Student Name: Pengcheng Xu

Collaboration Statement:

Total hours spent: 6 hours

I discussed ideas with these individuals:

- I did it on my own
- ...

I consulted the following resources:

- Course slides
- Recorded videos
- ...

By submitting this assignment, I affirm this is my own original work that abides by the course collaboration policy.

Links: [HW3 instructions] [collab. policy]

Contents

1a: Problem Statement

Define $\Sigma = LL^T$. Show the following:

$$|\det(L^{-1})| = \frac{1}{(\det\Sigma)^{\frac{1}{2}}} \tag{1}$$

1a: Solution

We'll show the derivation step-by-step and also add the comment along the way.

/* Σ has the Cholesky decomposition (i.e. $\Sigma = LL^T$) */

$$det(\Sigma) = det(LL^T)$$

/* Hint(v): $det(AB) = det(A)det(B)$, and Hint(vi): $det(L) = det(L^T)$ */

$$det(\Sigma) = det(L) \cdot det(L)$$

/* Divide $det^2(L) \cdot det(\Sigma)$ on both side */

$$\frac{1}{det^2(L)} = \frac{1}{det(\Sigma)}$$

/* Hint(iv): $det(A^{-1}) = \frac{1}{detA}$ */

$$det^2(L^{-1}) = \frac{1}{det(\Sigma)}$$

/* Squaring */

$$|det(L^{-1})| = \frac{1}{(det\Sigma)^{\frac{1}{2}}}$$

1b: Problem Statement

Show that the pdf of x is given by:

$$p(x) = \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{(\det\Sigma)^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{2}$$

1b: Solution

We'll show the derivation step-by-step and also add the comment along the way.

/* Using the fomula from the background */

$$p(x) = f(S(x))|det(J_S(x))|$$

/* Using $S(x) = L^{-1}(x-m)$, $J_S(x) = L^{-1}$, and $f(u)$'s defition that are given */

$$= (2\pi)^{\frac{-D}{2}}|L^{-1}| \cdot e^{\frac{-1}{2}(L^{-1}(x-m))^T(L^{-1}(x-m))}$$
$$= (2\pi)^{\frac{-D}{2}}|L^{-1}| \cdot e^{\frac{-1}{2}(x-m)^T(L^T)^{-1}L^{-1}(x-m)}$$

/* Hint(iii): $(AB)^{-1} = B^{-1}A^{-1}$, definition of Σ (i.e. $\Sigma = LL^T$),and result from 1a. i.e. $|det(L^{-1})| = \frac{1}{(det\Sigma)^{\frac{1}{2}}}$ */

$$= \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{(det\Sigma)^{\frac{1}{2}}} \cdot e^{\frac{-1}{2}(x-m)^T(LL^T)^{-1}(x-m)}$$

$$= \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{(det\Sigma)^{\frac{1}{2}}} \cdot e^{\frac{-1}{2}(x-m)^T\Sigma^{-1}(x-m)}$$

Thus, we've proved equation (2).

1c: Problem Statement

Complete the Python code below, to show how to turn samples from a standard 1D Gaussian, via NumPy's 'randn()' into a sample from a multivariate Gaussian.x

1c: Solution

```
import numpy as np

def sample_from_mv_gaussian(mu_D, Sigma_DD, random_state=np.random):
  ''' Draw sample from multivariate Gaussian

  Args
  ----
  mu_D : 1D array, size D
    Mean vector
  Sigma_DD : 2D array, shape (D, D)
    Covariance matrix. Must be symmetric and positive definite.

  Returns
  -------
  x_D : 1D array, size D
    Sampled value of Gaussian with provided mean and covariance
  '''
  D = mu_D.size
  L_DD = np.linalg.cholesky(Sigma_DD) # compute L from Sigma
  # GOAL: draw each entry of u_D from standard Gaussian
  u_D = random_state.randn(D) # use random_state.randn(...)
  # GOAL: Want x_D ~ Gaussian(mean = m_D, covar=Sigma_DD)
  x_D = L_DD@u_D + mu_D #  transform u_D into x_D
  return x_D
```

2a: Problem Statement

Show that the Metropolis-Hastings transition distribution $\mathcal{T}$ satisfies detailed balance with respect to the target distribution p^*.

That is, show that:

$$p^*(a)\mathcal{T}(b|a) = p^*(b)\mathcal{T}(a|b) \tag{3}$$

for all possible $a \neq b$, where a, b are any two distinct values of the random variable.

2a: Solution

In order to prove equation (3), we could prove $\frac{p^*(a)\tau(b|a)}{p^*(b)\tau(a|b)} = 1$ equivalently.
Next, we'll show the derivation step-by-step and also add the comment along the way.

/* Using definition of p and $\tau(z^{'}|z)$ from the background */

$$\frac{p^*(a)\tau(b|a)}{p^*(b)\tau(a|b)} = \frac{c\tilde{p}(a)}{c\tilde{p}(b)} \cdot \frac{Q(b|a) \cdot min(1, \frac{\tilde{p}(b)Q(a|b)}{\tilde{p}(a)Q(b|a)})}{Q(a|b) \cdot min(1, \frac{\tilde{p}(a)Q(b|a)}{\tilde{p}(b)Q(a|b)})}$$

/* Using the hint: $\frac{min(1, \frac{x}{y})}{min(1, \frac{y}{x})} = \frac{x}{y}$ */

$$= \frac{\tilde{p}(a)}{\tilde{p}(b)} \frac{Q(b|a)}{Q(a|b)} \cdot \frac{\tilde{p}(b)Q(a|b)}{\tilde{p}(a)Q(b|a)}$$

/* Could cancel out all the elements */

$$= 1$$

Thus, we've proved equation (3).

3a: Problem Statement

(See diagram on 3a Instructions web page)

You start at Medford/Tufts station, and take 1000 steps. What is your probability distribution over ending this journey at each of the 7 stations? Report as a vector (use order of nodes in the diagram, small to large). Round to 3 decimal places.

3a: Solution

We write a python script, and after setting up the initial values and running 1000 times, we get the following result:

[0.167 0.000 0.333 0.000 0.333 0.167 0.000]

Please the following as the screenshots of python script and result:

3b: Problem Statement

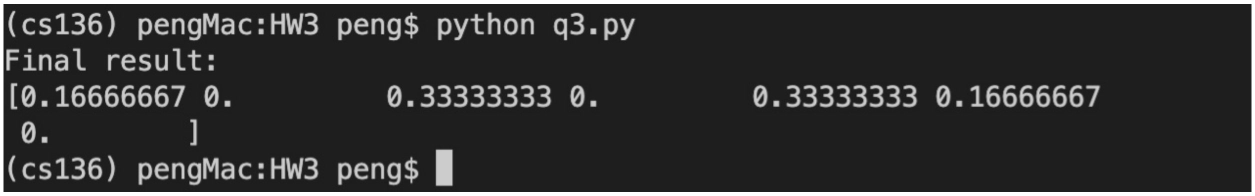
Is there a unique stationary distribution for this Markov chain? If so, explain why. If not, explain why not.

```python
q3.py > ...
import numpy as np

P = np.array(
        [[0, 1, 0, 0, 0, 0, 0],
          [0.5, 0, 0.5, 0, 0,0,0],
          [0, 0.5, 0, 0.5, 0, 0, 0],
          [0, 0, 0.5, 0, 0.5, 0, 0],
          [0, 0, 0, 0.5, 0, 0, 0.5],
          [0, 0, 0, 0, 0, 0, 1],
          [0, 0, 0, 0, 0.5, 0.5, 0],
        ]
)

K = 3333
RES = np.array([0, 0, 1, 0, 0, 0, 0])

def calc_stationary_distribution():
    for i in range(K):
        # print(i, " iteration:")
        global RES
        RES = RES @ P

    print("Final result:")
    print(RES)
calc_stationary_distribution()
```

Figure 1: 3a: python script

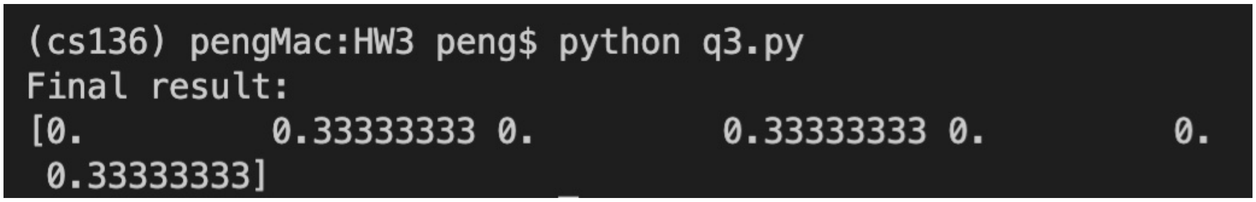
```
Final result:
[0.16666667 0.         0.33333333 0.         0.33333333 0.16666667
 0.        ]
(cs136) pengMac:HW3 peng$ 
```

Figure 2: 3a: result



```
(cs136) pengMac:HW3 peng$ python q3.py
Final result:
[0.16666667 0.         0.33333333 0.         0.33333333 0.16666667
 0.        ]
(cs136) pengMac:HW3 peng$
```

Figure 3: 3b: start at station 1 with 3000 steps



```
(cs136) pengMac:HW3 peng$ python q3.py
Final result:
[0.         0.33333333 0.         0.33333333 0.         0.
 0.33333333]
```

Figure 4: 3b: start at station 2 with 1000 steps

3b: Solution

There's no stationary distribution. If there exists a stationary distribution, then when the number of steps is big enough, the probability distribution vector will converge. However, if we experiment with it numerically, the final vector would be different if we start at a different station (i.e. other than Medford) or if we run different steps (e.g. 2000 or 3000 steps).

The following are the screenshots of starting at station 2 with 1000 steps, and starting at station 1 with 3333 steps.