Name Surname : **Göktuğ YILDIRIM**
Student ID : **041502012**

COMP462- Introduction to Machine Learning

Assignment 1

Table 1: k-NN classification accuracies for different k and distance metrics

	Euclidean Distance		**Manhattan Distance**	
	Accuracy (%)	Error Count	Accuracy (%)	Error Count
k=1	93.33	4/60	93.33	4/60
k=3	96.66	2/60	96.66	2/60
k=5	96.66	2/60	96.66	2/60
k=7	96.66	2/60	96.66	2/60
k=9	96.66	2/60	96.66	2/60
k=11	96.66	2/60	96.66	2/60
k=13	96.66	2/60	95	3/60
k=15	95	3/60	95	3/60
Average	96.03	2.375/60	95.82	2.5/60
Standard Deviation	1.15	0.69	1.17	0.707

1) The Euclidean Distance is better for the Iris dataset because of the averages and standard deviations of success metrics which are shown in Table 1. It indicates that k-NN algorithm may give more successful results by running with Euclidean Distance than Manhattan Distance. Because Euclidean Distance has a smaller average and standard deviation in terms of accuracy and error count.

2) k-NN algorithms performs with k parameters between 2-13 very well because of accuracies. Also, low accuracies have been observed when the k parameter was greater than 30.

Name Surname : **Göktuğ YILDIRIM**
Student ID : **041502012**

Bonus 1:

Figure 1: Decision boundaries found by the k-NN algorithm (k=1, distance metric=Euclidean distance) using the training set. Borders between red, blue and yellow regions are the decision boundaries.

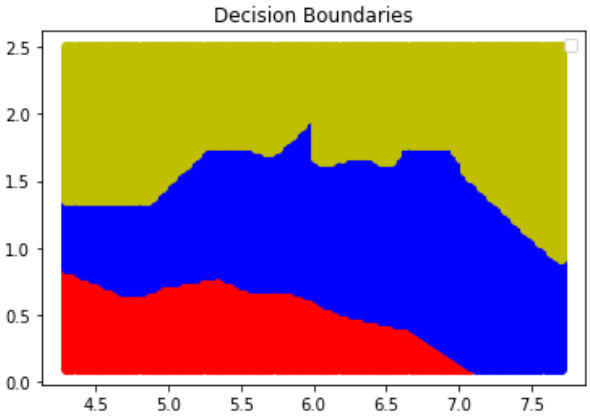


Figure 2: Decision boundaries found by the k-NN algorithm (k=5, distance metric=Euclidean distance) using the training set. Borders between red, blue and yellow regions are the decision boundaries.

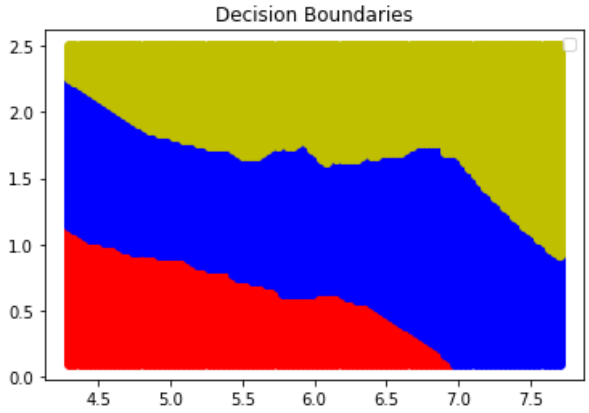
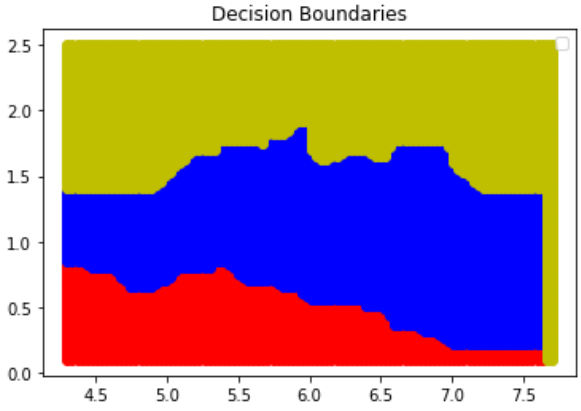


Figure 3: Decision boundaries found by the k-NN algorithm (k=1, distance metric=Manhattan distance) using the training set. Borders between red, blue and yellow regions are the decision boundaries.



Name Surname : **Göktuğ YILDIRIM**
Student ID : **041502012**

Bonus 2:

Table 2: k-NN classification accuracies for different k and distance metrics with Sckit Learn implementation

	Euclidean Distance		**Manhattan Distance**	
	Accuracy (%)	Error Count	Accuracy (%)	Error Count
k=1	93.33	4/60	93.33	4/60
k=3	96.66	2/60	96.66	2/60
k=5	96.66	2/60	96.66	2/60
k=7	96.66	2/60	96.66	2/60
k=9	96.66	2/60	96.66	2/60
k=11	96.66	2/60	95	3/60
k=13	96.66	2/60	95	3/60
k=15	95	3/60	95	3/60

- All results are the same as the model developed without using any machine learning framework except in case of k=11, distance: Manhattan Distance. I guess, this difference is due to majority voting. Code Block 1 shows us how my algorithm finds the most frequent class. In case of the equal count of class, it chooses the minimum value but the Sckit Learn algorithm may choose the frequent class randomly.

```
frequent_class = max(set(classes), key=classes.count) #find the most frequent class
```

Code Block 1: Finding the most frequent class

Name Surname : **Göktuğ YILDIRIM**
Student ID : **041502012**

Decision Boundaries with Sckit Learn Prediction:

Figure 4: Decision boundaries found by the k-NN algorithm (k=1, distance metric=Euclidean distance) using the training set. Borders between red, blue and yellow regions are the decision boundaries.

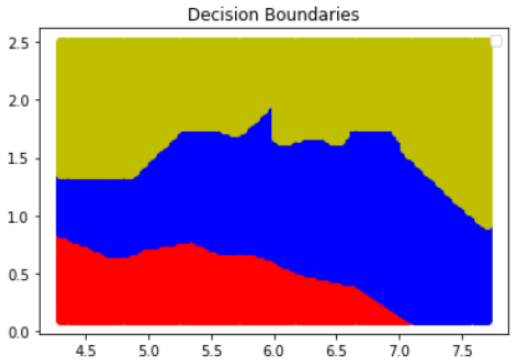


Figure 5: Decision boundaries found by the k-NN algorithm (k=5, distance metric=Euclidean distance) using the training set. Borders between red, blue and yellow regions are the decision boundaries.

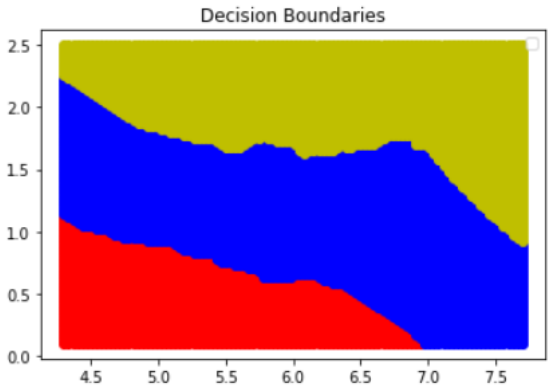


Figure 6: Decision boundaries found by the k-NN algorithm (k=1, distance metric=Manhattan distance) using the training set. Borders between red, blue and yellow regions are the decision boundaries.