Name: _____ **Andrew ID:** _____

Instructions

- Anything on paper is OK in arbitrary shape size, and quantity.
- Electronic devices are not acceptable. This includes iPods, iPads, Android tablets, Blackberries, Nokias, Windows phones, Microsoft Surface, laptops, Chromebooks, MP3 players, digital cameras, Google Glass, Android Wear, or anything else that requires electricity.[1]
- **If we catch you cheating or using any such device, the exam is over for you. Your results will not count and you will have failed this exam automatically. There might be more serious consequences, too. No exceptions. Switch off your phones before the exam.**
- There are more questions in the exam than what you are likely to be able to solve in 90 minutes time. Choose wisely and answer those first that you believe you can solve easily. This is an opportunity.
- None of the questions should require more than 1 page to answer. Its OK to be concise if you address the key points of the derivation (it should be human-readable, though).

Point Distribution

Problem	Points
1	/10
2	/10
3	/15
4	/10
5	/15
6	/10
7	/15
Total	/85

[1] Obvious exceptions for pacemakers and hearing aids.

1 Loss, Regularization and Optimization [10 points]

1.1 Quick Questions [4 points]

Explain in **one or two sentences** why the statements are true (or false).

1. L2 loss is more robust to outliers than L1 loss.

Solution:
False
The gradient of L2 loss can grow without bound whereas the L1 loss gradient is bounded, hence the influence of an outlier is limited.

2. Logistic loss is better than L2 loss in classification tasks.

Solution:
True
With logistic loss, correctly classified points that are far away from the decision boundary have much less impact on the decision boundary

3. In terms of feature selection, L2 regularization is preferred since it comes up with sparse solutions.

Solution:
False
L1 regularization (LASSO) comes up with sparse solutions due to nonvanishing gradient at 0.

1.2 Gradient Descent [6 points]

Denote by $x \in \mathbb{R}^d$ data, by $w \in \mathbb{R}^d$ the weight vector, by $y \in \mathbb{R}$ labels, by $\lambda > 0$ a regularization constant, and by m the total number of data points. Let $R(w)$ be the regularized risk function:

$$R(w) := \frac{1}{m}\sum_{i=1}^{m} l\left(y_i - \langle w, x_i\rangle\right) + \frac{\lambda}{2}\|w\|^2 \text{ where } l(\xi) = \begin{cases} \frac{1}{2}\xi^2 & \text{if } |\xi| < 1 \\ |\xi| - \frac{1}{2} & \text{otherwise} \end{cases}$$

1. Calculate the batch gradient of $R(w)$ with respect to w.

Solution:
Define g_i as the gradient of ith data point.

$$g_i = \begin{cases} (\langle w, x_i\rangle - y)x_i & \text{if } |\langle w, x_i\rangle - y| < 1 \\ \text{sgn}(\langle w, x_i\rangle - y)x_i & \text{otherwise} \end{cases}$$

$$\frac{\partial R}{\partial w} = \frac{1}{m}\sum_{i=1}^{m} g_i + \lambda w$$

2. Write out a stochastic gradient descent learning algorithm for minimizing $R(w)$

Solution:
Randomly select a training instance x_i, update w as follows:

$$w \leftarrow (1 - \lambda\eta_t)w - \eta_t g_i$$

3. Name two common choices for choosing the learning rate and briefly explain their properties.

Solution:
$O(\frac{1}{\sqrt{t}})$, $O(\frac{1}{t})$ for strong convexity, Polynomial decay, AdaGrad, etc.

2 Neural Networks

2.1 Quick Questions [3 points]

Provide a **brief** answer to the following questions (1-2 sentences).

1. State one advantage of linear rectified activation compared to logistic sigmoid activation.

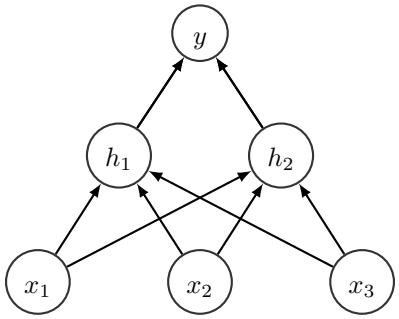
   Solution:
   Stable gradient, sparse activation, easy computation, etc.

2. Does it make sense to initialize all weights in a deep network to 0.

   Solution:
   No.
   Symmetry breaking: If all weights are equal, their gradients will be the same as well, and the gradients will possibly vanish. Hence all neurons will learn the same feature.

2.2 Forward and Backward Propagation [7 points]

The following graph shows the structure of a simple neural network with a single hidden layer. The input layer consists of three dimensions $x = (x_1, x_2, x_3)$. The hidden layer includes two units $h = (h_1, h_2)$. The output layer includes one unit y. We ignore bias terms for simplicity.



We use linear rectified units $\sigma(z) = \max(0, z)$ as activation function for the hidden and the output layer. Moreover, denote by $l(y, t) = \frac{1}{2}(y - t)^2$ the loss function. Here t is the target value for the output unit y.

Denote by W and V weight matrices connecting input and hidden layer, and hidden layer and output respectively. They are initialized as follows:

$$W = \begin{bmatrix} 1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix} \text{ and } V = \begin{bmatrix} 0 & 1 \end{bmatrix} \text{ and } x = [1, 2, 1] \text{ and } t = 1.$$

Also assume that we have at least one sample (x, t) given by the values above.

1. Write out *symbolically* (no need to plug in the specific values of W and V just yet) the mapping $x \to y$ using σ, W, V.

   Solution:

   $$y = \sigma(V\sigma(Wx))$$

2. Assume that the current input is $x = (1, 2, 1)$. The target value is $t = 1$. Compute the *numerical* output value y, clearly show all intermediate steps. You can reuse the results of the previous question. Using

matrix form is recommended but not mandatory.

Solution:

$$h = \sigma(Wx) = (2, 0)^\top$$
$$y = \sigma(Vh) = 0$$

3. Compute the gradient of the loss function with respect to the weights. In particular, compute the following terms *symbolically*:
   - The gradient relative to V, i.e. $\frac{\partial l}{\partial V}$
   - The gradient relative to W, i.e. $\frac{\partial l}{\partial W}$
   - Compute the values *numerically* for the choices of W, V, x, y given above.

Solution:
The gradients are computed with the chain rule as follows:

$$\frac{\partial l}{\partial V} = \left(\frac{\partial y}{\partial V}\right)^\top \frac{\partial l}{\partial y}$$
$$\frac{\partial l}{\partial W} = \left(\frac{\partial h}{\partial W}\right)^\top \left(\frac{\partial y}{\partial h}\right)^\top \frac{\partial l}{\partial y}$$

Plugging in numerical values yields

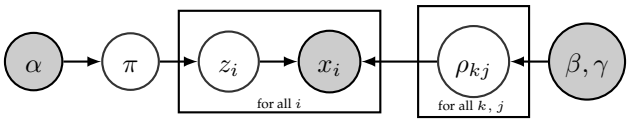
$$\frac{\partial l}{\partial V} = \left(\frac{\partial Vh}{\partial V}\right)^\top \left(\frac{\partial y}{\partial Vh}\right)^\top \frac{\partial l}{\partial y} = h^\top g(y - t) = [-2g, 0]$$

where $0 \leq g \leq 1$ is the subgradient of ReLU

$$\frac{\partial l}{\partial W} = \left(\frac{\partial Wx}{\partial W}\right)^\top \left(\frac{\partial h}{\partial Wx}\right)^\top \left(\frac{\partial y}{\partial h}\right)^\top \frac{\partial l}{\partial y} = MV^\top(y - t)x^\top = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ where } M = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

3 Expectation Maximization for Clustering Bit Vectors [15 points]

Assume that we observe binary vectors $x_i \in \{0,1\}^d$, e.g. $x_i = \boxed{0}\boxed{0}\boxed{1}\boxed{1}\boxed{1}\boxed{0}\boxed{0}\boxed{0}\boxed{1}$. Our goal is to find a generative model that represents these vectors. A mixture of binary distributions might be a good idea. In particular, we pick the following graphical model:



The associated generative model is given as follows:

1. For each topic $k \in \{1, ..., K\}$ and for each coordinate $j \in \{1, \dots d\}$
   (a) Draw $\rho_{kj} \sim \mathsf{Beta}(\beta, \gamma)$
2. Draw mixture weights $\pi \sim \mathsf{Dirichlet}(\alpha)$
3. For each observation $i \in \{1, ..., N\}$
   (a) Draw a component index $z_i \sim \mathsf{Categorical}(\pi)$
   (b) Draw each coordinate $x_{ij} \sim \mathsf{Binomial}(\rho_{z_i j})$

3.1 Generative Model

Write the joint probability $p(\pi, z, x, \rho | \alpha, \beta, \gamma)$ for a set of n vectors $\{x_1, \dots, x_n\}$ drawn from this distribution. Hint, the Beta and Dirichlet distributions are given by

$$\mathsf{Dirichlet}(\pi|\alpha) = \frac{\Gamma(k\alpha)}{\Gamma^k(\alpha)} \prod_{k=1}^{K} \pi_k^{\alpha-1} \text{ and } \mathsf{Beta}(\rho|\beta,\gamma) = \frac{\Gamma(\beta+\gamma)}{\Gamma(\beta)\Gamma(\gamma)} \rho^{\beta-1}(1-\rho)^{\gamma-1}$$

Solution:

$$\begin{aligned}
p(\pi, z, x, \rho | \alpha, \beta, \gamma) &= p(\pi|\alpha) \prod_{i=1}^{N} p(z_i|\pi) \prod_{j=1}^{d} p(x_{ij}|\rho_{z_i j}) \prod_{k=1}^{K} p(\rho_{kj}|\beta,\gamma) \\
&= \frac{\Gamma(\beta+\gamma)}{\Gamma(\beta)\Gamma(\gamma)} \frac{\Gamma(k\alpha)}{\Gamma^k(\alpha)} \prod_{k=1}^{K} \pi_k^{\alpha-1} \prod_{i=1}^{N} \pi_{z_i} \prod_{j=1}^{d} \rho_{z_i j}^{x_{ij}} (1-\rho_{z_i j})^{1-x_{ij}} \prod_{k=1}^{K} \rho_{kj}^{\beta-1} (1-\rho_{kj})^{\gamma-1} \\
&= \frac{\Gamma(\beta+\gamma)}{\Gamma(\beta)\Gamma(\gamma)} \frac{\Gamma(k\alpha)}{\Gamma^k(\alpha)} \prod_{k=1}^{K} \pi_k^{n_k+\alpha-1} \prod_{j=1}^{d} \rho_{kj}^{\beta-1+m_{kj}} (1-\rho_{kj})^{\gamma-1+n_k-m_{kj}}
\end{aligned}$$

Last step follows if we define $n_k = \sum_{i=1}^{N} \delta_{(z_i=k)}$, $m_{kj} = \sum_{i=1}^{N} \delta_{(z_i=k)} x_{ij}$.

3.2 Inference

1. Explain why it is nontrivial to infer the posterior $\pi, z, \rho | \alpha, \beta, \gamma, x$ directly.
2. Suggest at least one alternative approach to Expectation Maximization for solving this problem.

Solution:
3.2.1: There is no closed form solution $p(x)$ which is the numerical normalizer.
3.2.2. Gibbs Smapling or any other MCMC methods.

3.3 Expectation Maximization Algorithm

Use the variational inequality

$$\log p(a) \geq \mathbf{E}_{b\sim q(b)}\left[\log p(a,b)\right] + H[q]$$

to derive a lower bound on $p(x,\pi,\rho|\alpha,\beta,\gamma)$. Hint — you need to make an approximation for the distribution over z when obtaining the variational distribution.

Solution:

$$\begin{aligned}
& a = \rho, \pi,\ b = z \\
\Rightarrow \log p(x,\pi,\rho|\alpha,\beta,\gamma) &= \log \sum_z p(x,\pi,z,\rho|\alpha,\beta,\gamma) \\
&\geq \mathbf{E}_{z\sim q(z)}\left[\log p(\pi,z,\rho)\right] + H[q] \\
&= \mathbf{E}_{z\sim q(z)}\left[\sum_k^K (n_k+\alpha-1)\log\pi_k + \sum_{j=1}^d (\beta-1+m_{kj})\log\rho_{kj} + (\gamma-1+n_k-m_{kj})\log(1-\rho_{kj})\right] \\
&\quad + H[q] + \text{constant}
\end{aligned}$$

3.4 Expectation Step

Compute $p(z_i = k|x_i,\pi,\rho)$.

Solution:

$$\begin{aligned}
p(z_i = k|x_i,\pi,\rho) &= \frac{p(z_i=k|\pi)\prod_{j=1}^d p(x_{ij}|\rho_{kj})}{\sum_{k=1}^K p(z_i=k|\pi)\prod_{j=1}^d p(x_{ij}|\rho_{kj})} \\
&\propto \pi_k \prod_{j=1}^d \rho_{kj}^{x_{ij}}(1-\rho_{kj})^{1-x_{ij}}
\end{aligned}$$

3.5 Maximization step

Using $q_i(k) = p(z_i = k|x_i,\pi,\rho)$ compute the estimates for π and ρ. Hint — if you didn't manage to derive the explicit expression for $q_i(z_i)$, you may plug in the term symbolically and give the update in terms of the variational distribution q only.

Solution:

- π:

$$\hat{\pi}_k = \frac{\sum_i q_i(k) + \alpha - 1}{\sum_{i,k} q_i(k) + K(\alpha - 1)} .$$

- ρ: Recall froms 3.3 that

$$\log p(x, \pi, \rho | \alpha, \beta, \gamma) \geq = \mathbf{E}_{z \sim q(z)} \left[\sum_k^K (n_k + \alpha - 1) \log \pi_k + \sum_{j=1}^d (\beta - 1 + m_{kj}) \log \rho_{kj} + (\gamma - 1 + n_k - m_{kj}) \log(1 - \rho_{kj}) \right] + H[q] + \text{constant}$$

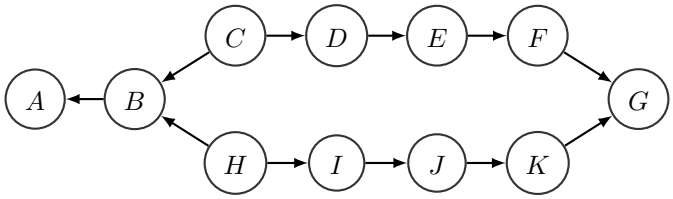
Maximizing this lower bound leads to

$$\rho_{kj} = \frac{\beta + \sum_i q_i(k) x_{ij} - 1}{\beta + \gamma + \sum_i q_i(k) - 2}$$

4 Graphical Models [10 points]

4.1 Independence [5 points]

Consider the following graphical model:



Check whether the following independence holds or not? Provide a brief reason for your answer. Alternatively, give a path for the Bayes ball algorithm as illustration.

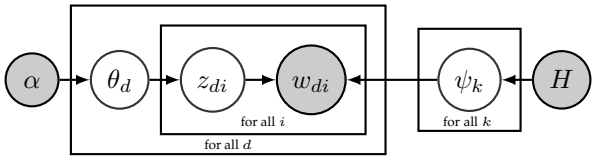
- $J \perp E|G$
- $J \perp E|K, G$
- $J \perp E|B$
- $J \perp E|A$
- $J \perp E|A, H$

Solution:

- $J \perp E|G$: No, explaining away at I
- $J \perp E|K, G$: Yes, Interaction block
- $J \perp E|B$: No, explaining away
- $J \perp E|A$: No, flow of influence
- $J \perp E|H, A$: Yes, observation block

4.2 Relevant Random Variables [5 points]

Consider the following graphical model:



1. For the given graphical model as it is, denote the minimal set of random variables which affect $z_{di} = k$ for a given d, i pair?
2. Suppose we integrate out the random variables θ_d and ψ_k. Which nodes affect $z_{di} = k$ for a given d, i pair now?

Solution:

- Depends on w_{di}, θ_{dk} and ψ_k.
- Depends on $\{w_{d'i'} : z_{d'i'} = k\}$.

Markov blanket!

5 Regression with Gaussian Prior [15 points]

Assume that we have some parameter $w \sim \mathcal{N}(w_0, \sigma^2 \mathbf{1})$ drawn from a d-dimensional Normal distribution. Moreover, assume that we perform regression with additive Normal noise ϵ. That is, assume that we have

$$y = \langle w, \phi(x) \rangle + \xi \text{ where } \xi \sim \mathcal{N}(0, \tau^2) \text{ and } w \sim \mathcal{N}(w_0, \sigma^2 \mathbf{1}) \text{ and } \phi(x) \in \mathbb{R}^d.$$

5.1 Gaussian Process

Prove that $y|x$ is drawn from a Gaussian Process. That is, show that $y|x$ is normal with a suitable mean function $\mu(x)$ and covariance kernel $k(x, x')$.

Solution:
y is a linear combination of w and ξ, each of which are Normal.

$$\begin{aligned}
\mu(x) &= \mathbb{E}[y] = \mathbb{E}[\langle w, \phi(x) \rangle + \xi] = \langle w_0, \phi(x) \rangle + 0 = \langle w_0, \phi(x) \rangle \\
k(x, x') &= \mathbb{E}[(y - \langle w_0, \phi(x) \rangle)(y' - \langle w_0, \phi(x') \rangle)^T] \\
&= \mathbb{E}[(\langle w, \phi(x) \rangle + \xi - \langle w_0, \phi(x) \rangle)(\langle w, \phi(x') \rangle + \xi - \langle w_0, \phi(x') \rangle)^T] \\
&= \mathbb{E}[(\langle w - w_0, \phi(x) \rangle + \xi)(\langle w - w_0, \phi(x') \rangle + \xi)^T] \\
&= \mathbb{E}[\langle w - w_0, \phi(x) \rangle \langle w - w_0, \phi(x') \rangle] + \mathbb{E}[\xi]\mathbb{E}[\langle w - w_0, \phi(x) \rangle] + \mathbb{E}[\xi]\mathbb{E}[\langle w - w_0, \phi(x') \rangle] + \mathbb{E}[\xi^2] \\
&= \phi(x)^T \mathbb{E}[(w - w_0)(w - w_0)^T] \phi(x') + \tau^2 \\
&= \sigma^2 \phi(x)^T \phi(x') + \tau^2 \\
&= \sigma^2 \langle \phi(x), \phi(x') \rangle + \tau^2
\end{aligned}$$

5.2 Kernels

What is the difference between a kernel obtained from the feature map $\phi(x)$ described above and a kernel such as $k(x, x') = \exp(-\frac{1}{2} \|x - x'\|^2)$. Can you characterize the type of functions that can be approximated? Hint: what does the rank of the kernel matrix tell you?

Solution:
The data point is mapped to an infinite dimensional feature space in case of $k(x, x') = \exp(-\frac{1}{2} \|x - x'\|^2)$. Polynomials of order d.

5.3 Posterior Distribution

Assume that we observe m pairs (x_i, y_i). Show that the posterior distribution is normal.

$$p\left(w | (x_1, y_1), \ldots (x_m, y_m), \tau^2, \sigma^2\right)$$

Solution:

$$\begin{aligned}
p\left(w | (x_1, y_1), \ldots (x_m, y_m), \tau^2, \sigma^2\right) &\propto p\left(w, (x_1, y_1), \ldots (x_m, y_m), \tau^2, \sigma^2\right) \\
&= p\left(w | \sigma^2\right) \prod_i p(y_i | x_i, w, \tau^2)
\end{aligned} \tag{1}$$

Each of them is Gaussian, so overall will be Gaussian.

5.4 Mean and Covariance

Compute the mean and covariance of the posterior distribution $p\left(w|(x_1, y_1), \dots (x_m, y_m), \tau^2, \sigma^2\right)$.

Solution:

$$\begin{aligned} \mathbb{E}[w|(x_1, y_1), \dots (x_m, y_m), \tau^2, \sigma^2] &= w_0 + \sigma^2 \Phi^T (\sigma^2 \Phi\Phi^T + \tau^2 \mathbf{1})^{-1}(y - \Phi w_0) \\ \mathbf{C}_{w|(x_1,y_1),\dots(x_m,y_m),\tau^2,\sigma^2} &= \sigma^2 \mathbf{1} - \sigma^4 \Phi(\sigma^2 \Phi\Phi^T + \tau^2 \mathbf{1})^{-1}\Phi \end{aligned} \tag{2}$$

where $\Phi = [\phi(x_1); \phi(x_2); ...; \phi(x_m)]$ and $y = [y_1; y_2; ...; y_m]^T$.

5.5 Prediction

Assume that we want to estimate $y'|x'$ using the previous observations, i.e. we regress on

$$p\left(y'|x', (x_1, y_1), \dots (x_m, y_m), \tau^2, \sigma^2\right)$$

Derive the mode of $p(y'|x', \text{rest})$.

Solution:
Mode of the posterior predictive $p(y'|x', \text{rest})$ is simply

$$\hat{y'} = \phi(x')^T (w_0 + \sigma^2 \Phi^T (\sigma^2 \Phi\Phi^T + \tau^2 \mathbf{1})^{-1}(y - \Phi w_0))$$

6 Matrix Factorization [10 points]

Recommendations can be generated by a wide range of algorithms. While user-based or item-based similarity methods are simple and intuitive, matrix factorization techniques are usually more effective because they allow us to discover the latent features underlying the interactions between users and items.

That is, for observed ratings r_{um} for a given (u, m) pair of a user u and a movie m, one typically tries to estimate the score by

$$f_{um} = \langle v_u, w_m \rangle + b_u + b_m.$$

Here v_u and w_m are vectors in $\mathbb{R}^d$ and b_u, b_m are scalars, indicating the bias.

6.1 Bias Updates

Assume that our objective is given by

$$\frac{1}{2} \sum_{u \sim m} (f_{um} - r_{um})^2 + \frac{\lambda}{2} \left[\sum_{u \in U} b_u^2 + \|v_u\|^2 + \sum_{m \in M} b_m^2 + \|w_m\|^2 \right] \text{ where } \lambda > 0.$$

Here U denotes the set of all users, M the set of all movies, and $u \sim m$ represents the sum over all (u, m) pairs for which a rating exists. Write the optimal values of b_u, provided that all other values are fixed. That is, compute the optimal value of $b_u | v, w, b_m, r$.

Solution:
Set the derivative wrt. a particular user u' to 0:

$$\sum_{u' \sim m} (f_{u'm} - r_{u'm}) + \lambda b_{u'} = 0$$

$$b_{u'} = \frac{\sum_{u' \sim m} (r_{u'm} - f_{u'm})}{\lambda}$$

where u' m are the movies rated by u'.

6.2 Optimization

Is the problem jointly convex in v and w? You need to prove this rather than just giving a binary answer. It suffices to prove this a very simplistic case, say for only 1 user and 1 movie.

Solution:
Since we assume only one user and one movie, by ignoring the bias terms, the Hessian of $O(f(v, w)) = \frac{1}{2} \sum_{u \sim m} (f_{um} - r_{um})^2$ can be calculated as:

$$\begin{bmatrix} 2w^2 & 4vw - 2r_{um} \\ 4vw - 2r_{um} & 2v^2 \end{bmatrix}$$

It is not positive semi-definite. Therefore the problem is only element-wise convex but not jointly convex in v and w.

6.3 Cold Start

1. How could you address the problem of recommending movies to a new user?

Solution:
When a new user come in, we have little information about them, and thus the matrix factorization method can not learn much associations between the new user and the existing users. We should use the demographics information of the user to bridge its associations with existing users. Many ideas can be applied here. The most intuitive way is perhaps to do regression based on the demographics features and compute a similarity between the new user and existing users, then approximate v_u with a linear combination.
Active learning?

2. How could you accomplish this for a new movie?

Solution:
Using meta data of the movie as additional information to encode the similarity, perhaps approximating the corresponding w_m as a linear combination of existing movies based on their similarities in terms of meta information. This can be encoded in the objective function.
Active learning?

7 Weather Prediction [15 points]

The weather in Pittsburgh is notoriously fickle. For simplicity we only consider sun and rain and we assume that the weather changes once per day. The weather satisfies the following transition probabilities:

- When it rains, the probability of sun the following day is 0.6.
- When the sun shines, the probability of rain the following day is 0.3.

7.1 Transition Matrix [3 points]

Write down the state transition matrix T for Pittsburgh's weather.

Solution:

	r	s
r	0.4	0.6
s	0.3	0.7

7.2 Stationary Distribution [5 points]

1. What are the eigenvalues of T? Bonus points if you can find stationary distribution.

   Solution:

   The first eigenvalue is 1 with eigenvector $< 1, 1 >$.
   The second eigenvalue is 0.1 with eigenvector $< 1, -0.5 >$.
   Solve a linear system $\bar{\pi}T = \bar{\pi}$ with the constraint that $\bar{\pi}_1 + \bar{\pi}_2 = 1$.
   The stationary point is that $P(r) = \frac{1}{3}$, $P(s) = \frac{2}{3}$

2. What does this mean for numerically computing the stationary distribution to at least 1% accuracy.

   Solution:
   With a large finite or countably infinite states, the standard way of computing the stationary distribution is not possible. One good example is to compute the PageRank scores of the web.
   We can use power iterations or MCMC which sinmulates random walks to find approximations to the stationary distribution.
   In terms of MCMC, with fixed parameters the Monte Carlo Standard Error (MCSE) is bounded by the sample size. We can compute a Confidence Interval regarding how far our estimation is from the truth.

7.3 Unobserved Days [7 points]

Assume that we observe the weather over a ten day period. In particular, we observe the following:

- The sun shines on the first day.
- It rains on day 5.
- It rains on day 7.
- The sun shines on day 10.

1. What is the probability of sun on day 6? Derive the equation.

Solution:
Given day 5 and day 7 are rainy ,

$$P(r_6|r_5, r_7) = 0.4 * 0.4 = 0.16$$
$$P(s_6|r_5, r_7) = 0.6 * 0.3 = 0.18$$
$$P(r_6) = \frac{0.18}{0.16 + 0.18} = \frac{9}{17}$$

2. What is the most likely *weather sequence* on days 8 and 9. Derive the equation rather than just stating the result.

Solution:
Given day 7 is rainy and day 10 is sunny, enumerate all possible sequence and evaluate their likelihoods.

$$P(r_8, r_9|r_7, s_10) = 0.4 * 0.4 * 0.6 = 0.096$$
$$P(r_8, s_9|r_7, s_10) = 0.4 * 0.6 * 0.7 = 0.168$$
$$P(s_8, r_9|r_7, s_10) = 0.6 * 0.3 * 0.6 = 0.108$$
$$P(s_8, s_9|r_7, s_10) = 0.6 * 0.7 * 0.7 = 0.294$$

Thereore the most likely sequence is $s_8 \rightarrow s_9$

(This page was left blank intentionally)

(This page was left blank intentionally)