머신러닝과 딥러닝

Report2

소프트웨어학과

2016312568 정희윤

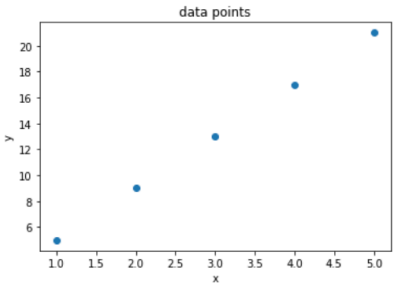
1. 선형회귀 테스트

Sample data를 y= 4x + 1 꼴로 바꾸어 실행한 결과

In [20]:
```
from sklearn.linear_model import LinearRegression
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
```

In [21]:
```
reg = LinearRegression()

#다른 샘플을 만들어 다양한 선형회귀식을 도출해볼 수 있다.
Xsample=[[1],[2],[3],[4],[5]]
Ysample=[[5],[9],[13],[17],[21]]

plt.title('data points')
plt.xlabel('x')
plt.ylabel('y')
plt.scatter(Xsample,Ysample)
```

Out[21]:
```
<matplotlib.collections.PathCollection at 0x1a20b899e8>
```



In [22]:
```
Model=reg.fit(Xsample,Ysample)
print("coef")
print(Model.coef_)
print("intercept")
print(Model.intercept_)
```

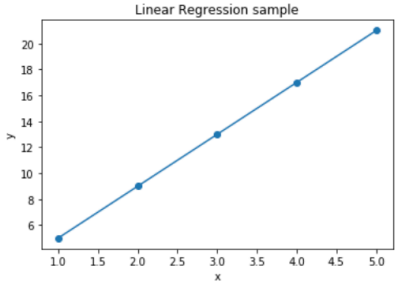
```
coef
[[4.]]
intercept
[1.]
```

In [23]:
```
Model.predict([[15]])
```

Out[23]:
```
array([[61.]])
```

In [24]:
```
plt.title('Linear Regression sample')
plt.xlabel('x')
plt.ylabel('y')
plt.scatter(Xsample,Ysample)
plt.plot(Xsample,Model.coef_*Xsample + Model.intercept_)
```

Out[24]:
```
[<matplotlib.lines.Line2D at 0x1a20c424a8>]
```



2. California Housing

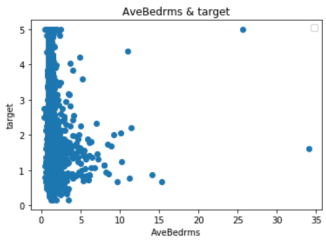
In [1]:
```
from sklearn.linear_model import LinearRegression
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
```

In [2]:
```
from sklearn.datasets import fetch_california_housing

california = fetch_california_housing()
X=california.data
DF=pd.DataFrame(X,columns=california.feature_names)
Y=california.target
print(DF)
```

```
20622    -121.44
20623    -121.37
20624    -121.41
20625    -121.52
20626    -121.43
20627    -121.32
20628    -121.48
20629    -121.39
20630    -121.32
20631    -121.40
20632    -121.45
20633    -121.53
20634    -121.56
20635    -121.09
20636    -121.21
20637    -121.22
20638    -121.32
20639    -121.24

[20640 rows x 8 columns]
```

In [3]:
```
i=3
plt.title(california.feature_names[i]+' & ' + 'target')
plt.xlabel(california.feature_names[i])
plt.ylabel('target')
plt.scatter(DF[california.feature_names[i]],Y)
plt.legend()
plt.show()
```

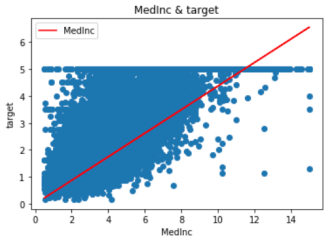
```
No handles with labels found to put in legend.
```



In [5]:
```
reg = LinearRegression()
Model=reg.fit(X,Y)
print("coef")
print(Model.coef_)
print("intercept")
print(Model.intercept_)
```

```
coef
[ 4.36693293e-01  9.43577803e-03 -1.07322041e-01  6.45065694e-01
 -3.97638942e-06 -3.78654265e-03 -4.21314378e-01 -4.34513755e-01]
intercept
-36.94192020718434
```

In [6]:
```
i=0
plt.title(california.feature_names[i]+' & ' + 'target')
plt.xlabel(california.feature_names[i])
plt.ylabel('target')
plt.scatter(DF[california.feature_names[i]],Y)
plt.plot(DF[california.feature_names[i]],Model.coef_[i]*DF[california.feature_names[i]],'r-')
plt.legend()
plt.show()
```



In [7]:
```
DF.mean()
```

Out[7]:
```
MedInc            3.870671
HouseAge         28.639486
AveRooms          5.429000
AveBedrms         1.096675
Population     1425.476744
AveOccup          3.070655
Latitude         35.631861
Longitude      -119.569704
dtype: float64
```

In [8]:
```
Model.predict([DF.mean()])
```

Out[8]:
```
array([2.06855817])
```