Mehmet Ali Pelit
21113622

Question 2: Naive Bayes

Q2.1

Weather	Temperature	Play Tennis
Sunny	Hot	No
Sunny	Mild	No
Overcast	Cool	Yes
Rainy	Cool	Yes
Rainy	Mild	Yes
Rainy	Cool	No
Overcast	Mild	Yes
Sunny	Cool	No
Sunny	Cool	Yes
Rainy	Mild	No

$$P(\text{Plays}) = \frac{5}{10} = 0.5$$
$$P(\text{Doesn't Play}) = \frac{5}{10} = 0.5$$

- For **Plays**:

$$P(\text{Sunny} \mid \text{Plays}) = \frac{1}{5} = 0.2$$
$$P(\text{Cool} \mid \text{Plays}) = \frac{3}{5} = 0.6$$

- For **Doesn't Play**:

$$P(\text{Sunny} \mid \text{Doesn't Play}) = \frac{3}{5} = 0.6$$
$$P(\text{Cool} \mid \text{Doesn't Play}) = \frac{2}{5} = 0.4$$

Naive Bayes Formula

$$P(\text{Plays} \mid \text{Sunny}, \text{Cool}) \propto P(\text{Sunny} \mid \text{Plays}) \times P(\text{Cool} \mid \text{Plays}) \times P(\text{Plays})$$
$$P(\text{Doesn't Play} \mid \text{Sunny}, \text{Cool}) \propto P(\text{Sunny} \mid \text{Doesn't Play}) \times P(\text{Cool} \mid \text{Doesn't Play}) \times P(\text{Doesn't Play})$$

Substituting the values:

$$P(\text{Plays} \mid \text{Sunny}, \text{Cool}) \propto 0.2 \times 0.6 \times 0.5 = 0.06$$
$$P(\text{Doesn't Play} \mid \text{Sunny}, \text{Cool}) \propto 0.6 \times 0.4 \times 0.5 = 0.12$$

Normalizing the probabilities:

$$\text{Total} = 0.06 + 0.12 = 0.18$$
$$P(\text{Plays} \mid \text{Sunny}, \text{Cool}) = \frac{0.06}{0.18} \approx 0.33$$
$$P(\text{Doesn't Play} \mid \text{Sunny}, \text{Cool}) = \frac{0.12}{0.18} \approx 0.67$$

Based on the Naive Bayes classifier, the probability of playing tennis given the conditions 'Sunny' and 'Cool' is approximately 33%, while the probability of not playing tennis is approximately 67%. Therefore, the model predicts that it is more likely not to play tennis under these conditions.

Q2.2

- **Step 2: Which Naive Bayes Classifier did you choose and why?**
  Since the features are continuous, Gaussian Naive Bayes is appropriate.

- **Step 7: Accuracy Scores:**

```
Accuracy without scaling: 0.9814814814814815
Accuracy with StandardScaler: 0.9814814814814815
Accuracy with MinMaxScaler: 0.9814814814814815
Accuracy without scaling (large magnesium): 0.46296296296296297
Accuracy with StandardScaler (large magnesium): 0.9814814814814815
Accuracy with MinMaxScaler (large magnesium): 0.9814814814814815
```

- **Step 8:**
  How do the accuracy scores compare with each other?
  All accuracy scores (without scaling, with StandardScaler, and with MinMaxScaler) are identical and very high (98.15%). After large magnesium scaling, without scaling accuracy drops significantly to 46.30% and it returns to 98.15% after scaling i.e. restores performance.

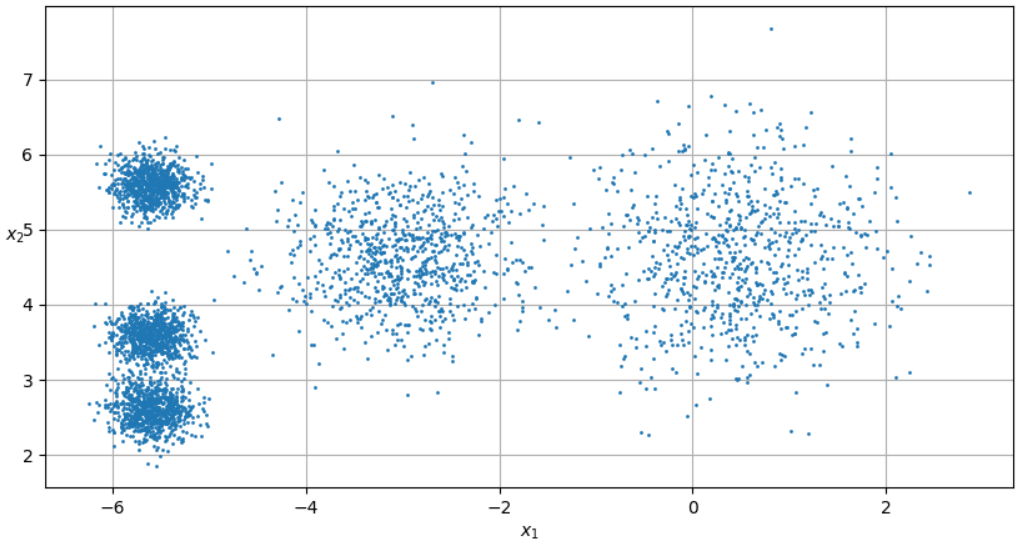
  It is generally said that the Naive Bayes Classifier is independent of feature scaling, why is it said so?
  Because it relies on the assumption that features are normally distributed and calculates probabilities based on feature distributions.

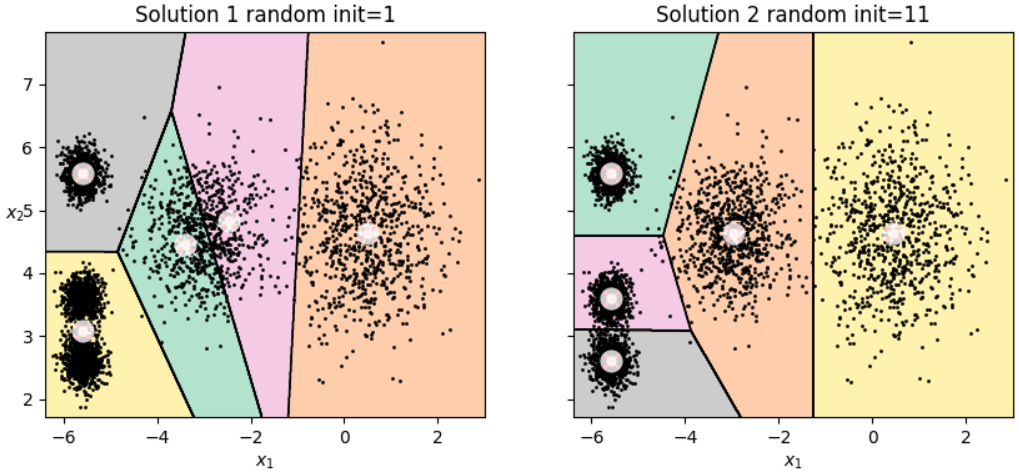
  As per our test, can we say that the Naive Bayes classifier is independent of feature scaling?
  Partially, yes. The theory works on the non-extreme feature values. A significant drop occurred in large magnesium case because the extreme value of the 'magnesium' feature created numerical instability.

Mehmet Ali Pelit
21113622

Question 3: K-means Limitations

Q3.1





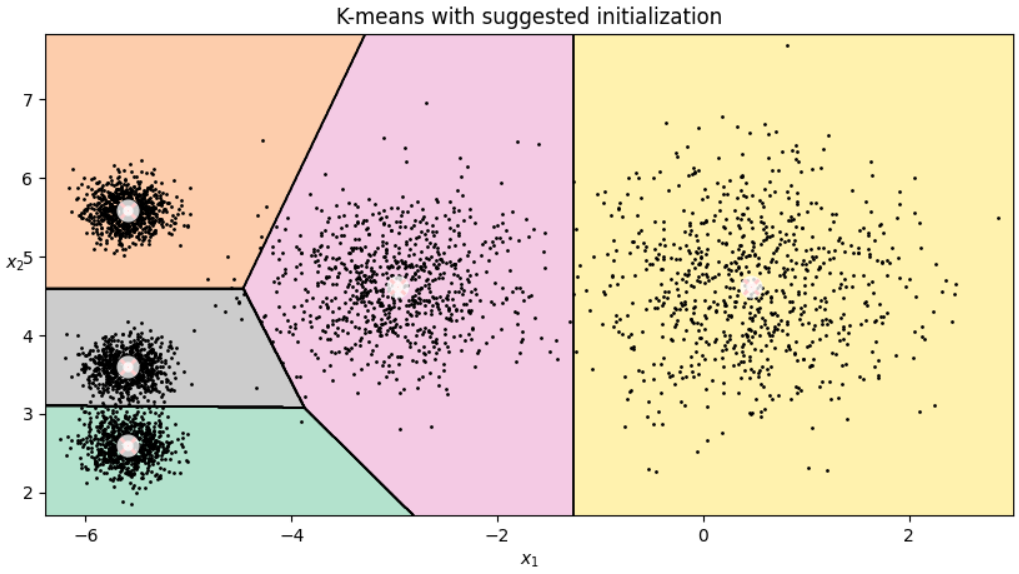
When looking at the plots, you can see that the clusters identified by K-means with random_state=1 and random_state=11 are different. This variation occurs because K-means converges to a local optimum, and the result is highly dependent on the initial positions of the centroids. Since the centroids are chosen randomly, different random states produce different initial centroids, leading to varying clustering outcomes.

Mehmet Ali Pelit
21113622

Q3.2

We can choose a set of points that are likely to be near the center of each cluster. This initialization is good because it is very close to the actual centers of the clusters in the data, and hence K-means should converge to a better solution.

Q3.3



By using a good set of initial centroids, the K-means algorithm converges to a solution that better represents the underlying data structure. The clusters are more accurate, and the centroids are closer to the actual centers of the blobs.

Q3.4

- Setting n_init to a higher value. K-means will run several times with different initial centroids and choose the best solution based on the within-cluster sum of squares.
  kmeans_multiple_inits = KMeans(n_clusters=5, n_init=10, random_state=11)
- K-means++ initializes the centroids in a smarter way that spreads them out more evenly.
  kmeans_plus_plus = KMeans(n_clusters=5, init='k-means++', n_init=1, random_state=11)

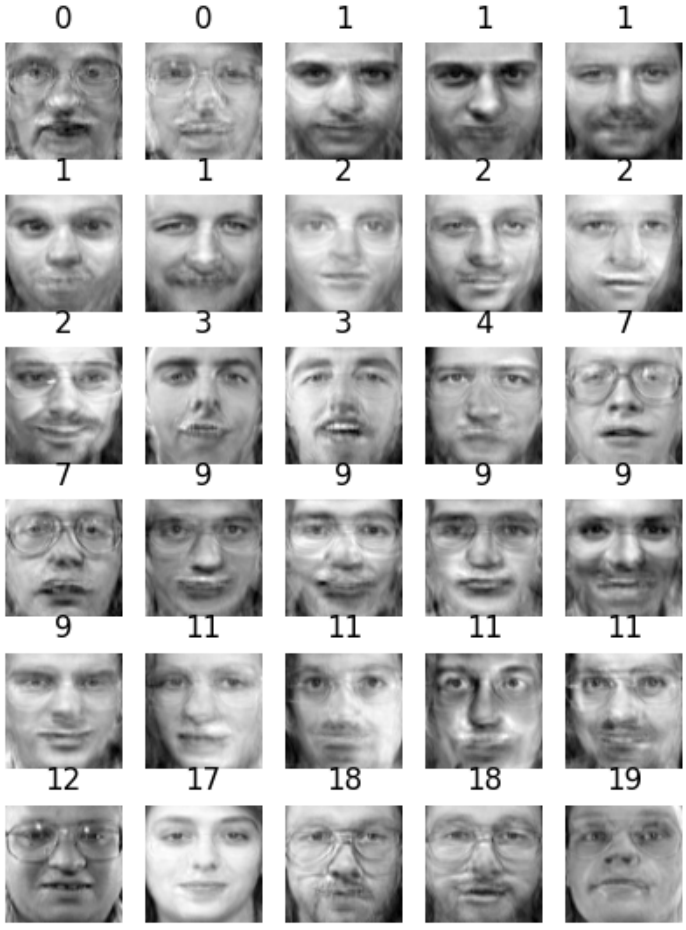
Mehmet Ali Pelit
21113622

Question 4: Gaussian Mixture Models for Generating New Faces

Q4.1

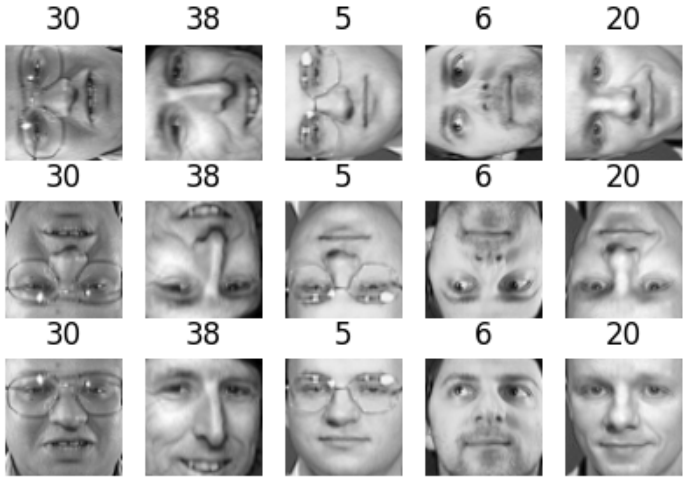
```
Reduced Dataset shapes:
(280, 137) (280,)
(80, 137) (80,)
(40, 137) (40,)
```

Q4.2



Mehmet Ali Pelit
21113622

Q4.3



Q4.4

```
Log-likelihood of modified samples:
[-13881459.47457401 -23701012.90010313 -20364379.40791821
 -20006304.79714391 -18213634.97993135 -18085209.70774318
 -10672924.36230035 -20558079.60849436 -23962248.00642896
 -16591721.4645312  -46284188.83977317 -61680685.81676133
 -33715051.59248367 -39362276.81499262 -44722475.32319631]

Log-likelihood of original samples:
[791.96194346 652.31781261 763.27663058 582.2466327  718.94169442
 568.49505638 568.49506171 582.24668345 568.49506666 652.69014477
 582.24671399 718.94169458 698.13521316 783.37681483 568.49521636]
```

The log-likelihood values represent how likely each sample is under the trained Gaussian Mixture Model. Lower log-likelihood values indicate samples that are less likely under the model, which can be considered as anomalies. We expect the modified images (rotated, flipped, darkened) to have lower log-likelihoods compared to the original images. This is because the modifications introduce features that are not well-represented by the model trained on the original dataset.

Gaussian Mixture Models can be used for anomaly detection by identifying samples with significantly lower log-likelihoods compared to the typical range observed in the training data. Such samples can be flagged as potential anomalies for further investigation.

Mehmet Ali Pelit
21113622

Question 6: Denoising Autoencoders

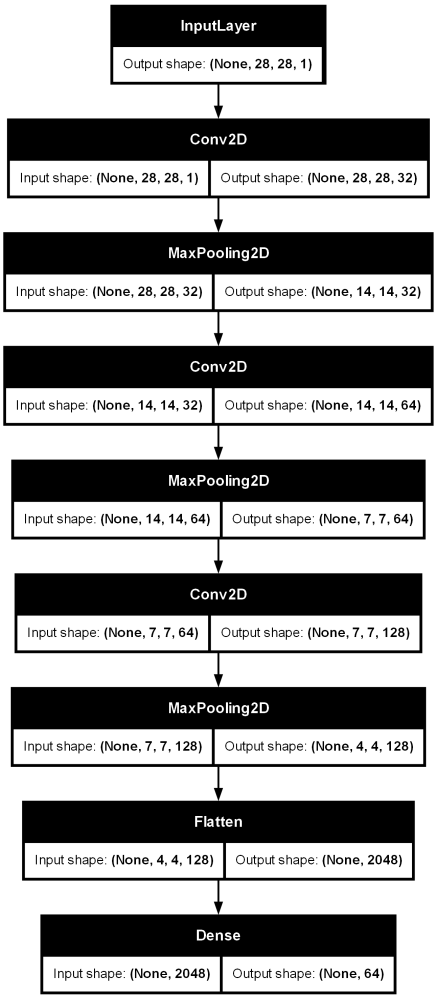
Q6.1

Build the Encoder

Model: "encoder"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 28, 28, 1)	0
conv2d (Conv2D)	(None, 28, 28, 32)	320
max_pooling2d (MaxPooling2D)	(None, 14, 14, 32)	0
conv2d_1 (Conv2D)	(None, 14, 14, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 7, 7, 64)	0
conv2d_2 (Conv2D)	(None, 7, 7, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 4, 4, 128)	0
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 64)	131,136

Total params: 223,808 (874.25 KB)
Trainable params: 223,808 (874.25 KB)
Non-trainable params: 0 (0.00 B)



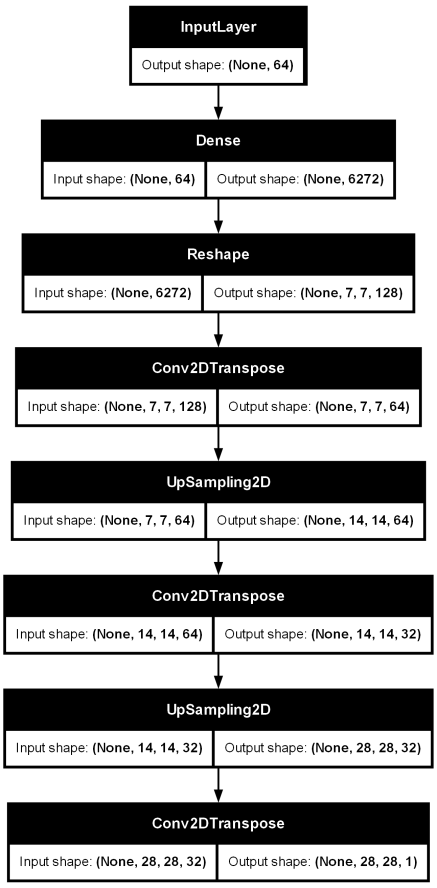
Q6.2

Build the Decoder

Model: "decoder"

Layer (type)	Output Shape	Param #
input_layer_1 (InputLayer)	(None, 64)	0
dense_1 (Dense)	(None, 6272)	407,680
reshape (Reshape)	(None, 7, 7, 128)	0
conv2d_transpose (Conv2DTranspose)	(None, 7, 7, 64)	73,792
up_sampling2d (UpSampling2D)	(None, 14, 14, 64)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 14, 14, 32)	18,464
up_sampling2d_1 (UpSampling2D)	(None, 28, 28, 32)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 28, 28, 1)	289

Total params: 500,225 (1.91 MB)
Trainable params: 500,225 (1.91 MB)
Non-trainable params: 0 (0.00 B)



Q6.4

Train the Autoencoder

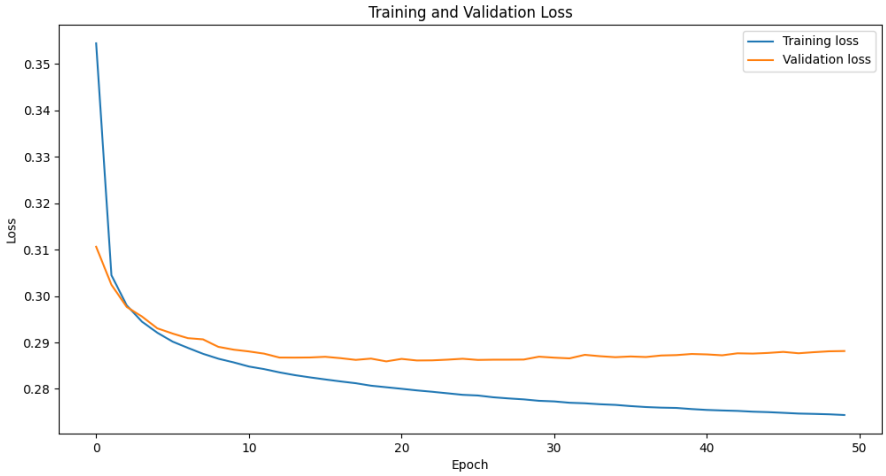


Figure 1: Training and Validation Losses

Q6.5

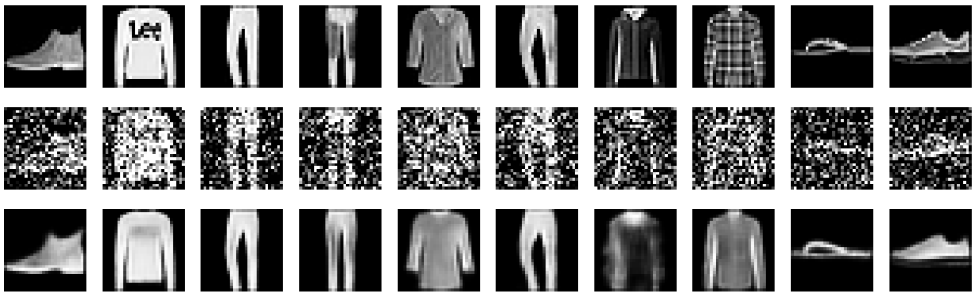


Figure 2: Results

Q6.6

Two application areas for denoising autoencoders

- **Medical Imaging:** To improve the quality of medical images, such as MRI or CT scans
- **Image Restoration:** To restore old or degraded photographs by removing artifacts and noise

How do you think an autoencoder architecture in general could be adapted for encryption purposes? Do you believe this approach would be effective?
Autoencoders can be adapted for encryption by encoding data into a compressed representation, i.e., ciphertext, that is difficult to interpret without the decoder. The way autoencoders compress data makes it difficult for unauthorized parties to reconstruct the original information without access to the decoder. This could potentially enhance security.