[STAT W4702] Statistical Inference & Modelling Group Project

Babies

12 December 2015

Abstract

Data Set

This project was conducted on the Low Birth Weight dataset collected in 1986 at Baystate Medical Center, Springfield, Massachusetts as a part of a bigger study on the factors influencing newborn infants' health and risk of serious health problems potentially leading to death. This dataset is distributed as a part of `MASS` library and contains **189 observations** and **10 variables**, among which `bwt` represents the exact amount of newborn infant's weight in grams and is used as the variable of interest we are trying to predict. The other 9 variables stand for different factors related to mothers' physiological parameters, such as age, weight and race, their health-related habits and behavior during pregnancy (smoking habits, presence of uterine irritability and number of physician visits). Also there is a low birth weight indicator `low`, which is defined as a binary variable showing whether the weight of an infant is below 2500 grams or not. Brief description of each variable is provided in the table below.

The goal of our research is to identify relationship between these variables and infant weight and understand the influence of each of them on the explained variable. The project pursue both inferential and predictive goals as it is equally important to be able to obtain inference about factors affecting newborn's health and to be able to react on the potential health risks in a timely manner, when the model predicts the low birth weight outcome for a certain observation. In order to accomplish this goal we tried to fit multiple linear and non-linear models exploring the rationale that could provide the evidence for certain types of models and finding balance between interpretability and predictive power of the model.

Cleaning and Exploring Dataset

For the purposes of the research the dataset was cleaned in the following way:

- birth weight variable `bwt` is converted from grams to kilgrams to reduce the order of magnitude for estimated model coefficients and error values;
- factor variable `race` was assigned with proper labels `white`, `black` and `other`;
- physisian visits were converted to a factor variable `ftv` with 3 labels `0`, `1` and `2+`;
- response is defined as an exact amount of infant's weight from `bwt`;
- all the columns are assigned with meaningful names.

Variable description table and summary statistics of the tidy dataset are provided below.

Variable	Description
`baby.grams`	weight of newborn infant in kg
`mother.age`	mother's age in years
`mother.weight`	mother's weight in pounds at last menstrual period
`race`	mother's race, factor variable with following labels: *white*, *black* or *other*
`smoke`	smoking status during pregnancy, binary variable

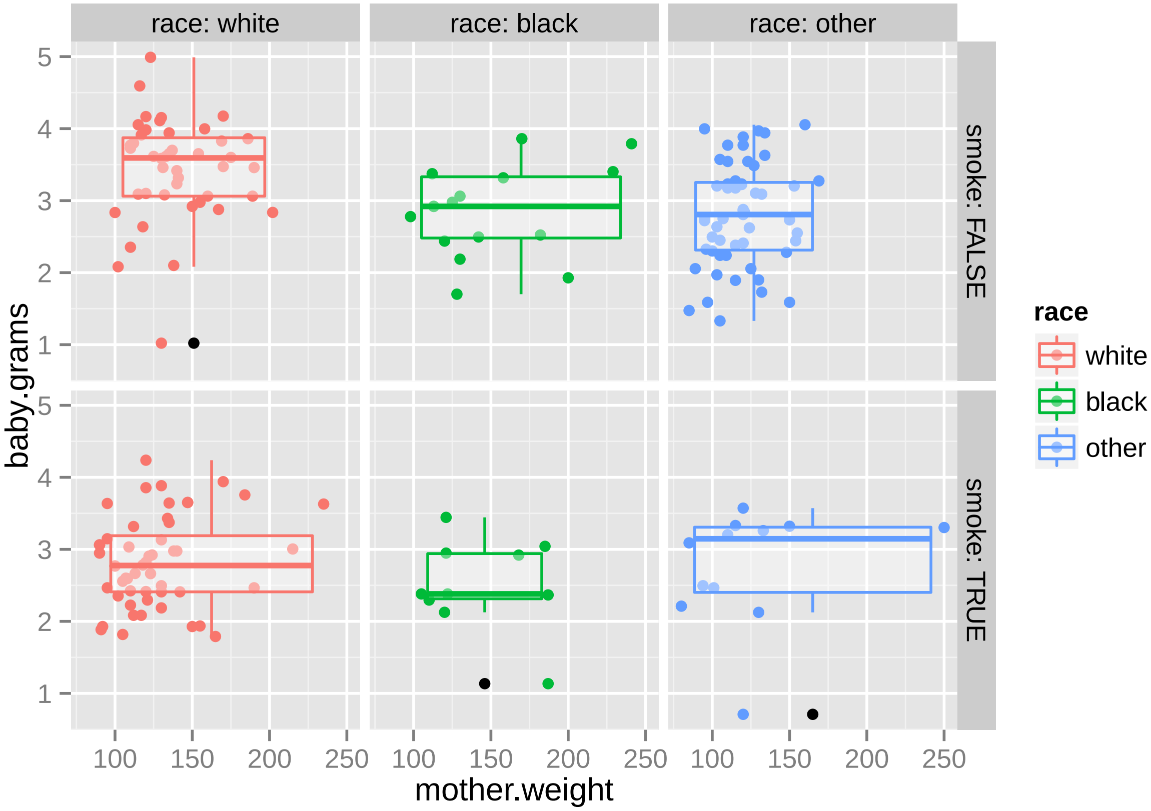
Variable	Description
`prem.labor`	binary variable showing whether mother had premature labors before or not
`hypertension`	binary variable showing whether mother had hypertension or not
`uterine`	binary variable showing presence of uterine irritability
`physician.visits`	number of physician visits during the first trimester: *0*, *1* or *2+*

```
##    baby.grams      mother.age    mother.weight      race
##  Min.   :0.709   Min.   :14.00   Min.   : 80.0   white:96
##  1st Qu.:2.414   1st Qu.:19.00   1st Qu.:110.0   black:26
##  Median :2.977   Median :23.00   Median :121.0   other:67
##  Mean   :2.945   Mean   :23.24   Mean   :129.8
##  3rd Qu.:3.487   3rd Qu.:26.00   3rd Qu.:140.0
##  Max.   :4.990   Max.   :45.00   Max.   :250.0
##    smoke         prem.labor  hypertension     uterine
##  Mode :logical   FALSE:159   Mode :logical   Mode :logical
##  FALSE:115       TRUE : 30   FALSE:177       FALSE:161
##  TRUE :74                    TRUE :12        TRUE :28
##  NA's :0                     NA's :0         NA's :0
##
##
##  physician.visits
##  0 :100
##  1 : 47
##  2+: 42
##
##
##
```

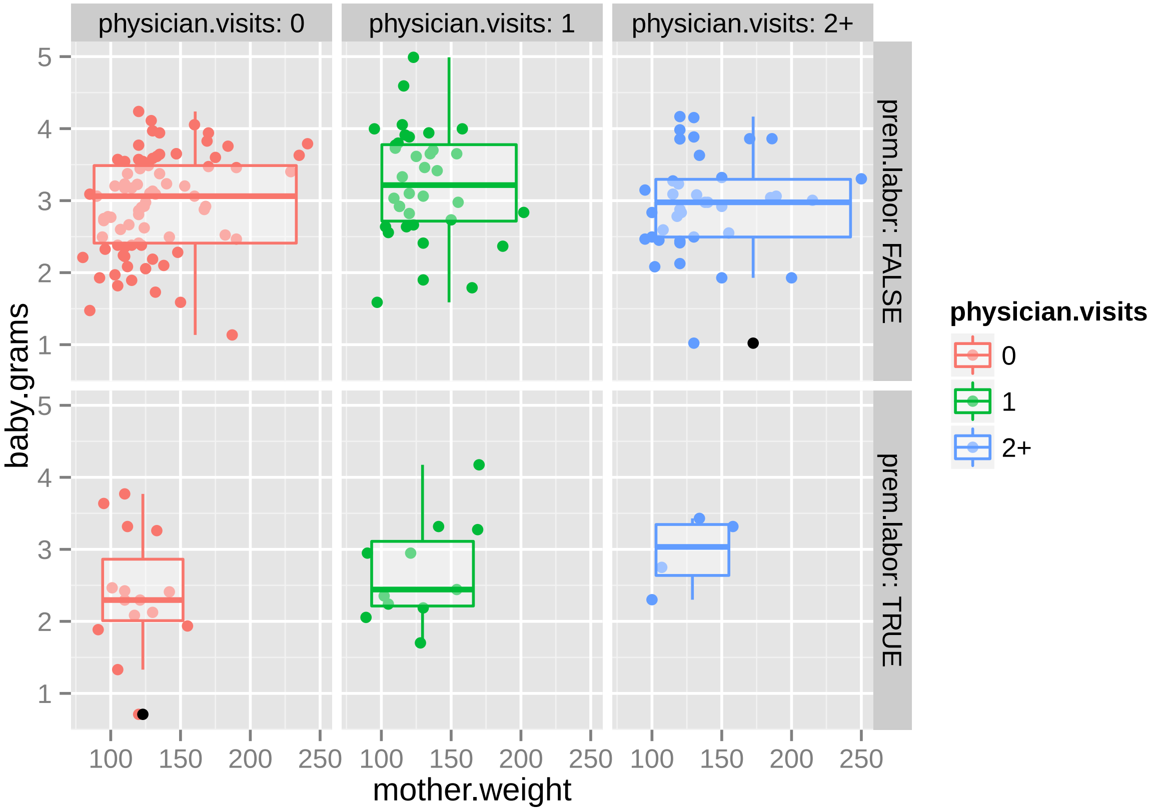
Datatset has only 2 quantitative variables apart from infant weights, however, as shown in the table below, they do not demonstrate strong correlation between each other, which suggests that these variables will not be sufficient themselves in explaining birth weight variation. Variable `mother.age` demonstrate the lowest correlation with `baby.grams` and will most probably be omitted in the prediction models further on.

```
##               baby.grams mother.age mother.weight
## baby.grams    1.00000000 0.09031781     0.1857333
## mother.age    0.09031781 1.00000000     0.1800732
## mother.weight 0.18573328 0.18007315     1.0000000
```

The following charts demonstrate boxplots and splits of the `baby.grams` data points vs `mother.weight` across various categorical and binary variables that make part of the working dataset.



First chart shows some evidence in importance of race in predicting the risk of giving birth to low weight baby, as well as smoking habits during pregnancy. Facet scatterplots show that data point corresponding to each of these factors' combinations group around different median values, which can suggest their predictive power on the newborn infant's weight.



The second chart splits all the observations in sample into several groups by number of physician vistis in the first trimester and occurance of premature labor by each subject of the study. For mothers without previous premature births no significant difference is observed with repsect to number of physician visits, whereas women who had premature labors before are exposed to the higher risk of giving birth to low weight baby if they do not pay enough visits to physician during the first trimester of their pregnancy term. However, we need to account for existing outliers in the sample dataset, as there are at three observations of infants that were born with weight less than or equal to 1 kg, which significantly differs from the majority of observations in this dataset.

Exploring Linear Relationships

Our first attempt to find a statistically significant model fit will go through fitting linear model of different factors in dataset vs `baby.grams`, which is the variable of our interest.

For the purposes of further validation and comparison of results we attribute 75% of the data to training set, saving the rest of the observations for test set.

As dataset consists of only 8 explaining variables, it is computationally acceptable to select the best possible subset of the variables explaining the response of the model.

```
library (leaps)
regfit.full=regsubsets(baby.grams~., bwt.grams.train, nvmax =19)
reg.summary = summary(regfit.full)
par(mfrow =c(2,2))
plot(reg.summary$rss ,xlab=" Number of Variables ",ylab=" RSS", type="l")
plot(reg.summary$adjr2 ,xlab =" Number of Variables ", ylab=" Adjusted RSq",type="l")
max.adjr2=which.max (reg.summary$adjr2)
max.adjr2
```

```
## [1] 6
```

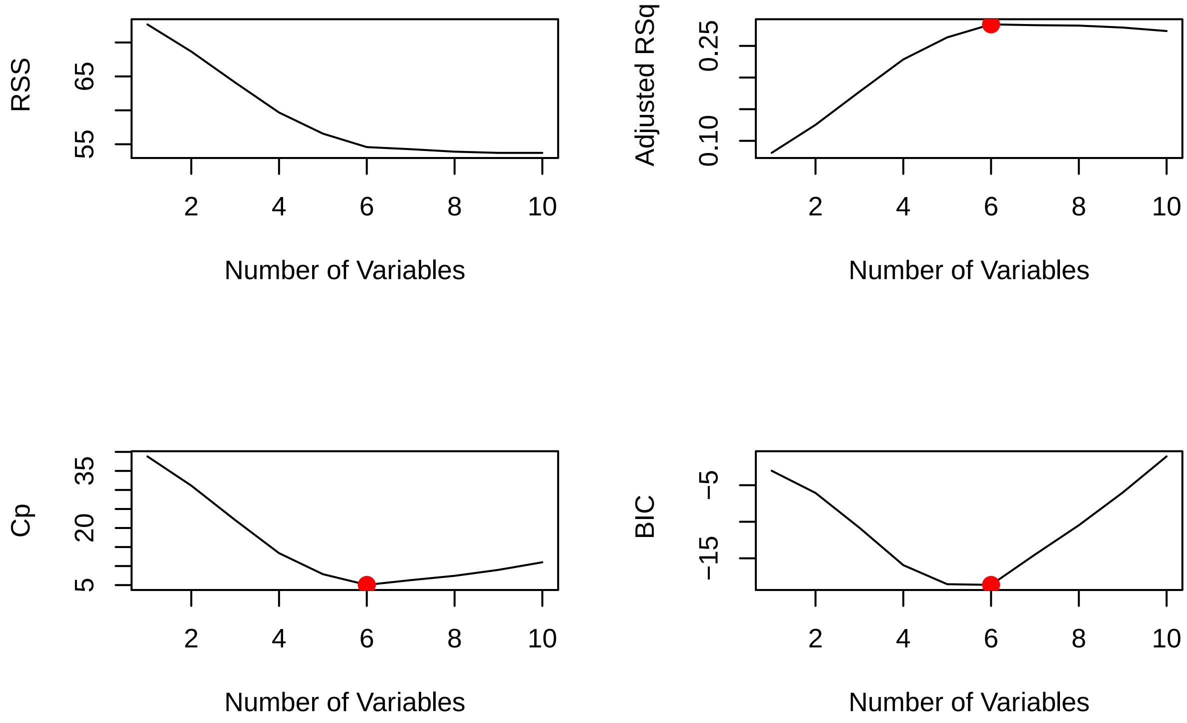
```
points (max.adjr2, reg.summary$adjr2[max.adjr2], col ="red",cex =2, pch =20)

plot(reg.summary$cp ,xlab =" Number of Variables ", ylab="Cp", type='l')
min.cp= which.min (reg.summary$cp )
min.cp
```

```
## [1] 6
```

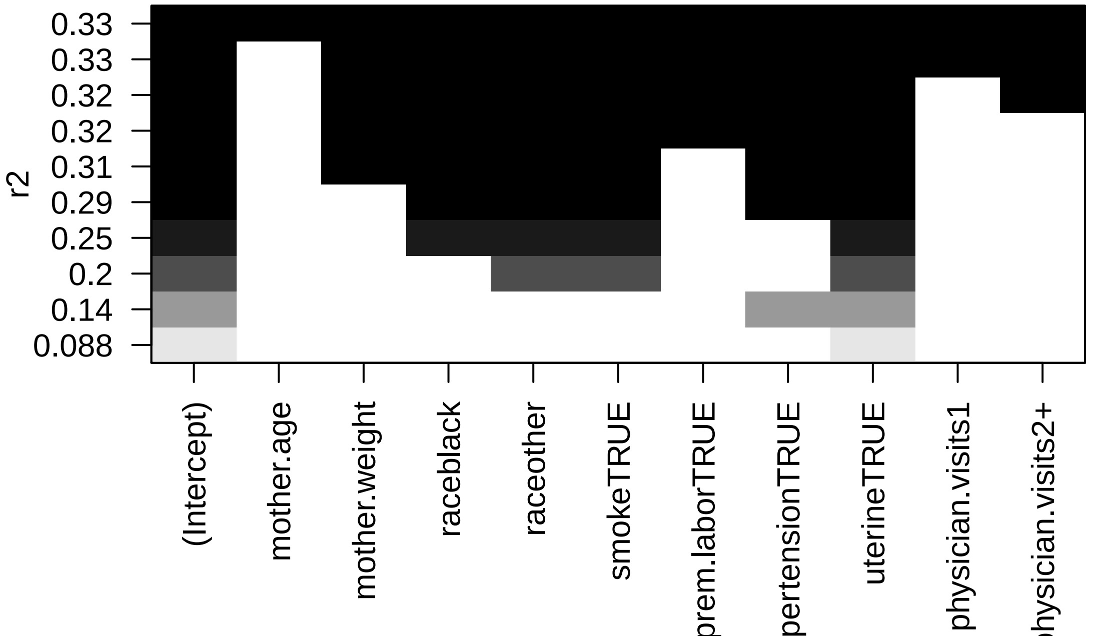
```
points (min.cp, reg.summary$cp[min.cp], col ="red",cex =2, pch =20)

min.bic = which.min(reg.summary$bic)
min.bic
```

```
## [1] 6
```

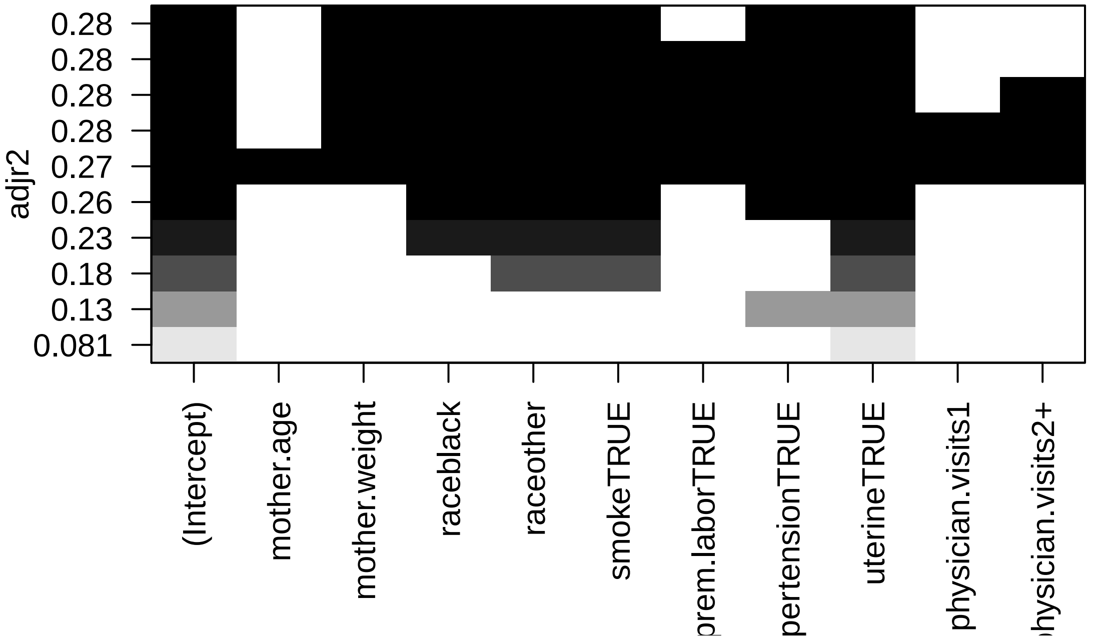
```
plot(reg.summary$bic ,xlab=" Number of Variables ",ylab=" BIC", type='l')
points (min.bic, reg.summary$bic [min.bic], col =" red",cex =2, pch =20)
```



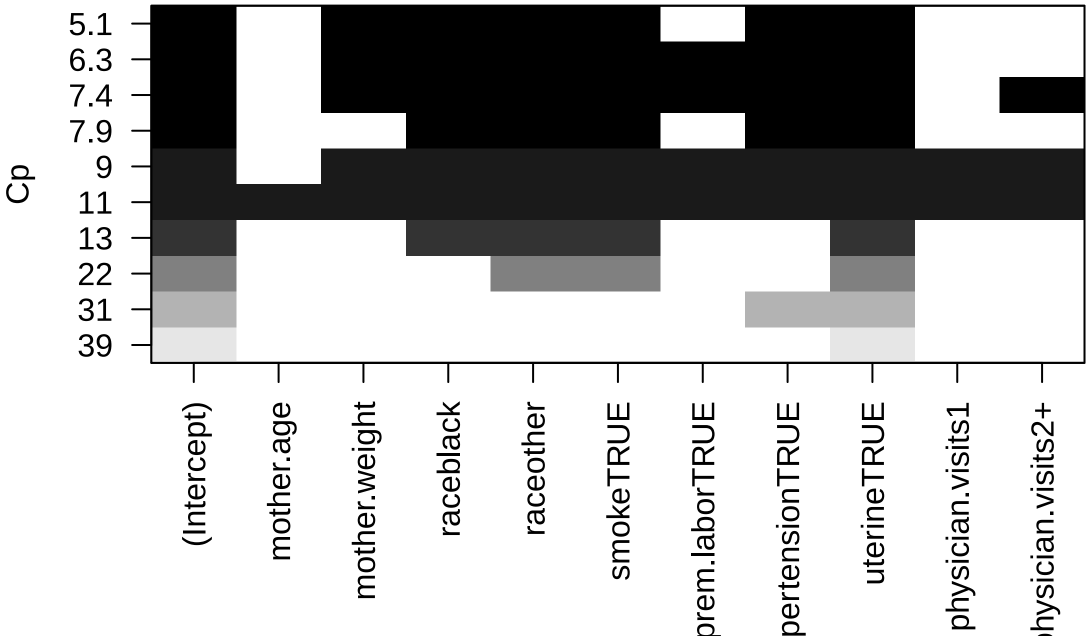
```
par(mfrow = c(1,1))
plot(regfit.full ,scale ="r2", cex.axis = 0.1, las = 1)
```



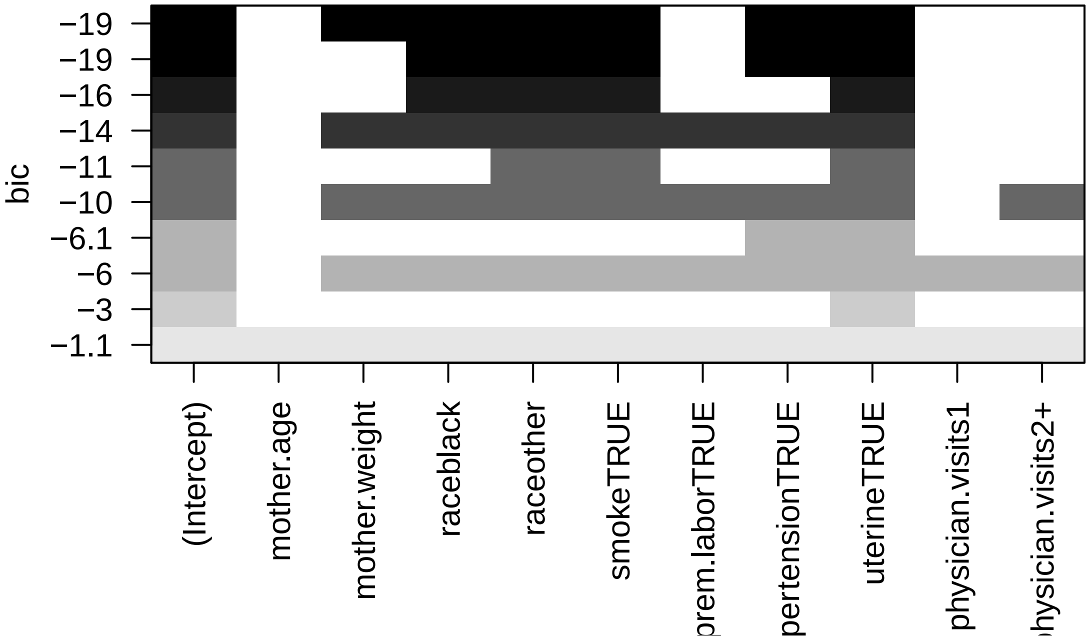
```r
plot(regfit.full ,scale ="adjr2", cex.axis = 0.1, las = 1)
```



```r
plot(regfit.full ,scale ="Cp", cex.axis = 0.1, las = 1)
```



```r
plot(regfit.full ,scale ="bic", cex.axis = 0.1, las = 1)
```



```r
coef(regfit.full, max.adjr2)
```

```
##       (Intercept)     mother.weight         raceblack         raceother
##       2.955952101       0.004267934      -0.566615552      -0.487289998
##         smokeTRUE hypertensionTRUE       uterineTRUE
##      -0.465884551      -0.675297653      -0.572809187
```

```
coef(regfit.full, min.cp)
```

```
##      (Intercept)    mother.weight        raceblack        raceother 
##      2.955952101      0.004267934     -0.566615552     -0.487289998 
##        smokeTRUE hypertensionTRUE      uterineTRUE 
##     -0.465884551     -0.675297653     -0.572809187
```

```
coef(regfit.full, min.bic)
```

```
##      (Intercept)    mother.weight        raceblack        raceother 
##      2.955952101      0.004267934     -0.566615552     -0.487289998 
##        smokeTRUE hypertensionTRUE      uterineTRUE 
##     -0.465884551     -0.675297653     -0.572809187
```

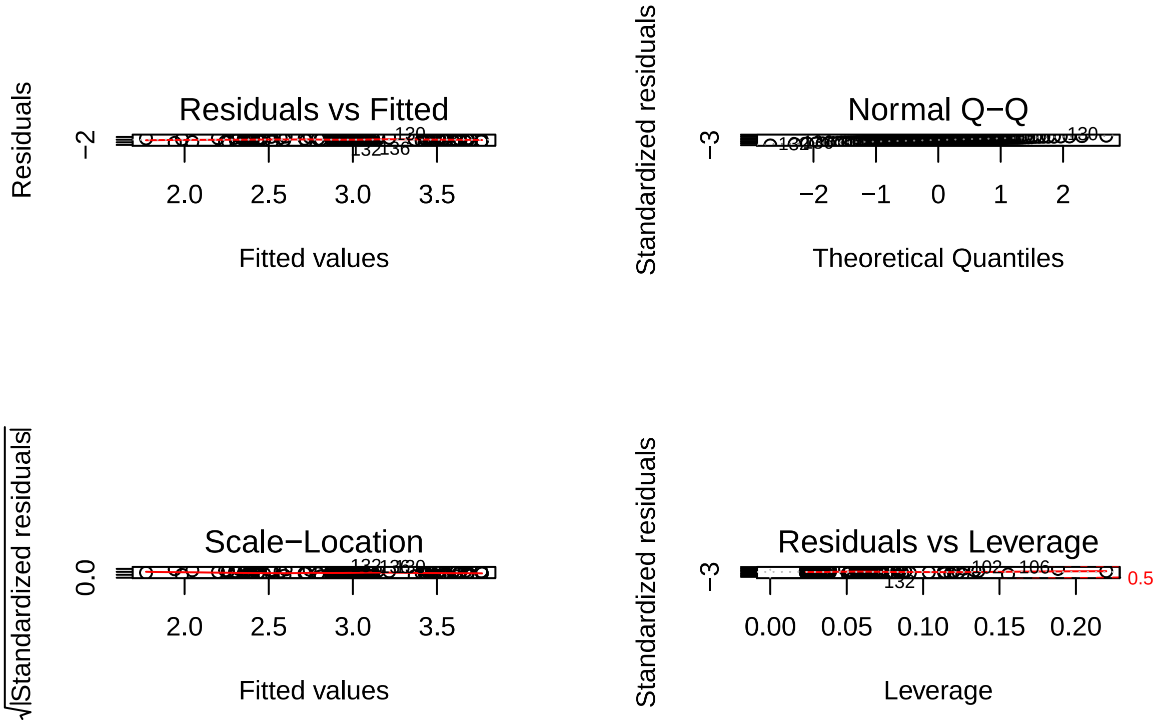
```
#Linear regression with the predictors selected by best subset
lm.fit = lm( baby.grams~ mother.weight+race+smoke+hypertension+uterine, data=bwt.grams.train)
summary(lm.fit)
```

```
## 
## Call:
## lm(formula = baby.grams ~ mother.weight + race + smoke + hypertension + 
##     uterine, data = bwt.grams.train)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.91697 -0.40046  0.04839  0.36803  1.50909 
## 
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)    
## (Intercept)       2.955952   0.283542  10.425  < 2e-16 ***
## mother.weight     0.004268   0.001932   2.209 0.028907 *  
## raceblack        -0.566616   0.165387  -3.426 0.000814 ***
## raceother        -0.487290   0.130845  -3.724 0.000288 ***
## smokeTRUE        -0.465885   0.120194  -3.876 0.000165 ***
## hypertensionTRUE -0.675298   0.215305  -3.136 0.002102 ** 
## uterineTRUE      -0.572809   0.155560  -3.682 0.000334 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.6382 on 134 degrees of freedom
## Multiple R-squared:  0.3147, Adjusted R-squared:  0.284 
## F-statistic: 10.25 on 6 and 134 DF,  p-value: 2.508e-09
```

```
confint(lm.fit)
```

```
##                          2.5 %       97.5 %
## (Intercept)       2.3951550476  3.516749155
## mother.weight     0.0004458445  0.008090024
## raceblack        -0.8937222132 -0.239508891
## raceother        -0.7460777241 -0.228502271
## smokeTRUE        -0.7036072464 -0.228161856
## hypertensionTRUE -1.1011324705 -0.249462836
## uterineTRUE      -0.8804804084 -0.265137966
```

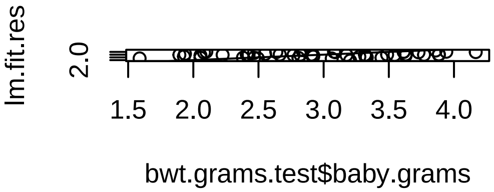
```r
par(mfrow = c(2, 2))
plot(lm.fit)
```



```r
lm.fit.res= predict(lm.fit, bwt.grams.test)
mean((lm.fit.res -bwt.grams.test$baby.grams)^2)
```

```
## [1] 0.467712
```

```r
plot(bwt.grams.test$baby.grams,lm.fit.res)
abline (0,1)
```

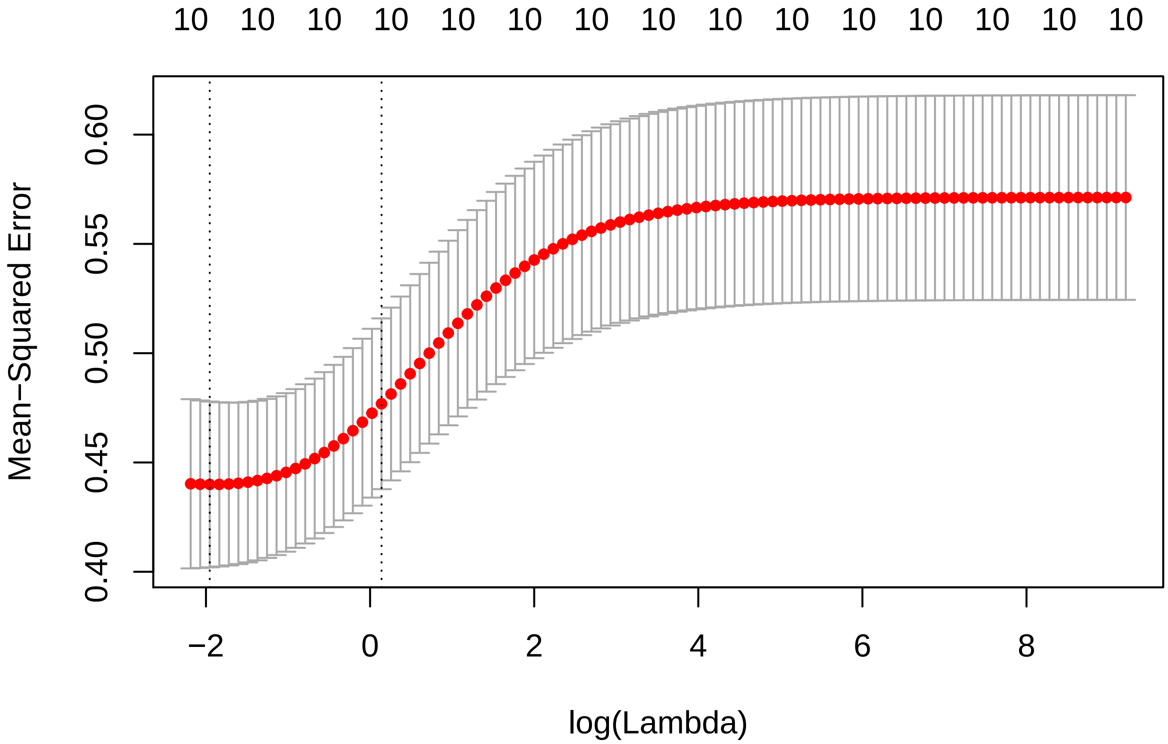


Fitting Penalized Linear Models

```r
library(glmnet)
```

```
## Loading required package: Matrix
## Loading required package: foreach
## Loaded glmnet 2.0-2
```

```
##
bwt.x.train=model.matrix( baby.grams~., data=bwt.grams.train)[,-1]
bwt.y.train=bwt.grams.train$baby.grams

bwt.x.test=model.matrix( baby.grams~., data=bwt.grams.test)[,-1]
bwt.y.test=bwt.grams.test[,1]

grid.bwt =10^seq (-1,4, length =100)

# With alpha =0, glmnet computes the ridge

ridge =cv.glmnet(bwt.x.train,bwt.y.train,alpha =0, lambda =grid.bwt, nfolds=6)
plot(ridge)
```



```
ridge.opt = glmnet(bwt.x.train,bwt.y.train,alpha =0, lambda =ridge$lambda.min)
ridge.opt$beta
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                               s0
## mother.age           0.002379072
## mother.weight        0.003740105
## raceblack           -0.390607098
## raceother           -0.320259968
## smokeTRUE           -0.311151546
## prem.laborTRUE      -0.195046362
## hypertensionTRUE    -0.567559864
## uterineTRUE         -0.436600708
## physician.visits1    0.124887593
## physician.visits2+ -0.068069904
```

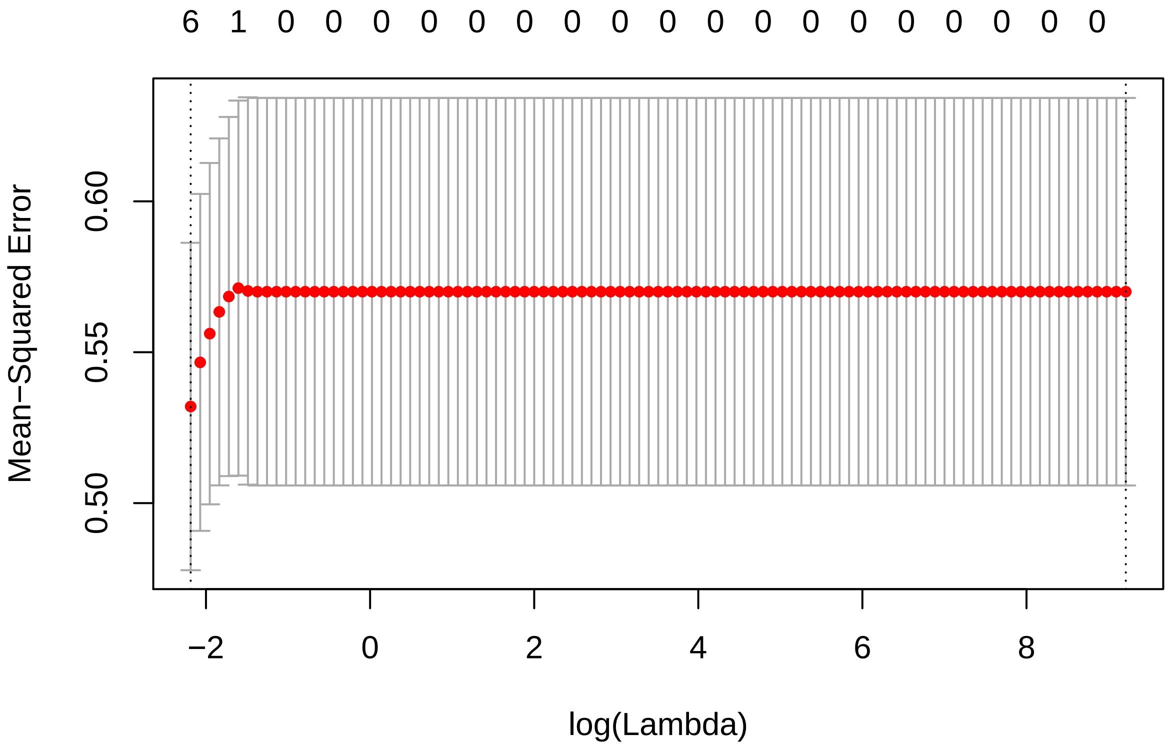
```
ridge.opt.res = predict(ridge.opt, s =ridge$lambda.min, newx=bwt.x.test)
mean((ridge.opt.res -bwt.y.test)^2)
```

```
## [1] 0.4173598
```

```
# With alpha =1, glmnet computes the lasso
lasso =cv.glmnet(bwt.x.train,bwt.y.train,alpha =1, lambda =grid.bwt, nfolds=6)
lasso$lambda.min
```

```
## [1] 0.1123324
```

```
plot(lasso)
```



```
lasso.opt = glmnet(bwt.x.train,bwt.y.train,alpha =1, lambda =lasso$lambda.min)
lasso.opt$beta
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                               s0
## mother.age           .
## mother.weight        0.0007578126
## raceblack            .
## raceother           -0.0657574959
## smokeTRUE           -0.1162215546
## prem.laborTRUE      -0.0383714410
## hypertensionTRUE    -0.1709866881
## uterineTRUE         -0.2835236922
## physician.visits1    .
## physician.visits2+   .
```

```
lasso.opt.res = predict(lasso.opt, s =lasso$lambda.min, newx=bwt.x.test)
mean((lasso.opt.res -bwt.y.test)^2)
```

```
## [1] 0.3968205
```

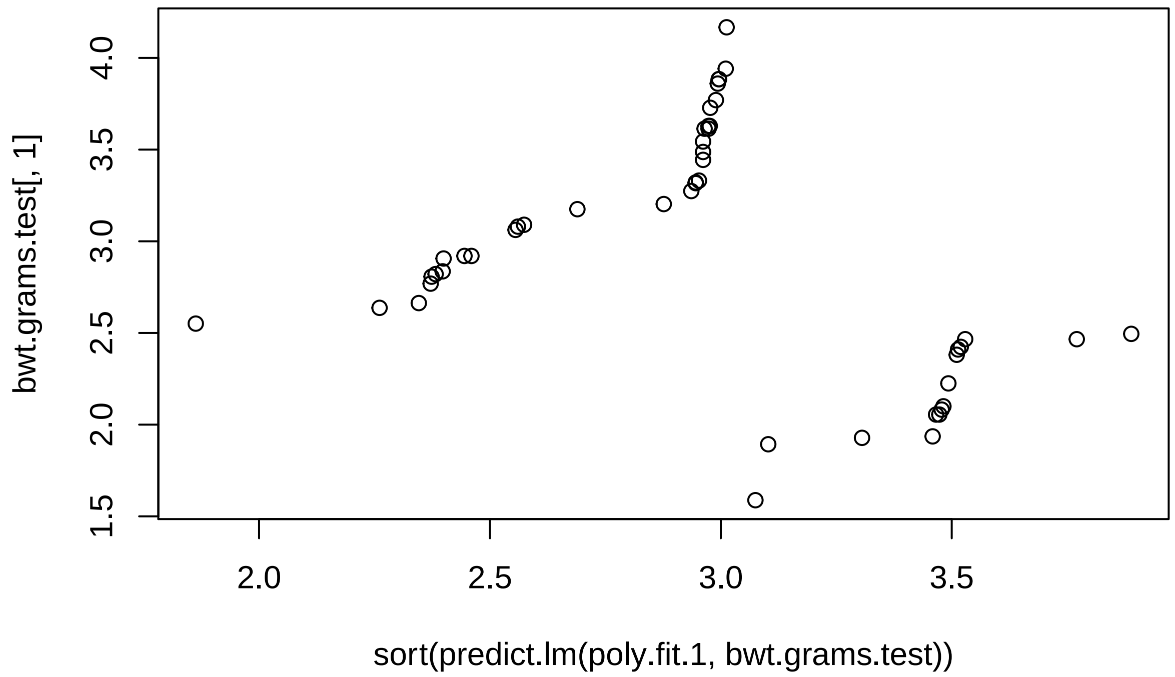
Testing for Non-linear Relationships

Fitting Polynomial Regression

```
#Create train and test
set.seed(1)
train <- sample(1:nrow(bwt.grams), floor(0.75*nrow(bwt.grams)))
bwt.grams.train <- bwt.grams[train,]
bwt.grams.test <- bwt.grams[-train,]

#Polynomial fit for best subset
poly.fit.1 = lm(baby.grams ~ hypertension + uterine + smoke + race + poly(mother.weight, 2), data = bwt
mean((predict.lm(poly.fit.1, bwt.grams.test) - bwt.grams.test[,1])^2)
```

```
## [1] 0.4813745
```

```
plot(sort(predict.lm(poly.fit.1, bwt.grams.test)), bwt.grams.test[,1])
```



```
anova(poly.fit.1)
```

```
## Analysis of Variance Table
##
```

```
## Response: baby.grams
##                         Df Sum Sq Mean Sq F value    Pr(>F)    
## hypertension             1  2.827  2.8270  6.9365  0.009446 ** 
## uterine                  1  8.145  8.1454 19.9863 1.654e-05 ***
## smoke                    1  3.026  3.0264  7.4259  0.007294 ** 
## race                     2  9.074  4.5371 11.1328 3.386e-05 ***
## poly(mother.weight, 2)   2  2.363  1.1813  2.8986  0.058590 .  
## Residuals              133 54.204  0.4075                      
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

When we fit a polynomial model on the predictors obtained from best subset, we observe a Mean Squared Error of `0.4813745`. The smaller the Mean Squared Error, the closer the fit is to the data. But, as he value of MSE is high, it suggests that this model does not provide a good fit for the data. The plot also shows that there are irregularities in the prediction and that the polynomial model of degree 2 obtained by using predictors suggested by the best subset is not sufficient. When we perform Analysis of Variance (ANOVA) on the polynomial fit, we see that, the *p-values* for the all the predictors - except `mother.weight` are less that `0.5` and thus, the NULL hypothesis that these variables affect the baby weight at birth can be rejected.

Different models were tried by increasing the degree of the polynomial but still using the predictors suggested by the best subset and the following results were obtained:

```
poly.fit.2 = lm(baby.grams ~ hypertension + uterine + smoke + race + poly(mother.weight, 3), data = bwt
mean((predict.lm(poly.fit.2, bwt.grams.test) - bwt.grams.test[,1])^2)
```

```
## [1] 0.4640868
```

```
poly.fit.3 = lm(baby.grams ~ hypertension + uterine + smoke + race + poly(mother.weight, 4), data = bwt
mean((predict.lm(poly.fit.3, bwt.grams.test) - bwt.grams.test[,1])^2)
```

```
## [1] 0.4619314
```

```
anova(poly.fit.1, poly.fit.2, poly.fit.3)
```

```
## Analysis of Variance Table
## 
## Model 1: baby.grams ~ hypertension + uterine + smoke + race + poly(mother.weight, 
##     2)
## Model 2: baby.grams ~ hypertension + uterine + smoke + race + poly(mother.weight, 
##     3)
## Model 3: baby.grams ~ hypertension + uterine + smoke + race + poly(mother.weight, 
##     4)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    133 54.204                           
## 2    132 53.825  1   0.37915 0.9239 0.3382
## 3    131 53.761  1   0.06393 0.1558 0.6937
```

We note that as the degree of the polynomial increases, the MSE decreases, but the drop is not significant, suggesting that these predictors are not sufficient enough to predict the correct baby weight. Performing the ANOVA test to compare how the three models perform with respect to each other, we observe high *p-values* which state that the none of the models are good enough.

When we remove the predictors with very low *p-values*, which were suggested by the best subset - namely `smoke`, `race` and add other predictors which were rejected by the best-subset, namely - `mother.age`, `prem.labor` and `physician.visits`, we see that the Mean Squared Error starts to decrease. A low MSE denotes a better fit. Thus, the predictors which were rejected by the best subset selection, were actually significant in predicting the correct birthweight.

```
poly.fit.4 = lm(baby.grams ~ hypertension + uterine + poly(mother.age,2) + poly(mother.weight,3), data =
mean((predict.lm(poly.fit.4, bwt.grams.test) - bwt.grams.test[,1])^2)
```

```
## [1] 0.3890751
```

```
poly.fit.5 = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor + poly(mother.age,2) + poly(mo
mean((predict.lm(poly.fit.5, bwt.grams.test) - bwt.grams.test[,1])^2)
```

```
## [1] 0.3865828
```

```
poly.fit.6 = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor + poly(mother.age,2) + poly(mo
mean((predict.lm(poly.fit.6, bwt.grams.test) - bwt.grams.test[,1])^2)
```

```
## [1] 0.3214657
```

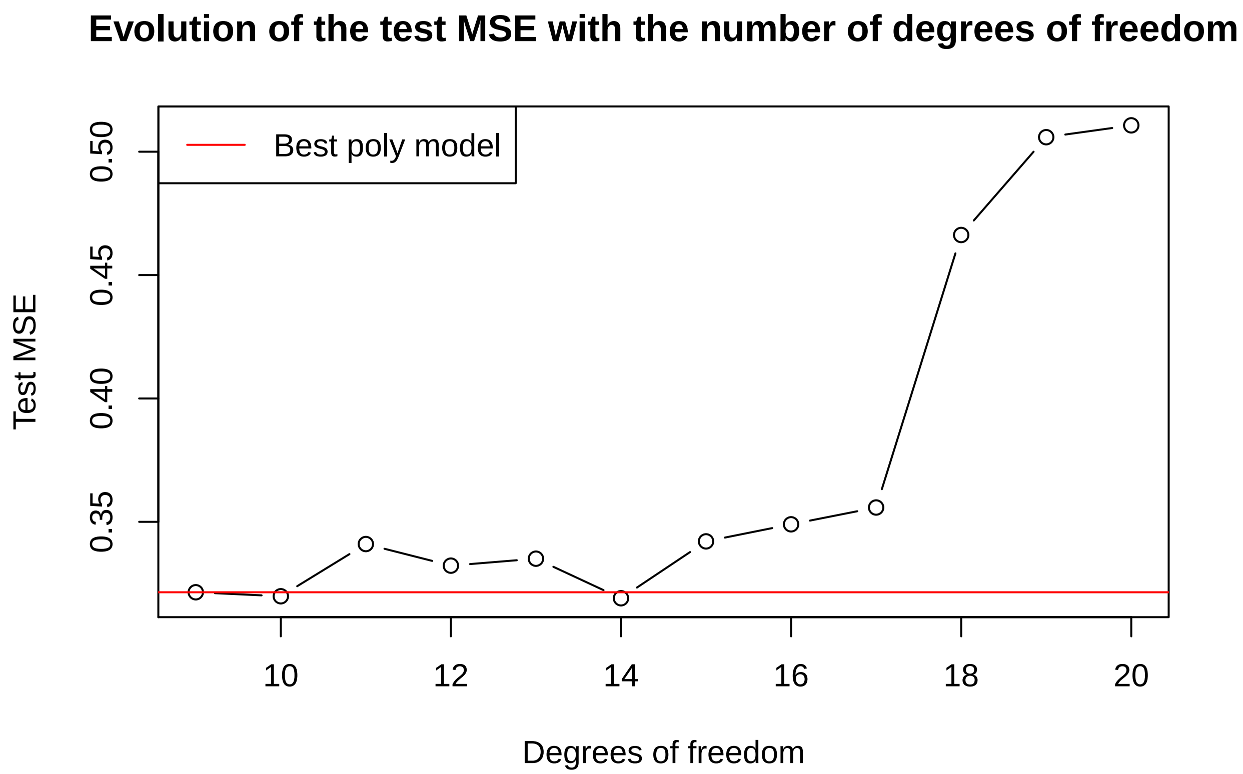
Fitting Natural Splines

```
library(splines)
poly.fit = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor + poly(mother.age, 2) + poly(moth
pred = predict.lm(poly.fit, bwt.grams.test)
mse = mean((pred - bwt.grams.test[,1])^2)
mse
```

```
## [1] 0.3214657
```

Now that we have tried a lot of different polynomial regressions we can wonder if it is possible to improve our best polynomial model by introducing splines. Here we added in the regression formula several basis functions for the variable *mother.weight*. Between each knots we fit a $9 - degree - polynomial$. We tried different values for the number of degrees of freedom so as to find the best parameter. Here is the resulting plot:

```
max_df = 20
MSE = 9:(max_df)
for (k in 9:max_df){
    splines.fit = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor
                     + bs(mother.weight, df=k, degree=9) + poly(mother.age, 2),
                     data=bwt.grams.train)
    pred = predict(splines.fit, bwt.grams.test)
    mse = mean((pred-bwt.grams.test$baby.grams)^2)
    MSE[k-8] = mse
}
plot(9:max_df, MSE, xlab='Degrees of freedom',
     ylab='Test MSE',
     main='Evolution of the test MSE with the number of degrees of freedom',
     type='b')
abline(.3214657, 0, col='RED')
legend("topleft", c('Best poly model'), col=c('RED'), lty=c(1))
```



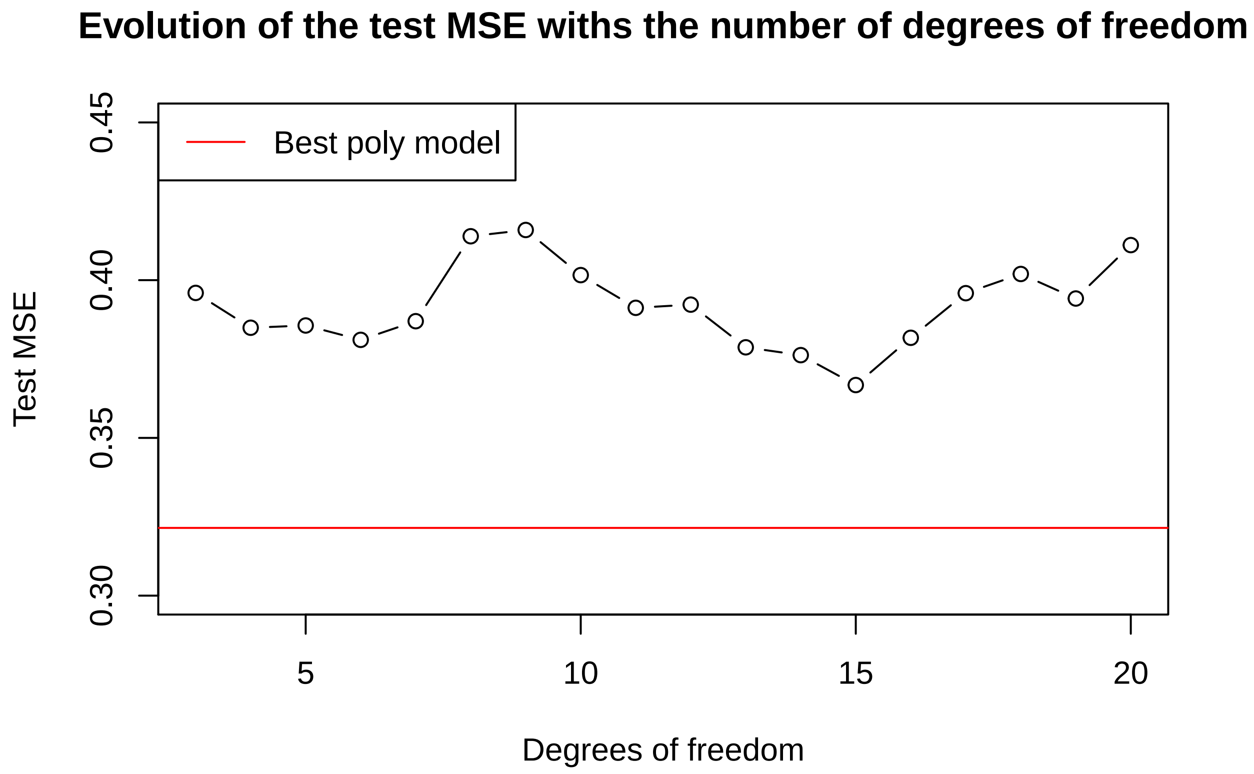
The minimum MSE is obtained when we have 14 degrees of freedom. With the R built-in function $bs()$, R automatically puts knots on the quantile values of the variable. Here for 14 degrees of freedom our knots are: $q_{16.7}$, $q_{33.3}$, q_{50}, $q_{66.7}$ and $q_{83.3}$. Thus between each quantile R fits a degree 9 polynomial on the mothers' weights. It also makes sure that the 1st, 2nd, ... and 8th derivatives are continuous at each knots. Thus the relation between the number of degrees of freedom d and the number of knots K is the following:

$$d = K + 9$$

We can see that this formula is verified in our case $(14 = 5 + 9)$.

Natural splines are fitted in order to account for more flexibility in the model in attempt to find a better fit.

```
max_df = 20
MSE = 3:(max_df)
for (k in 3:max_df){
    splines.fit = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor
                  + ns(mother.weight, df=k) + poly(mother.age, 2),
                  data=bwt.grams.train)
    pred = predict(splines.fit, bwt.grams.test)
    mse = mean((pred-bwt.grams.test$baby.grams)^2)
    MSE[k-2] = mse
}
plot(3:max_df, MSE, xlab='Degrees of freedom', ylab='Test MSE',
     main='Evolution of the test MSE withs the number of degrees of freedom',
     type='b', ylim=c(0.3,0.45))
abline(.3214657, 0, col='RED')
legend("topleft", c('Best poly model'), col=c('RED'), lty=c(1))
```



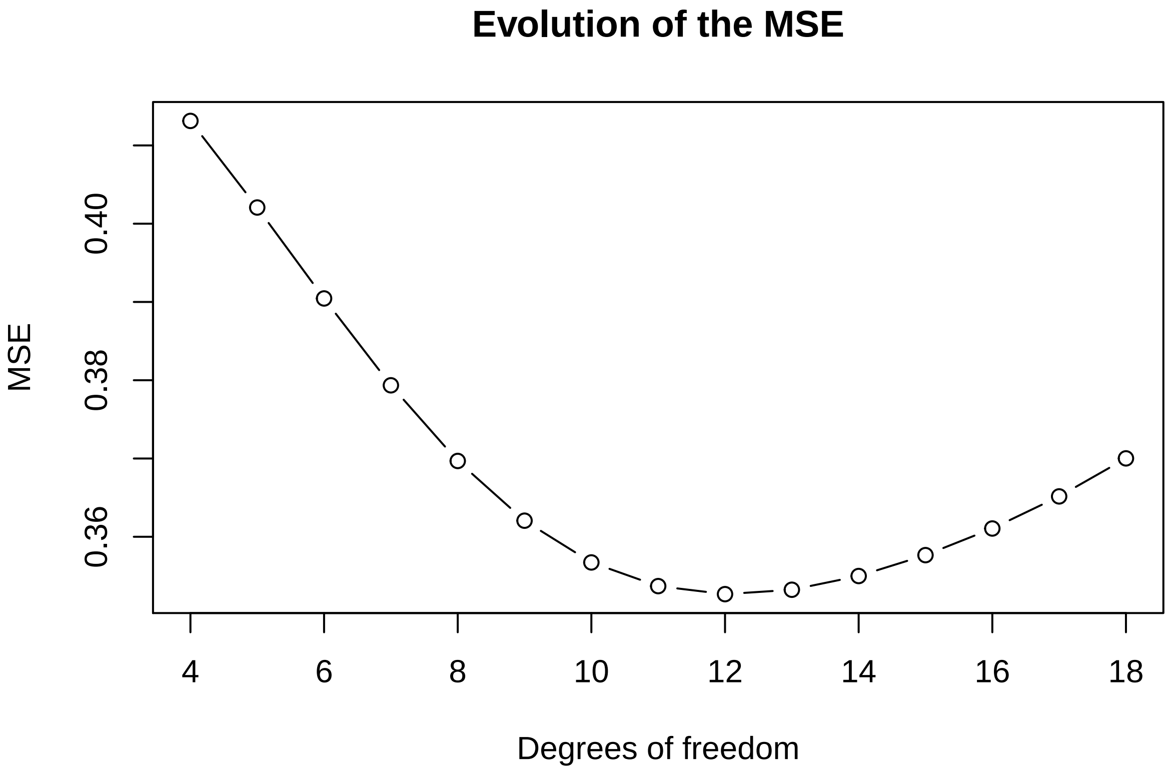
When we try with natural splines we have worse results than with the normal splines model. It is mainly due to the fact that R can only fit cubic natural splines, there is no degree argument in the R built-in function.

Now we can try to see if there is an improvement if we use smoothed splines. We have to use the General Additive Models R library to perform this analysis.

```
library(gam)
```

```
## Loaded gam 1.12
```

```
MSE = 4:18
for (k in 4:18){
    gam.fit = gam(baby.grams ~ prem.labor+uterine+ hypertension + smoke
                  + s(mother.weight, k) + poly(mother.age, 2), data=bwt.grams.train)
    pred = predict(gam.fit, bwt.grams.test)
    mse = mean((pred-bwt.grams.test$baby.grams)^2)
    MSE[k-3] = mse
}
plot(4:18, MSE, type='b', main='Evolution of the MSE', xlab='Degrees of freedom')
```



We can notice that the results are still not better than with our optimal model with degree 9 splines. The smoothing effect does not bring more predictive power to the final model. To conclude this part on splines we managed to find a model that outperforms slightly our best polynomial model. This was expected as splines models are more flexible than polynomial models. Nonetheless the improvement in test MSE is quite low and we can wonder if the splines model is really better than the polynomial model. Indeed, fitting a degree nine polynomial between each splines brings a lot of flexibility to the model but the increase of variance can be huge too. If we have had more observations we could have answered to this question by testing our models on a big test set. Nevertheless we can run a ANOVA test to verify if the difference between our best polynomial model and our best splines model is really significant:

```
best_poly = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor
               + poly(mother.age, 2) + poly(mother.weight, 9), data=bwt.grams.train)

best_splines = lm(baby.grams ~ hypertension + uterine + smoke + prem.labor
                  + ns(mother.weight, df=14) + poly(mother.age, 2), data=bwt.grams.train)

anova(best_poly, best_splines)
```

```
## Analysis of Variance Table
##
## Model 1: baby.grams ~ hypertension + uterine + smoke + prem.labor + poly(mother.age,
##     2) + poly(mother.weight, 9)
## Model 2: baby.grams ~ hypertension + uterine + smoke + prem.labor + ns(mother.weight,
##     df = 14) + poly(mother.age, 2)
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    125 56.829
## 2    120 52.185  5    4.6437 2.1356 0.06576 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can see that the resulting p-value is a little above 0.066. Thus depending on the level of the test we want, we may reject or accept H_0. Nevertheless we can say that the difference of performance between those two tests is not obvious. Thus we will maybe prefer to keep the less complex model *ie* the polynomial model.

Building Classification Model

Testing for Claassification Threshold

Fitting Logistic Regression

```
##   below.2500         mother.age    mother.weight      race
##   Mode :logical   Min.   :14.00   Min.   : 80.0   white:96
##   FALSE:153       1st Qu.:19.00   1st Qu.:110.0   black:26
##   TRUE :36        Median :23.00   Median :121.0   other:67
##   NA's :0         Mean   :23.24   Mean   :129.8
##                   3rd Qu.:26.00   3rd Qu.:140.0
##                   Max.   :45.00   Max.   :250.0
##     smoke         prem.labor  hypertension     uterine
##   Mode :logical   FALSE:159   Mode :logical   Mode :logical
##   FALSE:115       TRUE : 30   FALSE:177       FALSE:161
##   TRUE :74                    TRUE :12        TRUE :28
##   NA's :0                     NA's :0         NA's :0
##
##
##   physician.visits
##   0 :100
##   1 : 47
##   2+: 42
##
##
##
```

```
#Logistic regression with the predictors selected by best subset
log.fit = glm( below.2500~ mother.weight+race+smoke+hypertension+uterine, family = binomial, data=bwt[t
summary(log.fit)
```

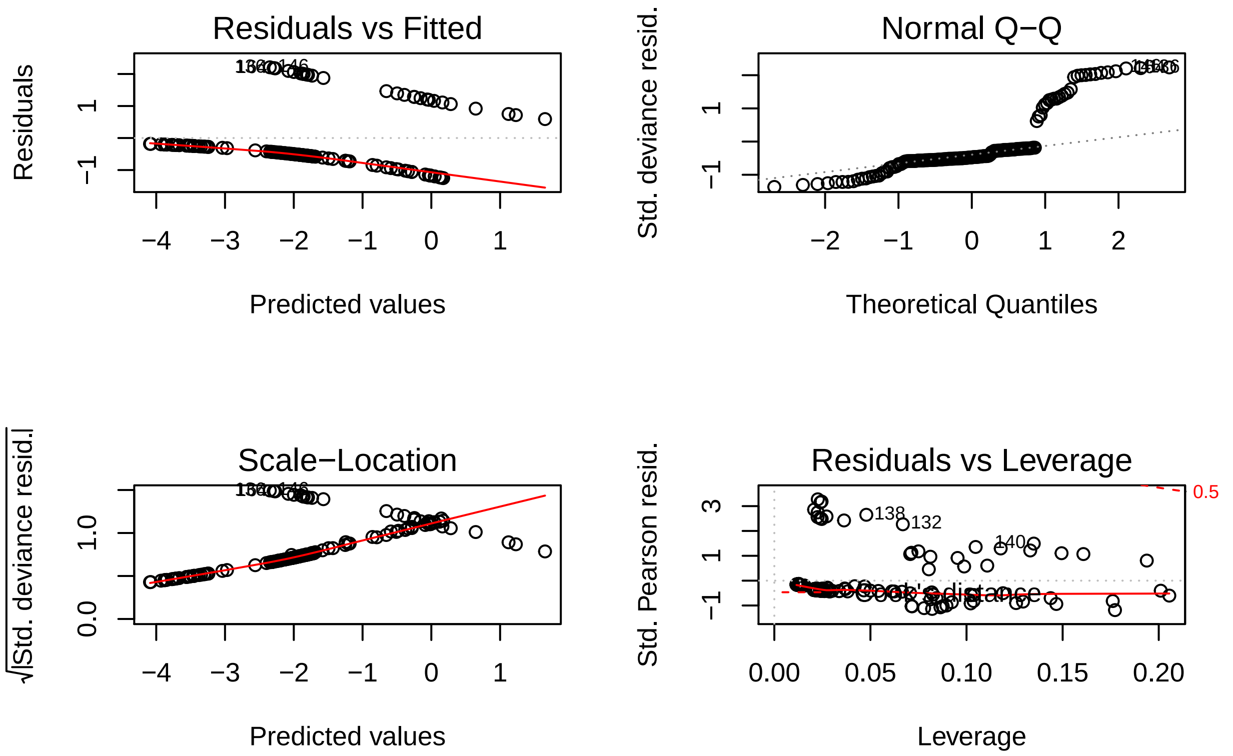
```
##
## Call:
## glm(formula = below.2500 ~ mother.weight + race + smoke + hypertension +
##     uterine, family = binomial, data = bwt[train, ])
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2514  -0.5621  -0.4725  -0.2155   2.2089
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -2.075662   1.308135  -1.587  0.11257
## mother.weight    -0.010603   0.009185  -1.154  0.24832
## raceblack         1.532566   0.730453   2.098  0.03590 *
## raceother         1.374900   0.637114   2.158  0.03093 *
## smokeTRUE         1.317744   0.564019   2.336  0.01947 *
## hypertensionTRUE  1.853591   0.780310   2.375  0.01753 *
## uterineTRUE       1.884512   0.576856   3.267  0.00109 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 137.72  on 140  degrees of freedom
## Residual deviance: 111.32  on 134  degrees of freedom
## AIC: 125.32
##
## Number of Fisher Scoring iterations: 5
```

```r
confint(log.fit)
```

```
## Waiting for profiling to be done...
```

```
##                          2.5 %      97.5 %
## (Intercept)      -4.58539093 0.580124576
## mother.weight    -0.03028210 0.006102122
## raceblack         0.07435925 2.988818942
## raceother         0.16417622 2.689380592
## smokeTRUE         0.24311635 2.476615909
## hypertensionTRUE  0.27811154 3.410580839
## uterineTRUE       0.76088981 3.046659702
```

```r
par(mfrow = c(2, 2))
plot(log.fit)
```



```r
pred.train <- predict(log.fit, type = "response")
low.train <- sapply(pred.train, function(x) {ifelse(x > 0.5, 1, 0)})
table(low.train, bwt$below.2500[train])
```

```
##
## low.train FALSE TRUE
##         0   110   20
##         1     4    7
```

```r
mean(low.train == bwt$below.2500[train])
```

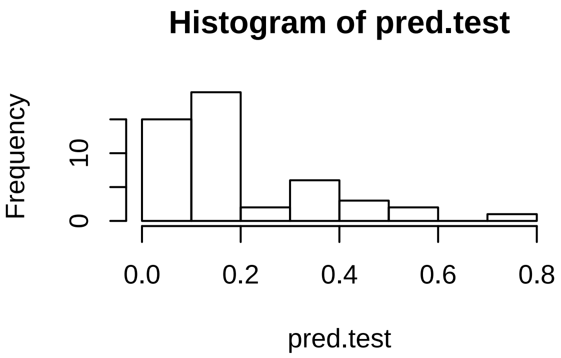
```
## [1] 0.8297872
```

```r
pred.test <- predict(log.fit, newdata = bwt[-train, -1], type = "response")
hist(pred.test)
low.test <- sapply(pred.test, function(x) {ifelse(x > 0.2, 1, 0)})
table(low.test, bwt$below.2500[-train])
```

```
##
## low.test FALSE TRUE
##        0    28    6
##        1    11    3
```

```r
mean(low.test == bwt$below.2500[-train])
```

```
## [1] 0.6458333
```



Results and Conclusion