10-715 Advanced Introduction to Machine Learning

Homework 3 Solutions *November 22, 2014*

1 Dimensionality Reduction (Samy)

1.1 Principal Components Analysis

First note that if $X = U\Sigma V^\top$ then the eigendecomposition of the empirical covariance matrix $S = \frac{1}{n}X^\top X$ is $V\left(\frac{1}{n}\Sigma^2\right)V^\top$. Therefore eigenvectors of S are the right singular vectors of X.

1. We wish to maximize $\frac{1}{n}\sum_i a_1^\top x_i x_i^\top a_1 = a_1^\top \Sigma a_1$ subject to the constraint $\|a_1\| = 1$. This is the first eigenvector of Σ.
2. Let $\tilde{X} = X - XAA^\top = X(I - AA^\top)$. We want to find orthonormal a_{k+1} so as to maximize $J(a) = a^\top(I - AA^\top)^\top \mathbf{X}^\top \mathbf{X}(I - AA^\top)a$. Any $a \in \mathcal{R}(a_1, \ldots, a_k)$ has $J(a) = 0$ so we can formulate the problem as maximize $a_{k+1}^\top \Sigma a_{k+1}$ subject to the constraints $\|a_{k+1}\| = 1$ and $a_{k+1} \perp a_i, i = 1, \ldots, k$. This is the $(k+1)^{st}$ eigenvector of Σ.

1.2 Affine Subspace Identification

First observe that we can write the objective as

$$J(A, b, Z) = \|X - ZA^\top - \mathbf{1}b^\top\|_F^2$$

Further the two constraints on Z can be written as $Z^\top \mathbf{1} = \mathbf{0}$ and $Z^\top Z = n\Psi$.

1. We will show that $Z_1 A_1^\top + \mathbf{1}b_1^\top = Z_2 A_2^\top + \mathbf{1}b_2^\top$, so both values will achieve the same objective.

$$\begin{aligned} Z_2 A_2^\top + \mathbf{1}b_2^\top &= (Z_1 C^\top + \mathbf{1}d^\top)C^{-T}A_1^\top + \mathbf{1}(b_1 - A_1 C^{-1}d)^\top \\ &= Z_1 C^\top C^{-\top} A_1^\top + \mathbf{1}d^\top C^{-\top} A_1^\top + \mathbf{1}b_1^\top - \mathbf{1}^\top d C^{-\top} A_1^\top \\ &= Z_1 A_1^\top + \mathbf{1}b_1^\top \end{aligned}$$

2. First note that the problem does not constrain b in any way so we can take the derivative and set it to zero.

$$\begin{aligned} \mathbf{0} = \nabla_b J &= 2(X - ZA^\top - \mathbf{1}b^\top)^\top \mathbf{1} \\ &= X^\top \mathbf{1}^\top - AZ^\top \mathbf{1} - b^\top \mathbf{1}^\top \mathbf{1} \\ b &= \frac{1}{n}X^\top \mathbf{1} = \bar{X} \end{aligned}$$

Here, first note that $\mathbf{1}^\top \mathbf{1} = n$. If we can find Z to satisfy the constraint $Z^\top 1 = 0$ then the last step holds. We will assume this and then show that we can find such a Z.

To minimize w.r.t A, Z, note that $ZA^\top$ needs to be the best rank k approximation to $X - \mathbf{1}b^\top$ in Frobenius norm. We can do this by first taking the SVD of $\tilde{X} = X - \mathbf{1}b^\top$ and then zeroing out the last $\min\{n, d\} - k$ singular values. Let $\tilde{X} = U\Sigma V^\top$ be the SVD. Here, $U \in \mathbb{R}^{n\times n}$, $\Sigma \in \mathbb{R}^{n\times D}$ is diagonal and $V \in \mathbb{R}^{D\times D}$. Denote the first d columns of U and V by U_d, V_d and the top $d \times d$ block of Σ by Σ_d. The rank d approximation is given by, $ZA^\top = U_d \Sigma_d V_d^\top$.

Now if we choose $Z = \sqrt{n} U_d \Psi^{1/2}$ and $A^\top = \frac{1}{\sqrt{n}} \Psi^{-1/2} \Sigma_d V_d^\top$

$$Z^\top Z = n \Psi^{1/2} U_d^\top U_d \Psi^{1/2} = n\Psi$$

Finally we need to show that this Z satisfies $Z^\top \mathbf{1} = \mathbf{0}$. For this first note that $\tilde{X}^\top \mathbf{1} = V \Sigma U^\top \mathbf{1} = 0$ which implies $U^\top \mathbf{1} = 0$ since $V\Sigma$ is full rank (as $\dim(\text{span}(X)) > d$). Then $U_d^\top \mathbf{1} = \mathbf{0} \implies Z^\top \mathbf{1} = \mathbf{0}$.

3. $z_* = A^\dagger (x_* - b)$, where $A^\dagger$ is the MP inverse. This gives the projection of $x_* - b$ onto the column space of A.

1.3 Factor Analysis

1. First note that we can write x conditioned on z as $x|z = Az + b + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \eta^2 I)$.
The joint distribution will be Gaussian since x is just a linear transformation of z. To specify the joint distribution we should know the mean of x, the variance of x and the covariance between x and z. They can be computed as follows.

$$\begin{aligned}
\mathbb{E}[x] &= \mathbb{E}[Az + b + \epsilon] = \mathbf{0} + b = b \\
\mathbb{E}\left[(z - \mathbb{E}[z])(x - \mathbb{E}[x])^\top\right] &= \mathbb{E}\left[z(x-b)^\top\right] = \mathbb{E}[z(Az + b + \epsilon)^\top] - \mathbb{E}[zb^\top] \\
&= \mathbb{E}[zz^\top A^\top + zb^\top + z\epsilon^\top] = \Psi A^\top \\
\mathbb{E}\left[(x - \mathbb{E}[x])(x - \mathbb{E}[x])^\top\right] &= \mathbb{E}\left[(x-b)(x-b)^\top\right] = \mathbb{E}\left[(Az + \epsilon)(Az + \epsilon)^\top\right] \\
&= \mathbb{E}\left[Azz^\top A^\top + 2\epsilon z^\top A^\top + \epsilon\epsilon^\top\right] = A\Psi A^\top + \eta^2 I
\end{aligned}$$

Therefore, we can write

$$\begin{bmatrix} z \\ x \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ b \end{bmatrix}, \begin{bmatrix} \Psi & \Psi A^\top \\ A\Psi^\top & A\Psi A^\top \eta^2 I \end{bmatrix} \right)$$

Using the hints given in the question, the marginal for x and the conditional $z|x$ can be written as

$$\begin{aligned}
x &\sim \mathcal{N}(b, A\Psi A^\top + \eta^2 I) \\
z|x &\sim \mathcal{N}\left(\Psi A^\top (A\Psi A^\top + \eta^2 I)^{-1}(x - b)\ ,\ \Psi - \Psi A^\top (A\Psi A + \eta^2 I)^{-1} A \Psi^\top\right)
\end{aligned}$$

We will denote the conditional mean and variance of $z|x$ by $\mu_{z|x}$ and $\Sigma_{z|x}$ respectively.

2. The log likelihood for A, b, η given data $(x_i)_{i=1}^n$ is,

$$\begin{aligned}
\ell(A, b, \eta) &= \log \prod_{i=1}^n p(x_i) = \sum_{i=1}^n \log p(x_i) \\
&= \sum_{i=1}^n -\frac{D}{2}\log(2\pi) - \frac{1}{2}\log\det(A\Psi A^\top + \eta^2 I) - \frac{1}{2}(x_i - b)^\top (A\Psi A^\top + \eta^2 I)^{-1}(x_i - b)
\end{aligned}$$

The MLE for b can be obtained easily. By taking the derivative of ℓ w.r.t. b and setting it to 0 we have,

$$\nabla_b \ell = \sum_{i=1}^n (A\Psi A^\top + \eta^2 I)^{-1}(x_i - b) = 0 \implies b = \frac{1}{n}\sum_{i=1}^n x_i$$

since $(A\Psi A^\top + \eta^2 I)^{-1}$ is full rank.

3. To perform EM, we use Jensen's inequality to construct the following lower bound.

$$\begin{aligned}
\ell(A,b,\eta) &\leq \sum_{i=1}^{n} \int R(z_i|x_i) \log \frac{p(x_i,z_i;A,b,\Psi)}{R(z_i|x_i)} \\
&= \sum_{i=1}^{n} \mathbb{E}_{R(z_i|x_i)} \left[\log p(x_i|z_i;A,b,\eta)\right] + C \\
&= \sum_{i=1}^{n} \mathbb{E}_{R(z_i|x_i)} \left[\log \left(\frac{1}{(2\pi)^{D/2}\eta^D} \exp\left(-\frac{(x_i-b-Az_i)^\top (x_i-b-Az_i)}{2\eta^2} \right) \right)\right] + C \\
&= \sum_{i=1}^{n} \mathbb{E}_{R(z_i|x_i)} \left[\log \left(-\frac{D}{2}\log(2\pi) - D\log(\eta) - \frac{1}{2\eta^2}(x_i-b-Az_i)^\top (x_i-b-Az_i) \right)\right] + C \\
&= -nD\log(\eta) + \frac{-1}{2\eta^2} \sum_{i=1}^{n} \mathbb{E}_{R(z_i|x_i)} \left[\|x_i-b\|^2 - 2z_i^\top A^\top (x_i-b) + z_i A^\top A z_i\right] + C'
\end{aligned}$$

Here C, C' are constants that do not depend on A, b, η. Let us call this lower bound ℓ_b

4. In the M-step we will maximize the lower bound above w.r.t the parameters A, η simply by taking the derivative and setting it to zero. First take the derivative w.r.t A–using the hints given in the question,

$$\nabla_A \ell_b(A,b,\eta) = \frac{-1}{2\eta^2} \sum_{i=1}^{n} \mathbb{E}_{R(z_i|x_i)} \left[2A(z_i z_i^\top) - 2(x_i-b)z_i^\top\right] \implies$$

$$A\left(\sum_i \mathbb{E}_{R(z_i|x_i)} z_i z_i^\top\right) = \sum_i (x_i-b)\mathbb{E}_{R(z_i|x_i)} z_i^\top \implies A = \left(\sum_i (x_i-b)\mu_{z_i|x_i}^\top\right)\left(\sum_i \mu_{z_i|x_i}\mu_{z_i|x_i}^\top + \Sigma_{z_i|x_i}\right)^{-1}$$

Similarly for η,

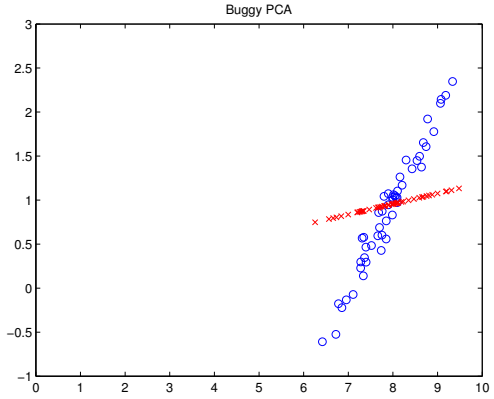
$$\frac{\partial \ell_b}{\partial \eta} = -\frac{nD}{\eta} + \frac{1}{\eta^3} \sum_{i=1}^{n} \mathbb{E}_{R(z_i|x_i)} \left[\|x_i-b\|^2 - 2z_i^\top A^\top (x_i-b) + z_i A^\top A z_i\right] \implies$$

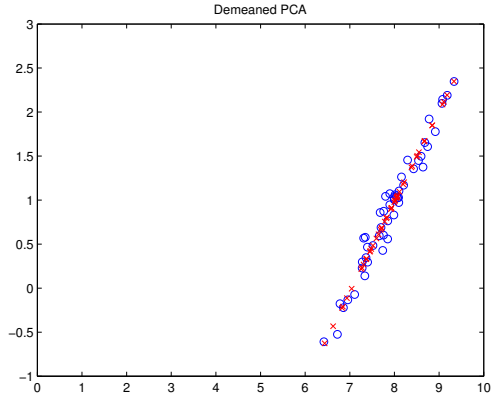
$$\eta^2 = \frac{1}{nD} \sum_{i=1}^{n} \left(\|x_i-b\|^2 - 2\mu_{z_i|x_i}^\top A^\top (x_i-b) + \|A\mu_{z_i|x_i}\|^2 + \mathrm{diag}(A\Sigma_{z_i|x_i}A^\top)^\top \mathbf{1} \right)$$

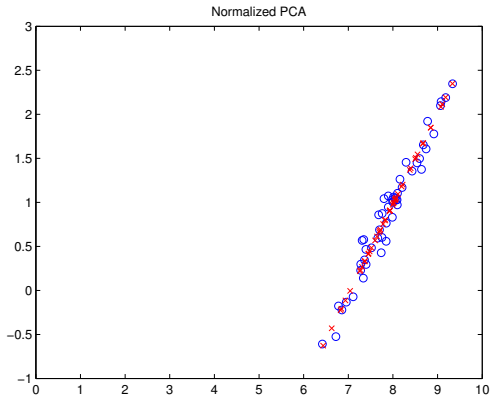
For the last step we used the fact that $\mathbb{E}_{R(z_i|x_i)} z_i = \mu_{z_i|x_i}$ and from the properties of the Gaussian, $\mathbb{E}_{R(z_i|x_i)} \|Az_i\|^2 = \|A\mu_{z_i|x_i}\|^2 + \mathrm{diag}(A\Sigma_{z_i|x_i}A^\top)^\top \mathbf{1}$. We can perform the M-step via the above update equations for A and η.

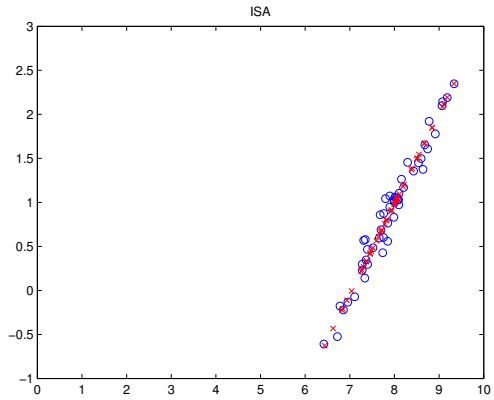
1.4 Experiment

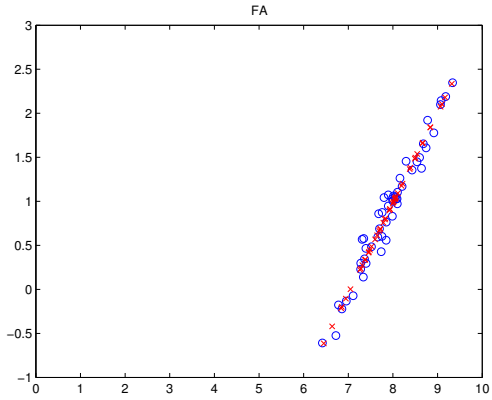
Results for the 2D dataset





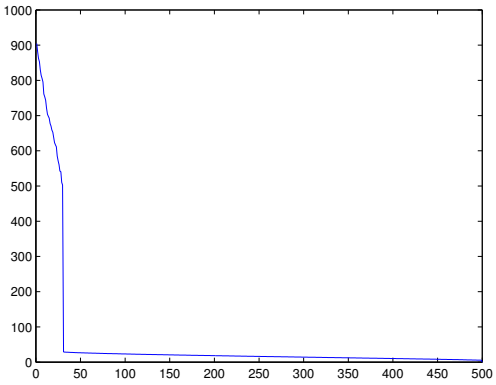






Results for the 1000D dataset

We used $d = 30$ since the singular values fall off sharply after this. See figure below:



This was the output.

```
>> q14
Reconstruction Errors:
Buggy PCA: 777.871447
Demeaned PCA: 272.546928
Normalized PCA: 273.140905
ASI: 272.546928
FA: 272.549605
```

Answers to Questions

1. When you SVD without demeaning on the dataset, you are find the best linear (as opposed to affine) subspace. The first principal component then will then be from the origin towards the data and other principal components will be orthogonal to this.
2. This is because the reconstructions are identical in both cases even if the representations are not.
3. No. Since in ASI we are directly minimizing this error criterion.

This is our implementation.

```
function [Z, params, Y] = deMeanPrinCompAnalysis(X, d)

  % First normalize the data
  meanX = mean(X);
  X_ = bsxfun(@minus, X, mean(X));

  % Now apply PCA
  [Z, params, Y_] = buggyPrinCompAnalysis(X_, d);

  % Now reconstruct
  Y = bsxfun(@plus, Y_, meanX);
  params.meanX = meanX;

end


function [Z, params, Y] = normPrinCompAnalysis(X, d)

  % First normalize the data
  meanX = mean(X);
  stdX = std(X);
  X_ = bsxfun(@rdivide, bsxfun(@minus, X, mean(X)), stdX);

  % Now apply PCA
  [Z, params, Y_] = buggyPrinCompAnalysis(X_, d);

  % Now reconstruct
  Y = bsxfun(@plus, bsxfun(@times, Y_, stdX), meanX);
  params.meanX = meanX;
  params.stdX = stdX;

end


function [Z, params, Y] = affineSubspaceIdentification(X, d)

  % prelims
  n = size(X, 1);

  b = mean(X)';
  [U,S,V] = svd( bsxfun(@minus, X, b') , 'econ');
%   plot(diag(S)),
  Ud = U(:, 1:d);
  Sd = S(1:d, 1:d);
  Vd = V(:, 1:d);
```

```
  A = (1/sqrt(n) * Sd * Vd')';

  % Z and Y
  Z = sqrt(n) * Ud;
  Y = bsxfun(@plus, Z*A', b');

  params.A = A;
  params.b = b;

end


function [Z, params, Y] = factorAnalysis(X, d)

  % prelims
  NUM_EM_ITERS = 10;
  D = size(X, 2);
  n = size(X, 1);

  % Initialize using ASI
  [~, initParams, Y] = affineSubspaceIdentification(X, d);
  b = initParams.b; % this will also be the final b
  A = initParams.A;
  eta = sqrt( mean(mean( (Y-X).^2 )) );

  for emIter = 1:NUM_EM_ITERS
    [A, eta] = emFA(X, A, eta, b);
  end
  params.A = A;
  params.b = b;
  params.eta = eta;

  % Finally obtain Z and Y
  Z = bsxfun(@minus, X, b') * (( A*A' + eta^2 *eye(D)) \ A) ;
  Y = bsxfun(@plus, Z*A', b');

end


function [ANew, etaNew, RzxMeans] = emFA(X, AOld, etaOld, b)

  % prelims
  D = size(X, 2);
  d = size(AOld, 2);
  n = size(X, 1);

  % E-step
  K = AOld*AOld' + etaOld^2 * eye(D);
  Kinv = inv(K);
  RzxMeans = bsxfun(@minus, X, b') * Kinv * AOld;
  RzxVar = eye(d) - AOld' * Kinv * AOld;
  % Compute the following which will be useful too
  EAZ = RzxMeans * AOld';
  EAZ2 = sum( EAZ.^2 , 2) + sum(diag(AOld * RzxVar * AOld'));
```

```
  Xmb = bsxfun(@minus, X, b');

  % M-step
  % First A
  M1 = Xmb' * RzxMeans;
  M2 = RzxMeans' * RzxMeans + n * RzxVar;
  ANew = M1 / M2;
  % Now eta
  etaNew = sqrt( 1/(n*D) * ( norm(Xmb, 'fro')^2 ...
                  - 2 * sum(sum( (Xmb .* EAZ) ) ) ...
                  + sum(EAZ2) ) );

end
```

2 Unsupervised Learning (Samy)

2.1 K-means Clustering

1. If $K \geq n$, we can set the first set n centres to the n data points and f to be the identity map. This gives $\mathcal{J}_K(X_1^n) = 0$ for all $K \geq n$. Now let $n < K$. Let μ_*, f_* be such that $\mathcal{J}_K(X_1^n) = J(\mu_1^K, f; X_1^n)$. Pick any point X' in the data set that is not a centre and modify f_* to $f'_*(X) = f_*(X)$ if $X \neq X'$ and $f'_*(X') = X'$. Then,

$$\mathcal{J}_K(X_1^n) \geq \mathcal{J}_{K+1}(\{\mu_1^K, X'\}, f'_*; X_1^n) \geq \min_{\mu_1^{K+1}, f} J(\mu_1^{K+1}, f; X_1^n) = \mathcal{J}_{K+1}(X_1^n)$$

2. First we will show that the objective can only decrease at each iteration. Denote $\gamma_{ij} = \mathbb{1}(f(X_i) = j)$ where f is the rule that assigns a point to the closest centre. At a given iteration, $J(\gamma, \mu) = \sum_{i=1}^n \sum_{j=1}^k \gamma_{ij} \|x_i - \mu_j\|^2 = \sum_i \|x_i - \Pi(x_i)\|$, where $\Pi(x_i) = \operatorname{argmin}_j \|x_i - \mu_j\|$ is the assignment of x_i to particular centre. In step 1 of the algorithm, we re-assign the x_i's to their closest centres. Hence,$\|x_i - \Pi(x_i)\|$ can only get smaller. In the second step we update the centres to the means of the assigned points, which can be interpreted as minimizing the squared distance to all the points.

$$\mu_j = \frac{\sum_i \gamma_{ij} x_i}{\sum_i \gamma_i} = \operatorname*{argmin}_{\mu \in \mathbb{R}^d} \sum_i \gamma_{ij} \|x_i - \mu\|^2$$

   Thus this step too can only decrease the objective. Therefore, the objective is non-increasing in each iteration.

   Since there are at most k^n assignments of points to cluster centres, the above objective can only achieve one of k^n different values and one of k^n different assignments. Therefore, it has to terminate in a finte number of steps.

2.2 Independent Components Analysis

Solutions are straightforward.

3 Graphical Models(20 points)

1. (a) $P(I, W, G, L) = P(I)P(W)P(G|I, W)P(L|G)$

(b) Without the knowledge of the graphical model, we need $d^4 - 1$ parametrize the full joint distribution. With the knowledge, we need $(d-1) + (d-1) + (d-1)(d^2) + (d_1)d = 1008$.

2. (a) No. L is a descendant of G and I, G, W form a V-structure.

   (b) No. There is an active trail between I and G after we execute the d-separation algorithm.

   (c) Yes. Any trail between I and L has to go through G but G is conditioned.

   (d) No. There is an active trail between G and L after we execute the d-separation algorithm.

3. Applying Bayes rule appropriately,

   (a)

$$\begin{aligned} P(L=1) &= \sum_{G,I,W} P(L=1,G,W,I) = \sum_{G} P(L=1|G) \sum_{I,W} P(G|I,W) \\ &= 0.3 * (0.1 * 0.06 + 0.6 * 0.24 + 0.7 * 0.14 + 0.1 * 0.56) \\ &+ 0.8 * (0.3 * 0.24 + 0.2 * 0.14 + 0.9 * 0.56) \\ &= 0.5744 \end{aligned}$$

   (b)

$$\begin{aligned} P(L=1|I=1,W=0) &= \sum_{G} P(L=1,G|I=1,W=0) = \sum_{G} P(L=1|G)P(G|I=1,W=0) \\ &= 0 + 0.3 \times 0.7 + 0.8 \times 0.2 = 0.37 \end{aligned}$$

4. (a) The probability of the sequence is 2.0646e-05 by forward algorithm.

   (b) The most likely path is $2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1$ by Viterbi decoding.

   See sample code in `viterbi.m, fwd.m`

4 Markov Chain Monte Carlo

4.1 Markov Chain properties

1. As all rows sum to 1, we have $T\mathbf{1} = 1.\mathbf{1}$ showing that 1 is an eigen value of T and hence an eigen value of T^T.

2. T is a reducible matrix, with two connected components, where the first and third states are in one component and the other two are in the second compnonent. A stationary distribution of any of the components is a stationary distribution of the whole Markov chain. The transition matrices of the individual components are

$$T_1 = \frac{1}{10}\begin{bmatrix} 3 & 7 \\ 4 & 6 \end{bmatrix}, T_2 = \frac{1}{10}\begin{bmatrix} 4 & 6 \\ 7 & 3 \end{bmatrix}$$

   and their stationary distributions are given by

$$v_1 = \frac{1}{11}\begin{bmatrix} 4 \\ 7 \end{bmatrix}, v_2 = \frac{1}{13}\begin{bmatrix} 7 \\ 6 \end{bmatrix}$$

   and so the two stationary distributions of T are given by

$$u_1 = \frac{1}{11}\begin{bmatrix} 4 \\ 0 \\ 7 \\ 0 \end{bmatrix}, v_2 = \frac{1}{13}\begin{bmatrix} 0 \\ 7 \\ 0 \\ 6 \end{bmatrix}$$

3. Consider an aperiodic matrix T, such as a permutation matrix which is not identity:

$$T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, p_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

4.2 Detailed balance property

1. **(4 points)** Suppose p is proportional to the target distribution. Let x, x' be in the domain of sampling. The detailed balance equation trivially holds if $x = x'$. Assume x, x' are distinct from now. To go from x to x', the proposal distribution $q(.|x)$ needs to generate x' and x' needs to be accepted. So, the transition kernel is given by $T(x \to x') = q(x'|x)A(x'|x)$ where $A(x'|x)$ is the acceptance probability

$$A(x'|x) = \min\{1, z(x'|x)\} \text{ where } z(x'|x) = \frac{p(x')q(x|x')}{p(x)q(x'|x)}$$

To show detailed balance, we need to show $p(x)T(x \to x') = p(x')T(x' \to x)$.

Note that $z(x'|x) = \frac{1}{z(x|x')}$. So at least one of $z(x'|x), z(x|x')$ is ≤ 1. Wlog assume $z(x'|x) \leq 1$, so that $A(x'|x) = z(x'|x)$ and $A(x|x') = 1$.

$$\begin{aligned} p(x)T(x \to x') &= p(x)q(x'|x)z(x'|x) \\ &= p(x)q(x'|x)\frac{p(x')q(x|x')}{p(x)q(x'|x)} \quad = p(x')q(x|x') = p(x')q(x|x')A(x|x') \quad = p(x')T(x' \to x) \end{aligned}$$

2. **(2 points)**

Let $p(x)T(x \to x') = p(x')T(x' \to x)$ for all x, x' in X, the domain of sampling. We claim that p is a stationary distribution of the Markov chain. For that, we need to show

$$\int_X p(x)T(x \to x')\,dx = p(x') \text{ for any } x, x' \in X$$

Let $x, x' \in X$. Integrating both sides of the detailed balance equation w.r.t x,

$$\begin{aligned} \int_X p(x)T(x \to x')\,dx &= \int_X p(x')T(x' \to x)\,dx \\ &= p(x')\int_X T(x' \to x)\,dx \\ &= p(x') \end{aligned}$$

The last equality holds because $T(x' \to .)$ is a probability density on X. Therefore p is a stationary distribution of the Markov chain.

4.3 Experiments

1. **Metropolis Hastings**
   (a) See `run_metropolis.m` for the code. With $\sigma = 0.5$, the chain gets stuck in one of Gaussians.The m sample means are close to either -5 or 5.
   (b) With $\sigma = 0.5$, the chain manages to move from one Gaussian to the other and so the sample means are closer to 0.
   (c) With $\sigma = 0.5$, as the chain gets stuck in one of the Gaussians chosen randomly, with a larger m, the average of the sample means is expected to go to 0.

2. **Gibbs sampling for Gaussian Mixture models**

(a) The first conditional distribution is given by

$$\begin{aligned} p(z_i = k|x, z_{-i}, \mu) &= \frac{p(z_i = k, x, z_{-i}, \mu)}{\sum_{k'} p(z_i = k', x, z_{-i}, \mu)} \\ &= \frac{p(x|z_i = k, z_{-i}, \mu)p(z_i = k)p(z_{-i})p(\mu)}{\sum_{k'} p(x|z_i = k', z_{-i}, \mu)p(z_i = k')p(z_{-i})p(\mu)} \\ &= \frac{p(x|z_i = k, z_{-i}, \mu)p(z_i = k)}{\sum_{k'} p(x|z_i = k', z_{-i}, \mu)p(z_i = k')} \\ &\propto p(x|z_i = k, z_{-i}, \mu)p(z_i = k) \\ &\propto p(x_i|z_i = k, \mu_k)p(z_i = k) \end{aligned}$$

The idea is to absorb the terms that do not depend on k into a proportionality constant.

The second one is given by,

$$\begin{aligned} p(\mu_k = u|x, z, \mu_{-k}) &\propto p(\mu_k = u, x, z, \mu_{-k}) \\ &\propto \prod_j p(x_j|z_j = k, \mu_k = u)^{\mathbb{I}(z_j=k)} p(\mu_k = u) \end{aligned}$$

by absorbing the terms that do not depend on u into the proportionality constant.

(b) Plugging in the known distributions, the conditionals can be written as

$$\begin{aligned} p(z_i = k|x, z_{-i}, \mu) &\propto \exp(-\|x_i - \mu_k\|^2/2) \\ p(\mu_k = u|x, z, \mu_{-k}) &\propto \exp\left(-\frac{1}{2}\left\{\|u\|^2 + \sum_{i=1}^{n} \|x_j - u\|^2 \mathbb{I}(z_i = k)\right\}\right) \end{aligned}$$

See the code in `sample_gibbs.m`.

References