In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import transforms, pyplot as plt
%matplotlib inline

# Reference:
# https://github.com/empathy87/The-Elements-of-Statistical-Learning-Python-Notebooks/blob/master/examples/South%20African%20Heart%20Disease.ipynb

# set common plots properties in advance
plt.rcParams['axes.linewidth'] = 0.5
GRAY4, GRAY7 = '#646369', '#929497'
```

In [2]:

```
# load training and test data set from a text file
df = pd.read_csv("https://raw.githubusercontent.com/JWarmenhoven/ISLR-python/master/Notebooks/Data/Wage.csv",
                 usecols=['age', 'wage'])
df.head()
```

Out[2]:

	age	wage
0	18	75.043154
1	24	70.476020
2	45	130.982177
3	43	154.685293
4	50	75.043154

In [3]:

```
# Convert data into X and y numpy arrays
X , y = df[['age']].values , df['wage'].values
```

1. Build a degree 4 polynomial model

In [4]:

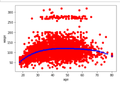
```
# Fitting Polynomial Regfression to the dataset
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
poly_reg4 = PolynomialFeatures(degree = 4)
X_poly4 = poly_reg4.fit_transform(X)
linear_reg = LinearRegression()
linear_reg.fit(X_poly4, y)
```

Out[4]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

In [5]:

```
# Visualization of Polynomial Regression results
def viz_polynomial():
    plt.scatter(X, y, color='red')
    plt.plot(X, linear_reg.predict(poly_reg4.fit_transform(X)),'b.')
    plt.xlabel('age')
    plt.ylabel('wage')
    plt.show()
    return
viz_polynomial()
```



2. 5-fold cross validation

In [6]:

```
poly_reg3 = PolynomialFeatures(degree = 3)
poly_reg5 = PolynomialFeatures(degree = 5)
X_poly3 = poly_reg3.fit_transform(X)
X_poly5 = poly_reg5.fit_transform(X)
```

In [7]:

```
from sklearn.model_selection import cross_val_score
R2_degree4 = cross_val_score(linear_reg, X_poly4, y, cv = 5)
R2_degree3 = cross_val_score(linear_reg, X_poly3, y, cv = 5)
R2_degree5 = cross_val_score(linear_reg, X_poly5, y, cv = 5)
print('The R^2 value for regression model of degree 4', np.mean(R2_degree4))
print('The R^2 value for regression model of degree 3 ',np.mean(R2_degree3))
print('The R^2 value for regression model of degree 5',np.mean(R2_degree5))
```

```
The R^2 value for regression model of degree 4 0.08355308129863867
The R^2 value for regression model of degree 3  0.08312462934678026
The R^2 value for regression model of degree 5 0.08258094601300044
```

- The regression model of degree 4 acheive greatest R^2 value on average.

3. Degree 5 polynomial model using statsmodel

In [8]:

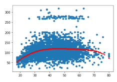
```
# Polynomial regression from statmodels
import statsmodels.api as sm
model = sm.OLS(y,X_poly5).fit()
ypred = model.predict(X_poly5)
```

In [9]:

```
plt.scatter(X,y,linewidth=0.1)
plt.plot(X,ypred,'r.', linewidth=0.1)
```

Out[9]:

```
[<matplotlib.lines.Line2D at 0x1a191dc110>]
```



In [10]:

```
model.summary()
```

Out[10]:

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.087
Model:	OLS	**Adj. R-squared:**	0.085
Method:	Least Squares	**F-statistic:**	56.71
Date:	Sat, 28 Mar 2020	**Prob (F-statistic):**	1.67e-56
Time:	00:15:15	**Log-Likelihood:**	-15314.
No. Observations:	3000	**AIC:**	3.064e+04
Df Residuals:	2994	**BIC:**	3.068e+04
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-49.7046	161.435	-0.308	0.758	-366.239	266.830
x1	3.9930	20.110	0.199	0.843	-35.438	43.424
x2	0.2760	0.958	0.288	0.773	-1.603	2.155
x3	-0.0126	0.022	-0.577	0.564	-0.056	0.030
x4	0.0002	0.000	0.762	0.446	-0.000	0.001
x5	-9.157e-07	1.02e-06	-0.897	0.370	-2.92e-06	1.09e-06

Omnibus:	1094.840	**Durbin-Watson:**	1.961
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	4940.229
Skew:	1.718	**Prob(JB):**	0.00
Kurtosis:	8.265	**Cond. No.**	9.39e+10

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 9.39e+10. This might indicate that there are
strong multicollinearity or other numerical problems.

Hypothesis test

- The p-value in the model summary related to the given hypothesis is 0.370
- For interpreation of the results, I am choosing significance level $\alpha = 5$ %. Here, we found that p-value > α, failing to reject the null hypothesis (i.e. not significant result).

In [ ]: