Data Mining and Statistical Learning

Homework 4

Introduction

Nonparametric regression estimators (also known as **"smoothers"**) attempt to estimate the unknown function $f(X)$ from a sample of noisy data over certain domain (R^P) by estimating what $f(x)$ is at a point x_0. This can be accomplished by using only those observations close to x_0 to fit a simple model and the resulting estimated function $\hat{f}(X)$ is smooth in R^p. This is all achieved using a kernel function $K_h(x_0, x_i)$ where h is the smoothing parameter which needs to be determined. In practice, the smoothing parameter $h \approx n^{\frac{-1}{5}}$ where n is the number of samples in the data.

There are several local smoothing methods like **LOESS**, **Nadaraya-Watson**, and **Spline** and the goal of this homework is to understand their statistical properties and computational challenges.

Problem Statement and Data Set

The objective of our analysis as previously stated is to understand the statistical properties and computational challenges of three different types of local smoothing methods: **LOESS**, **Nadaraya-Watson**, and **Spline** smoothing.

For this purpose, we will compute the empirical bias, empirical variances, and empirical mean square error (MSE) based on $m = 1000$ Monte Carlo runs, where in each run we simulate a data set of $n = 101$ observations from the additive noise model $Y_i = f(x_i) + \epsilon_i$ with the famous Ricker's wavelet, also known as the Mexican Hat function defined by $f(x) = (1 - x^2)e^{-0.5x^2}$ over the interval $-2\pi \leq x \leq 2\pi$.

The added white noise $\epsilon_1 ... \epsilon_n$ are independent and identically distributed (iid) $\approx N(0, 0.2^2)$.

The Mexican hat function is notoriously known to pose a variety of estimation challenges. Thus, this report will attempt to explore the inherent difficulties of this function.

Exploratory Data Analysis

We first start this analysis by looking what the Mexican hat looks like for both equidistant and non-equidistant designs.

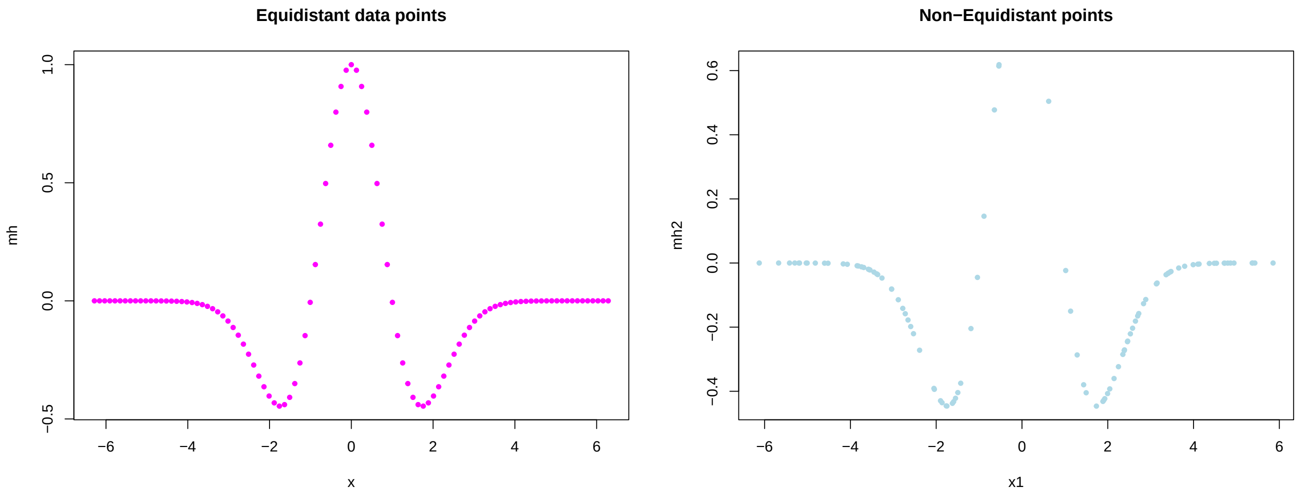

Figure 1: The Mexican hat function

We can see that the Mexican hat function converges to zero when x approaches $-\infty$ or $+\infty$. Now, let's look at the distribution of the Ricker's wavelet through a histogram plot

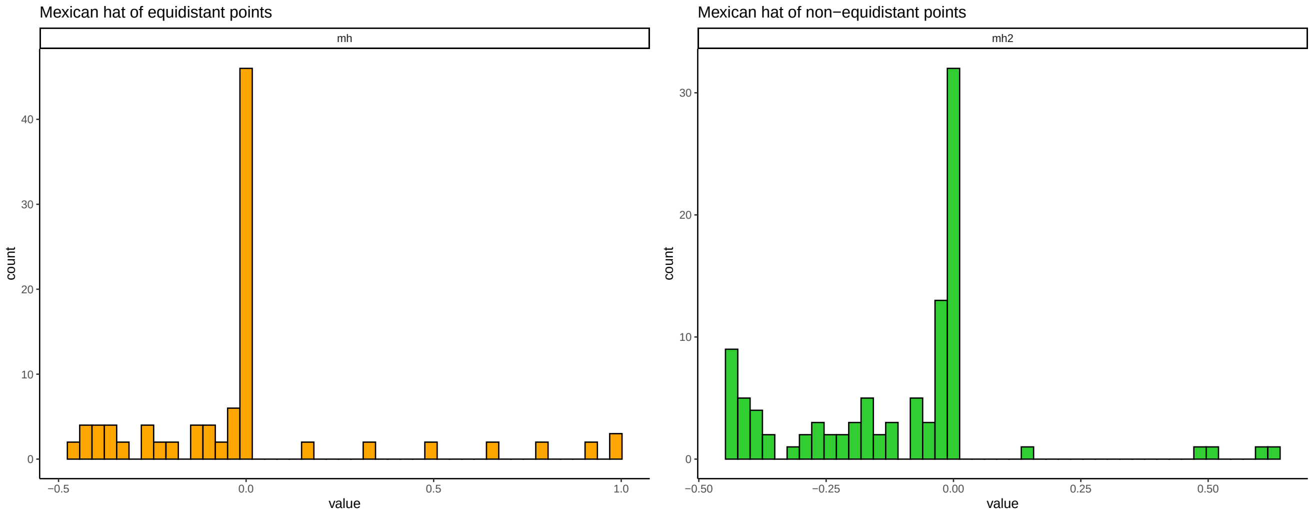

Figure 2: The histogram plot of the mexican hat

The histograms are as expected. Additionally, we can see roughly where the peaks of the distribution are, whether the distribution is skewed or symmetric, and if there are any outliers.

Methodology

This analysis is split into two deterministic experiments where in the first experiment, we use an equidistant points in the interval $[-2\pi, 2\pi]$, and in second we use a non equidistant points in the same interval as inputs to the Mexican hat function.

In the first experiment, we run a $m = 1000$ Monte Carlo run to generate a data set of the form (x_i, Y_i) with $x_i = 2\pi(-1 + 2\frac{i-1}{n-1})$ and $Y_i = f(x_i) + \epsilon_i$. For each Monte Carlo run, we compute the three different kinds of local smoothing estimates at every point in simulated dataset: loess (with span = 0.75), Nadaraya-Watson (NW) kernel smoothing with Gaussian Kernel and bandwidth = 0.2, and spline smoothing with the default tuning parameter.

At each point x_i; for each local smoothing method, based on $m = 1000$ Monte Carlo runs, we compute the empirical bias, empirical variance, and empirical mean square error (MSE), which are defined as:

- $\widehat{\text{Bias}(f(x_i))} = \bar{f}_m(x_i) - f(x_i)$, where $\bar{f}_m(x_i) = \frac{1}{m}\sum_{j=1}^{m} f^{\hat{(j)}}(x_i)$
- $\widehat{\text{Var}(f(x_i))} = \frac{1}{m}\sum_{j=1}^{m} \left(f^{\hat{(j)}}(x_i) - \bar{f}_m(x_i)\right)^2$
- $\widehat{\text{MSE}(f(x_i))} = \frac{1}{m}\sum_{j=1}^{m} \left(f^{\hat{(j)}}(x_i) - f(x_i)\right)^2$, where $f(x_i) = (1 - x^2)e^{-0.5x^2}$

In the second experiment, we repeat the first experiment with non-equidistant points as previously mentioned. For simplicity and reasonable comparison, we use span = 0.3365 for loess, bandwidth = 0.2 for NW kernel smoothing, and spar = 0.7163 for Splines smoothing.

Next, we will explore the effect smoothing parameter by trying different values and looking at the fit of each smoother. Additionally, we will tune each smoothing parameter using a Leave-One-Out Cross Validation or LOOCV. Cross Validation is a crucial step to estimate the prediction error of each of the regression estimators.

LOOCV works by splitting the data into a training and testing sets, using all but one data point as part of the training set. We then build a model using the training set, predict on the single observation we left out, then calculate the mean square error of the model. We repeat the process n times, where n is the number of observations we have in the dataset.

After tuning the smoothing parameters, we will repeat the two experimental designs with the newly tuned parameters to assess their affect.

Results

Part 1: Experiments with default smoothing parameters

The results of each experimental design accompanied with plots can be summarized below

1. Deterministic equidistant design

We can plot (seen below)the the mean of the three local smoothing estimators: loess, NW kernel, and spline smoothing along with the raw observations to compare with the fitted curves

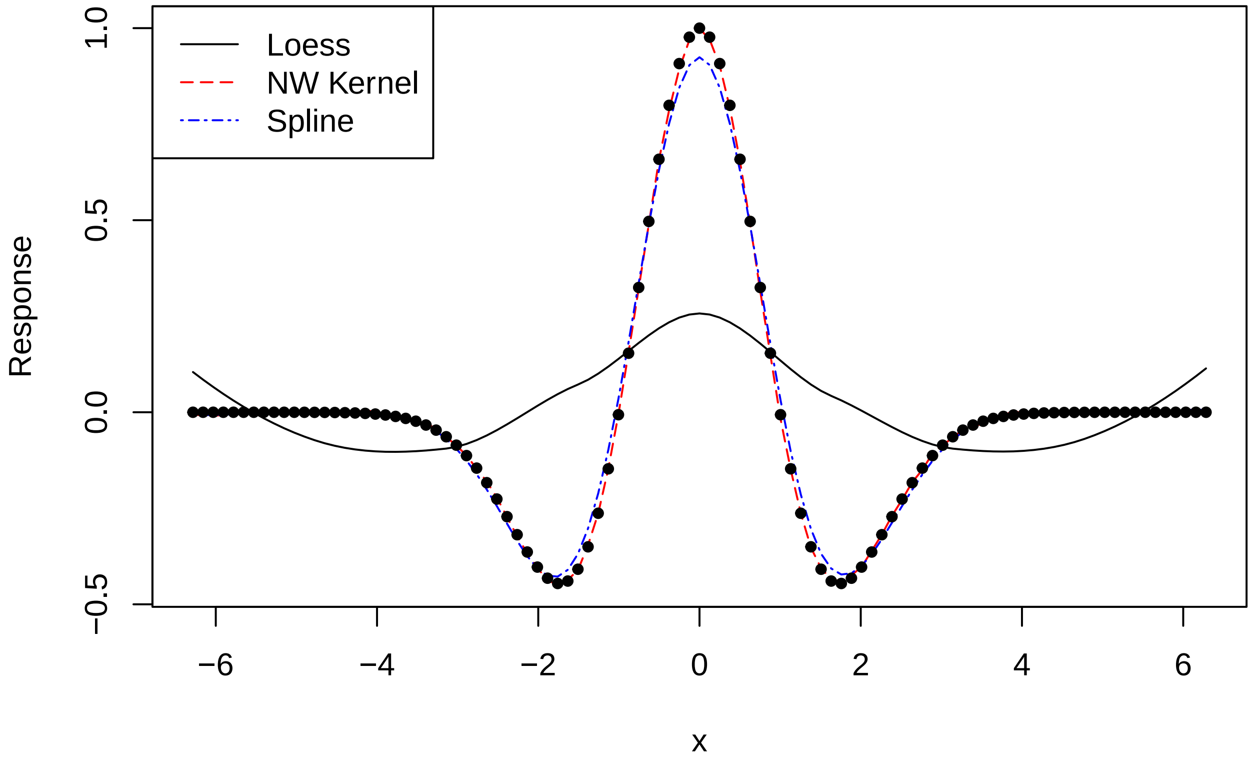

Figure 3: The mean of the three local smoothing estimators with raw observations

From looking at the mean of each estimator with the raw observations, we can clearly see that Loess kernel is probably not the best estimator for this special function. The other two kernel, seem to fit the data better with NW kernel performing the best.

Next, we will plot the bias and variance of three local smoothing estimators

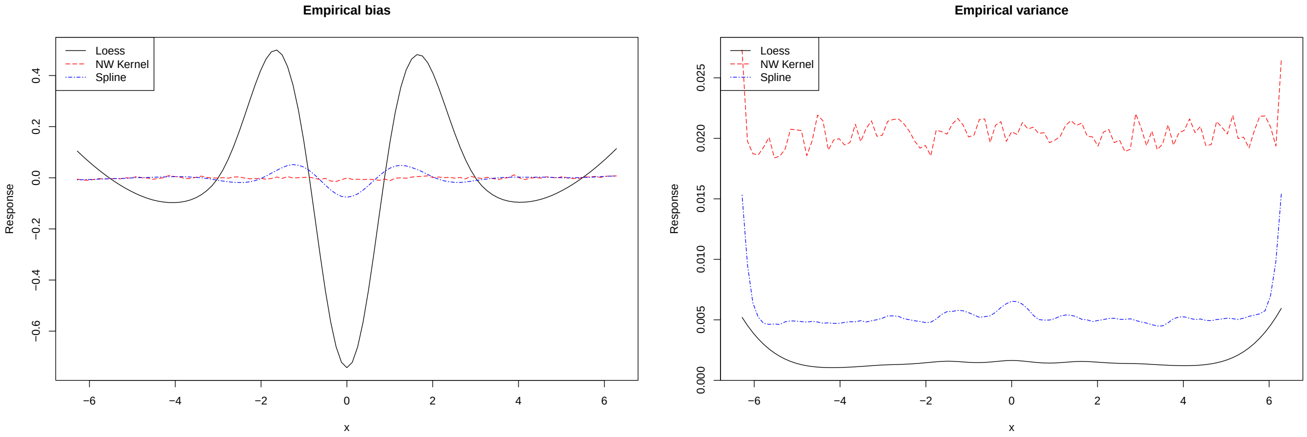

Figure 4: The empirical bias and variance of three local smoothing estimators

Loess estimator has the highest bias of the three estimator, while NW kernel has the lowest bias. Spline kernel performed better than Loess but we can see that it's very hard to estimate the Mexican hat function at $x_i = 0$, $x_i = +/-\frac{\pi}{2}$. In terms of variance, Loess has the lowest variance, while NW kernel has the highest variance

Next, we plot the empirical MSE of each local smoothing method using the following two methods:

- $\widehat{\text{MSE}(f(x_i))} = \frac{1}{m}\sum_{j=1}^{m}\left(f^{\hat{(j)}}(x_i) - f(x_i)\right)^2$
- $\text{MSE} = \text{BIAS}^2 + \text{VAR}$

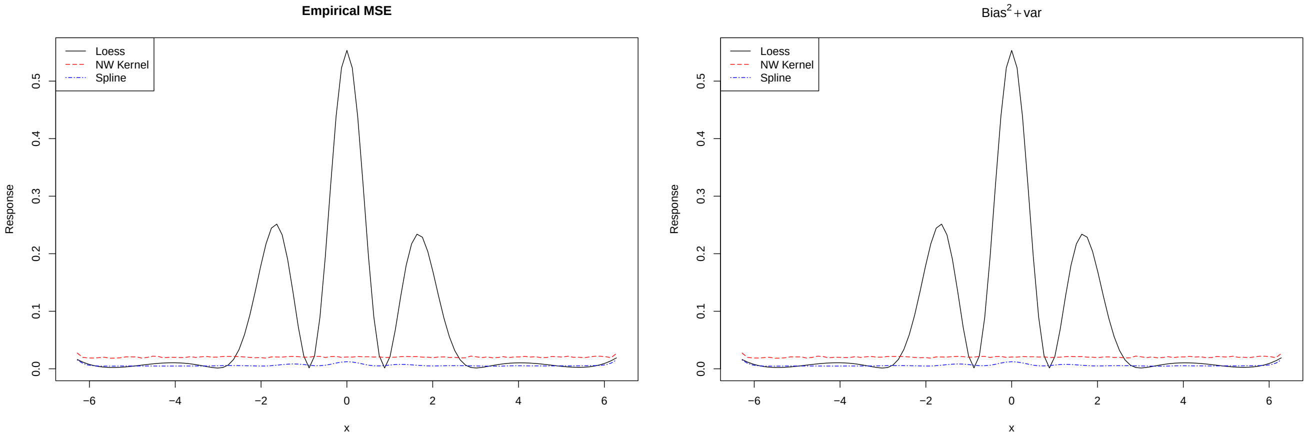

Figure 5: The MSE of the three local smoothing estimators

In terms of MSE, Loess has the highest MSE that peaks at $x_i = 0$, $x_i = +/-\frac{\pi}{2}$. Splines smoothing kernel has the lowest MSE.

2. Deterministic Non-Equidistant design

Again, we plot the the mean of the three local smoothing estimators along with the raw observations to compare with the fitted curves.

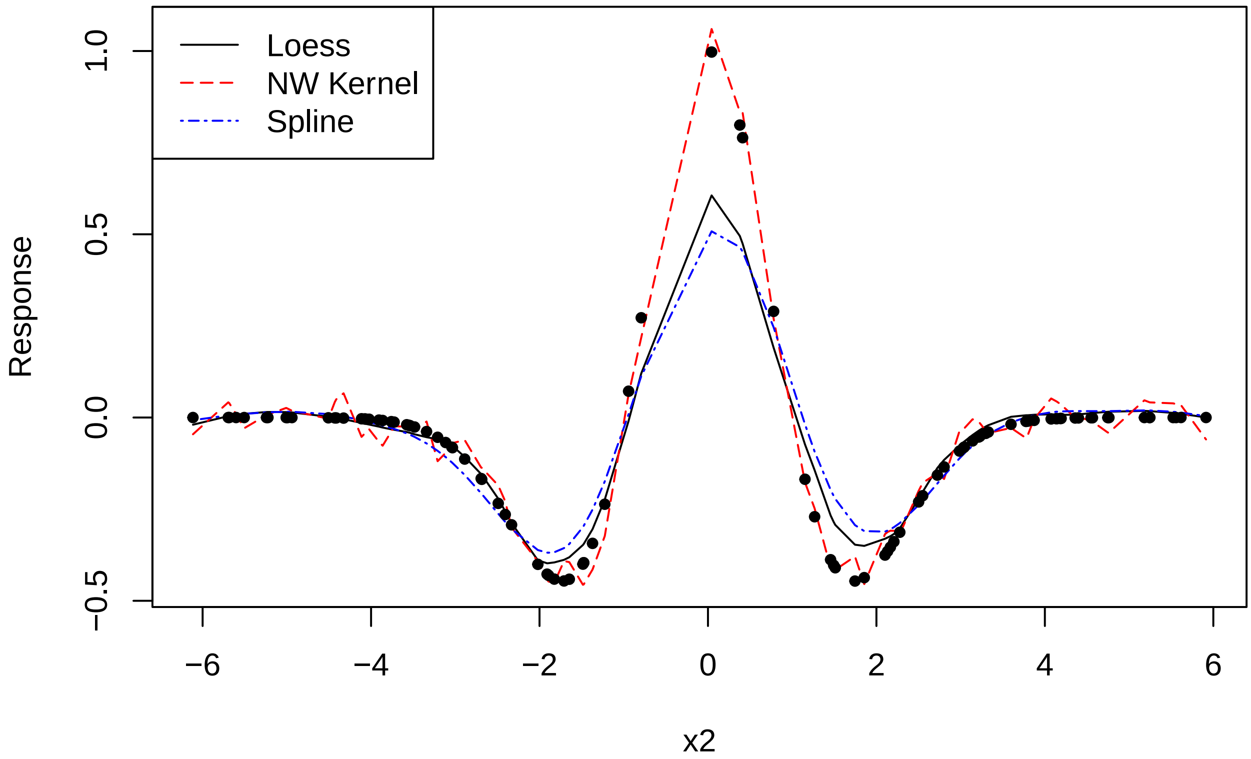

Figure 6: The mean of the three local smoothing estimators with raw observations

Looking at the mean of each estimator in this non-equidistant design, we see a non-smooth piece-wise fit. However, Loess estimator has performed much better than the equidistant design. NW kernel yet again outperforming the other two kernel estimators. Splines smoother didn't perform as good as the equidistant design.

Next, we look at the plots of the empirical bias and variance of three local smoothing estimators.

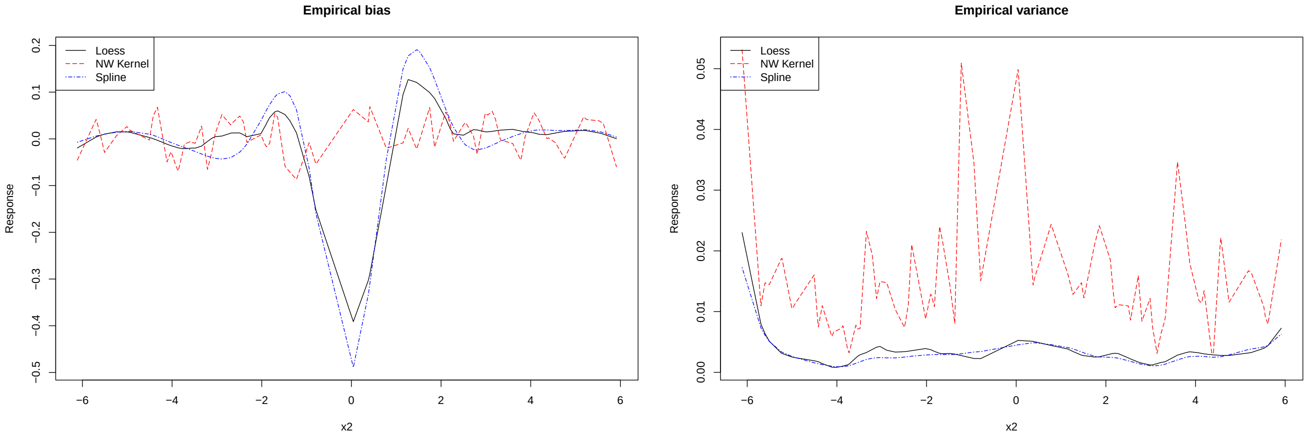

Figure 7: The empirical bias and variance of the three local smoothing estimators

From the bias plots above, we see that both Splines and Loess estimators had computational challenges at estimating the function at $x_i = 0$, $x_i = +/-\frac{\pi}{2}$. It is quite noticeable at $x_i = 0$. Both however has the lowest variance which hints to the bias-variance traeoff.

NW kernel has the lowest bias on average across all x_i points, but the highest variance.

Next, we look at the plots for the empirical MSE of each local smoothing method using the following:

- $\text{MSE}\widehat{(f(x_i))} = \frac{1}{m}\sum_{j=1}^{m}\left(f^{\hat{(j)}}(x_i) - f(x_i)\right)^2$
- $\text{MSE} = \text{BIAS}^2 + \text{VAR}$

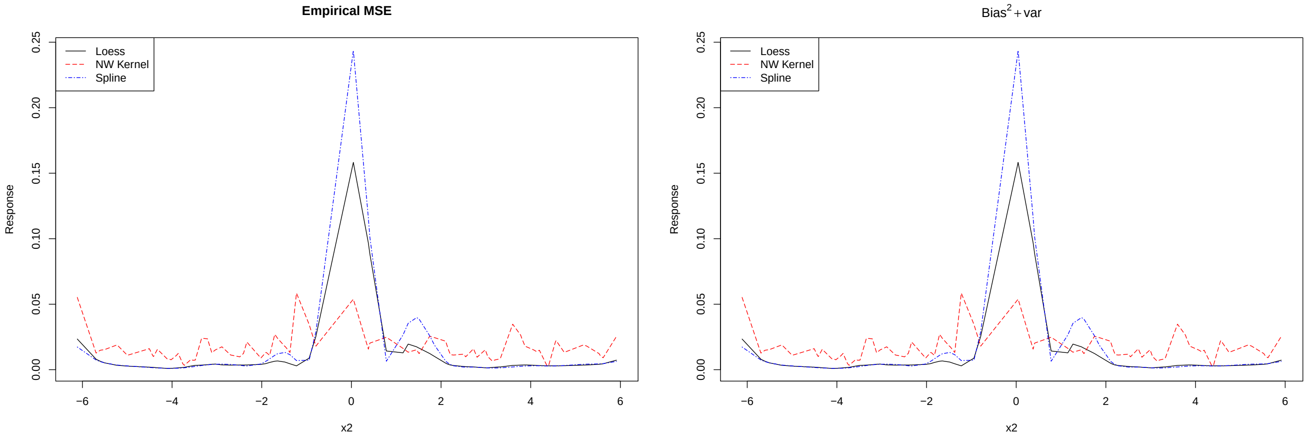

Figure 8: The MSE of the three local smoothing estimators

In terms of MSE, both Splines and Loess have the highest MSE that peaks at $x_i = 0$. NW smoothing kernel has the lowest MSE. The variance is so small that the meas square error is proportional to bias^2.

Part 2: Cross validation

Before we start fine tuning each of the smoothing parameters for each smoothing method, let's first see how the estimate model fit the function using different values for bandwidth, span, and spar for NW kernel, Loess, and Splines smoother respectively.

1. Effect of different smoothing parameters

We can examine the effect of each of the smoothing parameters using equidistant points through the following plots.

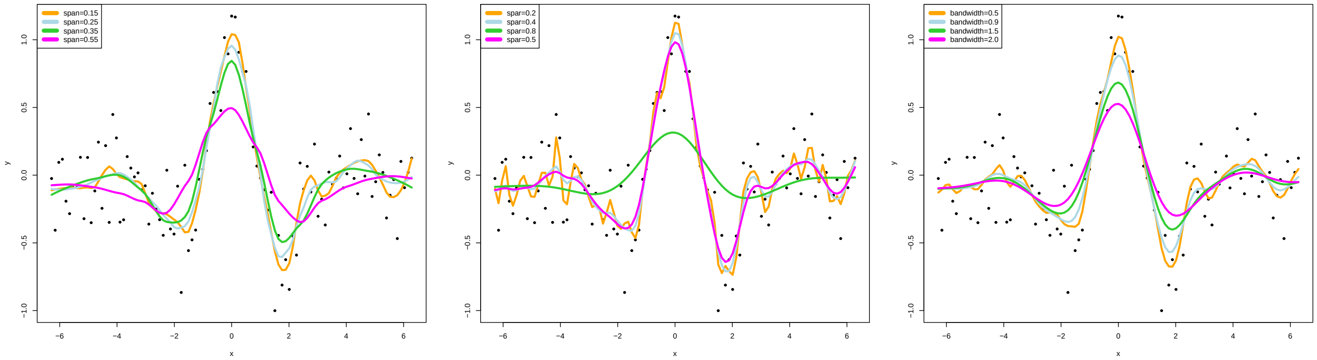

Figure 9: Smoothing methods with different parameter values

For NW kernel or splines smoothing, if h/λ (bandwidth/spar) is small, then the curve will be wiggly, because the estimate will depend heavily on points closest to x_0 . In this case, the model is trying to fit to a small neighborhood, thus we over-fit. Larger values for h/λ means that points further away will have similar influence as points that are close to x_0. For Loess, a large span (α) increases the smoothness but decreases the resolution of the smoothed data set, while a small span decreases the smoothness but increases the resolution of the smoothed data set.

2. Tuning the smoothing parameters (equidistant)

For each smoothing parameter, we compare the actual observed Y_i with its smoothed estimate for a given h, α, or λ based on the $n-1$ data points without using the i-th observation. We choose the optimal tuning parameters by minimizing the average mean square error based (MSE) of the leave-one-out cross-validation.

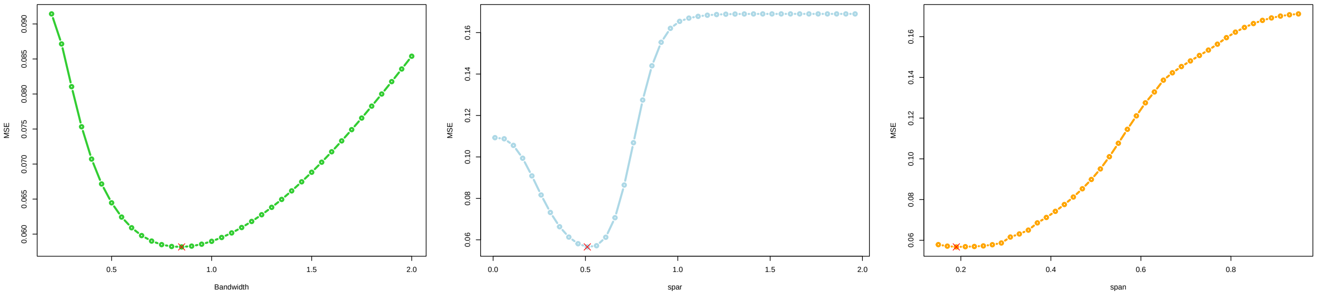

Figure 10: Leave-One-Out CV results

After Leave-One-Out cross validation, we get the following values for each smoothing parameter:

- The best bandwidth value that achieved the lowest MSE value is $h = 0.85$.
- The best spar value that achieved the lowest MSE value is $\lambda = 0.51$.
- The best span value that achieved the lowest MSE value is $\alpha = 0.19$.

3. Tuning the smoothing parameters (non-equidistant points)

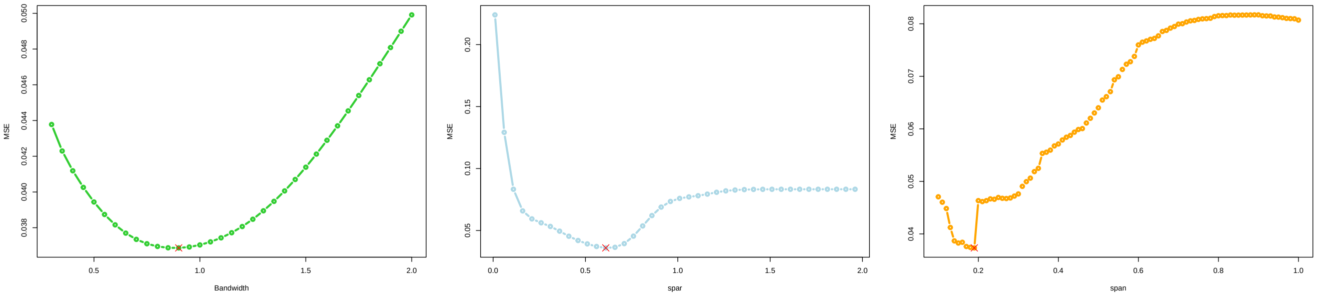

Figure 11: Leave-One-Out CV results

The Leave-One-Out cross validation using non-equidistant points yields the following results:

- The best bandwidth value that achieved the lowest MSE value is $h = 0.9$
- The best spar value that achieved the lowest MSE value is $\lambda = 0.61$
- The best span value that achieved the lowest MSE value is $\alpha = 0.19$

Part 3: Using newly tuned smoothing parameters

1. Equidistant design

After tuning the smoothing parameters, we plot the mean of each estimator with the raw observations for the equidistant design

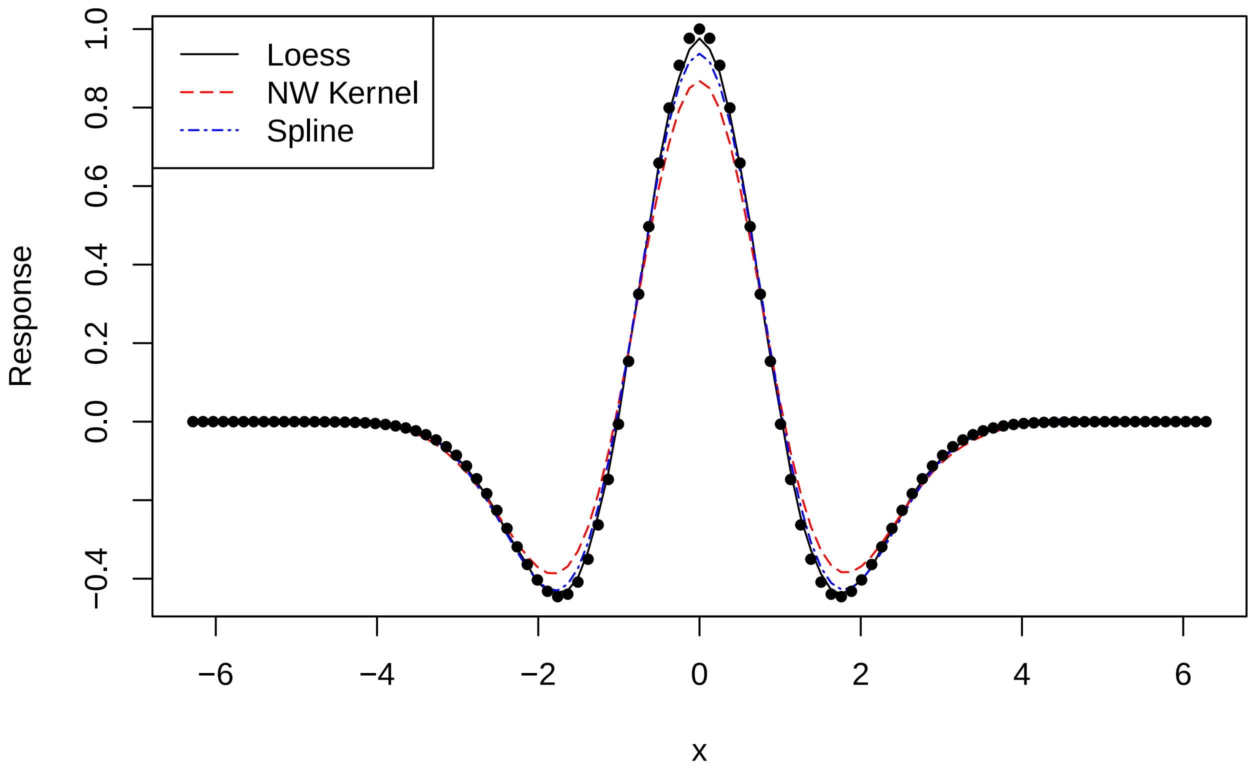

Figure 12: The mean of the three local smoothing estimators with raw observations

Looking at the mean of each estimator after tuning, we can clearly see that all three methods performed similarly and provide a almost perfect fit for $f(x_i)$

Next, we plot the new bias and variances for all three estimates

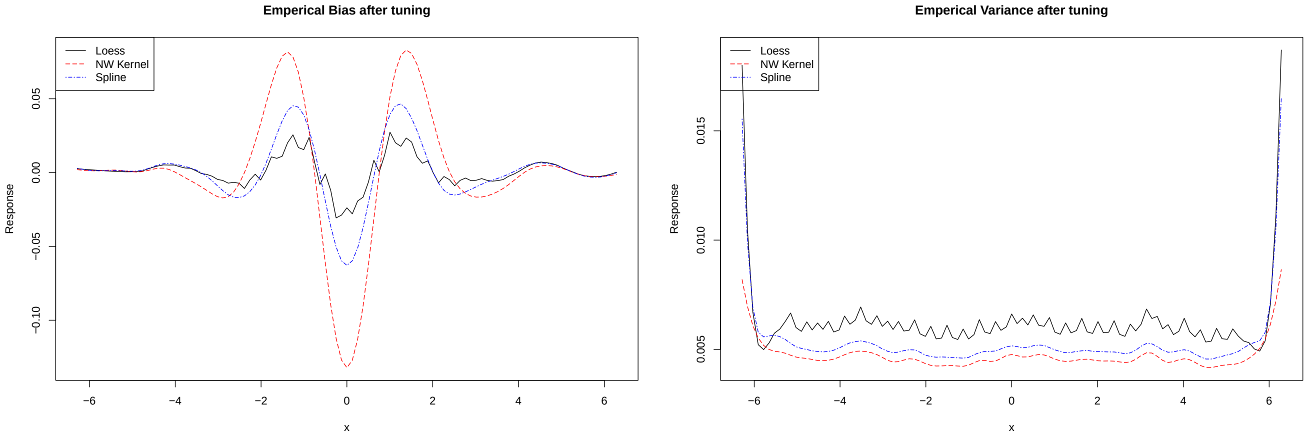

Figure 13: The empirical bias and variance of the three local smoothing estimators after tuning

In terms of bias, we see that all of the estimators have high bias at $x_i = +/-\frac{\pi}{2}$ and a negative bias at $x_i = 0$. NW kernel has the highest bias on average. In terms of variance, we can clearly see that NW kernel and Splines smoothing variances are much lower after tuning of their respective smoothing parameters. The variances show poor behavior at the boundaries. This happens because the kernel window at the boundaries has missing data. In other words, we have weights from the kernel, but no data to associate with them. Overall, we achieve a low variance after tuning.

Now, let's take a look at the two MSE plots.

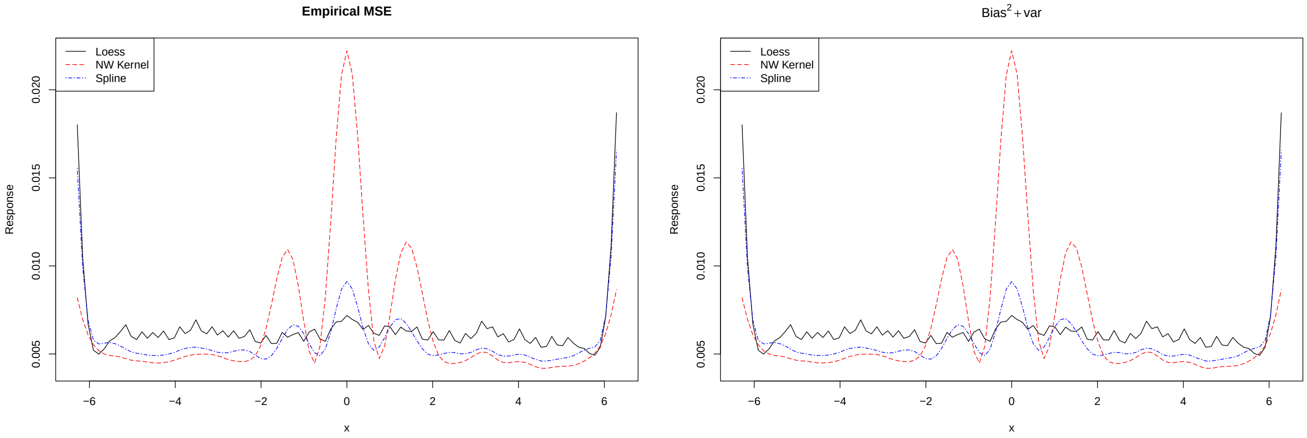

Figure 14: The MSE of the three local smoothing estimators after tuning

In terms of MSE, NW kernel has the highest MSE that peaks at $x_i = 0$ and $x_i = +/-\frac{\pi}{2}$. Loess and splines have on average similar MSE values across the board but exhibit poor behavior at the boundaries of x_i

2. Non-equidistant design

After tuning the smoothing parameters, we plot the mean of each estimator with the raw observations for the non-equidistant design

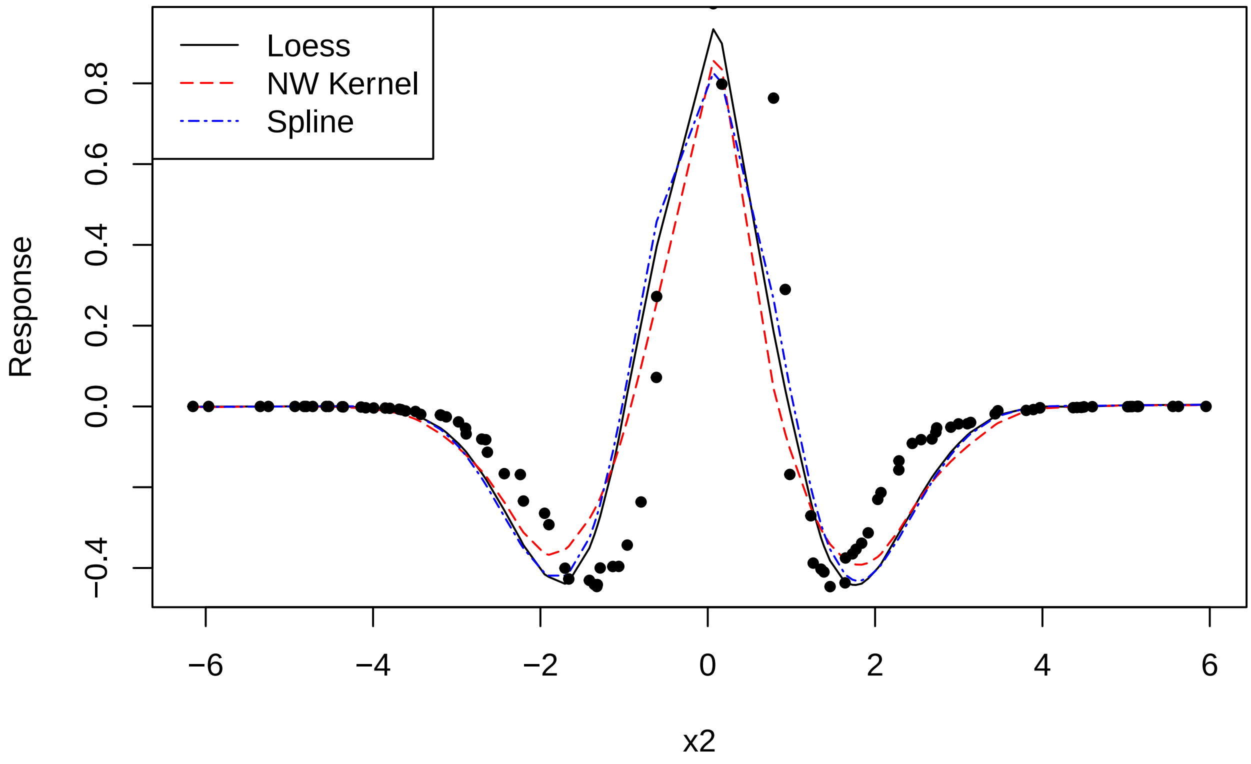

Figure 15: The mean of the three local smoothing estimators with raw observations

Again, we see that each estimator is able to fit the function with a very good accuracy. All estimators performed similarly.

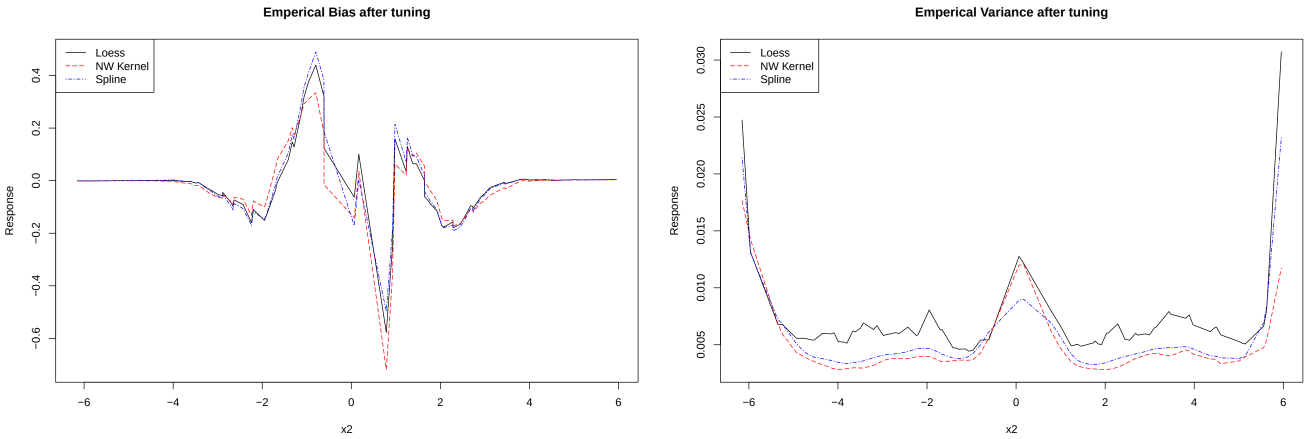

Figure 16: The empirical bias and variance of the three local smoothing estimators

Here again, the three estimators have high/negative bias at $x_i = 0$ and high bias at $x_i = +/-\frac{\pi}{2}$. The Bias is zero from $[-\infty, -\pi]$ and $[\pi, +\infty]$. In terms of variance, we can clearly see that all kernels performed equally likely. We also see that all three estimators exhibit very high variances at $x_i = 0$.

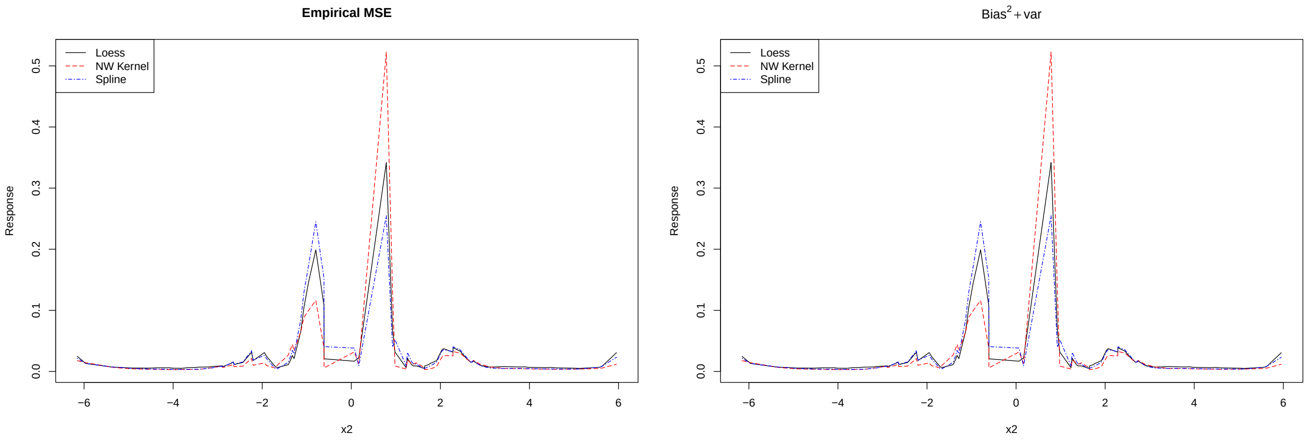

Figure 17: The MSE of the three kinds of local smoothing estimators

In terms of MSE, all kernels exhibit very high MSE values in the interval of $[-\frac{\pi}{2}, \frac{\pi}{2}]$, in particular at $x_i = -\frac{\pi}{2}$ and $x_i = \frac{\pi}{2}$. NW kernel has the highest MSE at $x_i = \frac{\pi}{2}$ and the lowest at $x_i = -\frac{\pi}{2}$

Conclusion and Findings

In this assignment, we try to understand the statistical properties and computational challenges of three different types of local smoothing methods: **LOESS**, **Nadaraya-Watson**, and **Spline** smoothing. To better evaluate the performance of each smoothing methods, we calculated their empirical bias, empirical variance and, the empirical mean square error.

In the initial equidistant design before tuning, Loess smoothing has low variance, a high bias at $x_i = -\frac{\pi}{2}/x_i = \frac{\pi}{2}$,and negative bias at $x_i = 0$. consequently, this leads to a high MSE at the three aforementioned points.

On the other hand, NW kernel has the lowest Bias, but the highest variance. Splines smoothing has a low bias/somewhat low variance, and low MSE. All three smoothing methods exhibit a poor behavior (high variance) at the boundaries of X.

In the non-equidistant design where the data points are not spaced at uniform distances, all estimators yield a nonlinear output. NW kernel shows a low bias/high variance and low MSE. While, Splines and Loess show low variance and a negative bias at $x_i = 0 \rightarrow$ high MSE at $x_i = 0$

After tuning the smoothing parameters of each estimator, we can see a lot of improvement. All estimators somewhat have comparable variance/bias in both equidistant and non-equidistant designs. All models show a balance between bias and variance with similar poor behavior on $x_i = -\frac{\pi}{2}$, $x_i = \frac{\pi}{2}$, and $x_i = 0$ especially in terms of bias and total error.

All these experiments have shed some light on the bias-variance tradeoff. In modeling, we seek an optimal balance between model complexity and the total error of a model (mse). Since $\textbf{mse} = \textbf{bias}^2 + \textbf{variance}$, the total error increases as bias or variance or both increases. This usually yields to more complex model, over-fitting or under-fitting. We saw firsthand the effect of tuning the smoothing parameters of each estimator and how that leads to a balance between variance and bias. It's worth mentioning that methods that have low variance tend to be less complex (Loess) and methods that have low bias tend to be more complex (NW kernel). Having a balance in model complexity and the model's statistical properties (bias, variance) can be challenging but it is crucial. Additionally, tuning of the smoothing parameters poses computational challenges because leave-one-out cross validation can be heavy. In practice, it is often recommended to use the so-called generalized cross-validation.

In summary, given the underlying structure and complexity of each of the three smoothing methods, I personally do not think it was a fair comparison. However, the analysis helped me gain a solid understanding of the bias-variance tradeoff and how important are cross validation and hyper-parameter tuning.