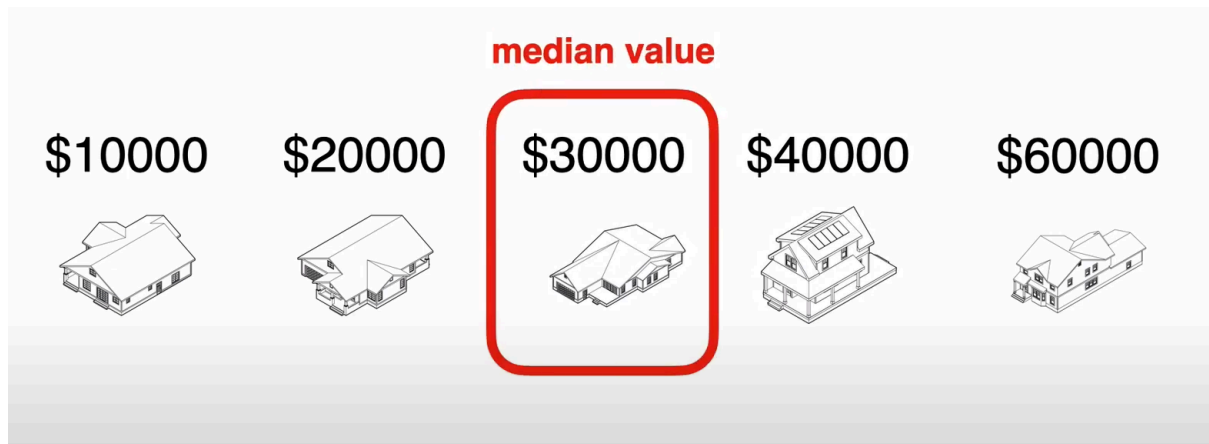


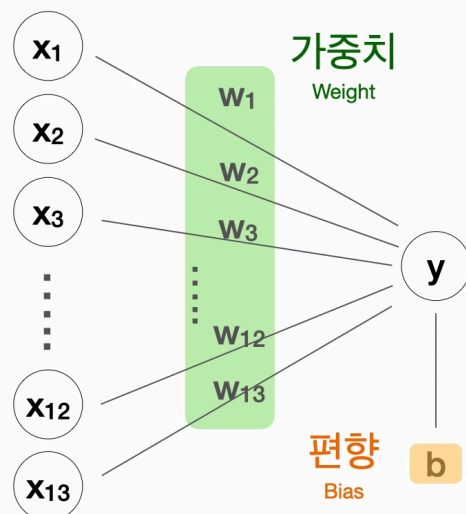
## 보스턴 집값 예측



아래 그림과 같이 독립변수가 많은 경우 텐서 플로어를 활용하면 복잡한 식을 우리가 만들지 않아도 예측할 수 있다.

$$y = w_1x_1 + w_2x_2 + \dots + w_{13}x_{13} + b$$

## 퍼셉트론 Perceptron

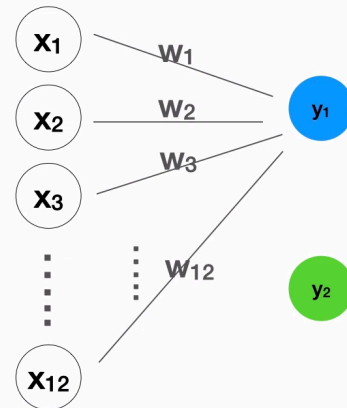


위와 같은 모형과 수식을 퍼셉트론이라는 이름이 붙어있습니다. 각  $w$ 들은 가중치라고 하고  $b$ 는 편향이라고 부릅니다. 그런데 종속변수가 두개인 아래 그림과 같이 결과가 두개가 필요합니다.

```
X = tf.keras.layers.Input(shape=[12])
Y = tf.keras.layers.Dense(2)(X)
model = tf.keras.models.Model(X, Y)
model.compile(loss='mse')
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14
CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4

$$y_1 = w_1x_1 + w_2x_2 + \dots + w_{12}x_{12} + b$$



$$y_2 = w_1x_1 + w_2x_2 + \dots + w_{12}x_{12} + b$$

수식에서  $w$  12개  $b$  1개, 두번째 수식에서  $w$  12개  $b$  1개, 총 26개의 숫자를 찾아야 합니다. 이렇게 복잡한 수식을 코드에 들어가 있으니 얼마나 편한 작업입니까.

```
1 # -*- coding: utf-8 -*-
2 import tensorflow as tf
3 import pandas as pd
4
5 파일경로 = './data/boston.csv'
6 보스턴 = pd.read_csv(파일경로)
7 print(보스턴.columns)
8 보스턴.head()
9
10 독립 = 보스턴[['crim', 'zn', 'indus', 'chas', 'nox', 'rm', 'age', 'dis', 'rad', 'tax',
11               'ptratio', 'b', 'lstat']]
12 종속 = 보스턴[['medv']]
13 print(독립.shape, 종속.shape)
14
15 X = tf.keras.layers.Input(shape=[13])
16 Y = tf.keras.layers.Dense(1)(X)
17 model = tf.keras.models.Model(X, Y)
18 model.compile(loss='mse')
19
20 model.fit(독립, 종속, epochs=1000, verbose=0)
21
22 model.fit(독립, 종속, epochs=10)
23
24 print(model.predict(독립[5:10]))
25 print(종속[5:10])
26
27 print(model.get_weights())
```