Your Name
email@fas.harvard.edu
CS281-F17

Assignment #3
Due: 5pm October 20, 2017

Collaborators: Jane Doe

Homework 3: Inference - Solutions

Grading Instructions

In the solutions, you will see several highlighted checkpoints. These each have a label that corresponds to an entry in the Canvas quiz for this problem set. The highlighted statement should clearly indicate the criteria for being correct on that problem. If you satisfy the criteria for a problem being correct, mark "Yes" on the corresponding position on the Canvas quiz. Otherwise, mark "No". Your homework scores will be verified by course staff at a later date.

That being said, many of the problems in this course will be proofs. If you find a proof that isn't referred to by the course solution, don't worry. If you're uncertain about the proof, make a Piazza post. If you're certain, mark it correct (and we'll look at it during verification).

Introduction

There is a mathematical component and a programming component to this homework. Please submit your PDF and Python files to Canvas, and push all of your work to your GitHub repository. If a question requires you to make any plots, please include those in the writeup.

Question 1 (30 pts)

(a) See the function `score(...)` in `p1.py`. The score is just 5: 2 + 3.

Check 1: Correctly compute the score with implementation - 2 pts. Get 1 pt if either the score is incorrect or the implementation does not yield the correct score..

(b) See the function `brute_force_log_partition`. $A(\theta) = 8.138$. The probability of that assignment is 0.0436.

Check 2: Correctly compute the partition term - 3pt; correctly compute the probability of the assignment, 2 pts.

(c) See the function `brute_force_log_probability`. $p(\text{RS} = 1) = 0.9847$.

Check 3: Implement a successful brute-force algorithm that runs within a practical time frame) - 3pts. Compute the probability correctly, 2 pts..

(d) We know from exponential families that the derivative of the log partition with respect to a parameter θ_s is the expectation of the corresponding sufficient statistic $\phi_s(x)$. To see this, you can simply write out the sum, throw away all terms other than the ones involving θ_s. You are left with $\theta_s \phi_s(x_i)$ for all cliques i (in this case unary cliques), and a denominator that normalizes them. Remember also that the expectation of a binary random variable x is equal to $P(x = 1)$.

$$\nabla_\theta A(\theta) = \mathbb{E}[\phi(x)]$$

So, we can take the gradient of the partition function w.r.t. each of the unary potentials to get the marginal probabilities.

Check 4: Note or use the fact that derivatives are related to expectation in exponential family distributions- 3 pts; derive a simpler expression for marginal probability - 2 pts. Thought exercise: what would it mean to differentiate w.r.t the binary parameters?

(e) See the function `autograd_marginal_probabilities`.

Check 5:; Implement a simplified way to compute marginals that takes advantage of the above results, 2 pts (get one point if you used some other way to simplify it); results check with previous results, 3pt.

(f) See the function `serial_mp`.

Check 6: Implement a functioning BP algorithm - 3 pts; get correct marginals - 3 pts.

(g)

Check 7: Give reasonable account of how the graphical structure contributes to the discrepancy. 2 pts.

Question 2 (25 pts)

(a) Directed Graphical Model with plates, assuming u_i and v_j are random vectors. See last part of this question but ignore priors. **Check 8:** (1 pt) Correct graphical model

(b) Log-likelihood of the model:

$$\log P(R) = \sum_{i,j} \left(-\frac{1}{2\sigma^2} \left(r_{i,j} - u_i^T v_j \right)^2 - \log \sigma - \frac{1}{2} \log(2\pi) \right)$$

The gradient:

$$\frac{\partial}{\partial \mathbf{u}_i} \log p(\mathcal{D}|\mathbf{u}, \mathbf{v}, \sigma) = \sum_j \frac{1}{2\sigma^2} \left(r_{i,j} - \mathbf{u}_i^T \mathbf{v}_j \right) \mathbf{v}_j$$
$$\frac{\partial}{\partial \mathbf{v}_j} \log p(\mathcal{D}|\mathbf{u}, \mathbf{v}, \sigma) = \sum_i \frac{1}{2\sigma^2} \left(r_{i,j} - \mathbf{u}_i^T \mathbf{v}_j \right) \mathbf{u}_i$$

Check 9: (2 pts) Correct log likelihood and gradient

(c) Implementation **Check 10:** (5 pts) Correct implementation

(d) Implementation **Check 11:** (5 pts) Correct implementation

(e) Implementation **Check 12:** (5 pts) Correct implementation

(f) We now introduce biases, so that our model is now

$$r_{i,j} \sim \mathcal{N}(\mathbf{u}_i^T \mathbf{v}_j + a_i + b_j + g, \sigma^2),$$

where a_i and b_j are the biases on the users and jokes respectively and g is the global bias. We get the following value for g: 3.41.

Using the values of b_j to determine the quality of jokes, we find that the top five jokes, with their offsets, are:

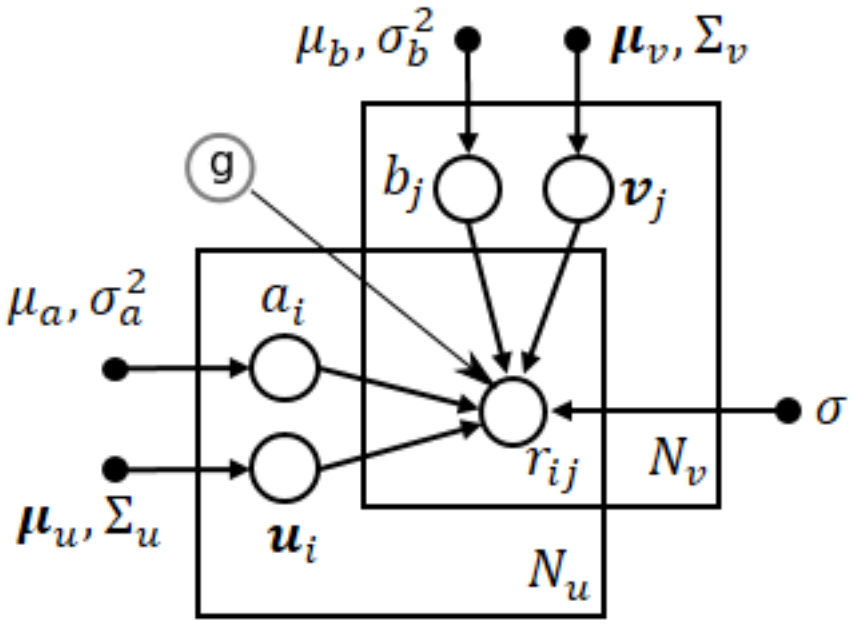
Ranking	Joke number	Bias
1	49	0.782922
2	127	0.732573
3	119	0.684446
4	72	0.676239
5	69	0.641234

Whereas the bottom five are:

Ranking	Joke number	Bias
150	141	-1.055245
149	75	-1.038110
148	58	-0.973321
147	124	-0.856399
146	7	-0.839214

Check 13 (5 pts): 3 pts for giving biases for top and bottom jokes. 2 pts for global bias value.

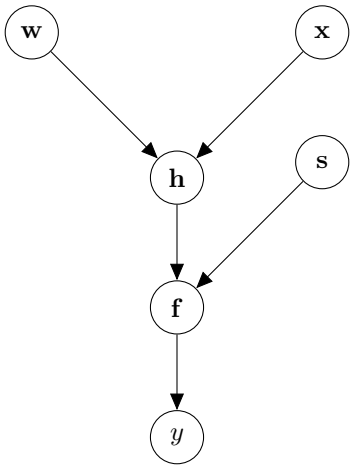
(g) We could introduce any priors we want to, but we'll go with Gaussians. We keep σ fixed as a hyper-parameter. The graphical model is as follows:



Check 14: (2 pts) Correct graphical model of reasonable priors.

Question 3 (25 pts)

1. Graphical Model



Ordinal Regression

Note that it's okay to have a node for σ or to separate the noise into another random variable like above.

Check 15:(2 pts) Correct graphical model

2. Let $s \sim \mathcal{N}(0, \sigma^2)$, so $f = h + s$.

$$\begin{aligned} p(y = r|h) &= \int p(y = r|f = h + s)\mathcal{N}(s|0, \sigma^2)\mathbf{d}s \\ &= \Phi(\frac{b_{r+1} = h}{\sigma}) - \Phi(\frac{b_r - h}{\sigma}) \end{aligned}$$

where $\Phi(x)$ is the standard normal Gaussian CDF given by $\int_{-\infty}^{x} \mathcal{N}(x'|0, 1)\mathbf{d}x'$. Notice that the first term in the product within the integral above only takes on the values 0 and 1 and "selects" the part of the real line over which to integrate the second term. Another way to write, by symmetry of the Gaussian, is:

$$p(y = r|h) = \Phi((h - b_r)/\sqrt{\sigma^2}) - \Phi((h - b_{r+1})/\sqrt{\sigma^2}), \tag{1}$$

Without an additive noise term, $p(y = r|h)$ will be a delta function, where $p(y = r^*|h) = 1$ for the $b_{r^*} \leq h < b_{r^*+1}$ and $p(y = r|h) = 0$) for the other r values.

Check 16:(2 pts) Computed form of $p(y = r|h)$ correctly

3. The predictive mean is

$$\mathbf{E}[r|\mathbf{w}, \sigma^2, \mathbf{x}] = \sum_{r=1}^{R} r \left\{ \Phi((\mathbf{w}^{\mathrm{T}}\mathbf{x} - b_r)/\sqrt{\sigma^2}) - \Phi(((\mathbf{w}^{\mathrm{T}}\mathbf{x} - b_{r+1})/\sqrt{\sigma^2}) \right\}, \tag{2}$$

As $\sigma \to 0$, the predictive mean $\to r^*$ where $b_{r^*} \leq \mathbf{w}^T\mathbf{x} < b_{r^*+1}$. As $\sigma \to \infty$, the predictive mean is the weighted average of all possible ratings, where the weights correspond to the lengths of the intervals. For intermediate values of σ, the predictive mean lies in between the above two values.

Check 17:(3 pts) 2 pts for correct form of predictive mean, 1 pt for explaining or including diagram on how this changes with σ.

4. The test RMSE is 1.236.

   Check 18:(7 pts) Implementation and reasonable RMSE

5. The test RMSE is 1.3. Performance decreases because the Gaussian likelihood does not take into account that the ratings are discrete.

   Check 19:(7 pts) Implement Gaussian likely correctly to get reasonable RMSE

6. For such a model, the σ depends on the text, which allows different jokes to have different rating variances, so this is a more general model compared to a fixed σ.

   Check 20:(2 pts) Recognize that different jokes will have different rating variances.

7. The performance of the latent factor model is better because it learns latent features for each joke and user using the ratings from the different users. These latent features are more informative than the text for the joke.

   Check 21:(2 pts) Correct answer for 1 pt. Reasonable explanation for 1 pt.