1 Statistical Estimation

Problem 1
Q1 Maximum Likelihood Estimation of λ

Solution
Steps:

1. Write out the log-likelihood function logP(D| λ)
   (a) $P(D \mid \lambda) = \prod_{i=1}^{n} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!}$
   (b) $log(P(D \mid \lambda)) = \sum_{i=1}^{n} log(\frac{\lambda^{x_i} e^{-\lambda}}{x_i!})$
   (c) $= \sum_{i=1}^{n} log(\lambda^{x_i}) - log(x_i!) + log(e^{-\lambda})$
   (d) $= \sum_{i=1}^{n} x_i log(\lambda) - log(x_i!) - \lambda$
   (e) $= log(\lambda) \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} log(x_i!) - n\lambda$
   (f) $\boldsymbol{Answer}$: $log(\lambda) \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} log(x_i!) - n\lambda$
2. Take the derivative of the log-likelihood with respect to the parameter λ
   (a) $\frac{d}{d\lambda}(log(\lambda) \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} log(x_i!) - n\lambda)$
   (b) $\frac{1}{\lambda} \sum_{i=1}^{n} x_i - n$
   (c) $\boldsymbol{Answer}$: $\frac{1}{\lambda} \sum_{i=1}^{n} x_i - n$
3. Set the derivative equal to zero and solve for $\lambda-$call this maximizing value $\hat{\lambda}_{MLE}$
   (a) $\frac{1}{\lambda} \sum_{i=1}^{n} x_i - n = 0$
   (b) $\boldsymbol{Answer}$: $\hat{\lambda}_{MLE} = \frac{1}{n} \sum_{i=1}^{n} x_i$

Problem 2
Q2 Maximum A Posteriori Estimate of λ with a Gamma Prior

Solution
Steps:

1. Write out the log-posterior $logP(\lambda \mid D) \propto logP(D \mid \lambda) + logP(\lambda)$
   (a) $logP(\lambda \mid D) \propto logP(D \mid \lambda) + logP(\lambda)$
   (b) $logP(D \mid \lambda) = log(\lambda) \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} log(x_i!) - n\lambda$
   (c) $logP(\lambda) = log(\frac{\beta^{\alpha} \lambda^{\alpha-1} e^{-\beta\lambda}}{\Gamma(\alpha)})$
   (d) $= \alpha log(\beta) + (a-1)log(\lambda) - \beta\lambda - log(\Gamma(\alpha))$
   (e) $\boldsymbol{Answer}$: $logP(D \mid \lambda) = log(\lambda) \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} log(x_i!) - n\lambda + \alpha log(\beta) + (a-1)log(\lambda) - \beta\lambda - log(\Gamma(\alpha))$
2. Take the derivative of the $logP(D \mid \lambda) + logP(\lambda)$ with respect to λ
   (a) $\frac{d}{d\lambda}(log(\lambda) \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} log(x_i!) - n\lambda + \alpha log(\beta) + (a-1)log(\lambda) - \beta\lambda - log(\Gamma(\alpha)))$
   (b) $= \frac{1}{\lambda} \sum_{i=1}^{n} x_i - n + \frac{(a-1)}{\lambda} - \beta$
   (c) $\boldsymbol{Answer}$: $\frac{1}{\lambda} \sum_{i=1}^{n} x_i - n + \frac{(a-1)}{\lambda} - \beta$
3. Set the derivative equal to zero and solve for $\lambda-$call this maximizing value $\hat{\lambda}_{MAP}$
   (a) $\frac{1}{\lambda} \sum_{i=1}^{n} x_i - n + \frac{(a-1)}{\lambda} - \beta = 0$

(b) $\frac{\sum_{i=1}^n x_i}{\lambda} + \frac{(a-1)}{\lambda} - \beta - n = 0$

(c) $\frac{\sum_{i=1}^n x_i + (a-1)}{\lambda} - (\beta + n) = 0$

(d) $\frac{\sum_{i=1}^n x_i + (a-1)}{\lambda} = (\beta + n)$

(e) $\lambda = \frac{\sum_{i=1}^n x_i + (a-1)}{(\beta+n)}$

(f) ***Answer***: $\hat{\lambda}_{MAP} = \frac{\sum_{i=1}^n x_i + (a-1)}{(\beta+n)}$

Problem 3
Q3 Deriving the Posterior of A Poisson-Gamma Model

Solution
Steps:

1. $P(\lambda \mid D) = \frac{P(D|\lambda)P(\lambda)}{P(D)} \propto P(D \mid \lambda)P(\lambda)$
2. $P(\lambda) = \frac{\beta^\alpha \lambda^{\alpha-1} e^{-\beta\lambda}}{\Gamma(\alpha)} \propto \lambda^{\alpha-1} e^{-\beta\lambda}$
3. $P(D \mid \lambda) = \prod_{i=1}^N \frac{\lambda^{x_i} e^{-\lambda}}{x_i!} = \frac{\lambda^{\sum_{i=1}^N x_i} e^{-N\lambda}}{\prod_{i=1}^N x_i!} \propto \lambda^{\sum_{i=1}^N x_i} e^{-N\lambda}$
4. $P(D \mid \lambda)P(\lambda) = (\lambda^{\sum_{i=1}^N x_i} e^{-N\lambda})(\lambda^{\alpha-1} e^{-\beta\lambda})$
5. $= \lambda^{\sum_{i=1}^N x_i + a - 1} e^{-(\beta+N)\lambda}$, which is approximately equal to gamma($\sum_{i=1}^N x_i + a, \beta + N$)
6. ***Answer***: $P(\lambda \mid D) \propto$ gamma($\sum_{i=1}^N x_i + a, \beta + N$)

2 k-Nearest Neighbor (kNN)

Problem 4
Q4 Encodings and Distance

Solution
Steps:

1. ***Answer***: Using the naive preprocessing method, the euclidean distance between a query and a sample could turn out to be very large, which could lead to improper weights. Moreover, the distance between a query and sample could result in unwanted bias due to the ordering of the categorical variables. Using the one-hot encoding method, the weights are more equalized, giving a distance of either 0 or 1 depending on whether the query's category matches the sample category.

Problem 5
Q5 Looking at Data

Solution
Steps:

1. ***Answer***: About 75.4 percent has income less than or equal to 50K and about 24.5 percent has income more than 50K. Since there are more sample data points that result in an income less than or equal to 50K, newer data points may be skewed towards having an income of less than or equal to 50K. If the distribution of data points with income less than or equal to 50K and those greater than 50K do not approximate that in real life, then the model may turn out to be inaccurate. A model that has achieved 70 percent accuracy is a good or poor model depending on how well the training data matches the population. Each data point has 85 dimensions not counting id and the label. This is from adding together the number of numerical and ordinal attributes and the number of one hot encoded variables.

Problem 6
Q6 Norms and Distances

Solution
Steps:

1. ***Answer***: $\|x\|_2 = \sqrt{\sum_{i=1}^{d}(z_i - x_i)^2}$

Problem 7
Q7 Implement kNN Classifier

Solution
Steps:

1. ***Answer***: Complete

Problem 8
Q8 Implement k-fold Cross Validation

Solution
Steps:

1. ***Answer***: Complete

Problem 9
Q9 Hyperparameter Search

Solution
Steps:

1. ***Answer***: The best number of neighbors (k) for train accuracy is when k = 1 while the best for validation accuracy is when k = 99. When k = 1, training error should be 0 percent because the closest neighbor to a point is itself. The trends I noticed with train and cross-validation accuracy rate was that as k increased, the train accuracy decreased while the cross-validation accuracy increased. This meant that as the k increased, the model went from being overfitted to a better fit. The lower train accuracy indicates lower details/noise were picked up in the training data, improving fit, and the higher validation accuracy indicates that the model's ability to generalize new data went up.

Problem 10
Q10 Kaggle Submission

Solution
Steps:

1. ***Answer***: For my modifications, I changed the K nearest neighbors to a weighted K nearest neighbors algorithm. I added weights to the sample neighbors based on their distance to the query so that neighbors that were closer to the query had more weight. I also set k to 99 since the validation accuracy did not improve beyond it.

3 Debriefing

Problem 1
Approximately how many hours did you spend on this assignment?

Solution

1. About 17 hours.

Problem 2
Would you rate it as easy, moderate, or difficult?

Solution

1. Moderate-Difficult.

Problem 3
Did you work on it mostly alone or did you discuss the problems with others?

Solution

1. Worked on it alone.

Problem 4
How deeply do you feel you understand the material it covers (0%–100%)?

Solution

1. I felt I understood about 85 percent of the material covered.

Problem 4
Any other comments?

Solution

1. None.