CS4342-HW6

Ivan Martinovic

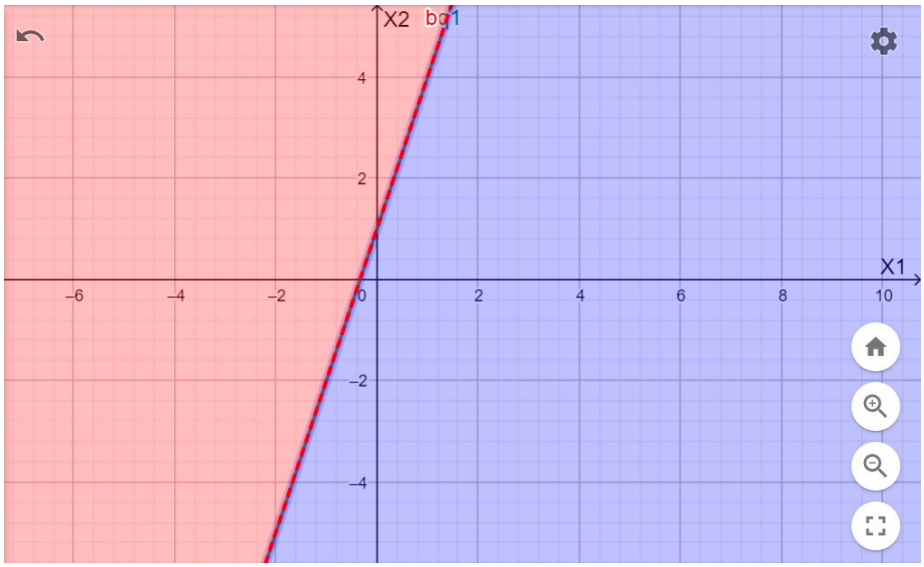
December 2021

1 Conceptual and Theoretcal Questions

1.1 Question 1

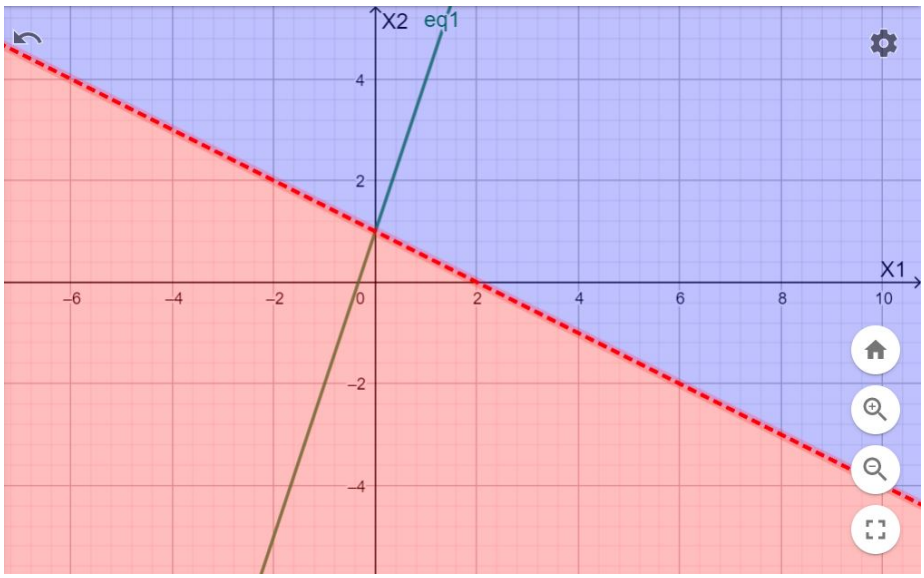
This problem involves hyperplanes in two dimensions.

(a) Sketch the hyperplane $1 + 3X_1 - X_2 = 0$. Indicate the set of points for which $1+3X_1-X_2 > 0$, as well as the set of points for which $1+3X_1-X_2 < 0$.



Blue region are all points where $1 + 3X_1 - X_2 > 0$.
Red region are all points where $1 + 3X_1 - X_2 < 0$.
The boundary between the two regions is the hyperplane $1+3X_1-X_2 = 0$.

(b) On the same plot, sketch the hyperplane $-2 + X_1 + 2X_2 = 0$. Indicate the set of points for which $-2+X_1+2X_2 > 0$, as well as the set of points for which $-2 + X_1 + 2X_2 < 0$.



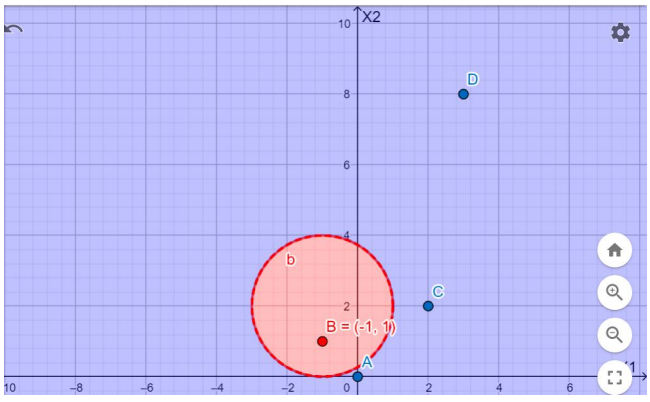
Blue region are all points where $-2 + X_1 + 2X_2 > 0$.
Red region are all points where $-2 + X_1 + 2X_2 < 0$.
The boundary between the two is the hyperplane $-2 + X_1 + 2X_2 = 0$.

1.2 Question 2

We have seen that in p = 2 dimensions, a linear decision boundary takes the form $\beta_0 + \beta_1 X_1 + \beta_2 X_2 = 0$. We investigate a non-linear decision boundary.

(a) Sketch the curve:

$$(1 + X_1)^2 + (2 - X_1)^2 = 4$$



The curve is a circle centered at (-1, 2) with radius 2.

(b) On your sketch, indicate the set of points for which $(1+X_1)^2+(2-X_1)^2 > 4$, as well as the set of points for which $(1+X_1)^2+(2-X_1)^2 \leq 4$

Blue region are all points where $(1+X_1)^2+(2-X_1)^2 > 4$.
Red region are all points where $(1+X_1)^2+(2-X_1)^2 \leq 4$.

(c) Suppose that a classifier assigns an observation to the blue class if

$$(1+X_1)^2+(2-X_1)^2 > 4$$

and to the red class otherwise. To what class is the observation (0, 0) classified? (-1, 1)? (2, 2)? (3, 8)?

(0, 0) is represented by A, and since it is in the blue region it will be assigned to the blue class.
(-1, 1) is represented by B, and since it is in the red region it will be assigned to the red class.
(2, 2) is represented by C, and since it is in the blue region it will be assigned to the blue class.
(3, 8) is represented by D, and since it is in the blue region it will be assigned to the blue class.

(d) Argue that while the decision boundary in (c) is not linear in terms of X_1 and X_2, it is linear in terms of X_1, X_2, X_1^2, and X_2^2.

Expand: $(1+X_1)^2+(2-X_1)^2 = 4$:

$$1+2X_1+X_1^2+4-4X_2+X_2^2 = 4$$

$$1+2X_1+X_1^2-4X_2+X_2^2 = 0$$

This can be thought of as an equation for a hyperplane in 4 dimensions, where dimensions are X_1, X_2, X_1^2, and X_2^2.

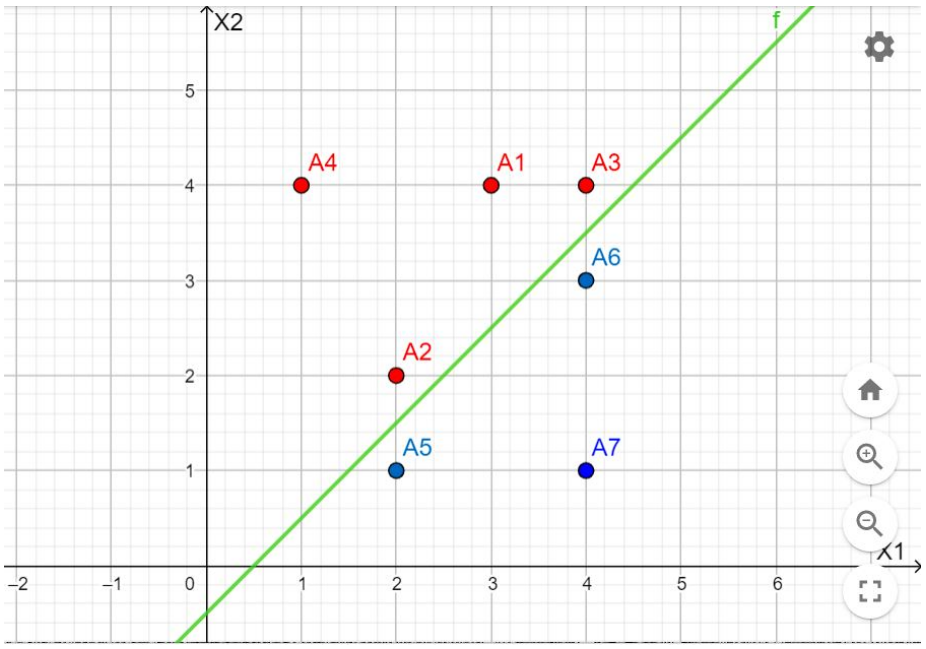
1.3 Question 3

Here we explore the maximal margin classifier on a toy data set.

(a) We are given n = 7 observations in p = 2 dimensions. For each observation, there is an associated class label.

Obs.	X_1	X_2	Y
1	3	4	Red
2	2	2	Red
3	4	4	Red
4	1	4	Red
5	2	1	Blue
6	4	3	Blue
7	4	1	Blue

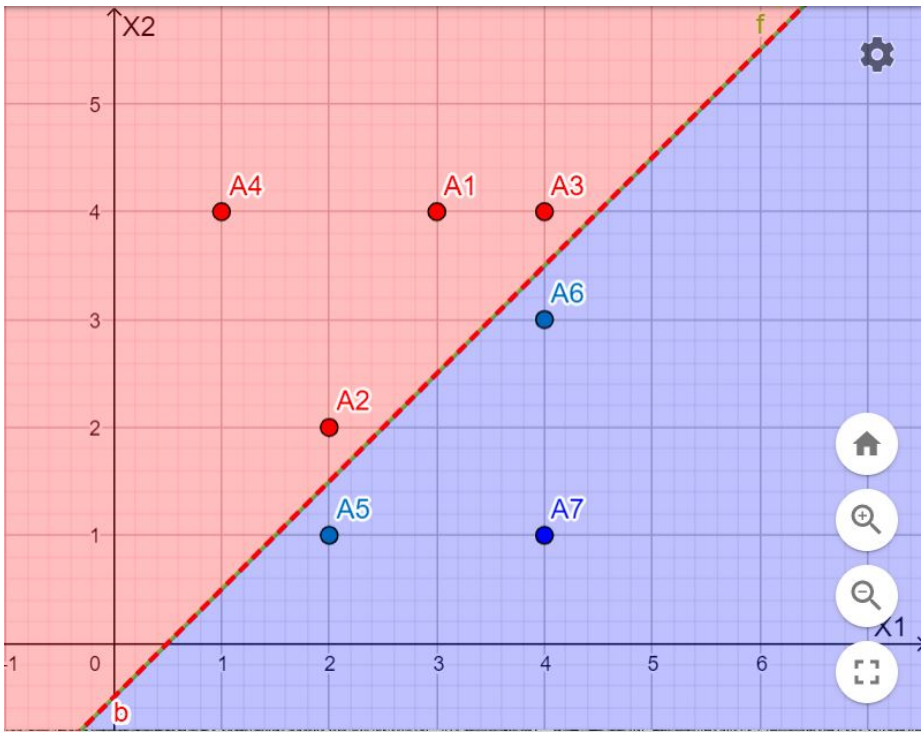
(b) Sketch the optimal separating hyperplane, and provide the equation for this hyperplane (of the form (9.1)).



The hyperplane equation:

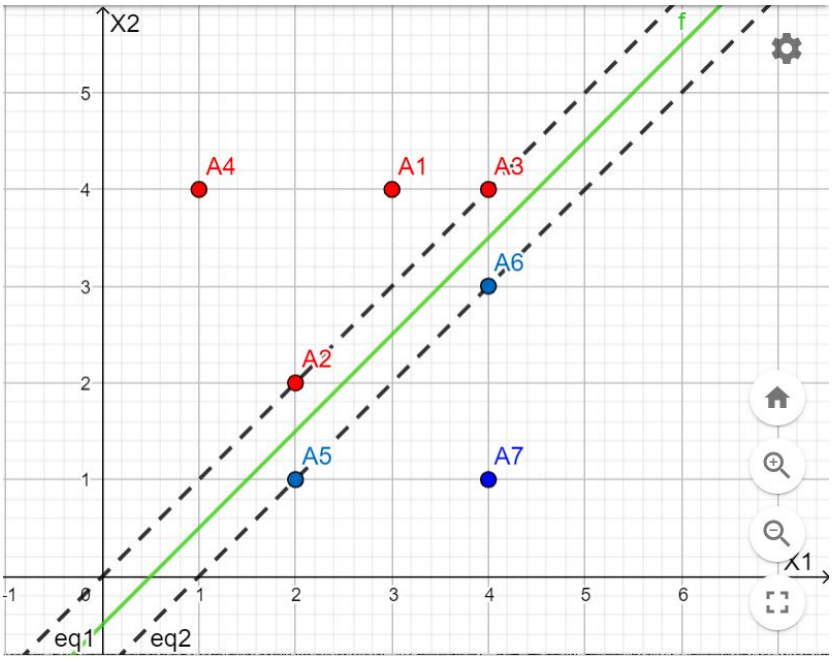
$$-0.5 + X_1 - X_2 = 0$$

(c) Describe the classification rule for the maximal margin classifier. It should be something along the lines of "Classify to Red if $\beta_0 + \beta_1 X_1 + \beta_2 X_2 > 0$, and classify to Blue otherwise." Provide the values for β_0, β_1 and β_2.

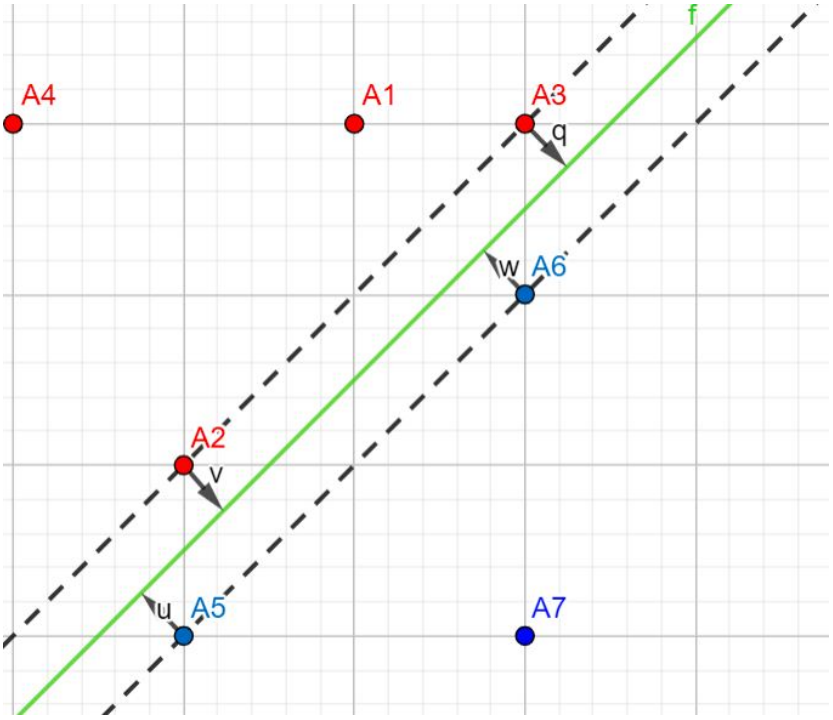


Classification rule:
Classify to Red if $-0.5 + X_1 - X_2 < 0$, which corresponds to points in the red region.
Classify to Blue if $-0.5 + X_1 - X_2 > 0$, which corresponds to points in the blue region.

(d) On your sketch, indicate the margin for the maximal margin hyperplane.



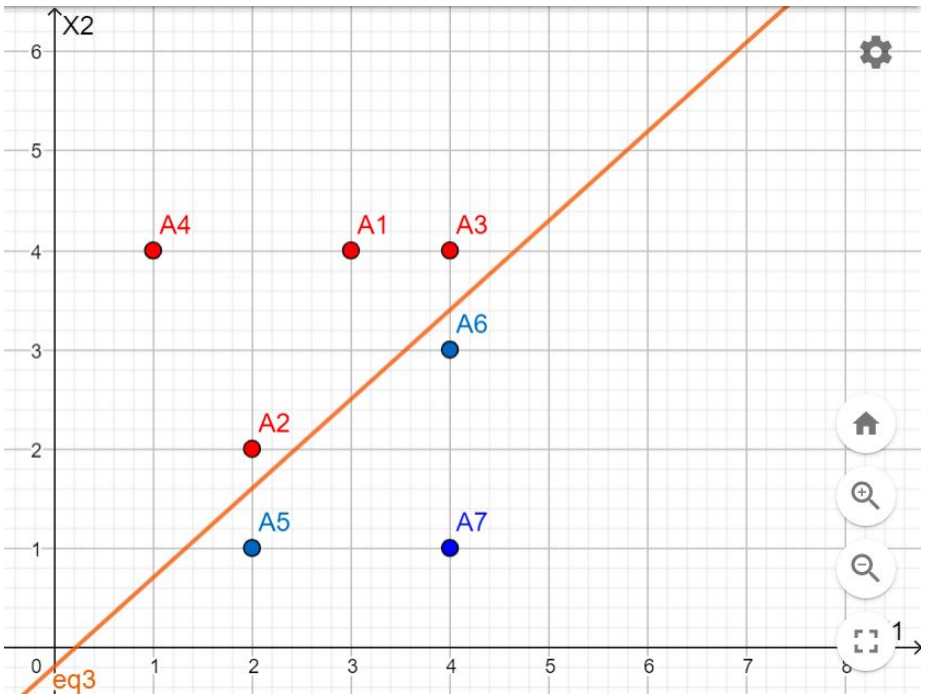
(e) Indicate the support vectors for the maximal margin classifier.



(f) Argue that a slight movement of the seventh observation would not affect the maximal margin hyperplane.

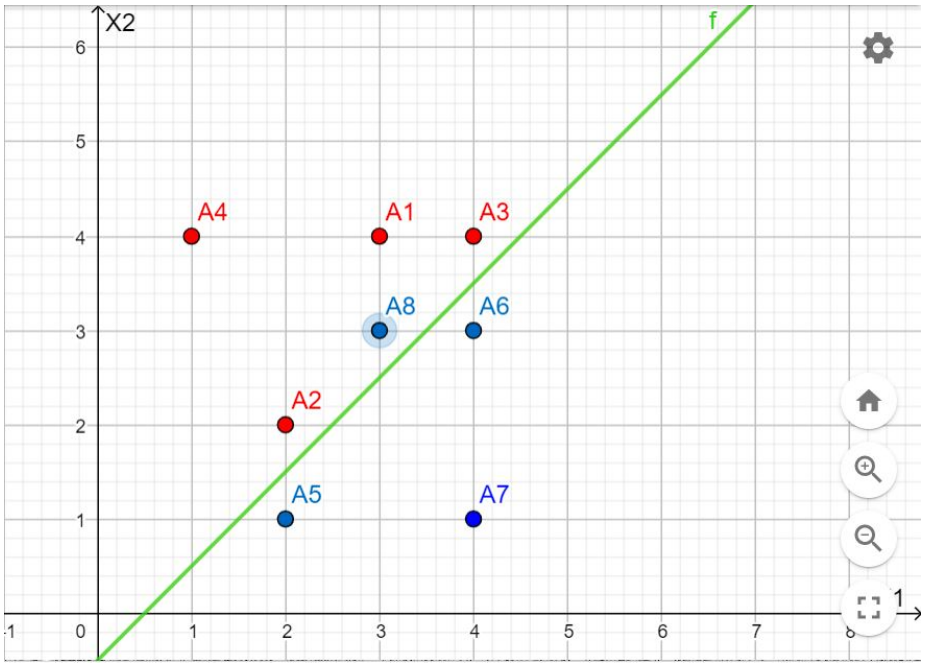
The seventh observation does not define a support vector. Since support vectors are all that define the hyperplane, a slight movement in the seventh observation would therefore not affect the maximal margin hyperplane, since it wouldn't change any of the support vectors.

(g) Sketch a hyperplane that is not the optimal separating hyperplane, and provide the equation for this hyperplane.



$$-0.2 + 0.9X_1 - X_2 = 0$$

(h) Draw an additional observation on the plot so that the two classes are no longer separable by a hyperplane.



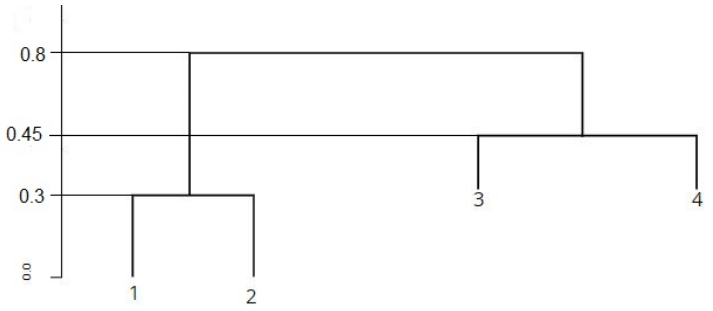
1.4 Question 4

Suppose that we have four observations, for which we compute a dissimilarity matrix, given by:

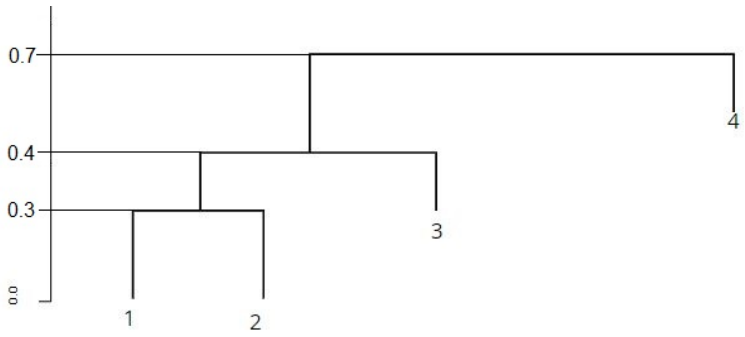
$$\begin{bmatrix} & 0.3 & 0.4 & 0.7 \\ 0.3 & & 0.5 & 0.8 \\ 0.4 & 0.5 & & 0.45 \\ 0.7 & 0.8 & 0.45 & \end{bmatrix}.$$

For instance, the dissimilarity between the first and second observations is 0.3, and the dissimilarity between the second and fourth observations is 0.8.

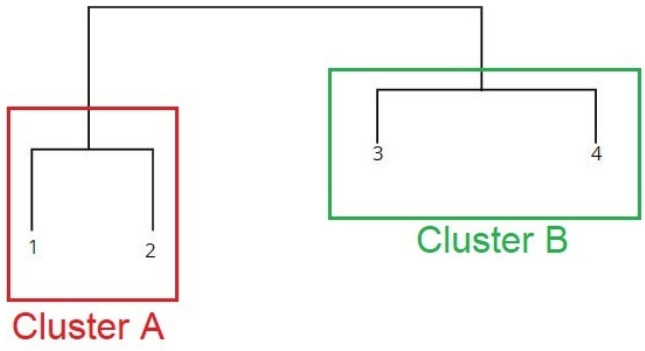
(a) On the basis of this dissimilarity matrix, sketch the dendrogram that results from hierarchically clustering these four observations using complete linkage. Be sure to indicate on the plot the height at which each fusion occurs, as well as the observations corresponding to each leaf in the dendrogram.



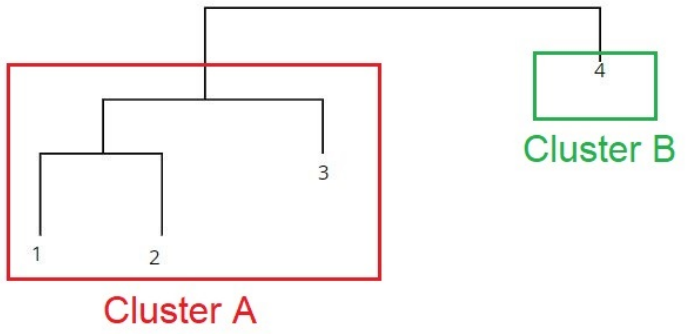
(b) Repeat (a), this time using single linkage clustering.



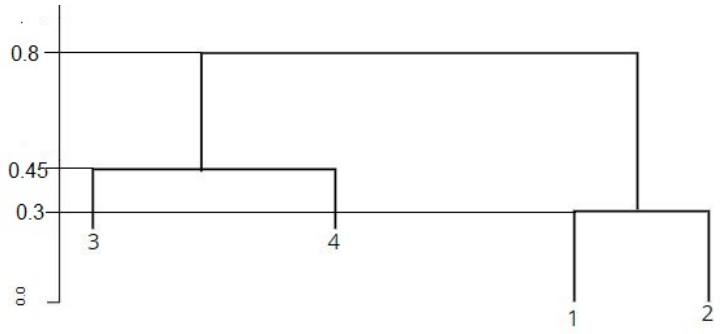
(c) Suppose that we cut the dendogram obtained in (a) such that two clusters result. Which observations are in each cluster?



(d) Suppose that we cut the dendogram obtained in (b) such that two clusters result. Which observations are in each cluster?



(e) It is mentioned in the chapter that at each fusion in the dendrogram, the position of the two clusters being fused can be swapped without changing the meaning of the dendrogram. Draw a dendrogram that is equivalent to the dendrogram in (a), for which two or more of the leaves are repositioned, but for which the meaning of the dendrogram is the same.

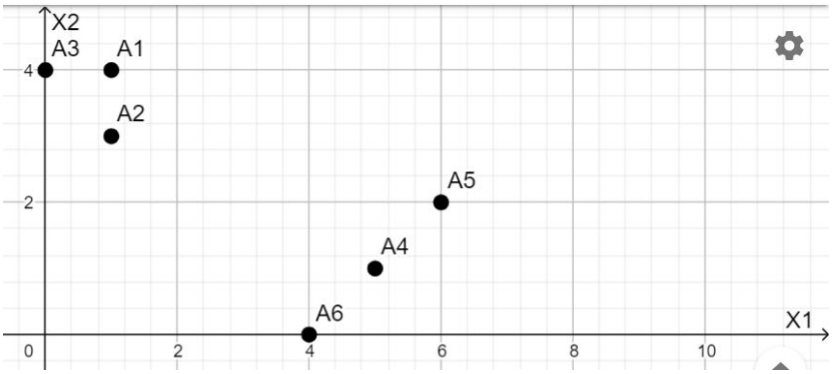


1.5 Question 5

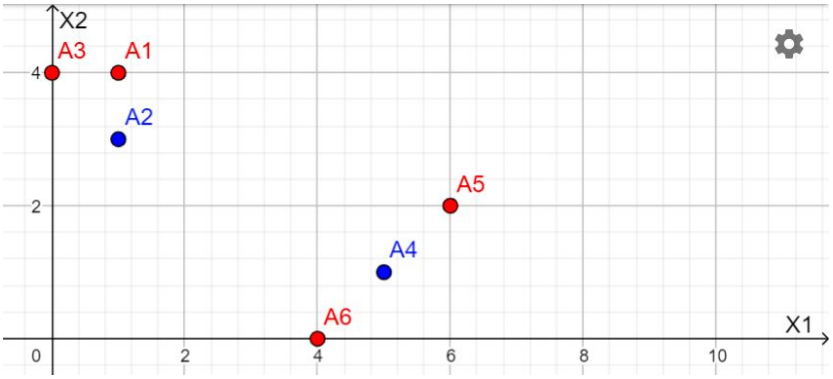
In this problem, you will perform K-means clustering manually, with K = 2, on a small example with n = 6 observations and p = 2 features. The observations are as follows.

Obs.	X_1	X_2
1	1	4
2	1	3
3	0	4
4	5	1
5	6	2
6	4	0

(a) Plot the observations.



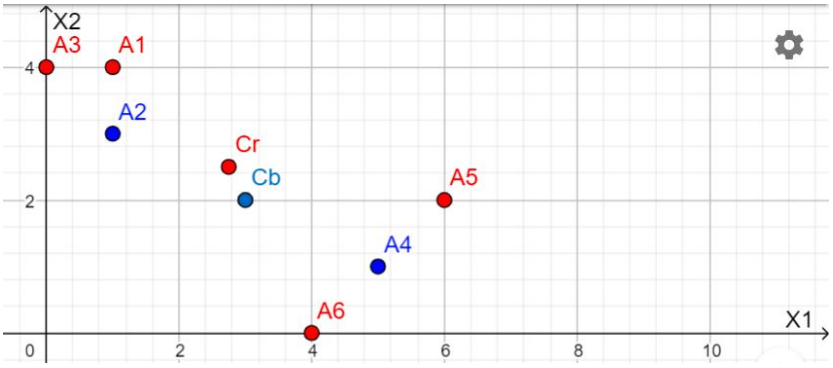
(b) Randomly assign a cluster label to each observation. Report the cluster labels for each observation.
Suppose 1,3,5 and 6 are assigned to Red and 2 and 4 are assigned to Blue.



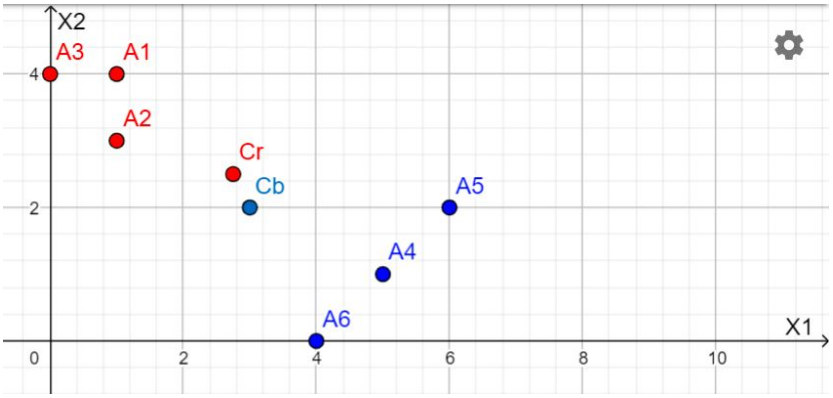
(c) Compute the centroid for each cluster.

$$C_R = (2.75, 2.5)$$

$$C_B = (3, 2)$$



(d) Assign each observation to the centroid to which it is closest, in terms of Euclidean distance. Report the cluster labels for each observation.

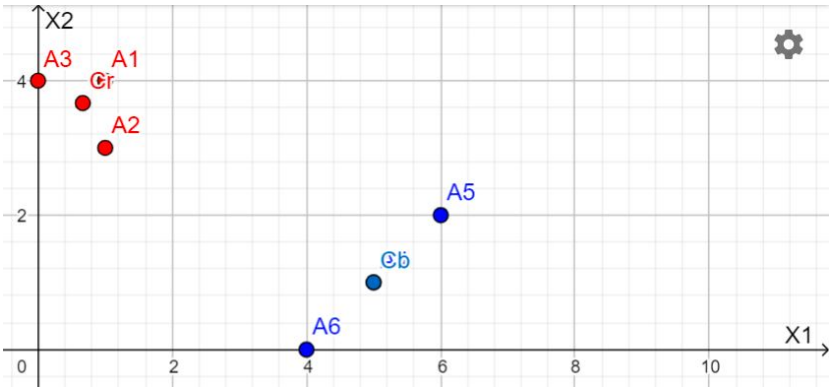


Observations 1,2 and 3 are now Red.
Observations 4,5 and 6 are now Blue.

(e) Repeat (c) and (d) until the answers obtained stop changing.

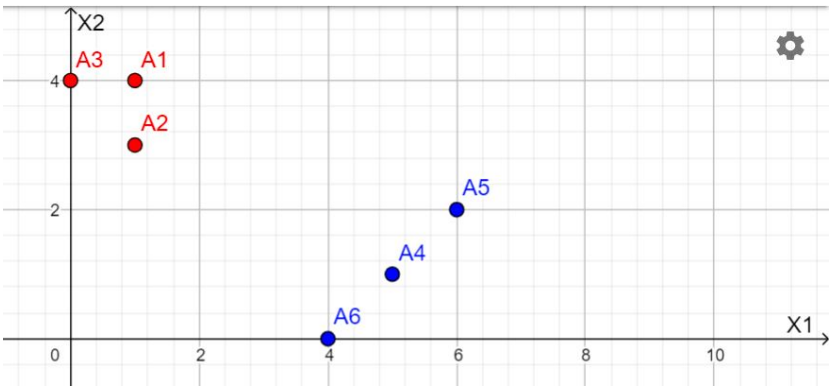
$$C_r = (\frac{2}{3}, \frac{11}{3})$$

$$C_b = (5, 1)$$



Same observations are assigned the same classes. We stop here.

(f) In your plot from (a), color the observations according to the cluster labels obtained.



1.6 Question 6

Suppose that for a particular data set, we perform hierarchical clustering using single linkage and using complete linkage. We obtain two dendrograms.

(a) At a certain point on the single linkage dendrogram, the clusters {1, 2, 3} and {4, 5} fuse. On the complete linkage dendrogram, the clusters {1, 2, 3} and {4, 5} also fuse at a certain point. Which fusion will occur higher on the tree, or will they fuse at the same height, or is there not enough information to tell?
There is not enough information to tell.
The height at which two cluster fuse together is the measure of their dissimilarity.
In single linkage dendograms, two clusters fuse together if the minimum dissimilarity between all pairs of observations in the two clusters is smallest amongst all clusters in the dendrogram up to that height. Say this minimum dissimilarity between clusters {1, 2, 3} and {4, 5} has a value a.
In complete linkage dendograms, two clusters fuse together if the maximum dissimilarity between all pairs of observations in the two clusters is smallest amongst all clusters in the dendrogram up to that height. Say this maximum dissimilarity between clusters {1, 2, 3} and {4, 5} has a value b.
If $a = b$ then the two clusters will merge at the same height for both dendrograms. if $b > a$, then the two clusters will not merge at the same height.

(b) At a certain point on the single linkage dendrogram, the clusters {5} and {6} fuse. On the complete linkage dendrogram, the clusters {5} and {6} also fuse at a certain point. Which fusion will occur higher on the tree, or will they fuse at the same height, or is there not enough information to tell?
The clusters will necessarily fuse at the same height in both dendrograms. Let's use the same analysis as previously:'

Say the minimum and maximum dissimilarity between clusters {5} and {6} are a and b respectively. Since the two clusters contain only a single observation, it must be true that $a = b$
Therefore the clusters must necessarily fuse at the same height in both dendrograms.

1.7 Question 7

In words, describe the results that you would expect if you performed K-means clustering of the eight shoppers in Figure 10.14 – shown below, on the basis of their sock and computer purchases, with K = 2. Give three answers, one for each of the variable scalings displayed. Explain.

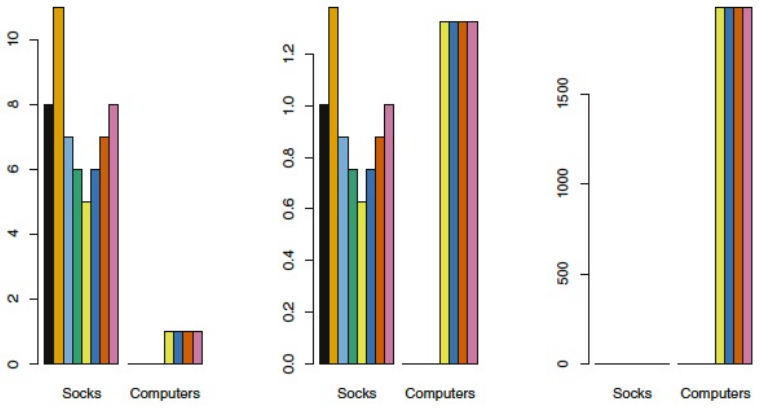


FIGURE 10.14. *An eclectic online retailer sells two items: socks and computers.*

In the left-hand plot, we are shown how many socks and computers each customer purchased. Since people bought much more socks and in different quantities than computers, then the main decision factor where to group each customer will largely be determined on the number of socks each person bought. We should expect the black, orange, red and purple customers to be grouped together in group A, and the cyan, green, yellow and blue customers to be grouped together in group B.
In the center plot, we are shown the scaled number of socks and computers purchased by each customer (they are scaled by their standard deviation). Now, whether someone bought a computer or not is largely the main factor determining where the customers are grouped. Here we expect the black, orange, cyan and green customers to be grouped together in group A, and the yellow, blue, red and purple customers to be grouped together in group B.
In the right plot, we are shown the value paid by each customers for the items purchased. Here the only factor determining how customers are

grouped is the value of the purchased computers. Here we expect the black, orange, cyan and green customers to be grouped together in group A, and the yellow, blue, red and purple customers to be grouped together in group B.

2 Applied Questions

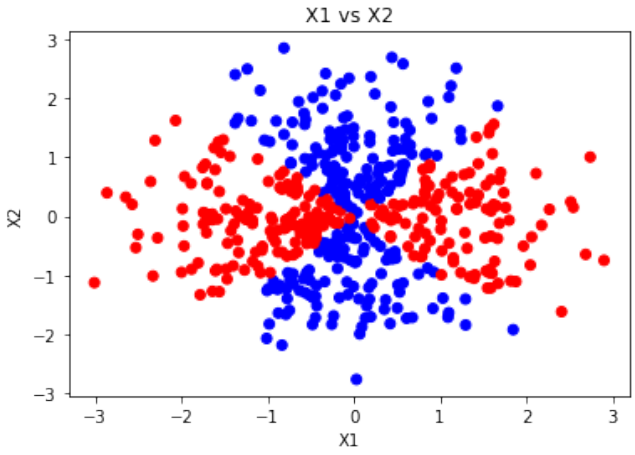
2.1 Question 1

We have seen that we can fit an SVM with a non-linear kernel in order to perform classification using a non-linear decision boundary.We will now see that we can also obtain a non-linear decision boundary by performing logistic regression using non-linear transformations of the features.

(a) Generate a data set with n = 500 and p = 2, such that the observations belong to two classes with a quadratic decision boundary between them. For instance, you can do this as follows:

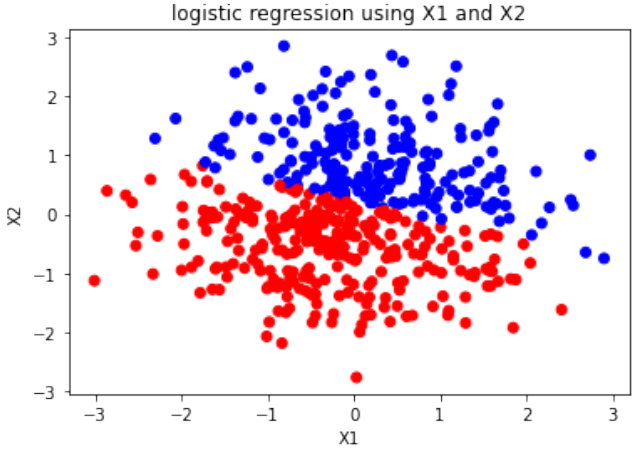
$X_1 = random.uniform(500) - 0.5$
https://numpy.org/doc/stable/reference/random/generated/numpy.random.uniform.html
$X_2 = random.uniform(500) - 0.5$
$y = 1 * (X_1^2 - X_2^2 > 0)$

(b) Plot the observations, colored according to their class labels. Your plot should display X_1 on the x-axis, and X_2 on the y-axis.



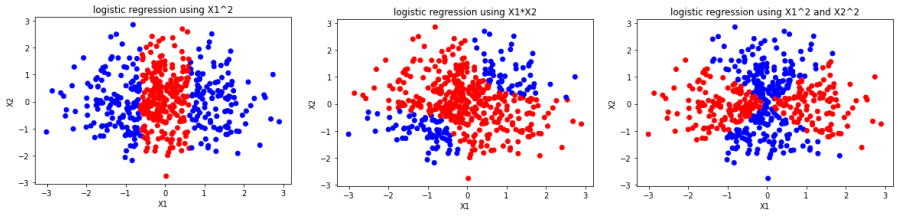
(c) Fit a logistic regression model to the data, using X_1 and X_2 as predictors.

(d) Apply this model to the training data in order to obtain a predicted class label for each training observation. Plot the observations, colored according to the predicted class labels. The decision boundary should be linear.



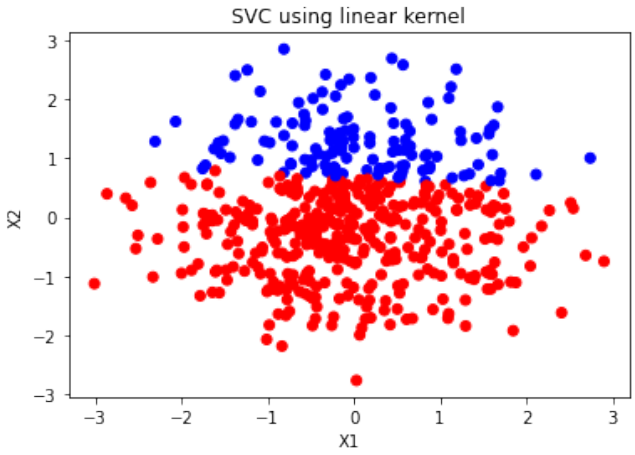
(e) Now fit a logistic regression model to the data using non-linear functions of X_1 and X_2 as predictors (e.g. X_1^2, $X_1 * X_2$, $\log(X_2)$, and so forth).

(f) Apply this model to the training data in order to obtain a predicted class label for each training observation. Plot the observations, colored according to the predicted class labels. The decision boundary should be obviously non-linear. If it is not, then repeat (a)-(e) until you come up with an example in which the predicted class labels are obviously non-linear.

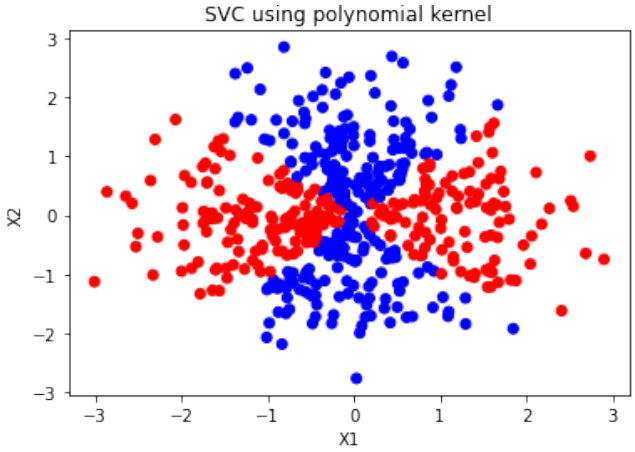


Left: Logistic Regression using X_1^2. Center: Logistic Regression using $X_1 * X_2$. Right: Logistic Regression using X_1^2 and X_2^2

(g) Fit a support vector classifier to the data with X_1 and X_2 as predictors. Obtain a class prediction for each training observation. Plot the observations, colored according to the predicted class labels.



(h) Fit a SVM using a non-linear kernel to the data. Obtain a class prediction for each training observation. Plot the observations, colored according to the predicted class labels.



(i) Comment on your results.
Logistic Regression using X_1 and X_2 has a very similar decision boundary as the one from SVC classifier using a linear kernel, main difference being the SVC classifier decision boundary is more horizontal. Both classifiers are way off when it comes to capturing the true classification.
Logistic Regression using X_1^2 and $X_1 * X_2$ produce non-linear decision boundaries, however they do not perform much better than the previous two classifiers, as their shapes do not correctly capture the shape of the true decision boundaries.
Logistic regression using X_1^2 and X_2^2 as predictors is very similar to the

SVC classifier using a polynomial kernel of degree 2. They perform extremely well, compared to the previous method, correctly capturing the shape of the true decision boundary. The main advantage of the polynomial SVC classifier, over the logistic classifier is ease of use. When using SVC using a polynomial kernel, one must only specify the degree of the polynomial, whereas for the logistic classifier one must provide the transformed predictors himself.

2.2 Question 2

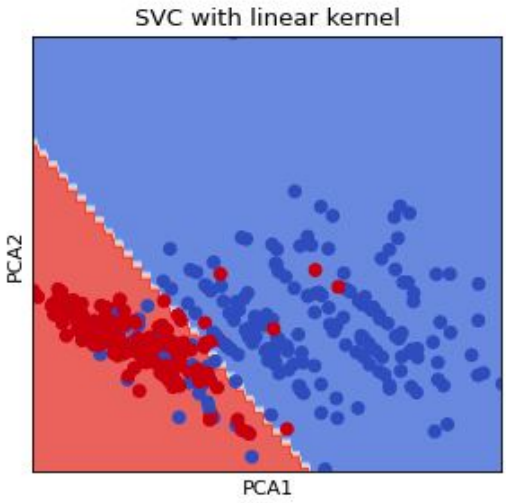
In this problem, you will use support vector approaches in order to predict whether a given car gets high or low gas mileage based on the Auto data set.

(a) Create a binary variable that takes on a 1 for cars with gas mileage above the median, and a 0 for cars with gas mileage below the median.

```
[10] import numpy as np
     import pandas as pd

     data = pd.read_csv("Auto.csv", na_values='?').dropna().reindex()
     data.reset_index(drop=True, inplace=True)
```

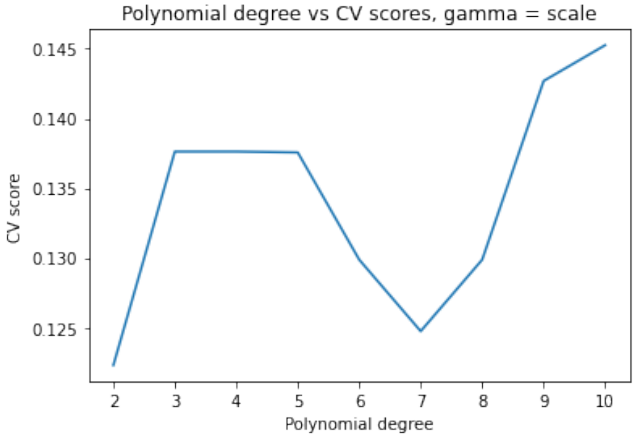
```
[11] medianMpg = np.median(data['mpg'])

     y = (data['mpg'] > medianMpg).astype(int)
     X = data.drop(['mpg', 'name'], axis=1)
```

(b) Fit a support vector classifier to the data with the linear kernel, in order to predict whether a car gets high or low gas mileage. Report the cross-validation error. Comment on your results. The cross-validation error rate was 0.11717948717948716. Here is plot of the data projected onto the first and second principal components:

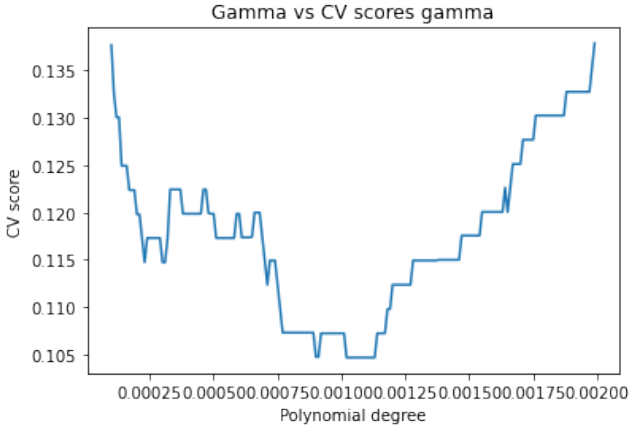


As we can see the decision boundary does quite a good job at separating the observations.

(c) Now repeat (b), this time using SVMs with radial and polynomial basis kernels, with different values of gamma and degree. Comment on your results.
Here is a plot of cross-validation error rates vs polynomial degree:

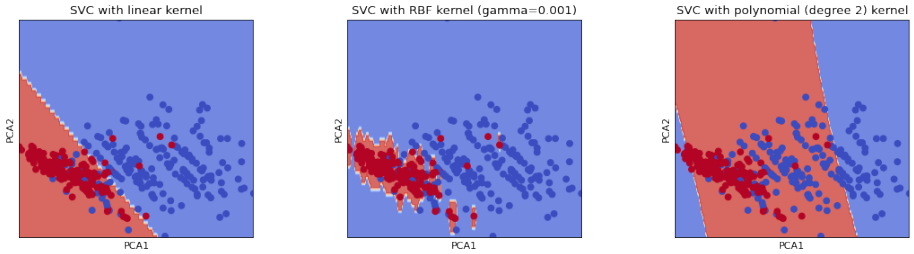


As we can see the best cross validation error rate is obtained when polynomial degree is 2. This implies that all other models are too flexible and are likely overfitting the data. However this error rate is still higher than that of the SVC classifier using the linear kernel.
Now let's look at the plot of cross-validation error rates vs gamma for the SVC classifier using a radial kernel:



Here we see the CV error rate achieve its lowest for a gamma of 0.001. Other values of gamma are either too flexible or too biased. The best CV score is even better than that of the linear model.

(d) Make some plots to back up your assertions in (b) and (c).
Here is plot of the data projected onto the first and second principal components, along with the decision boundaries from the 3 models:



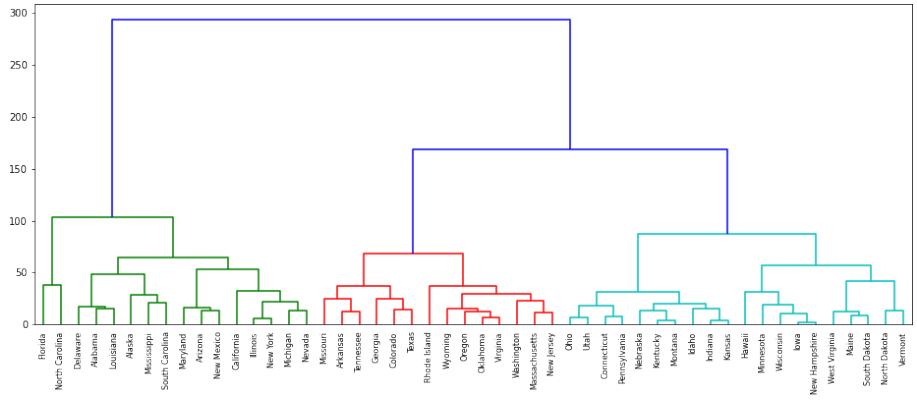
Here we see that the SVC classifier with a radial kernel (gamma=0.001) does the best job, followed closely by SVC classifier with the linear kernel.The SVC classifier with a degree 2 polynomial fares worst.

2.3 Question 3

Consider the USArrests data. We will now perform hierarchical clustering on the states.

(a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states.

(b) Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters?

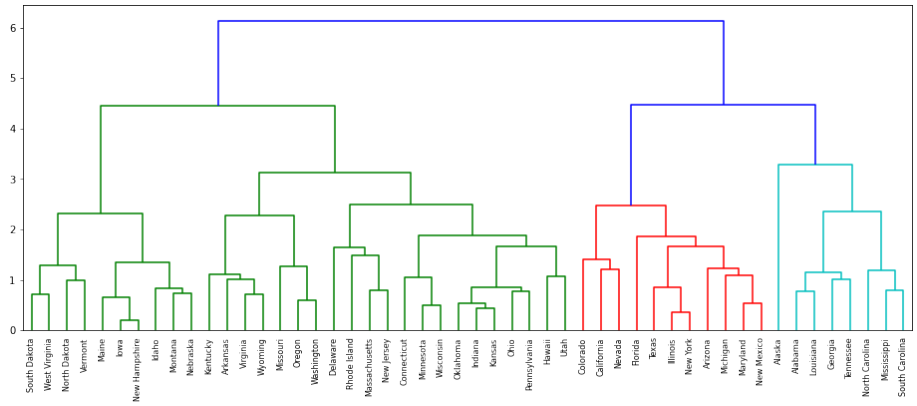


Green Cluster: Florida, North Carolina, Delaware, Alabama, Louisiana, Alaska, Mississippi, South Carolina, Maryland, Arizona, New Mexico, California, Illinois, New York, Michigan, Nevada

Red Cluster: Missouri, Arkansas, Tennessee, Georgia, Colorado, Texas, Rhode Island, Wyoming, Oregon, Oklahoma, Virginia, Washington, Massachusetts, New Jersey

Cyan Cluster: Ohio, Utah, Connecticut, Pennsylvania, Nebraska, Kentucky, Montana, Idaho, Indiana, Kansas, Hawaii, Minnesota, Wisconsin, Iowa, New Hampshire, West Virginia, Maine, South Dakota, North Dakota

(c) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one.



Green Cluster: South Dakota, West Virginia, North Dakota, Vermont, Maine, Iowa, New Hampshire, Idaho, Montana, Nebraska, Kentucky, Arkansas,

Virginia, Wyoming, Missouri, Oregon, Washington, Delaware, Rhode Island, Massachusetts, New Jersey, Connecticut, Minnesota, Wisconsin, Oklahoma, Indiana, Kansas, Ohio, Pennsylvania, Hawaii, Utah

Red Cluster: Colorado, California, Nevada, Florida, Texas, Illinois, New York, Arizona, Michigan, Maryland, New Mexico

Cyan Cluster: Alaska, Alabama, Louisiana, Georgia, Tennessee, North California, Mississippi, South Carolina

(d) What effect does scaling the variables have on the hierarchical clustering obtained? In your opinion, should the variables be scaled before the inter-observation dissimilarities are computed? Provide a justification for your answer. Scaling the variables so they have the same standard deviation, makes it so that all variables have roughly the same weight when computing the dissimilarity measure, which in this case is Euclidean distance.

Whether variables should be scaled depends on our data and what we are looking to achieve with our clustering algorithm. In the case of the USArrests data set, we probably would want to scale our data. Otherwise, since Assault occurs much more often than the other predictors, and it varies significantly from state to state, its number would therefore be the main deciding factor in our clustering algorithm. However, I would argue that our clustering algorithm should group together states with similarly high murder and rape incidents, as this would provide a better indication of the serious crime rate of a state.

2.4 Question 4

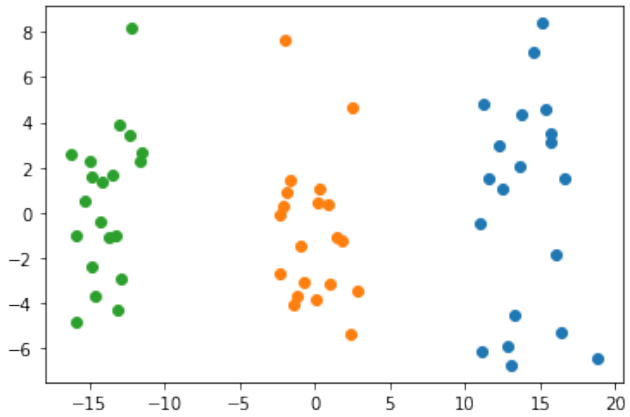
In this problem, you will generate simulated data, and then perform PCA and K-means clustering on the data.

(a) Generate a simulated data set with 20 observations in each of three classes (i.e. 60 observations total), and 50 variables. Use uniform or normal distributed samples.

```
import numpy as np
import pandas as pd

data = pd.DataFrame(columns=np.arange(50))

mu1 = -2
mu2 = 0
mu3 = 2
sigma = 2
n = 50

for i in range(0, 20):
  data = data.append(pd.DataFrame(np.random.normal(mu1, sigma, n)).T)

for i in range(0, 20):
  data = data.append(pd.DataFrame(np.random.normal(mu2, sigma, n)).T)

for i in range(0, 20):
  data = data.append(pd.DataFrame(np.random.normal(mu3, sigma, n)).T)

data.index = range(60)
```

Note: Inside the dataframe itself the first 20 observations belong to class 1, the next 20 to class 2 and the last 20 to class 3

(b) Perform PCA on the 60 observations and plot the first two principal component score vectors. Use a different color to indicate the observations in each of the three classes. If the three classes appear separated in this plot, then continue on to part (c). If not, then return to part (a) and modify the simulation so that there is greater separation between the three classes. Do not continue to part (c) until the three classes show at least some separation in the first two principal component score vectors. Hint: you can assign different means to different classes to create separate clusters.



(c) Perform K-means clustering of the observations with K = 3. How well do the clusters that you obtained in K-means clustering compare to the true class labels?

Here is the array of the predicted classes once the clustering was performed:

[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]

As we can see all class labels are correctly assigned.

(d) Perform K-means clustering with K = 2. Describe your results.

Here is the array of the predicted classes once the clustering was performed:

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

Here we see that observations belonging to class 2 and class 3 have been merged together into a single class. Observations belonging to class 1 were correctly labeled.

(e) Now perform K-means clustering with K = 4, and describe your results.

Here is the array of the predicted classes once the clustering was performed:

[1, 3, 1, 1, 1, 1, 3, 1, 3, 3, 3, 3, 3, 1, 1, 3, 1, 1, 1, 3,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Here we see that observations belonging to classes 2 and 3 were correctly labeled. However, observations belonging to class 1 were now fractured, as some observations were assigned a new fourth class label.

(f) Now perform K-means clustering with K = 3 on the first two principal component score vectors, rather than on the raw data. That is, perform K-means clustering on the 60 x 2 matrix of which the first column is the first principal component score vector, and the second column is the second principal component score vector. Comment on the results.

Here is the array of the predicted classes once the clustering was performed:

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

The results are the same as in part (c). The class labels were assigned perfectly.

(g) Using the z-score function to scale your variables, perform K-means clustering with K = 3 on the data after scaling each variable to have standard deviation one. How do these results compare to those obtained in (b)? Explain.

Here is the array of the predicted classes once the clustering was performed:

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

The results are same as in parts (c) and (f). I.e. labels are assigned perfectly. Because the observations were generated in such a way that all 50 variables for a single observation belonging to a single class, were generated from a normal distribution with same mean and variance, scaling them by their standard deviation, does not change the overall shape of the clusters, so we expect the k-means clustering algorithm to perform the same regardless whether the data was standardized or not.