In [1]:
```
###Load data from CSV file

import pandas as pd
```

In [2]:
```
CurrentPowerTime = pd.read_csv('current,power vs time.csv')
CurrentPowerTime
```

Out[2]:

	Category	Current	Power
0	18:17:11	-0.050400	0.003385
1	18:17:13	-0.050340	0.294197
2	18:17:15	-0.050338	-0.181814
3	18:17:17	-0.050326	-0.181596
4	18:17:19	-0.050310	-0.181576
...	...	...	...
861	18:45:55	-0.050308	-0.180528
862	18:45:56	-0.050312	-0.180405
863	18:45:58	-0.050312	-0.180467
864	18:45:59	-0.050285	-0.180478
865	18:46:01	-0.050284	-0.180475

866 rows × 3 columns

In [3]:
```
CurrentPowerTime.shape
```

Out[3]:
```
(866, 3)
```

In [4]:
```
CurrentPowerTime.size
```

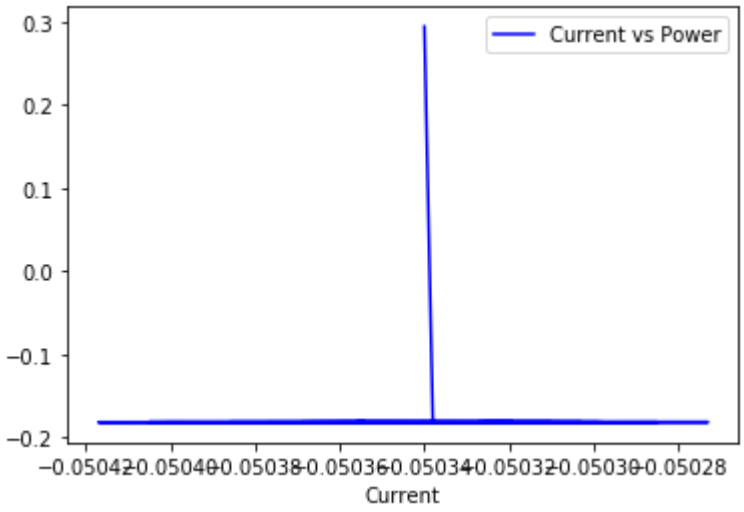
Out[4]:
```
2598
```

In [5]:
```
CurrentPowerTime.count()
```

Out[5]:
```
Category    866
Current     866
Power       866
dtype: int64
```

In [6]:
```
CPT = CurrentPowerTime[1:200]
CPT.plot(x='Current',y='Power',color='blue',label='Current vs Power')
```

Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x23303545c08>



In [7]:
```
CurrentPowerTime.dtypes
```

Out[7]:
```
Category     object
Current     float64
Power       float64
dtype: object
```

In [8]:
```
###Identifying Unwanted Rows

CurrentPowerTime.dtypes
```

Out[8]:
```
Category     object
Current     float64
Power       float64
dtype: object
```

In [9]:
```
CurrentPowerTime.columns
```

Out[9]:
```
Index(['Category', 'Current', 'Power'], dtype='object')
```

In [10]:
```
import numpy as np
CurrentPowerTime.columns
features = CurrentPowerTime[['Category', 'Current', 'Power']]
x=np.asarray(features)
y=np.array(CurrentPowerTime['Power'])
x[1:10]
```

Out[10]:
```
array([['18:17:13', -0.050339999999999996, 0.294197],
       ['18:17:15', -0.050338, -0.181814],
       ['18:17:17', -0.050325999999999996, -0.181596],
       ['18:17:19', -0.05031, -0.181576],
       ['18:17:21', -0.050337, -0.181523],
       ['18:17:23', -0.050318, -0.18146099999999998],
       ['18:17:25', -0.050362, -0.18154800000000001],
       ['18:17:27', -0.05036, -0.181475],
       ['18:17:29', -0.050328, -0.18162799999999998]], dtype=object)
```

In [11]:
```
###Divide Data into train/test data set

from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=
4)
x_train.shape
x_test.shape
```

Out[11]:
```
(174, 3)
```

In [12]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=
4)
y_train.shape
y_test.shape
```

Out[12]:
```
(174,)
```

In [13]:
```
###Modeling


from sklearn import svm
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=
4)
classifier = svm.SVC(kernel='poly',gamma='auto',C=0)
classifier
```

Out[13]:
```
SVC(C=0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='auto', kernel='poly',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

In [17]:
```
#classifier.fit(x_train,y_train)
#y_predict = classifier.predict(x_test)
```

In [ ]:
```

```

In [ ]:

In [ ]: