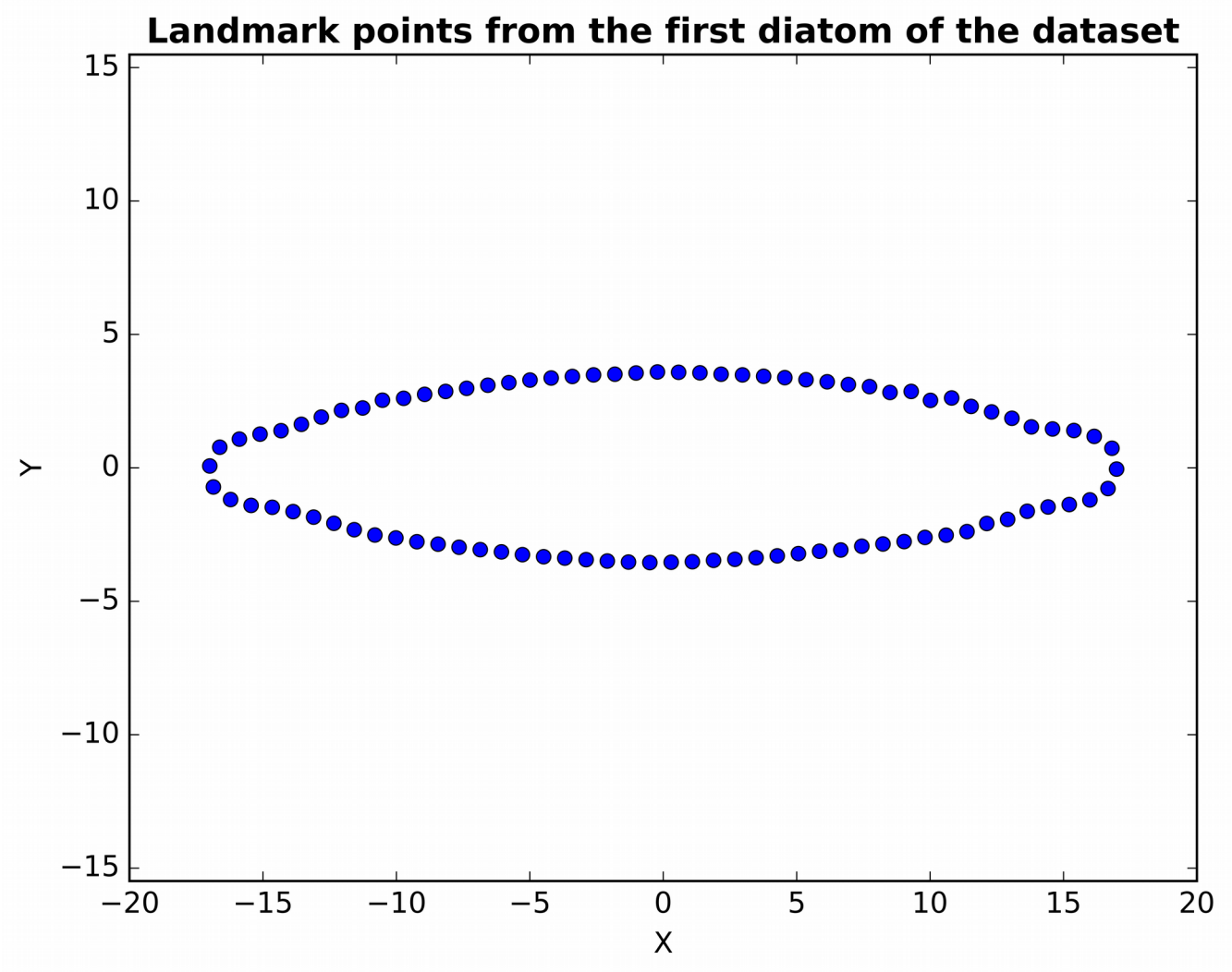
William Bullock

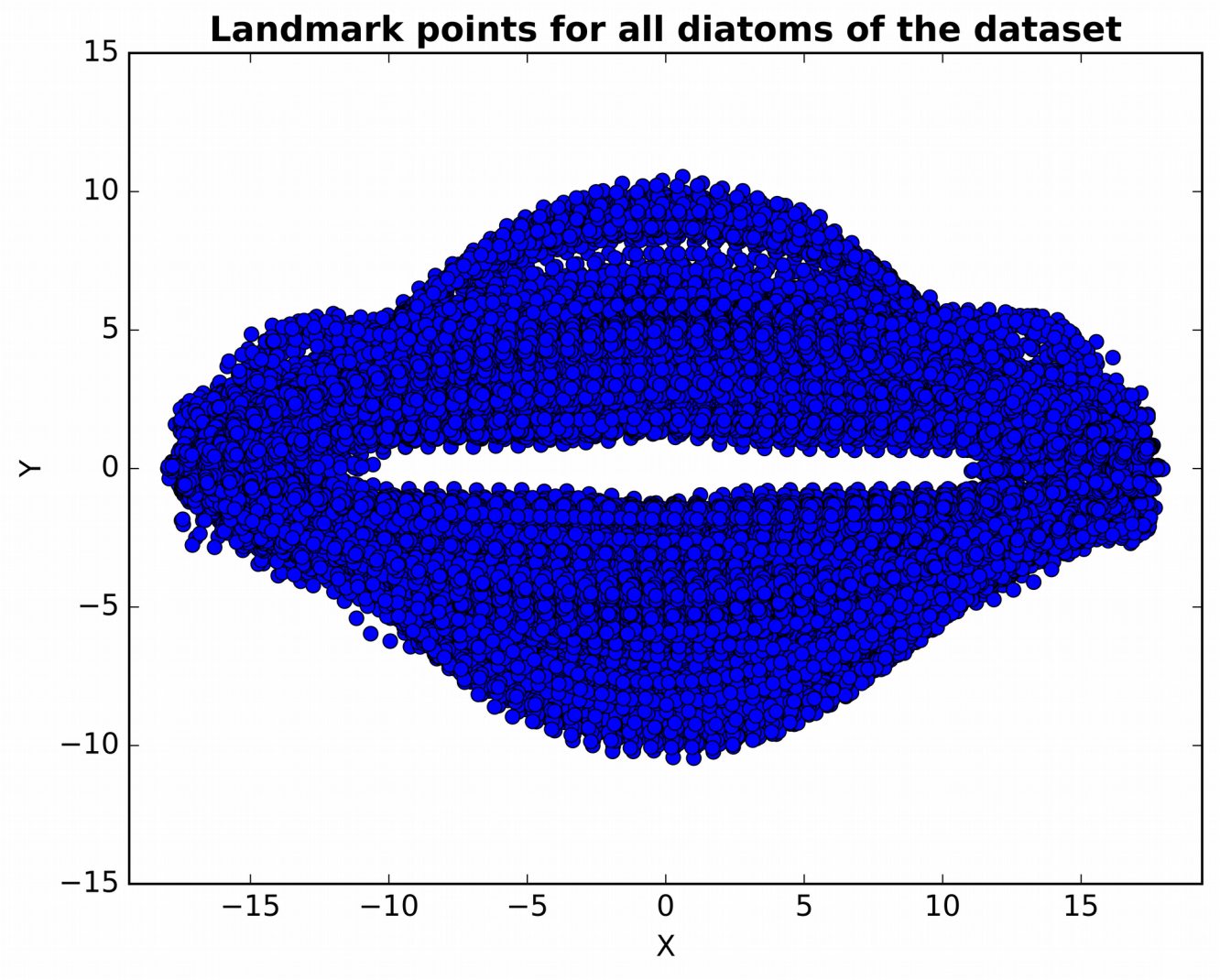
Introduction to Data Science

Assignment Four – Answer Sheet

(Please see companion .py file for the code used in this assignment)

Exercise 1:

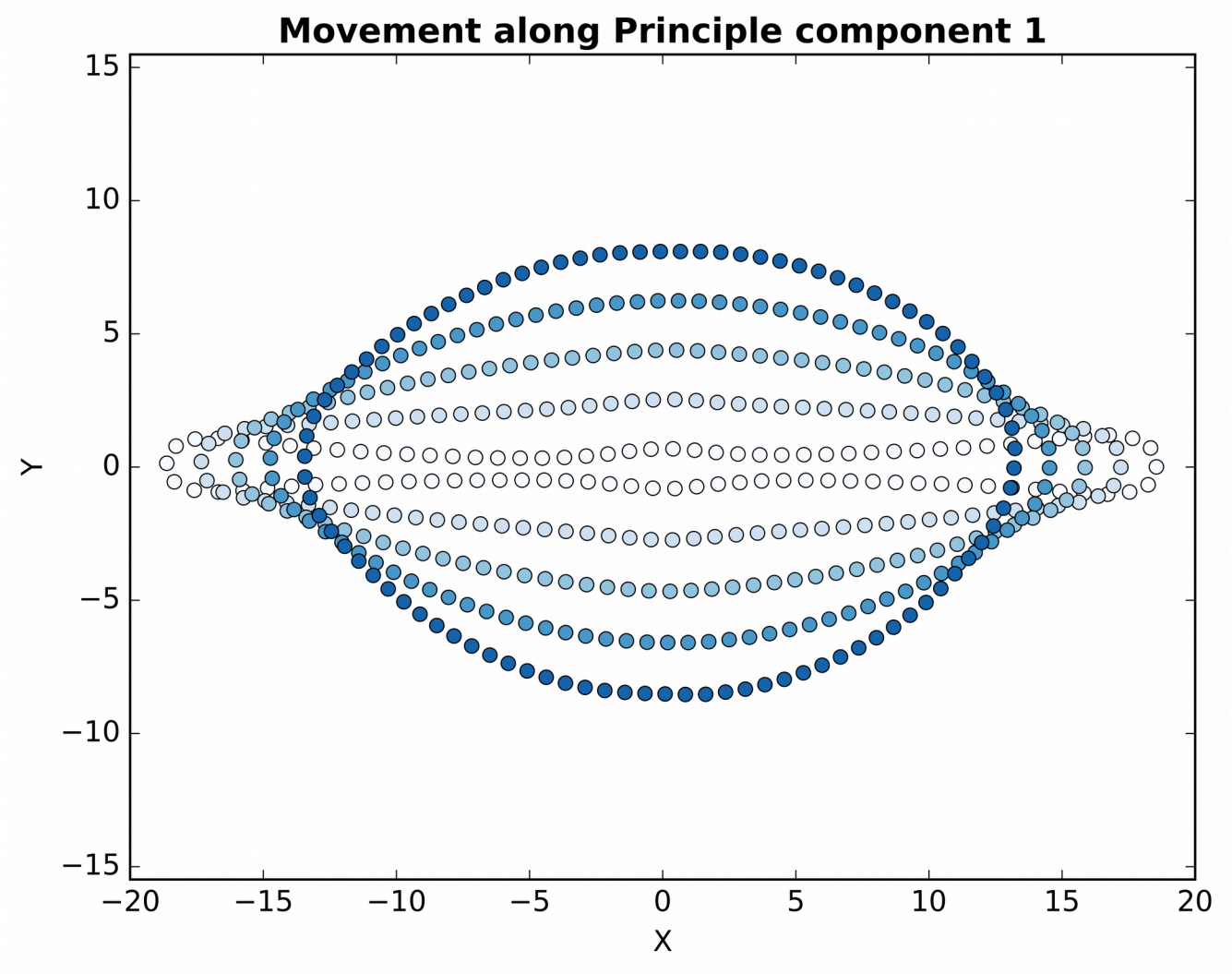


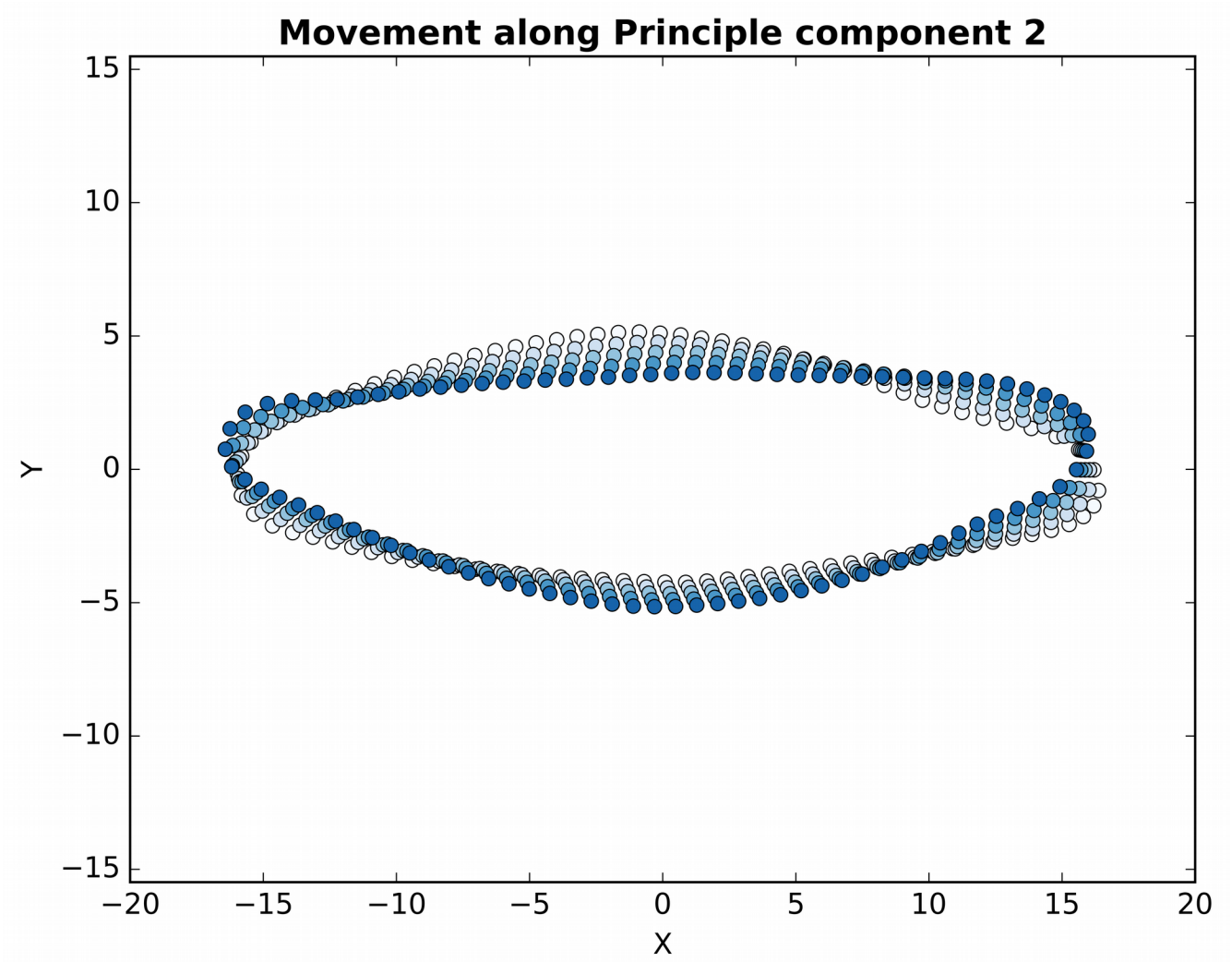


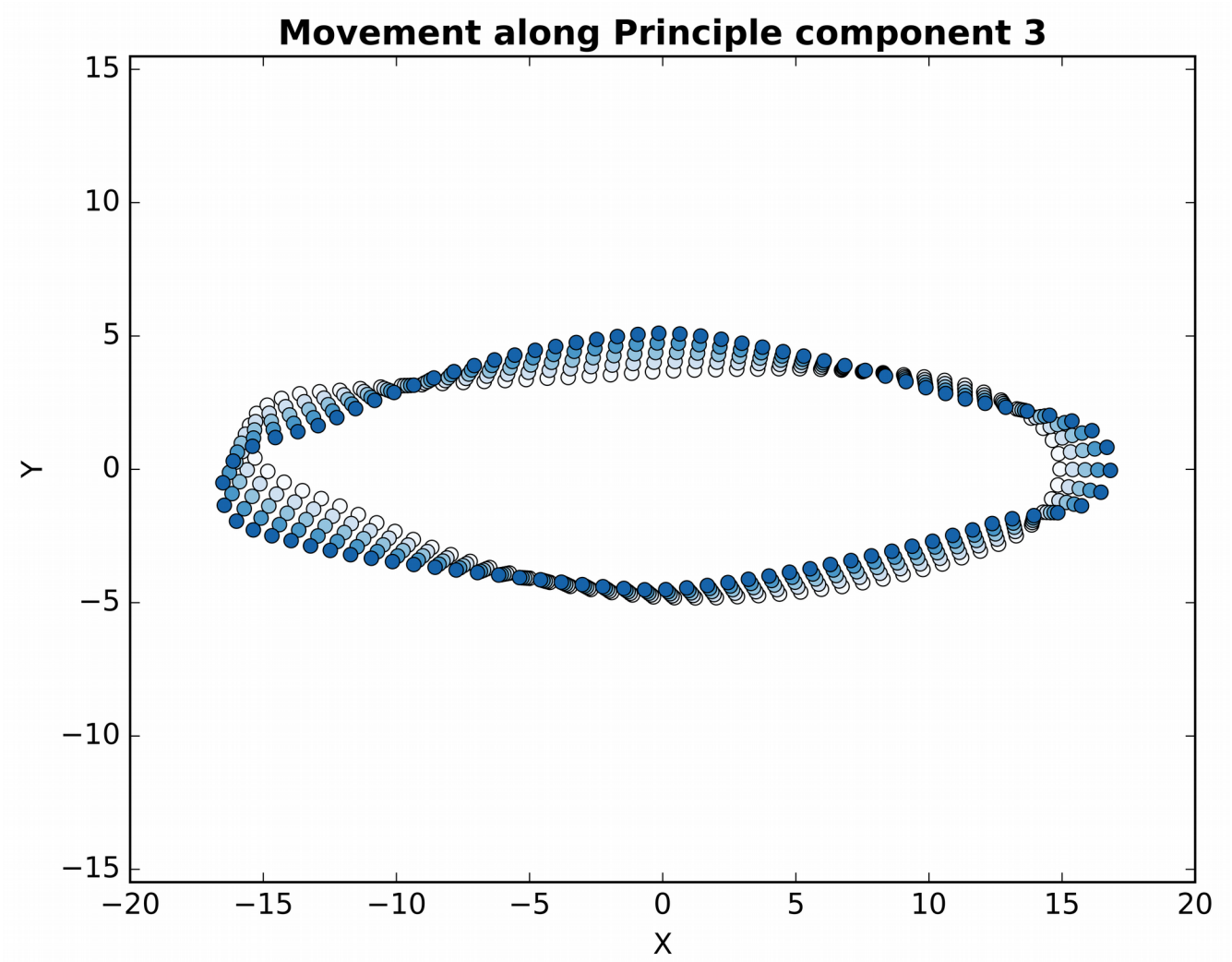
It seems that the greatest variance lies on the width of the cell, the Y – axis, the cells seem to be tapered at the end and have a central bulge, but the size of this bulge varies greatly.

The length of the cells (here, the X-axis) seem somewhat more consistent, but with some variety. Finally, the diagonals have the least variance, the cells tend to not be circular, but to remain convex, albeit to varying degrees.

Exercise 2:


Movement along Principle component 1
15
10
5
0
−5
−10
−15
Y
−20
−15
−10
−5
0
5
10
15
20
X


Movement along Principle component 2
15
10
5
0
−5
−10
−15
Y
−20
−15
−10
−5
0
5
10
15
20
X



The plot of the first principle component shows that, as previously speculated, the greatest variance is in the size (and potentially up/down orientation) of the central bulge in the diatom cell. But, from examining the colour gradient of the data points, we can see that ‘fatter’ cells ten to be shorter, and more oval (greater Y-axis range leads to a decrease in X-axis range), and the inverse is true for ‘slimmer’ diatom cells, which appear more cylindrical.

The plots of the second and third principle components are different to that of the first, but similar to each other. They seem to capture the variance in the ‘tilt’ of the cell, with some pertaining to being more ‘rhomboid’ or leaning, in one diagonal more than others.

It’s worth noting that the second and third plot complement each other, in that areas of low variance on plot 2 typically have grater variance on plot three and vice versa.

Exercise 3:

a)
i) Centering

Centering data is the process of subtracting the mean from each data point, it removes any bias in the inputs by translating the origin.

With regard to PCA centering removes intercepts and allows for all components to be drawn through the origin.

To compute a covariance matrix implies centering, thus, with a few rare exceptions (such as eigen-decomposition performed on the $\mathbf{X}\top\mathbf{X}/(n-1)$ matrix), centering is integral to the majority of PCA methods.

ii) Standardization

Standardization is the process of measuring how many standard deviations away from the mean each data point is in ecludian distance and taking this value in standard deviations to represent the data point.

Standardization allows each of the variables to be treated on the same scale. Without standardization, some variables would have potentially been deemed to contribute more than their fair share towards total variance, potentially due to larger numerical values or a greater range in one particular dimension.

This is useful in PCA to ensure the the variance captured in each PC is a true representation of the variability of the data.

Standardization is often, but not always useful, knowledge of and experience with the specific dataset being handled pays dividends when deciding whether or not to standardize.

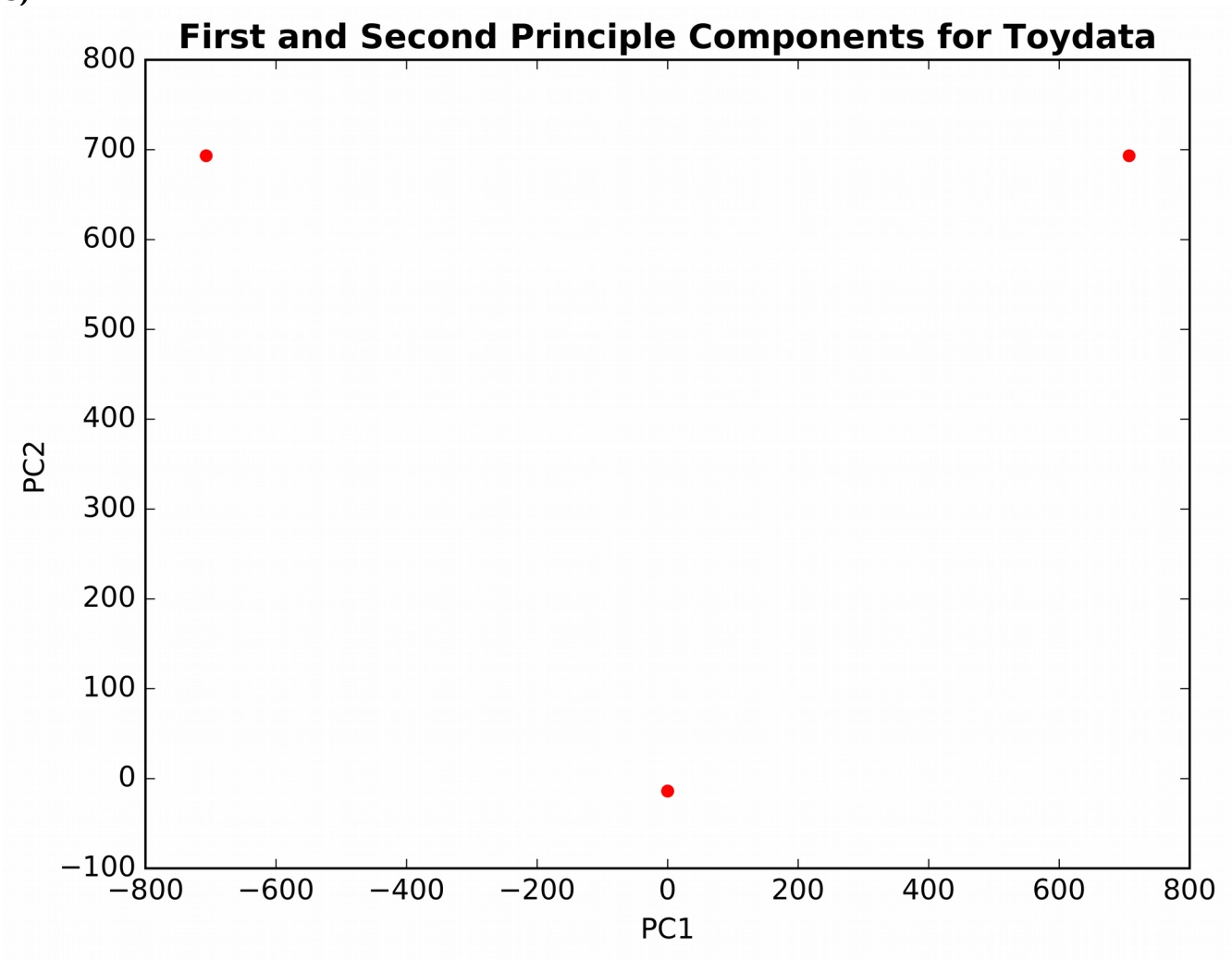
iii) Whitening

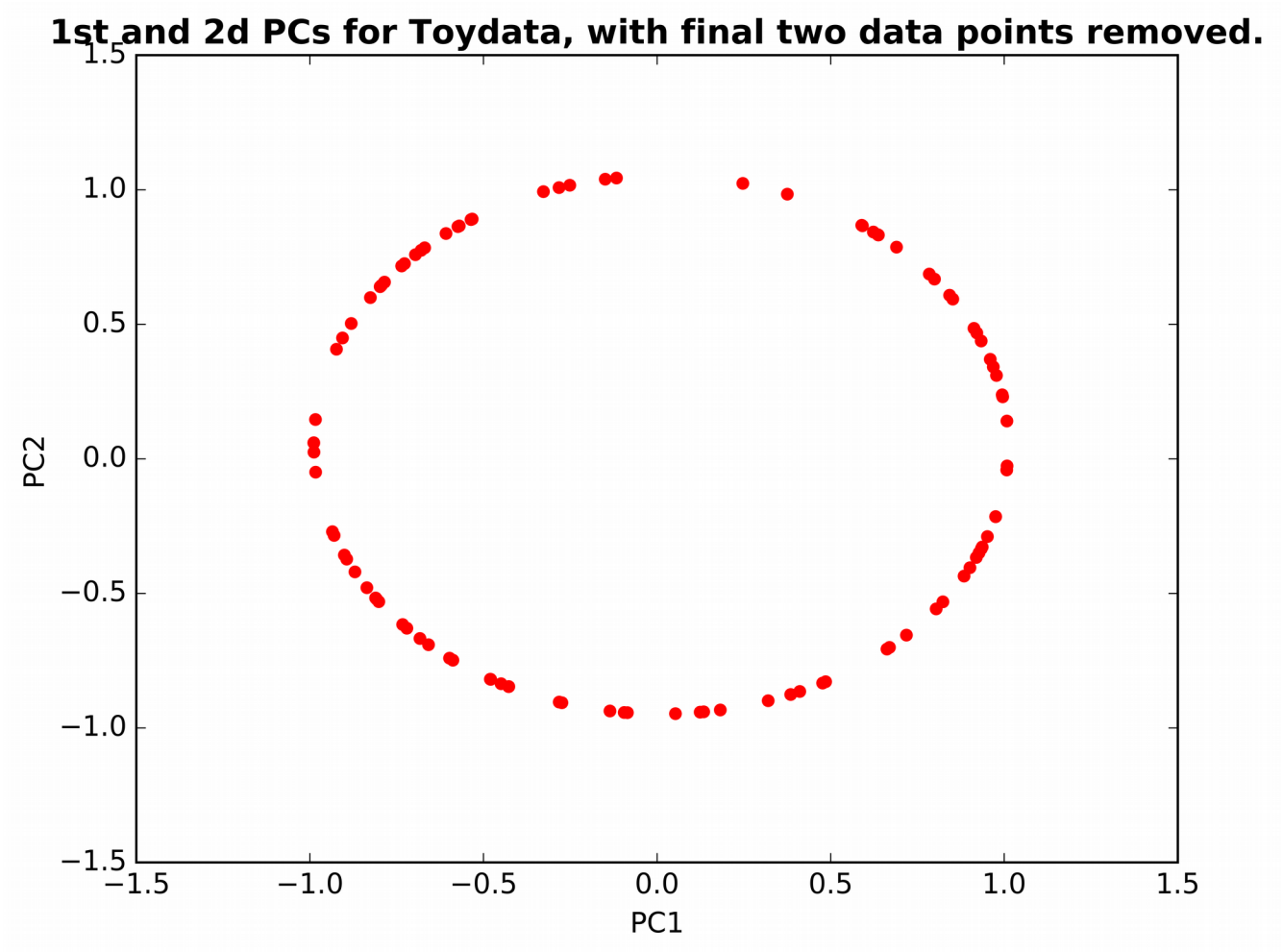
Whitening data refers to removing correlation between input variables, typically by transforming the data with a known covariance matrix that leaves all vectors with variance 1.

PCA is largely Dependant on variance differences between data points, and as such whitening data before performing PCA will cause useless results to be yielded.

Whitening is useful in other areas of machine learning, perhaps to avoid over-fitting algorithms to training data sets, but will ruin any data you wish to apply PCA to.

b)


First and Second Principle Components for Toydata
800
700
600
500
400
300
200
100
0
−100
PC2
−800
−600
−400
−200
0
200
400
600
800
PC1



The hidden structure is a circle.

The Final two data points have created a lot of false variance in by having no value (0) in columns one and two and by having exceedingly large values columns three and four.

In contrast the other data points in the data set have values close to 0, and at 0 respectively, for columns 3 and 4, and donot have 0s for columns one and two.

PCA has, in downscaling the data from 4D to 2D identified the huge numerical differences shown by the final two data points and chosen them alone to represent the principle component in the majority of variance in the data. Causing the first plot. When these data points are ignored, the true distribution of the majority of the data's variance is revealed.

This Illustrates the effect prominent outliers can have on PCA.

Exercise 4:

Although I've implemented both PCA and K-clustring myself before, they were imperfect and did not achieve full marks in the previous assignments.

As such, here, for both; I've used the scikit-learn functions.

PCA method

PCA for dimensionality reduction via covariance works; in summary, via the below process:

Data preprocessing → covariance matrix calculation → eigen-decomposition → projection

The data is almost always standardized (when calculating a covariance matrix), and usually normalized so as to insure all data points have an intercept at the origin, and that the scaling of each data point is consistent and that they contribute to the covariance matrix with equal weight.

The covariance matrix for the dataset is calculated, then a matrix of the eigenvectors and a matrix of eigenvalues which diagonalize the covariance matrix are computed. These two matrices are then sorted in order of decreasing eigenvalue.

The eigenvectors of the highest value N eigenvalues, where N is the number of principle components desired, are then used to draw straight lines through the origin, along which the majority of variance is captured.

The process of transforming the data set maximizes the variance in the original data that has been preserved, while minimizing the total squared reconstruction error, allowing dimensionality reduction.

The N principle components can be drawn to be N new axis, and the transformed data set can be projected into the N dimensional space.

K-Clustering method

K-clustering was used here to partition the pesticide training data into two distinct groups (hard clustering), based on the data-points similarity to each other (euclidean distance).

For the purposes of this assignment; Initialization has begun at the beginning two values of the dataset. Usually initialization would be done from values as far from each other as possible to speedup convergence, or points can be chosen at random.
After initialization points are chosen, these points are used as centroids, and the data points nearest to each are assigned to each centroid, forming a cluster. Ties are broken randomly, or by a per-defined deterministic rule.

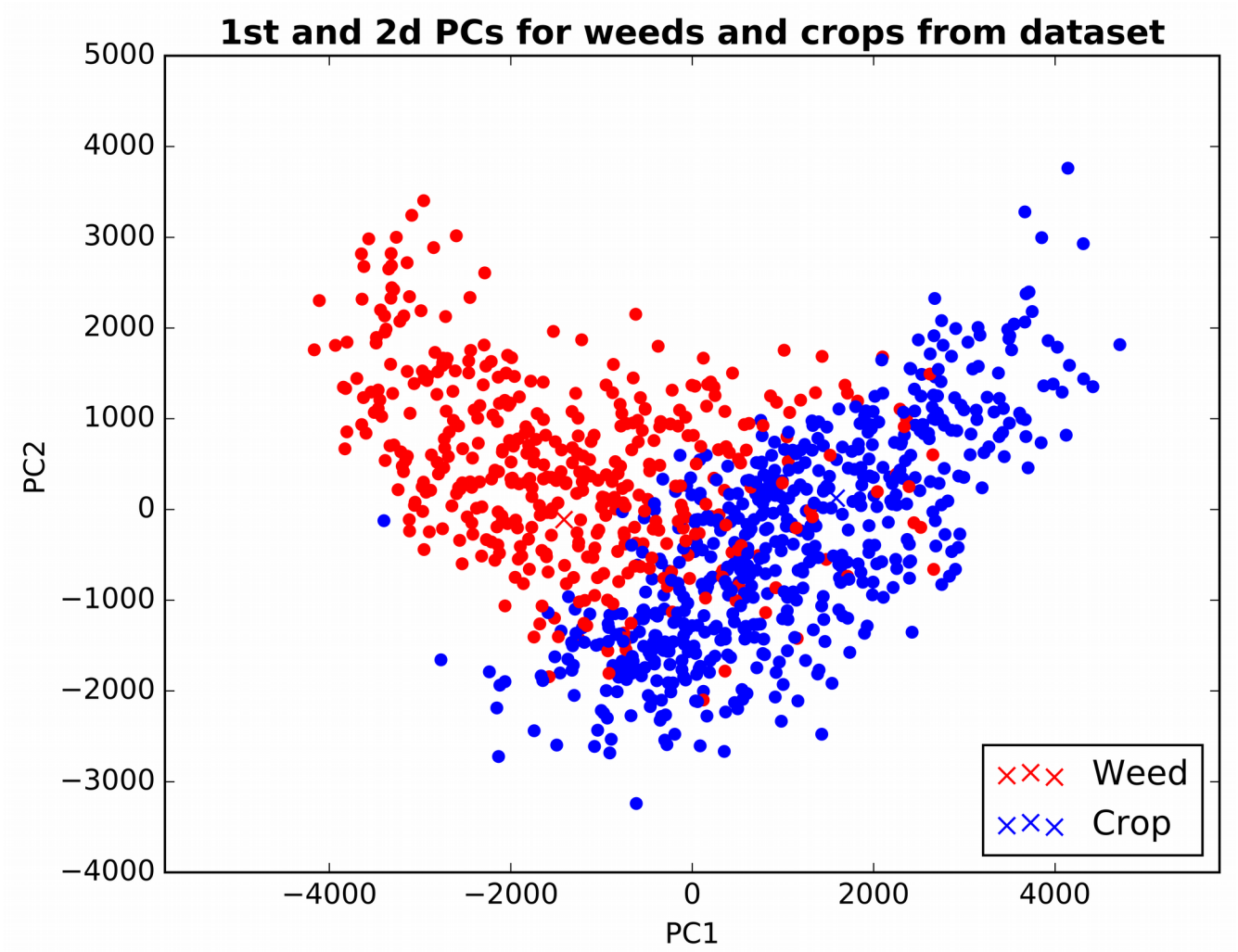
Once all the points are assigned, the centroid is recomputed, the mean of the data points in each cluster is calculated and are used as the new centroids.The clustering stage then repeats again.

This iterates until either; a per-defined maximum number of iterations is run, or the algorithm has reached a minimal value, whereby reassigning centroids would only increase distance between the

centroid and the surrounding data points, the parameter it is trying to minimize.

This method of K-means clustering is heuristic, like many others. And Re-running the algorithm, particularly with different initialization points is likely to give different results. For the purposes of the assignment, clustering has been performed only once and the points of initialization have remained constant.

The coordinates for the centroids of these clusters were then subjected to the same PCA protocol as aforementioned, please note that the PCA was **not re-fitted,** the fit found for the training data was used to transform the clusters and project them onto the 2-D PCs, whereby they were then added to the plot, and can bee seen as plus / 'x' marks.



In the above figure, crop patches are identified in blue, weed patches are red, and the K-clustering determined centroids are marked as plus / 'x' marks.

When coloured in this manner, there is a clear visual difference between between crop and weed data points.

However in terms of numerical value, there is a large overlap between the two classes of data points, representing a large potential for false-positives / false-negatives with regard to the drones being able to identify the class of vegetation visually.

It seems likely that a very precise algorithm or series of algorithms will be needed to allow the drones to reliably, successful, determine the class of the vegetation they are photographing.

Clustering seems to have worked well, the centroids appear well positioned within the points of hteir respective classes .

Perhaps a joint approach, or layered, tree-style investigation (neural networking?) between the methods explored in this assignment and those assignments previous, would be a difficult, but successful path to programming the drones to operate intended.