Homework 1

Tuesday, October 15, 2024 3:09 PM

Problem 1

a) write out the log-likelihood function $\log P(D|A)$.

$$P_{ois}(X=x;\lambda)=\frac{\lambda^{x}e^{-\lambda}}{x!}\quad \forall x\in\{0,1,2,..\}\quad \lambda\geq 0$$

$$P(D|\lambda)=\prod_{i=1}\frac{\lambda^{x}e^{-\lambda}}{x!}$$

take log like we did in class because of e

$$\log P(D|\lambda)=\log\left(\prod_{i=1}\frac{\lambda^{x}e^{-\lambda}}{x!}\right)$$

$$\log P(D|\lambda)=\sum_{i=1}^{N}\log\left(\frac{\lambda^{x}e^{-\lambda}}{x_i!}\right)$$

$$\log P(D|\lambda)=\sum_{i=1}^{N}(x\log\lambda-\lambda-\log(x!))$$

$$\boxed{\log P(D|\lambda)=\sum_{i=1}^{N}(x_i\log\lambda-\lambda)}$$

b) take the log likelihood with respects to the param λ

$$\log P(D|A)=\sum_{i=1}^{N}(x_i\log\lambda-\lambda)$$

$$\frac{d}{d\lambda}(x_i\log\lambda-\lambda)$$

$$\frac{d}{d\lambda}(x_i\log\lambda)\quad\frac{d}{d\lambda}(-\lambda)$$

$$\frac{d}{d\lambda}(x_i\log\lambda)=\frac{x_i}{\lambda}$$

$$\frac{d}{d\lambda}(-\lambda)=-1$$

$$\boxed{\frac{d}{d\lambda}\log P(D|\lambda)=\sum_{i=1}^{N}\left(\frac{x_i}{\lambda}-1\right)}$$

c) set derivative to zero and solve for λ - call this maximizing value $\hat{\lambda}_{map}$

$$\sum_{i=1}^{N}\left(\frac{x_i}{\lambda}-1\right)=0$$

$$\sum_{i=1} \left(\frac{x_i}{\lambda} - 1\right) = 0$$

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

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

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

$$\sum_{i=1}^{N} x_i = N\lambda$$

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

$$\boxed{\hat{\lambda}_{MLE} = \frac{1}{N} \sum_{i=1}^{N} x_i}$$

Problem 2

Maximum Pasta question ← all i could think of during class

$$\text{Gamma}(\Lambda = \lambda; \alpha, \beta) = \frac{\beta^{\alpha} \lambda^{\alpha - 1} e^{-\beta\lambda}}{\Gamma(\alpha)} \quad \forall \lambda \geq 0 \quad \alpha, \beta > 0$$

$P(\lambda | D) \propto P(D | \lambda) P(\Lambda)$ - Slides

a)

$$\log P(\lambda | D) \propto \underbrace{\log P(D | \lambda)}_{\substack{\text{Poisson} \\ \text{Problem 1}}} + \log \underbrace{P(\lambda)}_{\substack{\text{Gamma} \\ \text{Given before}}}$$

$$\log P(D | \lambda) = \sum_{i=1}^{N} (x_i \log \lambda - \lambda)$$

$$\log P(\lambda) = \frac{\beta^{\alpha} \lambda^{\alpha - 1} e^{-\beta\lambda}}{\Gamma(\alpha)}$$

$$\log P(\lambda) = (\alpha - 1) \log \lambda - \beta\lambda$$

$$\log P(\lambda | D) \propto \sum_{i=1}^{N} (x_i \log \lambda - \lambda) + (\alpha - 1) \log \lambda - \beta\lambda$$

$$\log P(\lambda | D) \propto \left(\sum_{i=1}^{N} x_i + \alpha - 1\right) \log \lambda - (N + \beta)\lambda$$

$$\log P(\lambda|D) \propto \left(\sum_{i=1}^{N} x_i + \alpha - 1\right) \log \lambda - (N+B)\lambda$$

b)

$$\log P(\lambda|D) = \left(\sum_{i=1}^{N} x_i + \alpha - 1\right) \log \lambda - (N+\beta)\lambda$$

$$\frac{d}{d\lambda}\left(\left(\sum_{i=1}^{N} x_i + \alpha - 1\right) \log \lambda\right)$$

chain rule

$$\frac{\sum_{i=1}^{N} x_i + \alpha - 1}{\lambda}$$

$$\frac{d}{d\lambda}(-(N+\beta)\lambda) \quad -(N+\beta) = 0$$

$$\boxed{\frac{d}{d\lambda} \log P(\lambda|D) = \frac{\sum_{i=1}^{N} x_i + \alpha - 1}{\lambda} - (N+\beta)}$$

c)

$$\frac{\sum_{i=1}^{N} x_i + \alpha - 1}{\lambda} = (N+\beta)$$

$$\sum_{i=1}^{N} x_i + \alpha - 1 = (N+\beta)\lambda$$

$$\frac{\sum_{i=1}^{N} x_i + \alpha - 1}{(N+\beta)} = \lambda$$

$$\boxed{\hat{\lambda}_{MAP} = \frac{\sum_{i=1}^{N} x_i + \alpha - 1}{(N+\beta)}}$$

Problem 3)

$$P(D|\lambda) = \prod_{i=1}^{N} P(x_i|\lambda)$$

$$= \prod_{i=1}^{N} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!}$$

$$= \lambda^{\sum_{i=1}^{N} x_i} e^{-N\lambda} \overbrace{\prod_{i=1}^{N} \frac{1}{x_i!}}^{\text{Constant}}$$

$$P(D|\lambda) \propto \lambda^{\sum_{i=1}^{N} x_i} e^{-N\lambda}$$

ok $P(D|\lambda)$ done now

$$P(\lambda) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta\lambda}$$

$$P(\lambda) \propto \lambda^{a-1} e^{-\beta\lambda}$$

now $P(\lambda|D)$

$$P(\lambda|D) \propto P(D|\lambda)P(\lambda)$$

$$P(\lambda|D) \propto \left(\lambda^{\sum_{i=1}^{N} x_i} e^{-N\lambda}\right)\left(\lambda^{\alpha-1} e^{-\beta\lambda}\right)$$

$$\left(\lambda^{\sum_{i=1}^{N} x_i} e^{-N\lambda}\right)\left(\lambda^{\alpha-1}\right) = \lambda^{\sum_{i=1}^{N} x_i + \alpha - 1}$$

$$\left(e^{-N\lambda}\right)\left(e^{-\beta\lambda}\right) = e^{-(N+B)\lambda}$$

$$P(\lambda|D) \propto \lambda^{\sum_{i=1}^{N} x_i + a - 1} e^{-(N+B)\lambda}$$

general form gamma

$$\text{gamma}(\lambda; \alpha', \beta') \propto \lambda^{\alpha'-1} e^{-\beta'\lambda}$$

$$\alpha' = \sum_{i=1}^{N} x_i + a$$

$$\beta' = N + B$$

still a posterior distribution

Part 2

Problem 4)

In the use of one-hot encoding technique aka the use of 0's and 1's to see if a attribute is met we would see some change to the knn. This change to knn would mostly stem from the idea that this idea views the category's as un sorted meaning, each category is unrelated. This means that when taking the Euclidean distance between any two categories it will be consistent and only reflect that they are different. On the other side with Ordinal encoding we see a "sorted" method where we have an ordering between category's. This means that when taking the Euclidean distance differs from each category setup. When using this method its important to think about the possibility that these weights may not be as connected as we think resulting in incorrect neighbors being selected.

Problem 5

$\frac{6033}{8000} \div 75.41\%$

$\frac{1967}{8000} \approx 24.59\%$

The number of data points making less then 50k is 6,033 and the number making more is 1,967. This comes out 75.41% making less then 50k and 24.59% making more. When thinking about this relative to the model I do think that there will be a significant higher skew to those making lower then 50k.

In retrospect to a model that produces 70% accuracy it really depends on the amount of data we are training on here with 8000 points of data I do think that good or ok due to each point of data having 87 total dimensions within it.

Problem 6

$$\|x\|_2 = \sqrt{\sum_{i=1}^{d} x_i^2}$$

$x = (x_1, x_2, \dots, x_d)$ and $y = (y_1, y_2, \dots, y_d)$

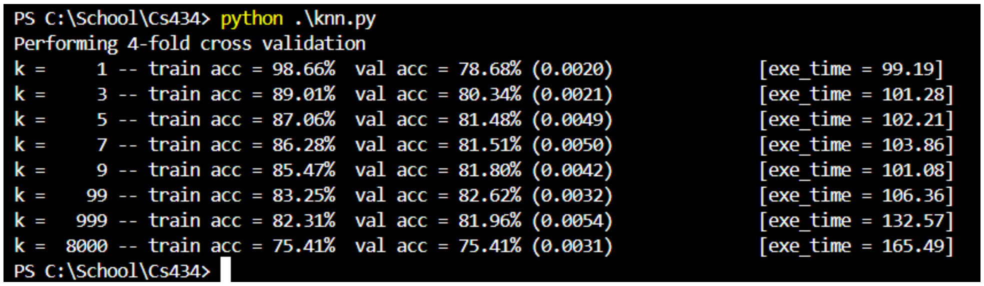
$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_d - y_d)^2}$$

$$= \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$$

$$\|x - y\|_2 = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$$

it can we written as L_2 form

Problem 9)



```
PS C:\School\Cs434> python .\knn.py
Performing 4-fold cross validation
k =     1 -- train acc = 98.66%  val acc = 78.68% (0.0020)          [exe_time = 99.19]
k =     3 -- train acc = 89.01%  val acc = 80.34% (0.0021)          [exe_time = 101.28]
k =     5 -- train acc = 87.06%  val acc = 81.48% (0.0049)          [exe_time = 102.21]
k =     7 -- train acc = 86.28%  val acc = 81.51% (0.0050)          [exe_time = 103.86]
k =     9 -- train acc = 85.47%  val acc = 81.80% (0.0042)          [exe_time = 101.08]
k =    99 -- train acc = 83.25%  val acc = 82.62% (0.0032)          [exe_time = 106.36]
k =   999 -- train acc = 82.31%  val acc = 81.96% (0.0054)          [exe_time = 132.57]
k =  8000 -- train acc = 75.41%  val acc = 75.41% (0.0031)          [exe_time = 165.49]
PS C:\School\Cs434>
```

K = 99 gives me the best result that I observed. As for why k = 1 is a 0% training error it is because its like memorizing the training model and its practicly only learning for that own case. This would lead to overfitting and does not capture the true nature of the model. I see that in general as we reach higher levels of k untill we get to 99 we are seeing an increase to train acc and val acc but when we go beyond 99 we start to see a slight drop in val acc and train acc.

This is because overfitting happens at low values of k and underfitting occurs at high values of k.

Problem 10)

The only thing I really changed was K I did mess a little with drawing and k nearest weights a bit but reverted them back after playing around. In the start of running the code I noticed that 99 we the best performance so I went and did 150 and submitted it alongside 99 to kaggle. Then later went somewhere in the middle for 121 where it was a slight increase to the score of 0.01 increase form 99

Debriefing

1) Approximately how long did you spend on this assignment?
   - 10 hours
2) Would you rate it as easy, moderate, or difficult?
   - moderate
3) Did you work on it mostly alone or did you discuss the problem with others?
   - I did most of it alone, I asked one question to a friend when I got 60% on k=1 and he just gave me a hit that the TA gave him when he was stuck on the same problem.
4) How deeply do you feel you understand the material it covers (0%-100%)
   - 90%
5) Any other comments?