Northeastern University, Khoury College of Computer Science

CS 6220 Data Mining | Assignment 4

Due: February 15, 2023(100 points)

Chang Liu

https://github.com/Chang-Liu-Harry/6220DataMining

1.

Since the observations of Poisson distribution are independent, the likelihood function for a data set of n observations is to be multiplying the probability of X equals to x1 to xn:

$$L(\lambda) = \prod_{i=1}^{n} p(x_i|\lambda) = \prod_{i=1}^{n} \frac{e^{-\lambda}\lambda^{x_i}}{x_i!}$$

After we have the likelihood function, we want to know at what lambda value, we are most likely to get this n observations. We take the log of likelihood function and like it's derivative to be zero to get the result:

$$\ell(\lambda) = \log L(\lambda) = \sum_{i=1}^{n} \log\left(\frac{e^{-\lambda}\lambda^{x_i}}{x_i!}\right)$$

$$\frac{d\ell}{d\lambda} = \sum_{i=1}^{n}\left(-1 + \frac{x_i}{\lambda}\right)$$

$$0 = -n + \frac{1}{\lambda}\sum_{i=1}^{n} x_i$$

And the result:

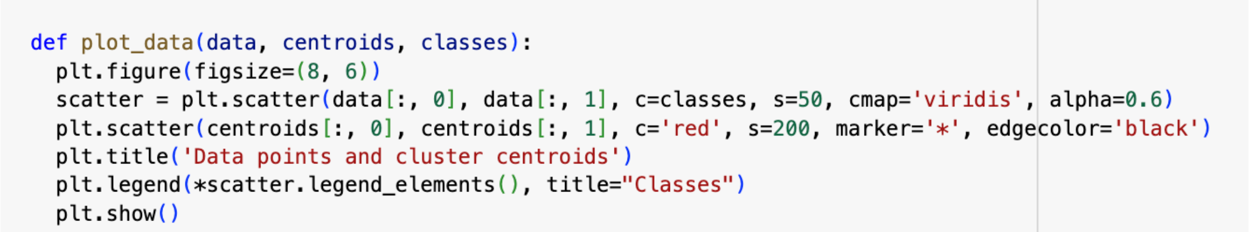
$$\lambda = \frac{1}{n}\sum_{i=1}^{n} x_i$$

2.

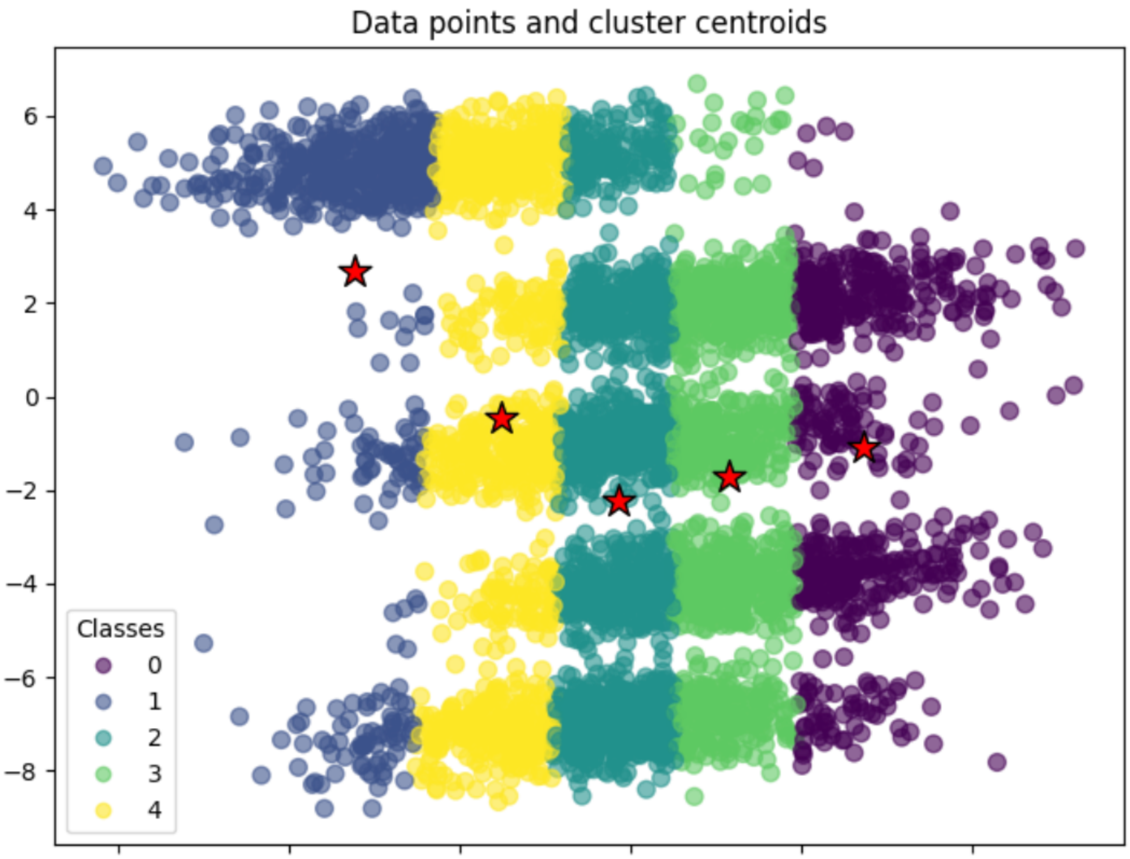
Colab link: https://colab.research.google.com/drive/1jb1ImqPOaUxSYq1jJ39Cfh-S8AJKmwmH?usp=sharing

When we don’t use P, it’s just a vanilla k-means algorithm, because the default p: dot(P,P’) = 1.

3.



```
def plot_data(data, centroids, classes):
  plt.figure(figsize=(8, 6))
  scatter = plt.scatter(data[:, 0], data[:, 1], c=classes, s=50, cmap='viridis', alpha=0.6)
  plt.scatter(centroids[:, 0], centroids[:, 1], c='red', s=200, marker='*', edgecolor='black')
  plt.title('Data points and cluster centroids')
  plt.legend(*scatter.legend_elements(), title="Classes")
  plt.show()
```

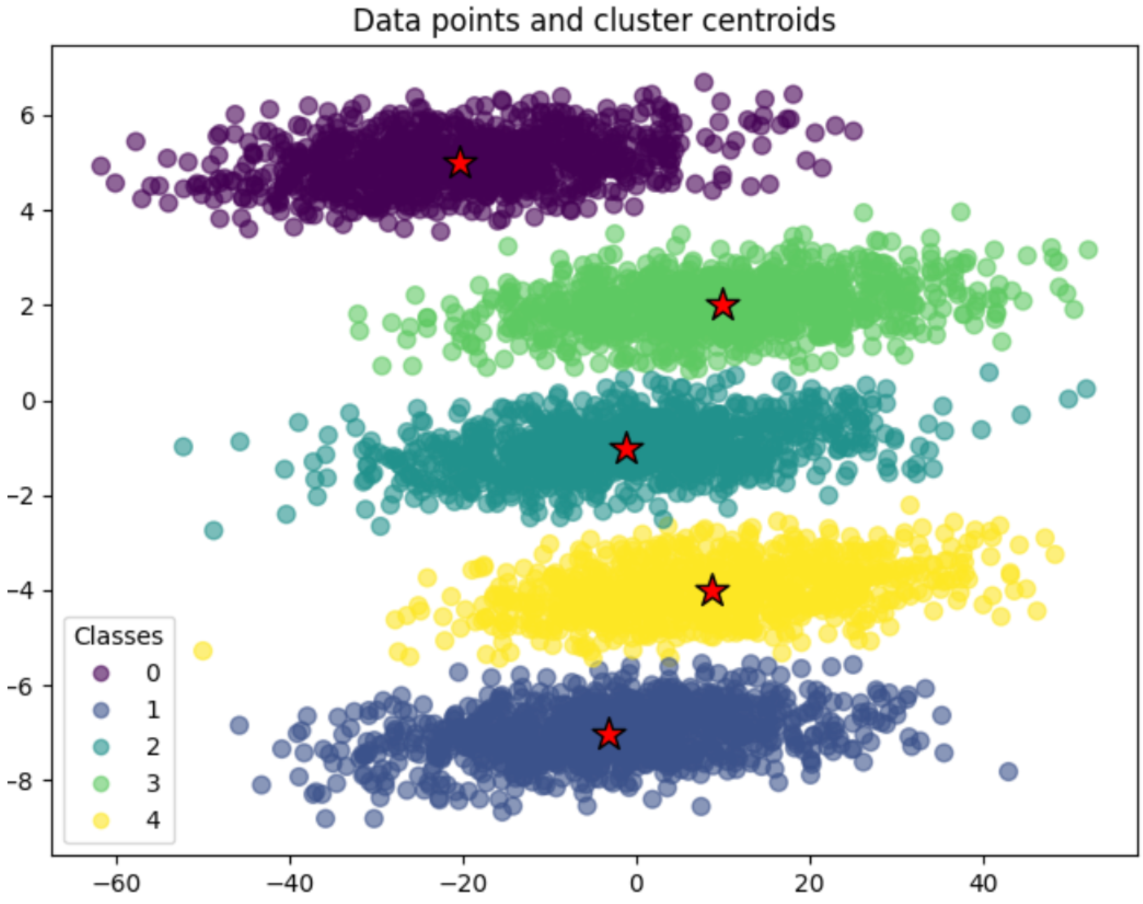


4.

Picking 5 clusters is reasonable since our data production F-150 trucks over 5 years. And as can be seen in the graph, the data is scattered into 5 main parts, but our clustering is not getting the 5 groups but 5 stripes, which is not exactly what we want.

5.

I implemented a k-means clustering function, and defined the Mahalanobis distance function as well as updating centroids for readability of the code. Setting the p to default, which is not using p, the now we are getting the correct clusters:



6.

```
First principal component for all data : [ 0.99838317 -0.05684225]
7.
First principal component for cluster 0: [0.99993527 0.01137789]
First principal component for cluster 1: [0.99992533 0.01222027]
First principal component for cluster 2: [0.99990986 0.01342629]
First principal component for cluster 3: [0.99993306 0.01157047]
First principal component for cluster 4: [0.99989374 0.01457781]
No they are all different.
```