CS689: Graphical Models - Fall 2018

Final Exam

Dec 19, 2018   Name: _____

Instructions: Write your name on the exam sheet. The duration of this exam is **two hours**. No electronic devices may be used during the exam. You may consult your notes during the exam. Up to a two inch stack of notes is permitted. No other sources are permitted. Sharing of notes during the exam is strictly prohibited. Show your work and provide answers that are as detailed as possible. Attempt all problems. Partial credit may be given for incorrect or incomplete answers. If you need extra space for answers, write on the back of the preceding page. If you have questions at any time, raise your hand.

Problem	Topic	Page	Points	Score
1	Optimization	1	10	
2	Maximum Likelihood	2	10	
3	Linear Regression	3	10	
4	Non-Linear Classification	4	10	
5	Mixture Models	5	10	
6	EM Algorithm	6	10	
7	Factor Analysis and Autoencoders	7	10	
8	Bayesian Inference	8	10	
9	Experiment Design	9	10	
10	Applications	10	10	
Total:			100	

1. (*10 points*) Optimization

(a) (5 points) Consider the objective function $f(\mathbf{m}) = \frac{1}{2} \sum_{n=1}^{N} (\mathbf{x}_n - \mathbf{m})^T \mathbf{A} (\mathbf{x}_n - \mathbf{m})$ where $\mathbf{m} \in \mathbb{R}^D$, $\mathbf{x}_n \in \mathbb{R}^D$, and $\mathbf{A}$ is a $D \times D$ strictly positive real diagonal matrix. Find the minimizer of $f(\mathbf{m})$ with respect to $\mathbf{m}$. Show your work.

(b) (5 points) Consider the objective function $f(\mathbf{x}) = \sum_{k=1}^{K} n_k \log(x_k)$ where n_k are non-negative constants. Find the maximizer of this objective function subject to the constraints $x_k > 0$ for all k, and $\sum_{k=1}^{K} x_k = 1$. Show your work and explain your solution.

2. (*10 points*) **Maximum Likelihood:** Suppose we are interested in modeling the number of cars y crossing a particular intersection as a function of a set of feature $\mathbf{x} \in \mathbb{R}^D$ including time of day, day of week, weather, etc. We decide to use a Poisson regression model as shown below where $g(\mathbf{x}, \mathbf{w}) = \exp(\mathbf{w}^T\mathbf{x})$.

$$P(Y = y|X = x, \mathbf{w}) = \frac{1}{y!}g(\mathbf{x}, \mathbf{w})^y \exp(-g(\mathbf{x}, \mathbf{w}))$$

(a) (5 points) What is the conditional log likelihood of this model given a data set $\mathcal{D} = \{(y_n, \mathbf{x}_n)|n = 1, ..., N\}$? Provide an expression that is as detailed as possible and explain your answer.

(b) (5 points) Explain how you would find the optimal value of $\mathbf{w}$ using maximum conditional likelihood estimation. Provide supporting equations and/or sketch an algorithm.

3. (*10 points*) **Linear Regression:** Consider the linear regression model $y = \mathbf{w}^T\mathbf{x} + b$. Suppose we have access to code that fits the model by minimizing the objective

$$f(\mathbf{w}, b) = C \cdot \sum_{n=1}^{N} |y_n - (\mathbf{w}^T\mathbf{x}_n + b)| + \sum_{d=1}^{D} w_d^2$$

for a given constant C and a data set $\mathcal{D} = \{(y_n, \mathbf{x}_n)|n = 1, ..., N\}$. What probabilistic model has a conditional log likelihood function corresponding to this objective function? Provide the conditional density $P(Y = y|\mathbf{X} = \mathbf{x})$ and the prior $P(\mathbf{W} = \mathbf{w})$ in terms of $\mathbf{w}$, b and C. Show your work and explain your answer.

4. (*10 points*) Non-Linear Classification:

(a) (5 points) Explain two ways that a linear support vector classifier can be modified to produce non-linear decision boundaries. Provide equations to support your answer.

(b) (5 points) Explain what the primary trade-offs are between using a non-linear support vector machine compared to a multi-layer neural network for learning a non-linear classifier.

5. (*10 points*) **Mixture Models** Suppose we have a data set $\mathcal{D}$ consisting of data vectors $\mathbf{x} = [\mathbf{x}_B, \mathbf{x}_R]$ where $\mathbf{x}_B$ is a block of D_B binary variables and $\mathbf{x}_R$ is a block of D_R real-valued variables.

(a) (5 points) Explain how you could model these data using a mixture model. Provide an expression for the joint distribution $P(\mathbf{X} = \mathbf{x}, Z = z)$ for your proposed model.

(b) (5 points) Provide an expression for $P(Z = z | \mathbf{X} = \mathbf{x})$ for your proposed model. Show your work.

6. (*10 points*) **EM Algorithm** Suppose we have a data set $\mathcal{D} = \{x_n | n = 1, ..., N\}$ consisting of N non-negative real numbers $x_n \in \mathbb{R}^{\geq 0}$ that we wish to model using a mixture of exponential distributions. The resulting Jensen's inequality lower bound on the log marginal likelihood is given below. Derive the M-Step update for the parameter λ_k. Show your work.

$$J(\mathcal{D}; \lambda, \phi_{1:N}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \phi_{kn}(\log(\lambda_k) - \lambda_k x_n + \log \theta_k - \log \phi_{nk})$$

7. (*10 points*) Factor Analysis and Autoencoders

(a) (5 points) Factor analysis can be generalized by replacing the linear relationship between the mean of $\mathbf{x}$ and z with a multi-layer neural network $g_{\mathbf{w}}(\mathbf{z})$ resulting in the likelihood $P(\mathbf{X} = \mathbf{x}|\mathbf{Z} = \mathbf{z}) = \mathcal{N}(\mathbf{x}; g_{\mathbf{w}}(\mathbf{z}), \Psi)$. Explain what the problem is with learning this model using maximum marginal likelihood estimation. Support your answer with equations as needed.

(b) (5 points) Suppose we train a deep, non-linear autoencoder on images of digits. We then input noisy digit images into the autoencoder and find that the autoencoder produces outputs that exactly match the noisy inputs (i.e., the MSE between *noisy inputs* and reconstructions is 0). Explain what problem the learned network appears to have, why it might have this problem, and how we might be able to fix this problem.

8. (*10 points*) **Bayesian Inference:** Suppose that $x \in \mathbb{R}^{\geq 0}$ is an exponentially distributed random variable $P(X = x|\theta) = \theta \exp(-\theta x)$ where $\theta > 0$. The conjugate prior to θ is a Gamma distribution as defined below. Use this information to answer the following questions.

$$P(\theta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} \theta^{\alpha-1} \exp(-\beta \cdot \theta)$$

(a) (5 points) Given data $\mathcal{D} = \{x_n|n = 1, ..., N\}$, what is the likelihood of $\mathcal{D}$ given θ? Explain your answer and simplify as much as possible.

(b) (5 points) Given data $\mathcal{D} = \{x_n|n = 1, ..., N\}$, what is the posterior distribution of θ given $\mathcal{D}$? Show your work and explain your answer.

9. (*10 points*) **Experiment Design:** We are attempting to select the latent dimension k of a factor analysis model. Given a data set $\mathcal{D} = \{\mathbf{x}_n | n = 1, ..., N\}$, we train the model for different values of k obtaining the maximum likelihood parameters θ_k^{MLE}. We then evaluate the log likelihood of $\mathcal{D}$ using θ_k^{MLE}, obtaining the value L_k. We then plot the values of L_k and see that they are monotonically increasing with k, so we decide to increase the maximum value of k, and re-run the experiment. Use this information to answer the following questions.

(a) (5 points) What mistake have we made in this experimental design?

(b) (5 points) Describe an alternative experimental design that will allow us to properly select the value of k.

10. (*10 points*) **Applications:** Suppose we have a trained deep neural network classifier $f_{\mathbf{w}}(\mathbf{x})$ that maps complete data cases $\mathbf{x} \in \mathbb{R}^D$ to class probabilities. Assume there are C classes. Unfortunately, some of our test data instances are incomplete (i.e., $\mathbf{x}_n = [\mathbf{x}_n^o, \mathbf{x}_n^m]$ where $\mathbf{x}_n^o$ are observed and $\mathbf{x}_n^m$ are missing). Propose an approach to use the existing neural network $f_{\mathbf{w}}(\mathbf{x})$ combined with a model of the features $P(\mathbf{X} = \mathbf{x})$ to make predictions for data cases with missing values. Explain the choices you make and provide pseudo code implementing your approach.