ENGR 421/DASC 521: Introduction to Machine Learning
Homework 3: Nonparametric Regression
Deadline: April 21, 2025, 11:59 PM

In this homework, you will implement three nonparametric regression algorithms in Python. Here are the steps you need to follow:

1. Read Section 8.8 from the textbook.

2. You are given a bivariate regression data set, which contains 1000 training data points in the file named `hw03_data_set_train.csv` and 1000 test data points in the file named `hw03_data_set_test.csv`.

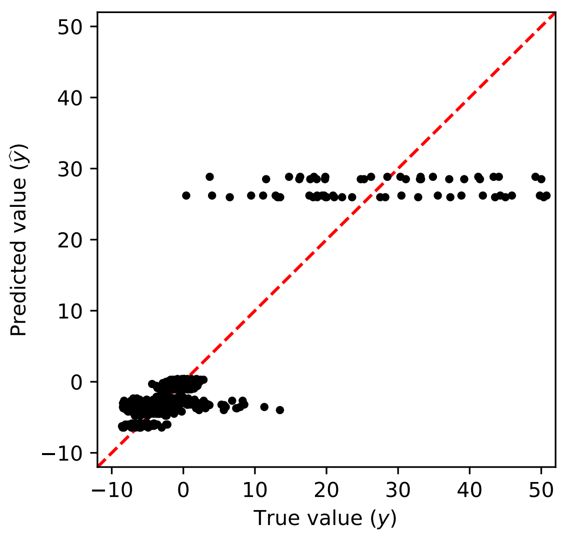
3. Learn a regressogram by setting the bin width parameter to 0.5 and the origin parameter to $(0, 0)$. (30 points)

$$g(\boldsymbol{x}) = \frac{\sum_{i=1}^{N_{train}} b(\boldsymbol{x}, \boldsymbol{x}_i) y_i}{\sum_{i=1}^{N_{train}} b(\boldsymbol{x}, \boldsymbol{x}_i)} \qquad \text{where } b(\boldsymbol{x}, \boldsymbol{x}_i) = \begin{cases} 1 & \text{if } \boldsymbol{x}_i \text{ is in the same bin with } \boldsymbol{x}, \\ 0 & \text{otherwise.} \end{cases}$$

When you calculate the root mean squared error (RMSE) of your regressogram for training data points, you should obtain the following sentence as your output.

```
Regressogram => RMSE on training set is 3.8878335748844512 when h is 0.5
```

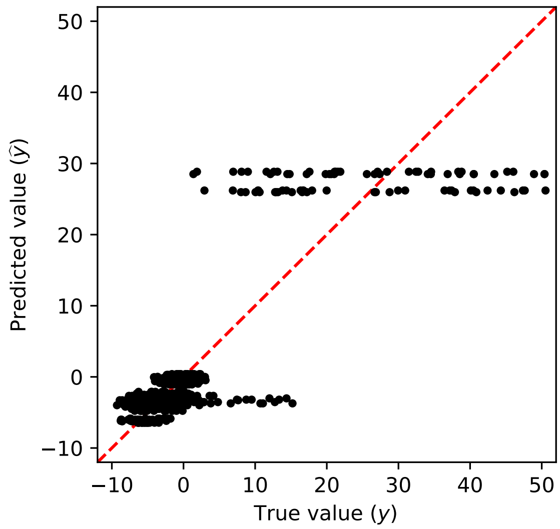
When you draw true values and predicted values obtained by your regressogram for training data points, you should obtain the following figure.



When you calculate the root mean squared error (RMSE) of your regressogram for test data points, you should obtain the following sentence as your output.

```
Regressogram => RMSE on test set is 4.583204089846181 when h is 0.5
```

When you draw true values and predicted values obtained by your regressogram for test data points, you should obtain the following figure.



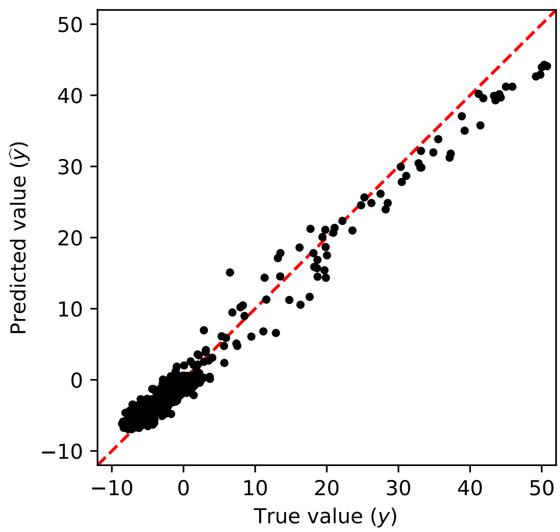
4. Learn a running mean smoother by setting the bin width parameter to 0.5. (35 points)

$$g(\boldsymbol{x}) = \frac{\sum_{i=1}^{N_{train}} w\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h}\right) y_i}{\sum_{i=1}^{N_{train}} w\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h}\right)} \qquad \text{where } w(\boldsymbol{u}) = \begin{cases} 1 & \text{if } -\frac{1}{2}\mathbf{1}_D \leq \boldsymbol{u} < \frac{1}{2}\mathbf{1}_D, \\ 0 & \text{otherwise.} \end{cases}$$

When you calculate the RMSE of your running mean smoother for training data points, you should obtain the following sentence as your output.

```
Running Mean Smoother => RMSE on training set is 1.4592587037889608 when h is 0.5
```

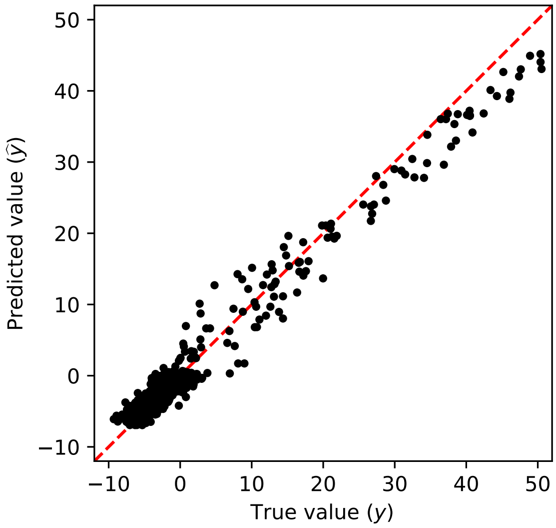
When you draw true values and predicted values obtained by your running mean smoother for training data points, you should obtain the following figure.



When you calculate the RMSE of your running mean smoother for test data points, you should obtain the following sentence as your output.

```
Running Mean Smoother => RMSE on test set is 1.689617624797976 when h is 0.5
```

When you draw true values and predicted values obtained by your running mean smoother for test data points, you should obtain the following figure.



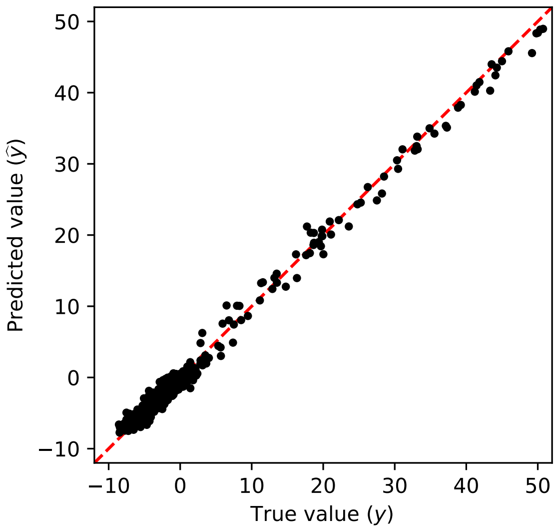
5. Learn a kernel smoother by setting the bin width parameter to 0.08. (35 points)

$$g(\boldsymbol{x}) = \frac{\sum_{i=1}^{N_{train}} K\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h}\right) y_i}{\sum_{i=1}^{N_{train}} K\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h}\right)} \qquad \text{where } K(\boldsymbol{u}) = \frac{1}{\sqrt{(2\pi)^D}} \exp\left(-\frac{\boldsymbol{u}^\top \boldsymbol{u}}{2}\right)$$

When you calculate the RMSE of your kernel smoother for training data points, you should obtain the following sentence as your output.

```
Kernel Smoother => RMSE on training set is 0.859816920753896 when h is 0.08
```

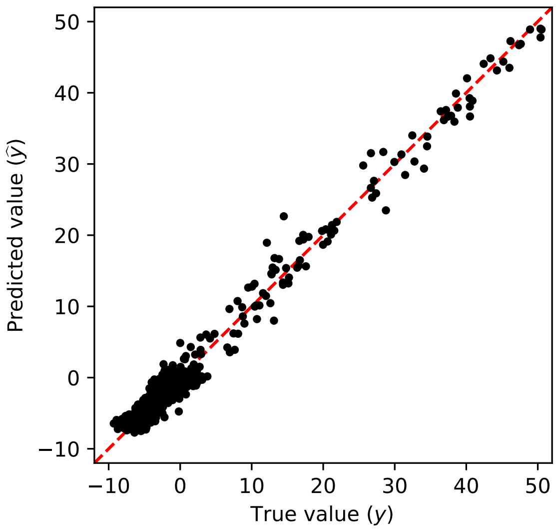
When you draw true values and predicted values obtained by your kernel smoother for training data points, you should obtain the following figure.



When you calculate the RMSE of your kernel smoother for test data points, you should obtain the following sentence as your output.

```
Kernel Smoother => RMSE on test set is 1.3297515638899235 when h is 0.08
```

When you draw true values and predicted values obtained by your kernel smoother for test data points, you should obtain the following figure.



What to submit: You need to submit your source code in a single file (`.py` file). You are provided with a template file named as `0099999.py`, where `99999` should be replaced with your 5-digit student number. You are allowed to change the template file between the following lines.

```
# your implementation starts below

# your implementation ends above
```

How to submit: Submit the file you edited to LearnHub by following the exact style mentioned. Submissions that do not follow these guidelines will not be graded.

Late submission policy: Late submissions will not be graded.

Cheating policy: Very similar submissions will not be graded.
