CSCI 567, Spring 2021
Sirisha Rambhatla

Quiz 2

Date: March 03, 2021
Duration: 2 hours
Total scores: 100 points

Instructions

Notes on notation:

- Unless stated otherwise, scalars are denoted by small letter in normal font, vectors are denoted by small letters in bold font and matrices are denoted by capital letters in bold font.
- $\|.\|$ means L2-norm unless specified otherwise i.e. $\|.\| = \|.\|_2$

Problem 1 Single Choice Questions (10 points)

Note that there is only ONE correct answer to each question.

1. When applying a GMM of K components to a dataset of N points, if we represent $\gamma_{nk} = p(z_n = k|\boldsymbol{x}_n)$ (which is the term used to update component weights) as a matrix of N rows and K columns, what is the sum of this matrix?

   (A) NK

   (B) N

   (C) K

   (D) 1

   Solution: B, $\sum_k \gamma_{n,k} = \sum_k p(z_n = k|\boldsymbol{x}_n) = 1$

2. Which of the following statements is NOT TRUE about Support Vector Machine (SVM)?

   (A) For two dimensional data points, the decision boundary learned by any SVM will be a straight line.

   (B) In theory, a Gaussian kernel SVM can model any complex decision boundary.

   (C) SVM is robust to overfitting.

   (D) SVM cannot be solved in primal form when the kernel function is infinite dimensional

   Solution: A

   A: note that we can apply kernel to obtain non-linear decision boundary.

   B: a Gaussian kernel is infinite dimensional

   D: note that in primal form you need to learn parameters $w \in \mathbb{R}^d$ explicitly, which has the same dimension as the kernel feature.

3. Which of the following statements is TRUE?

   (A) Given a dataset, the optimal parameters of Hidden Markov Model (HMM) can be solved in closed form.

   (B) The value of $\sum_s \alpha_s(t)\beta_s(t)$ is independent of t, where $\alpha_s(t)$ is forward message, $\beta_s(t)$ is backward message.

   (C) We can learn a HMM using the forward algorithm.

   (D) HMM cannot be learnt by the EM algorithm.

Solution: B

A: there is no closed form solution, we need EM to solve it.

B: $P(X_{1:T} = x_{1:T}) = \sum_s \alpha_s(t)\beta_s(t)$ for any t.

4. Vovk's real polynomial kernel $k : \mathbb{R}^D \times \mathbb{R}^D \leftarrow \mathbb{R}$ is defined as: $k(\boldsymbol{x}, \boldsymbol{x}') = \frac{1-(\boldsymbol{x}^\top \boldsymbol{x}')^p}{1-(\boldsymbol{x}^\top \boldsymbol{x}')}$, where p is a non-negative integer. Which of the following is the corresponding non-linear mapping for this kernel when $D = 2$ and $p = 3$?

(A) $\phi(\boldsymbol{x}) = [x_1^2 x_2^2, 2x_1x_2, x_1, x_2, 1]^\top$

(B) $\phi(\boldsymbol{x}) = [x_1^2, x_2^2, 2x_1x_2, x_1, x_2, 1]^\top$

(C) $\phi(\boldsymbol{x}) = [x_1^2, x_2^2, \sqrt{2}x_1x_2, x_1, x_2, 1]^\top$

(D) $\phi(\boldsymbol{x}) = [x_1^2 x_2^2, \sqrt{2}x_1x_2, x_1, x_2, 1]^\top$

Solution: C

$$\begin{aligned}\frac{1-(\boldsymbol{x}^\top \boldsymbol{x}')^3}{1-\boldsymbol{x}^\top \boldsymbol{x}'} &= 1 + \boldsymbol{x}^\top \boldsymbol{x}' + (\boldsymbol{x}^\top \boldsymbol{x}')^2 \\ &= 1 + x_1x_1' + x_2x_2' + (x_1x_1' + x_2x_2')^2 \\ &= 1 + x_1x_1' + x_2x_2' + (x_1x_1')^2 + 2x_1x_1'x_2x_2' + (x_2x_2')^2\end{aligned}$$

5. Which of the following statements is TRUE?

(A) because decision trees learn to classify discrete valued outputs, it is impossible for them to overfit.

(B) given a dataset for binary classification, k-Nearest Neighbors (kNN) with large k tends to have a smoother decision boundary (assuming $N >> k$).

(C) a low training error and a high testing error is a good sign of underfitting.

(D) $R(\boldsymbol{w}) = \exp(-\sum_i w_i^2)$ is a good penalty function to regularize model complexity.

Solution: B

C: a sign of overfitting.

D: it does not penalize large model weight.

Problem 2 Multiple Choice Questions (10 points)

Note that there are ONE or MORE correct answers to each question.

1. Which of the followings statements are TRUE?

(A) The Adaboost algorithm will eventually give zero training error regardless of the type of weak classifier it uses, provided enough iterations are performed.

(B) Boosting algorithm will not select the same weak classifier more than once.

(C) In the Adaboost algorithm, weights of the misclassified examples goes up.

(D) The Adaboost algorithm can be treated as greedily minimizing the exponential loss

Solution: C, D

2. Which of the followings are NOT kernel functions?

   (A) $k(x, x') = -\|x - x'\|_2^2$

   (B) $k(x, x') = \|x - x'\|_2^2$

   (C) $k(x, x') = e^{\|x-x'\|_2^2}$

   (D) $k(x, x') = e^{-\|x-x'\|_2^2}$

   Solution: A, B, C

   We can construct Gram matrix of $x_1 = 0, x_2 = 1$ to disprove the first three items. The last one is Radial basis function kernel.

3. Which of the following statements about Principle Component Analysis (PCA) are true?

   (A) PCA can be used to visualize data, compress data, or de-noise data.

   (B) PCA can give a compressed dataset that is a linear transformation of the original dataset.

   (C) PCA requires computing eigenvalues and eigenvectors of the covariance matrix.

   (D) The first step of PCA is to center the original dataset.

   Solution: A, B, C, D

4. Which of the following procedures can be done without labels?

   (A) S-fold cross validation

   (B) PCA

   (C) Naive Bayes

   (D) Learning a GMM with Expectation-Maximization (EM)

   Solution: B, D

   A: you need label to compute validation error.

   C: it models the joint probability of data point and label.

   D: GMM is a clustering algorithm, no label is needed.

5. Which of the followings are TRUE?

   (A) Pre-centering data is necessary for kernel PCA.

   (B) In HMM, the first state of the most likely hidden path is the same as the most likely first state.

   (C) Density estimation is a supervised learning algorithm.

   (D) Density estimation is an important component of the naive Bayes algorithm.

   Solution: D

   A: centered data does not necessary give centered feature matrix.

   B: it is possible that many small probability paths share the same first state and thus their first state has dominated probability.

Problem 3 Clustering and Naive Bayes (20 points)

1. Consider the dataset given in Table 1with features, Humidity $\in$ {High, Low}, Outlook $\in$ {Yes, No} and Windy in $\in$ {Yes, No}. The class label is given by the attribute **Play** $\in$ {True, False}. Build a Naive Bayes classifier using the dataset table, to classify the test example with: Humidity = High, Outlook = Yes, Windy = No (Show your working and write the predicted label for Play on the test example). **(6 points)**

Table 1: Dataset table.

Humidity	Outlook	Windy	**Play**
L	N	N	T
L	N	Y	T
H	Y	N	T
H	Y	Y	F
L	Y	N	F
L	Y	Y	T
H	N	N	F

Solution:

$$\hat{y} = \arg\max_c P(y = c|x) \propto \arg\max_c P(y = c, x) = \arg\max_c P(y = c)P(x|y = c)$$

The MLE estimate using the dataset table for the probabilities are as follows,

$$P(y = T) = \frac{4}{7} \quad P(y = F) = \frac{3}{7}$$

Based on the Naive assumption, features are assumed to be conditionally independent given the label.

$$P(x|y = c) = \prod_{i=1}^{D} p(x_d|y = c)$$

Therefore,

$$\begin{aligned} &P(y = T|Humidity = H, Outlook = Y, Windy = N) \\ &\propto P(y = T)P(Humidity = H|y = T)P(Outlook = Y|y = T)P(Windy = N|y = T) \\ &= \frac{4}{7} \times \frac{1}{4} \times \frac{1}{2} \times \frac{1}{2} = \frac{1}{28} = 0.036 \end{aligned}$$

$$\begin{aligned} &P(y = F|Humidity = H, Outlook = Y, Windy = N) \\ &\propto P(y = F)P(Humidity = H|y = F)P(Outlook = Y|y = F)P(Windy = N|y = F) \\ &= \frac{3}{7} \times \frac{2}{3} \times \frac{2}{3} \times \frac{2}{3} = \frac{8}{63} = 0.127 \end{aligned}$$

Since $0.127 > 0.036$, the predicted class with arg max of the joint probabilities is Play = F (False).

Rubric:

- 1 pt for using the Bayes Rule in $\arg\max_c P(y = c|x)$.

- 1 pt for $P(y = T) = \frac{4}{7}$ or $P(y = F) = \frac{3}{7}$.
- 1 pt for applying Naive Bayes assumption of conditional independence.
- 1 pt for correct calculation of conditional probabilities for $P(y = T|H, Y, N) \propto 0.036$ case.
- 1 pt for correct calculation of $P(y = F|H, Y, N) \propto 0.127$ case.
- 1 pt for final answer of predicted class as $\hat{y} = F$.

2. For the same dataset above, construct a decision stump classifier (tree with depth 1, i.e., root with 2 children). Prediction for each leaf is the majority class of examples belonging to the leaf (break ties arbitrarily). Find the best decision stump, by selecting which feature to split on at the root based on the *quality* of the constructed decision stump (in terms of weighted classification error) . **(6 points)**

   Solution:

   T1 (Humidity): Weighted Classification error = $\frac{1}{4} \times \frac{4}{7} + \frac{1}{3} \times \frac{3}{7} = \frac{2}{7}$
   Left Child ("Low") = TTT F
   Right Child ("High") = T FF

   T2 (Outlook): Weighted Classification error = $\frac{1}{2} \times \frac{4}{7} + \frac{1}{3} \times \frac{3}{7} = \frac{3}{7}$
   Left Child ("Yes") = TT FF
   Right Child ("No") = TT F

   T3 (Windy): Weighted Classification error = $\frac{1}{3} \times \frac{3}{7} + \frac{1}{2} \times \frac{4}{7} = \frac{3}{7}$
   Left Child ("Yes") = TT F
   Right Child ("No") = TT FF

   Best decision stump splits on Humidity feature at the root (T1).

   Rubric:

   - 3 pt for weighted classification error answer (1 pt per stump)
   - 3 pt for distribution of labels at each leaf (1 pt per stump, if both leaves are correct)

3. **Clustering** Given a set of data points $\{\boldsymbol{x}_n\}_{n=1}^N \in \mathbb{R}^d$, K-means minimizes the distortion objective given below, where $\mu_k \in \mathbb{R}^d$ is the representative of the k-th cluster and γ_{nk} is a binary indicator which is 1 if $\mathbf{x}_n$ is assigned to the cluster k and 0 otherwise.

$$D = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{nk} \|\mathbf{x}_n - \mu_k\|_2^2$$

   (a) K-means. Assuming all γ_{nk} are known (i.e., assignments of all N points), show that the objective D is minimized when μ_k is the mean of all data points assigned to cluster k. **(4 points)**

   Solution: Setting derivative to 0, we get the denominator is number of examples which belong to cluster k and the numerator is sum of examples that belong to cluster k.

$$\frac{dD}{d\mu_k} = \sum_{n=1}^{N} -2r_{nk}(x_n - \mu_k) = 0$$

$$\implies \mu_k = \frac{\sum_{n=1}^{N} r_{nk} x_n}{\sum_{n=1}^{N} r_{nk}}$$

   Rubric:

- 1 pt for solving by setting derivative to 0.
- 2 pt for correct derivative.
- 1 pt for correct answer for μ_k.

(b) K-medioids. If we change the distortion objective to use the L_1 norm instead ($||\boldsymbol{z}||_1 = \sum_d |z_d|$), then with all γ_{nk} known, show that μ_k which minimizes D' is the element-wise median of all data points assigned to the k-th cluster (The element-wise median of a set of vectors is defined as a vector whose j-th element is the median of j-th elements of the vectors in the set). **(4 points)**

$$D' = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{nk} \|\boldsymbol{x}_n - \mu_k\|_1$$

Solution:

$$\frac{dD'}{d\mu_{kj}} = \frac{d\left(\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{j=1}^{d}\gamma_{nk}|\mathbf{x}_{nj} - \mu_{kj}|\right)}{d\mu_{kj}}$$
$$= \sum_{n=1}^{N} -\gamma_{nk} sgn(x_{nj} - \mu_{kj}) = 0$$

The gradient term will only have x_n that belong to k^{th} cluster for μ_k. Assume there are odd number of samples in the k^{th} cluster, lets say there are total N_k samples and all x_{nj} are sorted. Observe that gradient will be zero when $\mu_{kj} = x_{\frac{N_k}{2}j}$ i.e the median. Similarly for even N_k, we can see gradient will be zero if $x_{\frac{N_k}{2}j} < \mu_{kj} < x_{\left(\frac{N_k}{2}+1\right)j}$ which is the median.

Rubric:

- 2 pt for fully correctly computed derivative (no partial pt).
- 2 pt for correct reasoning about the element-wise median from the derivative (no partial pt).

Problem 4 PCA and Kernel PCA **(20 points)**

1. In the lecture, to find principal components, we find the direction $\mathbf{v} \in \mathbb{R}^D$ with $\|\mathbf{v}\|_2 = 1$ such that the projection $(\mathbf{x}_i^\top \mathbf{v})$ of the dataset on this direction has the most variance. Prove that is equivalent to finding direction $\mathbf{v}$ such that the projected vector minimizes the reconstruction error, given as follows, assuming the data is already centered. **(6 points)**

$$\min_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{v})\mathbf{v}\|_2^2$$

Solution:

$$\min_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{v})\mathbf{v})^\top(\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{v})\mathbf{v})$$

$$= \min_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i^\top - (\mathbf{x}_i^\top \mathbf{v})\mathbf{v}^\top)(\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{v})\mathbf{v})$$

$$= \min_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i^\top \mathbf{x}_i - 2(\mathbf{x}_i^\top \mathbf{v})^2 + (\mathbf{x}_i^\top \mathbf{v})^2\|\mathbf{v}\|^2)$$

$$= \min_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i^\top \mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{v})^2) = \min_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i^\top \mathbf{x}_i - \mathbf{v}^\top \mathbf{x}_i\mathbf{x}_i^\top \mathbf{v})$$

$$=> \max_{\mathbf{v}:\|\mathbf{v}\|=1} \mathbf{v}^T(\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^\top)\mathbf{v} = \max_{\mathbf{v}:\|\mathbf{v}\|=1} \mathbf{v}^T(\mathbf{X}^\top\mathbf{X})\mathbf{v}$$

Rubric:

- 2 pt for simplifying L2 norm to vector products.
- 2 pt for using the fact that $\mathbf{v}$ has unit norm.
- 2 pt for the final step, eliminating the first term dependent only on $\mathbf{x}_i$.

2. **Kernal PCA.** Consider a dataset of 2 examples with 2 features (f1, f2) below. For kernel PCA on the dataset, assume a polynomial kernel $k(x, x') = x^\top x'$. Compute the Gram/kernel matrix K and find the top eigenvector of the centered Gram matrix with appropriate scaling. **(14 points)**

Example	f1	f2
1	10	2
2	6	0

Solution:

- Gram/kernel matrix with the given polynomial kernel,

$$K = \begin{bmatrix} 10\times 10 + 2\times 2 & 10\times 6 + 2\times 0 \\ 10\times 6 + 2\times 0 & 6\times 6 + 0\times 0 \end{bmatrix} = \begin{bmatrix} 104 & 60 \\ 60 & 36 \end{bmatrix}$$

- Centered Gram matrix $\bar{K}$

$$\bar{K} = K - EK - KE + EKE$$
$$= \begin{bmatrix} 104 & 60 \\ 60 & 36 \end{bmatrix} - \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}\begin{bmatrix} 104 & 60 \\ 60 & 36 \end{bmatrix} - \begin{bmatrix} 104 & 60 \\ 60 & 36 \end{bmatrix}\begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} + \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}\begin{bmatrix} 104 & 60 \\ 60 & 36 \end{bmatrix}\begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}$$
$$= \begin{bmatrix} 104 & 60 \\ 60 & 36 \end{bmatrix} - \begin{bmatrix} 82 & 48 \\ 82 & 48 \end{bmatrix} - \begin{bmatrix} 82 & 82 \\ 48 & 48 \end{bmatrix} + \begin{bmatrix} 65 & 65 \\ 65 & 65 \end{bmatrix} = \begin{bmatrix} 5 & -5 \\ -5 & 5 \end{bmatrix}$$

- Solving for eigenvalues $\bar{K}\alpha = \lambda\alpha$, so $|\bar{K} - \lambda I| = 0$ since it has to be singular for $\alpha \neq 0$.

$$(5-\lambda)^2 - 25 = 0 \quad i.e. \quad \lambda_1 = 10, \lambda_2 = 0.$$

- (Top) Eigenvalues are 10 and 0 respectively. The eigenvectors corresponding to them after appropriate scaling are the 1st and 2nd principal components respectively (many possible answers). Note: Scale α so that its L2-norm is $1/\sqrt{\lambda}$,

$$\alpha_1 = [\frac{1}{\sqrt{20}}, \frac{-1}{\sqrt{20}}]^\top$$

Rubric:

- 1 pt for Technique of K (pt awarded if correctly computed K directly written).
- 1 pt for Answer of K.
- 2 pt for Technique of $\bar{K}$ (pt awarded if correctly computed $\bar{K}$ directly written).
- 2 pt for Answer of $\bar{K}$ (pts deducted even if mistake is due to wrong K values).
- 2 pt for Technique to solve for eigenvalues to get eigenvectors ($Av = \lambda v$) or directly det = 0.
- 1 pt for Technique to calculate determinant (the determinant should be of $\bar{K} - \lambda I$).
- 1 pt for Answer of $\lambda_1 = 10$ (need not mention $\lambda_2 = 0$).
- 1 pt for Noting the 1st PC as for the larger eigenvalue (even if the values calculated are wrong).
- 1 pt for Technique of finding eigenvector ($\bar{K}v = \lambda_i v$). No pts assigned for computed answers since many possible correct answers, and answers can also differ due to mistakes in $\bar{K}, \lambda$.
- 2 pt for Appropriate scaling (ignore errors due to previous calculations).

Problem 5 Expectation maximization algorithm (20 points)

Consider the following generative mixture model for integers $x \geq 0$,

$$P(x) = \begin{cases} \pi(1) + (1-\pi)e^{-\lambda} & \text{if } x = 0 \\ \pi(0) + (1-\pi)Poisson(x;\lambda) & \text{if } x > 0, \end{cases}$$

i.e. x is generated from a mixture of two different types of distributions (Poisson with parameter $\lambda > 0$, and singleton distribution that always assigns value 0, with probability 1). **Note** $Poisson(x;\lambda) = \frac{\lambda^x e^{-\lambda}}{x!} \quad \forall x \geq 0$.

That means, under the given mixture model, we can assume a latent binary variable z drawn from Bernoulli distribution with parameter $\pi \in (0,1)$, that is, $p(z=1;\pi) = \pi$ and $p(z=0;\pi) = 1-\pi$. Then if $z = 1$, x is drawn from the singleton distribution, otherwise if $z = 0$, x is drawn from the Poisson distribution.

1. Write down the posterior distribution $P(z|x;\lambda,\pi)$ for the four cases of values of x, z (i) $z = 1, x = 0$, (ii) $z = 0, x = 0$, (iii) $z = 1, x > 0$ and (iv) $z = 0, x > 0$. **(4 points)**

   Solution: We can compute the posterior as follows,

$$P(z|x;\lambda,\pi) = \frac{P(x,z;\lambda,\pi)}{P(x;\lambda,\pi)}$$

We have 4 cases:

$$P(z=1|x=0;\lambda,\pi) = \frac{\pi}{\pi+(1-\pi)e^{-\lambda}}$$
$$P(z=0|x=0;\lambda,\pi) = \frac{(1-\pi)e^{-\lambda}}{\pi+(1-\pi)e^{-\lambda}}$$
$$P(z=1|x>0;\lambda,\pi) = 0$$
$$P(z=0|x>0;\lambda,\pi) = \frac{(1-\pi)Poisson(x;\lambda)}{(1-\pi)Poisson(x;\lambda)} = 1$$

Rubric:

- 4 pt of exact correct answers matched to correct cases (1 pt per case).

2. Express the log-likelihood of a dataset of N i.i.d samples $\{x_i\}_{i=1}^N$ for $x_i = 0$ or $x_i > 0$ as observed. *Note:* It is expected to correctly plug in P(x) given in the question for each range of values of x_i, in the equation for the dataset log-likelihood (by splitting the sum or using an indicator function). **(3 points)**

Solution:

$$\log\prod_{i=1}^{N} P(x_i;\lambda,\pi) = \sum_{i=1}^{N}\log P(x_i;\lambda,\pi) = \sum_{i:x_i=0}\log P(x_i=0;\lambda,\pi) + \sum_{i:x_i>0}\log P(x_i>0;\lambda,\pi)$$
$$= \sum_{i:x_i=0}\log(\pi+(1-\pi)e^{-\lambda}) + \sum_{i:x_i>0}\log(1-\pi)\frac{\lambda^{x_i}e^{-\lambda}}{x_i!}$$

Rubric:

- 1 pt for taking the sum of log probabilities of the N samples (or leaving all terms in log product form is acceptable too).
- 1 pt for using the indicator or splitting the sum to plug in the form.
- 1 pt for simplifying expression and plug in specific value (writing x_i instead of x). Last term left as $Poisson(x_i;\lambda)$ instead is acceptable.

3. Similarly, write down the complete log-likelihood function $\log\left(\prod_{i=1}^N P(x_i,z_i;\lambda,\pi)\right)$. Show that it can be simplified to the following form, **(4 points)**

$$\sum_{i:x_i>0}(\log(1-\pi)+x_i\log\lambda-\lambda-\log x_i!) + \sum_{i:x_i=0}(z_i\log\pi+(1-z_i)\log(1-\pi)-\lambda(1-z_i))$$

Solution: The joint distribution of x,z is:

$$P(x,z;\lambda,\pi) = \begin{cases}\pi, & \text{if } z=1, x=0\\ 0, & \text{if } z=1, x>0\\ (1-\pi)Poisson_\lambda(x), & \text{if } z=0, x\geq 0\end{cases}$$

Since the $z = 1, x > 0$ case has a zero probability and can never be observed in collected data, the complete log-likelihood for the given dataset is:

$$\begin{aligned}\log\left(\prod_{i=1}^{N} P(x_i, z_i; \lambda, \pi)\right) &= \log\left[\prod_{i:x_i>0}(1-\pi)\frac{\lambda^{x_i}e^{-\lambda}}{x_i!}\cdot\prod_{i:x_i=0}\pi^{z_i}[(1-\pi)e^{-\lambda}]^{1-z_i}\right] \\ &= \sum_{i:x_i>0}(\log(1-\pi) + x_i\log\lambda - \lambda - \log x_i!) \\ &+ \sum_{i:x_i=0}(z_i\log\pi + (1-z_i)\log(1-\pi) - \lambda(1-z_i))\end{aligned}$$

Rubric:

- 1 pt for the observation that $z_i = 1, x_i > 0$ will not appear, otherwise undefined log.
- 1 pt for splitting of cases or use of indicator functions (for both x and z).
- 1 pt for substituting correct joint probability cases/values
- 1 pt for correct proof to reach simplified form.

4. Write *complete* M-step **optimization problem** with $\theta = (\lambda, \pi)$: $Q(\theta, \theta^{old})$. Use results from previous parts, do not leave the answer in terms of Q or expectation. **(4 points)**

Solution: For the M-step, we need to maximize $Q(\theta, \theta^{old})$ with $\theta = (\lambda, \pi)$.

$$\begin{aligned}\arg\max_{\lambda,\pi} Q(\theta, \theta^{old}) &= \sum_{i:x_i>0}(\log(1-\pi) + x_i\log\lambda - \lambda - \log x!) \\ &+ \sum_{i:x_i=0}(\gamma\log\pi + (1-\gamma)(\log(1-\pi) - \lambda)) \\ s.t. \quad & \lambda > 0, \quad 0 < \pi < 1\end{aligned}$$

where $\gamma = P(z_i = 1 | x_i = 0; \lambda^{old}, \pi^{old}) = \frac{\pi^{old}}{\pi^{old} + (1-\pi^{old})e^{-\lambda^{old}}}$.

In second summation term, $\mathbb{E}[z_i] = 1.P(z_i = 1 | x_i = 0; \theta^{old}) + 0.P(z_i = 0 | x_i = 0; \theta^{old})$ where z_i is binary/Bernoulli random variable and expectation is wrt the current posterior probability.

Rubric:

- 1 pt for taking arg max or maximize $Q(\theta, \theta^{old})$ or arg min or minimize negative Q.
- 1 pt for correct expectation over z_i as a Bernoulli indicator variable
- 1 pt for separate usage of θ and θ^{old} in the M-step, without interchanging them.
- 1 pt if both constraints are correct with strict inequality (no partial credit if one constraint is missing or incorrect, or if strictly less/greater inequality is not used).

5. Solve the M-step optimization problem to get the updated parameters (π^*, λ^*). For convenience, we define $\gamma = P(z_i = 1 | x_i = 0; \theta^{old})$, $N_0 = \sum_{i:x_i=0} 1$ and $\sum_{i:x_i>0} 1 = N - N_0$. Represent your answers in terms of these notations (your answers can be entirely written in terms of γ, N, N_0, x_i). **(5 points)**

Solution: To get the update equation, we take derivative of Q with respect to λ and π and set it to zero (note that after taking Lagrangian for inequality constraints in the M-step, with KKT conditions on the convex minimization problem, by complementary slackness and strict inequality on primal variables, the dual variables at optimality are all 0. Therefore using stationarity condition, we are left with derivative of Q set to 0).

$$\frac{\partial Q}{\partial \lambda} = \sum_{i:x_i>0}\left(\frac{x_i}{\lambda} - 1\right) - \sum_{i:x_i=0}(1-\gamma) = 0$$

$$\lambda^* = \frac{\sum_{i:x_i>0} x_i}{N - \gamma N_0}$$

$$\frac{\partial Q}{\partial \pi} = \sum_{i:x_i>0} \left(\frac{-1}{1-\pi}\right) + \sum_{i:x_i=0} \left(\frac{\gamma}{\pi} - \frac{1-\gamma}{1-\pi}\right) = 0.$$

$$\pi^* = \frac{N_0\gamma}{N - N_0 + N_0(1-\gamma) + N_0\gamma} = \frac{N_0\gamma}{N}$$

(You may also treat π and $1-\pi$ as two components of a simplex and directly apply the optimization over simplex results from the homework).

Rubric:

- 4 pt for correct values of λ^* and π^* (2 pt each for fully correct value, no partial credits).
- 1 pt for application of Lagrangian or attempting to set derivative of Q to 0. (If only the final values are directly mentioned for λ^* and π^*, this pt will not be awarded).

Problem 6 Hidden Markov Models (20 points)

1. Recall that a hidden Markov model is parameterized by

   - initial state distribution $P(Z_1 = s) = \pi_s$,
   - transition distribution $P(Z_{t+1} = s'|Z_t = s) = a_{s,s'}$,
   - emission distribution $P(X_t = o|Z_t = s) = b_{s,o}$.

   In this problem, you need to develop an algorithm that finds the most likely hidden state path given observation data from an arbitrary subset of time steps. More concretely, for a given subset $M \subset 1, ..., T$, find

   $$\arg\max_{z_{1:T}} P(Z_{1:T} = z_{1:T}|X_t = x_t, \forall t \in M)$$

   No derivation/reasoning is needed — simply fill in the blank of the pseudocode below. **(10 points)**

Algorithm 1 Viterbi Algorithm with Missing Data

Input: Observations $\{x_t\}_{t\in M}$.
Output: The most likely path $z_1^*, \ldots, z_T^*$.
Initialize: *(to be filled)*
for $t = 2, \ldots, T$ **do**
  for each $s \in [S]$ **do**
    Compute *(to be filled)*
Backtracking: *(to be filled)*

Solution:

Initialize: For each $s \in [S]$, compute $\delta_s(1) = \begin{cases} \pi_s b_{s,x_1} & \text{if } 1 \in M, \\ \pi_s & \text{else.} \end{cases}$ **(3 points)**

Compute: **(4 points)**

$$\delta_s(t) = \begin{cases} b_{s,x_t} \max_{s'} a_{s',s}\delta_{s'}(t-1) & \text{if } t \in M \\ \max_{s'} a_{s',s}\delta_{s'}(t-1) & \text{else} \end{cases}$$
$$\Delta_s(t) = \arg\max_{s'} a_{s',s}\delta_{s'}(t-1).$$

Backtracking: let $z_T^* = \arg\max_s \delta_s(T)$. For $t = T, \ldots, 2$, set $z_{t-1}^* = \Delta_{z_t^*}(t)$ **(3 points)**

Rubric: deducted 1 points for missing $1 \in M$.

2. Given an HMM with the following probabilities:

Transition probabilities:

	Next			
Current	Start	A	B	End
Start	0	0.7	0.3	0
A	0	0.2	0.7	0.1
B	0	0.7	0.2	0.1
End	0	0	0	1

Emission probabilities:

	Word		
State:	''	'*fight*'	'*on*'
Start	1	0	0
A	0	0.4	0.6
B	0	0.7	0.3
End	1	0	0

We assume that the process stops at state 'End'. Suppose that the process starts from state 'Start' at $t = 0$ and that we observe $o_{1:2} =$ *fight on* , write down the forward messages $\alpha_s(2)$ and determine the most likely state at $t = 3$ by computing the probability for each state. Round your numbers to 4 decimal points. **(10 points)**

Solution: $t = 1$:

$$\alpha_A(1) = 0.7 * 0.4 = 0.28$$
$$\alpha_B(1) = 0.3 * 0.7 = 0.21 \quad \textbf{(2 points)}$$

$t = 2$:

$$\alpha_A(2) = 0.6 * [0.7 * 0.21 + 0.2 * 0.28] = 0.1218$$
$$\alpha_B(2) = 0.3 * [0.2 * 0.21 + 0.7 * 0.28] = 0.0714 \quad \textbf{(3 points)}$$

$$P(X_3 = A|O_{1:2} = o_{1:2}) = \frac{\sum_{s'} \alpha_{s'}(2) a_{s',A}}{\sum_{s''} \alpha_{s''}(2)}$$
$$= \frac{\alpha_A(2)a_{A,A} + \alpha_B(2)a_{B,A}}{\alpha_A(2) + \alpha_B(2)}$$
$$\approx 0.3848$$
$$P(X_3 = B|O_{1:2} = o_{1:2}) \approx 0.5152$$
$$P(X_3 = End|O_{1:2} = o_{1:2}) = 0.1 \quad \textbf{(3 points)}$$

Therefore, the most likely state at $t = 3$ given the observed sequence is B. **(2 points)**

Rubric: writing definition without final numerical value gives 0 pt.