3) **[50 pts]** Locally Weighted Linear Regression

In this part of the assignment you are going to implement locally weighted linear regression. The difference of locally weighted linear regression from normal linear regression is that in the latter case all the weights (the $w^{(i)}$'s) were considered to be the same (i.e. all 1's). You will minimize the following function.

$$MSE(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} w^{(i)}\left(\boldsymbol{\theta}^T \boldsymbol{x}^{(i)} - y^{(i)}\right)^2$$

Generate synthetic data using a function $y = \sin(2\pi x) + \varepsilon$, where m=10 (number of training examples) and $\varepsilon$ is a random noise. Example in python:

```
m=100;
X=np.random.rand(m,1)*2
y = np.sin(2*math.pi*X)+np.random.randn(m,1)
```

I. **[30 pts]** Update the linear regression function that you implemented in (2) so that local weights are also considered in taking the gradient of a cost function. The function should have the following header:

*theta= weighted_linear_regression(X, Y, iteration_cnt, eta, x, tau)*

where x is a query point, tau is bandwidth parameter.

Use *eta=0.4* and *iterNo=100.*

When evaluating $h(\cdot)$ at a query point $x$, use weights

$$w^{(i)} = \exp\left(-\frac{(x^{(i)} - x)^2}{2\tau^2}\right)$$

With a bandwidth parameter $\tau = 0.1$. (Again, remember to include intercept term). Plot on the same figure the data and the curve resulting from your fit.

II. **[20 pts]** Repeat (I) five times with $\tau = 0.001, 0.01, 0.3, 1\ and\ 10$. Comment **briefly** on what happens to the fit when $\tau$ is too small or too large.