HW3 - Machine Learning in Healthcare

Shir Ricon

204632780

Part 1 - Clustering

a. In the K-means algorithm the updates of the centroids at each iteration are done by calculating the mean of the examples that belong to each cluster, whereas in the K-medoid algorithm the updates of the medoids at each iteration is done by swapping the medoid points (that are selected from points in the dataset) with none-medoid points (also selected from the dataset) that minimizes the L_1 distances between the medoid point and the points that belong to the cluster. Because the mean calculation is more sensitive to outliers, in comparison with selecting a representative point from the cluster, the K-medoid algorithm is more robust to noise and outliers.

b. In order to find the centroid (μ) that minimize the term $\sum_{i=1}^{m}(x_i-\mu)^2$, we will calculate it's derivation by μ and compare it to 0:

$$\frac{d}{d\mu}\left(\sum_{i=1}^{m}(x_i-\mu)^2\right)=0 \quad\rightarrow\quad -2\sum_{i=1}^{m}(x_i-\mu)=0 \quad\rightarrow\quad \sum_{i=1}^{m}x_i-m\mu=0 \quad\rightarrow\quad \mu=\frac{\sum_{i=1}^{m}x_i}{m}$$

$$\rightarrow\quad \mu=\frac{\sum_{i=1}^{m}x_i}{m}=mean(x_i)$$

To confirm it is indeed the minimum, we will calculate a second derivation:

$$\rightarrow\quad \frac{d^2}{d\mu^2}\left(\sum_{i=1}^{m}(x_i-\mu)^2\right)=\frac{d}{d\mu}\left(-2\cdot\sum_{i=1}^{m}(x_i-\mu)\right)=\frac{d}{d\mu}\left(-2\cdot\sum_{i=1}^{m}x_i+2\sum_{i=1}^{m}\mu\right)=2m>0$$

Bonus:

We want to prove that the centroid (medoid) which minimizes the term $\sum_{i=1}^{m}|x_i-\mu|$ is the median of m examples, given that μ belongs to the dataset. In the case when m (number of examples) is an odd number we will define three new parameters:

μ – An example from the dataset
k – The number of examples with lower value then μ
l – The number of examples with higher value then μ

First, we'll split the dataset into three groups:

- All the examples with lower value then μ (index by $i \,\epsilon\, [1,k]$)
- The example μ
- All the examples with higher value then μ (index by $j \,\epsilon\, [1,l]$)

The term $\sum_{i=1}^{m}|x_i-\mu|$ can be presented as the sum of the three groups above:

$$\rightarrow\quad \sum_{i=1}^{m}|x_i-\mu|=\left(\sum_{i=1}^{k}|x_i-\mu|\right)+(\mu-\mu)+\left(\sum_{j=1}^{l}|x_j-\mu|\right)=$$

$$=-\sum_{i=1}^{k}(x_i-\mu)+\sum_{j=1}^{l}(x_j-\mu)=-\sum_{i=1}^{k}(x_i)+k\mu+\sum_{j=1}^{l}(x_j)-l\mu$$

$$\rightarrow \quad \frac{d}{d\mu}\left(-\sum_{i=1}^{k}(x_i)+k\mu+\sum_{j=1}^{l}(x_j)-l\mu\right)=k-l$$

When μ is set to be the median, we get that $k=l$. By comparing the derivate to 0 we get that the minimum of the term $\sum_{i=1}^{m}|x_i-\mu|$ occurs when $k=l$, which proves that choosing μ as the median minimizes the term when μ belongs to the dataset.

Part 2 – SVM

- A → 1 (Linear Kernel with $C=0.01$)
- D → 2 (Linear Kernel with $C=1$)

Explanation:
The hyper parameter C in SVM classifiers can be adjusted to set the strength of the regularization. Higher value of C will increase the penalty for samples inside the margins or samples that are misclassified. With a lower value of C we will receive the opposite outcome. While both linear classifiers classified correctly all the examples it appears that in figure A the margins set by the support vectors include some of the examples between them. Because of this, it seems that figure A represent a linear classifier with lower regularization (smaller C) compared to the linear classifier represented in figure D.

- C → 3 $\left(2^{nd}\ order\ polynomial\ kernel\right)$
- F → 4 $\left(10^{nd}\ order\ polynomial\ kernel\right)$

Explanation:
In SVM classifiers with polynomial kernels the decision boundary is not linear and doesn't form a closed shape. The complexity of the borders created by the classifier is dependent on the degree of the polynomial of the kernel, so that with higher degree the complexity of the borders will be greater. From the figures representing the polynomial SVM classifiers figure F has a more complicated decision boundary borders compared to figure C, which is why the degree of the polynomial kernel in figure F is higher.

- E → 5 (RBF Kernel with $\gamma=0.2$)
- B → 6 (RBF Kernel with $\gamma=1$)

Explanation:
In figures B and E, the border of the classifier forms a closed shape. This indicates the use of an RBF SVM classifier. In this classifier the hyper parameter γ decides the curvature of the decision boundary and is inversely proportional to the variance of in the kernel. A high value of γ will lead to more curvature which will result in a smaller boundary area. From the figures representing the RBF SVM classifiers figure B has a smaller area compared to figure E, which is why figure B has a bigger γ.

Part 3 – Capability of generalization

a. The scientific term of balance in machine learning that Einstein meant is **Generalization**.

b. The AIC can be used as a model selection criterion to assess and compare the quality of different GMM models. The first term is 2p, where p is the number of parameters. Increasing the number of parameters will increase the complexity of the model. Increasing the number of parameters too much might damage the generalization ability of the model and increase the AIC. In the second term, $2\ln(\hat{L})$, $\hat{L}$ is the likelihood that measures how well the model fits our training examples. This term increases as $\hat{L}$ is larger. Because this term is negative in the AIC equation, the AIC decreases as $\hat{L}$ is larger.

c. The two options are:
- Underfitting
- Overfitting

d. We are aiming for a low AIC. The AIC decreases when the model better fits the training examples. To avoid overfitting the AIC increases when we increase the number of learned parameters.

Bibliography:

- https://medium.com/@myselfaman12345/c-and-gamma-in-svm-e6cee48626be
- https://en.wikipedia.org/wiki/Akaike_information_criterion