

## Code

[illegible]

```

#load data points
data = pd.read_csv('tips.csv')
bill = np.array(data.bill)
tip = np.array(data.tip)
mbill = np.mat(bill)
mtip = np.mat(tip)
m = np.shape(mbill)[1]
one = np.mat(np.ones(m))
X = np.hstack((one.T, mbill.T))

```

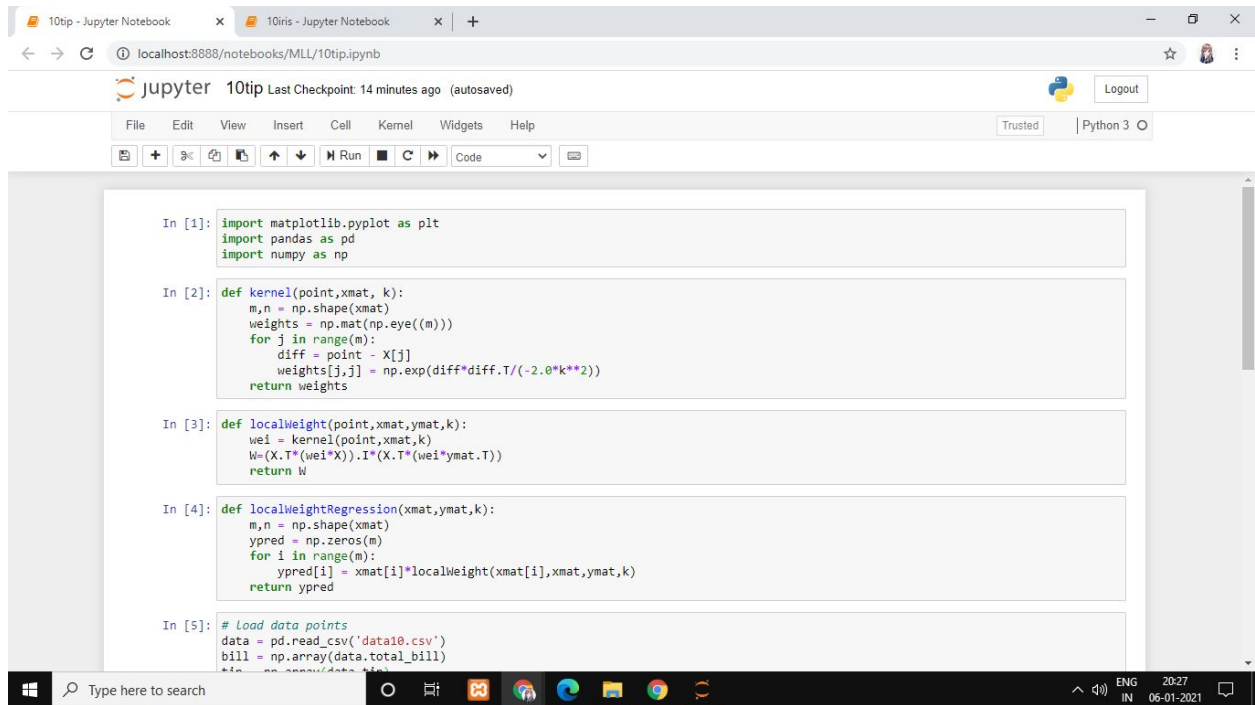
```

#set k here
ypred = local.WeightedRegression(X, mtip, 2)
sortIndex = X[:, 1].argsort(0)
xsort = X[sortIndex][:, 0]
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
ax.scatter(bill, tip, color='green')
ax.plot(xsort[:, 1], ypred[sortIndex], color='red',
        linewidth=5)
plt.xlabel('bill')
plt.ylabel('Tip')
plt.show();

```

# Output

## For BillTip Dataset



The image shows a Jupyter Notebook interface with the following code cells:

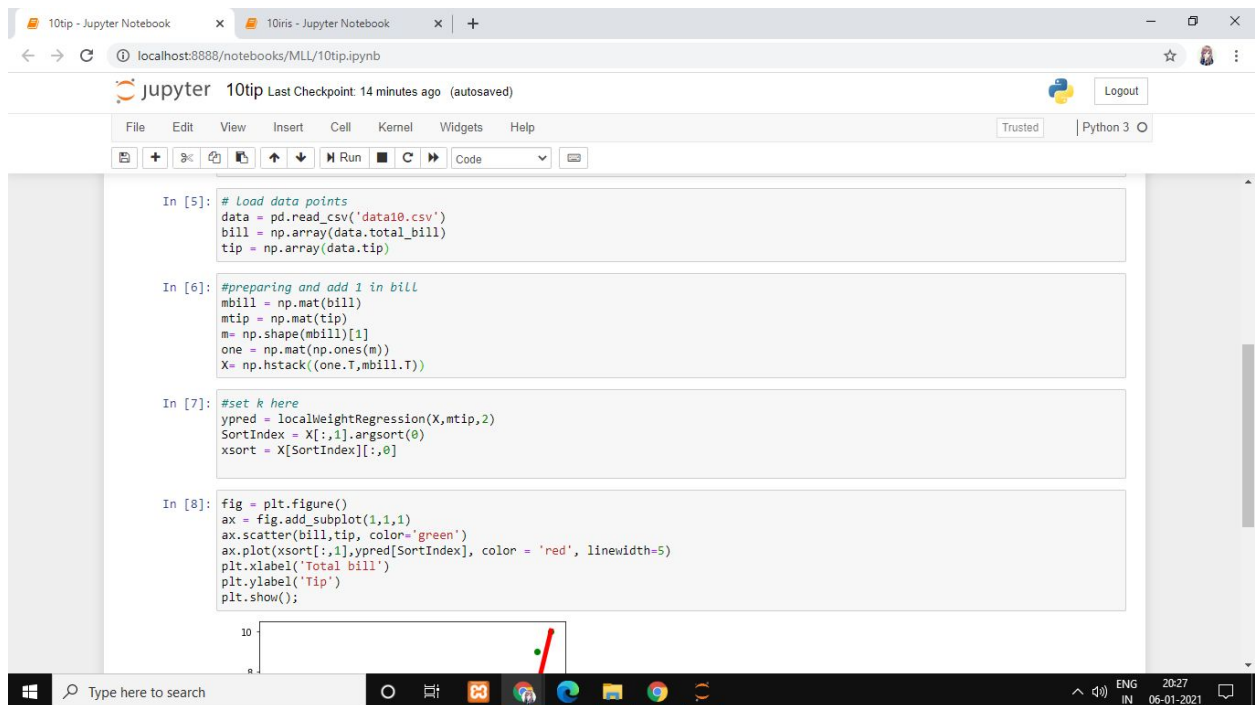
```
In [1]: import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

In [2]: def kernel(point,xmat, k):
m,n = np.shape(xmat)
weights = np.mat(np.eye((m)))
for j in range(m):
diff = point - X[j]
weights[j,j] = np.exp(diff*diff.T/(-2.0*k**2))
return weights

In [3]: def localWeight(point,xmat,yamat,k):
wei = kernel(point,xmat,k)
W=(X.T*(wei*X)).I*(X.T*(wei*yamat.T))
return W

In [4]: def localWeightRegression(xmat,yamat,k):
m,n = np.shape(xmat)
ypred = np.zeros(m)
for i in range(m):
ypred[i] = xmat[i]*localWeight(xmat[i],xmat,yamat,k)
return ypred

In [5]: # Load data points
data = pd.read_csv('data10.csv')
bill = np.array(data.total_bill)
tip = np.array(data.tip)
```



The image shows a Jupyter Notebook interface with the following code cells:

```
In [5]: # Load data points
data = pd.read_csv('data10.csv')
bill = np.array(data.total_bill)
tip = np.array(data.tip)

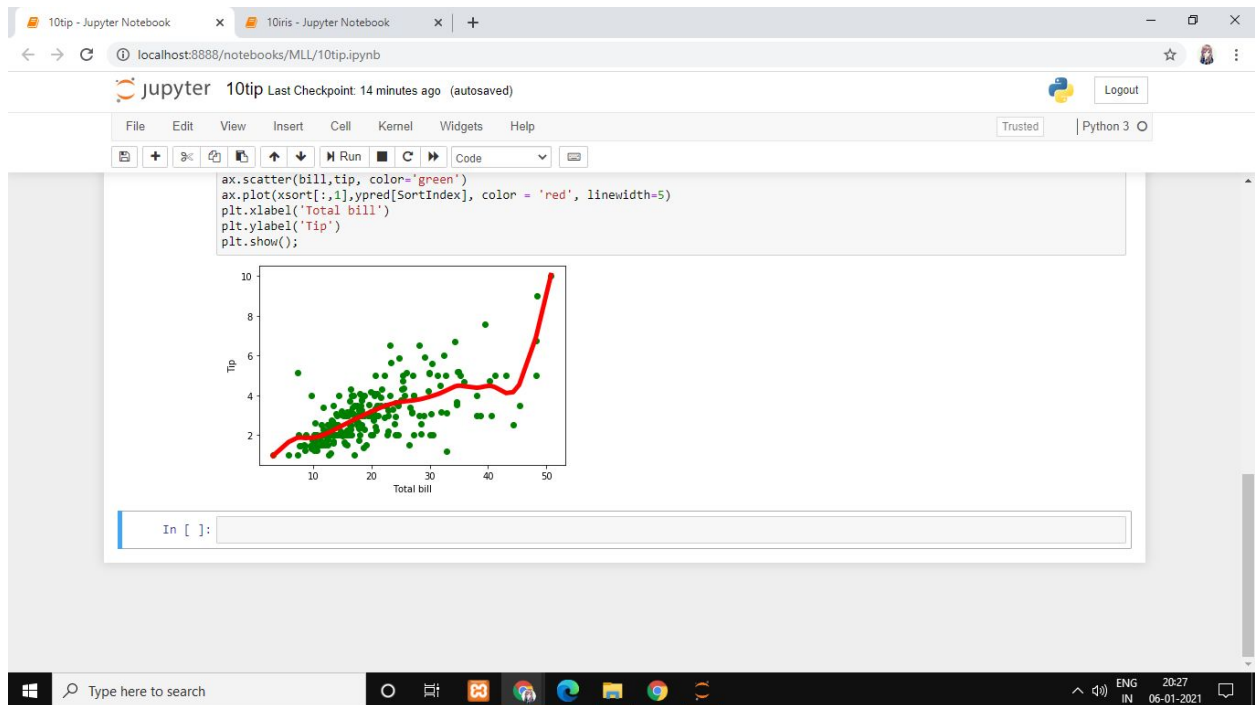
In [6]: #preparing and add 1 in bill
mbill = np.mat(bill)
mtip = np.mat(tip)
m = np.shape(mbill)[1]
one = np.mat(np.ones(m))
X = np.hstack((one.T,mbill.T))

In [7]: #set k here
ypred = localWeightRegression(X,mtip,2)
SortIndex = X[:,1].argsort(0)
xsort = X[SortIndex][:,0]

In [8]: fig = plt.figure()
ax = fig.add_subplot(1,1,1)
ax.scatter(bill,tip, color='green')
ax.plot(xsort[:,1],ypred[SortIndex], color = 'red', linewidth=5)
plt.xlabel('Total bill')
plt.ylabel('Tip')
plt.show();
```

The output of the final cell shows a scatter plot with 'Total bill' on the x-axis and 'Tip' on the y-axis. Green dots represent the data points, and a red line represents the local weight regression fit. The plot is displayed in a window titled 'Figure'.





## For Iris Dataset

10tip - Jupyter Notebook x 10iris - Jupyter Notebook x +

localhost:8888/notebooks/MLL/10iris.ipynb

jupyter 10iris Last Checkpoint: 11 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

```
In [1]: import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

In [2]: def kernel(point,xmat, k):
m,n = np.shape(xmat)
weights = np.mat(np.eye((m)))
for j in range(m):
diff = point - X[j]
weights[j,j] = np.exp(diff*diff.T/(-2.0*k**2))
return weights

In [3]: def localWeight(point,xmat,ymat,k):
wei = kernel(point,xmat,k)
W=(X.T*(wei*X)).I*(X.T*(wei*ymat.T))
return W

In [4]: def localWeightRegression(xmat,ymat,k):
m,n = np.shape(xmat)
ypred = np.zeros(m)
for i in range(m):
ypred[i] = xmat[i]*localWeight(xmat[i],xmat,ymat,k)
return ypred

In [9]: # Load data points
data = pd.read_csv('irisregression.csv')
sepal = np.array(data.sepal)
petal = np.array(data.petal)
```

