# Homework 1
# CS228: Probabilistic Graphical Models

Instructor: Stefano Ermon
ermon@stanford.edu

Available: Jan. 10, 2017
Due date: 11:59 p.m. on January 24, 2017, via GradeScope.
Total points: 100

**Problem 1: Probability theory (4 points)**

The doctor has bad news and good news for X. The bad news is that X tested positive for a serious disease, and that the test is 99% accurate (i.e., the probability of testing positive given that you have the disease is 0.99, as is the probability of testing negative given that you don't have the disease). The good news is that this is a rare disease, striking only one in 10,000 people. Why is it good news that the disease is rare? What are the chances that X actually has the disease?

**Solution:**

This is good news because the prior on having the disease significantly decreases the probability that X has the disease. Let $D$ indicate whether an individual has the disease, and let $T$ indicate whether the test returned positive results. The conditional probability of testing positive given that one has the disease is $\Pr(T = 1|D = 1) = .99$. The desired probability is that of the person having the disease given a positive test result, $\Pr(D = 1|T = 1)$. We can find this by using the given prior, $\Pr(D = 1) = .0001$ and the probability of testing negative given that one does not have the disease, $\Pr(T = 0|D = 0) = .99$.

$$\Pr(D = 1|T = 1) = \frac{\Pr(T = 1|D = 1)\Pr(D = 1)}{\Pr(T = 1)}$$

$$= \frac{\Pr(T = 1|D = 1)\Pr(D = 1)}{\Pr(T = 1|D = 1)\Pr(D = 1) + \Pr(T = 1|D = 0)\Pr(D = 0)}$$

$$= \frac{\Pr(T = 1|D = 1)\Pr(D = 1)}{\Pr(T = 1|D = 1)\Pr(D = 1) + (1 - \Pr(T = 0|D = 0)))(1 - \Pr(D = 1))}$$

$$= \frac{(.99)(.0001)}{(.99)(.0001) + (1 - (.99))(1 - (.9999))}$$

$$= .0098$$

**Problem 2: Review of dynamic programming (7 points)**

Suppose you have a probability distribution $P$ over random variables $X_1, X_2, ..., X_n$ which all take values in the set $\mathcal{S} = \{v_1, ..., v_m\}$, where the $v_j$ are some distinct values (e.g., integers or letters).

Suppose that $P$ satisfies the *Markov assumption*: for all $i \geq 2$ we have

$$P(x_i|x_{i-1}, ..., x_1) = P(x_i|x_{i-1}).$$

In other words, $P$ factorizes as

$$P(x_1, x_2, ..., x_n) = P(x_1)P(x_2|x_1) \cdots P(x_n|x_{n-1}).$$

For each factor $P(x_i|x_{i-1})$ for $i \geq 2$ you are given the probability $P(X_i = u|X_{i-1} = v)$ for each $u, v \in \mathcal{S}$ in the form of a $m \times m$ table. You are also given $P(X_1 = v)$ for each $v \in \mathcal{S}$.

- (7 points) Give an $O(m^2 n)$ algorithm for solving the problem

$$\max_{x_1, x_2, ..., x_n \in \mathcal{S}^n} P(x_1, x_2, ..., x_n).$$

  Hint: think dynamic programming!

**Solution:**

There are many ways to solve this problem and many different ways to formulate the solution. The intuition that was necessary to solve the problem efficiently was similar for both problems. It relies on the fact that you can change the order of the maximizations and push the cpds out of the maximizations (exactly as was shown in the lecture about Variable Elimination on Thursday). Here we present one possible solution using this intuition, note we can write:

$$\begin{aligned}\max_{x_1, x_2, ..., x_n \in \mathcal{S}^n} P(x_1, x_2, ..., x_n) &= \max_{x_1} \max_{x_2} ... \max_{x_n} P(x_1)P(x_2|x_1) \cdots P(x_n|x_{n-1}) \\ &= \max_{x_1} \max_{x_2} ... \max_{x_{n-1}} P(x_1)P(x_2|x_1) \cdots P(x_{n-1}|x_{n-2}) \max_{x_n} P(x_n|x_{n-1})\end{aligned} \tag{1}$$

The term $\max_{x_n} P(x_n|x_{n-1})$ is a function of $X_{n-1}$, that is for $X_{n-1} = s_j$ we can compute $\psi_{X_{n-1}}(s_i)$ $= \max_{x_n} P(x_n|x_{n-1} = s_i)$ which takes O(m) time (since we are maximizing over the $m$ possible values that $X_n$ can take, if we cache this value and do the same for all $s_1, ..., s_m$ (this takes $O(m^2)$ time) we can now think of having a cached table of $\psi_{X_{n-1}}(s_i)$ for each $s_i \in S$. We can proceed recursively calculating $\psi_{X_{n-2}}(s_i) = \max_{x_{n-1}} P(x_{n-1}|x_{n-2} = s_i)\psi_{X_{n-1}}(x_{n-1})$, again computing this term for each $s_i$ takes O(m) time and caching the entire table for each $s_i \in S$ takes $O(m^2)$ time. We repeat the process backwards until we have processed all the cpds (there are O(n) cpds) and our final output is $\max_{x1} P(x1)\psi_{X_1}(x1)$.

We can think of this problem as choosing a variable elimination ordering $X_n, X_{n-1}, ...X1$ and "maximizing out" the variables. As was discussed in lecture this induces a clique tree (where the scope of the largest clique has 2 variables, hence $O(m^2 n)$ complexity). Note that we could have also solved the problem in a ordering $X_1, X_2, ...X_n$ by first computing $\zeta_{X_{n-2}}(s_i) = \max_{x1} P(x1)P(X2 = s_i|x1)$ in O(m) time, cached the results for each $s_i \in S$ in $O(m^2)$ time and processed O(n) cpds according in the forwards direction. Again we can think of this processes as inducing a clique tree where the scopre of the largest clique has 2 variables.

**Common Mistakes** Most mistakes in this problem were caching maximizations that were not consistent with the original problem. For example, many students presented a (incorrect) solution that maximized pairs of variables successively and then outputted the max value of the product of the cached values. That is, the solution computed $(\max_{x_1, x_2} P(x_1)P(x_1|x_2)) \times (\max_{x_2, x_3} P(x_2|x_3)) \times$ $... \times (\max_{x_{n-1}, x_n} P(x_n|x_{n-1}))$

**Problem 3: Bayesian networks (6 points)**

Let us try to relax the definition of Bayesian networks by removing the assumption that the directed graph is acyclic. Suppose we have a directed graph $G = (V, E)$ and discrete random variables $X_1, \cdots, X_n$, and define

$$f(x_1, \cdots, x_n) = \prod_{v \in V} f_v(x_v|x_{pa(v)})$$

where $pa(v)$ refers to the parents of variable $X_v$ in $G$ and $f_v(x_v|x_{pa}(v))$ specifies a distribution over $X_v$ for every assignment to the parents of $X_v$, i.e. $0 \leq f_v(x_v|x_{pa}(v)) \leq 1$ for all $x_v \in Val(X_v)$ and $\sum_{x_v \in Val(X_v)} f_v(x_v|x_{pa}(v)) = 1$. Recall that this is precisely the definition of the joint probability distribution associated with the Bayesian network $G$, where the $f_v$ are the conditional probability distributions. Show that if $G$ has a directed cycle, $f$ may no longer define a valid probability distribution. In particular, give an example of a cyclic graph $G$ and distributions $f_v$ that lead to improper probability distributions. Remember, a valid probability distribution must be non-negative and sum to one. This is why Bayesian networks must be defined on acyclic graphs.

**Solution:** A simple counterexample could be $P(X, Y) = P(X \mid Y)P(Y \mid X)$, where $P(X = 0 \mid Y = 0) = P(X = 1 \mid Y = 1) = 1$, and $P(Y = 0 \mid X = 1) = P(Y = 1 \mid X = 0) = 1$.

**Problem 4: Conditional Independence (12 points)**

The question investigates the way in which conditional independence relationships affect the amount of information needed for probabilistic calculations. Let $\alpha$, $\beta$, and $\gamma$ be three random variables.

- (6 points) Suppose we wish to calculate $\Pr(\alpha|\beta, \gamma)$ and we have no conditional independence information. Which of the following sets of numbers is sufficient for the calculation?

  1. $\Pr(\beta, \gamma)$, $\Pr(\alpha)$, $\Pr(\beta|\alpha)$ and $\Pr(\gamma|\alpha)$.
  2. $\Pr(\beta, \gamma)$, $\Pr(\alpha)$ and $\Pr(\beta, \gamma|\alpha)$
  3. $\Pr(\beta|\alpha)$, $\Pr(\gamma|\alpha)$ and $\Pr(\alpha)$.

  For each case, justify your response either by showing how to calculate the desired answer or by explaining why this is not possible.

  **Solution:**

  $$\Pr(\alpha|\beta, \gamma) = \frac{\Pr(\alpha, \beta, \gamma)}{\Pr(\beta, \gamma)} = \frac{\Pr(\beta, \gamma|\alpha)\Pr(\alpha)}{\Pr(\beta, \gamma)}$$

  1. No, in this case we cannot retrieve the information encoded in the full joint distribution.
  2. Yes, we have all of the pieces that we need.
  3. No, this is a subset of the first set and suffers from the same lack of information.

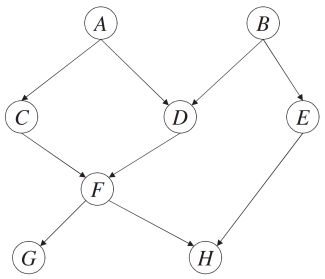
- (6 points) Suppose we know that $\beta$ and $\gamma$ are conditionally independent given $\alpha$. Now which of the preceeding three sets is sufficient. Justify your response as before.

  **Solution:**

  $$\Pr(\alpha|\beta, \gamma) = \frac{\Pr(\alpha, \beta, \gamma)}{\Pr(\beta, \gamma)} = \frac{\Pr(\beta, \gamma|\alpha)\Pr(\alpha)}{\Pr(\beta, \gamma)} = \frac{\Pr(\beta|\alpha)\Pr(\gamma|\alpha)\Pr(\alpha)}{\Pr(\beta, \gamma)} = \frac{\Pr(\beta|\alpha)\Pr(\gamma|\alpha)\Pr(\alpha)}{\sum_\alpha \Pr(\beta|\alpha)\Pr(\gamma|\alpha)\Pr(\alpha)}$$

  1. Yes
  2. Yes
  3. Yes, by marginalization in the denominator.

**Problem 5: Bayesian networks (AD Exercise 4.1) (5 points)**




| $A$ | $\Theta_A$ |
|---|---|
| 1 | .2 |
| 0 | .8 |

| $B$ | $\Theta_B$ |
|---|---|
| 1 | .7 |
| 0 | .3 |

| $B$ | $E$ | $\Theta_{E\|B}$ |
|---|---|---|
| 1 | 1 | .1 |
| 1 | 0 | .9 |
| 0 | 1 | .9 |
| 0 | 0 | .1 |

| $A$ | $B$ | $D$ | $\Theta_{D\|AB}$ |
|---|---|---|---|
| 1 | 1 | 1 | .5 |
| 1 | 1 | 0 | .5 |
| 1 | 0 | 1 | .6 |
| 1 | 0 | 0 | .4 |
| 0 | 1 | 1 | .1 |
| 0 | 1 | 0 | .9 |
| 0 | 0 | 1 | .8 |
| 0 | 0 | 0 | .2 |

Consider the Bayesian network $\mathcal{B}$ given above.

1. (2 points) Compute $\Pr(A = 0, B = 0)$ and $\Pr(E = 1|A = 1)$. Justify your answers.
2. (3 points) True or false? Why?
   (a) d-sep$_{\mathcal{B}}(A; E|\{B, H\})$
   (b) d-sep$_{\mathcal{B}}(G; E|D)$
   (c) d-sep$_{\mathcal{B}}(\{A, B\}; \{G, H\}|F)$

**Solution:**

1. $\Pr(a, b, c, d, e, f, g, h) = \Pr(G = g|F = f)\Pr(H = h|F = f, E = e)\Pr(E = e|B = b)\Pr(F = f|C = c, D = d)\Pr(C = c|A = a)\Pr(D = d|A = a, B = b)\Pr(A = a)\Pr(B = b) = \Theta_{g|f}\Theta_{h|f,e}\Theta_{e|b}\Theta_{f|c,d}\Theta_{c|a}\Theta_{d|a,b}\Theta_a\Theta_b$.
2. Upon marginalizing $G, H, F, C, D$ and $E$ in sequence, we end up with $A$ and $B$ which are independent. So, $\Pr(A = 0, B = 0) = \Pr(A = 0)\Pr(B = 0) = (0.8)(0.3) = 0.24$. For computing $\Pr(E = 1|A = 1)$, note that we can marginalize $G, H, F, C$ and $D$ out, which d-separates $A$ and $E$. Hence, $\Pr(E = 1|A = 1) = \Pr(E = 1) = \sum_b \Pr(E = 1|B = b)\Pr(B = b) = (0.9)(0.3) + (0.1)(0.7) = 0.34$.
3. (a) False. Conditioning on $H$ activates the v-structure in $F \to H \leftarrow E$, which allows information to flow to $A$ via $C, D$.
   (b) False. The v-structure $A \to D \leftarrow B$ in $G \leftarrow F \leftarrow C \leftarrow A \to D \leftarrow B \to E$ gets activated upon conditioning on $D$.
   (c) False. Although conditioning on $F$ isolates $G$ from the rest of graph, $E$ allows information to flow between $A, B$ and $H$.

**Problem 6: Bayesian Networks and explaining away (7 points)** You want to model the admission process of Farm University. Students are admitted based on their Creativity ($C$) and Intelligence ($I$). You decide to model them as continuous random variables, and your data suggests that both are uniformly distributed in $[0, 1]$, and are independent of each other. Formally $I \sim \text{Uniform}[0, 1]$, $C \sim \text{Uniform}[0, 1]$, $C \perp I$. Being very prestigious, the school only admits students such that $C + I \geq 1.5$.

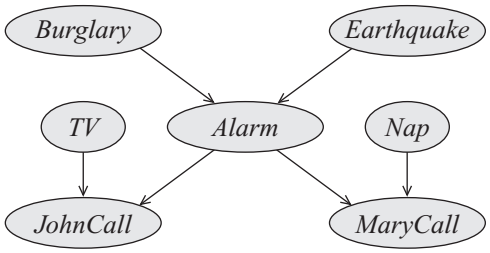
1. (1 points) What's the expected Creativity score of a student?
2. (2 points) What's the expected Creativity score of an admitted student?
3. (2 points) What's the expected Creativity score of a student with $I = 0.95$ (a highly intelligent student)?
4. (2 points) What's the expected Creativity score of an admitted student with $I = 0.95$? How does it compare to the expected Creativity score of an admitted student (computed in 2)?

Hint: it might be helpful to think about the correlation between Creativity and Intelligence in the general student population and among admitted students.

**Solution:**

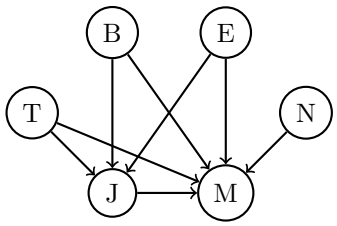
1. $\mathbb{E}[C] = \int_0^1 cdc = \frac{1}{2} = 0.5$.
2. $\mathbb{E}[C \mid C + I \geq 1.5] = \int_{0.5}^1 cp(c \mid \exists i \in [0, 1].c + i \geq 1.5) = \int_{0.5}^1 c(8(c - 0.5)) = \frac{5}{6} = 0.8\dot{3}$.
3. $\mathbb{E}[C \mid I = 0.95] = \mathbb{E}[C] = \frac{1}{2} = 0.5$ since $C \perp I$.
4. $\mathbb{E}[C \mid I = 0.95, C + I \geq 1.5] = \mathbb{E}[C \mid I = 0.95, C \geq 0.55] = \mathbb{E}[C \geq 0.55] = \int_{0.55}^1 cdc = 0.775 < 0.8\dot{3}$. Explaining away.

**Problem 7: Bayesian networks (Exercise 3.11 from Koller-Friedman) (16 points)**




1. (8 points) Consider the Burglary Alarm network given above. Construct a Bayesian network over all the node **except** the Alarm that is a minimal I-map for the marginal distribution over the remaining variables (namely, over $B, E, N, T, J, M$). Be sure to get all the dependencies from the original network.

   **Solution:**



In order to construct a minimal I-map, we would like to preserve all independencies (assuming Alarm is always unobserved) that were present in the original graph, without adding any unnecessary edges. Let's start with the remaining nodes and add edges only as needed. We see that with Alarm unobserved, there exist active paths between Alarm's direct ancestors and children. Thus, direct edges between the parents, Burglary and Earthquake, should be added to connect to both children, JohnCall and Mary Call. Similarly, since any two children of Alarm also now have an active path between them, a direct edge between JohnCall and MaryCall should be added. Without loss of generality, we direct this edge to go from JohnCall to MaryCall. Next, since SportsOnTv and JohnCall as well as Naptime and MaryCall were directly connected in the original graph, removing Alarm doesn't affect their dependencies and the two edges must be preserved. Now we must consider any independencies that may have changed. In the old graph, because of the v-structure between Alarm and co-parent SportsOnTv, if Alarm was unobserved and JohnCall observed, there existed an active path between SportsOnTv and MaryCall. In the new graph however, because of the added direct edge between the two children JohnCall andMaryCall, if JohnCall is observed, the path between SportsOnTv and MaryCall is actually rendered inactive. Thus, an alternate path that does not introduce any other dependencies needs to be introduced, and a direct edge is added between SportsOnTv and MaryCall.

2. (8 points) Generalize the procedure you used above to an arbitrary network. More precisely, assume we are given a network BN, an ordering $X_1, \cdots, X_n$ that is consistent with the ordering of the variables in BN, and a node $X_i$ to be removed. Specify a network $BN'$ such that $BN'$ is consistent with this ordering, and such that $BN'$ is a minimal I-map of $P_{BN}(X_1, \cdots, X_i, X_{i+1} \cdots X_n)$. Your answer must be an explicit specification of the set of parents for each variable in $BN'$.

**Solution:**

A general node elimination algorithm goes as follows. Suppose we want to remove $X$ from $BN$. This is similar to skipping $X$ in the I-map construction process. As the distribution in question is the same except for marginalizing $X$, the construction process is the same until we reach the removed node. Suppose the algorithm now adds the first direct descendent of $X$, which we'll call $E$. What arcs must be added? Well, all the original parents of $E$ must be added (or we'd be asserting incorrect independence assumptions). But how do we replace the arc between $X$ and $E$? As before, we must add arcs between the parents of $X$ and $E$ – this preserves the v-structure d-separation between $X$ and its parents as well as the dependence of $E$ on the parents of $X$. Now suppose we add another direct descendant, called $F$. Again, all the original parents of $F$ must be added and we also connect the parents of $X$ to $F$. Is that all? No, we must alsoconnect $E$ to $F$, in order to preserve the dependence that existed in the original graph (as an active path existed between $E$ and $F$ through $X$). Now is that all? No. Notice that if $E$, is observed then there is a active path between $C$ and $F$. But this path is blocked in our new graph if all other parents of $F$ $(E, A, B, D)$ are observed. Hence, we have to add and edge between $C$ and $F$. So for every direct descendant of $X$, we add arcs to it from the parents of $X$, from the other direct descendants of $X$ previously added and from the parents of the previously added direct descendants of $X$. What about the remaining nodes in the ordering? No changes need to be made for the arcs added for these nodes: if a node $X_m$ was independent of $X_1, \cdots, X_{m-1}$(including $X$), given its parents, it is also independent of $\{X_1, \cdots, X_{m-1}\} - \{X\}$ given its (same set of) parents. Guaranteeing that the local Markov assumptions hold for all variables is sufficient to show that the new graph has the required I-map properties. The following specification captures this procedure.

Let $T = X_1, ..., X_n$ be the topological ordering of the nodes and let $C = X_j | X_j \in Children_{X_i}$ be the set of children of $X_i$ in $BN$. For each node $X'_j$ in $BN'$ we have the following parent set:

$$Pa'_{X_j} = \begin{cases} Pa_{X_j} & \text{for all } X_j \notin C \\ Pa_{X_j} \cup Pa_{X_i} \cup \{X_k, Pa_{X_k} | X_k \in C, k < j\} - \{X_i\} & \text{for all } X_j \in C \end{cases}$$

**Problem 8: Towards inference in Bayesian networks (8 points)**

1. (4 points) Suppose you have a Bayes' net over variables $(X_1, \cdots, X_n)$ and all variables except $X_i$ are observed. Using the chain rule and Bayes' rule, find an efficient algorithm to compute $P(x_i|x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_n)$. In particular, your algorithm should not require evaluation of the full joint distribution.

   **Solution:**

$$P(X_i \mid X_{-i} = x_{-i}) = \frac{P(X_i \mid X_{\text{pa}(i)} = x_{\text{pa}(i)}) \prod_{j:i \in \text{pa}(j)} P(X_j = x_j \mid X_{\text{pa}(j)-i} = x_{\text{pa}(j)-i}, X_i)}{\sum_{X_i} P(X_i \mid X_{\text{pa}(i)} = x_{\text{pa}(i)}) \prod_{j:i \in \text{pa}(j)} P(X_j = x_j \mid X_{\text{pa}(j)-i} = x_{\text{pa}(j)-i}, X_i)}$$

2. (4 points) Find an efficient algorithm to generate random samples from the probability distribution defined by a Bayesian network. You can assume access to a routine that generates random samples from any given multinomial distribution. Hint: Show that for any joint distribution $P(X, Y)$ you can sample by first drawing a sample $x \sim P(X)$ and then drawing a sample $y \sim P(Y|X = x)$.

**Solution:**

$$P(X) = \prod_i P(X_i \mid X_{\text{pa}(i)})$$

Topologically sort $X$ such that it's consistent with the graph defined by pa. Then by order for each $X_i$ in $X$, sample $X_i = x_i$ from $P(X_i \mid X_{\text{pa}(i)} = x_{\text{pa}(i)})$.

Using induction based on the hint (prove $P(x, y) = P(x)P(y \mid x)$)

**Problem 9: Programming assignment (35 points)**

In this programming assignment, we will investigate the structure of the binarized MNIST dataset of handwritten digits using Bayesian networks. The dataset contains images of handwritten digits with dimensions $28 \times 28$ (784) pixels. Consider the Bayesian network in Figure 1. The network contains two layers of variables. The variables in the bottom layer, $X_{1:784}$ denote the pixel values of the flattened image and are referred to as *manifest variables*. The variables in the top layer, $Z_1$ and $Z_2$, are referred to as *latent variables*, because the value of these variables will not be explicitly provided by the data and will have to be inferred.

The Bayesian network specifies a joint probability distribution over binary images and latent variables $p(Z_1, Z_2, X_{1:784})$. The model is trained so that the marginal probability of the manifest variables, $p(x_{1:784}) = \sum_{z_1, z_2} p(z_1, z_2, x_{1:784})$ is high on images that look like digits, and low for other images. We consider a model parameterized using neural networks, trained using stochastic gradient descent. Bayesian networks specified as such are popularly referred to as variational autoencoders and represent one of the most powerful existing deep generative models in current use. We will return to the exact details of learning such models later in the course.

For this programming assignment, we provide a pretrained model `trained_mnist_model`. The starter code `pa1.py` loads this model and provides functions to directly access the conditional probability tables. Further, we simplify the problem by discretizing the latent and manifest variables such that $Val(Z_1) = Val(Z_2) = \{-3, -2.75, \ldots, 2.75, 3\}$ and $Val(X_j) = \{0, 1\}$, i.e., the image is binary.

1. (2 points) How many values can the random vector $X_{1:784}$ take, i.e., how many different $28 \times 28$ binary images are there?

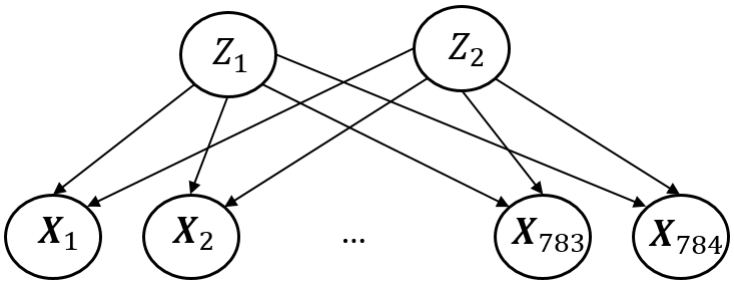



Figure 1: Bayesian network for the MNIST dataset. $X_{1:784}$ variables correspond to pixels in an image. $Z_1$ and $Z_2$ variables are latent.

**Solution:**
$2^{784}$

2. (2 points) How many parameters would you need to specify an arbitrary probability distribution over all possible $28 \times 28$ binary images?

**Solution:**
$2^{784} - 1$

3. (4 points) How many parameters do you need to specify the Bayesian network in Figure 1?

**Solution:**
$25 \times 25 \times 784 + 2 \times (25 - 1)$

For parts 4-7 below, refer to `pa1.py`. The starter code contains some helper functions for solving these questions. It is not compulsory to use them and you are allowed to use your own implementations, nor are the helper functions sufficient so feel free to introduce your own functions if required.

4. (5 points) Produce 5 samples from the joint probability distribution $(z_1, z_2, x_{1:784}) \sim p(Z_1, Z_2, X_{1:784})$, and plot the corresponding images (values of the pixel variables).

Hint: they should look like (binarized) handwritten digits. Imagine we could build such a model not for handwritten digits, but for Renaissance paintings. Each sample from the model would produce a new piece of art!

**Solution:**
Figure 2.

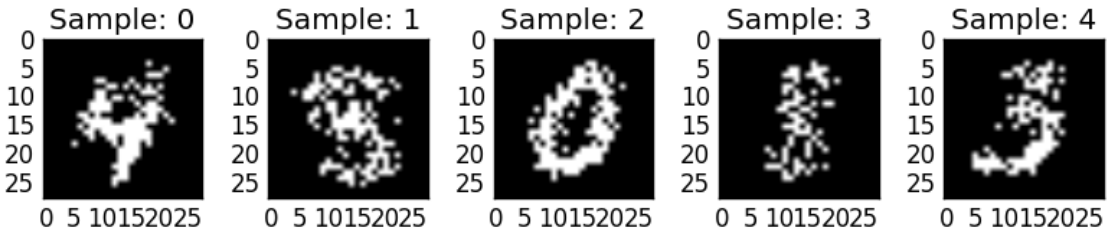



Figure 2

5. (5 points) For each possible value of

$$(\overline{z}_1, \overline{z}_2) \in \{-3, -2.75, \ldots, 2.75, 3\} \times \{-3, -2.75, \ldots, 2.75, 3\},$$

compute the conditional expectation $E[X_{1:784}|Z_1, Z_2 = (\overline{z}_1, \overline{z}_2)]$. This is the expected image corresponding to each possible value of the latent variables $Z_1, Z_2$. Plot the images on on a 2D grid where the grid axes correspond to $Z_1$ and $Z_2$ respectively. What is the intuitive role of the $Z_1, Z_2$ variables in this model?

**Solution:**
Figure 3.

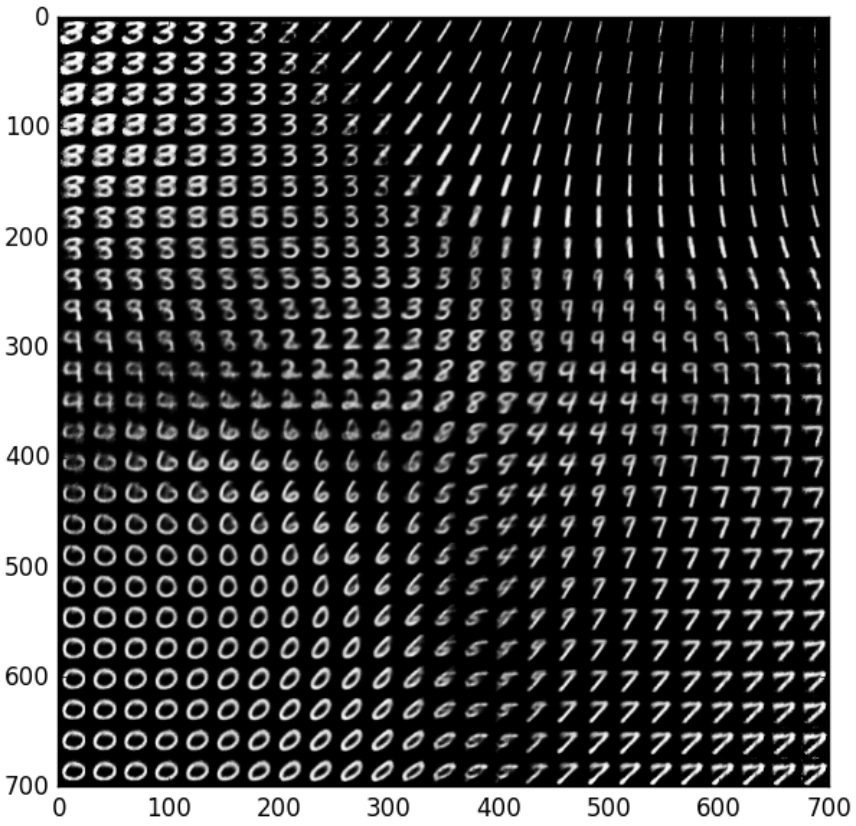



Figure 3

The latent variables here provide a compressed representation that statistically correlates the patterns in the manifest variables. Any other semantic interpretation (curve, thickness etc.) also acceptable.

6. (10 points) In `q6.mat`, you are given a *validation* and a *test* dataset. In the test dataset, some images are "real" handwritten digits, and some are anomalous (corrupted images). We would like to use our Bayesian network to distinguish real images from the anomalous ones. Intuitively, our Bayesian network should assign low probability to corrupted images and high probability to the real ones, and we can use this for classification. To do this, we first compute the average

marginal log-likelihood,

$$\log p(x_{1:784}) = \log \sum_{z_1} \sum_{z_2} p(z_1, z_2, x_{1:784})$$

on the validation dataset, and the standard deviation (again, standard deviation over the validation set). Consider a simple prediction rule where images with marginal log-likelihood, $\log p(x_{1:784})$, outside three standard deviations of the average marginal log-likelihood are classified as corrupted. Classify images in the test set as corrupted or real using this rule. Then plot a histogram of the marginal log-likelihood for the images classified as "real". Plot a separate histogram of the marginal log-likelihood for the images classified as "corrupted".

Hint: If you run into any flow issues, search for the "log-sum-exp trick" online for help.

**Solution:**

Figure 4.

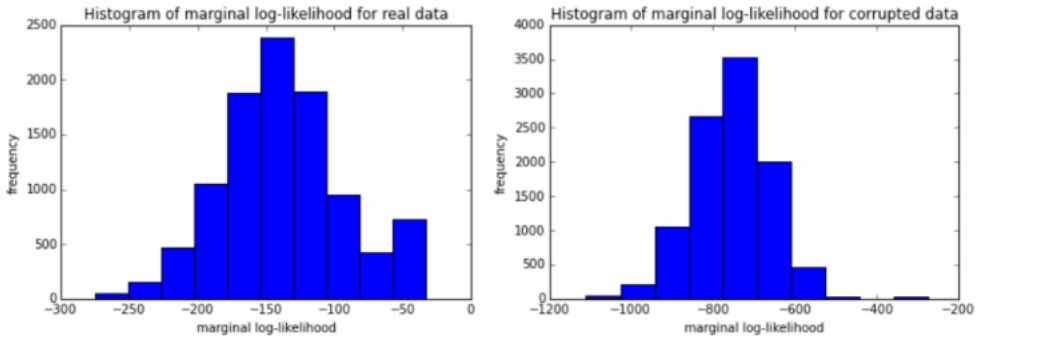



Figure 4

7. (7 points) In `q7.mat`, you are given a labeled dataset of images of handwritten digits (the label corresponds to the digit identity). For each image $I^k$, compute the conditional probabilities $p((Z_1, Z_2) = (\overline{z}_1, \overline{z}_2)|X_{1:784} = I^k)$. Use these probabilities to compute the conditional expectation

$$E[(Z_1, Z_2)|X_{1:784} = I^k)]$$

Plot all the conditional expectations in a single plot, color coding each point as per their label. What is the relationship with the figure you produced for part 5?

**Solution:**

Figure 5.

By Bayes Rule', the posterior probability is directly proportional to the likelihood which leads to a similar clustering of points for the conditional expectations in part 5 and 7.

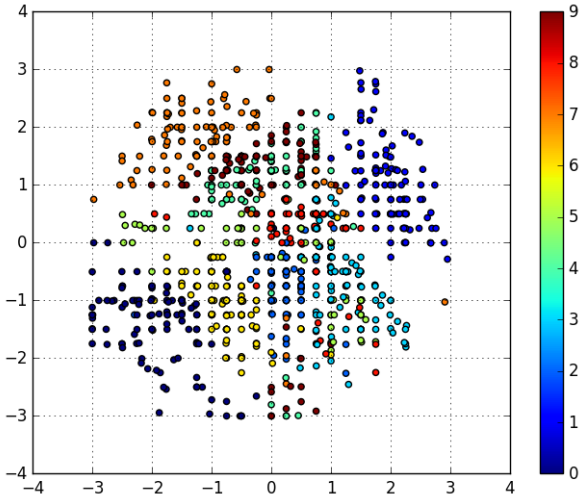



Figure 5