[9]:
```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import pylab as pl
%matplotlib inline
```

[10]:
```
df = pd.read_csv("FuelConsumption.csv")
```

[11]:
```
df.head()
```

[11]:

	MODELYEAR	MAKE	MODEL	VEHICLECLASS	ENGINESIZE	CYLINDERS	\
0	2014	ACURA	ILX	COMPACT	2.0	4	
1	2014	ACURA	ILX	COMPACT	2.4	4	
2	2014	ACURA	ILX HYBRID	COMPACT	1.5	4	
3	2014	ACURA	MDX 4WD	SUV - SMALL	3.5	6	
4	2014	ACURA	RDX AWD	SUV - SMALL	3.5	6	

	TRANSMISSION	FUELTYPE	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	\
0	AS5	Z	9.9	6.7	
1	M6	Z	11.2	7.7	
2	AV7	Z	6.0	5.8	
3	AS6	Z	12.7	9.1	
4	AS6	Z	12.1	8.7	

	FUELCONSUMPTION_COMB	FUELCONSUMPTION_COMB_MPG	CO2EMISSIONS
0	8.5	33	196
1	9.6	29	221
2	5.9	48	136
3	11.1	25	255
4	10.6	27	244

[12]:
```
cdf = ␣
 ↪df[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_CITY','FUELCONSUMPTION_HWY','FUELCONSUMPTION_
cdf.head(9)
```

[12]:

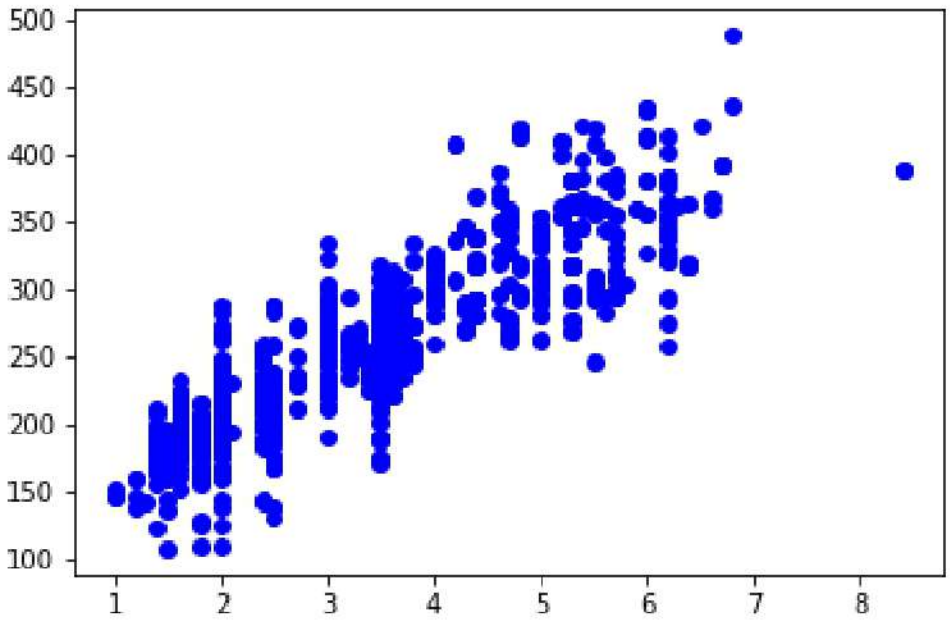
	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	\
0	2.0	4	9.9	6.7	

1	2.4	4	11.2	7.7
2	1.5	4	6.0	5.8
3	3.5	6	12.7	9.1
4	3.5	6	12.1	8.7
5	3.5	6	11.9	7.7
6	3.5	6	11.8	8.1
7	3.7	6	12.8	9.0
8	3.7	6	13.4	9.5

	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	8.5	196
1	9.6	221
2	5.9	136
3	11.1	255
4	10.6	244
5	10.0	230
6	10.1	232
7	11.1	255
8	11.6	267

[13]:
```
plt.scatter(cdf.ENGINESIZE, cdf.CO2EMISSIONS, color='blue')
```

[13]:
```
<matplotlib.collections.PathCollection at 0x7fcbe3226be0>
```



[14]:
```
msk= np.random.rand(len(cdf))<0.8
```

[15]:
```
train= cdf[msk]
```

[16]:
```
test= cdf[~msk]
```

[17]:
```
train.describe()
```

[17]:

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	\
count	887.000000	887.000000	887.000000	887.000000	
mean	3.346336	5.786922	13.260541	9.458963	
std	1.420316	1.773497	4.083516	2.782675	
min	1.000000	3.000000	4.600000	4.900000	
25%	2.000000	4.000000	10.300000	7.500000	
50%	3.400000	6.000000	12.500000	8.800000	
75%	4.300000	8.000000	15.400000	10.850000	
max	8.400000	12.000000	30.200000	20.500000	

	FUELCONSUMPTION_COMB	CO2EMISSIONS
count	887.000000	887.000000
mean	11.554115	255.775648
std	3.471230	62.886658
min	4.700000	108.000000
25%	9.050000	209.000000
50%	10.900000	250.000000
75%	13.300000	294.000000
max	25.800000	488.000000

[18]:
```
from sklearn import linear_model
```

[19]:
```
regressor = linear_model.LinearRegression()
```

[20]:
```
train_x = np.
↪asanyarray(train[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_CITY']])
```

[21]:
```
train_y = np.asanyarray(train[['CO2EMISSIONS']])
```

[22]:
```
regressor.fit(train_x,train_y)
```

[22]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
         normalize=False)
```

[ ]:
```

```