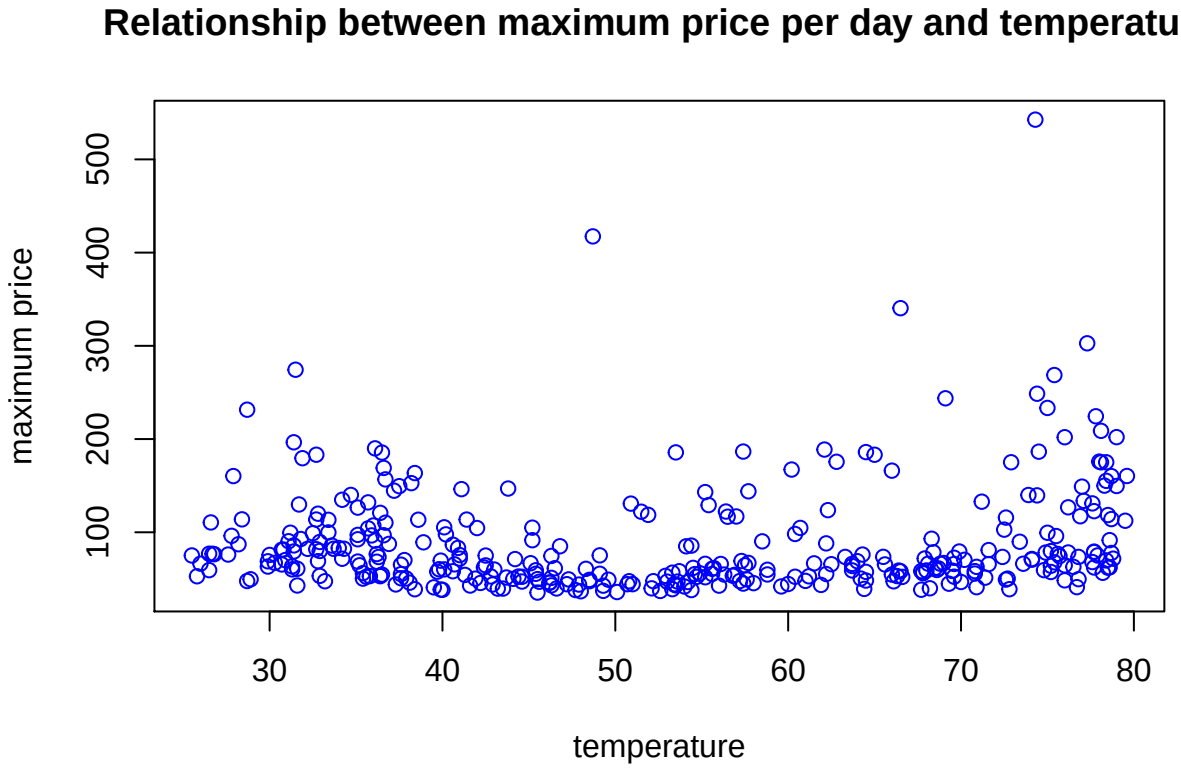
PS 2

Aastha

2/13/2020

1.1 Relationship between Maximum price per day and the temperature



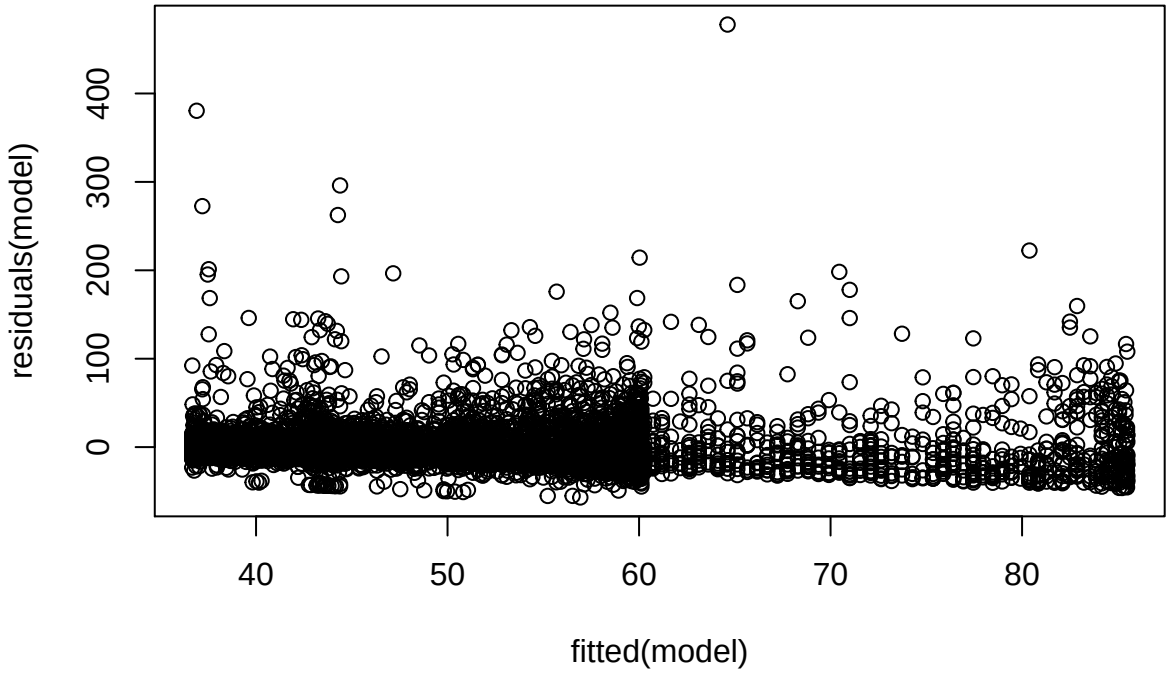
No, this relationship does not look linear. We might need to check for nonlinear relationships.

##1.2 Predicting the relationship using a polynomical regression

```
##
## Call:
## lm(formula = dt$lmp ~ poly(dt$temp, i))
##
## Coefficients:
##        (Intercept)   poly(dt$temp, i)1   poly(dt$temp, i)2   poly(dt$temp, i)3
##            49.2186            116.4241            887.9329            335.1381
##  poly(dt$temp, i)4   poly(dt$temp, i)5   poly(dt$temp, i)6   poly(dt$temp, i)7
##           -45.1422            293.8033            -65.5527           -152.6110
##  poly(dt$temp, i)8   poly(dt$temp, i)9  poly(dt$temp, i)10
##            22.1290            -42.2822             -0.4861
```

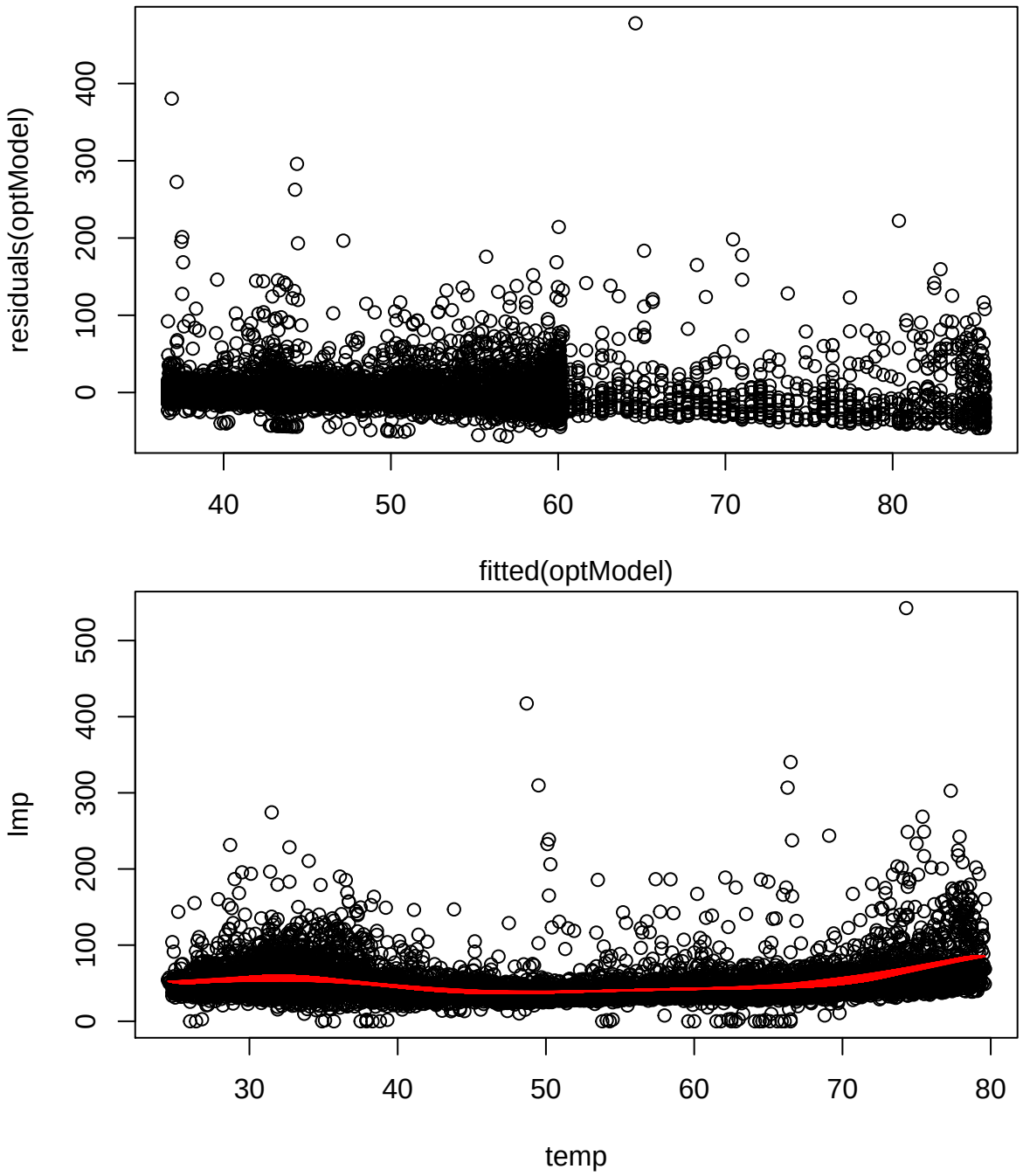
```
##                         2.5 %       97.5 %
```

```
## (Intercept)          48.72824   49.7088966
## poly(dt$temp, i)1    70.72631  162.1219087
## poly(dt$temp, i)2   842.23509  933.6306811
## poly(dt$temp, i)3   289.44033  380.8359253
## poly(dt$temp, i)4   -90.84000    0.5555923
## poly(dt$temp, i)5   248.10550  339.5010927
## poly(dt$temp, i)6  -111.25045  -19.8548543
## poly(dt$temp, i)7  -198.30883 -106.9132304
## poly(dt$temp, i)8   -23.56880   67.8267920
## poly(dt$temp, i)9   -87.97997    3.4156273
## poly(dt$temp, i)10  -46.18386   45.2117374
```



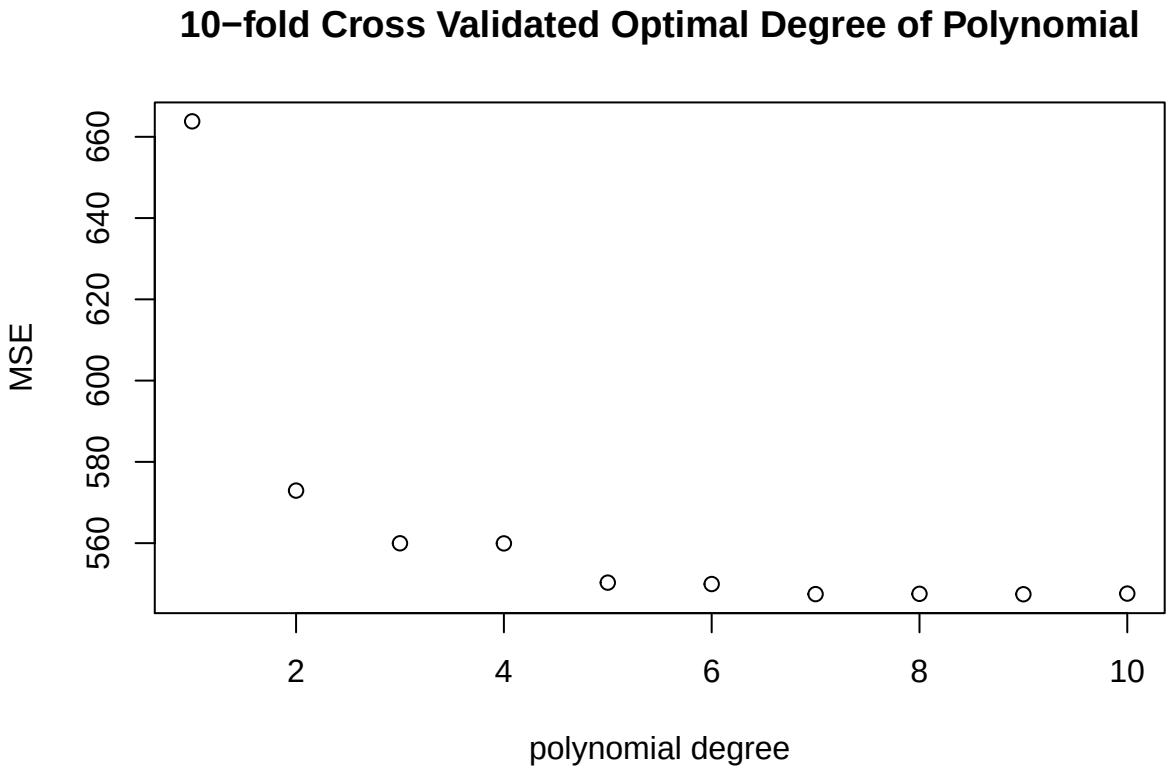

```
## [1] 543.467
```

```
## [1] 543.4044
```



The best fit is polynomial of degree = 09. However, the value for MSE is similar for polynomial of degree = 07 as well.

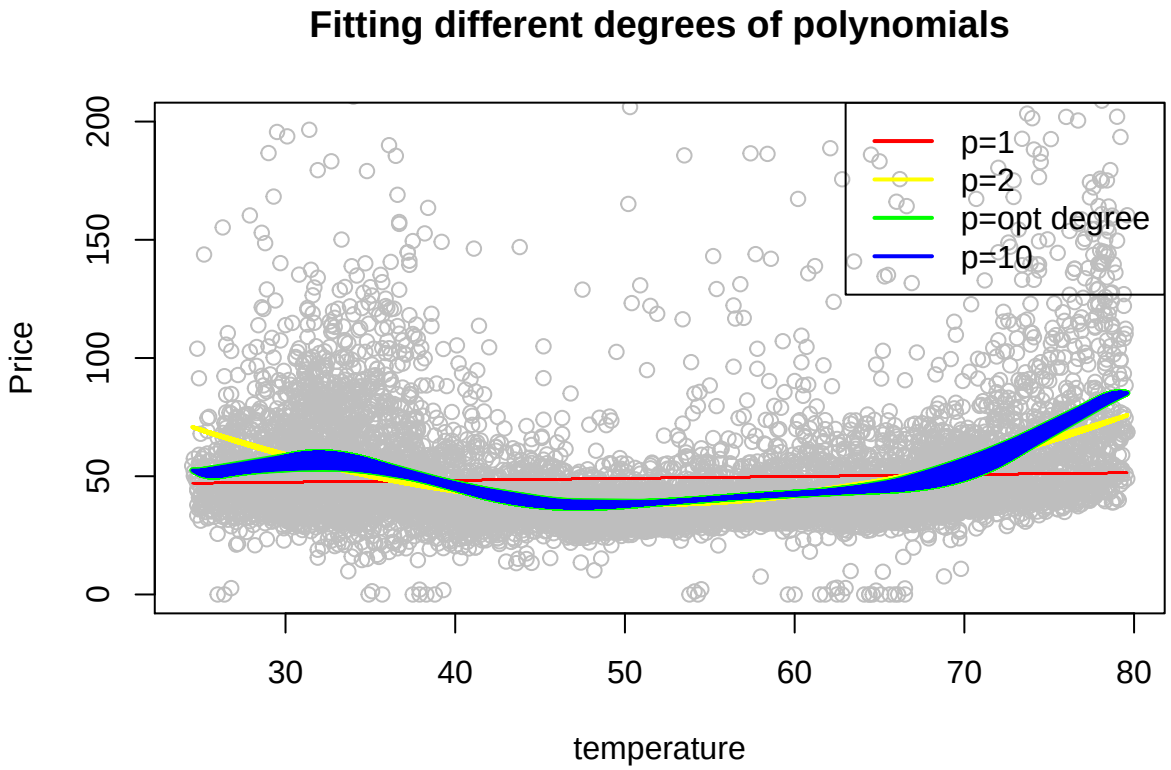
##1.3 Using K-Fold Cross Validation to check for optimal degree of Polynomial




```
## [1] 9
```

I obtain the optimal degree of polynomial to be 7, and predict the values of the variable lmp using the same.

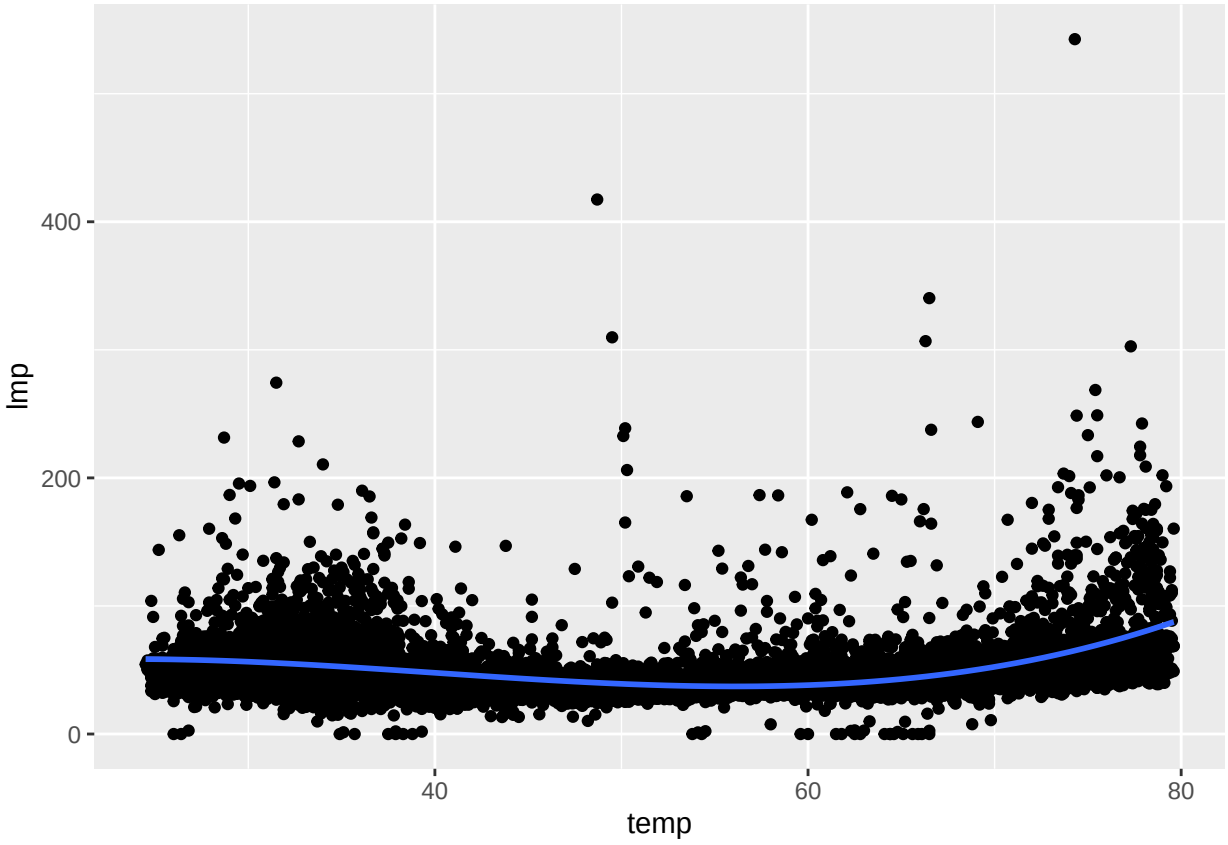
##1.4 Plotting different degrees of polynomial regression along with the optimal degree



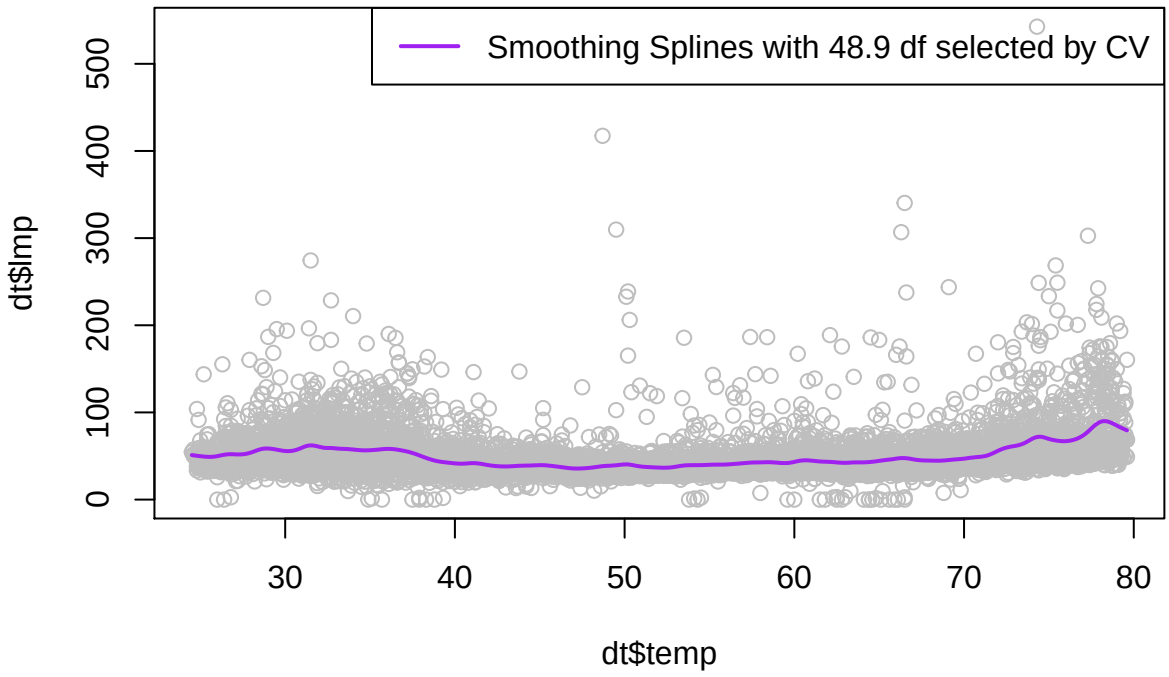
It is clear that the optimal polynomial degree regression(blue) fits the data musch better than the other degree of

polynomial regressions.

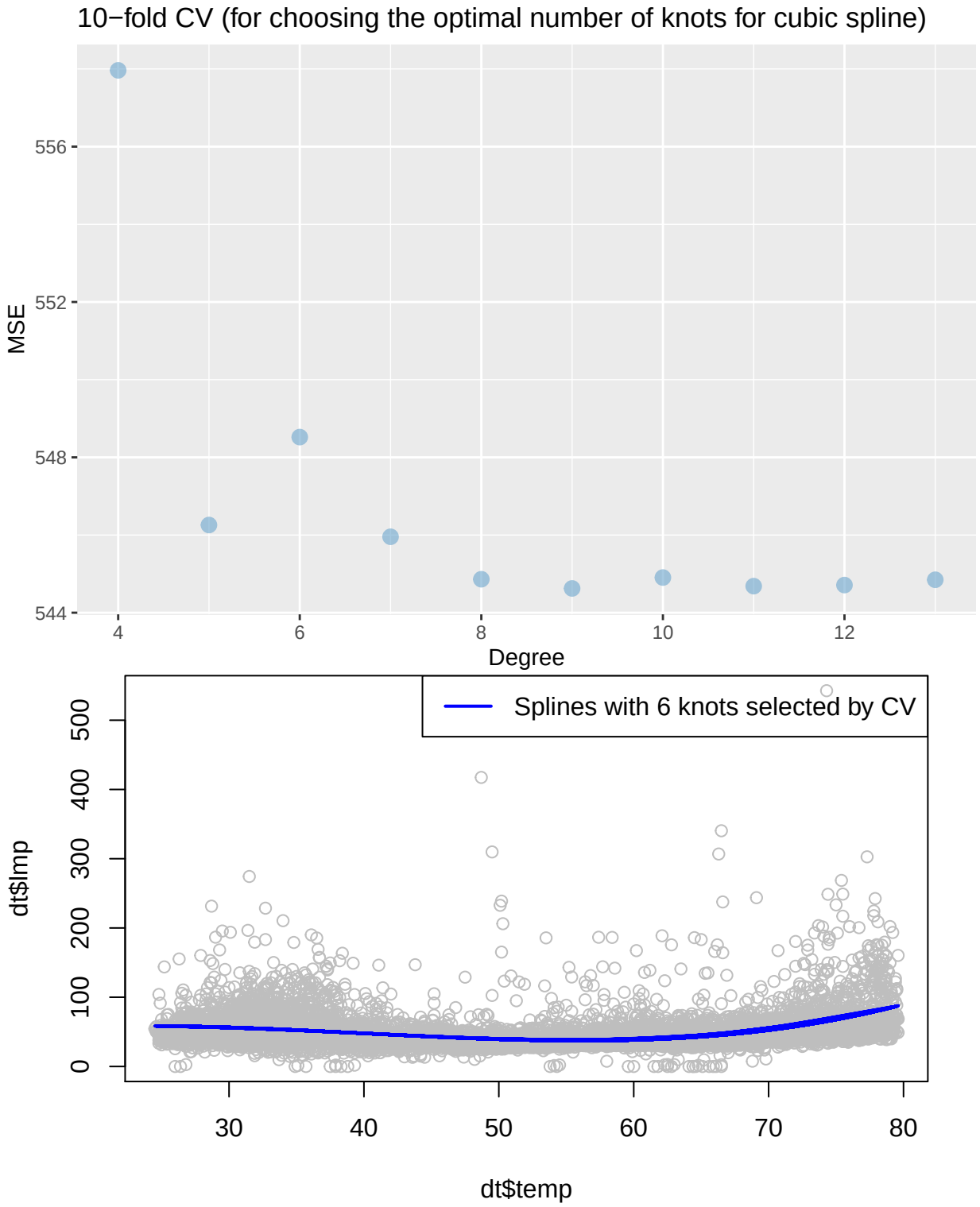
##1.5 Prediction of lmp using cubic splines




```
## Call:
## smooth.spline(x = dt$temp, y = dt$lmp, cv = TRUE)
##
## Smoothing Parameter  spar= 0.400068  lambda= 2.279919e-05 (12 iterations)
## Equivalent Degrees of Freedom (Df): 48.94408
## Penalized Criterion (RSS): 237840.9
## PRESS(l.o.o. CV): 541.9
```

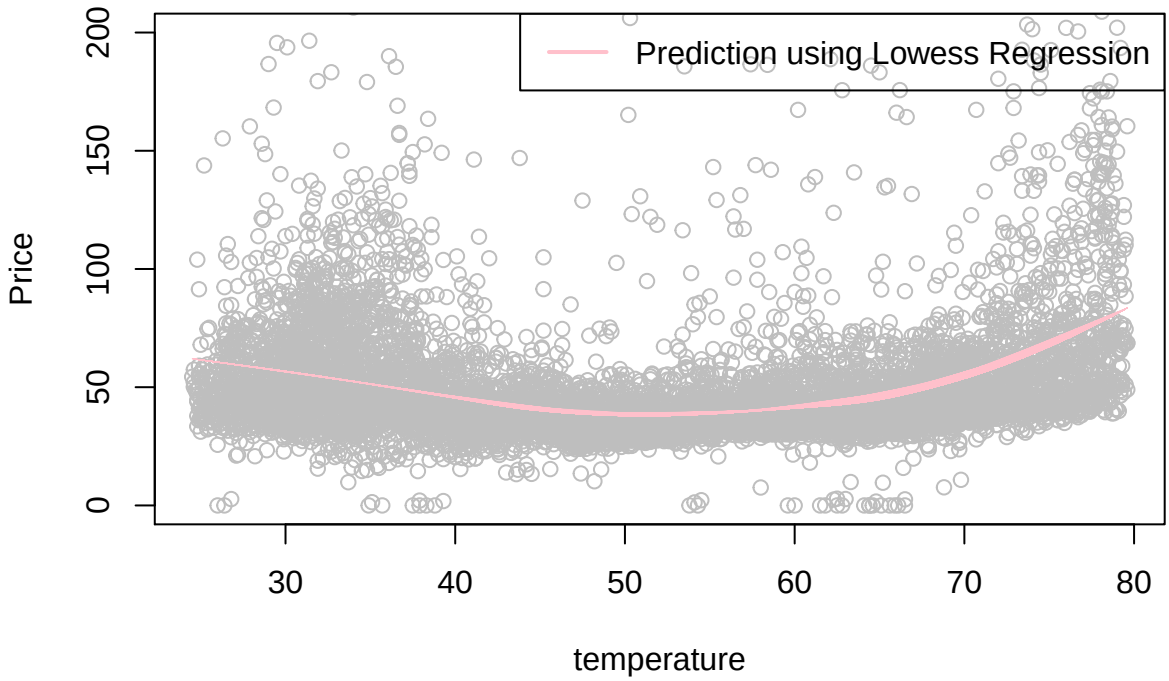

Smoothing Splines with 48.9 df selected by CV
dt$lmp
0
100
200
300
400
500
30
40
50
60
70
80
dt$temp

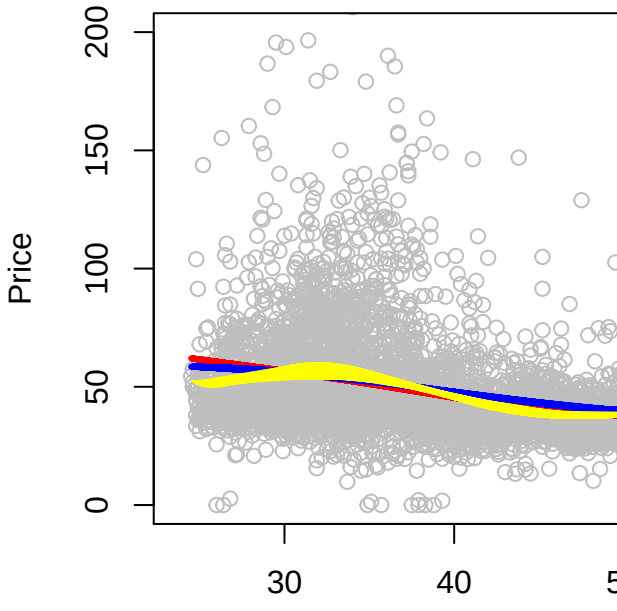
manual cross validation to find the optimal number of knots in cubic splines:



I find that the optimal degree of knots is 6 for the natural splines to fit the data well.

1.6 Prediction using Loess regression

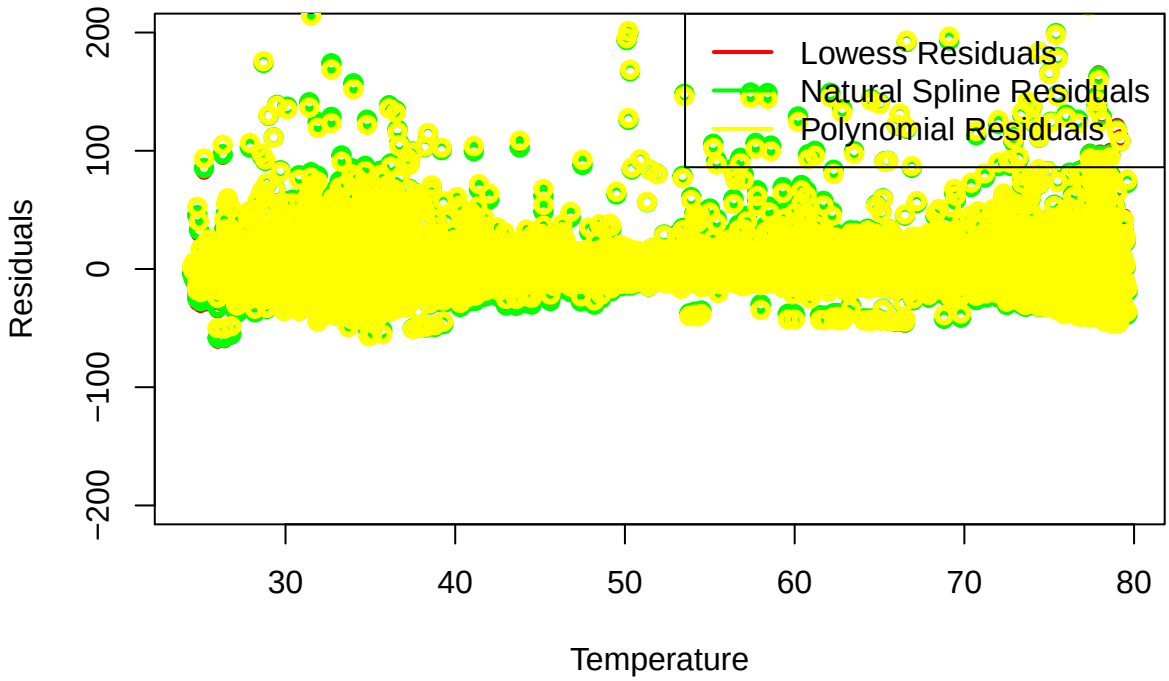




Fitting Optimal-Polynomial, 'Optimal-Spline and Lowess Predictions

Polynomial regression with optimal degree (09) seems to perform the best among all the regressions here.

1.7 Plotting residuals of each of the curves against temperature



All three seem to perform similar in the interior region. Note that, lowess seems to perform bad at the boundaries, however, note that natural cubic splines perform better on the boundary since they put additional constraints and therefore reduce variance around the boundary.