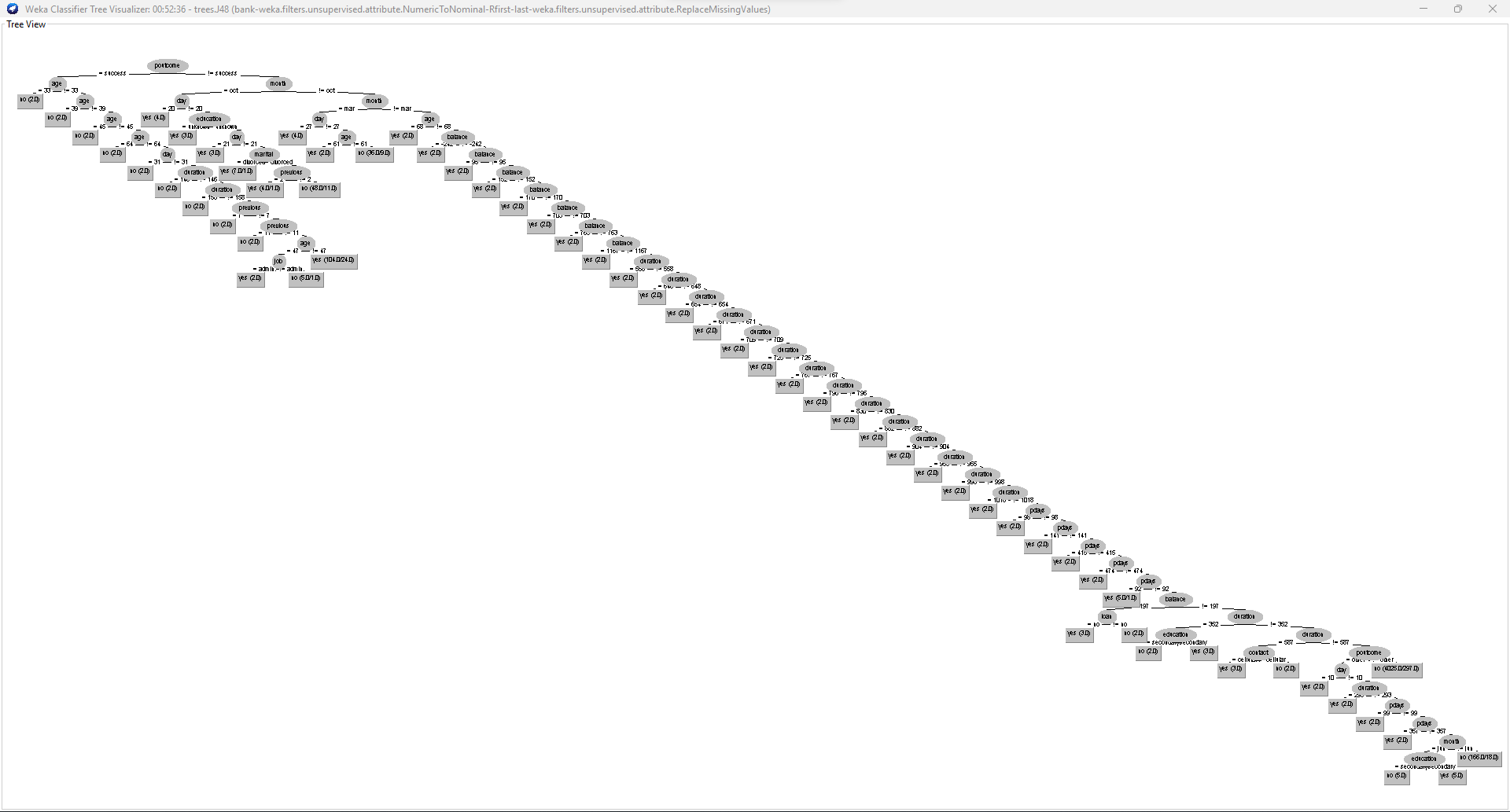


Figure 12: visualization of tree

**Results:**

Time is taken to build the model: 0.8 seconds

Correctly Classified Instances 4003 88.5424 %

Incorrectly Classified Instances 518 11.4576 %

=== Confusion Matrix ===

a b <-- classified as

3921 79 | a = no

439 82 | b = yes

**Discussion & Conclusion:**

Naive Bayes:

Total instances were 4522 instances in the data set. In naive Bayes,

correctly classified Instances were 3907 (86.4189 %) and incorrectly classified instances were 614 (13.5811 %). Total time is taken to build the model: 0.02 seconds.

k-NN:

The number of instances was 4522. In k-nn (nearest neighbor), correctly classified instances were 3910(86.4853 %) and incorrectly classified Instances were 611(13.5147 %). Total time is taken to build the model: 0.01 seconds.

Decision tree:

The number of instances was 4522.In decision tree,correctly classified Instances were 4003(88.5424 %),incorrectly classified Instances were 518(11.4576 %).Total time taken to build the model: 0.8 seconds.

We can clearly see that in the decision tree(88.54%)the number of correctly classified instances is higher than k-NN(86.43%) and Naive Bayes(86.41%). Because the Decision tree has a higher percentage of properly classified instances than Nave Bayes and KNN, we may conclude that it is better in this dataset. Also, we can conclude the answer by seeing the confusion matrix

| Naive Bayes  === Confusion Matrix === | k-NN  === Confusion Matrix === | Decision tree  === Confusion Matrix === |
| --- | --- | --- |
| a b <-- classified as  3717 283 | a = no  331 190 | b = yes | a b <-- classified as  3847 153 | a = no  458 63 | b = yes | a b <-- classified as  3921 79 | a = no  439 82 | b = yes |

If we see closely in decision tree no is predicted correctly 3921,in k-nn no is correctly predicted 3847 times and in naive Bayes it has correctly predicted 3717 times. In the decision tree yes is correctly predicted 439 times wrongly and 82 times correctly.

In conclusion, it is clearly seen that the Decision tree is the best algorithm for this dataset.