Homework #0 Solution A

Spring 2020, CSE 446/546: Machine Learning
Jiahui Xi
Due: 4/8/20 11:59 PM
A: 37 points. B: 3 points

April 2020

Probability and Statistics

A.1 **Solution:**
From the problem, let's set T:=Tested positive and D:=Have disease. And we can know that $P(T \mid D) = 0.99$ and $P(D) = \frac{1}{10000}$. Now we need $P(D \mid T)$. By Bayes Rule, we have:

$$P(D \mid T) = \frac{P(D \cap T)}{P(T)} = \frac{P(T \mid D)P(D)}{P(T \mid D)P(D) + P(T \mid \text{no}D)P(\text{no}D)} = \frac{0.99\frac{1}{10000}}{0.99\frac{1}{10000} + 0.01\frac{9999}{10000}} = \frac{1}{102}.$$

A.2 **Solution:**

a. By Law of Total Expectation,

$$\begin{aligned}&Cov(X,Y)\\&= E[E[(X - E[X])(Y - E[Y]) \mid X]]\\&= E[(X - E[X])(E[Y \mid X] - E[Y] \mid X)]\\&= E[(X - E[X])(X - E[X])]\\&= E[(X - E[X])^2]\end{aligned}$$

b. if X and Y are independent, then:

$$\begin{aligned}&E[(X - E[X])(Y - E[Y])]\\&= E[X - E[X]]E[Y - E[Y]]\\&= 0\end{aligned}$$

A.3 **Solution:**

a. For the cdf of Z, $H(Z) = P(Z \le$z) = P(X+Y≤z)
$= \int_{-\infty}^{\infty} \int_{-\infty}^{z-x} f(x)g(y)dydx$
To get the density of Z we need to differentiate this with respect to Z.
The only z dependence is in the upper limit of the inside integral.
so $h(z) = \frac{\partial H}{\partial h} = \int_{-\infty}^{\infty} \frac{\partial \int_{-\infty}^{z-x} f(x)g(y)dy}{\partial z} dx$
finally we get $h(z) = \int_{-\infty}^{\infty} f(x)g(z-x)dx$

b. According to our solution above,
f(z) = z for 0¡z≤1;
f(z) = 2-z for 1¡z¡2;
f(z) = 0 otherwise.

A.4 **Solution:**

$$a = \frac{1}{\sigma}, b = -\frac{\mu}{\sigma} \tag{1}$$

A.5 **Solution:** From CLM,

$$mean = 0, variance = \sigma^2 \tag{2}$$

A.6 **Solution:**

a. E[$\widehat{F}_n(x)$] = $\frac{n \cdot E[F]}{n}$
= E[F] = F(x)

b. Notice that 1{X $\leq x\} is a Bernoulli random variable with$
P[1{X $\leq x\}] = F(x)$ so we can see $\widehat{F}_n(x)$ is just a mean of n Bernoulli variables
Hence the variance is in a form with $\frac{p(1-p)}{n}$ so it's $\frac{F(x)(1-F(x)}{n}$

c. Set $G(x) = x \cdot (1-x)$, for $0 leq$ x$leq 1$, G(x) has the maximum value at x = 0.5, and G(x) = 0.25.
So the maximum of [(F(x)] is 0.25, and we can conclude that $E[(\widehat{F}_n(x) - F(x))^2] \leq \dfrac{1}{4n}$.

Linear Algebra and Vector Calculus

A.7 **Solution:**

a. rank(A) = 2 and rank(B) = 2

b. $\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} \right\}$.

A.8 **Solution:**

a. Ac $= \begin{bmatrix} 0 & 2 & 4 \\ 2 & 4 & 2 \\ 3 & 3 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 8 \\ 7 \end{bmatrix}$

b. set $x = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$ and from Ax=b, we can get a set of linear equations.

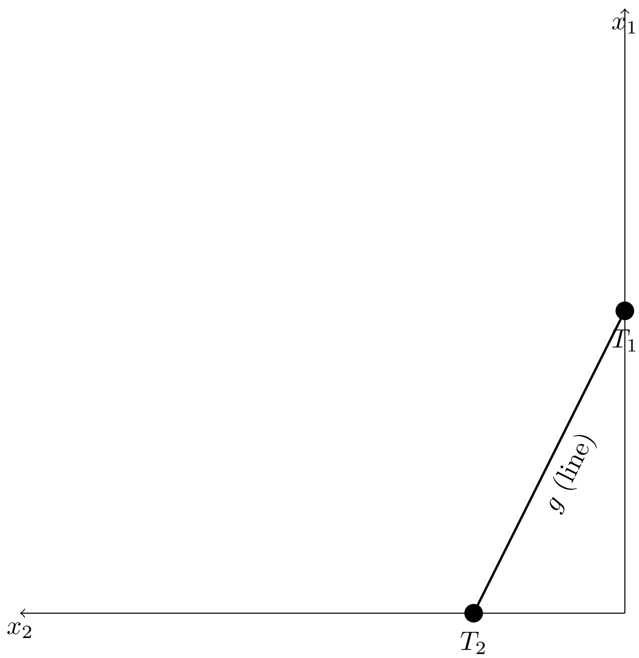
$$\begin{cases} 2b + 4c = -2 \\ (1) \\ \\ 2a + 4b + 2c = -2 \\ (2) \\ \\ 3a + 3b + c = -4 \\ (3) \end{cases}$$

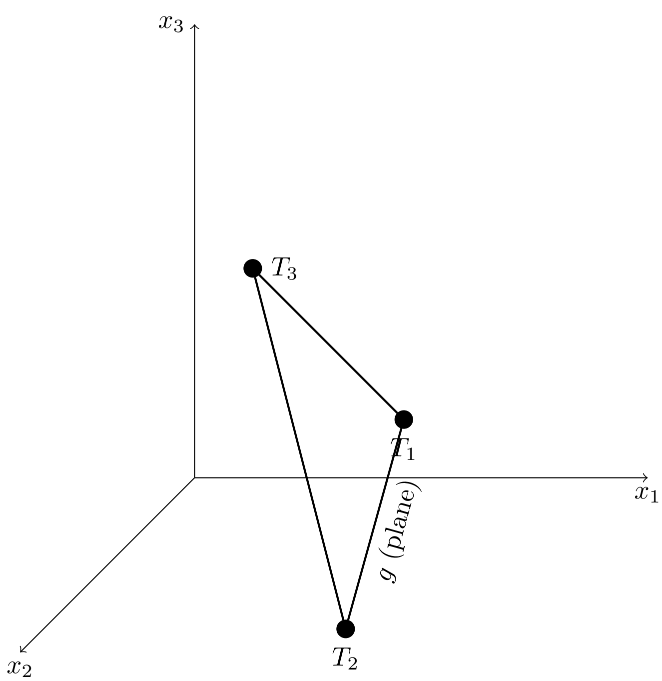
if we use $3 \cdot (2) - 2 \cdot (3)$,

we can get 6b+4c=2, and we combine this and (1)

b = 1, c = -1 and so a = -2 and fianlly$x = \begin{bmatrix} -2 \\ 1 \\ -1 \end{bmatrix}$

A.9 **Solution:**

a. 

b. 

c. according to the hint, the minimized distance is $\|x_0 - \widetilde{x}_0\| = |\frac{w^T \cdot x_0 + b}{\|w\|}|$

A.10 **Solution:**

a. $f(x,y) = \sum_{i,j=1}^{n} x_i A_{i,j} x_j + \sum_{i,j=1}^{n} y_i B_{i,j} x_j + c.$

b. $\nabla_x f(x,y) = (A + A^T)x + B^T y$

c. $\nabla_y f(x,y) = By$

A.11 **Solution:**
the solutions to both questions are included in this screenshot:

```
print(A_inv)
print(A_inv.dot(b))
print(np.dot(A, c))
```

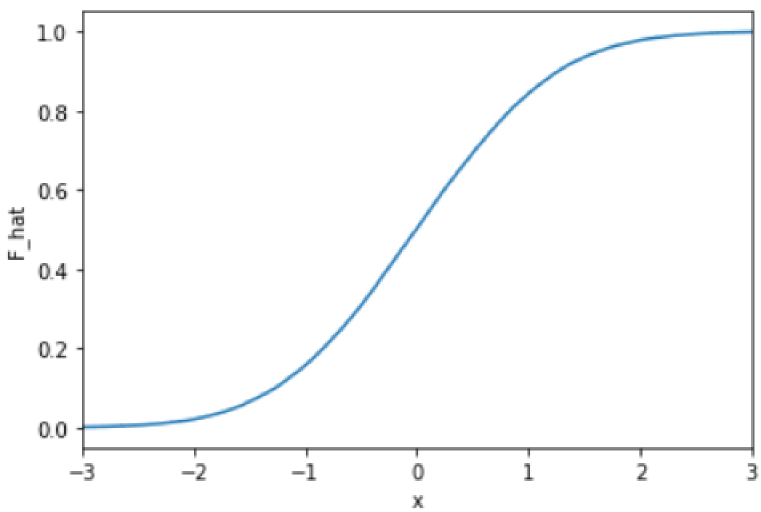
```
[[ 0.125 -0.625  0.75 ]
 [-0.25   0.75  -0.5  ]
 [ 0.375 -0.375  0.25 ]]
[[-2.]
 [ 1.]
 [-1.]]
[[6]
 [8]
 [7]]
```

Listing 1: A.11 code

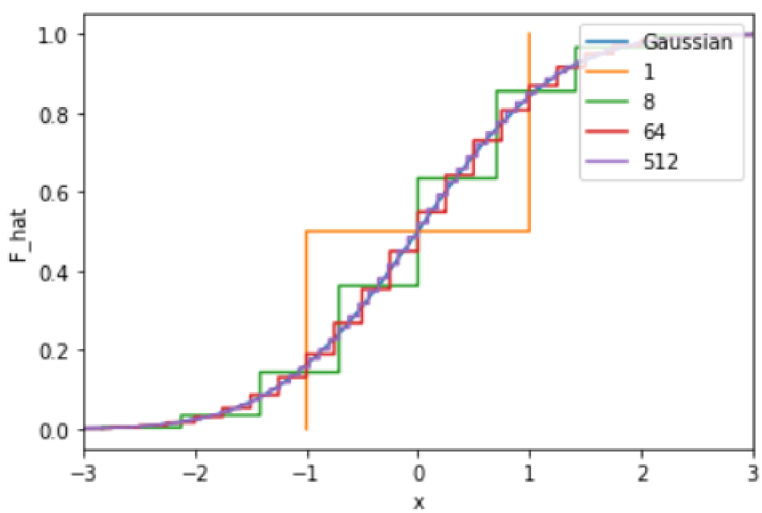
```
#!/usr/bin/env python
# coding: utf-8

# In[1]:


import numpy as np
A = ([0,2,4],
     [2,4,2],
     [3,3,1]);
b = ([-2],[-2],[-4]);
c = ([1],[1],[1]);
A_inv = np.linalg.inv(A)
print(A_inv)
print(A_inv.dot(b))
print(np.dot(A,c))


# In[ ]:
```

A.12 **Solution:**

a. Since $\frac{1}{4n} = 0.0025^2 = \frac{1}{160000}$, we can get that $n = \frac{1}{40000}$ and the plot:



b. The plot is as follows:



Listing 2: A.12 code

```
#!/usr/bin/env python
# coding: utf-8

# In[13]:


import numpy as np
n = 40000
Z=np.random.randn(n);
import matplotlib.pyplot as plt
plt.step(sorted(Z), np.arange(1,n+1)/float(n))
plt.xlim(-3,3)
plt.xlabel("x")
plt.ylabel("F_hat")

Y_1 = np.sum(np.sign(np.random.randn(n, 1))*np.sqrt(1./1), axis=1)
```

```
Y_8 = np.sum(np.sign(np.random.randn(n, 8))*np.sqrt(1./8), axis=1)
Y_64 = np.sum(np.sign(np.random.randn(n, 64))*np.sqrt(1./64), axis=1)
Y_512 = np.sum(np.sign(np.random.randn(n, 512))*np.sqrt(1./512), axis=1)
plt.step(sorted(Y_1), np.arange(1,n+1)/float(n))
plt.step(sorted(Y_8), np.arange(1,n+1)/float(n))
plt.step(sorted(Y_64), np.arange(1,n+1)/float(n))
plt.step(sorted(Y_512), np.arange(1,n+1)/float(n))
plt.legend(["Gaussian","1","8","64","512"],loc="upper right")


# In[ ]:
```