Xuecheng_Zhang_HW1

January 27, 2020

1 Homework 1

1.1 Name:Xuecheng Zhang ID:915942842

1.1.1 Group Study: Zhongrui Wang, Yiqi Ren, Xiaopeng Lan, Yuhong Fang

1.2 Question 1

Training set is the one on which we train and fit our model basically to fit the parameters whereas test data is used only to assess performance of model. Training data's output is available to model whereas testing data is the unseen data for which predictions have to be made.

1.2.1 part a)

In general, the cubic regression is more fixible to fit the datset, so the RSS of cubic regression will be smaller than linear regression. In this case, the true relationship is linear regression, the cubic regression is same as linear regression, because the β_2 and β_3 are equal to 0, so they are same for the training data.

1.2.2 part b)

For the test data, I will choose linear regression,because the true relationship between X and Y is linear, linear regression will have a smaller RSS, it correctly assumes the true data generating process,and it will tend to better fit additional data. Cubic Regression may overfit for the test data.

1.2.3 part c)

The true relationship between X and Y is not linear, but we don't know how far it is from linear. I will prefer the cubic regression. Because the cubic regression more fiexbility, so it has lower RSS.

1.2.4 part d)

First it difficult to tell, because we do not know engough infoemation. We don't know which one will lead to a lower RSS in the test data, so we don't know the true relarionship bewteen X and Y.

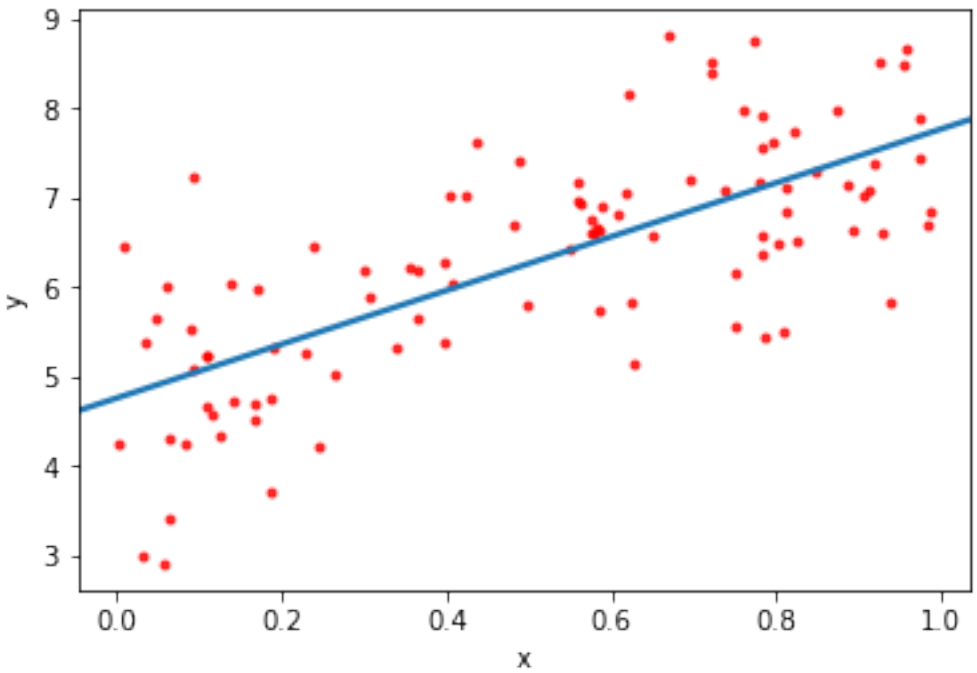
1.3 Question 2

[3]:
```
# %load
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import scale
import sklearn.linear_model as skl_lm
import statistics
```

1.3.1 part a)

[5]:
```
np.random.RandomState(1) # Set the random seed
x_1 = np.random.uniform(0, 1, size = (100, )) # X  from the uniform distribution
e_1 = np.random.normal(0,1,size=(100,)) # the residual from the given Gaussin␣
 ↪distribution
y_1 = 5 + 3*x_1 + e_1 #  the model from given relation
data= pd.DataFrame({'x':x_1,'y':y_1})# set the x_1 and y_1 in
sns.regplot(data.x,
            data.y, order=1, ci=None,
            scatter_kws={'color':'r', 's':9})# code from leture notes
```

[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a1c661c50>
```



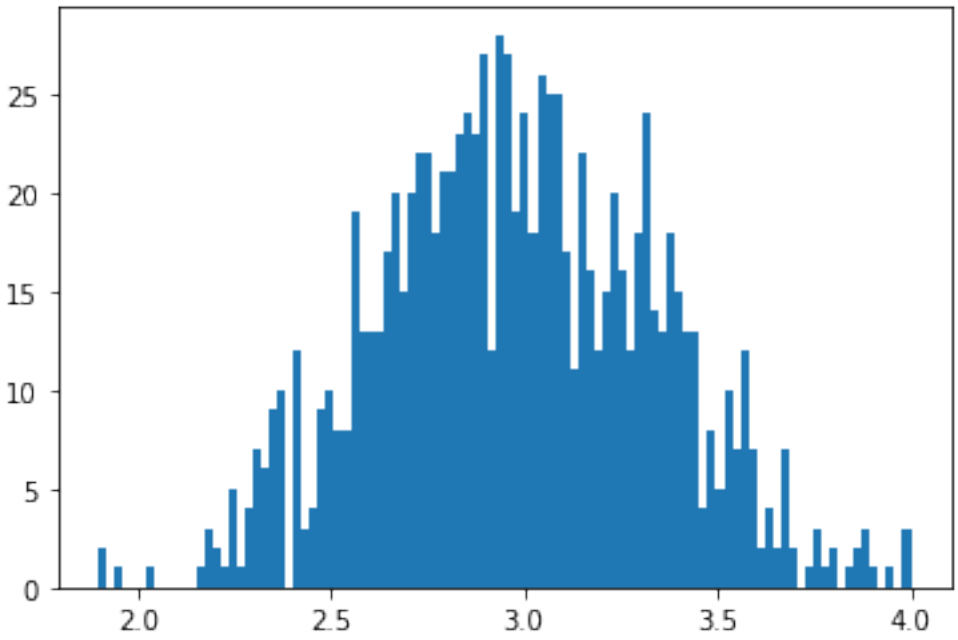
1.3.2 part b)

[6]:
```
places=[]#define an emoty list
for i in range(1,1000):
    x_2 = np.random.uniform(0, 1, size = (100, )) # X  from the uniform␣
 ↪distribution
    e_2 = np.random.normal(0,1,size=(100,)) # the residual from the given␣
 ↪Gaussin distribution
    y_2 = 5 + 3*x_2 + e_2
    data2= pd.DataFrame({'x':x_2,'y':y_2})
    regr = skl_lm.LinearRegression()
    X = scale(data2.x, with_mean=True, with_std=False).reshape(-1,1)
    y = data2.y
    regr.fit(X,y)
    Beta= regr.coef_[0]#i only need the first index verible
    places.append(Beta)
data_beta=pd.DataFrame({'beta':places})#set a datafram of list named "places"
import statistics
betamean=statistics.mean(data_beta.beta)#find the mean of beta
print('The mean of beta is:',betamean)
plt.hist(data_beta.beta, bins = 100)# I set bins is 100
```

```
The mean of beta is: 2.9790471545622217
```

[6]:
```
(array([ 2.,  0.,  1.,  0.,  0.,  0.,  1.,  0.,  0.,  0.,  0.,  0.,  1.,
         3.,  2.,  1.,  5.,  1.,  4.,  7.,  6.,  9., 10.,  0., 12.,  3.,
         4.,  9., 10.,  8.,  8., 19., 13., 13., 13., 17., 20., 15., 20.,
        22., 22., 18., 21., 21., 23., 24., 23., 27., 12., 28., 27., 19.,
        24., 18., 26., 25., 25., 17., 11., 22., 16., 12., 15., 20., 16.,
        12., 18., 24., 14., 13., 18., 15., 13., 13.,  4.,  8.,  5., 10.,
         7., 12.,  7.,  2.,  4.,  2.,  7.,  2.,  0.,  1.,  3.,  1.,  2.,
         0.,  1.,  2.,  3.,  1.,  0.,  1.,  0.,  3.]),
 array([1.8998293 , 1.92082699, 1.94182468, 1.96282237, 1.98382005,
        2.00481774, 2.02581543, 2.04681312, 2.06781081, 2.08880849,
        2.10980618, 2.13080387, 2.15180156, 2.17279924, 2.19379693,
        2.21479462, 2.23579231, 2.25678999, 2.27778768, 2.29878537,
        2.31978306, 2.34078075, 2.36177843, 2.38277612, 2.40377381,
        2.4247715 , 2.44576918, 2.46676687, 2.48776456, 2.50876225,
        2.52975994, 2.55075762, 2.57175531, 2.592753  , 2.61375069,
        2.63474837, 2.65574606, 2.67674375, 2.69774144, 2.71873912,
        2.73973681, 2.7607345 , 2.78173219, 2.80272988, 2.82372756,
        2.84472525, 2.86572294, 2.88672063, 2.90771831, 2.928716  ,
        2.94971369, 2.97071138, 2.99170907, 3.01270675, 3.03370444,
        3.05470213, 3.07569982, 3.0966975 , 3.11769519, 3.13869288,
        3.15969057, 3.18068825, 3.20168594, 3.22268363, 3.24368132,
        3.26467901, 3.28567669, 3.30667438, 3.32767207, 3.34866976,
        3.36966744, 3.39066513, 3.41166282, 3.43266051, 3.4536582 ,
```

```
        3.47465588, 3.49565357, 3.51665126, 3.53764895, 3.55864663,
        3.57964432, 3.60064201, 3.6216397 , 3.64263738, 3.66363507,
        3.68463276, 3.70563045, 3.72662814, 3.74762582, 3.76862351,
        3.7896212 , 3.81061889, 3.83161657, 3.85261426, 3.87361195,
        3.89460964, 3.91560733, 3.93660501, 3.9576027 , 3.97860039,
        3.99959808]),
 <a list of 100 Patch objects>)
```



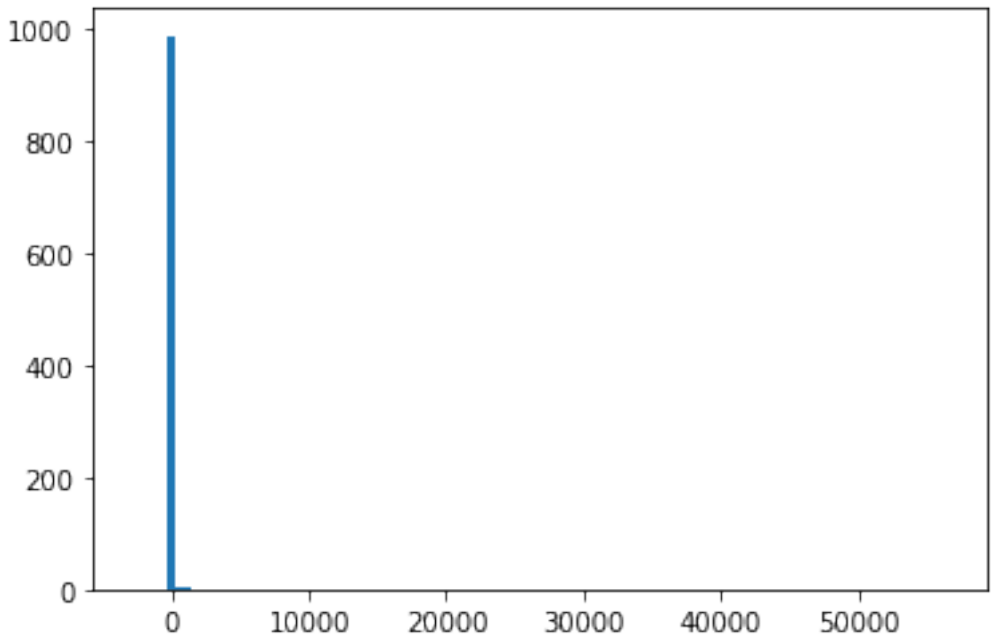
1.3.3 part c)

[7]:
```
places_2=[]#define an emoty list
for i in range(1,1000):
    x_3 = np.random.uniform(0, 1, size = (100, )) # X  from the uniform␣
 ↪distribution
    e_3 = np.random.standard_cauchy(100) # the residual from the given Gaussin␣
 ↪distribution
    y_3 = 5 + 3*x_3 + e_3
    data3= pd.DataFrame({'x':x_3,'y':y_3})
    regr = skl_lm.LinearRegression()
    X = scale(data3.x, with_mean=True, with_std=False).reshape(-1,1)
    y = data3.y
    regr.fit(X,y)
    Beta= regr.coef_[0]
    places_2.append(Beta)
data_beta3=pd.DataFrame({'beta':places_2})
```

```
betamean2=statistics.mean(data_beta3.beta)#find the mean of beta
print('The mean of beta is:',betamean2)
#
plt.hist(data_beta3.beta, bins = 100)# I set bins is 100, since I saw it around␣
 ↪0, and in part b)
#the range is around 2~4.5, so I set the rage is same as part b.
```

```
The mean of beta is: 68.69789445605205
```

[7]:
```
(array([  1.,   0.,   0.,   0., 989.,   3.,   2.,   1.,   1.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   1.,   0.,   0.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
          0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
          1.]),
 array([-2771.97758772, -2179.20081475, -1586.42404177,  -993.6472688 ,
         -400.87049583,   191.90627715,   784.68305012,  1377.4598231 ,
         1970.23659607,  2563.01336905,  3155.79014202,  3748.56691499,
         4341.34368797,  4934.12046094,  5526.89723392,  6119.67400689,
         6712.45077986,  7305.22755284,  7898.00432581,  8490.78109879,
         9083.55787176,  9676.33464474, 10269.11141771, 10861.88819068,
        11454.66496366, 12047.44173663, 12640.21850961, 13232.99528258,
        13825.77205555, 14418.54882853, 15011.3256015 , 15604.10237448,
        16196.87914745, 16789.65592043, 17382.4326934 , 17975.20946637,
        18567.98623935, 19160.76301232, 19753.5397853 , 20346.31655827,
        20939.09333125, 21531.87010422, 22124.64687719, 22717.42365017,
        23310.20042314, 23902.97719612, 24495.75396909, 25088.53074206,
        25681.30751504, 26274.08428801, 26866.86106099, 27459.63783396,
        28052.41460694, 28645.19137991, 29237.96815288, 29830.74492586,
        30423.52169883, 31016.29847181, 31609.07524478, 32201.85201775,
        32794.62879073, 33387.4055637 , 33980.18233668, 34572.95910965,
        35165.73588263, 35758.5126556 , 36351.28942857, 36944.06620155,
        37536.84297452, 38129.6197475 , 38722.39652047, 39315.17329344,
        39907.95006642, 40500.72683939, 41093.50361237, 41686.28038534,
        42279.05715832, 42871.83393129, 43464.61070426, 44057.38747724,
        44650.16425021, 45242.94102319, 45835.71779616, 46428.49456914,
        47021.27134211, 47614.04811508, 48206.82488806, 48799.60166103,
        49392.37843401, 49985.15520698, 50577.93197995, 51170.70875293,
        51763.4855259 , 52356.26229888, 52949.03907185, 53541.81584483,
        54134.5926178 , 54727.36939077, 55320.14616375, 55912.92293672,
        56505.6997097 ]),
 <a list of 100 Patch objects>)
```



From part b) there are less outlier than part c), so beta is around 2~4, the range of part c) is wider than part b) beacuse the ϵ_i is flowing standard Cauchy Distribution.

1.4 Question 3

1.4.1 part a)

As we know the probability of logistic regression is

$$p(Y = 1|X = x) = \frac{1}{1 + e^{-(x^T \times \beta)}}$$

And we know $\hat{\beta}_0$=-6, $\hat{\beta}_1$=0.05, and $\hat{\beta}_2$=1, we put in the funtion, then we got

$$p(Y = 1|X = x) = \frac{1}{1 + e^{-0.05x_1 - x_2 + 6}}$$

Next $x_1 = 40$ hours studies, and GPA $x_2 = 3.5$, we get

$$p(Y = 1|X = x) = \frac{1}{1 + e^{-0.05 \times 40 - 3.5 + 6}} \approx 0.37754$$

So the probability that a student who studies for 40 h and has an undergrad GPA of 3.5 gets an A in the class is 0.37754.

1.4.2 part b)

First we know the student in part (a) have a 50 % chance of getting an A in the class, so the the probability of logistic regression is

$$p(Y = 1|X = x) = \frac{1}{1 + e^{-0.05x_1 - 3.5 + 6}} = 0.5$$

then we got $ e^{0.05x}=12.1825$, and

$$x \approx 50$$

So 50h would the student in part (a) need to study to have a 50 % chance of getting an A in the class.

1.5 Question 4

In this problem we know that the function of y is

$$f(x) = \begin{cases} 1 & p = 1/2 \\ 0 & p = 1/2 \end{cases}$$

From $X|Y = 0 : P_{(X=1|Y=0)} = \frac{1}{3} \,\&\, P_{(X=2|Y=0)} = \frac{2}{3}$, we get $P_{(X=3|Y=0)} = 0$, also we get $P_{(X=1|Y=1)} = 0$ from $X|Y = 1 : P_{(X=2|Y=1)} = \frac{1}{3} \,\&\, P_{(X=3|Y=1)} = \frac{2}{3}$, from those, we get

$$P_{(X=1)} = \frac{1}{6}, P_{(X=2)} = \frac{1}{2}, P_{(X=3)} = \frac{1}{3}$$

Next, we need to calculate η, for

$$\eta(x = 1) = P_{(Y=1|X=1)} = \frac{P_{(X=1|Y=1)} \times P_{(Y=1)}}{P_{(X=1)}} = 0$$

$$\eta(x = 2) = P_{(Y=1|X=2)} = \frac{P_{(X=2|Y=1)} \times P_{(Y=1)}}{P_{(X=2)}} = \frac{1}{3}$$

$$\eta(x = 3) = P_{(Y=1|X=3)} = \frac{P_{(X=3|Y=1)} \times P_{(Y=1)}}{P_{(X=3)}} = 1$$

since we know the bayes classifier is

$$f^*(x) = \begin{cases} 1 & x = 3 \\ 0 & OW \end{cases}$$

So when x=3,$f^* = 1$, when x=1 and 2,$f^* = 0$