Northeastern University, Khoury College of Computer Science

CS 6220 Data Mining — Assignment 4

Due: October 17, 2023 (100 points)

Chun Wei Tseng
https://github.com/Chun-Wei-Tseng/CS6220-Assignment-4
tseng.wayne@gmail.com

Parameter Estimation

It is well-known that light bulbs commonly go out according to a Poisson distribution, and are independent regardless of whether or not they're made in the same factory. An architect has outfitted a building with 32,000 of the same lightbulbs.

Assuming the Poisson distribution has the form:

$$p(X|\lambda) = \frac{exp^{-\lambda}\lambda^{x_i}}{x_i!} \qquad (0.1)$$

1. derive the maximum likelihood estimate of the parameter λ.

Ans:

- X: random variable.
- x_i: number of occurrences
- λ : expected value
- For n observations, the joint likelihood is: $L(\lambda) = \prod_1^n \frac{exp^{-\lambda}\lambda^{x_i}}{x_i!}$ $or \prod_1^n \frac{e^{-\lambda}\lambda^{x_i}}{x_i!}$
- Take nature log on both side:
- $\log(L(\lambda)) = \log(\prod_1^n \frac{e^{-\lambda}\lambda^{x_i}}{x_i!}) = \log(\frac{e^{-\lambda}\lambda^{x_1}}{x_1!} * \frac{e^{-\lambda}\lambda^{x_2}}{x_2!} * \dots * \frac{e^{-\lambda}\lambda^{x_n}}{x_n!})$
  $= \sum_1^n(\log(e^{-\lambda}) + \log(\lambda^{x_i}) - \log(x_i!)) = \sum_1^n(-\lambda + x_i * \log(\lambda) - \log(x_i!))$
- Take derivative of the log of likelihood fumction, then set it as 0 to find the maximum likelihood:
  $\frac{d}{d\lambda}\log(L(\lambda)) = \sum_1^n(-1 + \frac{x_i}{\lambda}) = 0$ → $-n + \frac{\sum_1^n x_i}{\lambda} = 0$ → $\lambda = \frac{\sum_1^n x_i}{n}$
- This means that the maximum likelihood estimate of the parameter λ is equal to the mean of observations.

K-Means

The normalized automobile distributor timing speed and ignition coil gaps supplied are from production F-150 trucks over the years of 1996, 1999, 2006, 2015, and 2022. We have stripped out the labels for the five years of data. Each sample in the dataset is two-dimensional, i.e. $x_i \in \mathbb{R}^2$, and there are N = 5000 instances in the data.

Vanilla *k*-Means

In this part of the homework, we'll take a look at how we can identify patterns in this data despite not having the labels. We'll start with the simplest approach, the k-Means unsupervised clustering algorithm.

2. Implement a simple k-means algorithm in Python on Colab with the following initialization:

$$x_1 = \begin{pmatrix}10\\10\end{pmatrix}, x_2 = \begin{pmatrix}-10\\-10\end{pmatrix}, x_3 = \begin{pmatrix}2\\2\end{pmatrix}, x_4 = \begin{pmatrix}3\\3\end{pmatrix}, x_5 = \begin{pmatrix}-3\\-3\end{pmatrix}, \tag{0.2}$$

You need only 100 iterations, maximum, and your algorithm should run very quickly to get the results. In order to maintain consistency between submissions, use a random seed of 27. You can do this with

```
>> numpy.random.seed(seed=27)
```

3. Scatter the results in two dimensions with different clusters as different colors. You can use matplotlib's pyplot functionality:

```
>> import matplotlib.pyplot as plt
>> plt.scatter(<YOUR CODE HERE>)
```

4. You will notice that in the above, there are only five initialization clusters. Why is k = 5 a logical choice for this dataset? After plotting your resulting clusters and. What do you notice?

→ Ans: Those five initialization clusters are spaced relatively far from each other, which means that we are expected to get 5 separate centroids after processing them with vanilla k-means. I notice that points in the same class are locate vertically in the graph that seem further from its neighbors in the same class than horizontal neighbors in different class.

With Production Information

Very often, it is possible to obtain additional information about the collected data. This some- times allows us to define a new mathematical operators (including distances). In this part of the homework, we'll look at how to use this information to improve our modeling with an understanding of how two features in each sample are related.

A common distance metric is the Mahalanobis Distance with a specialized covariance.

$$d(x,y) = (x-y)^T(P^TP)^{-1}(x-y) \tag{0.3}$$

where x and y are two points of dimensionality m (2 in this case), and d(x, y) is the distance between them. In the case of the F150 engine components, P is a known relationship through Ford's quality control analysis each year, where it is numerically shown as below:

$$P = \begin{pmatrix} 10 & 0.5 \\ -10 & 0.25 \end{pmatrix} \tag{0.4}$$

5. Implement a specialized k-means with the above Mahalanobis Distance. Scatter the results with the different clusters as different colors. What do you notice? You may want to pre- compute P^{-1} so that you aren't calculating an inverse every single loop of the k-Means algorithm.

→ Ans: They way of mapping data points into classes (clusters) is different. When we use vanilla k mean to process the data, the classes seem to be cut vertically in the graph. When I use Mahalanobis Distance to process k means on the same data set, classes are clustered horizontally. Centroids also change from spreading horizontally to spreading vertically.

6. Calculate and print out the first principle component of the aggregate data.

→ Ans:

```
First Principle Component of the aggregate data: [ 0.99838317 0.05684225]
```

7. Calculate and print out the first principle components of each cluster. Are they the same as the aggregate data? Are they the same as each other?

→ Ans:

```
The first principle component of 1 Cluster: [ 0.99992533 -0.01222027]
The first principle component of 2 Cluster: [ 0.99989374 -0.01457781]
The first principle component of 3 Cluster: [ 0.99990986 -0.01342629]
The first principle component of 4 Cluster: [ 0.99993306 -0.01157047]
The first principle component of 5 Cluster: [ 0.99993527 -0.01137789]
```

- The first principle components of each cluster are different from the aggregate data by a small amount.