Machine Learning (HWS24)

Assignment 4: Latent Variable Models

Nursultan Mamatov, nmamatov, 1983726
Cagan Yigit Deliktas, cdelikta, 1979012

December 8, 2024

1 Task 1: Probabilistic PCA

1.1 Part a

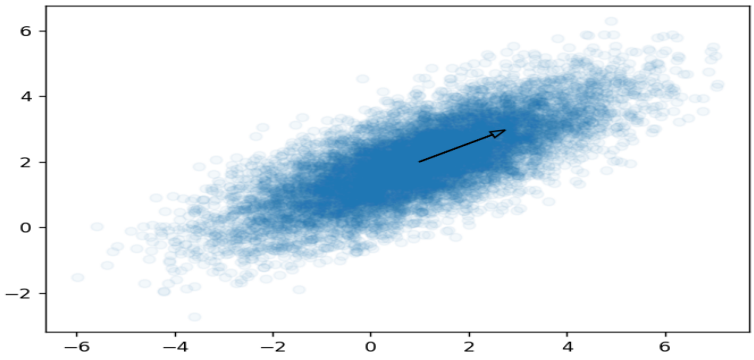


Figure 1: PPCA Data Visualization with $\sigma^2 = 0.5$

The Figure 1 visualizes **PPCA-generated data** with $\sigma^2 = 0.5$ (noise level), where:

$$\mathbf{X} = \mathbf{Z}\mathbf{W}^\top + \boldsymbol{\mu} + \boldsymbol{\epsilon},$$

where:

- $\mathbf{Z}$ are latent variables,
- $\mathbf{W}$ is the loading matrix (defining the direction of variance),
- $\boldsymbol{\mu}$ is the mean, and

- ϵ is noise ($\sigma^2 = 0.5$).

Plot Interpretation:

- **Scatterplot:** Points are clustered along a line, showing a strong low-dimensional structure.
- **Arrow:** Represents the main direction of variance (first principal component).

The scatter is tightly along the line, with noise slightly dispersing the points.

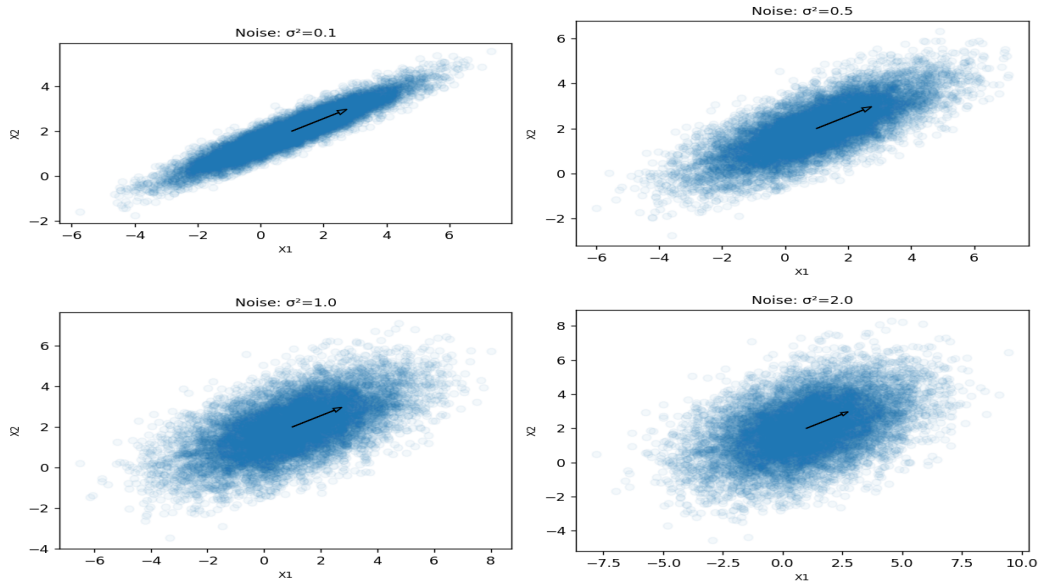


Figure 2: Effect of Noise on PPCA Data Structure

As σ^2 increases as shown in Figure 2:

- The points spread further from the principal direction, resulting in increased scatter.
- The alignment of the data with the arrow (principal component) becomes weaker.
- High noise levels obscure the underlying structure, making it harder to discern the latent relationships.

1.2 Part b

MLE for PPCA

To implement Maximum Likelihood Estimation (MLE) for PPCA, we compute the following parameters:

- $\boldsymbol{\mu}_{\text{MLE}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$, the mean vector.
- $\mathbf{W}_{\text{MLE}} = \mathbf{V}_L\sqrt{\boldsymbol{\Lambda}_L}$, the loading matrix, where:
  - $\mathbf{V}_L$ is the matrix of top L eigenvectors (right singular vectors) of the covariance matrix.
  - $\boldsymbol{\Lambda}_L$ is the diagonal matrix of top L eigenvalues (scaled squared singular values).
- $\sigma^2_{\text{MLE}} = \frac{1}{D-L}\sum_{i=L+1}^{D}\lambda_i$, the noise variance, where λ_i are the eigenvalues of the covariance matrix.

Why Does $\hat{\sigma}^2_{\text{MLE}} = 0$ When $L = 2$?

The noise variance σ^2_{MLE} captures the variance not explained by the latent dimensions. It is computed as:

$$\sigma^2_{\text{MLE}} = \frac{1}{D-L}\sum_{i=L+1}^{D}\lambda_i.$$

In the case of the toy PPCA dataset:

- The data has $D = 2$ features.
- When $L = 2$, the model uses both dimensions to explain the data.

Since there are no remaining dimensions ($D - L = 0$), all variance is captured by the latent structure, leaving no residual variance. Therefore:

$$\hat{\sigma}^2_{\text{MLE}} = 0.$$

Key Insight

When $L = D$, the PPCA model explains 100% of the variance using the latent dimensions. As a result, the noise variance σ^2 is zero.

1.3 Part d

1.3.1 Studying the scree plot

Based on the scree plot (see Figure 3), L=20 is a reasonable choice. This means that the dataset can be effectively described using 20 latent variables, as the remaining components (after the 20th) contribute very little additional information.

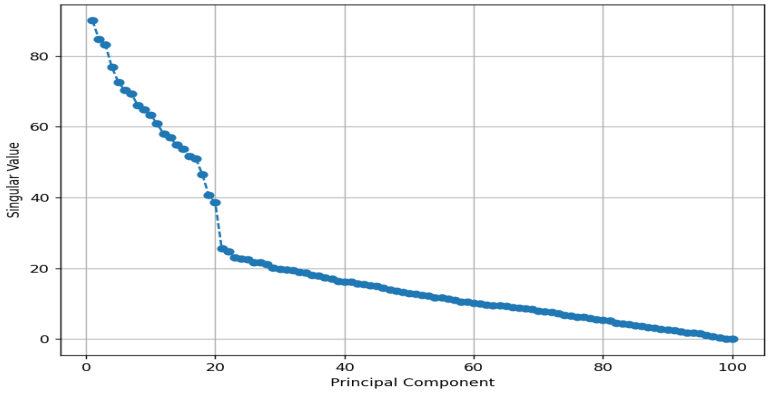


Figure 3: Scree Plot

1.3.2 Using validation data

Based on the NLL curve (see Figure 4), we observe that for small values of L, the negative log-likelihood (NLL) is relatively high and fluctuates, indicating that the model is underfitting and not capturing enough of the underlying structure in the data. As L increases and approaches 20, the NLL stabilizes, suggesting that the model with this number of latent variables provides a good fit. After a certain point, further increasing L continues to decrease the NLL, indicating that the model's fit improves with more latent variables. However, while a decrease in NLL suggests better performance, it also raises concerns about potential overfitting as the model becomes more complex. Thus, the optimal number of latent variables is likely the point where the NLL stabilizes, balancing the model's complexity and its ability to generalize well to unseen data.

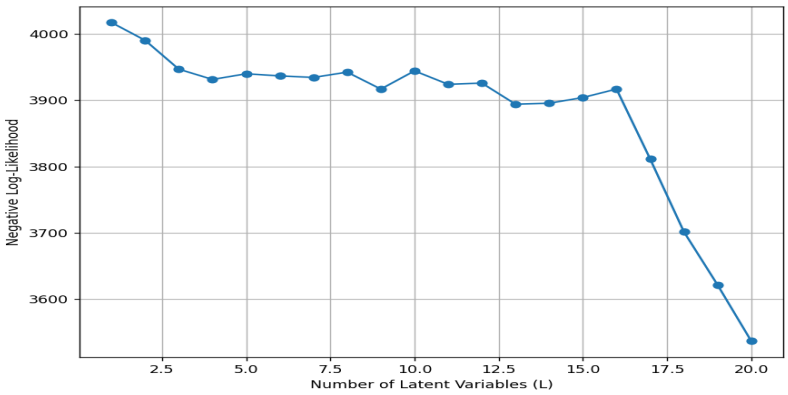


Figure 4: Validation NLL for Different Latent Variables (L)

2 Task 2: Gaussian Mixture Models

2.1 Part a

The plot (see Figure 5a) shows a dataset of 10,000 points generated from a Gaussian Mixture Model (GMM) with 5 components. Each component represents a Gaussian distribution defined by its mean, covariance matrix, and mixing coefficient. The data points are clustered into 5 distinct groups, each corresponding to one of the Gaussian components. The clusters have varying sizes, shapes, and orientations due to the differences in their covariance matrices. These covariance matrices were generated using a combination of scaled variances and random rotations, resulting in elliptical clusters with unique properties. The color of each point in the plot indicates its assigned component, demonstrating how GMM models complex data distributions with diverse clusters.

(a) Toy dataset clustering with GMM

(b) Toy dataset clustering with K-Means

Figure 5: Comparison of clustering results for GMM and K-Means

2.2 Part b

The plot (see Figure 5b) shows the clustering results of K-Means applied to the `toy_gmm` dataset, with data points colored by their assigned cluster labels. While K-Means captures some clusters correctly, it fails in overlapping regions and for clusters with non-spherical shapes due to its assumption of equally sized, spherical clusters. This result was expected because the `toy_gmm` dataset was generated using a Gaussian Mixture Model (GMM) with clusters of varying shapes, sizes, and orientations. These differences highlight the limitations of K-Means when applied to complex datasets, especially those with overlapping or elongated clusters.

2.3 Part d

The GMM clustering (see Figure 6) with $K = 5$ aligns well with the true cluster structure in the data. Each data point is assigned to the component with the highest posterior

probability, resulting in a plot that captures the shapes, sizes, and orientations of the clusters better than K-Means (see Figure 5b). In contrast, K-Means struggles with overlapping regions and non-spherical clusters, as it assumes equally sized, spherical clusters. The GMM's flexibility allows it to model complex cluster shapes, showing an improvement over K-Means, particularly in areas where clusters overlap.

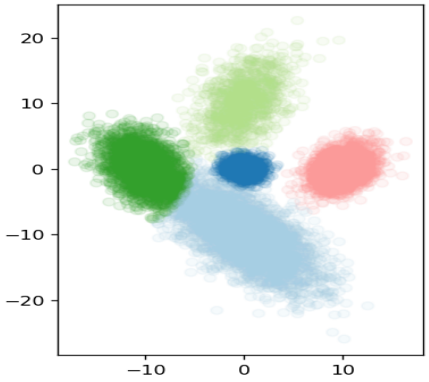


Figure 6: GMM clustering with K=5

2.4 Part e

For $K = 4$:

- When the number of components is smaller than the actual number of clusters, the GMM tends to merge some clusters (see Figure 7). For example, two nearby clusters might be combined into a single component.
- However, across different runs, the results are **not consistent**. The GMM may merge different clusters depending on the initialization, leading to variability in how the clusters are merged across repetitions.
- This indicates that the GMM is sensitive to the initial conditions when the number of components is too small for the data, resulting in inconsistent merging of the actual clusters.

For $K = 6$:

- When the number of components exceeds the actual number of clusters, the GMM may split existing clusters into smaller parts (see Figure 8). For example, a single large cluster might be divided into multiple components.
- As with $K = 4$, the results across multiple runs are **not consistent**. Different runs may produce different splits depending on the random initialization.

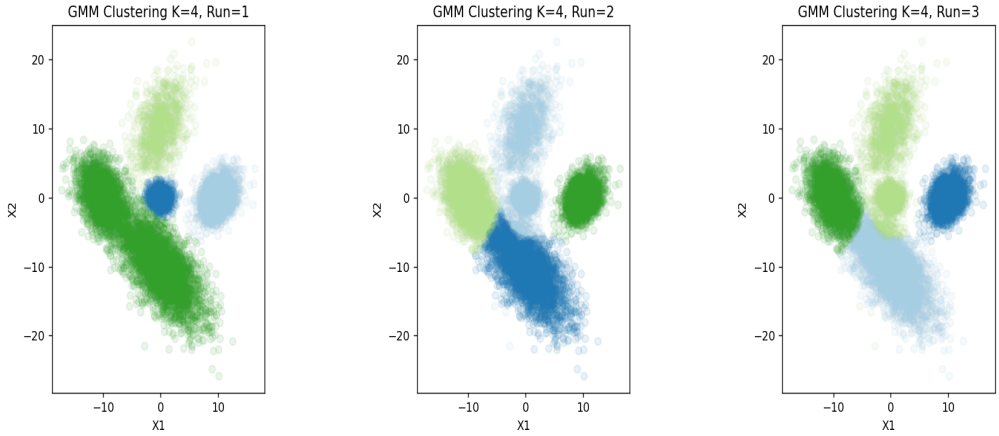


Figure 7: GMM clustering with K=4

- This highlights the overfitting nature of using too many components, where the GMM tries to fit the noise in the data, and the variability across runs shows the model's instability when K is larger than necessary.

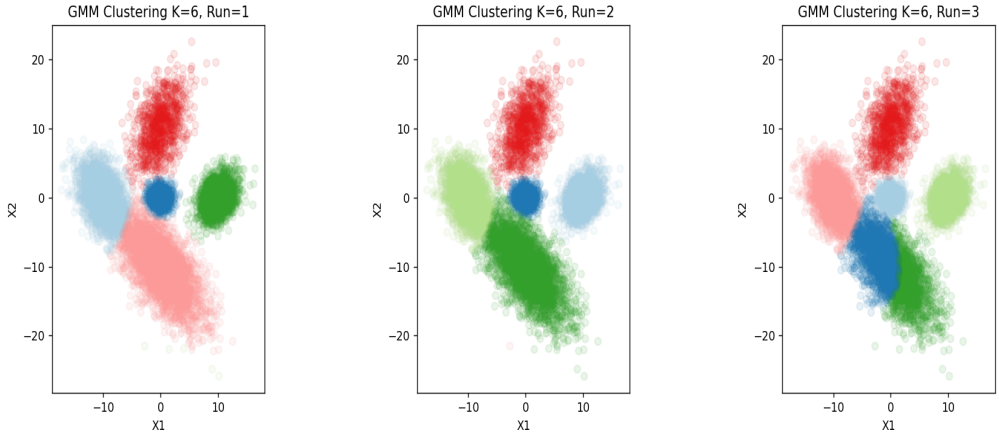


Figure 8: GMM clustering with K=4

Key Observations:

- **Variability Across Runs:** The variability across runs suggests that GMM is sensitive to initial conditions, especially when the number of components K is not well-suited for the data.
- **Choosing K:** The best clustering results are obtained when $K = 5$, which matches the true structure of the data. $K = 4$ results in underfitting by merging clusters, while $K = 6$ leads to overfitting by splitting clusters.

2.5 Part f: Optional

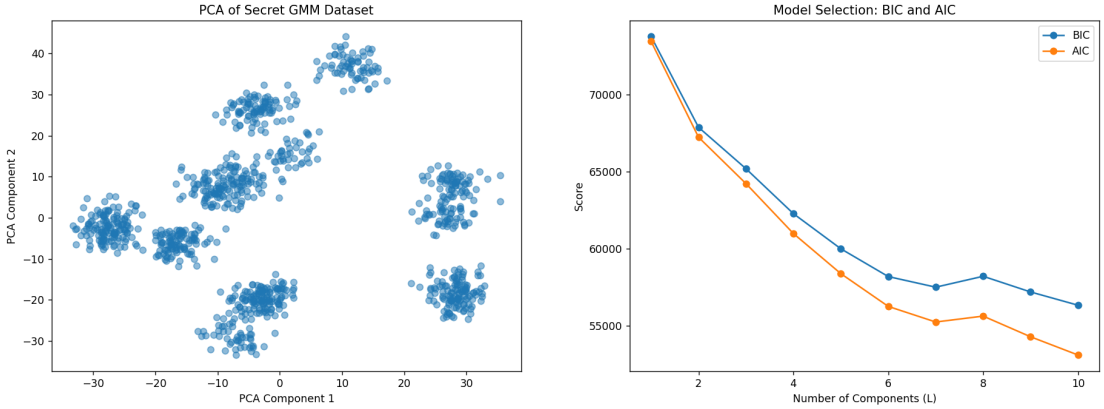
Based on the analysis of the secret dataset, the following observations were made:

- **PCA Visualization**: The PCA plot (see Figure 9) suggests the presence of **9 distinct clusters**, indicating that the data likely has 9 components. However, this is based on a 2D projection of the data, which may not capture all the underlying structure.

- **BIC/AIC Model Selection**: The GMM model, when evaluated using **BIC** and **AIC**, suggests that the optimal number of components is **10**. This suggests the model fits the data best with 10 components, even though the PCA plot shows only 9 clusters.

Justification: While the PCA plot shows 9 clusters, the BIC and AIC results indicate 10 components because the GMM model captures finer variations in the higher-dimensional space that PCA does not. The extra component may correspond to subtle variations or noise in the data.

Thus, the data likely has **9 main clusters**, but **10 components** is optimal according to the GMM model selection.



(a) PCA of Secret GMM Dataset

(b) BIC and AIC graph

Figure 9: Comparison of PCA and BIC/AIC for Selecting the Number of Components

References

[1] Gemulla, R. (2024). *IE 675b: Machine Learning* [Lecture slides]. University of Mannheim.

[2] Murphy, K. P. (2022). *Probabilistic machine learning: An introduction*. The MIT Press. https://probml.github.io/pml-book/book1.html

Ehrenwörtliche Erklärung

Ich versichere, dass ich die beiliegende Bachelor-, Master-, Seminar-, oder Projektarbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und in der untenstehenden Tabelle angegebenen Hilfsmittel angefertigt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen. Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Declaration of Used AI Tools

Tool	Purpose	Where?	Useful?
ChatGPT	Rephrasing	Throughout	+
ChatGPT	Enhancing understanding of mathematics and code	Throughout	+

Unterschrift
Mannheim, den 8. December 2024