Elements of Machine Learning, WS 2023/2024

Jilles Vreeken and Krikamol Muandet

Exercise Sheet #5: *Unsupervised Learning*

Deadline: Thursday, January 18, 2024, 15:00

Before solving the exercises, read the instructions on the course website.

- For each theoretical problem, submit a single `pdf` file that contains your answer to the respective problem. This file may be a scan of your (legible) handwriting.
- For each practical problem, submit a single `zip` file that contains
  - the completed jupyter notebook (`.ipynb`) file,
  - any necessary files required to reproduce your results, and
  - a `pdf` report generated from the jupyter notebook that shows all your results.
- For the bonus question, submit a single `zip file` that contains
  - a `pdf` file that includes your answers to the theoretical part,
  - the completed jupyter notebook (`.ipynb`) file for the practical component,
  - any necessary files required to reproduce your results, and
  - a `pdf` report generated from the jupyter notebook that shows your results.
- **Every team member** has to submit a signed Code of Conduct.
- **IMPORTANT** You must make the team on CMS *before* you upload the solutions. If you upload the solutions first and create the team after it, the solution will not show for the new team member!

Problem 1 (T, 2 + 1 + 5 Points). **Dimensionality Reduction** (ISLP 12.2)
In the course, you learned about different techniques for dimensionality reductions that embed high dimensional data into a low dimensional space.

1. Explain in at most 50 words, how *PCA* and *tSNE* approach dimensionality reduction respectively. Name one disadvantage for each of the methods, again using a maximum of 50 words.
2. Why is it important to normalize the data before applying PCA?
3. Given a dataset of n observations $X \in \mathbb{R}^{n \times p}$ and a column wise mean of zero, the first principal component is the direction of maximum variance in the data, i.e.

$$\text{argmax}_{w \in \mathbb{R}^p, ||w||=1} \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} w_j x_{i,j} \right)^2 .$$

   Show that the first principal component also corresponds to the eigenvector with the largest eigenvalue of the covariance matrix $X^T X$ of the data.

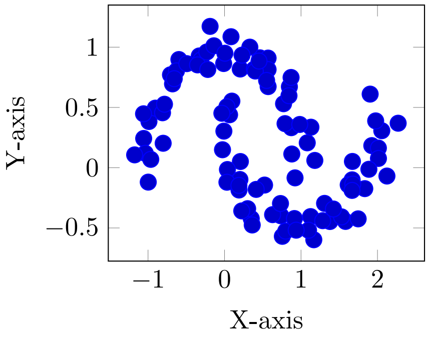

Figure 1: Data Set 1

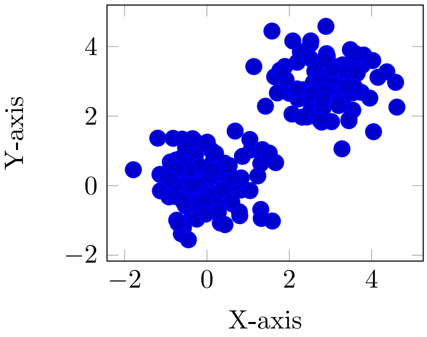

Figure 2: Data Set 2

Problem 2 (T, 2 + 2 + 1 Points). **Hierarchical Clustering** (ISLP 12.4.2)

1. Consider the two datasets given in Figure 1 and Figure 2. Argue for each dataset whether single linkage clustering or complete linkage clustering is more appropriate.

2. Hierarchical clustering can be useful for clustering documents based on their content. Let each document be represented as a vector of word counts, i.e. the i-th entry of the vector is the number of times the i-th word appears in the document. Argue why the euclidean distance is not a good choice for measuring the dissimilarity between two documents. What alternative distance could be used and what advantage does it offer?

3. Consider the CIFAR-10 Image Dataset[1]. Assume we use a hierarchical clustering method to cluster the set of unlabeled images. What additional insights beyond the class labels could be gained from the hierarchical clustering? (Hint: What top level hierarchy would you expect to see in the dendogram for CIFAR 10?)

[1] Contains 32x32 RGB images with class labels from {`Airplane`,`Automobile`,`Bird`,`Cat`,`Deer`,`Dog`,`Frog`,`Horse`,`Ship`,`Truck`}

Problem 3 (T,4+1+2 Points). **K-Means**

1. (Alternate interpretation of K-means, ISLP 12.4.1 and Lecture-10, Slide 7/8.)
   In K-means we iteratively optimize the cluster assignment $C_1, \dots, C_K$ and centroids $\bar{x}_1, \dots, \bar{x}_K$ as to minimize the Objective

$$\min_{\bar{x}_k, C_k} \sum_{k=1}^{K} W(C_k) = \min_{\bar{x}_k, C_k} \sum_{k=1}^{K} \sum_{x_i \in C_k} 2||x_i - \bar{x}_k||^2 \ .$$

   Show that Llyods algorithm, once randomly initialized, converges to a local optimum. As a reminder, the algorithm involves the following two steps.

   (a) Fix C_k , and recompute $\bar{x}_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} x_i$.

   (b) Update $C_k = \{x_i \mid \forall l \neq k : ||x_i - \bar{x}_l||^2 > ||x_i - \bar{x}_k||^2\}$ to assign each datapoint x_i to the closest centroid $\bar{x}_k$.

   For ease of notation, you may use the cluster index $c(i)$ of a datapoint to denote its assigned centroid as $\bar{x}_{c(i)}$.

2. You are given a dataset of fruits $\{Apple, Orange, Rasperry, Strawberry\}$ and pairwise dissimilarities between these fruits as specified in this table.

d(a,b)	Apple	Orange	Rasperry	Strawberry
Apple	0	1	5	4
Orange	1	0	4	5
Rasperry	5	4	0	1
Strawberry	4	5	1	0

   Given the choice to use Llyods algorithm with either K-means or K-medoids, state your opinion which method is better suited for this dataset and why.

3. Clearly you should not only compares apples to oranges, but also cluster {Apple,Orange} as well as {Rasperry,Strawberry}. Provide a scenario with $K = 2$ using your selected method from 3.2 where we do not obtain the desired result. Assume that we randomly initialize the centroids/medoids from the dataset in the beginning.

Problem 4 (P). **Programming Problem**
In this problem, you will perform and analyze different types of clustering on the data.

1. Clustering (3 + 3 + 2 + 2 Points)

   (a) Implement the K-means algorithm from scratch. Fill out the functions in the jupyter notebook and run your code on the example dataset provided to you. Generate a plot of the data and the cluster centers using your implementation of K-Means.

   (b) Next, implement the K-medians variant. Here, the centroids are made up of the median of each dimension instead of the average. Compute the respective within cluster variations for k-means and k-medians on the provided dataset and compare them. Does a better objective also result in a subjectively better clustering?

   (c) Load the `moons.csv` file provided to you. Run your K-Means algorithm on this dataset and plot the resulting clustering. Analyze how well K-means performs and why it is/isn't a good choice for this dataset.

   (d) Use the hierarchical clustering implementation from `sklearn` to cluster the dataset with single linkage and complete linkage. Generate a plot of the solution for each of the two methods and explain where the differences in results come from.

2. Dimensionality reduction (1.5 + 1.5 + 2 Points)

   (a) We are using `sklearn` to load the digits dataset. Use the `PCA` class from `sklearn` to reduce the dimensionality of the dataset to 2 dimensions. Plot the resulting 2-dimensional dataset and color the points according to their class label.

   (b) Use the `TSNE` class from `sklearn` to reduce the dimensionality of the dataset to 2 dimensions. Plot the resulting 2-dimensional dataset and color the points according to their class label.

   (c) Compare the results of the two methods and explain the differences.

   Note: Useful Python packages for this assignment are `sklearn`.

Problem 5 (Bonus). **Clustering Conditional Distributions**
In Linear Regression, we assume that the data Y is generated by a linear function $Y = f(X) + \epsilon$, where ϵ is a Gaussian distributed noise variable. While the main focus of subsequent lectures has been on the estimation of a function f that is non-linear, in this task we will be focussing on the noise variable ϵ. In particular, often the distribution of ϵ is not constant, but depends on the input variable X, a phenomenon also known as heteroscedasticity. One example for this is the relationship between the variance of income and age. While children generally do not have any income, the variance of income increases with age until retirement, after which it decreases again.

- We consider a noise variable $\epsilon \in \mathbb{R}$ and a variable $X \in \{1, \ldots, b\}$ with a domain comprised of a finite amount of b integer values.
- We assume that the noise variable $\epsilon \sim \mathcal{N}(0, \sigma^2(X))$ is Gaussian distributed with a mean of 0 and a variance determined as $\sigma^2(x) > 0$.
- You are given a dataset $\{(x_i, e_i)\}_{i=1}^n$ of n observations of X and ϵ.

Your task is to estimate the conditional variance of ϵ given X. This can easily be done by estimating the variance $\sigma^2(x)$ for each value of $x \in \mathcal{X}$. The problem here is the finite/limited amount of data we have in practice. Therefore, we want to cluster *neighboring* values of X together where the noise is similar and estimate the variance within each cluster. In particular, we use the Bayesian Information Criterion from the previous lecture to determine the optimal tradeoff between the number of clusters and the data likelihood given the clusters. To this end, your tasks are

1. Derive the maximum likelihood estimate for the variance σ^2 given a set of m observations $\{e_i\}_{i=1}^m$.
2. Derive the Bayesian Information Criterion for a given set of clusters $\{C_j\}_{j=1}^K$, where the clusters $C_j = \{e_i | \alpha_j \leq x_i \leq \beta_j\}$ are described by the interval $[\alpha_j, \beta_j]$ on X. As a reminder, the BIC is given by
$$\mathrm{BIC}(\mathcal{L}, p, n) = -\log \mathcal{L} + p \log(n) ,$$
where $\mathcal{L}$ is the likelihood of the dataset given the model and p is the number of parameters in the model. Use the maximum likelihood estimate as per 5.1 to estimate the variance σ_j^2 for each cluster.
3. Implement a greedy algorithm in Python to find a clustering of noise in this fashion. Start by implementing the BIC score to evaluate a given clustering. The algorithm then has the following steps:
   - Initialize $K = b$ clusters where each cluster C_b contains all observations where $x_i = b$.
   - Greedily merge the two neighboring clusters C_j and C_k with $(\beta_j = \alpha_k)$ that lead to the lowest BIC score.
   - Repeat until no further improvement is possible.

   Execute your code on the provided `bonus.csv` dataset and plot the results. Is the clustering produced by the greedy algorithm guaranteed to be optimal in regard to the BIC?

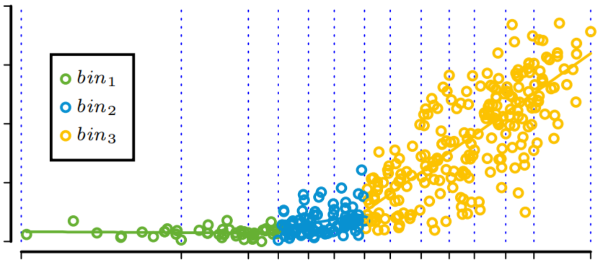

Figure 3: Example of a dataset with clustered heteroscedastic noise.