HomeWork

Jiang Yufeng 21009200038

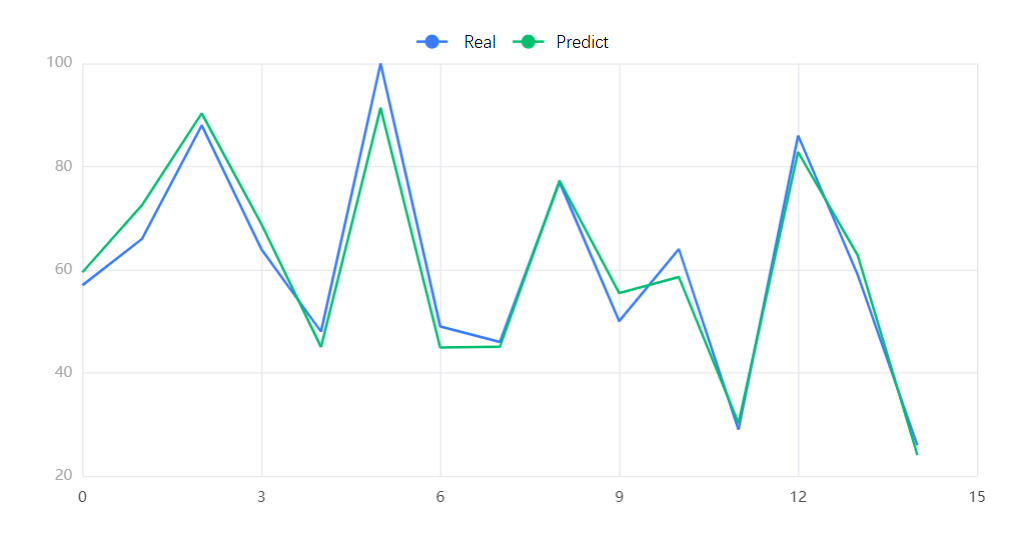
## P-1

#### Code

|  |
| --- |
| **import** numpy  **import** pandas  **from** algorithm **import** statistical\_model\_analysis  #Generate case data  data\_x1 **=** pandas**.**DataFrame**({**  "A"**:** numpy**.**random**.**random**(**size**=100),**  "B"**:** numpy**.**random**.**random**(**size**=100)**  **})**  data\_x2 **=** pandas**.**DataFrame**({**"C"**:** numpy**.**random**.**choice**([**"1"**,** "2"**,** "3"**],** size**=100)})**  data\_y **=** pandas**.**Series**(**data**=**numpy**.**random**.**choice**([1, 2],** size**=100),** name**=**"Y"**)**  result **=** statistical\_model\_analysis**.**linear\_regression**(**data\_y**=**data\_y**,** data\_x1**=**data\_x1**,** data\_x2**=**data\_x2**)**  **print(**result**)** |

#### Prediction results

**Marks=1.711\*Lec+2.638\*HW+15.396**



#### Predict

Predict a student’s mark who will be spending 19 hrs in lecture, according to the results, his marks estimated depends on his hours spent in homework, and the expression is

**Marks=2.638\*HW+47.865**

## P-2

Process

First, considering that one d value corresponds to three different t2 values, I choose to eliminate the values that are apparently unreasonable and then calculate the average value of each t2 value. Then I use the method similar to that in P-1 to calculate the result.

Here is the code.

|  |
| --- |
| **import** numpy  **import** pandas  **from** algorithm **import** statistical\_model\_analysis  #Genereate case data  data\_x1 **=** pandas**.**DataFrame**({**  "A"**:** numpy**.**random**.**random**(**size**=100),**  "B"**:** numpy**.**random**.**random**(**size**=100)**  **})**  data\_x2 **=** pandas**.**DataFrame**({**"C"**:** numpy**.**random**.**choice**([**"1"**,** "2"**,** "3"**],** size**=100)})**  data\_y **=** pandas**.**Series**(**data**=**numpy**.**random**.**choice**([1, 2],** size**=100),** name**=**"Y"**)**  result **=** statistical\_model\_analysis**.**linear\_regression**(**data\_y**=**data\_y**,** data\_x1**=**data\_x1**,** data\_x2**=**data\_x2**)**  **print(**result**)** |

Ultimately, the value g I estimate is 9.618m/s2, which is a bit smaller than the typical value of g the acceleration of gravity.

## P-3

### Q1

#### Code(MATLAB)

|  |
| --- |
| day**=[0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60]**  infected\_cases**=[1000, 1524, 3131, 3600, 6751, 7318, 8945, 7514, 9890, 8634, 14925, 18729, 27710]**  p**=**polyfit**(**day**,**infected\_cases**,3)**  **print(**p**)** |

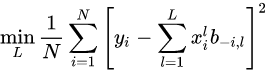
#### Results

infected\_cases=0.3730\*d3-26.6791\*d2+706.7162\*d-467.6346

### Q2

We can use a simple method called cross-validation:

Suppose you have n samples, then for an order L, you do n times of fitting, delete the ith variable each time, and then calculate the sum of squares of the residuals of the ith observation, and then find the one that makes this value the smallest, that is:



*b-i,l* is the estimated value after deleting the ith observation.

## P-4

### Code

d**=0.05** # set the value of α

x**,**y**,**z**=0.1,0.2,0.2**

cnt**=0**

while 4**\*((**x**-2)\*\*2+(**y**+3)\*\*2+(**z**-2)\*\*2)>=0.001** and cnt**<2:**

cnt**+=1**

x**-=**d**\*2\*(**x**-2)**

y**-=**d**\*2\*(**y**+3)**

z**-=**d**\*2\*(**z**-2)**

**print(**"w0="**,**x**,**"w1="**,**y**,**"w2="**,**z**)**

### Result

|  |
| --- |
| w0**= 0.29000000000000004** w1**= -0.12000000000000005** w2**= 0.38**  w0**= 0.4610000000000001** w1**= -0.40800000000000003** w2**= 0.542** |