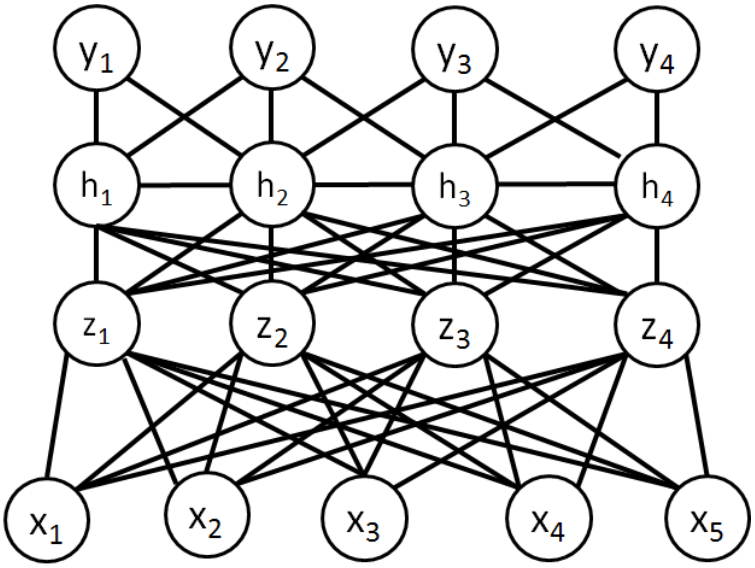
CPSC 540 Assignment 5 (due April 10)

UGMs, Bayes, and Literature Survey

1 Undirected Graphical Models

1.1 Conditional UGM

Consdier modeling the dependencies between sets of binary variables x_j and y_j with the following UGM which is a variation on a stacked RBM:

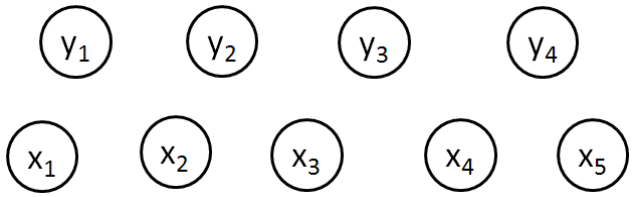


Computing univariate marginals in this model will be NP-hard in general, but the graph structure allows efficient block updates by conditioning on suitable subsets of the variables (this could be useful for designing approximate inference methods). For each of the conditioning scenarios below, draw the conditional UGM and comment on how expensive it would be to compute univariate marginals (for all the variables) in the conditional UGM.

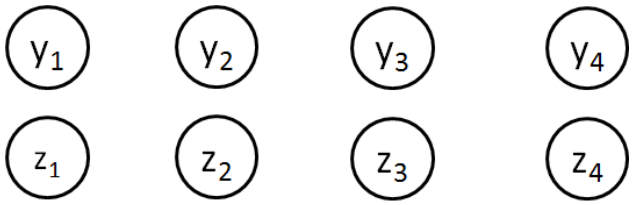
1. Conditioning on all the z and h values.
2. Conditioning on all the x and h values.
3. Conditioning on all the z and y values.
4. Conditioning on all the x and z values.

Answer:

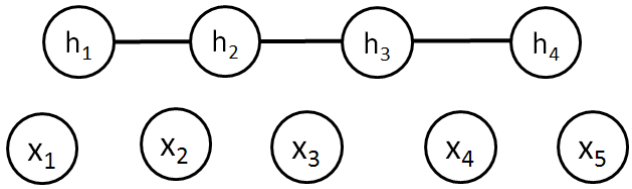
1. This gives a disconnected graph, so computing marginals is trivial (it only involves univarate probabilities).



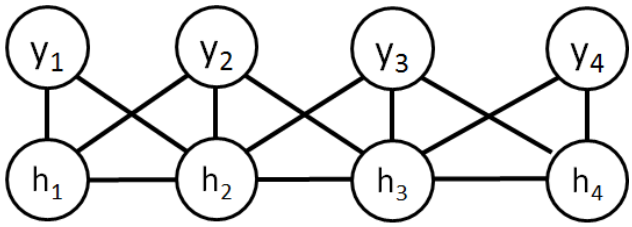
2. This gives a disconnected graph, so computing marginals is trivial (it only involves univarate probabilities).



3. This gives a disconnected graph and a chain, so computing marginals is cheap by using message-passing (the messages will depend on only 1 variable).



4. This graph is not a tree so message passing will be less cheap, but the messages will depend on at most 2 parents so it will still be fairly cheap.



1.2 Fitting a UGM to PINs

The function *example_UGM* loads a dataset X containing samples of PIN numbers, based on the probabilities from the article at this URL: `http://www.datagenetics.com/blog/september32012`.[1]

This function shows how to use the UGM software to fit a UGM model to the dataset, where all node/edge parameters are tied and the graph is empty. It then performs decoding/inference/sampling in the fitted model. Unfortunately, this is not a very good model of the data for several reasons:

1. The decoding is 1 1 1 1, whereas in the data the most likely value by far is 1 2 3 4. Similarly, the sampler doesn't tend to generate 1 2 3 4 even though this happens in more than 1/10 samples.

[1] I got the probabilities from the reverse-engineered heatmap here: `http://jemore.free.fr/wordpress/?p=73`.

2. The marginal probability of the first number being 1 is 22.06%, which is acutally too low (it should be 38.54%). In addition, the marginal probabilities of the remaining nubmers being 1 are also 22.06%, and these numbers are too high.

3. Conditioned on the first three numbers being 1 2 3, the probability that the last number is 4 is less than 10% in the model, whereas in the data it's more than 90%.

In this question you'll explore better models of this data and different aspects of UGMs.

1. Why does w have a length of 9?

2. Write an equation for the model being used by the code.

3. What are potential sources of the problems above?

4. Modify the demo to use a *tied* value of 0 and re-run the demo. Why does the model now have 36 parameters? Comment on whether this fixes each of the above 3 issues.

5. Modify the demo to use chain-structured dependency (keeping the *tied* value at 0). Comment on whether this fixes each of the above 3 issues.

6. Modify the demo to use a completely-connected graph (keeping the *tied* value at 0). Comment on whether this fixes each of the above 3 issues.

7. UGM only support pairwise graphs, what would the effect of higher-order potentials be? What would the disdavantages of higher-order potentials be?

If you want to further explore UGMs, there are quite a few demos on the UGM webpage that you can go through which cover all sorts of things like approximate inference and CRFs.

Answer:

1. There are 10 states, but because of the sum-to-1 property we don't need to have a parameter for one of the states.

2. It should be something like

$$p(x_1, x_2, x_3, x_4) = \frac{1}{Z} \exp(w_{x_1} + w_{x_2} + w_{x_3} + w_{x_4}),$$

where $w_{10} = 0$ (or $w_0 = 0$).

3. One issue is that the parameters are tied. This means that it can't take into account that 1 is very likely in the first spot but much less likely in other spots. Another issue is that we aren't modeling dependencies between variables, so conditioning doesn't actually do anything.

4. We have 4 nodes each with 9 parameters, giving 36 parameters. This addresses the marginal probability issue, although the marginal probabilities are now correct. It partially addresses the decoding issue in that the decoding is correct, but the samples still don't tend to generate the most common value of 1 2 3 4. It doesn't fix the third issue at all as the conditional probability is still under 20%.

5. With a Markov chain, it alleviates these issues but doesn't fix them. It now sometimes generates 1 2 3 4 and the conditional probability of 4 is now over 50%.

6. With a full graph, it still isn't perfect but it's generating 1 2 3 4 close to 10% of the time and the probability of seeing a 4 after 1 2 3 is now 78%.

7. With threeway or a fourway potential you could get closer to the true frequencies in the data. However, the risk of overfitting would get much higher.

2 Bayesian Inference

2.1 Conjugate Priors

Consider a $y \in \{1, 2, 3\}$ following a multinoulli distribution with parameters $\theta = \{\theta_1, \theta_2, \theta_3\}$,

$$y|\theta \sim \text{Mult}(\theta_1, \theta_2, \theta_3).$$

We'll assume that θ follows a Dirichlet distribution (the conjugate prior to the multinoulli) with parameters $\alpha = \{\alpha_1, \alpha_2, \alpha_3\}$,

$$\theta \sim \mathcal{D}(\alpha_1, \alpha_2, \alpha_3).$$

Thus we have

$$p(y|\theta, \alpha) = p(y|\theta) = \theta_1^{I(y=1)}\theta_2^{I(y=2)}\theta_3^{I(y=3)}, \quad p(\theta|\alpha) = \frac{\Gamma(\alpha_1 + \alpha_2 + \alpha_3)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\Gamma(\alpha_3)}\theta_1^{\alpha_1 - 1}\theta_2^{\alpha_2 - 1}\theta_3^{\alpha_3 - 1}.$$

Compute the following quantites:

1. The posterior distribution,
$$p(\theta|y, \alpha).$$
2. The marginal likelihood of y given the hyper-parameters α,
$$p(y|\alpha) = \int p(y, \theta|\alpha)d\theta,$$
3. The posterior mean estimate for θ,
$$\mathbb{E}_{\theta|y,\alpha}[\theta_i] = \int \theta_i p(\theta|y, \alpha)d\theta,$$
which (after some manipulation) should not involve any Γ functions.
4. The posterior predictive distribution for a new independent observation $\hat{y}$ given y,
$$p(\hat{y}|y, \alpha) = \int p(\hat{y}, \theta|y, \alpha)d\theta.$$

Hint: You can use $D(\alpha) = \frac{\Gamma(\alpha_1)\Gamma(\alpha_2)\Gamma(\alpha_3)}{\Gamma(\alpha_1+\alpha_2+\alpha_3)}$ to represent the normalizing constant of the prior and $D(\alpha^+)$ to give the normalizing constant of the posterior. You will also need to use that $\Gamma(\alpha + 1) = \alpha\Gamma(\alpha)$. For some calculations you may find it a bit cleaner to parameterize the posterior in terms of $\beta_j = I(y = j) + \alpha_j$, and convert back once you have the final result.

Answer:

1.

$$\begin{aligned}
p(\theta|y, \alpha) &\propto p(y|\theta, \alpha)p(\theta|\alpha) \\
&= p(y|\theta)p(\theta|\alpha) \\
&\propto \theta_1^{I(y=1)}\theta_2^{I(y=2)}\theta_3^{I(y=3)}\theta_1^{\alpha_1 - 1}\theta_2^{\alpha_2 - 1}\theta_3^{\alpha_3 - 1} \\
&= \theta_1^{I(y=1)+\alpha_1 - 1}\theta_2^{I(y=2)+\alpha_2 - 1}\theta_3^{I(y=3)+\alpha_3 - 1} \\
&= \theta_1^{(I(y=1)+\alpha_1)-1}\theta_2^{(I(y=2)+\alpha_2)-1}\theta_3^{(I(y=3)+\alpha_3)-1}.
\end{aligned}$$

This is proportional to a Dirichlet distribution,

$$\theta \sim \mathcal{D}(I(y=1)+\alpha_1, I(y=2)+\alpha_2, I(y=3)+\alpha_3),$$

so we have

$$p(\theta|y,\alpha) = \frac{\Gamma(\sum_{j=1}^3 [I(y=j)+\alpha_j])}{\prod_{j=1}^3 \Gamma(I(y=j)+\alpha_j)} \theta_1^{I(y=1)+\alpha_1-1} \theta_2^{I(y=2)+\alpha_2-1} \theta_3^{I(y=3)+\alpha_3-1}$$

2.

$$\begin{aligned}
p(y|\alpha) &= \int p(y,\theta|\alpha) d\theta \\
&= \int p(y|\theta) p(\theta|\alpha) d\theta \\
&= \int \prod_{j=1}^3 \theta_j^{I(y=j)} \frac{1}{D(\alpha)} \prod_{j=1}^3 \theta_j^{\alpha_j - 1} d\theta \\
&= \frac{1}{D(\alpha)} \int \prod_{j=1}^3 \theta_j^{I(y=j)+\alpha_j} d\theta \\
&= \frac{D(\alpha^+)}{D(\alpha)}
\end{aligned}$$

This is the normalizing constant of the posterior divided by the normalizing constant of the prior.

3.

$$\begin{aligned}
\theta_i &= \int \theta_i p(\theta|y,\alpha) d\theta \\
&= \int \theta_i \frac{\Gamma(\sum_{j=1}^3 \beta_j)}{\prod_{j=1}^3 \Gamma(\beta_j)} \prod_{j=1}^3 \theta_j^{\beta_j - 1} d\theta \\
&= \frac{\Gamma(\sum_{j=1}^3 \beta_j)}{\prod_{j=1}^3 \Gamma(\beta_j)} \int \theta_i \prod_{j=1}^3 \theta_j^{\beta_j - 1} d\theta \\
&= \frac{\Gamma(\sum_{j=1}^3 \beta_j)}{\prod_{j=1}^3 \Gamma(\beta_j)} \int \prod_{j=1}^3 \theta_j^{I(i=j)+\beta_j - 1} d\theta \\
&= \frac{\Gamma(\sum_{j=1}^3 \beta_j)}{\prod_{j=1}^3 \Gamma(\beta_j)} \frac{\prod_{j=1}^3 \Gamma(I(i=j)+\beta_j)}{\Gamma(\sum_{j=1}^3 I(i=j)+\beta_j)} \\
&= \frac{\Gamma(\sum_{j=1}^3 \beta_j)}{\prod_{j=1}^3 \Gamma(\beta_j)} \frac{\prod_{j=1}^3 \Gamma(I(i=j)+\beta_j)}{\Gamma(1+\sum_{j=1}^3 \beta_j)} \\
&= \frac{\Gamma(\sum_{j=1}^3 \beta_j)}{\Gamma(1+\sum_{j=1}^3 \beta_j)} \frac{\prod_{j=1}^3 \Gamma(I(i=j)+\beta_j)}{\prod_{j=1}^3 \Gamma(\beta_j)} \\
&= \frac{\beta_i}{\sum_{j=1}^3 \beta_j} \\
&= \frac{I(y=i)+\alpha_i}{\sum_{j=1}^3 [I(y=i)+\alpha_i]}.
\end{aligned}$$

(The posterior mean minimizes the ℓ_2-risk when predicting on new data. So if all $\alpha_i = 1$, this gives a theoretical justification for the 'add one' trick that is often used when estimating probabilities. If you were interested in the ℓ_1-risk, you would instead take the posterior median.)

4.

$$\begin{aligned}
p(\hat{y}|y,\alpha) &= \int p(\hat{y},\theta|y,\alpha)d\theta \\
&= \int p(\hat{y}|\theta,y,\alpha)p(\theta|y,\alpha)d\theta \\
&= \int p(\hat{y}|\theta)p(\theta|\alpha)d\theta \\
&= \int \prod_{j=1}^{3}[\theta_j^{I(\hat{y}=j)}]\frac{1}{D(\alpha^+)}\prod_{j=1}^{3}[\theta_j^{I(y=j)+\alpha_j-1}]d\theta \\
&= \frac{1}{D(\alpha^+)}\int \prod_{j=1}^{3}\theta_j^{I(\hat{y}=j)+I(y=j)+\alpha_j-1}d\theta \\
&= \frac{D(\alpha^{++})}{D(\alpha^+)},
\end{aligned}$$

where $D(\alpha^{++})$ is the normalizing constant of the Dirichlet with parameters $(I(\hat{y}=j)+I(y=j)+\alpha_j)$. Note that this is the marginal likelihood of the new data, if we treat the posterior we got from the old data as our prior.

2.2 Empirical Bayes

Consider the model

$$y_i \sim \mathcal{N}(w^T\phi(x_i),\sigma^2), \quad w_j \sim \mathcal{N}(0,\lambda^{-1}).$$

By using properties of Gaussians the marginal likelihood has the form

$$p(y_i|x_i,\sigma,\lambda) = (2\pi)^{-d/2}|C|^{-1/2}\exp\left(-\frac{y^TC^{-1}y}{2}\right),$$

which gives a negative log-marginal likelihood of

$$-\log p(y_i|x_i,\sigma,\lambda) = \log|C| + y^TC^{-1}y + \text{const.}$$

where

$$C = \frac{1}{\sigma^2}I + \frac{1}{\lambda}\Phi(X)\Phi(X)^T,$$

As discussed in class, the marginal likelihood can be used to optimize hyper-parameters like σ, λ, and even the basis ϕ.

The demo *example_basis* loads a dataset and fits a degree-2 polynomial to it. Normally we would use a test set to choose the degree of the polynomial but here we'll use the marginal likelihood of the training set. Write a function, *leastSquaresEmpiricalBaysis*, that uses the marginal likelihood to choose the degree of the polynomial as well as the parameters λ and σ (you can assume that all λ_j are equal, and you can restrict your search for λ and σ to powers of 10). Hand in your code and report the marginally most likely values of the degree, σ, and λ. You can use the *logdet* function to compute the log-determinant.

Hint: computing $C^{-1}y$ by explicitly forming C^{-1} may give you numerical issues that lead to non-sensical solution. You can avoid these by using $y^TC^{-1}y = y^Tv$ where v is a solution to $Cv = y$ (Matlab will still give a warning due to ill-conditioning, but it won't return non-sensical results).

Answer:

The code should look roughly like this:

```
function [model] = leastSquaresEmpiricalBaysis(x,y)

[n,d] = size(x);

maxMargLik = -inf;
for lambda = 10.^[-6:6]
    for sigma = 10.^[-6:6]
        for degree = 0:20
            Xpoly = polyBasis(x,degree);
            C = (1/sigma^2)*eye(n) + (1/lambda)*Xpoly*Xpoly';
            logMargLik = -logdet(C,inf) - y'*(C\y);
            if logMargLik > maxMargLik
                bestDegree = degree;
                bestLambda = lambda;
                bestSigma = sigma;
                maxMargLik = logMargLik;
            end
        end
    end
end

Xpoly = polyBasis(x,bestDegree);
size(Xpoly)
w = ((1/bestSigma^2)*Xpoly'*Xpoly + bestLambda*eye(bestDegree+1))\Xpoly'*y/bestSigma^2;

model.w = w;
model.degree = bestDegree;
model.predict = @predict;
```

The values that optimized the marginal likelihood were degree $= 3$, $\sigma = 0.1$, and $\lambda = 0.0001$. This approach actually prefers a slightly simpler model than the fifth degree polynomial that optimizes the test error.

3 Literature Survey

Reading academic papers is a skill that takes practice. When you first start out reading papers, you may find that you need to re-read things several times before you understand them, or that details will still be very fuzzy even after you've put a great amount of effort into trying to understand a paper. Don't panic, this is normal.

Even if you are used to reading papers from your particular sub-area, it can be challenging to read papers about a completely different topic. Usually, people in different areas use different language/notation and focus on very different issues. Nevertheless, many of the most-successful people in academia and industry are those that are able to understand/adapt ideas from different areas. (There are a ton of smart people in the world working on all sorts of amazing things, it's good to know how to communicate with as many of them as possible.)

A common technique when trying to understand a new topic (or reading scientific papers for the first time) is to read and write notes on 10 papers on the topic. When you read the first paper, you'll often find that it's hard to follow. This can make reading take a long time and might still leave you feeling that many things don't make sense; keep reading and trying to take notes. When you get to the second paper, it might still be very hard to follow. But when you start getting to the 8th or 9th paper, things often start making more sense. You'll start to form an impression of what the influential works in the area are, you'll start getting to used to the language and jargon, you'll start to understand what the main issues that people who work on the topic care about, and you'll probably notice some important references that weren't on your initial list of 10 papers. Ideally, you'll also start to notice how the topic has changed over time and you may get ideas of future work that you could do on the topic.

To help you make progress on your project or to give you an excuse to learn about a new topic, for this

part you should write a literature survey of at least 10 academic papers on a particular topic. While your personal notes on the papers may be longer, the survey should be at most 4 pages of text (excluding references/tables/figures) in a format similar to the one for this document. Some logical components of a literature survey might be:

- A description of the overall topic, and the key themes/trends across the papers.
- A short high-level description of what was explored in each paper. For example, describe the problem being addressed, the key components of the proposed solution, and how it was evaluated. In addition, it is important to comment on the *why* questions: why is this problem important and why would this particular solution method make progress on it? It's also useful to comment on the strengths and weaknesses of the various works, and it's particularly nice if you can show how some works address the weaknesses of prior works (or introduce new weaknesses).
- One or more logical "groupings" of the papers. This could be in terms of the variant of the topic that they address, in terms of the solution techniques used, or in chronological terms.

Some advice on choosing the topic:

- The most logical/easy topic for your literature survey is a topic related to your course project, given that your final report will need a (shorter) literature survey included.
- If you are an undergrad, or a masters student without a research project yet, you may alternately want to choose a general area (like variance-reduced stochastic gradient, non-Gaussian graphical models, recurrent neural networks, matrix factorization, neural style transfer, Bayesian optimization, etc.) as your topic.
- If you are a masters student that already has a thesis project, it could make sense to do a survey on a topic where ML intersects with your thesis (or where ML *could* intersect your thesis).
- If you are a PhD student, you could use this an excuse to learn about a *completely different* topic than what you normally work on. This can be invaluable to your future research, because during your PhD it's often hard to allocate time to learn completely new topics.

Bonus: Final Questions

For this question I want you to prepare up to 10 questions that could be added to the final exam. These questions won't directly be graded, but for each of your questions that are ultimately used on the exam you'll get an extra 1% added to your exam mark. This has the added advantage that you may know what some of the final questions are going to be. However, please do not collude by sharing the questions you submit with other groups: I would treat this as academic dishonesty and it could lead to a mark of 0 on the final.

In terms of topics, keep in mind that we are only testing on the topics covered in the assignments. This means that questions can be on topics like cross-validation, coordinate optimization, and Viterbi decoding. But questions should not be on things like ICA, neural networks, or treewidth.

In terms of difficulty, I am aiming to have a few "easier" questions that test general knowledge, and a couple questions that require more thinking. I am not looking for trick/language questions that require a particular interpretation of the phrasing of the question or questions that require reading large amounts of text.

In terms of questions, we are looking for questions that can be graded quickly since we have limited TA hours remaining. Given this constraint, some possible question formats are:

- True/false.

- Multiple choice.
- Select all that apply.
- Matching of items to concepts or definitions.
- Short calculations.

You can submit these questions up until April 17.