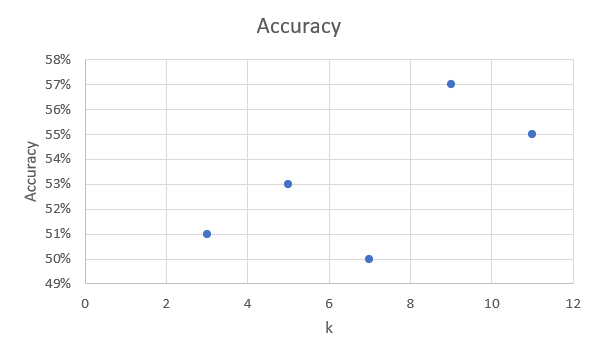
**Assignment 7 kNN**

**Q1 Related py file : Xu\_Yuhan\_Assign\_7\_stock\_knn\_q1.py**

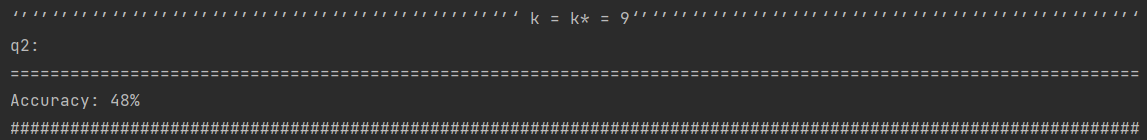
**Other Related py file : Xu\_Yuhan\_Assign\_7\_stock\_knn\_q2-5.py**

## **1. take k = 3, 5, 7, 9, 11. For each value of k compute the accuracy of your k-NN classifier on year 1 data. On x axis you plot k and on y-axis you plot accuracy. What is the optimal value of k for year 1?**

According to the py file, we can get k\* = 9

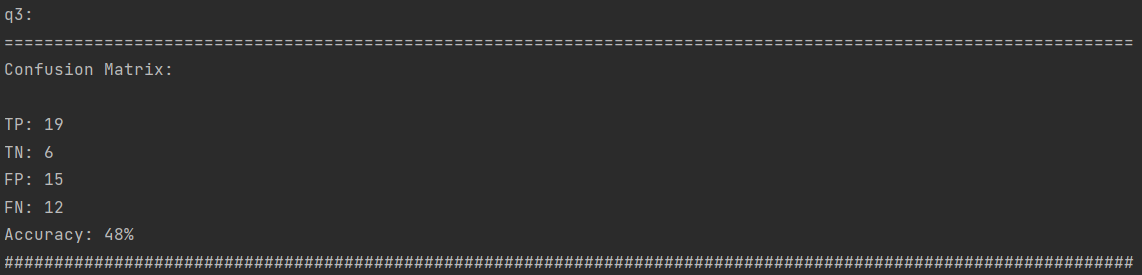


## **2. use the optimal value of k from year 1 to predict labels for year 2. What is your accuracy?**



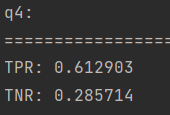
The optimal value of k from year 1 is 9, and the accuracy of year 2 when k = 9 is 48

## **3. using the optimal value for k from year 1, compute the confusion matrix for year 2**



| TP | TN | FP | FN | Accuracy |
| --- | --- | --- | --- | --- |
| 19 | 6 | 15 | 12 | 48% |

## **4. what is true positive rate (sensitivity or recall) and true negative rate (specificity) for year 2?**



The true positive rate (sensitivity or recall), TPR, is 0.612903

The true negative rate (specificity), TNR, is 0.285714

## **5. implement a trading strategy based on your labels for year 2 and compare the performance with the ”buy-and-hold” strategy. Which strategy results in a larger amount at the end of the year?**

Answer:

Because the second year is a big drop compared to the first year, it is impossible to hold for a long time. Therefore, the strategies we can take are simple, all short-term actions. Buy when it falls compared to the previous day, and sell when it rises compared to the previous day, so as to maximize the benefits.

Compared to “buy-and-hold”, I think my strategy will result in a larger amount at the end of the year.