Data Science

December 25, 2018

In [1]:
```
%matplotlib inline
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn.feature_selection import SelectKBest, f_regression
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.ensemble import  RandomForestRegressor
```

In [2]:
```
dataset = datasets.load_boston()
X_full = dataset.data
Y = dataset.target
print(X_full.shape)
print(Y.shape)
```

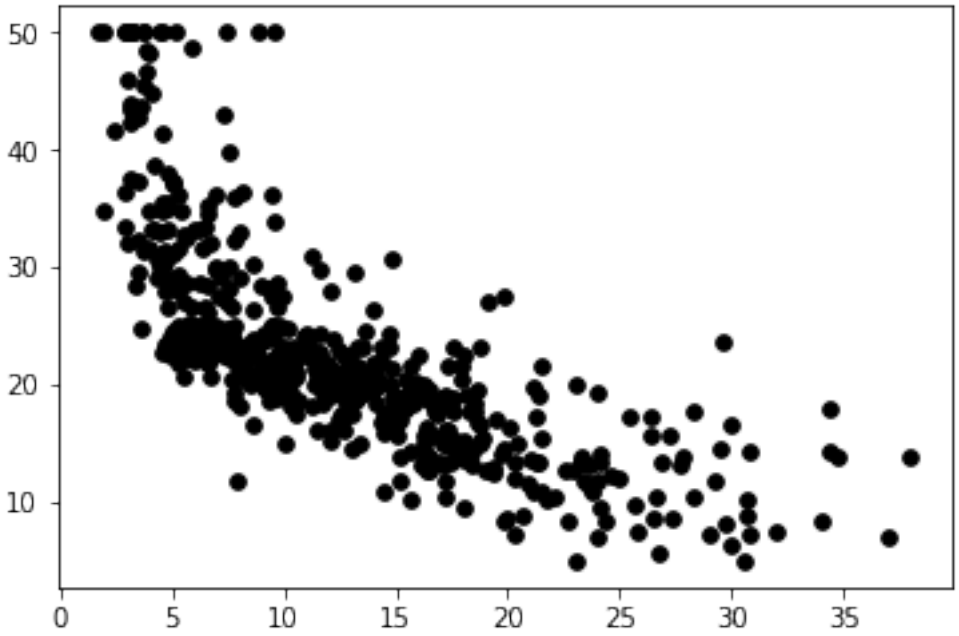
```
(506, 13)
(506,)
```

In [3]:
```
selector = SelectKBest(f_regression, k=1)
selector.fit(X_full, Y)
X = X_full[:, selector.get_support()]
print(X.shape)
```

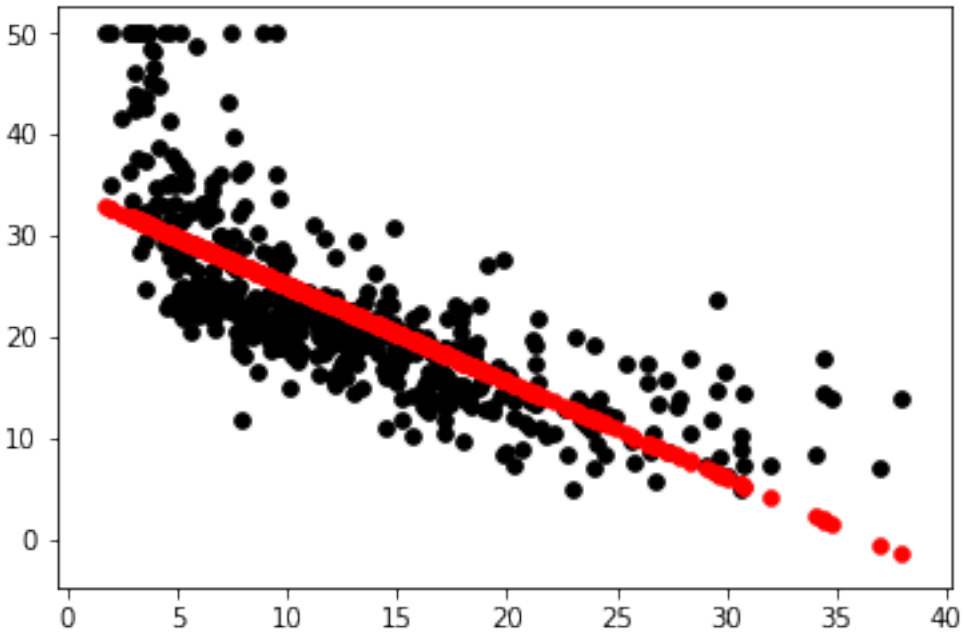
```
(506, 1)
```

In [4]:
```
def plot_scatter(X, Y, R=None):
    plt.scatter(X, Y, s=32, marker='o', facecolors='black')
    if R is not None:
        plt.scatter(X, R, color='red', linewidth=0.5)
    plt.show()
```

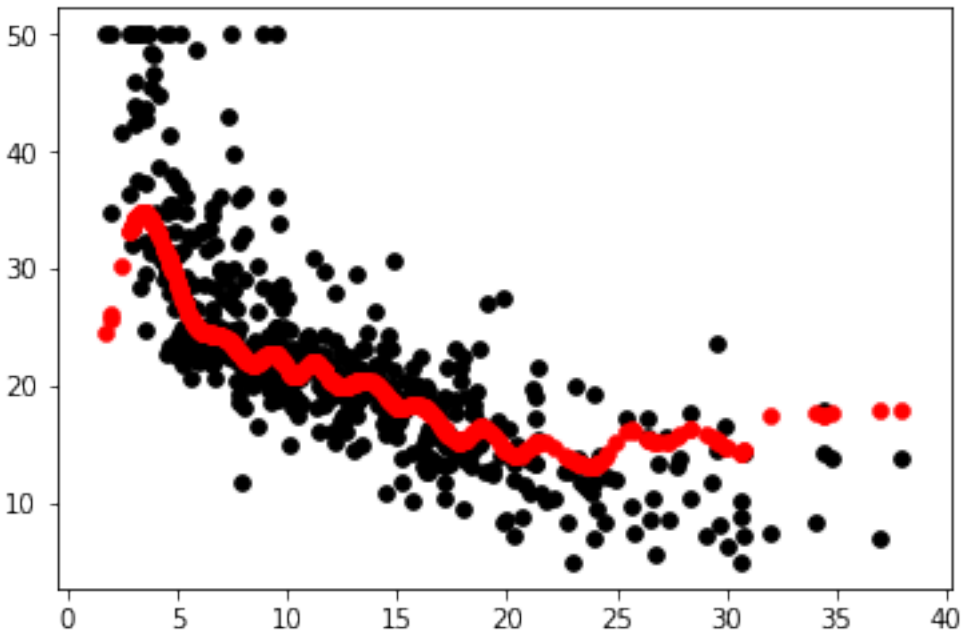
In [5]:
```
plot_scatter(X, Y)
```



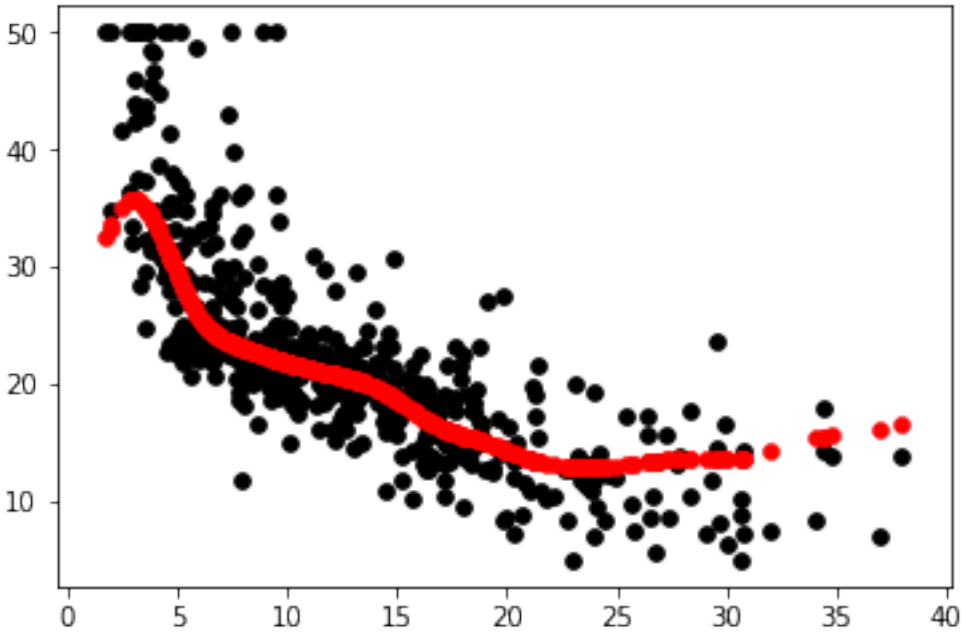
In [6]:
```
regresor = LinearRegression(normalize = True).fit(X, Y)
plot_scatter(X, Y, regresor.predict(X))
```



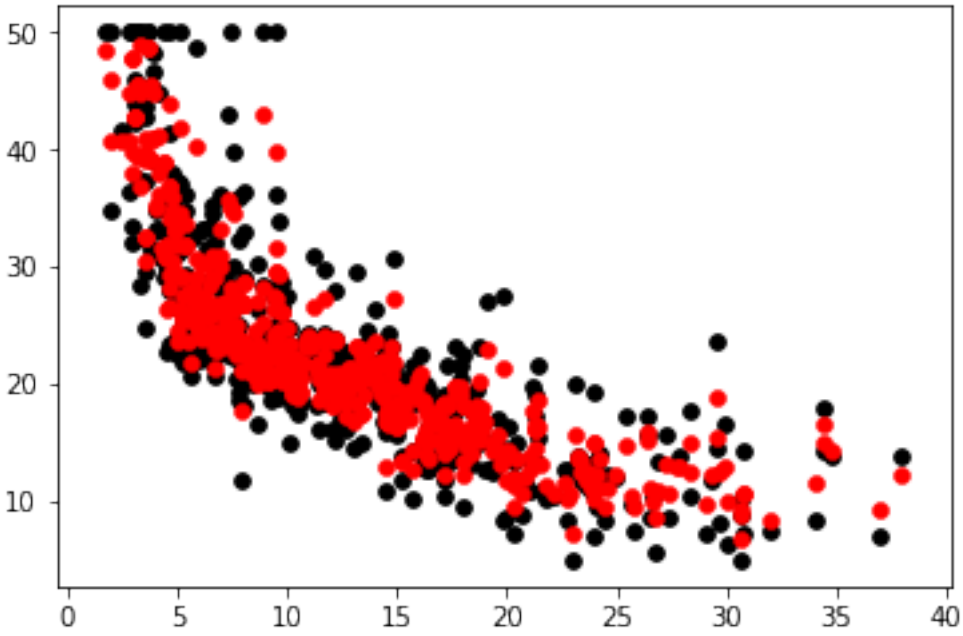
In [7]:
```
regresor = SVR(gamma ='auto').fit(X,Y)
plot_scatter(X, Y, regresor.predict(X))
```



In [8]:
```
regresor = SVR(gamma ='scale').fit(X,Y)
plot_scatter(X, Y, regresor.predict(X))
```



In [9]:
```
regresor = RandomForestRegressor(n_estimators =100).fit(X,Y)
plot_scatter(X, Y, regresor.predict(X))
```



In [ ]: