Assignment 01

In this question, you need to: (1) implement multiple linear regression using radial basis functions; (2) investigate the performance of the RBF regression and generate plots to illustrate that performance under various parameter settings; and (3) answer four short answer questions based on your implementation and investigation.

Step 1: Implementation

First, the radial basis function (RBF) with a fixed center and band-width is defined as:

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\} \text{ where } x\epsilon R^1$$

Then, one bias term and M RBFs with M centers and band-width values are used to define one (M+1)-dimensional feature vector as follows:

$$\phi(x) = [1, \phi_1(x), \phi_2(x), \dots, \phi_M(x)]$$

In this assignment, the center values for the M RBFs are defined and implemented in the following two ways:

a) *Evenly Spaced Center*. Search the max and min of the *x* values in the training set, then divide the *x* range between min and max into M intervals and have a center value to represent each interval:

numpy.linspace(min(x), max(x), M)

b) *Randomly Sampled Center*. Concerning that the training set have 20 data points, we generate M random and different values ranging from 0 to 19, then choose M random center values from the training set and sort out the M centers by value in an ascending order:

numpy.sort(x[numpy.random.choice(len(x),M,replace=False)])

Define a *fitdata(x,t,M,s,mu)* function to calculate the weights of the (M+1)-dimensional feature vector. To ensure that the matrix (X.T*X) is invertible, we convert *inv(X.T*X)* into *inv(X.T*X+numpy.eye ((X.T@X).shape[1])*10**-6)*.

Finally, using the testing set, we calculate the estimated *y* values based on the feature vector and the weights, and also calculate the real *t* values based on the function *x/(1+x)*, and then plot a comparison between the estimated and real values, shown in Step 2.

Additionally, some other functions are defined to make the program succinct and easy to implement and understand:

a) *plotData(x1,t1,x2=None,t2=None,x3=None,t3=None,x4=None,t4=None,legend=[])*: plot multiple curves to show the changing patterns between *x* and *t*.

b) *plot_Ms(x1,t1,x2,M,s)*: plot multiple figures under different M and s values.

c) *testdata_even(x1,t1,x2,M,s)*: a loop to calculate the average absolute error *(|y-t|)* on the testing set with different M values using the RBFs with evenly spaced centers.

d) *testdata_random(x1,t1,x2,M,s)*: a loop to calculate the average absolute error *(|y-t|)* on the testing set with different M values using the RBFs with randomly sampled centers, and then calculate the mean and standard derivation of the average absolute errors for 10 runs regarding each M value.

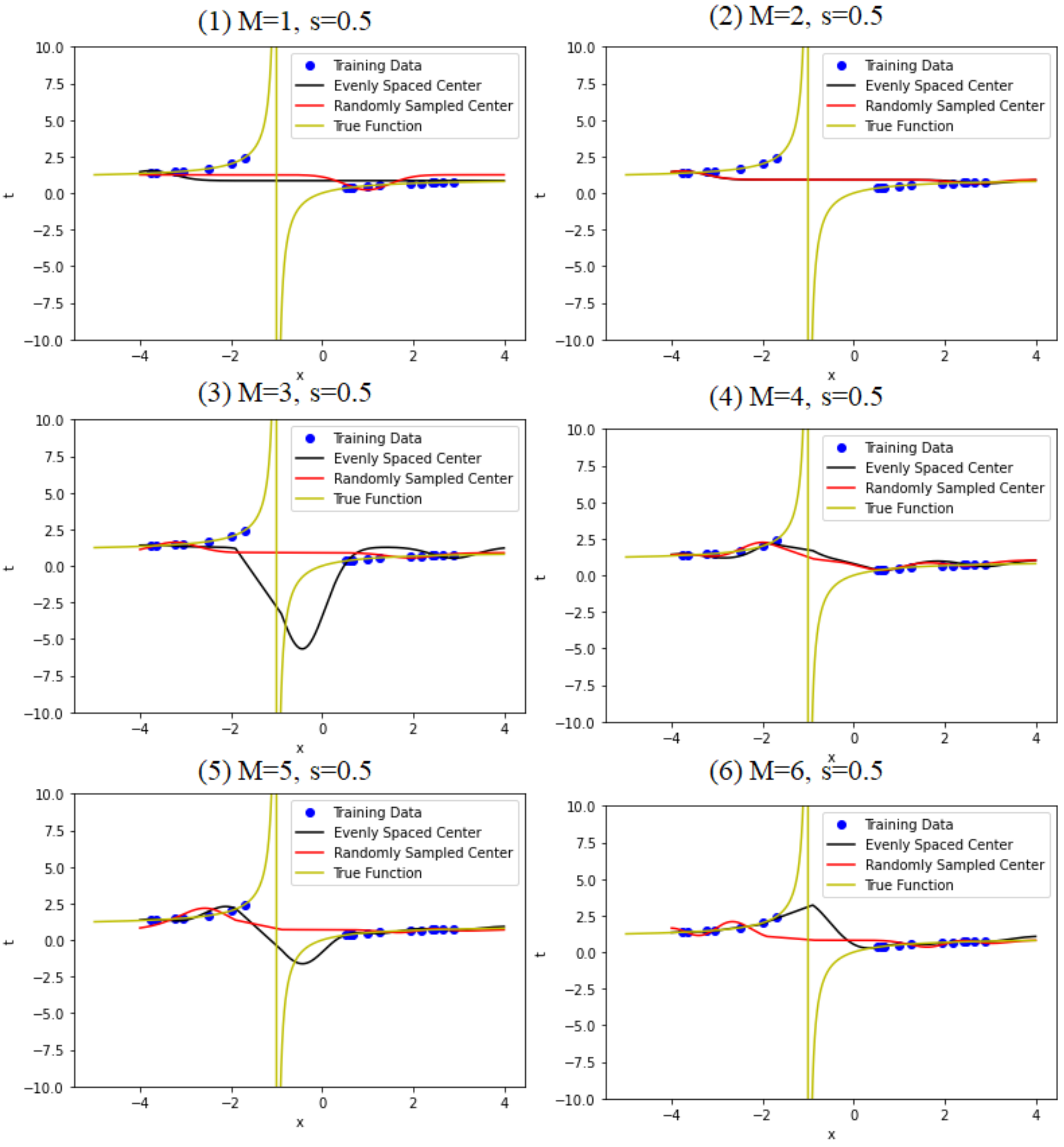
e) *testdata_polynomial(x1,t1,x2,M)*: a loop to calculate the average absolute error *(|y-t|)* on the testing set with various M values using polynomial basis function.

Step 2: Investigation

In this assignment, we investigate and plot the performance of Step 1 Implementation under different parameter settings (a range of M and *s* values).

A) Vary M from 1 to 20 (the number of the data points in the training set) and set s=0.5.

a) The scatter plot of training data, and the lines describing the estimated *y* values using the RBFs with evenly spaced centers and randomly sampled centers and the real t values based on the true function *x/(1+x)* on the testing set.



(1) M=1, s=0.5

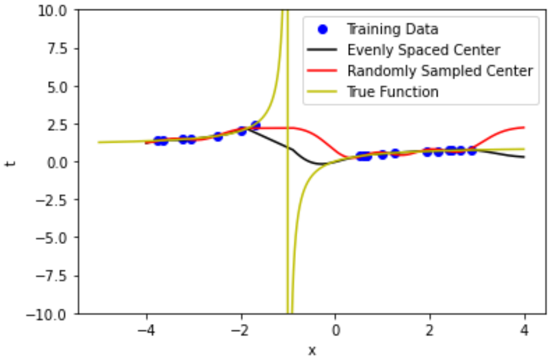
(2) M=2, s=0.5

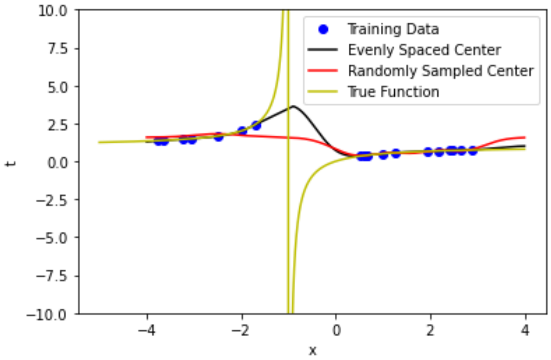
(3) M=3, s=0.5

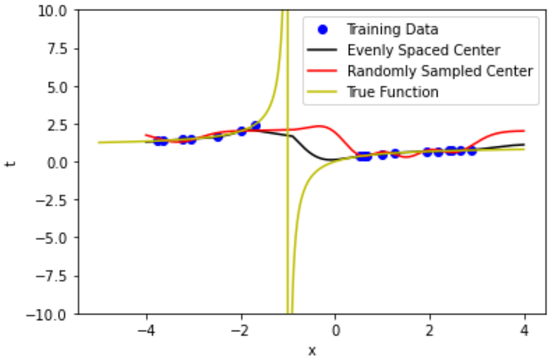
(4) M=4, s=0.5

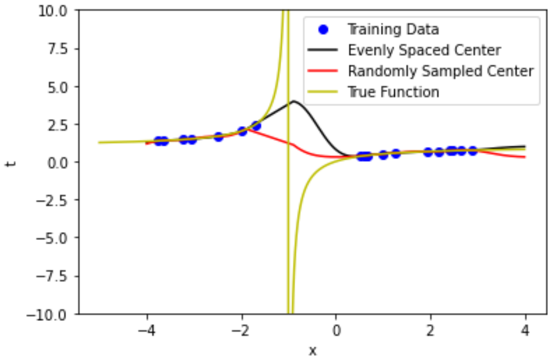
(5) M=5, s=0.5

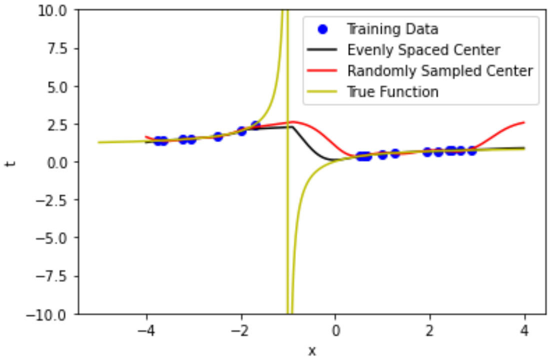
(6) M=6, s=0.5

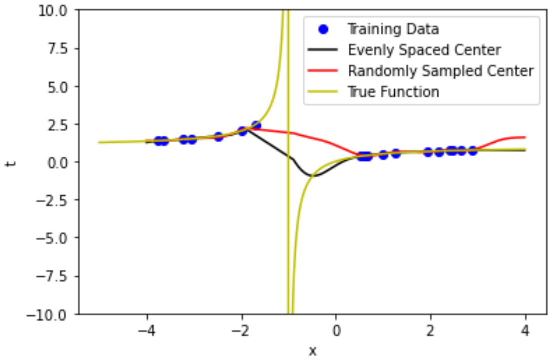

(7) M=7, s=0.5

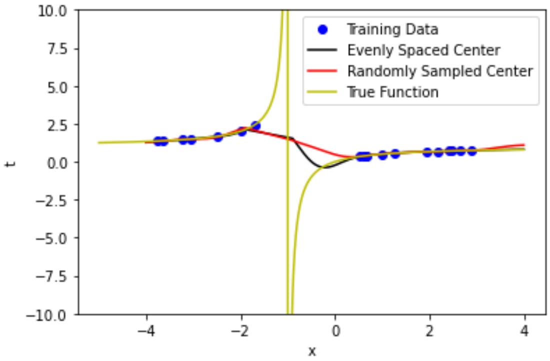

(8) M=8, s=0.5

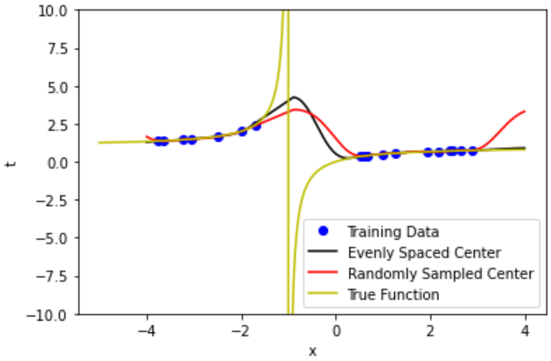

(9) M=9, s=0.5

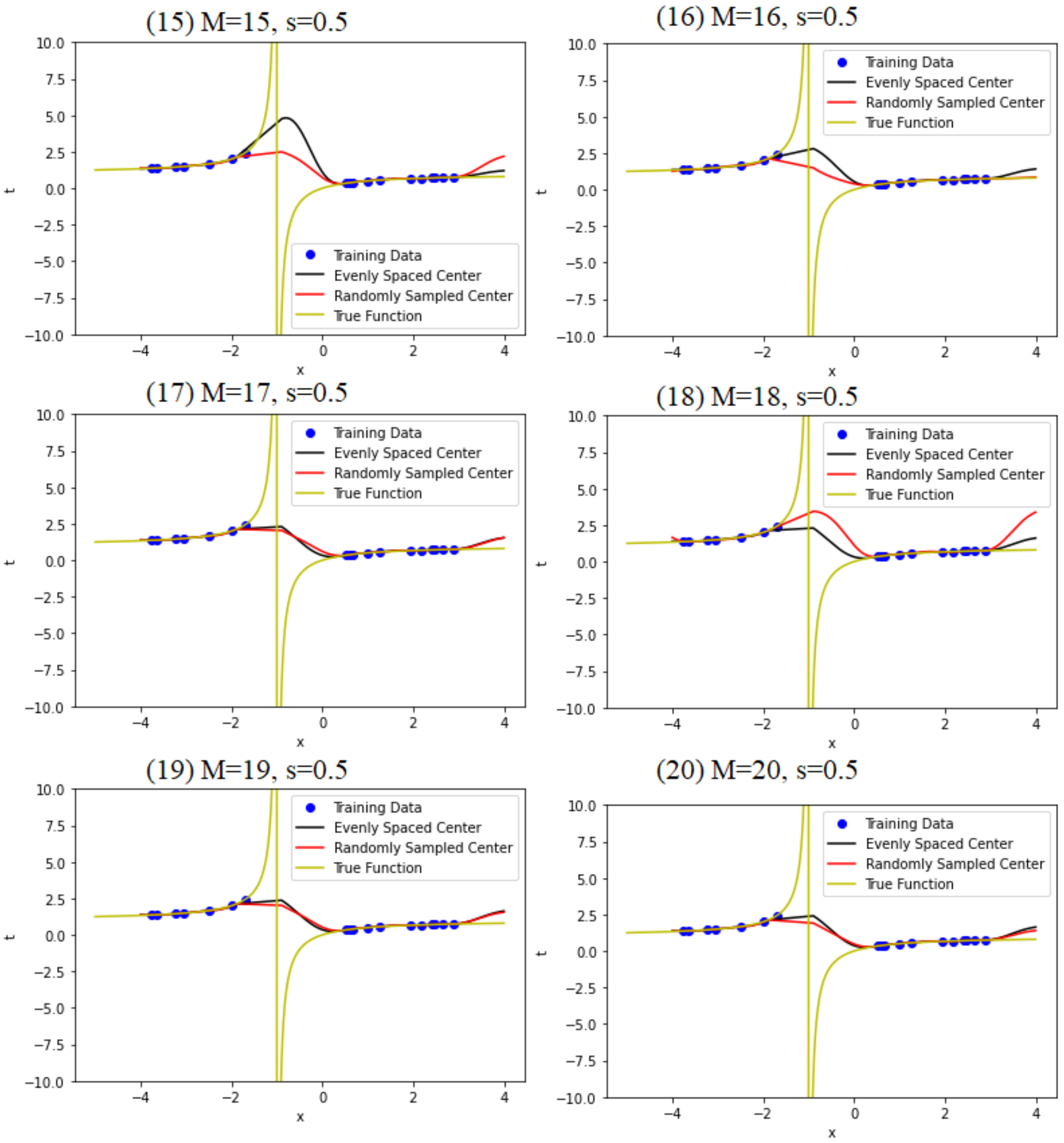

(10) M=10, s=0.5


(11) M=11, s=0.5


(12) M=12, s=0.5


(13) M=13, s=0.5


(14) M=14, s=0.5


(15) M=15, s=0.5

(16) M=16, s=0.5

(17) M=17, s=0.5

(18) M=18, s=0.5

(19) M=19, s=0.5

(20) M=20, s=0.5

Figure 1 The performance of the RBFs with varying M from 1 to 20 and s=0.5

b) A line plotting the variations between the M value and the average absolute error (|y-t|) on the testing set using the RBFs with evenly spaced centers, and an error bar describing the mean and standard derivation of the average absolute error for 10 runs using the RBFs with randomly sampled centers.

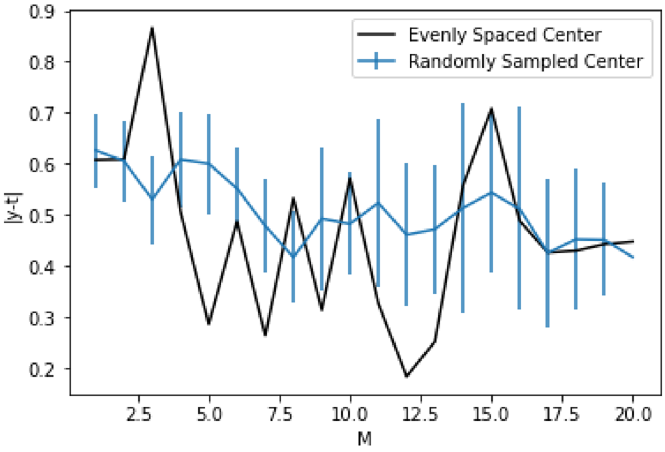
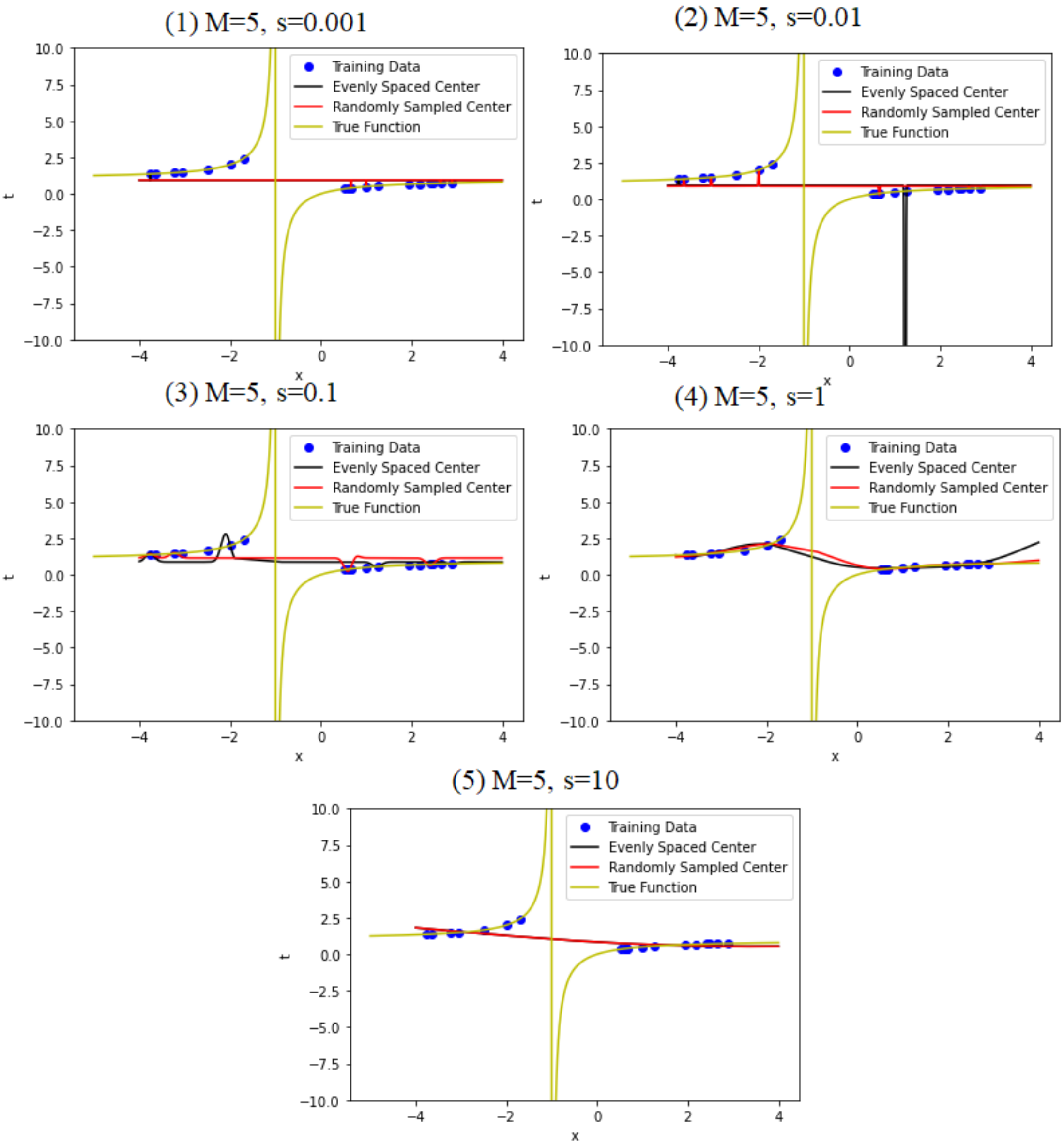


Figure 2 The changes in average absolute error on the testing set with varying M from 1 to 20 and s=0.5

In summary, when M varies from 1 to 20 and s=0.5, using the RBFs with randomly sampled centers shows better performances than that with evenly spaced centers due to less average absolute error on the data points in the testing set. Furthermore, using the RBFs with randomly sampled centers indicates more stable model performances with the changes in M than that with evenly spaced centers. However, when M=11 and s=0.5, using the RBFs with evenly spaced centers brings the least average absolute error and exhibits the best model performance on the testing set. By contrast, when M=3 and s=0.5, using the RBFs with evenly spaced centers brings the largest average absolute error and exhibits the worst model performance on the testing set.

B) Vary s from 0.001 to 10 and set M=5

a) The scatter plot of training data, and the lines describing the estimated *y* values using the RBFs with evenly spaced centers and randomly sampled centers and the real t values based on the true function *x/(1+x)* on the testing set. Vary the s value in the range [0.001,0.01,0.1,1,10].


(1) M=5, s=0.001

(2) M=5, s=0.01

(3) M=5, s=0.1

(4) M=5, s=1

(5) M=5, s=10

Figure 3 The performance of the RBFs with varying s from 0.001 to 10 and M=5

b) A line plotting the variations in the s value and the average absolute error (|y-t|) on the testing data using the RBFs with evenly spaced centers, and one error bar to describe the mean and standard derivation of the average absolute errors for 10 runs using the RBFs with randomly sampled centers. Vary the s value in the range [0.001,0.005,0.01,0.05,0.1,0.5,1,5,10]. In this figure, the logarithms of the s values are calculated as the x-axis to make the figure readable and easy to understand.

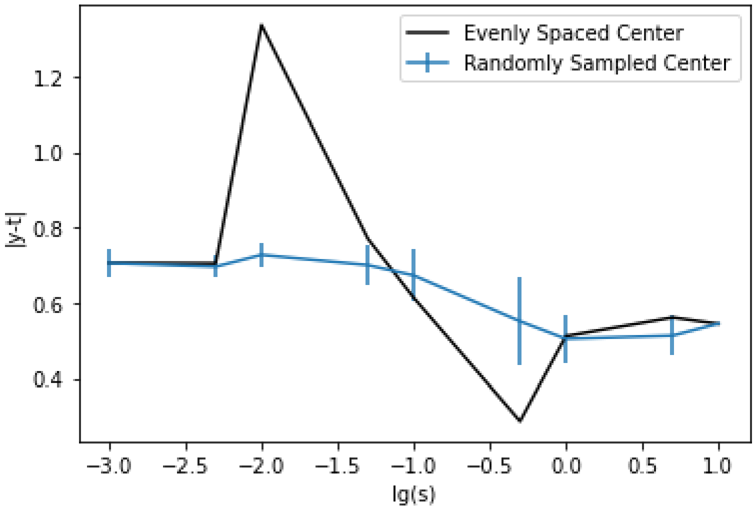


Figure 4 The changes in average absolute error on the testing set with varying s from 0.001 to 10 and M=5

Similarly, when s varies from 0.001 to 10 and M=5, using the RBFs with randomly sampled centers shows better performances than that with evenly spaced centers due to less average absolute error on the data points in the testing set. Furthermore, using the RBFs with randomly sampled centers indicates more stable model performances with the changes in s than that with evenly spaced centers. However, when M=5 and s=0.5, using the RBFs with evenly spaced centers brings the least average absolute error and exhibits the best performance on the testing set. By contrast, when M=5 and s=0.01, using the RBFs with evenly spaced centers brings the largest average absolute error and exhibits the worst performance on the testing set.

Step 3: Discussion

1) Do the RBFs outperform (in terms of error between predicted and desired) the polynomial basis function with the same M value on the provided training and testing data? Why or why not? Are there cases where the RBF outperforms and cases where the polynomial basis outperforms? What are those cases and why does (or does not) this occur?

A: As illustrated in Figure 5, when M varies from 1 to 20, the RBFs with evenly spaced centers and randomly sampled centers generally outperform the polynomial basis function with the same M value on the testing set, particularly for larger M values. When the M value is less than 8, the polynomial basis function presents slightly worse performances than the RBFs. This mainly lies in that the model performance on the testing set is mainly limited by the under-fitting phenomenon due to very limited number of the data points in the training set. However, when the M value gradually increases from 8 to 20, the feature vector consisting of the RBFs or polynomial basis functions exerts a greater influence on the model performances than other factors (e.g., the training data). This is mainly due to the fact that overfitting occurs more easily on the polynomial basis functions than the RBFs. Therefore, the RBFs are a better choice to fit the true function in this assignment.

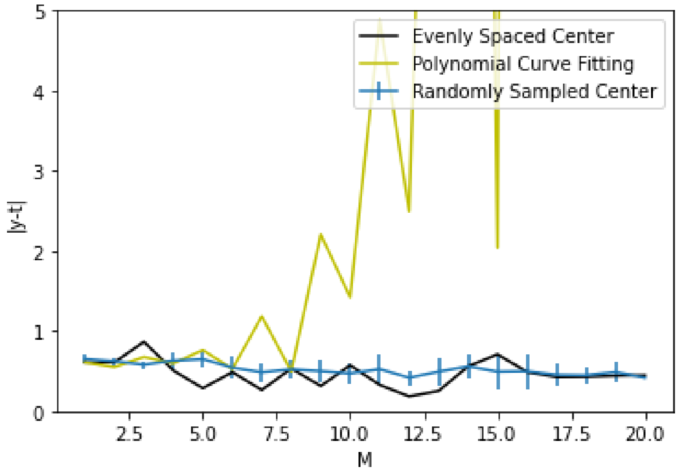


Figure 5 The comparison between the RBFs and polynomial basis functions in the average absolute error on the testing set with varying M from 1 to 20

2) Do any of the generated plots show indication of overfitting in any of your results? Why or why not?

A: Yes, a few of the generated plots show indication of overfitting in some of my results, as shown in Figure 1 and 2, especially for those with larger M values. The RBFs with larger M values present better performances on the data points in the training set, however, they show worse model performances on the testing set, in terms of the error between the estimated and real values, particularly in the *x* range closer to -1.0. This result is mainly attributed to very limited number of data points in the training set so that the RBFs based on the training set cannot learn many features hidden in the true function. Furthermore, an absence of the *x* values closer to -1.0 in the training set exerts a great influence on the model performance of using the RBFs to fit the true function because these *x* values determine some of the most important features hidden in the true function.

3) What is the role of the parameter s and how does the choice of s effect results?

A: The parameter s partly determines the Gaussian distance between the *x* values and the center values. This means that the *x* values far away from the centers exhibit weaker effects on calculating the weights of the feature vector that those closer to the centers. The RBFs with the parameter s can efficiently weaken the interfere of the *x* values far away from the centers on the model results. As shown in Figure 4, as the s value increases, the average absolute error between the estimated and real values on the testing set generally shows a gradually decreasing trend, particularly for the RBFs with randomly sampled centers. This demonstrates that a larger s value presents better model performances, as shown in this assignment.

4) How do the evenly space vs. randomly selected center values impact performance? When is one better than the other? Why?

A: The evenly spaced centers and randomly sampled centers are closely related to the RBFs and the feature vectors $\phi(x)$. Specifically, when the center values vary, the format of the RBFs will be different, which also leads to the changes in the weights of the feature vectors. Therefore, the model performances of using the RBFs and the feature vectors to fit the true function, in terms of the error between the estimated and real values, will vary depending on the selection of the center values.

As a whole, when M and s separately vary from 1 to 20 and from 0.001 to 10, using the RBFs with randomly sampled centers shows better performances than that with evenly spaced centers due to less average absolute error on the data points in the testing set. Furthermore, using the RBFs with randomly sampled centers indicates more stable model performances with the changes in M and s than that with evenly spaced centers. However, when M=11 and s=0.5, using the RBFs with evenly spaced centers brings the least average absolute error and exhibits the best performance on the testing set. By contrast, when M=5 and s=0.01, using the RBFs with evenly spaced centers brings the largest average absolute error and exhibits the worst performance on the testing set.

Generally, when M is smaller (1 to 3), using the RBFs with randomly sampled centers exhibits better model performances than that with evenly spaced centers. This is mainly due to the fact that when the M value is small, evenly spaced centers often refer to the min and max of the *x* values, which is unfavorable for the model performances of the RBFs. When M varies from 4 to 14, using the RBFs with evenly spaced centers shows less average absolute error on the testing set than that with randomly sampled centers. This can be explained by that using the RBFs with evenly spaced centers exhibits higher potentials for selecting more representative center values from the training set than that with randomly sampled centers. When the M value is larger than 15, almost all *x* values of the training set are chosen as the center values, and therefore, using the RBFs with evenly spaced centers shows nearly equivalent model performances to that with randomly sampled centers.