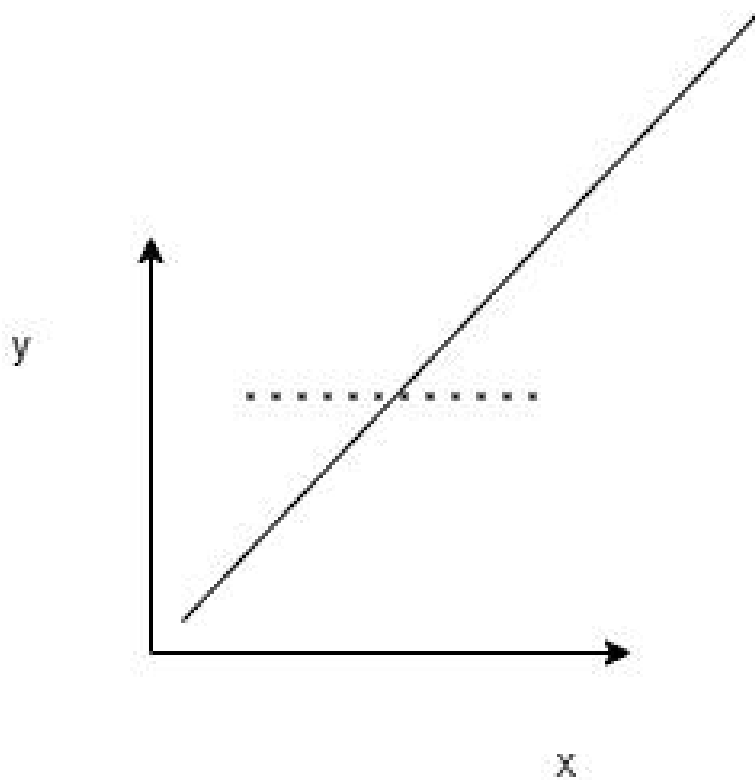
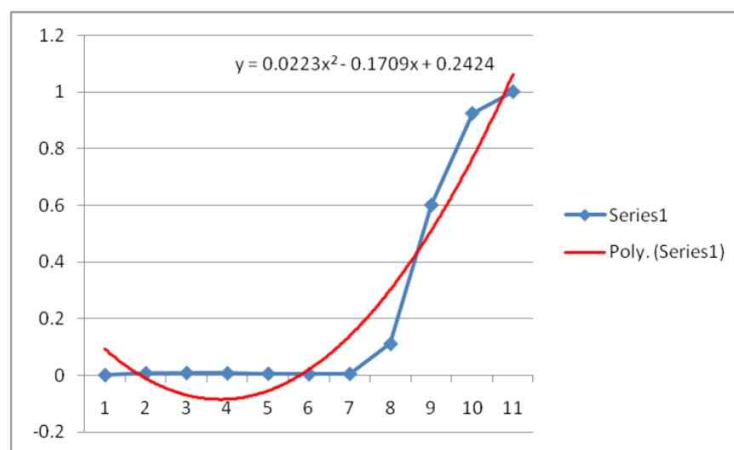


다항회귀



수많은 데이터에서 1차인  $y=ax+b$



$y=ax^2+bx+c$  2차로도 회귀할 수 있다.  
 데이터를 설명하는 것은 차수가 높을수록 좋다.  
 3차로도 추정가능하고 11차로도추정가능하지만  
 분석도안되고 과적합문제가발생한다.

회귀에대해서 알고싶은거는 현상을 예측하고 이해하고싶어서

$y=ax^2+bx+c$

5차 6차 까지 너무커지면 쓸자떼기없어진다.

적절한차수를 찾아야한다.

2차까지만해도 큰발견이다. 1차나 2차나 똑같은 원리다.

낮은차수가 더좋다 가급적 짝수나 부드러운 곡선이좋다 최대 최소로 나오면서

2차,4차 제일좋은거는 2차가좋다 최소값 나올때 생각할이이 없어지기때문이다

```
[3]: import numpy as np # 넘파이
import pandas as pd # 판다스
import matplotlib.pyplot as plt # 매트plotlib
```

```
[5]: data=pd.read_csv("test2.csv") # csv불러오기
data
```

```
[5]:
```

	height	weight
0	170	65
1	160	55
2	180	70
3	170	60
4	175	62
5	180	72
6	160	60
7	165	58
8	186	100
9	172	67

```
[7]: height=data['height'] # height 에있는 데이터만
weight=data['weight'] # weight 에 있는 데이터만
```

```
[9]: height
```

```
[9]: 0    170
1    160
2    180
3    170
4    175
5    180
6    160
7    165
8    186
9    172
Name: height, dtype: int64
```

```
[11]: weight
```

```
[11]: 0     65
1     55
2     70
3     60
4     62
5     72
6     60
7     58
8    100
9     67
Name: weight, dtype: int64
```

```
[17]: height=np.array(height) # 배열로 변경
weight=np.array(weight) # 배열로 변경
```

```
[19]: height
```

```
[19]: array([170, 160, 180, 170, 175, 180, 160, 165, 186, 172], dtype=int64)
```

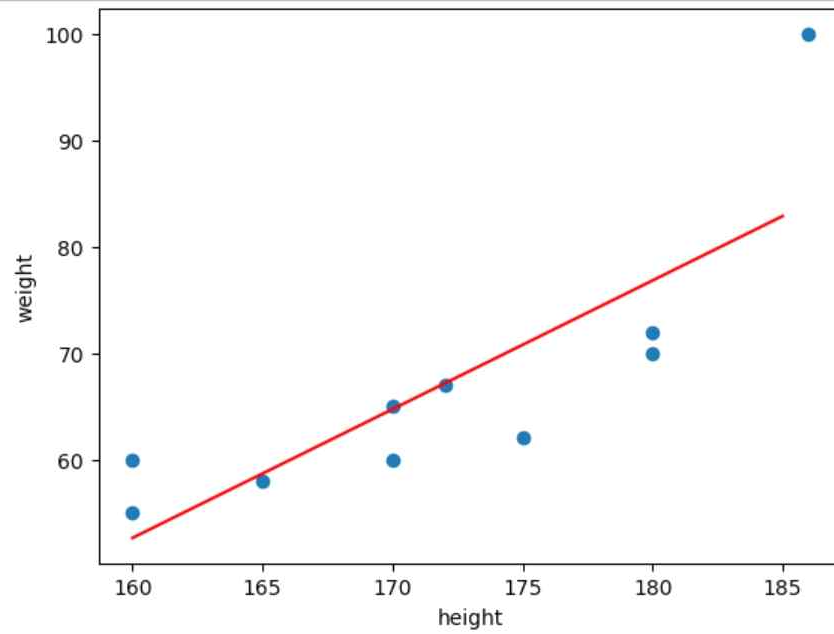
```
[21]: weight
```

```
[21]: array([ 65,  55,  70,  60,  62,  72,  60,  58, 100,  67], dtype=int64)
```

```
*[13]: reg=np.polyfit(height,weight,1) # y=ax+b 다항식을 플러노미이
reg #첫번째에 기울기 두번째에 y절편
```

```
[13]: array([ 1.20985832, -140.95365998])
```

```
[17]: reg=np.polyfit(height,weight,1)
x=np.linspace(160,185)
y=reg[0]*x+reg[1] #y=ax+b 1.20985832 가 a이고 -140.95365998 b이다.
plt.plot(x,y,"r")
plt.scatter(height,weight) # 점들
plt.xlabel("height")
plt.ylabel("weight")
plt.show()
```



< 다항회귀 1차>

```
[25]: import numpy as np # 넘파이
import pandas as pd # 판다스
import matplotlib.pyplot as plt # 매트plotlib
```

```
[27]: data=pd.read_csv("test2.csv") # csv 불러오기
data
```

```
[27]:
```

	height	weight
0	170	65
1	160	55
2	180	70
3	170	60
4	175	62
5	180	72
6	160	60
7	165	58
8	186	100
9	172	67

```
[29]: height=data['height'] # height 에 있는 데이터만
weight=data['weight'] # weight 에 있는 데이터만
```

```
[31]: height
```

```
[31]:
```

0	170
1	160
2	180
3	170
4	175
5	180
6	160
7	165
8	186
9	172

Name: height, dtype: int64

```
[33]: weight

[33]: 0    65
      1    55
      2    70
      3    60
      4    62
      5    72
      6    60
      7    58
      8   100
      9    67
      Name: weight, dtype: int64

[35]: height=np.array(height) # 배열로 변경
      weight=np.array(weight) # 배열로 변경

[37]: height

[37]: array([170, 160, 180, 170, 175, 180, 160, 165, 186, 172], dtype=int64)

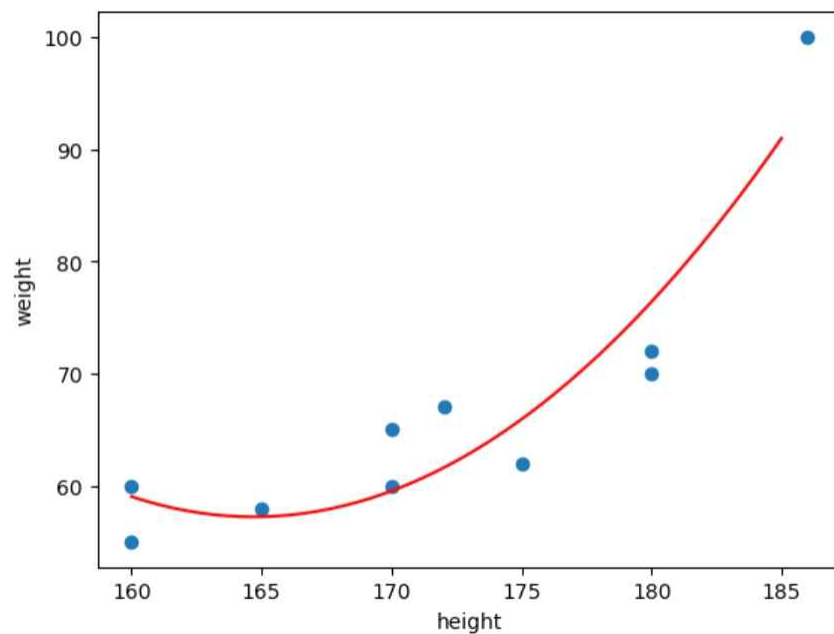
[41]: weight

[41]: array([ 65,  55,  70,  60,  62,  72,  60,  58, 100,  67], dtype=int64)

[43]: reg=np.polyfit(height,weight,2)
      reg # reg[0]*x**2+reg[1]*x+reg[2]

[43]: array([ 8.18264433e-02, -2.69523979e+01,  2.27665029e+03])

* [54]: reg=np.polyfit(height,weight,2)
      x=np.linspace(160,185)
      y=reg[0]*x**2+reg[1]*x+reg[2]
      plt.plot(x,y,"r") # y=8.18264433e-02x^2-2.69523979e+01x+2.27665029e+03 reg 0/션을 피팅한다고한다.
      plt.scatter(height,weight) # 점들
      plt.xlabel("height")
      plt.ylabel("weight")
      plt.show()
```



< 다항회귀 2차>