HOMEWORK SET #11

EE 510: Linear Algebra for Engineering

Assigned: 22 November 2023
Due: 2 December 2023

Directions: Each problem is worth 25 points.

1. The likelihood $g(x|\theta)$ of a Gaussian mixture is

$$g(x|\theta) = \sum_{j=1}^{J} w_j \; g(x|\theta_j) \tag{1}$$

where J is number of mixture component and

$$g(x|\theta_j) = \mathcal{N}(x|\mu_j, K_j) = \frac{1}{(2\pi)^{\frac{d}{2}}\sqrt{Det(K_j)}} e^{-\frac{1}{2}(x-\mu_j)^T K_j^{-1}(x-\mu_j)} \tag{2}$$

and $x \in \mathbb{R}^d$ with $d = 2$.

   a) Download the dataset: gmm_mixture.csv

   b) Load the dataset. Set the number of mixture component $J = 2$ and use the *sklearn* library in Python to fit a GMM on the dataset. (Figure 1a)

   c) Use the means, covariance matrices, and mixing weights to compute the mixture likelihood $g(x|\theta)$.

   d) Repeat (b) and (c) for $J = 3, 4, 5, ..., 10$.

   e) Plot the number of mixing weights against the likelihood .

   f) Determine the value of J^* with the maximum likelihood. Report the corresponding means, covariance matrices, and mixing weights.

2. Reduce the dimension of a dataset with 700 features and plot the explained variance.

   a) Download the dataset: pca_data.csv

   b) Load the dataset. Set the number of components ($n = 5$) and use the *sklearn* library in Python to fit a principal component analysis (PCA) model on the dataset. Use the code snippet in Figure 1b.

   c) Plot the explained variance against the index of the reduced components.

   d) Repeat (b) and (c) for $n = 10, 50, 100, 200$. Use the code snippet in Figure 1b.

```python
import numpy as np
from sklearn.mixture import GaussianMixture

# Load the dataset
x = np.loadtxt("gmm_mixture.csv",
               delimiter=",", dtype=float)


# Set the number of components and fit the GMM model
gmm = GaussianMixture(n_components=2, n_init=10)
gmm.fit(x)


# Print the means, covariance matrices,and weights
print(gmm.means_)
print(gmm.covariances_)
print(gmm.weights_)

```

(a)

```python
import numpy as np
from sklearn.decomposition import PCA

# Load the dataset
x = np.loadtxt("pca_data.csv",
               delimiter=",", dtype=float)

# Set the number of components
pca = PCA(n_components=100)
pca.fit(x)

# Plot the explained  variance
plt.figure(1, figsize=(4, 3))
plt.grid()
plt.plot(pca.explained_variance_, linewidth=2)
plt.axis('tight')
plt.xlabel('n_components')
plt.ylabel('Explained_variance')
```

(b)

Figure 1: Code snippets