Matea Marinkovic
March-April 2022
Introduction to Data Science
Assignment 5

Assignment 5

Exercise 1

a) A Gaussian noise term is a noise where the probability density function is the same as the normal frequency distribution, given below:

normal distribution $= \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$

For the given equation with Gaussian noise, $\mu = 0$ and $x = \varepsilon$ therefore;

probability density function $(\varepsilon) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\varepsilon^2}{\sigma^2}\right)}$

b) The conditional probability distribution implies fixed inputs meaning probability $(y \mid a, x, b)$ The linear model given solved for ε can give us the probability as follows:

$y = ax + b + \varepsilon$

$\varepsilon = y - ax - b$

c) Parameters of the given linear model are:
- a = slope
- b = intercept; can be offset with noise
- ε = normally distributed noise or a random element determined

d) For a Bayesian treatment of the linear model 3 priors are needed, and Gaussian priors can be used for all three priors as the model is linear and can be solved for a normal distribution (positive or negative).

Exercise 2

W1 = Wine Acidity vs. All Features: [5.2057261 0.05035934]

Here we can see that the weight of the wine acidity (first feature) compared to the weights of all the features, is much higher, indicating that the acidity of the wine has a high impact on the overall quality score of red wine.

W2 = All Features vs. Wine Quality: [5.16573718e+01 1.95852727e-02 -1.06193618e+00 2.58896285e-02 5.02281634e-02 -2.75489463e+00 5.65346092e-03 -3.80728880e-03 -4.72092423e+01 -4.26639379e-01 8.50478130e-01 2.37895900e-01]

Here we can see with all the feature weights impacting quality, the first physiochemical property (fixed acidity) has the highest impact on quality, with chlorides, total sulfur dioxide and alcohol amount also having large impacts on the quality of the red wine. The other factors also have impact but are lower in weight than these.

Matea Marinkovic
March-April 2022
Introduction to Data Science
Assignment 5

Exercise 3

The RMSE function that I utilized is as follows:

```
def rmse(f, t):
    rmse = np.sqrt((1/len(f))*sum((f-t)**2))
    return rmse
```

The result of running the RMSE function over the acidity feature in the regression model is as following:

```
0.7860892754162224
```

The result of running the RMSE function over all the features in the regression model is as following:

```
0.6447172773160612
```

Comparatively, the RMSE of all the feature is lower than just the acidity feature, perhaps because the acidity feature has an overall larger impact on the wine than all the features combined.

Exercise 4

While normalization is a common preprocessing step that has many benefits including removing undesired biases due to differences in scaling, when implementing a random forest classifier, it is not necessary. The reason for this is that the random forest classifier is a tree-based decision algorithm that looks for the best points to split each feature. The split points are determined by percentages of labels correctly classifying a feature, which scaling does not impact.

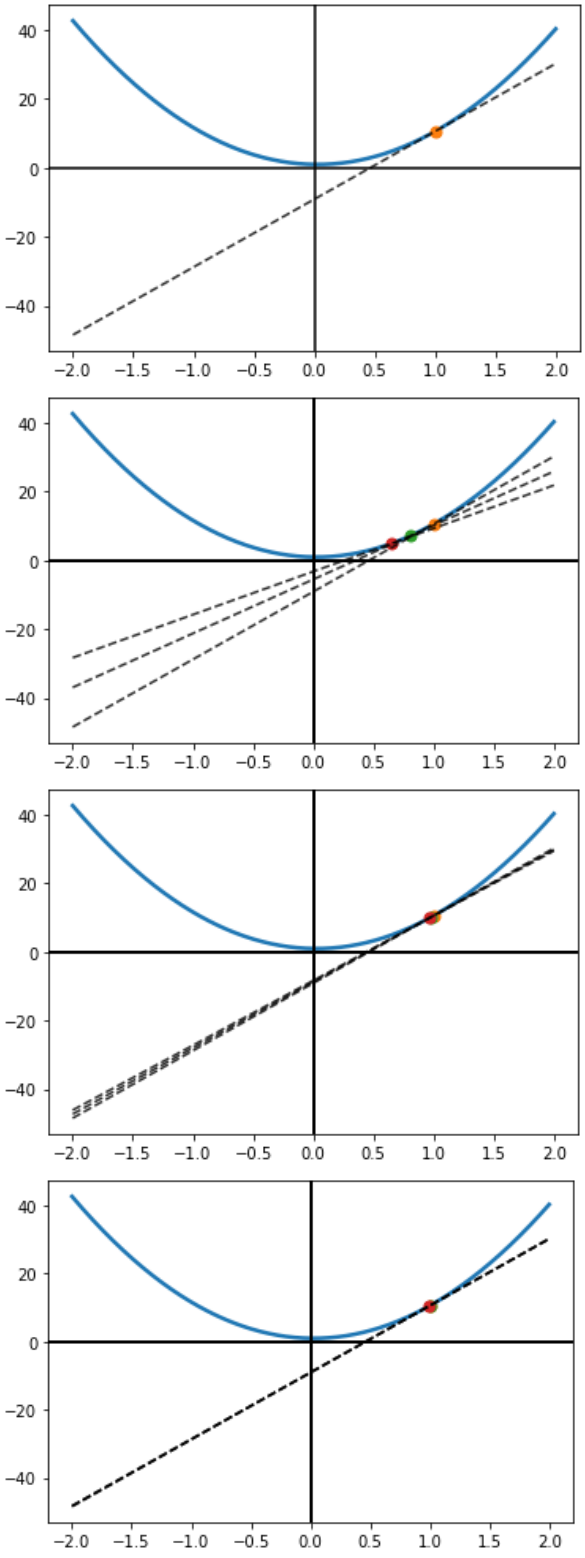
Exercise 5

```
Prediction Accuracy of the Random Forest 0.9668989547038328
```

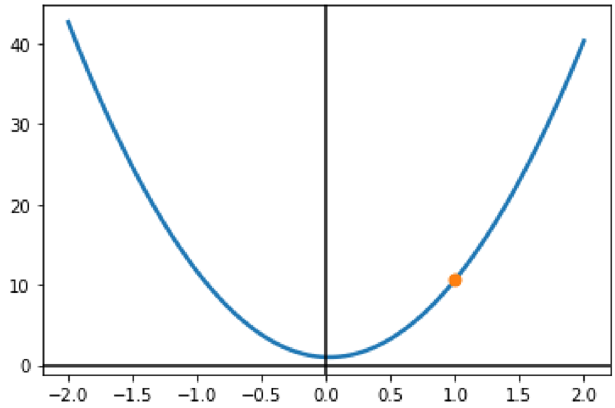
Exercise 6

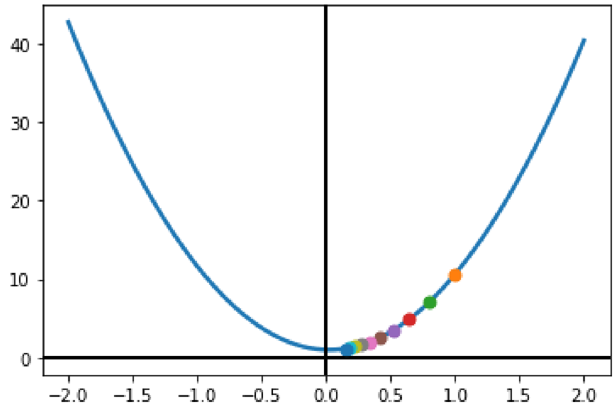
a. *Visualized Tangent Lines and Gradient descent steps for first three iterations at initial point x=1 (below):*

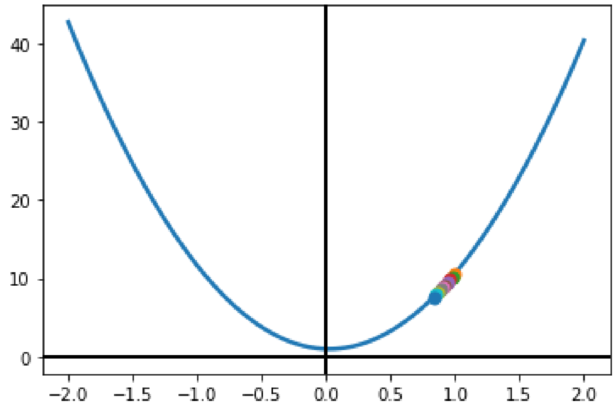
Matea Marinkovic
March-April 2022
Introduction to Data Science
Assignment 5

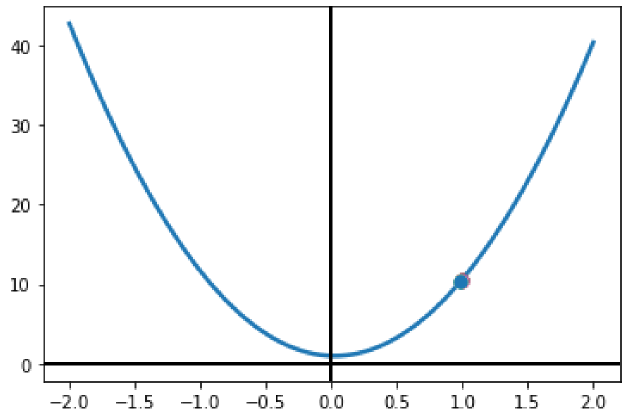


b. *Visualized Gradient descent steps for first 10 iterations (below):*


40
30
20
10
0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0


40
30
20
10
0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0


40
30
20
10
0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0


40
30
20
10
0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

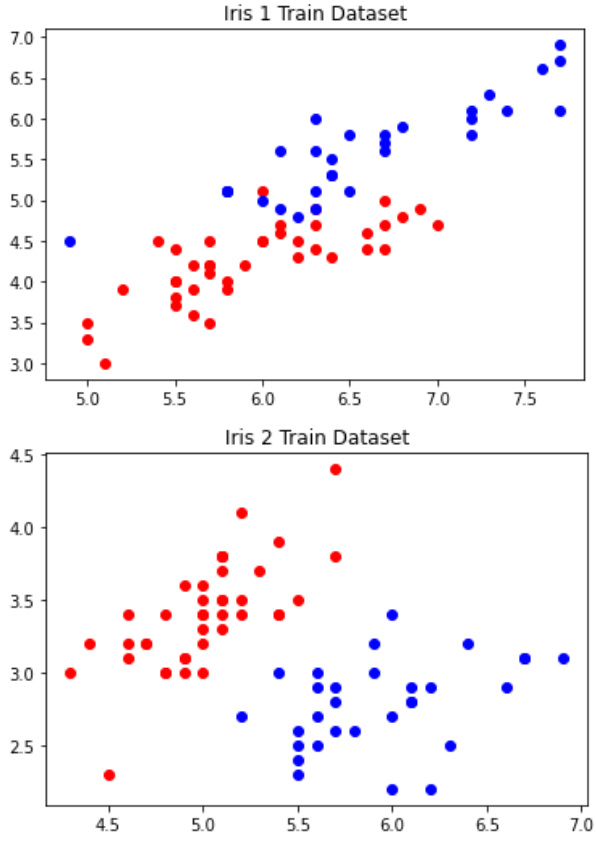
c.

For the last segment of the exercise, my algorithm would run for a while but then fail to produce plots and return errors because the size of the steps was too large. I did find the convergence as shown below:

Gradient descent has converged after {85} iterations. Gradient descent has converged after {822} iterations. Gradient descent has converged after {7154} iterations.

Exercise 7

Scatter plots from section 1 of this exercise:



Convergence Results:

The descent procedure did not converge after {1000} iterations, the last gradient norm was {0.11412853841372629} The descent procedure did not converge after {1000} iterations, the last gradient norm was {0.007051084114254621} The descent procedure did not converge after {1000} iterations, the last gradient norm was {0.02573558177479641} The descent procedure did not converge after {1000} iterations, the last gradient norm was {0.006968422503788121}

Binary Classification Results:

The three parameters of the affine linear model applied to the Iris 1 Train dataset: [-2.03331787 -3.29125363 4.5953671] The three parameters of the affine linear model applied to the Iris 1 Test dataset: [-1.60084343 6.91726373 -11.31268778] The three parameters of the affine linear model applied to the Iris 2 Train dataset: [-

2.44419742 4.98693176 -8.02544699] The three parameters of the affine linear model applied to the Iris 2 Test dataset: [-1.60766393 6.92473932 -11.31771734]

Error Rate Results:

Error rate in Iris 1 Train dataset: 0.33 Error rate in Iris 1 Test dataset: 0.17 Error rate in Iris 2 Train dataset: 0.29 Error rate in Iris 2 Test dataset: 0.07

Exercise 8

(1)

Taking the derivative of $E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N} \ln(1+e^{-y_n w^T x_n})$ with respect to w we get:

The gradient of the in sample as defined in the lecture: $\nabla E_{in}(w) = -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n x_n e^{-y_n w^T x_n}}{1+e^{-y_n w^T x_n}}$

$$= -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n x_n}{1+e^{y_n w^T x_n}}$$

$$= \frac{1}{N}\sum_{n=1}^{N} -y_n x_n \theta(-y_n w^T x_n)$$

(2)

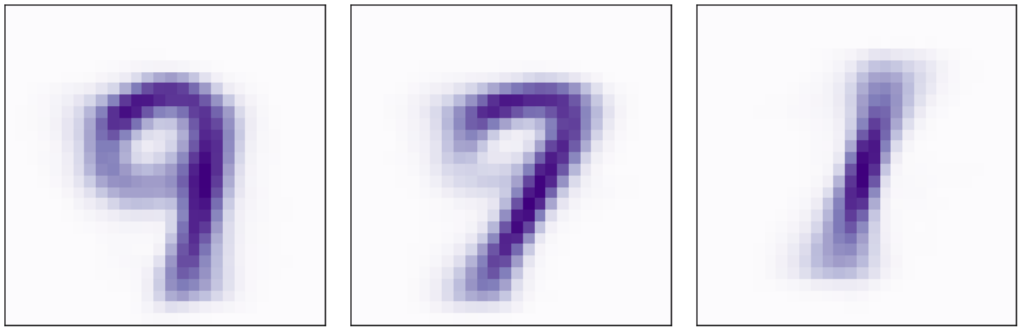
When samples are misclassified: $y_n w^T x_n < 0$, so $\theta(-y_n w^T x_n) > 0.5$

When samples are correctly classified: $\theta(-y_n w^T x_n) < 0.5$

The contribution of a misclassified sample therefor contributes more to a gradient than a classified sample.

Exercise 9

Three Cluster Centers



Percentage of each digit per cluster: There are 0.872 % 1s in cluster 0 There are 33.14 % 7s in cluster 0 There are 65.988 % 9s in cluster 0 There are 0.272 % 1s in cluster 1 There are 62.228 % 7s in cluster 1 There are 37.5 % 9s in cluster 1 There are 89.831 % 1s in cluster 2 There are 7.748 % 7s in cluster 2 There are 2.421 % 9s in cluster 2

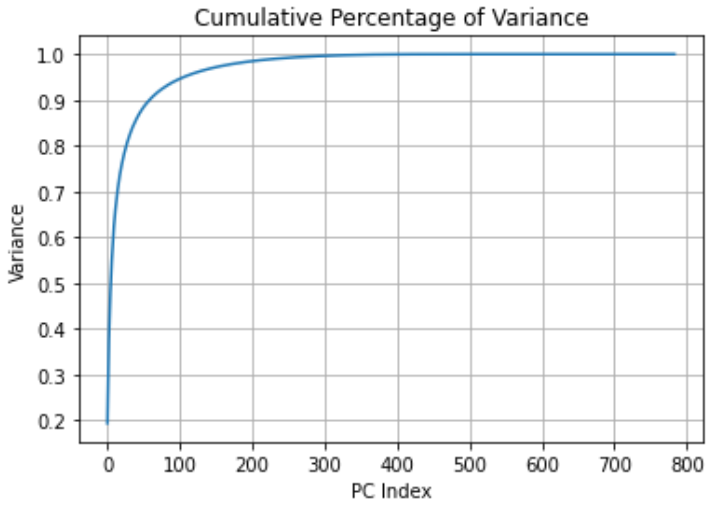
Description of software: Clustering is essentially identifying subgroups within the data where the datapoints within the group or "cluster" are very similar while the points in other groups are different. Kmeans clustering is a learning algorithm which separates each data point into only one cluster where the mean of the cluster is as low as possible. This is used to determine the first two clusters in this exercise. The PCA essentially summarizes data by means of individual principal components, which in this exercise only utilizes the first two PCs. Projecting is then used to reduce the dimensionality of both the clusters and the dataset with the most minimal loss of data to the dataset.

Results: In the images of the clusters, you can see that the first cluster has mostly 9s and is also shaped like a 9, while the second has mostly 7s and is shaped like a 7, and the third has mostly 1s and is shaped like a 1.

Test Accuracy for K Best: K best Value = 1 Testing Accuracy: 0.9733333333333334

Exercise 10

a.



Description of the software: For this exercise, I used a PCA function over the digits data. A PCA or principal component analysis, is a statistical procedure that can summarize large amounts of data into smaller summary indices (principle components) that are easier to visualize and analyze. PCA analysis is considered a form of multivariate data analysis based on projection methods.

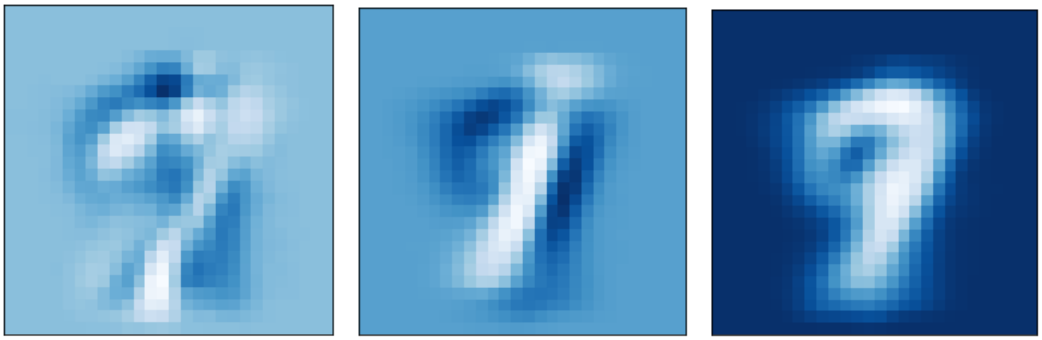
W.r.t. components: [0.19293803 0.2896408 0.37407791 0.43370813 0.47634636 0.51486835 0.54855152 0.57653198 0.60399775 0.62470521]

Matea Marinkovic
March-April 2022
Introduction to Data Science
Assignment 5

Description: You can see from the graph that as PC index is increasing the variance is also increasing until around 300 where it reaches 1.0 and levels out.

b.

Three Cluster Centers with 20 Components



Percentage of each digit per cluster: There are 22.222 % 1s in cluster 0 There are 50.0 % 7s in cluster 0 There are 27.778 % 9s in cluster 0 There are 100.0 % 1s in cluster 1 There are 0.0 % 7s in cluster 1 There are 0.0 % 9s in cluster 1 There are 0.0 % 1s in cluster 2 There are 100.0 % 7s in cluster 2 There are 0.0 % 9s in cluster 2

Results: Similar to exercise 9, the larger percentage of either 7s,1s or 9s is shown in each cluster image. However, in this exercise, the majority is not so high as in exercise 9, leading to a more distorted cluster image influenced by the high presence of the other numbers.

c.

Test accuracies for 20 components and 200 components respectively: Testing Accuracy: 0.5 Testing Accuracy: 0.425

Comparison of performances in 9b and 10c: The test accuracy in 9b is much greater than in 10c, perhaps because of the greater proportion of one digit or another within the cluster influencing the best value decision.