Data Science and Social Inquiry: HW1 Solution

1 Question1

(a) (2pts) What is $\hat{F}_n(4)$? How about $\hat{F}_n(-3)$?

Sol.

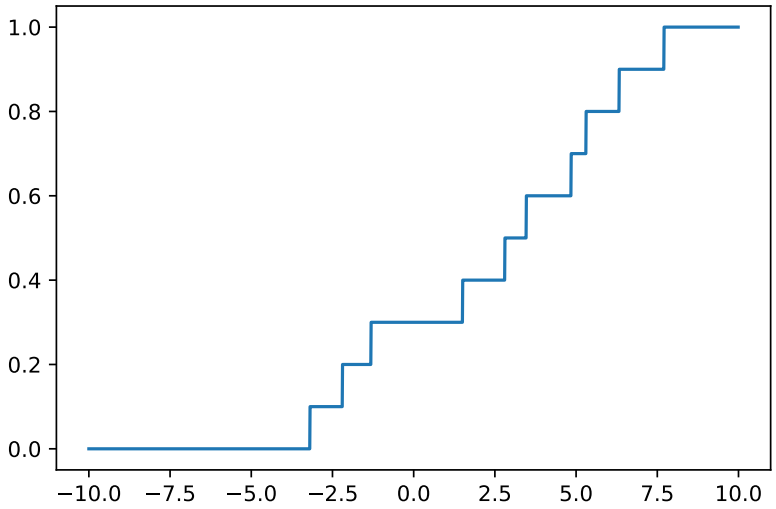
$$\hat{F}_n(4) = \frac{6}{10}, \quad \hat{F}_n(-3) = \frac{1}{10}.$$

□

(b) (1pt) In the previous part, we find $\hat{F}_n(x)$ at two points, namely $x = 4$ and $x = -3$. We can of course keep going and try other values of x, but this is rather repetitive and boring. Luckily, we have computers, and we know how to write program. Write a program to find out $\hat{F}_n(x)$ for $x = -10,\ -9.99,\ -9.98, ...,\ 9.99,\ 10$.

(c) (1pt) Use the result from (b) and plot $\hat{F}_n(\cdot)$. How does it look like? Is it non-decreasing?

Sol.



□

(d) (1pt) What is the expected value of $\hat{F}_n(0)$? Does it depend on n? **Hint**: $\mathbb{1}_{(-\infty,x]}(X_i)$ takes value only in 0 and 1. Which family of random variable only takes value in 0 and 1? What is its expected value? P.S. Your answer can be is related to $F_X(\cdot)$.

Sol.

$$\mathbb{1}_{(-\infty,0]}(X) = \begin{cases} 1 & prob = F_X(0) \\ 0 & prob = 1 - F_X(0) \end{cases}$$

Not depend on n. Bernoulli distribution. For $X \sim$ Bernoulli (p), $\mathbb{E}[X] = p$. So

$$\mathbb{E}\left[\hat{F}_n(0)\right] = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\mathbb{1}_{(-\infty,0]}(X)\right] = F_X(0)$$

□

(e) (1pt) What is the variance of $\hat{F}_n(0)$? Does it depend on n?

Sol.

$$Var\left(\hat{F}_n(0)\right) = \frac{1}{n^2}\sum_{i=1}^{n} Var\left(\mathbb{1}_{(-\infty,0]}(X)\right) = \frac{1}{n}F_X(0)(1 - F_X(0))$$

□

(f) (1pt) What happens when $n \to \infty$? Do you think $\hat{F}_n(0)$ is a good estimator of $F_X(0)$?

Sol.

Since the estimator is unbiased and $Var\left(\mathbb{1}_{(-\infty,0]}(X)\right) \to 0$, so it is consistent.

□

(g) (1pt) What is $F_X(0)$, the true value of the parameter of interest, given that $X \sim N(0,1)$?

Sol.

$$F_X(0) = \frac{1}{2}$$

□

(h) (1pt) Use the **numpy.random.normal** module to generate $X_1, X_2, ..., X_n \overset{i.i.d.}{\sim} N(0,1)$ for $n = 100$ and calculate $\hat{F}_{n,1}(0)$, where the subscript 1 means that $\hat{F}_{n,1}(0)$ is obtained in the first round of simulation. Repeat $10,000$ times and collect the estimates $\hat{F}_{n,1}(0)$, $\hat{F}_{n,2}(0)$, ..., and $\hat{F}_{n,10000}(0)$. Calculate the **mean squared error** (MSE)

$$\frac{1}{10000}\sum_{b=1}^{10000}\left[\hat{F}_{n,b}(0) - F_X(0)\right]^2,$$

which is the average squared distance between the estimator $\hat{F}_{n,b}(0)$ and its true value $F_X(0)$.

(i) (1pt) Repeat (h) with $n = 200$ and $n = 500$. Is MSE larger or smaller when n is larger?

Sol.

n=100	n=200	n=500
0.0025	0.0013	0.0005

□

(j) **(Bonus, 2pts)** the **Central Limit Theorem** (CLT) implies that

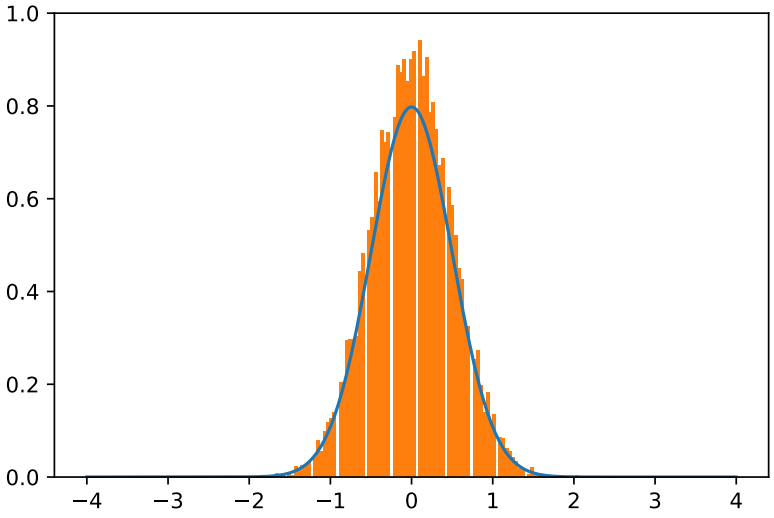
$$\sqrt{n}(\hat{F}_n(0) - F_X(0))$$

will converge to the normal distribution. We can verify that CLT holds in our case by plotting the histogram of

$$\sqrt{n}(\hat{F}_{n,b}(0) - F(0)),\ b = 1, 2, ..., 10000$$

for $n = 500$. Does your plot support CLT?

Sol.



□

2 Question2

The following is a reference answer.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px
df = pd.read_csv('/content/drive/MyDrive/hotels.csv')

### (a)
df = pd.read_csv('/content/drive/MyDrive/hotels.csv')

# Convert the ” price”  column to numerical format (float)
df['price'] = df['price'].str.replace('TWD', '').str.replace(',', '').astype(float)

# Handle missing values, if any, in the ” rating”  column.
# delete
df.dropna(subset=['rating'], inplace=True)

# Convert 'rating' column to float.
```

```
df['rating'] = df['rating'].astype(float)

# Convert ‘distance’ to integer.
# Convert all distance to meter.
def convert_distance(distance_s):
if '公尺' in distance_s:
distance = float(distance_s.split(' ')[1])
elif '公里' in distance_s:
distance = float(distance_s.split(' ')[1]) * 1000
else:
distance = None
return distance

df['distance'] = df['distance'].apply(convert_distance).astype(int)

#Remove the [,] and , from ‘comment’ .
#comment doesn't have [,]
print(df)

### (b)
# plot scatter
fig = px.scatter(df, x='price', y='distance', color='rating', hover_data=['name',
'rating', 'price'],
 labels={'price': 'price (in TWD)', 'distance': 'distance from Center (in m)'})

fig.update_traces(texttemplate='%{customdata[0]}<br>',
  textposition='top center', mode='markers+text')

fig.show()

# we can see that 台北文華東方酒店 may be a outlier
df.drop(df[df['name'] == '台北文華東方酒店'].index, inplace=True)
fig = px.scatter(df, x='price', y='distance', color='rating',hover_data=['name',
'rating', 'price'],
labels={'price': 'price (in TWD)', 'distance': 'distance from Center (in m)'})

fig.update_traces(texttemplate='%{customdata[0]}<br>',
  textposition='top center', mode='markers+text')

fig.show()
```