MultiLinear Regression

Cycle Power Plant Data Set

Data loaded

In [39]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df=pd.read_excel('C:/Users/vaitheeswaran/OneDrive/Documents/public_data.xlsx')
```

In [3]:
```
df.head()
```

Out[3]:

	AT	V	AP	RH	PE
0	8.34	40.77	1010.84	90.01	480.48
1	23.64	58.49	1011.40	74.20	445.75
2	29.74	56.90	1007.15	41.91	438.76
3	19.07	49.69	1007.22	76.79	453.09
4	11.80	40.66	1017.13	97.20	464.43

Preprocessing

In [26]:
```
df.shape
```

Out[26]:
```
(79, 5)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 79 entries, 0 to 78
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   AT      79 non-null     float64
 1   V       79 non-null     float64
 2   AP      79 non-null     float64
 3   RH      79 non-null     float64
 4   PE      79 non-null     float64
dtypes: float64(5)
memory usage: 3.2 KB
```

In [5]:
```
df.describe()
```

Out[5]:

	AT	V	AP	RH	PE
count	79.000000	79.000000	79.000000	79.000000	79.000000
mean	20.680000	55.525443	1012.726203	69.670633	452.217342
std	8.098244	12.986639	6.277290	16.845526	18.000144
min	5.230000	35.570000	994.170000	32.970000	426.250000
25%	14.685000	41.425000	1008.810000	58.605000	437.195000
50%	22.100000	58.490000	1012.400000	72.460000	447.420000
75%	27.150000	67.915000	1017.570000	84.020000	464.885000
max	34.200000	75.600000	1025.530000	97.460000	487.690000

In [6]:

In [6]:
```
df.isnull().sum()
```

Out[6]:
```
AT    0
V     0
AP    0
RH    0
PE    0
dtype: int64
```

In [7]:
```
df.corr()
```

Out[7]:

	AT	V	AP	RH	PE
AT	1.000000	0.846184	-0.495145	-0.635554	-0.966368
V	0.846184	1.000000	-0.391515	-0.346131	-0.885379
AP	-0.495145	-0.391515	1.000000	0.031756	0.497474
RH	-0.635554	-0.346131	0.031756	1.000000	0.527514
PE	-0.966368	-0.885379	0.497474	0.527514	1.000000

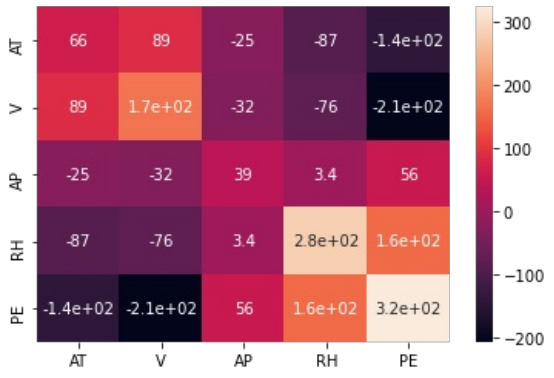
In [11]:
```
sns.heatmap(df.corr(),annot=True)
```

Out[11]:
```
<AxesSubplot:>
```

In [ ]:
```
df.cov()
```

In [12]:
```
sns.heatmap(df.cov(),annot=True)
```

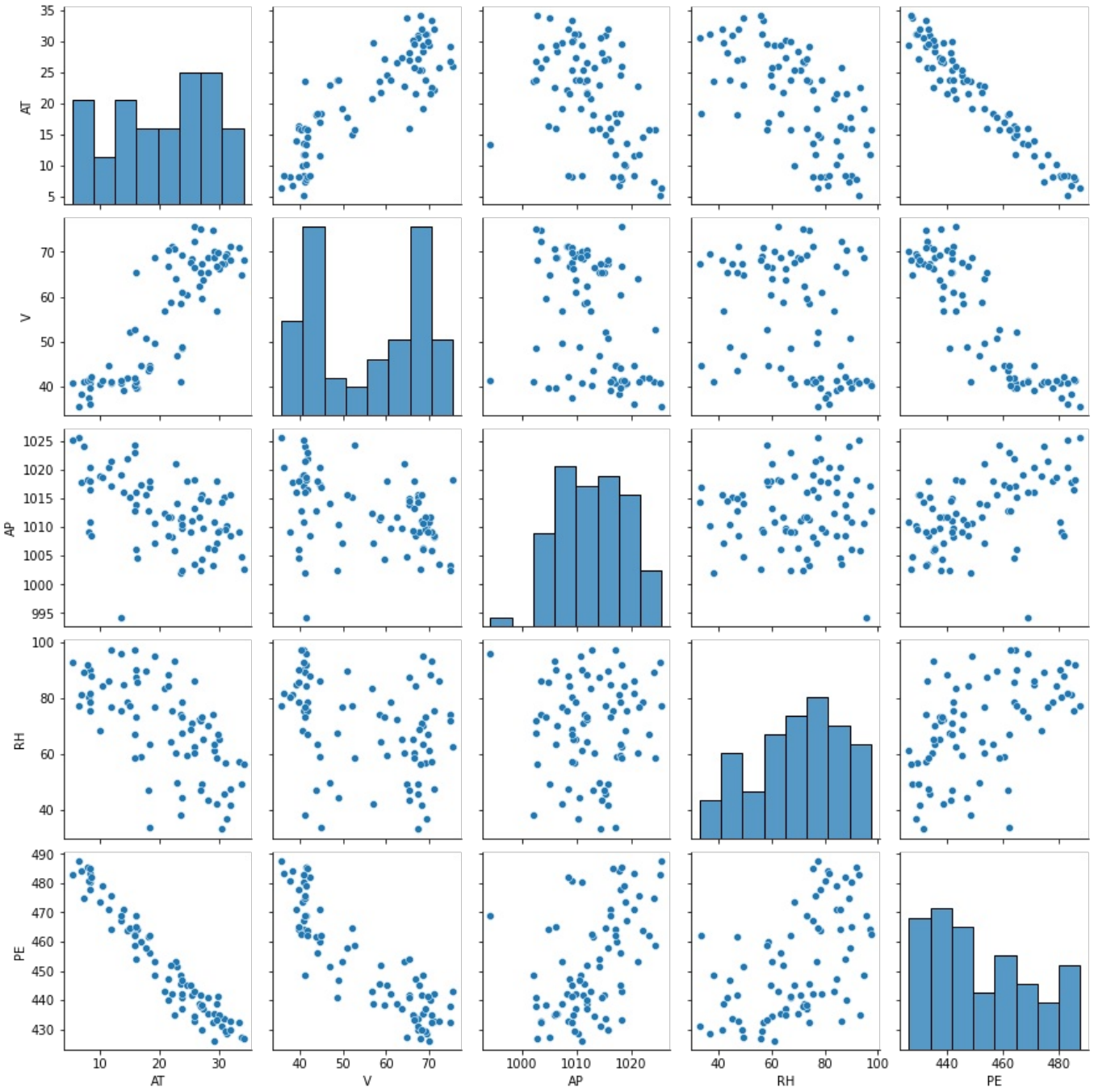
Out[12]:
```
<AxesSubplot:>
```



Exploratory Data Analysis

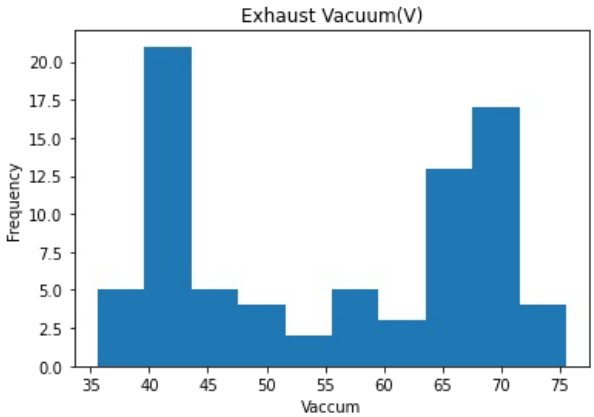
In [13]:
```
sns.pairplot(df)
```

Out[13]:
```
<seaborn.axisgrid.PairGrid at 0x20902777a30>
```



In [16]:
```
df['V'].plot.hist()
plt.xlabel('Vaccum')
plt.title('Exhaust Vacuum(V)')
```

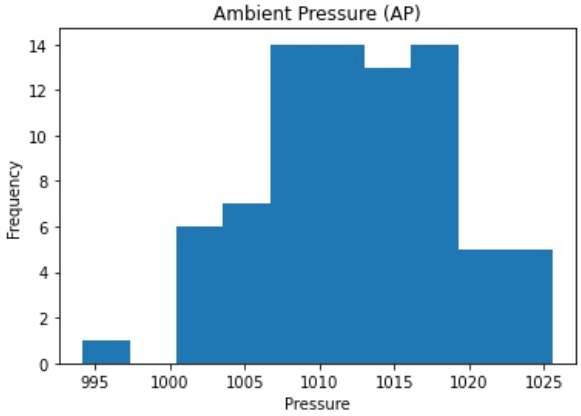
Out[16]: Text(0.5, 1.0, 'Exhaust Vacuum(V)')



In [17]:
```
df['AP'].plot.hist()
```

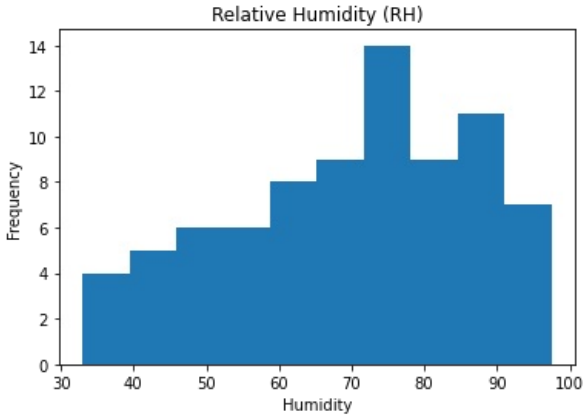
```
plt.xlabel('Pressure')
plt.title('Ambient Pressure (AP)')
```

Out[17]: Text(0.5, 1.0, 'Ambient Pressure (AP)')



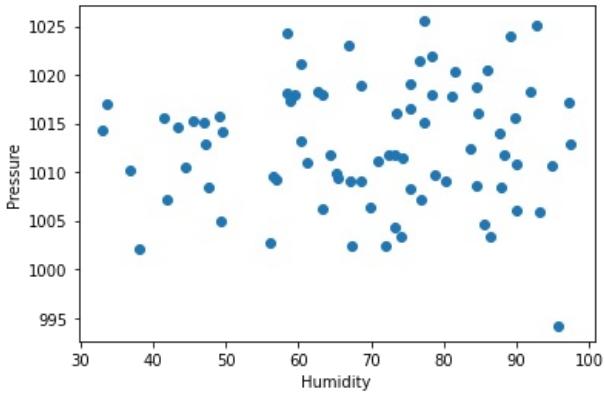
In [19]:
```
df['RH'].plot.hist()
plt.xlabel('Humidity')
plt.title('Relative Humidity (RH)')
```

Out[19]: Text(0.5, 1.0, 'Relative Humidity (RH)')



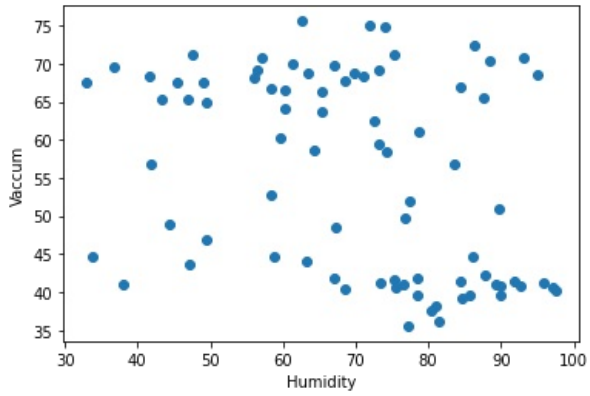
In [21]:
```
plt.scatter(df['RH'],df['AP'])
plt.xlabel('Humidity')
plt.ylabel('Pressure')
```

Out[21]: Text(0, 0.5, 'Pressure')



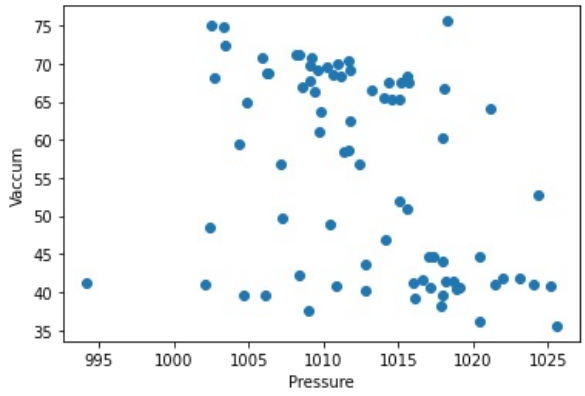
In [22]:
```
plt.scatter(df['RH'],df['V'])
plt.xlabel('Humidity')
plt.ylabel('Vaccum')
```

Out[22]: Text(0, 0.5, 'Vaccum')



In [23]:
```
plt.scatter(df['AP'],df['V'])
plt.xlabel('Pressure')
plt.ylabel('Vaccum')
```

Out[23]:
```
Text(0, 0.5, 'Vaccum')
```



Training and Testing Data

In [24]:
```
import sklearn
```

In [25]:
```
from sklearn.model_selection import train_test_split
```

In [28]:
```
x=df.drop(['PE'],axis=1).values
y=df['PE'].values
```

In [29]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.4,random_state=2)
```

In [30]:
```
from sklearn.linear_model import LinearRegression
```

In [31]:
```
LR=LinearRegression()
```

In [32]:
```
LR.fit(x_train,y_train)
```

Out[32]:
```
LinearRegression()
```

Evaluation

In [33]:
```
ypred=LR.predict(x_test)
```

In [34]:
```
from sklearn.metrics import r2_score
```

In [36]:
```
r2_score(y_test,ypred)
```

Out[36]:
```
0.9450621235840357
```

In [38]:
```
plt.scatter(y_test,ypred)
plt.xlabel('Actual')
plt.ylabel('Prediction')
plt.title('Actual and Predictions')
```

Out[38]:
```
Text(0.5, 1.0, 'Actual and Predictions')
```