Statistical Inference

IFT6758 Fall 2019

Classical Inference

1. [ISLR 3.7.3] Suppose we have a dataset with five predictors, $X_1 =$ GPA, $X_2 = IQ$, $X_3 =$ Gender (1 for Female and 0 for Male), $X_4 =$ Interaction between GPA and IQ, and $X_5 =$ Interaction between GPA and Gender. The response is starting salary after graduation (in thousands of dollars). Suppose we use least squares to fit the model, and get $\hat{\beta}_0 = 50, \hat{\beta}_1 = 20, \hat{\beta}_2 = 0.07, \hat{\beta}_3 = 35, \hat{\beta}_4 = 0.01, \hat{\beta}_5 = -10$.

   a. Which is correct, and why? i. For a fixed value of IQ and GPA, males earn more on average than females. ii. For a fixed value of IQ and GPA, females earn more on average than males. iii. For a fixed value of IQ and GPA, males earn more on average than females provided that the GPA is high enough. iv. For a fixed value of IQ and GPA, females earn more on average than males provided that the GPA is high enough.

   b. Predict the salary of a female with IQ of 110 and a GPA of 4.0.

   c. True or false: Since the coefficient for the GPA / IQ interaction term is very small, there is very little evidence of an interaction effect. Justify your answer.

Answer

a) The right one is iii.)

The equation for the starting salary after graduation is : Y = 50 + 20(GPA) + 0.07(IQ) + 35(Gender) + 0.01(GPA *IQ) -10(GPA* Gender)

For male it gives us; Y = 50 + 20(GPA) + 0.07(IQ) + 0.01(GPA * IQ)

For a female it gives us ; Y = 50 + 20(GPA) + 0.07(IQ) + 35 + 0.01(GPA * IQ) -10(GPA)

Given that the GPA is high enough, even if the equation for a female has the +35 due to 35(Gender), the -10(GPA) is penalizing compared to equation for a male.

b) Y = 50 + 20 *4.0* + *0.07* 110 + 35 + 0.01(4.0 * 110) -10(4.0)

Y = 137.1

Which is 137 100$

c) False, the value of the coefficient signifies how much the mean of the dependent variable changes given a one-unit shift in the independent variable while holding the other variables in the model constant. To know whether there is an evidence of an interaction effect, we would need to know the p-value of the GPA/IQ term.

The Bootstrap

1. Suppose that $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_m$ are two independent samples. As a measure of the difference in location of the two samples, the difference of the 20% trimmed means is used (each trimmed mean is the mean after discarding the 10% smallest and 10% largest values in the group). Explain how the bootstrap could be used to estimate the standard error of this difference.

Answer

What we would need to do is to :

1.Sample with replacement from our independent sample X1,...,Xn

2.Calculate and store the trimmed mean of our new sample

3.Sample with replacement from our independent sample Y1,...,Yn

3.Calculate and store the trimmed mean of our new sample

4.Calculate the difference between the trimmed mean of our sample from X1,...,Xn and the trimmed mean of our sample from Y1,...,Yn and record that number

5.Repeat steps 1 to 4 n times

6.Calculate the standard deviation of the n recorded differences, which gives us a bootstrapped standard error of the difference

1. [ISLR 4.5.9] Consider the Boston housing dataset (https://gist.githubusercontent.com/krisrs1128/2c1ce8d004b1efc18b2d6e03e84a27c6/raw/9e95c9782b46f1ede7c28846639

   a. Based on this dataset, provide an estimate for the population mean of `medv`. Call this estimate $\hat{\mu}$.

   b. Provide an estimate of the standard error of $\hat{\mu}$. Interpret this result. *Hint: We can compute the standard error of the sample mean by dividing the sample standard deviation by the square root of the number of observations.*.

   c. Now estimate the standard error $\hat{\mu}$ using the bootstrap. How does this compare to your answer from (b)?

   d. Based on your bootstrap estimate from (c), provide a 95% confidence interval for the mean of `medv`. *Hint: You can approximate a 95% confidence interval using the formula* $[\hat{\mu} - 2s.e.(\hat{\mu}), \hat{\mu} + 2s.e.(\hat{\mu})]$.

   e. Based on this dataset, provide an estimate $\hat{\mu}_{med}$ for the median value of `medv` in the population.

   f. We now would like to estimate the standard error of $\hat{\mu}_{med}$. Unfortunately, there is no simple formula for computing the standard error of the median. Instead, estimate the standard error of the median using the bootstrap. Comment on your findings.

   g. Based on this dataset, provide an estimate for the 10th percentile of `medv` in Boston suburbs. Call this quantity $\hat{\mu}_{0.1}$.

   h. Use the bootstrap to estiamte the standard error of $\hat{\mu}_{0.1}$. Comment on your findings.

In [3]:
```
import pandas as pd
import numpy as np
import scipy.stats as st
df = pd.read_csv("https://gist.githubusercontent.com/krisrs1128/2c1ce8d004b1efc18b2d6e03
e84a27c6/raw/9e95c9782b46f1ede7c288466390c8f937aecc08/boston.csv")
np.random.seed(1)

#a
mu = np.mean(df["medv"])
print("a)")
print(mu)

#b
std = np.std(df["medv"])
print("b)")
print(std / np.sqrt(len(df["medv"])))

#c
sample_props = []
for i in range(1000):
    sample = np.random.choice(df["medv"], size=int(len(df["medv"])), replace=True)
    sample_props.append(sample.mean())


print("c)")
print(np.std(sample_props))
print("The answer is similar to answer b, having a really small difference")
std = np.std(sample_props)
mu = np.mean(sample_props)
#d
print("d)")
print((mu - 2*std, mu + 2*std))

#e
print("e)")
print(np.median(df["medv"]))

#f
sample_prop = []
for i in range(1000):
    samp = np.random.choice(df["medv"], size=int(len(df["medv"])), replace=True)
    sample_prop.append(np.median(samp))

print("f)")
print(np.std(sample_prop))
print("It's a small standard error relative to the median of 21.2")

#g
print("g)")
print(np.percentile(df["medv"], 10))

#h
sample_pro = []
for i in range(1000):
    sam = np.random.choice(df["medv"], size=int(len(df["medv"])), replace=True)
    sample_pro.append(np.percentile(sam, 10))
print("h)")
print(np.std(sample_pro))
print("It is a small standard error relative to the tenth-percentile value of 12.75")
```

```
a)
22.532806324110677
b)
0.4084569346972866
c)
0.412402701271868
The answer is similar to answer b, having a really small difference
d)
(21.701916929472073, 23.351527734559543)
e)
21.2
f)
0.3845271901959597
It's a small standard error relative to the median of 21.2
g)
12.75
h)
0.4897669624423436
It is a small standard error relative to the tenth-percentile value of 12.75
```

Large-Scale Inference

1. The data here (https://gist.githubusercontent.com/krisrs1128/ff7b6498c89316b9dd526a0f44d92d31/raw/2683592cd4ed08d00e17d12adbc are a simulation of 10,000 experiments, each seeking to detecting a difference between treatment and control. Only the last 10% of hypotheses are actually nonnull (ids 9001 to 10000).

   a. For experiment, run a t-test comparing treatment and control, using an $\alpha = 0.05$ significance level. How many false positives (rejected hypotheses among the null IDs) do you find? What is the FDR in this instance (the fraction $\frac{V}{R}$)?

   b. Apply a Bonferroni correction to all p-values. How many false positives do you find? What is the false discovery rate?

   c. Apply the Benjamini-Hochberg procedure. What is the false discovery rate?

In [100]:
```
import pandas as pd
import numpy as np
from scipy.stats import ttest_ind

df = pd.read_csv("https://gist.githubusercontent.com/krisrs1128/ff7b6498c89316b9dd526a0f
44d92d31/raw/2683592cd4ed08d00e17d12adbc90a6bafac434c/experiments.csv")

df_control = df[df["type"]=="control"]
df_treatment = df[df["type"]=="treatment"]

store_p_value = []
for i in range(1, 10000+1):
    array_mean_control = np.asarray(df_control[df_control["experiment_id"]== i]["value"
])
    array_mean_treatment = np.asarray(df_treatment[df_treatment["experiment_id"]== i]["v
alue"])
    p_value = ttest_ind(array_mean_control, array_mean_treatment)[1]
    store_p_value.append(p_value)
```

In [109]:
```
#supposed null IDs are store_p_value[:9001]
false_positive = 0
for i in store_p_value[:9001]:
    if i < 0.05:
        false_positive +=1

print("a)")
print("There are",false_positive,"false positives, denoted as V")

number_rejected = 0
for i in store_p_value:
    if i < 0.05:
        number_rejected +=1

print("There are",number_rejected,"rejected hypothesis")
print("The false discovery rate is",false_positive/number_rejected)
```

```
a)
There are 411 false positives, denoted as V
There are 703 rejected hypothesis
The false discovery rate is 0.5846372688477952
```

In [110]:
```
alpha_corrected = 0.05/len(store_p_value)
false_positive_bonf = 0
for i in store_p_value[:9001]:
    if i < alpha_corrected:
        false_positive_bonf +=1

print("b)")
print("There are now",false_positive_bonf,"false positive with the bonferroni correctio
n, the false discovery rate is 0")
```

```
b)
There are now 0 false positive with the bonferroni correction, the false discovery rate
is 0
```

In [119]:
```
asc_p_value = sorted(store_p_value)

new_alpha = 0
for idx, value in enumerate(asc_p_value):
    if value < (0.05*idx+1) / len(store_p_value):
        new_alpha = value

f_positive = 0
for i in store_p_value[:9001]:
    if i < new_alpha:
        f_positive +=1

print("c)")
truc= 0
for i in store_p_value:
    if i < new_alpha:
        truc +=1


print("There are now",f_positive,"false positive with the Benjamini-Hochberg procedure,
 the FDR is",f_positive)
```

```
c)
There are now 0 false positive with the Benjamini-Hochberg procedure, the FDR is 0
```

Unsupervised Learning

IFT6758 Fall 2019

Clustering

1. [ISLR 10.7.4] Suppose that for a particular data set, we perform hierarchical clustering using single inkage and using complete linkage. We obtain two dendrograms.

   a. At a certain point on the single linkage dendrogram, the clusters $\{1, 2, 3\}$ and $\{4, 5\}$ fuse. On the complete linkage dendrogram, the clusters $\{1, 2, 3\}$ and $\{4, 5\}$ also fuse at a certain point. Which fusion will occur higher on the tree, or will they fuse at the same height, or is there not enough information to tell?

   b. At a certain point on the single linkage dendrogram, the clusters $\{5\}$ and $\{6\}$ fuse. On the complete linkage dendrogram, the clusters $\{5\}$ and $\{6\}$ also fuse at a certain point. Which fusion will occur higher on the tree, or will they fuse at the same height, or is there not enough information to tell?

a)

We do not have enough information to tell, because the maximal intercluster dissimilarity (complete linkage) could be equal to the minimal interculster dissimilarity (single linkage). In the case where they would not be equal, the single linkage would occur at a lower point. While if they were equal, they would occur at the same height.

b)

They would fuse at the same height, because clusters 5 and 6 are leaves and linkage only affects groups of information

Dimensionality Reduction

1. Consider the numbers $\lambda_1 = \|z_1\|_2, \ldots, \lambda_p = \|z_p\|_2$, giving the proportion of variance explained, as in equation ISLR (10.7). Define the statistics $\sum_{k=1}^{p} \left(\lambda_k - \bar{\lambda}\right)^2$, where $\bar{\lambda}$ is the average of all the $\lambda_1, \ldots, \lambda_p$. Discuss the relative usefulness of dimensionality reduction when this statistic is large vs. small.

The statistic can be defined as the sum of the squared difference of the amount of variance each principal component is explaining.

When the statistic is large, it means that there is a lot of difference between the variance each principal component is explaining. Thus, there exists projections that would reduce the number of dimensions without losing too much information from our initial input space. In this case, PCA would be a good option for dimensionality reduction, because we would be able to find a low-dimensional representation of a data set while retaining a good amount of variation. When the statistic is small PCA wouldn't be a good choice, because we would lose too much information by reducing our initial input space.