Challenge B

Rossi Abi-Rafeh

11/15/2017

Task 1B - Predicting house prices in Ames, Iowa (continued)

In this part of Task 1, you will use a common machine learning algorithm to predict the house prices in Ames, Iowa. Non-parametric methods that you study in Intermediate Econometrics are OK and you can choose one of them (implementing a method is a good way to understand how it works…). You will not be graded on the complexity of the method, but rather on whether you can explain nicely what it does, and make a simple implementation of it.

Step 1 - Choose a ML technique : non-parametric kernel estimation, random forests, etc… Give a brief intuition of how it works. (1 points)

Step 2 - Train the chosen technique on the training data. Hint : packages `np` for non-parametric regressions, `randomForest` for random forests. Don't use the variable `Id` as a feature. (2 points)

Step 3 - Make predictions on the test data, and compare them to the predictions of a linear regression of your choice. (2 points)

Task 2B - Overfitting in Machine Learning (continued) - 1 point for each step

Step 1 - Estimate a low-flexibility local linear model on the training data. For that, you can use function `npreg` the package `np`. Choose `ll` for the method (local linear), and a bandwidth of 0.5; Call this model `ll.fit.lowflex`

Step 2 - Estimate a high-flexibility local linear model on the training data. For that, you can use function `npreg` the package `np`. Choose `ll` for the method (local linear), and a bandwidth of 0.01; Call this model `ll.fit.highflex`

Step 3 - Plot the scatterplot of x-y, along with the predictions of `ll.fit.lowflex` and `ll.fit.highflex`, on only the training data. See Figure 1.

Step 4 - Between the two models, which predictions are more variable? Which predictions have the least bias?

Step 5 - Plot the scatterplot of x-y, along with the predictions of `ll.fit.lowflex` and `ll.fit.highflex` now using the test data. Which predictions are more variable? What happened to the bias of the least biased model?

Now let's see what happens to the overall error rate, that is the mean square error. Remember the mean squared error MSE : $MSE^{model} = \frac{1}{n}\sum_i(\hat{y}_i^{model} - y_i)^2$.

Step 6 - Create a vector of bandwidth going from 0.01 to 0.5 with a step of 0.001

Step 7 - Estimate a local linear model y ~ x on the training data with each bandwidth.

Step 8 - Compute for each bandwidth the MSE on the training data.

Step 9 - Compute for each bandwidth the MSE on the test data.

Step 10 - Draw on the same plot how the MSE on training data, and test data, change when the bandwidth increases. Conclude.

Task 3B - Privacy regulation compliance in France

The CNIL (Commission Nationale de l'Informatique et des Libertés) is the French government body that regulates digital freedom, as well as user data protection and privacy. Each company or organization in France wishing to adopt the regulatory framework of the CNIL has to nominate a CIL : a delegate that will ensure that the internal use of user data is consistent with the CNIL's recommendations. This procedure is not mandatory in general, but becomes so if the company wishes to deal with sensitive data (like healthcare.)

The list of companies and organizations that nominated a CIL is publicly available. However we do not know anything about the size or the sector of these companies.

The SIREN dataset, made public by the french government, compiles a list of all french companies or organizations, as well as details about their employment size, and sector. This dataset is too big to manage on personal computers, I made a smaller version for you to use. It is available on moodle. The original data is available here

Step 1 - Merge the two datasets by SIREN number. (2 points)

Step 2 - Plot the histogram of the size of the companies that nominated a CIL. Comment. (2 points)

Step 3 - Provide a vizualization that distribution of the sectors of activities of companies that nominated a CIL. Comment. (1 points)

Institutions with CIL regulation

Figures for task 2B

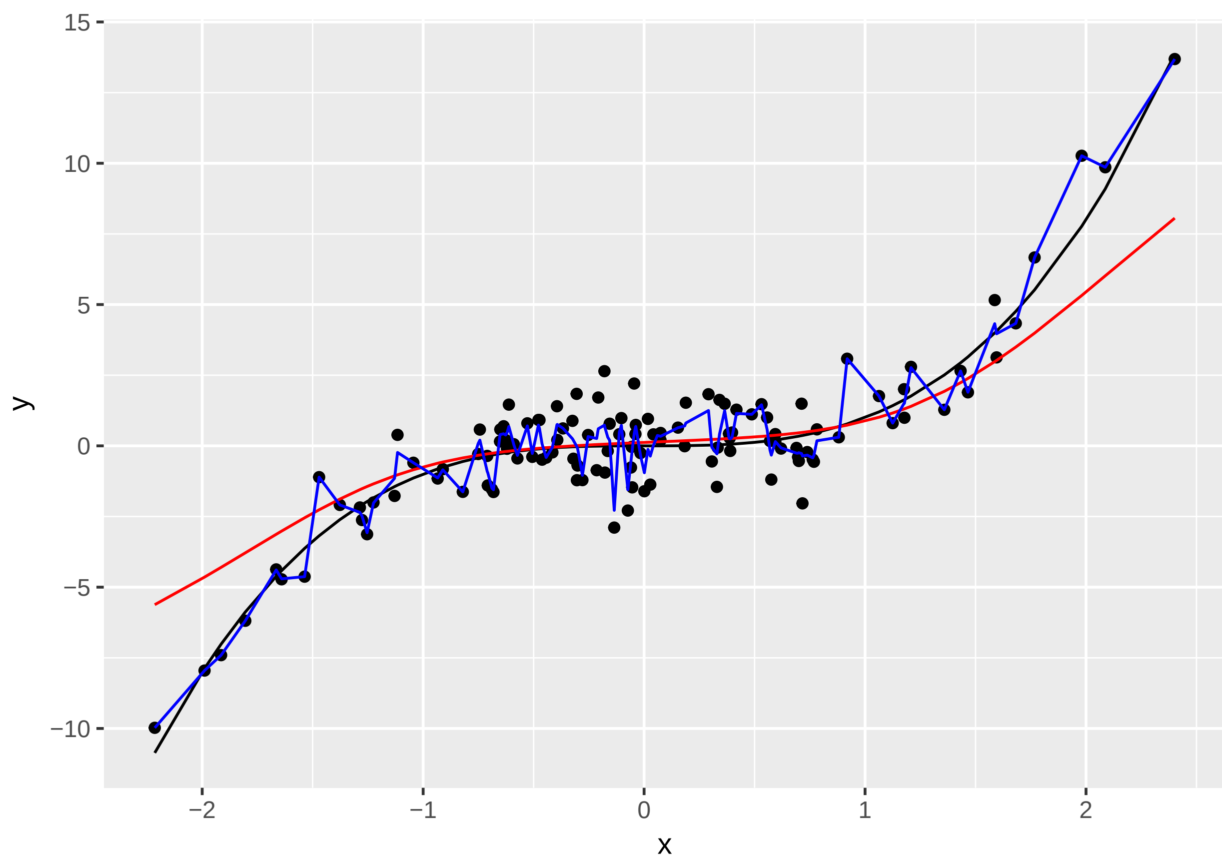

Figure 1: Step 3 - Predictions of ll.fit.lowflex and ll.fit.highflex on training data.

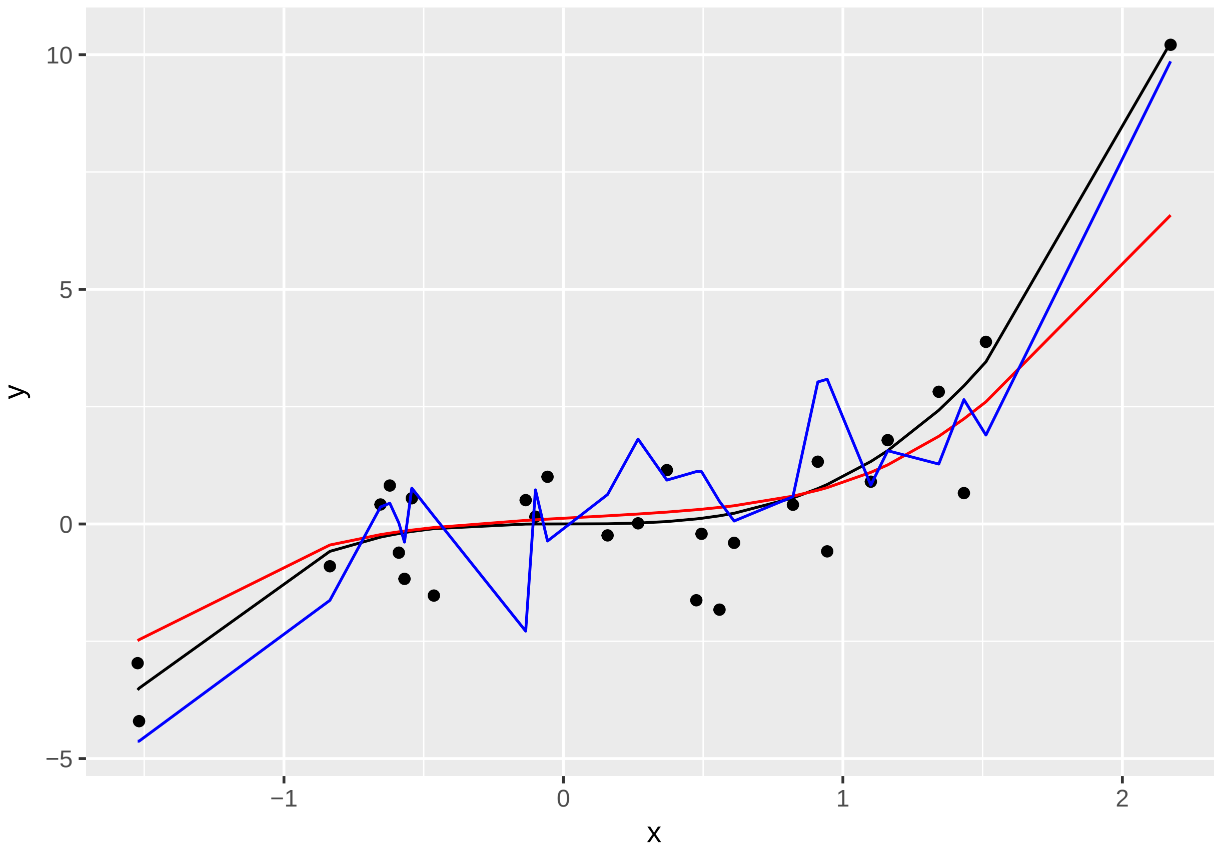

Figure 2: Step 5 - Predictions of ll.fit.lowflex and ll.fit.highflex on test data.

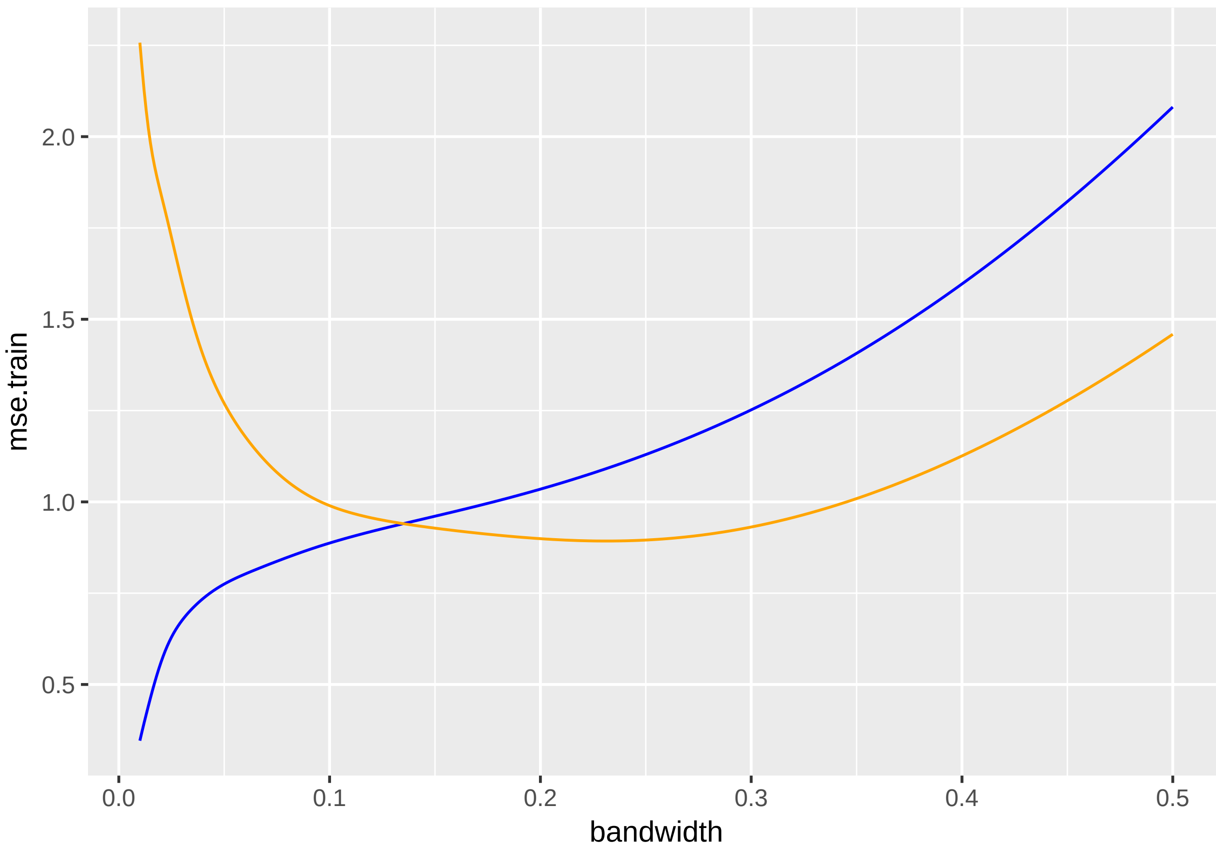

Figure 3: Step 10 - MSE on training and test data for different bandwidth - local linear regression