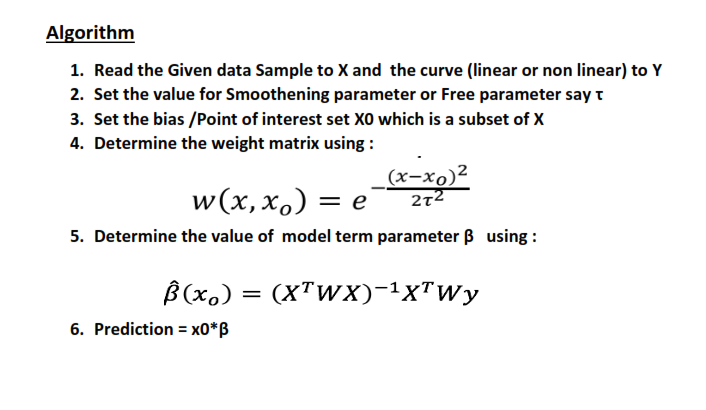
10.Problem: Implement the non-parametric Locally Weighted Regression (LOWESS) algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.



Program code :

|  |
| --- |
| import numpy as np  from bokeh.plotting import figure, show, output\_notebook  from bokeh.layouts import gridplot  from bokeh.io import push\_notebook  output\_notebook() |

|  |
| --- |
| import numpy as np  def local\_regression(x0, X, Y, tau):  # add bias term  x0 = np.r\_[1, x0]  X = np.c\_[np.ones(len(X)), X]      xw = X.T \* radial\_kernel(x0, X, tau)    beta = np.linalg.pinv(xw @ X) @ xw @ Y      return x0 @ beta  def radial\_kernel(x0, X, tau):  return np.exp(np.sum((X - x0) \*\* 2, axis=1) / (-2 \* tau \* tau)) |

|  |
| --- |
| n = 1000  # generate dataset  X = np.linspace(-3, 3, num=n)  print("The Data Set ( 10 Samples) X :\n",X[1:10])  Y = np.log(np.abs(X \*\* 2 - 1) + .5)  print("The Fitting Curve Data Set (10 Samples) Y :\n",Y[1:10])  # jitter X  X += np.random.normal(scale=.1, size=n)  print("Normalised (10 Samples) X :\n",X[1:10]) |

|  |
| --- |
| domain = np.linspace(-3, 3, num=300)  print(" Xo Domain Space(10 Samples) :\n",domain[1:10])  def plot\_lwr(tau):  prediction = [local\_regression(x0, X, Y, tau) for x0 in domain]  plot = figure(plot\_width=400, plot\_height=400)  plot.title.text='tau=%g' % tau  plot.scatter(X, Y, alpha=.3)  plot.line(domain, prediction, line\_width=2, color='red')  return plot |

*# Plotting the curves with different tau*

|  |
| --- |
| # Plotting the curves with different tau  show(gridplot([  [plot\_lwr(10.), plot\_lwr(1.)],  [plot\_lwr(0.1), plot\_lwr(0.01)]  ])) |

