

Assignment 1

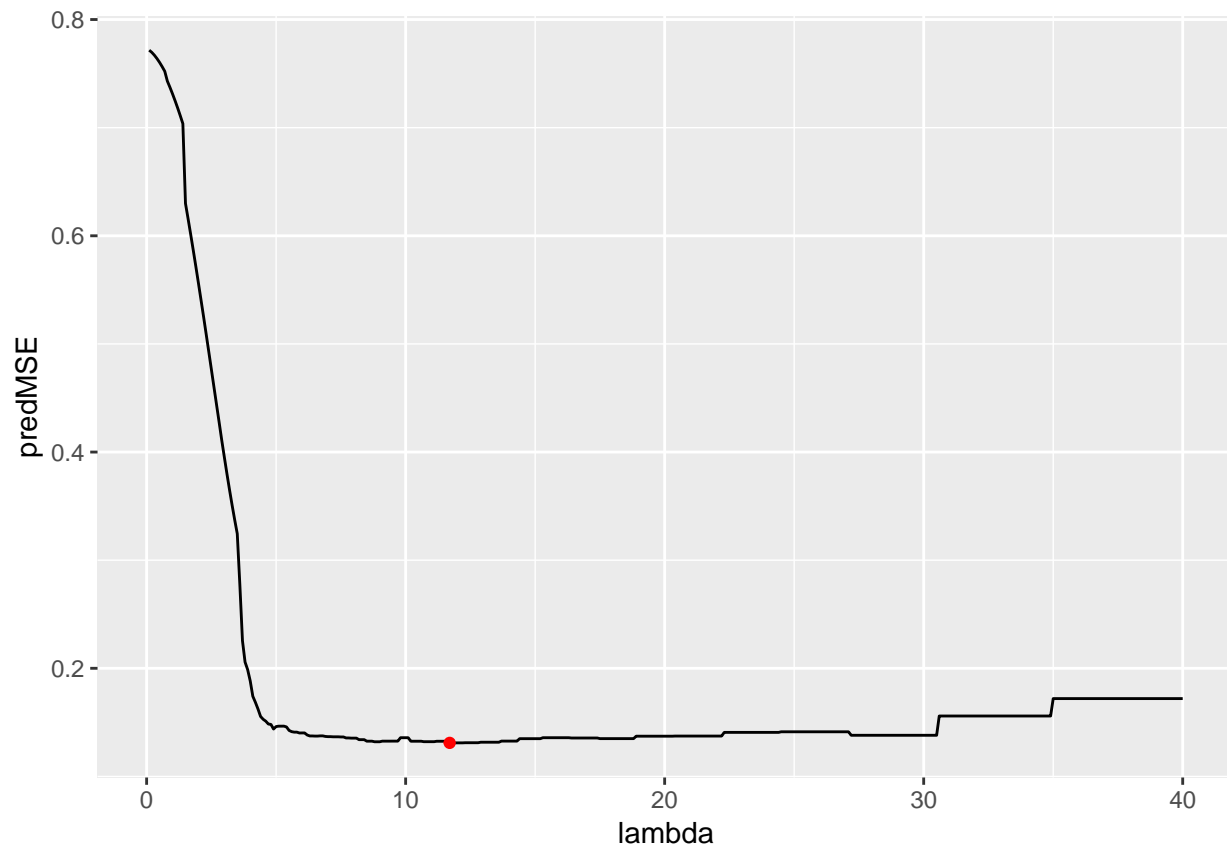
Reading Data

	Day	Rate	LMR
1	1	0.0014	-6.571283
2	2	0.0040	-5.521461
3	3	0.0051	-5.278515
4	4	0.0064	-5.051457
5	5	0.0075	-4.892852
6	6	0.0098	-4.625373

Reading Data into variable `mortRate` and adding `log` of the rate column to the data frame.

Created `My_MSE` function

Using the Function Created for different values of `lambda`.

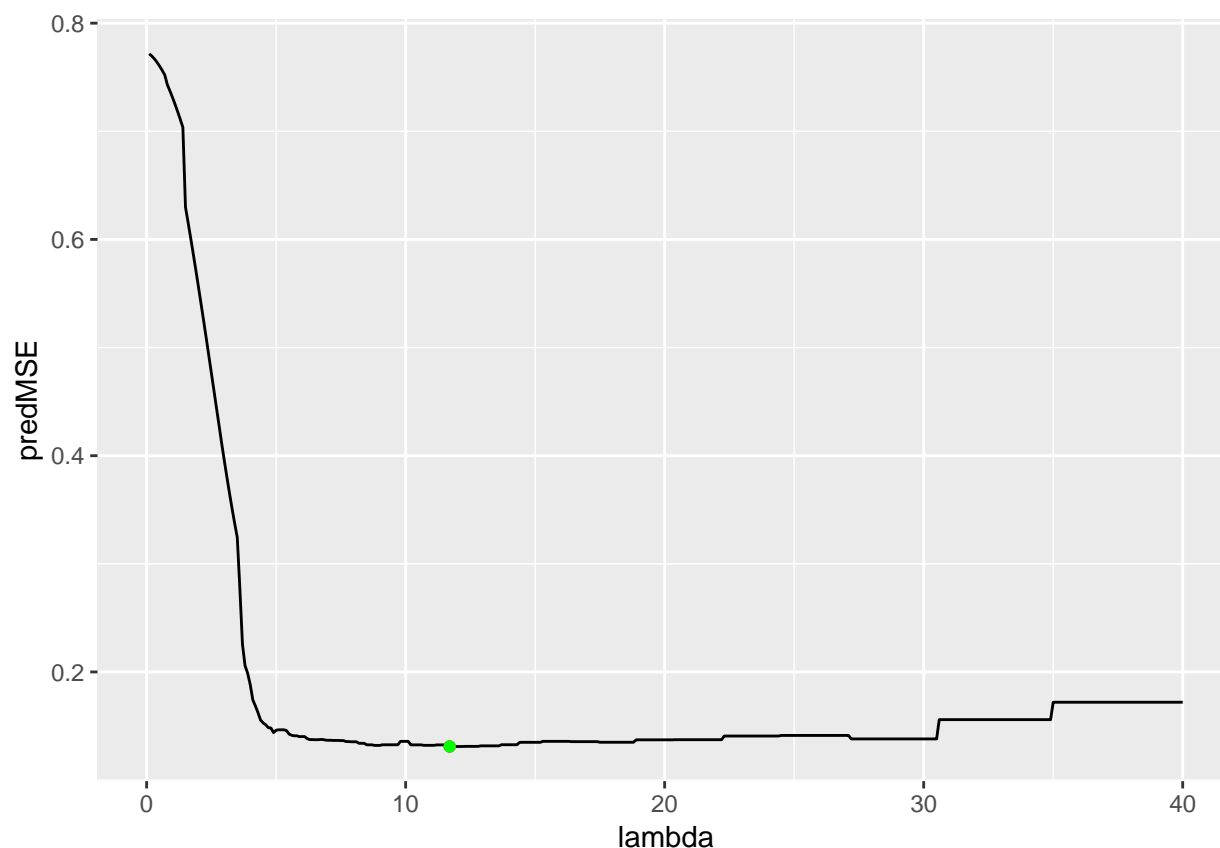


The minimum value of MSE found by the iterative method is : 0.131046964831872

The corresponding value of lambda at which this minimum was found is : 11.7

The minimum found by the iterative method is shown on the plot with a red point.

Using the My_MSE to find minimum iteratively with a threshold



The minimum value of MSE found by the iterative method is : 0.131046964831872

The corresponding value of lambda at which this minimum was found is : 11.7

The method had tolerance value of 0 and it took 245 steps to converge.

We use the counts to keep count of the number of times the function has been called. The function MyMSE was called 245 times until convergence, using the tolerance value of (0.01). The loop stops as soon as we get an increment in the MSE value greater than the tolerance value. The step size we are taking is very small 0.1, this is the reason it took 245 steps to converge.

The min value of MSE and the corresponding lambda found in this way is exactly the same to the actual minimum. This is the reason the points overlap in the plot.

Using Optimize with threshold on Accuracy : 0.01

Number of function calls taken for the optimize function to find minimum : 18

The minimum found by the optimize function :

MIN MSE : 0.132144141920378

LAMBDA : 10.6936106512018

The function takes fewer steps to converge when we introduce a threshold value on accuracy. With the threshold(0.01) in the iterative method it took 245 steps and with the threshold(0.01) using optimize it took just 18 steps to converge. I think the function optimize takes a larger step that lets it get to the min value faster, it is not considering all the intervals we provided it with. This is the reason the min value found by this function is not the global minimum, it is close to it, but not the exact minimum.

The minimum found in the previous case was the actual minimum and in this case the algorithm was forced to declare convergence as the threshold value of accuracy was satisfied.

Using `optim()` function and BFGS to find lambda value.

Number of function calls taken for the optimize function to find minimum : 3

The minimum found by the optimize function :

MIN MSE : 0.171999614458471

LAMBDA : 35

Number of Iterations : 3

The initialization point we set for the algorithm was 35, there is a steep downhill at that point, so the algorithm takes that step, but then the function is plateaued towards the lower bound from that point and has a peak as lambda value increase from that point. This is the reason the algorithm declares convergence at the local optima, and takes just three steps to declare convergence.

This function depends heavily on the initial point we specify as the start. If our initialization is bad the function will get stuck at a local optima. The previous optimize function took in a list of values to search over and find the minimum, this is the reason the convergence found by the previous function was better.