Probabilistic Graphical Models 10-708
Homework 3: Due March 21, 2014 at 4 pm

Directions. This homework assignment covers the material presented in Lectures 9-12. You must complete all four problems to obtain full credit. To submit your assignment, please upload a pdf file containing your writeup and a zip file containing your code to Canvas by 4 pm on Friday, March 21st. We highly encourage that you type your homework using the LaTeX template provided on the course website, but you may also write it by hand and then scan it.

Important Note. You homework writeup should be saved as a pdf file. Before submitting your assignment, please double check that everything you want to include shows up in the compiled pdf. You must also make sure that this document is legible. This means that if you don't use LaTeX, your handwriting must be neat and your answers must be organized. Next, place all of your code in a directory and compress it into a zip file. Do not place tex files or anything else inside of this directory. Finally, when submitting the assignment, please separately attach both the pdf file and the zip file to your Canvas submission. This makes it much easier for us to grade the non-programming questions because your writeup will load on the page in the grading tool that we use. If you do not follow these instructions, e.g. if you place your writeup inside of the zip file or write illegibly, we will take 5 points off of your final homework grade, and possibly more if we cannot read your answers to certain questions.

1 Fundamentals [25 points]

Part 1: Designing PGMs to Answer Queries [20 points]

In this question, you will get practice starting with a problem and some real-world observations, designing a graphical model, and suggesting algorithms to perform inference and solve the problem. We list a couple of scenarios below; for each one, carry out the following steps:

(a) List and define the random variables that will exist in your model.

(b) Define a generative process for your model (explicitly write out a parametric form for all local conditional distributions, define potentials over cliques, etc.—after this step, you should have a fully specified model).

(c) Draw the graphical model.

(d) Explain what quantity you would like to learn or infer (describe mathematically the quantity you'd like to end up with) and explain how this quantity solves the given problem.

(e) Suggest a PGM inference or learning algorithm that you could run on this graphical model to get the quantity of interest (that you've just described above), and describe the main parts of the algorithm you would need to derive to use it.

Note that there is not one correct answer, so feel free to be creative. We will grade based on the feasibility of your answer.

Here are the scenarios:

1. You are a TA for a graphical models class at your local university. You'd like to know which students are friends, but cannot figure it out based on their in-class behavior (they are really good pupils who do not talk to each other during class). However, you know that friends will work on homework together, and so once a month you collect and analyze all of their homework for the month to try and figure out the friendships. You are also keenly aware that each month may have a slightly different result, since friendships may begin, end, and shift from month to month.
2. You were recently hired as a research scientist at Google, and your boss Geoff wants you to make a graphical model that can be applied to wildlife images to find the underlying groups of animals that tend to coexist in the same habitat. You are given a large stack of photos, each containing a number different animals traveling together, and you are told there are 200 total animals in the dataset. In each photo, you have access to pixel data.

Part 2: Time Series Models [5 points]

The following questions will ask you to compare hidden Markov models (HMMs), conditional random fields (CRFs), and linear dynamical systems (LDSs, also known as kalman filter models).

1. Describe an example type of data and inferential task where it is advantageous to use an HMM over a CRF or an LDS. Explain why.
2. Describe an example type of data and inferential task where it is advantageous to use a CRF over an LDS or an HMM. Explain why.
3. Describe an example type of data and inferential task where it is advantageous to use an LDS over an HMM or a CRF. Explain why.
4. Explain how to modify the forward-backward algorithm for inference in HMMs to be applied to inference in LDSs. How does this new form dictate or limit the modeling assumptions we are able to choose?
5. Describe how the forward-backward algorithm for inference in HMMs can be used in the context of a CRF.

2 E-M for the Mixture of Experts Model [25 points]

In this problem, we will derive the update steps of the Expectation-Maximization (E-M) algorithm for the mixture of experts model shown in Figure 1. Here E-M will be used to estimate the values of the parameters θ_k for $k = 1, \ldots, K$. You may assume that all other parameters are fixed.

In the mixture of experts model, we have observed variables $X_n \in \mathbb{R}^p$, $Y_n \in \mathbb{R}$ and latent variable $Z_n \in \{1, ..., K\}$, where the subscript n denotes the nth data instance. The generative process for this model is as follows.

$$Z_n \mid X_n = x_n \sim \text{Categorical}\left(\frac{\exp\{\gamma_1^T x_n\}}{\sum_k \exp\{\gamma_k^T x_n\}}, \ldots, \frac{\exp\{\gamma_K^T x_n\}}{\sum_k \exp\{\gamma_k^T x_n\}}\right)$$

$$Y_n \mid X_n = x_n, Z_n = k \sim \text{Normal}\left(\theta_k^T x_n \,,\, \sigma_k^2\right)$$

Note that when $Z_n = k$ is observed, $Y_n \mid X_n$ follows a simple regression model with feature weights θ_k and error variance σ_k^2. In the remainder of this problem we will use the indicator vector notation $Z_n = (Z_{n1}, \ldots, Z_{nK})$ where $Z_{nk} = 1$ if $Z_n = k$ and $Z_{nk} = 0$ otherwise.

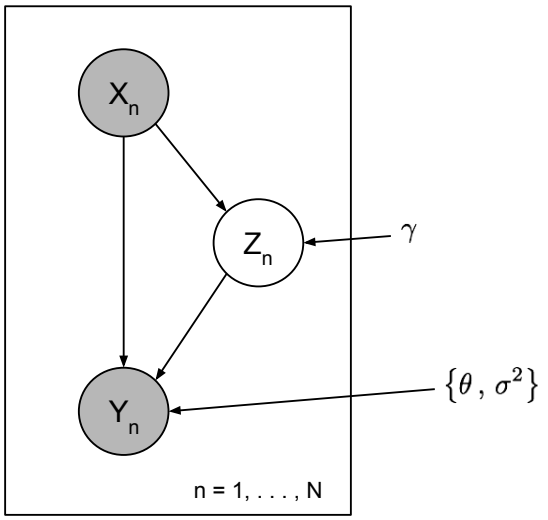

Figure 1: The mixture of experts model.

1. [15 pts] If all of the variables in this model were fully observed, we would obtain parameter estimates by maximizing the conditional log likelihood of the data, given by

$$\ell(\theta\,;\,\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) = \log p(\boldsymbol{y},\boldsymbol{z}\,|\,\boldsymbol{x},\theta\,;\,\sigma^2,\gamma)$$

where the model parameters are denoted by $\theta = \{\theta_k\}_{k=1}^K, \sigma^2 = \{\sigma_k^2\}_{k=1}^K, \gamma = \{\gamma_k\}_{k=1}^K$ and the data is $\boldsymbol{x} = \{x_n\}_{n=1}^N$, $\boldsymbol{y} = \{y_n\}_{n=1}^N$, $\boldsymbol{z} = \{z_n\}_{n=1}^N$. However, since the Z's are actually latent variables, the above expression does not decompose nicely, and the optimization problem cannot be solved in closed form. In the E-M algorithm, we instead work with a lower bound on the conditional log likelihood. Show that such a lower bound is given by

$$\ell(\theta\,;\,\boldsymbol{x},\boldsymbol{y}) \geq \sum_{n=1}^N \mathbb{E}_{p(z|x,y)}\big[\log p(z_{nk}\,|\,x_n\,;\,\gamma_k)\big] + \sum_{n=1}^N \mathbb{E}_{p(z|x,y)}\left[\log p(y_n\,|\,x_n, z_{nk}\,;\,\theta_k,\sigma_k^2\right]$$

Make sure to show all of your work, and provide clear and complete explanations for the steps you take when they are not immediately evident from the math.

2. [10 pts] Derive the E step and M step update equations for θ_k where $k = 1,\ldots,K$.

3 Conditional Random Fields [25 points]

Conditional Random Fields (CRFs) have been successfully applied in many structured prediction problems. For example, in text analysis, we may be interested in finding the noun phrases (NPs) in a sentence. To illustrate, the NPs are underlined in the following sentence:

"I am Gandalf the White, and I come back to you now at the turn of the tide." – Gandalf in *The Lord of the Rings*.

Let x_i be the tokens in the sentence, and $y_i \in \mathcal{L} = \{B, I, O\}$ be the labels where B is the beginning of an NP, I is an intermediate token in NP, and O stands for others. We would like to have the following labels:

I [B] am [O] Gandalf [B] the [I] White [I], and [O] I [B] come [O] back [O] to [O] you [B] now [O] at [O] the [B] turn [I] of [I] the [I] tide [I]

We consider the following parametric form of CRF:

$$p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{w}) = \frac{1}{Z(\boldsymbol{x};\boldsymbol{w})}\exp\{\boldsymbol{w}^\top \sum_{i=1}^{n} \boldsymbol{f}(x_i, y_i, y_{i-1})\}$$

$$\text{where} \quad Z(\boldsymbol{x};\boldsymbol{w}) = \sum_{\boldsymbol{y'}\in\mathcal{L}^n} \exp\{\boldsymbol{w}^\top \sum_{i=1}^{n} \boldsymbol{f}(x_i, y'_i, y'_{i-1})\} \tag{1}$$

Here i indexes the tokens in sentence $\boldsymbol{x}$, and $\boldsymbol{w} \in \mathbb{R}^d$ is the parameter vector, $\boldsymbol{f} : \Sigma \times \mathcal{L} \times \mathcal{L} \to \mathbb{R}^d$ is the d-dimensional feature vector function where Σ is the set of English vocabularies, and n is the number of words in the sentence. Note that this is a much more restricted CRF than the usual parametric form that uses the entires sequence $\boldsymbol{x}$ in the features function (e.g., $\boldsymbol{f}(\boldsymbol{x}, y_i, y_{i-1})$).

1. [3 pts] Draw the graphical model corresponding to the conditional distribution defined in Eq. 1. You may assume that $\boldsymbol{f}(x_i, y_i, y_{i-1}) = \boldsymbol{g}(x_i, y_i)+\boldsymbol{h}(y_i, y_{i-1})$ and $\phi_i(x_i, y_i) := \exp\{\boldsymbol{w}^\top \boldsymbol{g}(x_i, y_i)\}$ and $\psi_i(y_i, y_{i-1}) := \exp\{\boldsymbol{w}^\top \boldsymbol{h}(y_i, y_{i-1})\}$ (these assumptions only apply to this question). You can do it in directed, undirected, or factor graph form.

2. [4 pts] We wish to define feature functions $f_j : \Sigma \times \mathcal{L} \times \mathcal{L} \to \mathbb{R}$ to extract properties of local parts of the sequence that are useful for the NP identification. Define two feature functions such that one is likely to be associated with a positive weight w_j and the other associated with a negative weight. Briefly justify why you expect w_j would be positive or negative. [Hint: you can consider using an indicator function of certain configuration of x_i, y_i, and/or y_{i-1}]

3. [8 pts] Define a polynomial time algorithm that calculates the exact (marginal) probability that subsequence $\{x_i, ..., x_{i+k}\}$ is a NP, given the sequence and the CRF. That is, you have $\boldsymbol{w}$ and the feature function $\boldsymbol{f}$, i, k, and the input sentence $\boldsymbol{x}$. You should provide sufficient mathematical details by giving the necessary update equations. Note that a sequence $\{x_i, ..., x_{i+k}\}$ is a NP iff $y_i = B$ and $y_{i+1} = ... = y_{i+k} = I$ (We ignore the possibility that this NP is incomplete and can be followed by another I).

4. [5 pts] To learn the parameters of a CRF, we maximize the conditional log-likelihood of a sample of training data $\mathcal{D} = \{x_i^{(t)}, \boldsymbol{y}_i^{(t)}\}_{i=1}^n$,

$$\mathcal{L}(\boldsymbol{w}) = \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}} \log p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{w}) \tag{2}$$

$$= \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}} \left\{ \boldsymbol{w}^\top \sum_{i=1}^{|\boldsymbol{x}|} \boldsymbol{f}(x_i, y_i, y_{i-1}) - \log Z(\boldsymbol{x}, \boldsymbol{w}) \right\} \tag{3}$$

   by adjusting the parameter $\boldsymbol{w}$. Most methods involve computing gradient with respect to $\boldsymbol{w}$. Show that

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}} \left\{ \sum_{i=1}^{|\boldsymbol{x}|} \boldsymbol{f}(x_i, y_i, y_{i-1}) - \underbrace{\mathbb{E}_{p(\boldsymbol{y'}|\boldsymbol{x};\boldsymbol{w})}\left[\boldsymbol{f}(x_i, y'_i, y'_{i-1})\right]}_{\text{model expectation}} \right\} \tag{4}$$

5. [5 pts] Give a polynomial time algorithm that computes $\sum_{i=1}^{|\boldsymbol{x}|} \mathbb{E}_{p(\boldsymbol{y'}|\boldsymbol{x};\boldsymbol{w})}\left[\boldsymbol{f}(x_i, y'_i, y'_{i-1})\right]$ for a training instance $(\boldsymbol{x}, \boldsymbol{y})$. Again you should provide sufficient mathematical details. [Hint: you may reuse the algorithm you defined in part (3)]

4 Learning Gaussian Graphical Models [25 points]

Suppose we have n data points $\boldsymbol{x}_1, ..., \boldsymbol{x}_n$ where each instance $\boldsymbol{x}_i \in \mathbb{R}^p$ is one sample of a p-dimensional random variable drawn independently from the multivariate Normal distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

parameterized by a mean vector $\boldsymbol{\mu} \in \mathbb{R}^p$ and a covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$. Let $\boldsymbol{\Theta} = \Sigma^{-1}$ be the inverse covariance matrix, or precision matrix, of this distribution. The goal of this problem is to obtain a sparse estimate of the precision matrix. Note that we can first transform the data distribution to a zero-mean multivariate Normal distribution by applying the following transformation:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i$$
$$\boldsymbol{x}_i := \boldsymbol{x}_i - \hat{\boldsymbol{\mu}} \quad \text{for } i = 1, \ldots, n$$

This doesn't affect the covariance of the distribution, and will make our lives much easier.

As we learned in class, estimating the zeroes in the precision matrix is equivalent to estimating the complete set of pairwise conditional independencies among the variables in our multivariate Normal distribution. To be precise, we have the following equivalence relationship.

$$X_j \perp X_k \mid \boldsymbol{X}_{-jk} \iff \theta_{jk} = 0$$

where $\boldsymbol{X}_{-jk}$ denotes the set of all variables except for X_j and X_k. Because of this, estimating the precision matrix $\boldsymbol{\Theta}$ is equivalent to estimating the structure of a Markov random field defined over the variables $X_1, ..., X_p$, where an edge between two variables exists if and only if the corresponding precision matrix element is nonzero. This model is therefore often referred to as a Gaussian graphical model (GGM). Note that GGMs are often used to model networks (i.e. pairwise relationships among a set of objects, represented by the nodes) whose structure is unknown.

Part 1: Maximum Likelihood Inverse Covariance Estimation [5 points]

First, we will derive the maximum likelihood estimate of $\boldsymbol{\Theta}$, assuming without loss of generality that $\boldsymbol{\mu} = 0$. Let

$$\boldsymbol{X} = \begin{bmatrix} \text{—} & \boldsymbol{x}_1 & \text{—} \\ \text{—} & \boldsymbol{x}_2 & \text{—} \\ & \vdots & \\ \text{—} & \boldsymbol{x}_n & \text{—} \end{bmatrix} \in \mathbb{R}^{n \times p}$$

be the data design matrix.

1. Show that the log likelihood of the data is given by

   $$\ell(\boldsymbol{\Theta}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \log p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{\Theta}) \propto \log \det(\boldsymbol{\Theta}) - \operatorname{tr}(\boldsymbol{S}\boldsymbol{\Theta})$$

   up to a constant, where $\boldsymbol{S} = \frac{1}{n}\boldsymbol{X}^T\boldsymbol{X}$ is the sample covariance matrix.

2. Derive the maximum likelihood estimate of $\boldsymbol{\Theta}$. Show your work.

3. Give an example of a scenario in which the MLE cannot be computed. Also provide one reason why we might not want to use the MLE even if it can be computed.

Part 2: Sparse Inverse Covariance Estimation [20 points]

Next, we will implement two known procedures for obtaining a sparse estimate of $\boldsymbol{\Theta}$. For this problem, we will use the data provided in the "ggm_data.mat" file.

1. Implement the neighborhood selection algorithm of Meinshausen & Bühlmann, High-dimensional graphs and variable selection with the lasso, *The Annals of Statistics*, 2006.

2. Implement the glasso algorithm of Friedman et al., Sparse inverse covariance estimation with the graphical lasso, *Biostatistics*, 2008.

3. Apply both of the above methods to the data provided using each of the following values for the regularization parameter: $\lambda = [0, 20, 30, 40]$ for neighborhood selection; $\lambda = [0, 0.2, 0.5, 0.8]$ for glasso. Plot each of the estimated precision matrices as binary images, using black for elements that are nonzero and white for elements whose value is zero. Make sure to label each image with its corresponding value of λ.

Hint: Don't forget to transform the data to a zero-mean distribution.