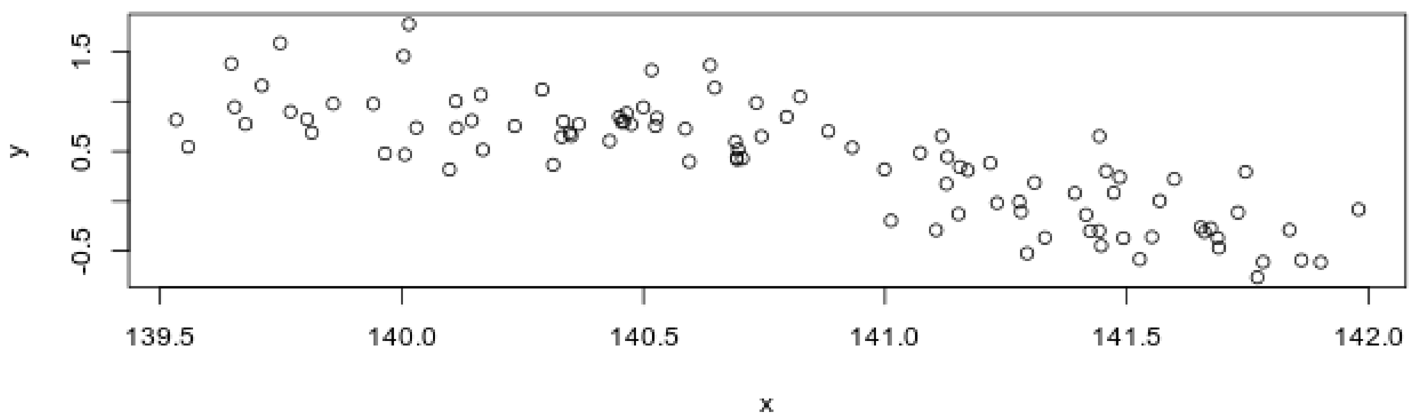
Chapter 7

Lab

Polynomial Functions and Cut Points

```
load(paste0(here::here(), "/ISLR/7.R.RData"))

plot(x, y)
```




```
fit <- lm(y ~ x)
fit2 <- lm(y ~ 1 + x + I(x^2))
```

```
wage <- data.table(ISLR::Wage)
```

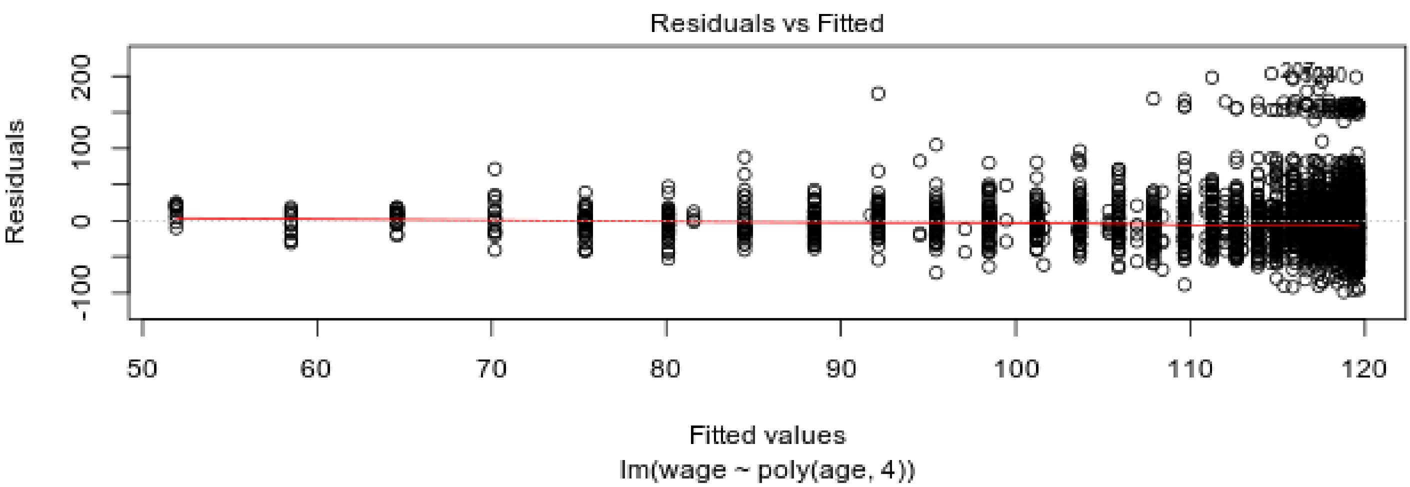
Polynomial Regression and Step Functions

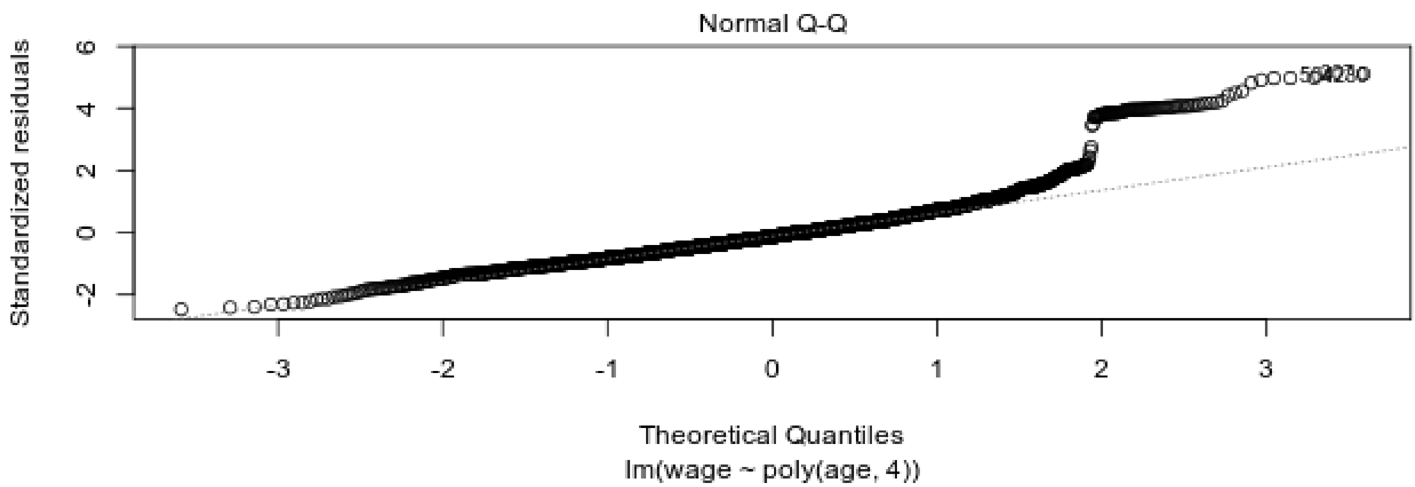
```
fit <- lm(wage ~ poly(age, 4), data = wage)

summary(fit)
```

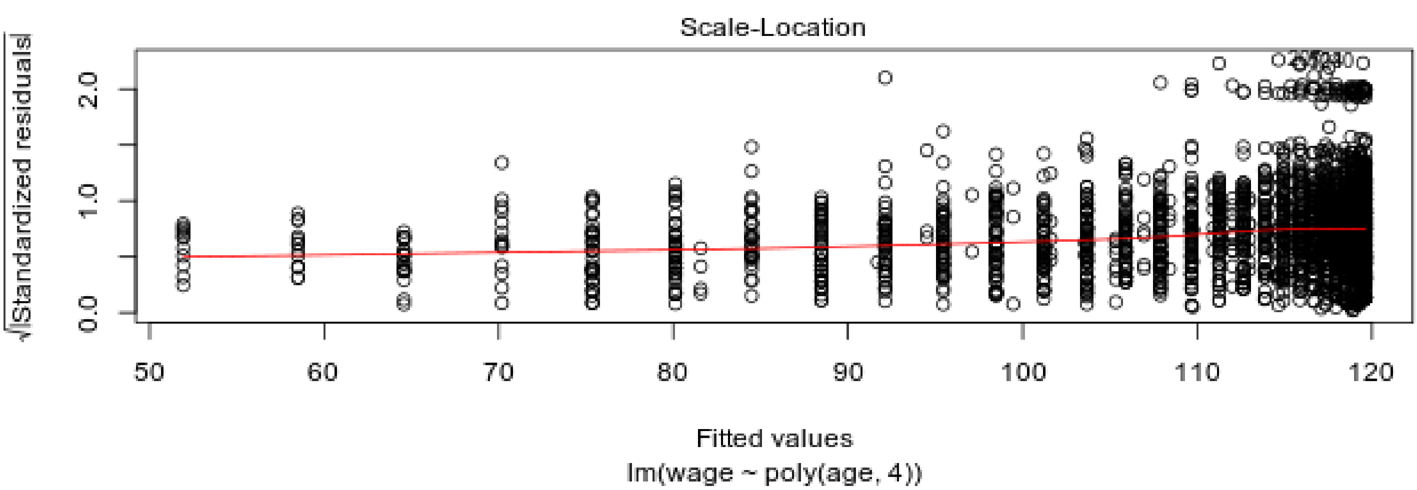
```
Call:
lm(formula = wage ~ poly(age, 4), data = wage)

Residuals:
```

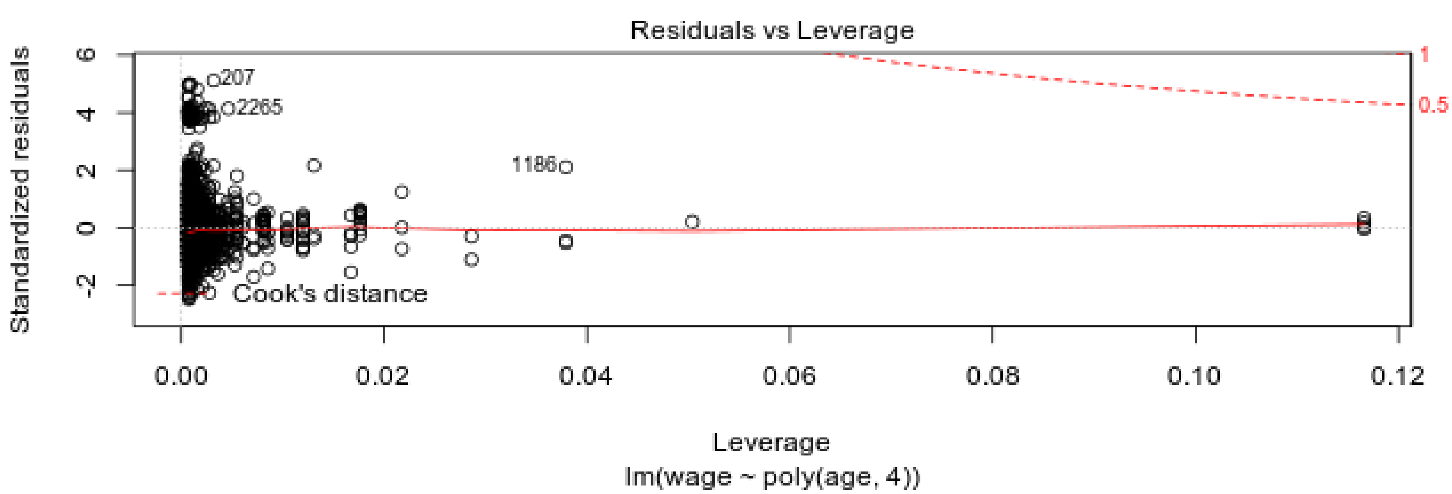
```
    Min      1Q  Median      3Q     Max
-98.707 -24.626  -4.993  15.217 203.693

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    111.7036     0.7287 153.283  < 2e-16 ***
poly(age, 4)1  447.0679    39.9148  11.201  < 2e-16 ***
poly(age, 4)2 -478.3158    39.9148 -11.983  < 2e-16 ***
poly(age, 4)3  125.5217    39.9148   3.145  0.00168 **
poly(age, 4)4  -77.9112    39.9148  -1.952  0.05104 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 39.91 on 2995 degrees of freedom
Multiple R-squared:  0.08626,   Adjusted R-squared:  0.08504
F-statistic: 70.69 on 4 and 2995 DF,  p-value: < 2.2e-16
```

```
plot(fit)
```




Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(wage ~ poly(age, 4))


Scale-Location
√|Standardized residuals|
Fitted values
lm(wage ~ poly(age, 4))


Residuals vs Leverage
Standardized residuals
207
2265
1186
Cook's distance
1
0.5
Leverage
lm(wage ~ poly(age, 4))

```r
coef(summary(fit))
```

```
                 Estimate Std. Error    t value     Pr(>|t|)
(Intercept)    111.70361  0.7287409 153.283015 0.000000e+00
poly(age, 4)1  447.06785 39.9147851  11.200558 1.484604e-28
poly(age, 4)2 -478.31581 39.9147851 -11.983424 2.355831e-32
poly(age, 4)3  125.52169 39.9147851   3.144742 1.678622e-03
poly(age, 4)4  -77.91118 39.9147851  -1.951938 5.103865e-02
```

```r
fit2 <- lm(wage ~ poly(age, 4, raw = T), data = wage)
coef(summary(fit2))
```

```
                            Estimate   Std. Error   t value     Pr(>|t|)
(Intercept)            -1.841542e+02 6.004038e+01 -3.067172 0.0021802539
poly(age, 4, raw = T)1  2.124552e+01 5.886748e+00  3.609042 0.0003123618
poly(age, 4, raw = T)2 -5.638593e-01 2.061083e-01 -2.735743 0.0062606446
poly(age, 4, raw = T)3  6.810688e-03 3.065931e-03  2.221409 0.0263977518
poly(age, 4, raw = T)4 -3.203830e-05 1.641359e-05 -1.951938 0.0510386498
```

Alternative:

```r
fit2a <- lm(wage ~ age + I(age^2) + I(age^3) + I(age^4), data = wage)
coef(summary(fit2a))
```

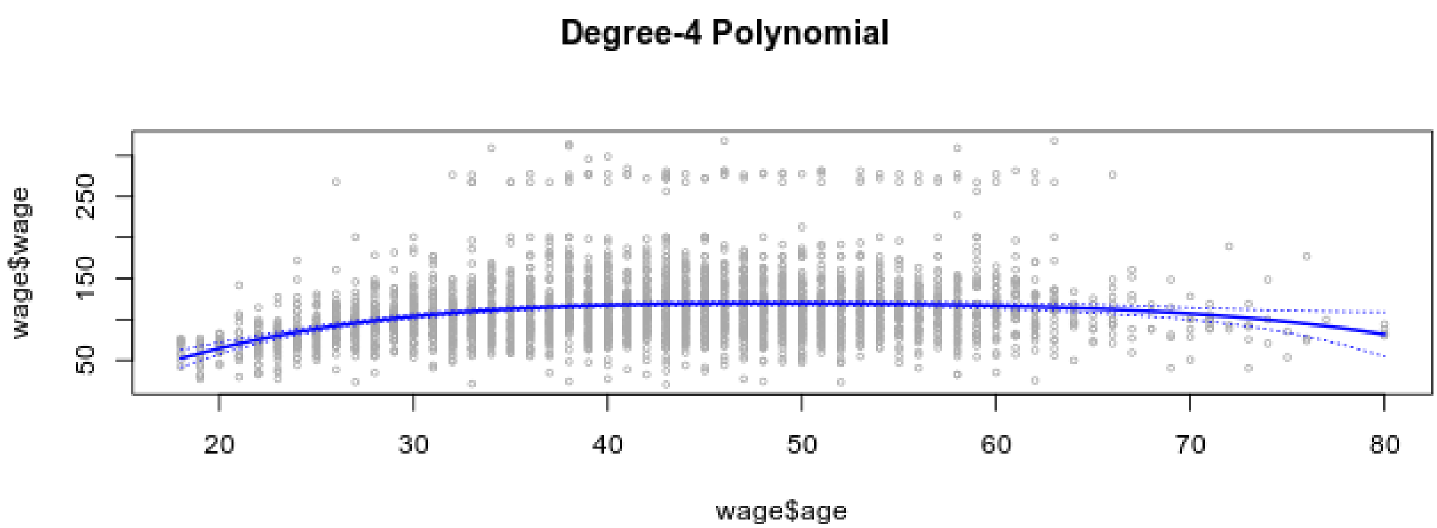
```
                 Estimate   Std. Error   t value     Pr(>|t|)
(Intercept) -1.841542e+02 6.004038e+01 -3.067172 0.0021802539
age          2.124552e+01 5.886748e+00  3.609042 0.0003123618
I(age^2)    -5.638593e-01 2.061083e-01 -2.735743 0.0062606446
I(age^3)     6.810688e-03 3.065931e-03  2.221409 0.0263977518
I(age^4)    -3.203830e-05 1.641359e-05 -1.951938 0.0510386498
```

```r
fit2b <- lm(wage ~ cbind(age, age^2, age^3, age^4), data = wage)
coef(fit2b)
```

```
                          (Intercept) cbind(age, age^2, age^3, age^4)age
                        -1.841542e+02                       2.124552e+01
   cbind(age, age^2, age^3, age^4)    cbind(age, age^2, age^3, age^4)
                        -5.638593e-01                       6.810688e-03
   cbind(age, age^2, age^3, age^4)
                        -3.203830e-05
```

```r
agelims <- range(wage$age)
age.grid <- seq(from = agelims[1], to = agelims[2])

pred <- predict(fit, newdata = list(age = age.grid), se = T)

se.bands <- cbind(pred$fit + 2*pred$se.fit, pred$fit - 2*pred$se.fit)
```

```
par(mfrow = c(1, 1), mar = c(4.5, 4.5, 1, 1), oma = c(0, 0, 4, 0))
plot(wage$age, wage$wage, xlim = agelims, cex = .5, col = "darkgrey")
title("Degree-4 Polynomial", outer = T)
lines(age.grid, pred$fit, lwd = 2, col = "blue")
matlines(age.grid, se.bands, lwd = 1, col = "blue", lty = 3)
```




```
pred2 <- predict(fit2, newdata = list(age = age.grid), se = T)
max(abs(pred$fit - pred2$fit))
```

```
[1] 7.81597e-11
```

```
fit1 <- lm(wage ~ age, data = wage)
fit2 <- lm(wage ~ poly(age, 2), data = wage)
fit3 <- lm(wage ~ poly(age, 3), data = wage)
fit4 <- lm(wage ~ poly(age, 4), data = wage)
fit5 <- lm(wage ~ poly(age, 5), data = wage)

anova(fit1, fit2, fit3, fit4, fit5)
```

```
Analysis of Variance Table

Model 1: wage ~ age
Model 2: wage ~ poly(age, 2)
Model 3: wage ~ poly(age, 3)
Model 4: wage ~ poly(age, 4)
Model 5: wage ~ poly(age, 5)
  Res.Df     RSS Df Sum of Sq        F    Pr(>F)
1   2998 5022216
2   2997 4793430  1    228786 143.5931 < 2.2e-16 ***
3   2996 4777674  1     15756   9.8888  0.001679 **
```

```
4   2995 4771604  1      6070   3.8098  0.051046 .
5   2994 4770322  1      1283   0.8050  0.369682
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
coef(summary(fit5))
```

```
                Estimate Std. Error     t value     Pr(>|t|)
(Intercept)    111.70361  0.7287647 153.2780243 0.000000e+00
poly(age, 5)1  447.06785 39.9160847  11.2001930 1.491111e-28
poly(age, 5)2 -478.31581 39.9160847 -11.9830341 2.367734e-32
poly(age, 5)3  125.52169 39.9160847   3.1446392 1.679213e-03
poly(age, 5)4  -77.91118 39.9160847  -1.9518743 5.104623e-02
poly(age, 5)5  -35.81289 39.9160847  -0.8972045 3.696820e-01
```

```r
fit1 <- lm(wage ~ education + age, data = wage)
fit2 <- lm(wage ~ education + poly(age, 2), data = wage)
fit3 <- lm(wage ~ education + poly(age, 3), data = wage)

anova(fit1, fit2, fit3)
```

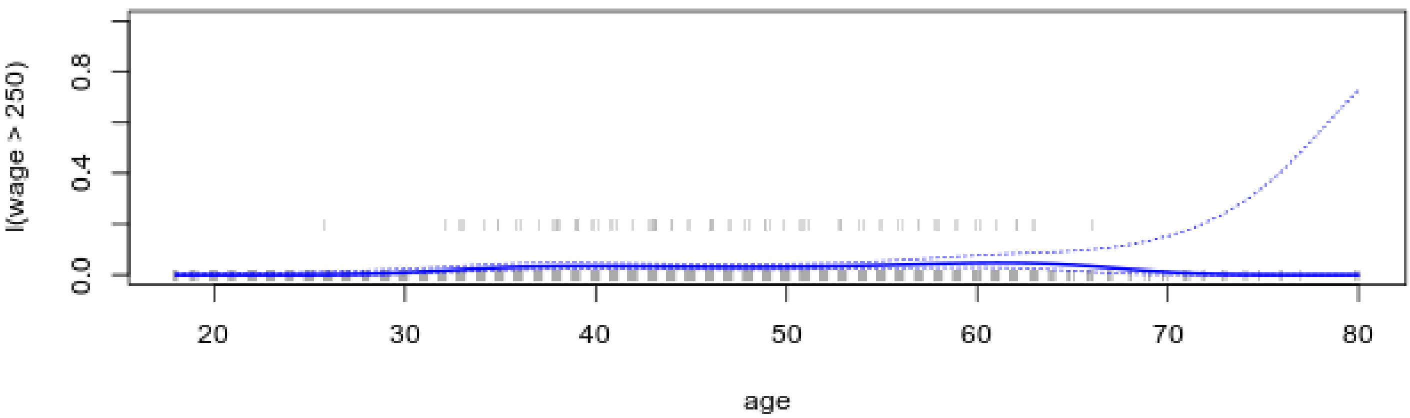
```
Analysis of Variance Table

Model 1: wage ~ education + age
Model 2: wage ~ education + poly(age, 2)
Model 3: wage ~ education + poly(age, 3)
  Res.Df     RSS Df Sum of Sq        F Pr(>F)
1   2994 3867992
2   2993 3725395  1    142597 114.6969 <2e-16 ***
3   2992 3719809  1      5587   4.4936 0.0341 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fit <- glm(I(wage > 250) ~ poly(age, 4), data = wage, family = "binomial")

pred <- predict(fit, newdata = list(age = age.grid), se = T)

pfit <- exp(pred$fit) / (1 + exp(pred$fit))

se.bands.logit <- cbind(pred$fit + 2 * pred$se.fit, pred$fit - 2*pred$se.fit)

se.bands <- exp(se.bands.logit) / (1 + exp(se.bands.logit))
```

Alternatively:

```r
pred <- predict(fit, newdata = list(age = age.grid), type = "response", se = T)
```

```r
with(wage, {
   plot(age, I(wage > 250), xlim = agelims, type = "n")
   points(jitter(age), I((wage > 250)/5), cex = .5, pch = "|", col = "darkgrey")
   lines(age.grid, pfit, lwd = 2, col = "blue")
   matlines(age.grid, se.bands, lwd = 1, col = "blue", lty = 3)
})
```




```r
table(cut(wage$age, 4))
```

```
(17.9,33.5]   (33.5,49]   (49,64.5] (64.5,80.1]
        750        1399         779          72
```

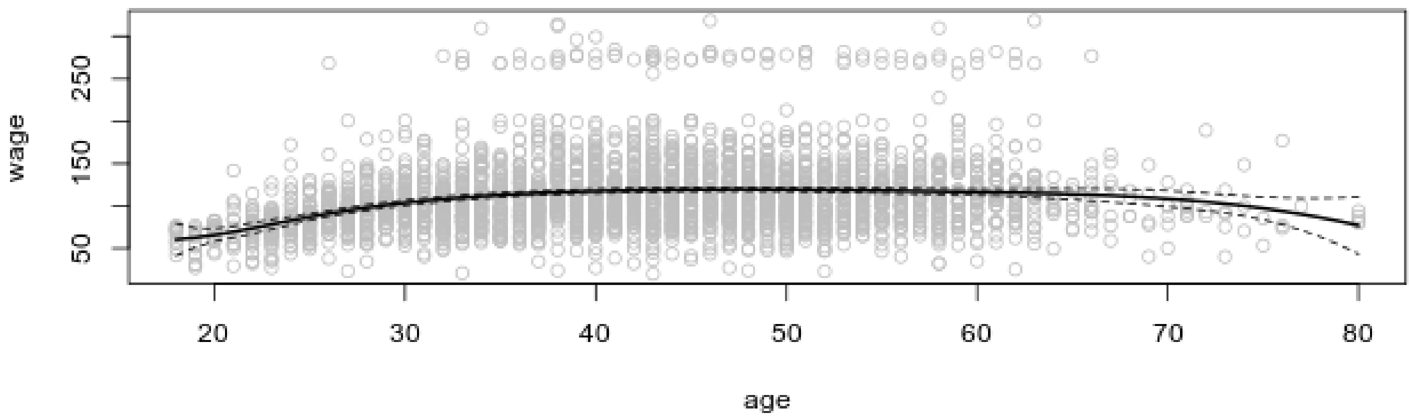
```r
fit <- lm(wage ~ cut(age, 4), data = wage)
coef(summary(fit))
```

```
                        Estimate Std. Error   t value     Pr(>|t|)
(Intercept)            94.158392   1.476069 63.789970 0.000000e+00
cut(age, 4)(33.5,49]   24.053491   1.829431 13.148074 1.982315e-38
cut(age, 4)(49,64.5]   23.664559   2.067958 11.443444 1.040750e-29
cut(age, 4)(64.5,80.1]  7.640592   4.987424  1.531972 1.256350e-01
```

Splines

```r
fit <- lm(wage ~ bs(age, knots = c(25, 40, 60)), data = wage)

pred <- predict(fit, newdata = list(age = age.grid), se = T)

with(wage, {
```

```
   plot(age, wage, col = "gray")
   lines(age.grid, pred$fit, lwd=2)
   lines(age.grid, pred$fit+2*pred$se.fit, lty="dashed")
   lines(age.grid, pred$fit-2*pred$se.fit, lty="dashed")
})
```




```
dim(bs(wage$age, knots = c(25, 40, 60)))
```

```
[1] 3000    6
```

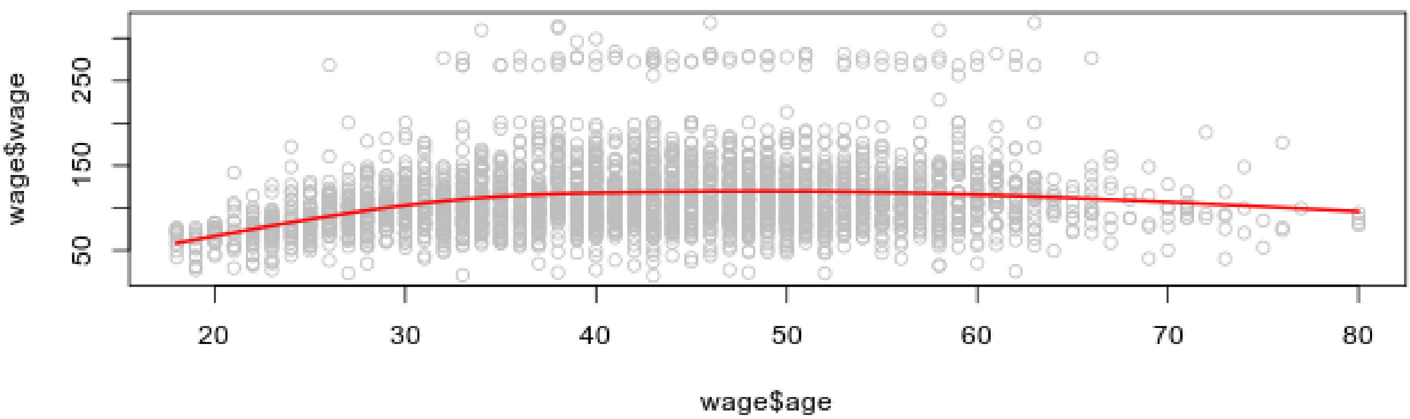
```
dim(bs(wage$age, df = 6))
```

```
[1] 3000    6
```

```
attr(bs(wage$age, df = 6), "knots")
```

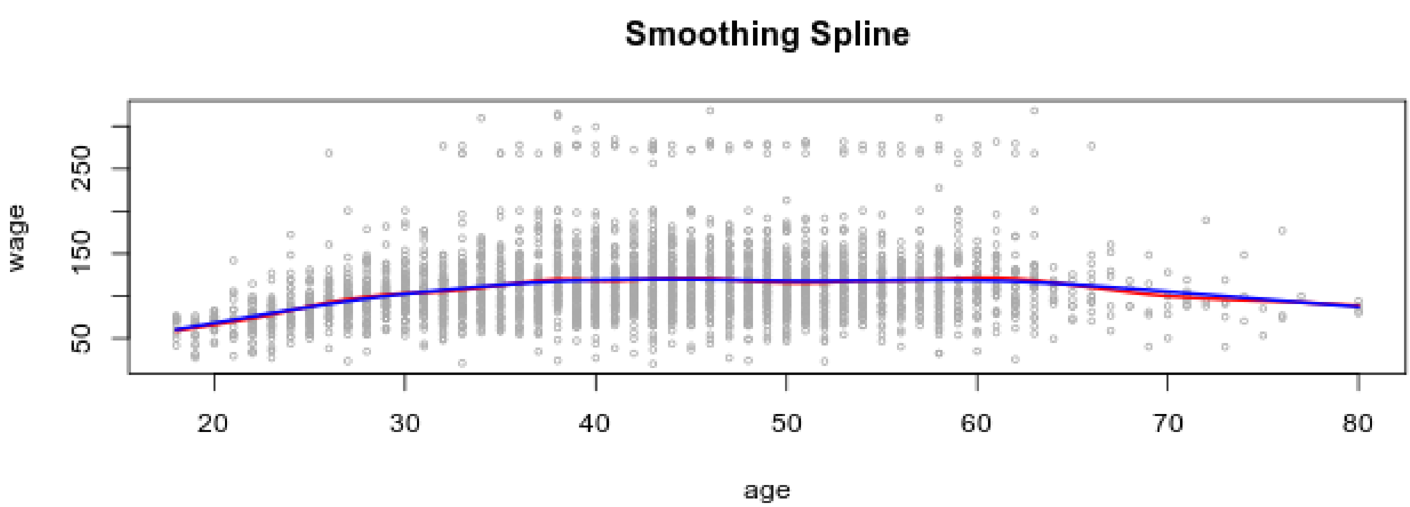
```
  25%   50%   75% 
33.75 42.00 51.00 
```

```
fit2 <- lm(wage ~ ns(age, df = 4), data = wage)
pred2 <- predict(fit2, newdata =  list(age = age.grid), se = T)
par(mfrow=c(1,1))
plot(wage$age, wage$wage, col = "gray")
lines(age.grid, pred2$fit, col = "red", lwd = 2)
```

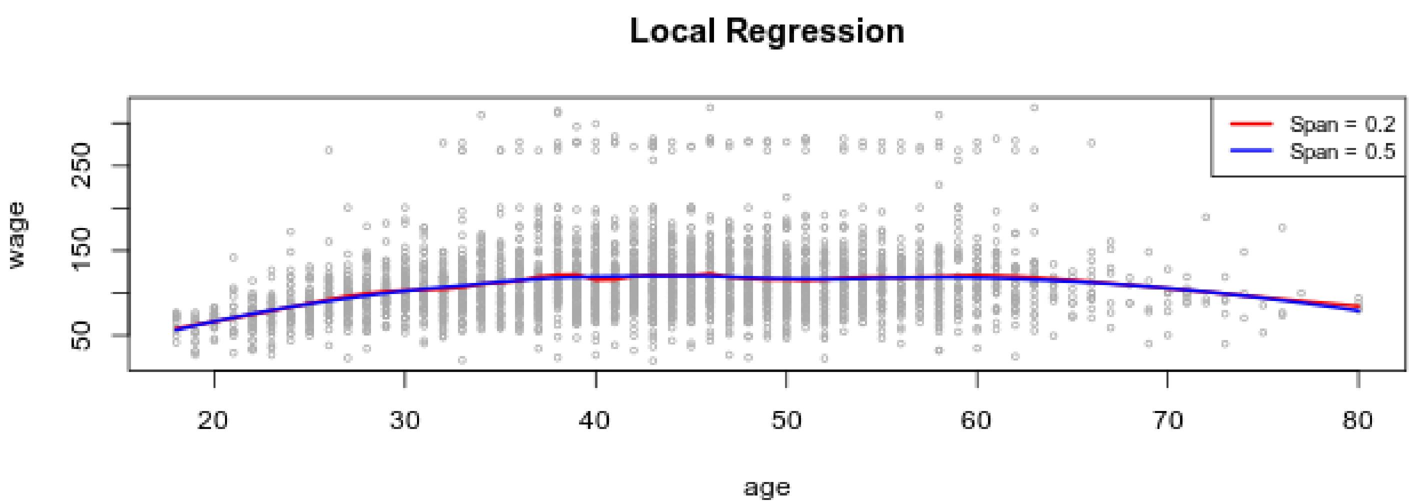



```
with(wage,{
   plot(age, wage, xlim = agelims, cex = .5, col = "darkgrey")
   title("Smoothing Spline")
   fit <- smooth.spline(age, wage, df = 16)
   fit2 <- smooth.spline(age, wage, cv = T)

   lines(fit, col = "red", lwd = 2)
   lines(fit2, col = "blue", lwd = 2)
})
```

```
Warning in smooth.spline(age, wage, cv = T): cross-validation with non-unique
'x' values seems doubtful
```



```r
with(wage, {
  plot(age, wage, xlim = agelims, cex = .5, col = "darkgrey")
  title("Local Regression")
  fit <- loess(wage ~ age, span = .2)
  fit2 <- loess(wage ~ age, span = .5)
  lines(age.grid, predict(fit, data.frame(age = age.grid)), col = "red", lwd = 2)
  lines(age.grid, predict(fit2, data.frame(age = age.grid)), col = "blue", lwd = 2)
  legend("topright", legend = c("Span = 0.2", "Span = 0.5"), col = c("red", "blue"), lty = 1,
})
```

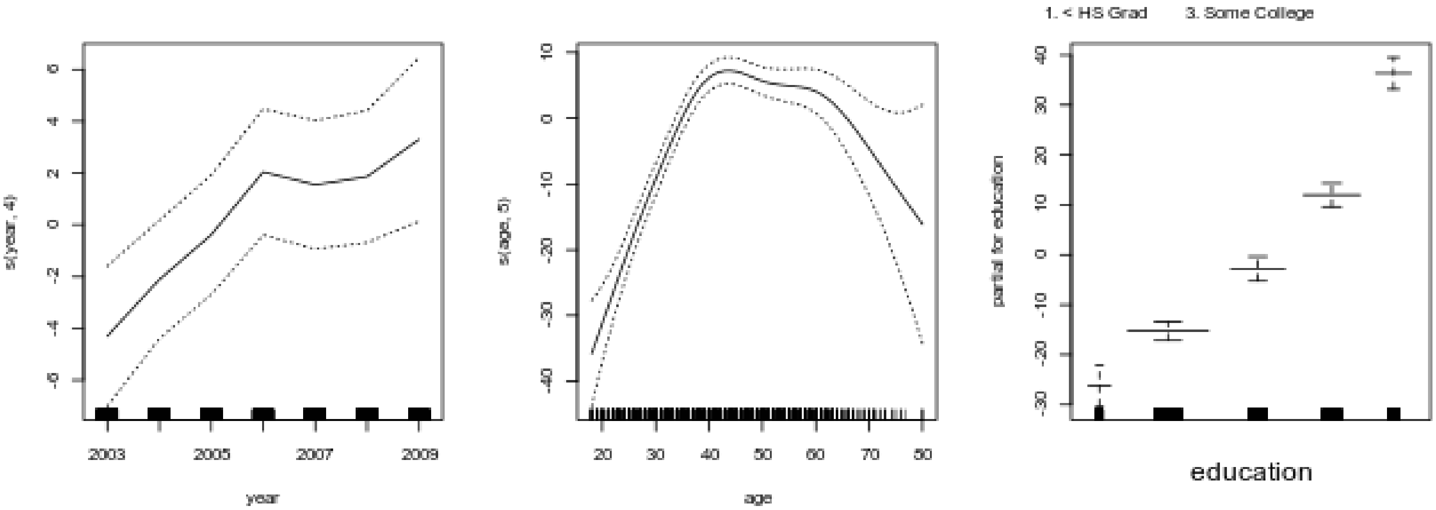


GAMs

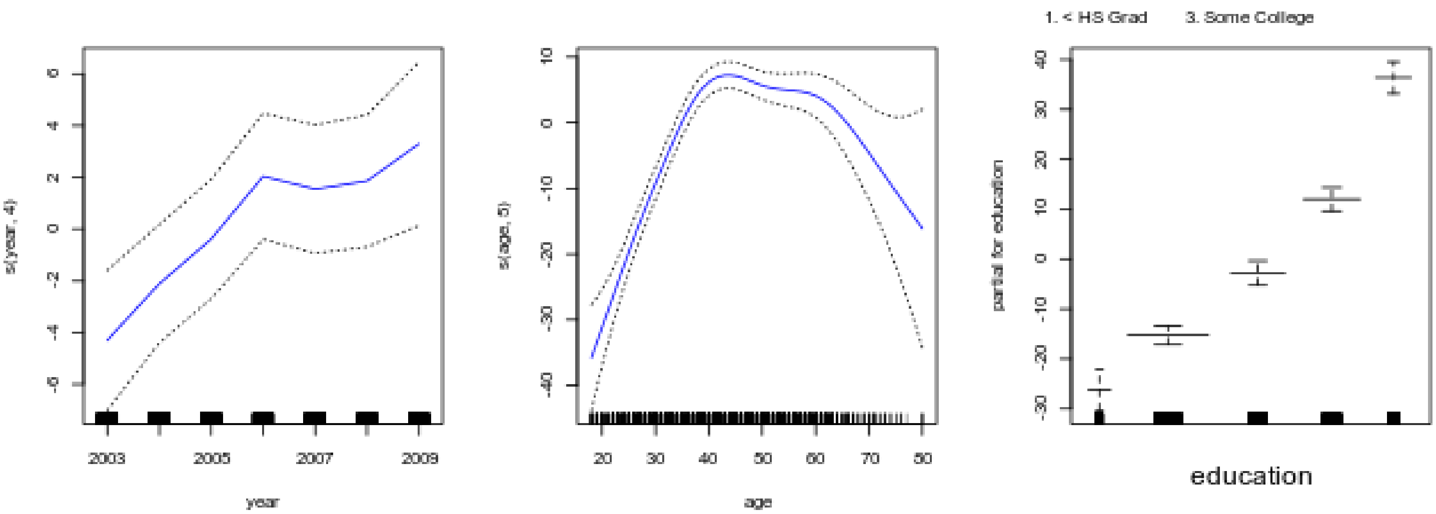
```r
gam1 <- lm(wage ~ ns(year, 4) + ns(age, 5) + education, data = wage)

gam.m3 <- gam(wage ~ s(year, 4) + s(age, 5) + education, data = wage)
```

```
Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
ignored
```

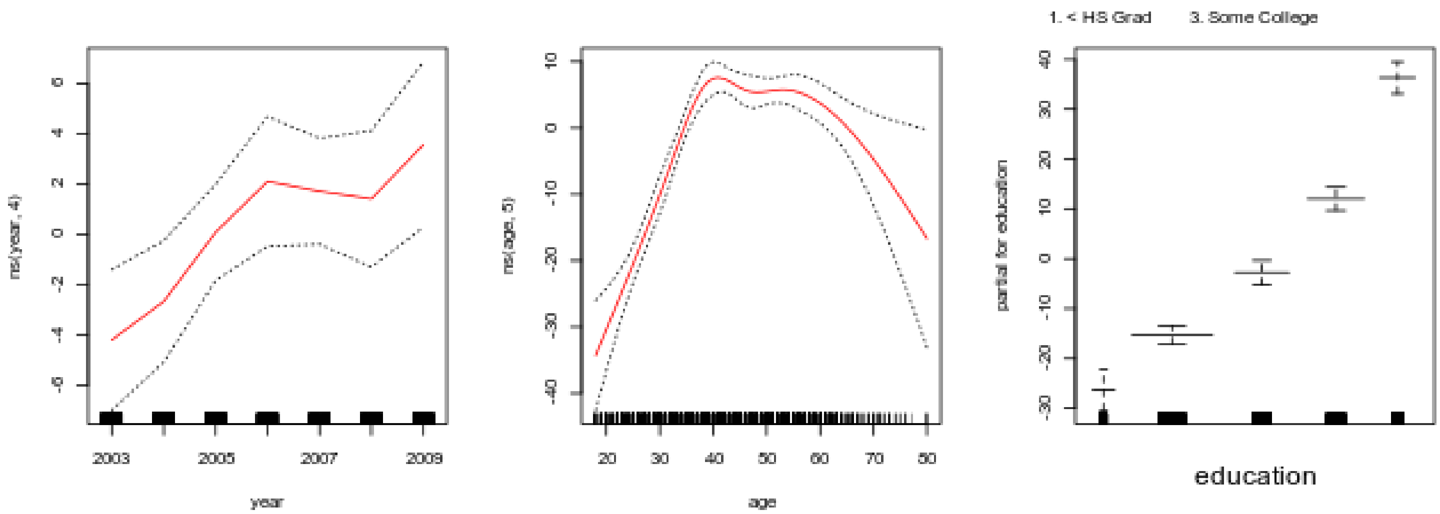
```r
par(mfrow = c(1, 3))
plot.Gam(gam.m3, se = T)
```




```r
par(mfrow = c(1, 3))
plot(gam.m3, se = T, col = "blue")
```




```r
plot.Gam(gam1, se = T, col = "red")
```




```r
gam.m1 <- gam(wage ~ s(age, 5) + education, data = wage)
```

```
Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
ignored
```

```r
gam.m2 <- gam(wage ~ year + s(age, 5) + education, data = wage)
```

```
Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
ignored
```

```r
anova(gam.m1, gam.m2, gam.m3)
```

```
Analysis of Deviance Table

Model 1: wage ~ s(age, 5) + education
Model 2: wage ~ year + s(age, 5) + education
Model 3: wage ~ s(year, 4) + s(age, 5) + education
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      2990    3711731
2      2989    3693842  1  17889.2 0.0001419 ***
3      2986    3689770  3   4071.1 0.3483897
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(gam.m3)
```

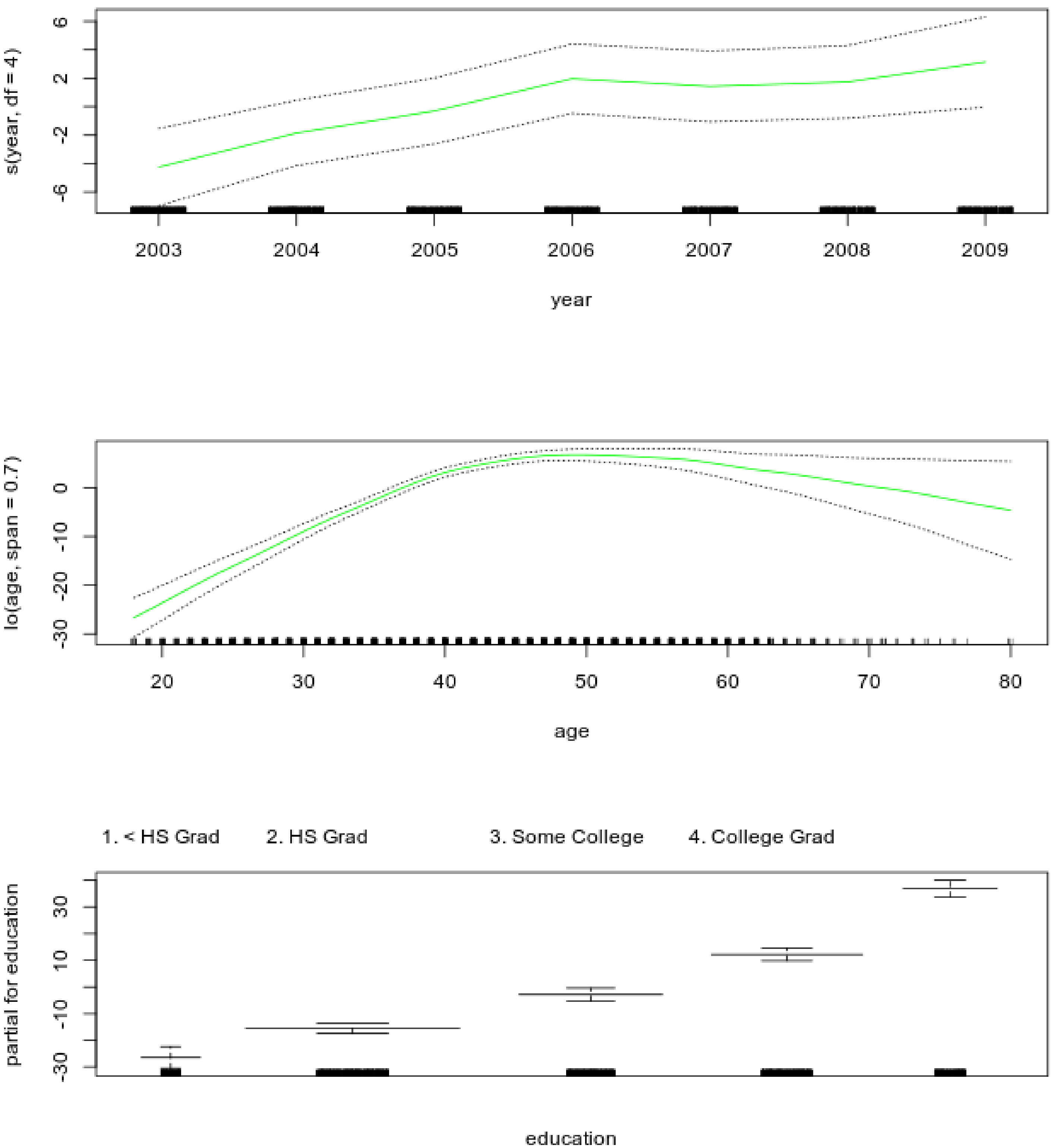
```

Call: gam(formula = wage ~ s(year, 4) + s(age, 5) + education, data = wage)
Deviance Residuals:
    Min      1Q  Median      3Q     Max
-119.43  -19.70   -3.33   14.17  213.48
```

```
(Dispersion Parameter for gaussian family taken to be 1235.69)

    Null Deviance: 5222086 on 2999 degrees of freedom
Residual Deviance: 3689770 on 2986 degrees of freedom
AIC: 29887.75

Number of Local Scoring Iterations: 2

Anova for Parametric Effects
             Df  Sum Sq Mean Sq F value    Pr(>F)
s(year, 4)    1   27162   27162  21.981 2.877e-06 ***
s(age, 5)     1  195338  195338 158.081 < 2.2e-16 ***
education     4 1069726  267432 216.423 < 2.2e-16 ***
Residuals  2986 3689770    1236
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Anova for Nonparametric Effects
            Npar Df Npar F  Pr(F)
(Intercept)
s(year, 4)        3  1.086 0.3537
s(age, 5)         4 32.380 <2e-16 ***
education
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

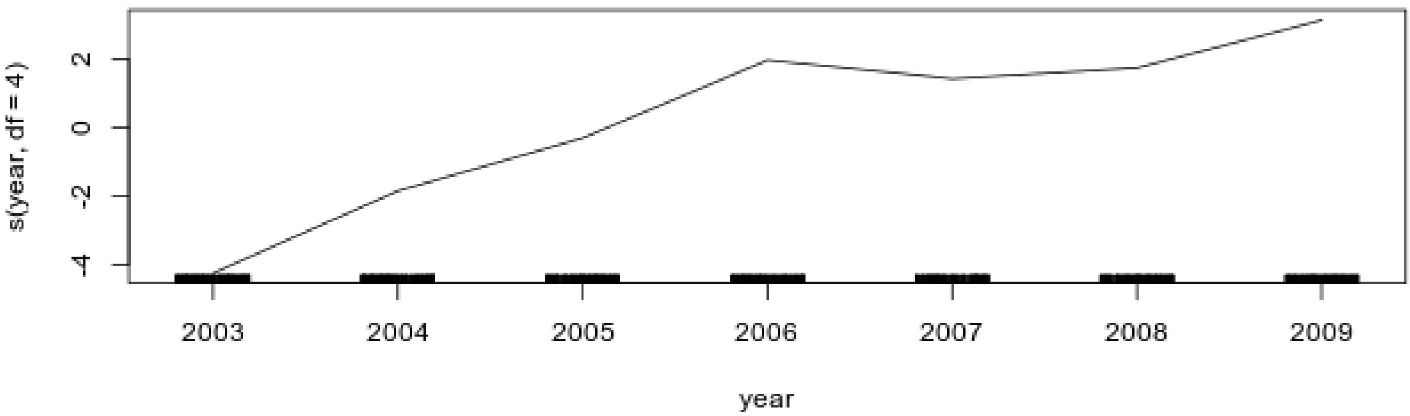
```r
pred <- predict(gam.m2, newdata = wage)

gam.lo <- gam(wage ~ s(year, df = 4) + lo(age, span = 0.7) + education, data = wage)
```

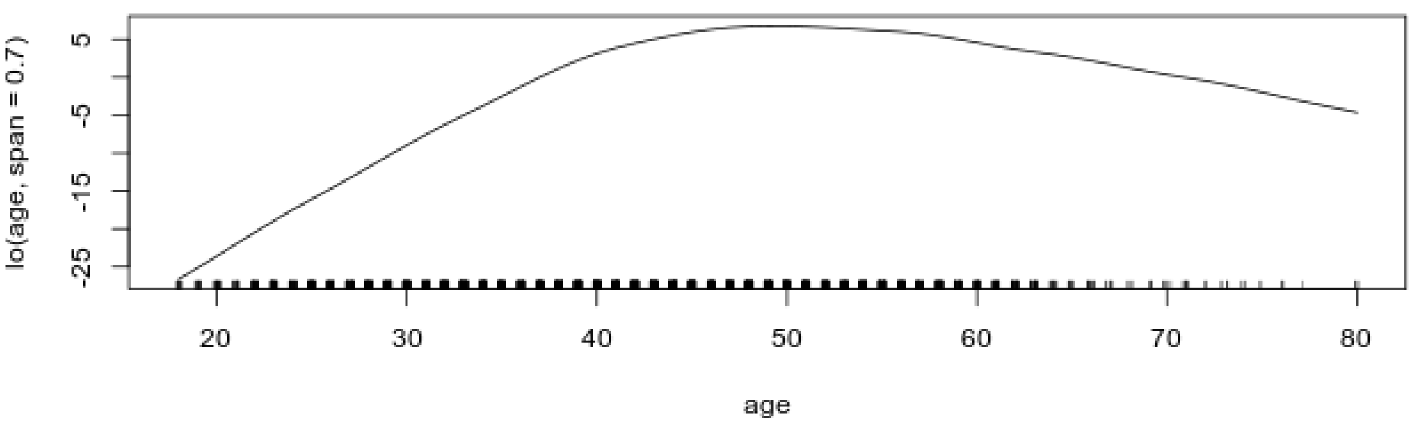
```
Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
ignored
```

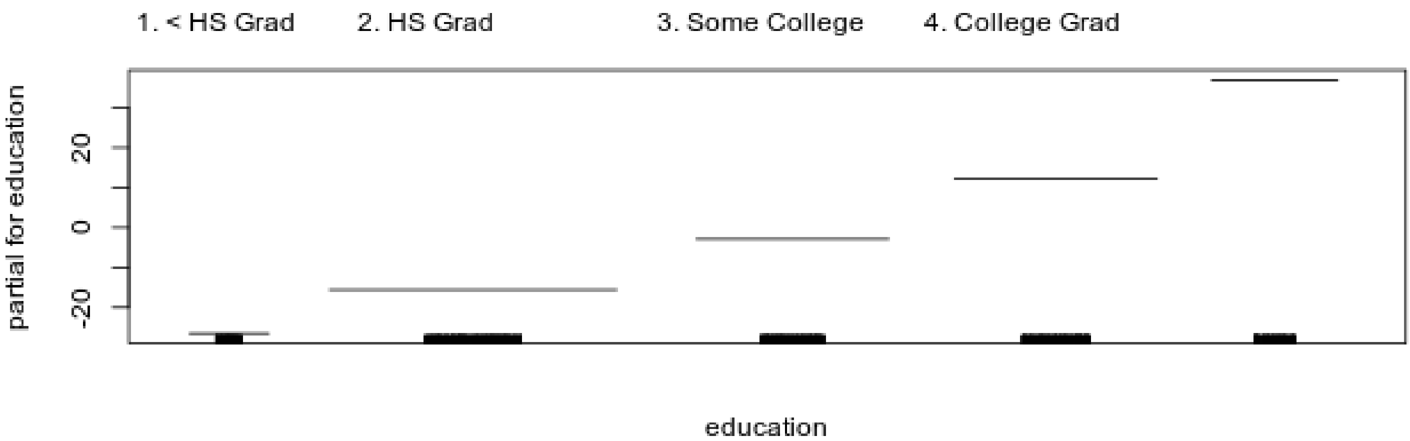
```r
plot.Gam(gam.lo, se = T, col = "green")
```


s(year, df = 4)
year
lo(age, span = 0.7)
age
1. < HS Grad
2. HS Grad
3. Some College
4. College Grad
partial for education
education

```r
plot(gam.lo)
```


s(year, df = 4)
-4
-2
0
2
2003
2004
2005
2006
2007
2008
2009
year

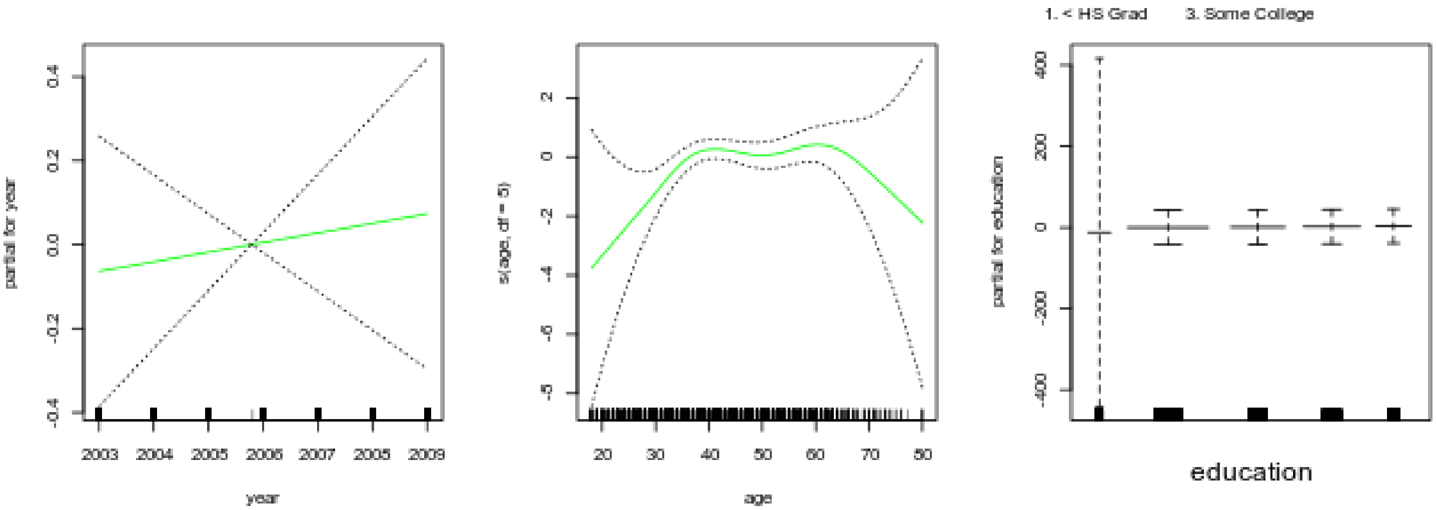

lo(age, span = 0.7)
-25
-15
-5
5
20
30
40
50
60
70
80
age




```r
gam.lr <- gam(I(wage > 250) ~ year + s(age, df = 5) + education, family = binomial, data = wage
```

```
Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
ignored
```

```r
par(mfrow = c(1, 3))
plot(gam.lr, se = T, col = "green")
```




```r
table(wage$education, I(wage$wage > 250))
```

```
                   FALSE TRUE
  1. < HS Grad       268    0
  2. HS Grad         966    5
  3. Some College    643    7
```

```
  4. College Grad       663   22
  5. Advanced Degree    381   45
```

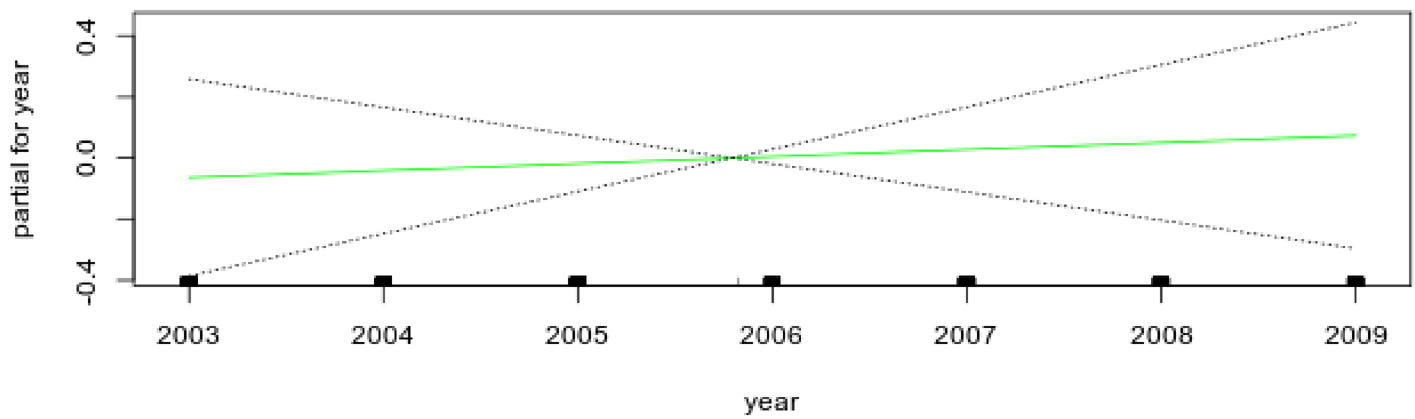
```r
levels(wage$education)
```

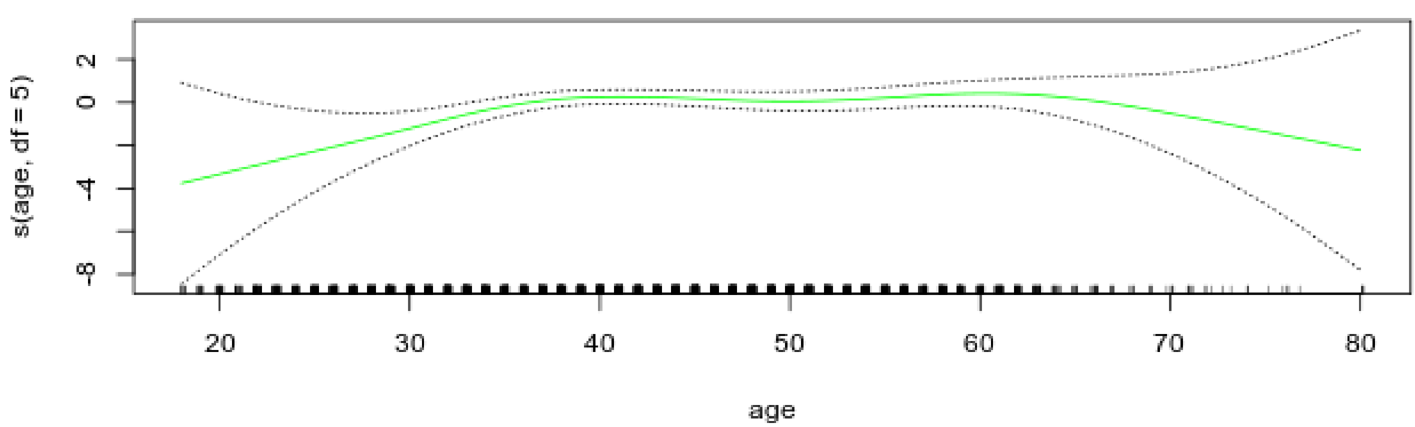
```
[1] "1. < HS Grad"       "2. HS Grad"          "3. Some College"
[4] "4. College Grad"    "5. Advanced Degree"
```

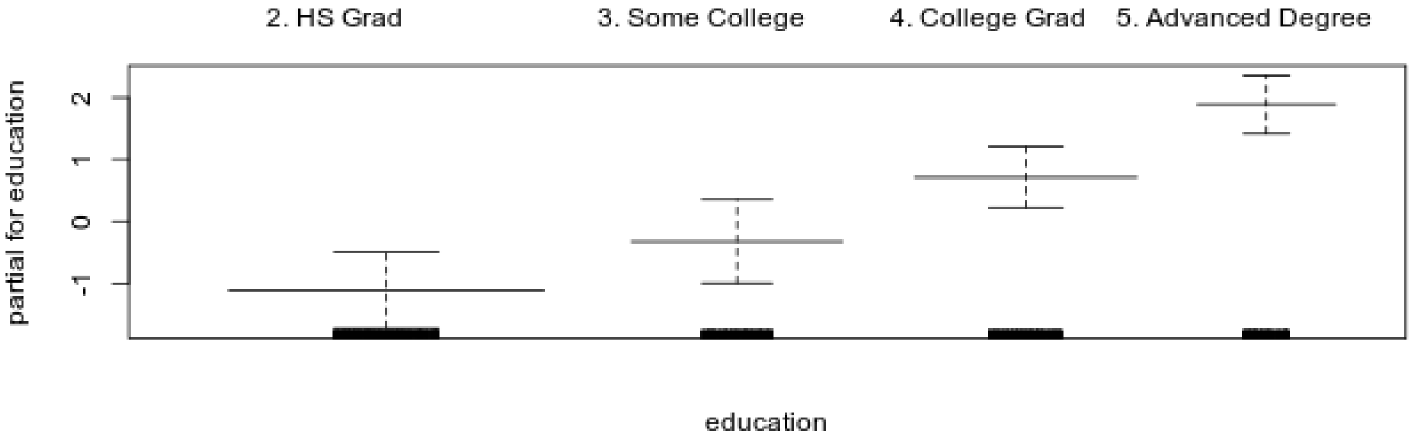
```r
gam.lr.s <- gam(I(wage > 250) ~ year + s(age, df = 5) + education, family = binomial, data = wa
```

```
Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
ignored
```

```r
plot(gam.lr.s, se = T, col = "green")
```







Applied

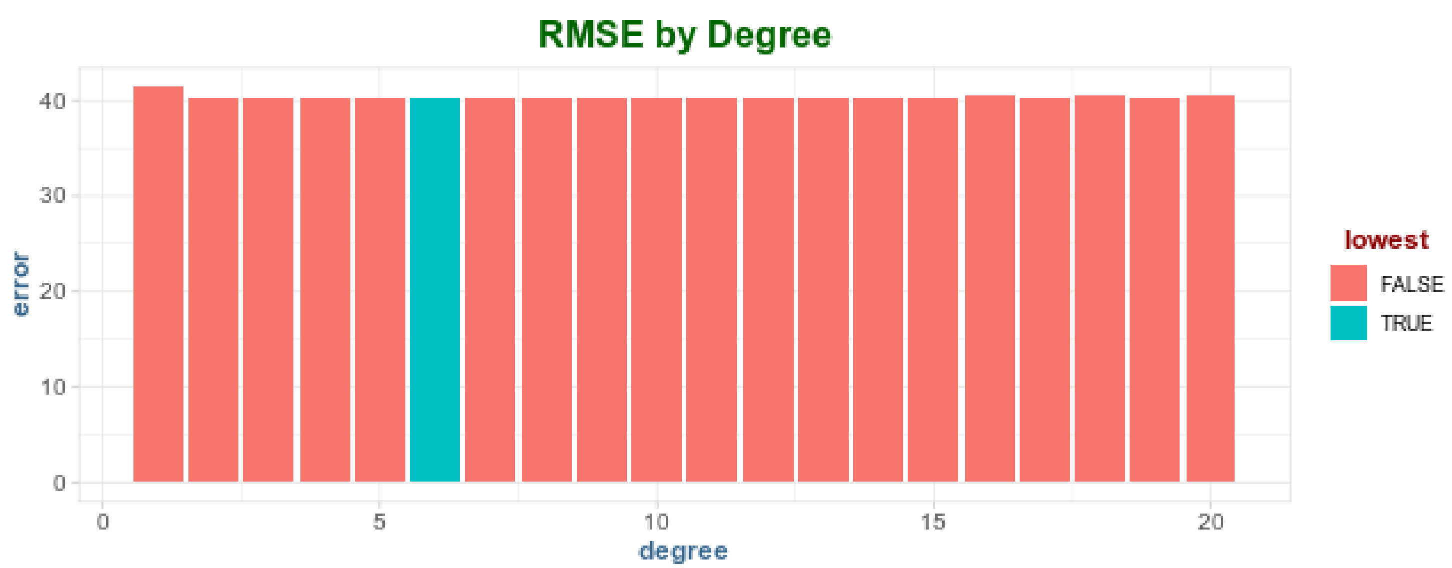
In this exercise, you will further analyze the wage data set considered throughout this chapter.

```
test.size <- .7
index <- sample(nrow(wage), nrow(wage) * test.size, replace = F)

train <- wage[index]
test <- wage[!index]
```

a.) Perform polynomial regression to predict wage using age. Use cross-validation to select the optimal degree d for the polynomial. What degree was chosen, and how does this compare to the result of hypothesis testing using ANOVA? Make a plot of the fit obtained.

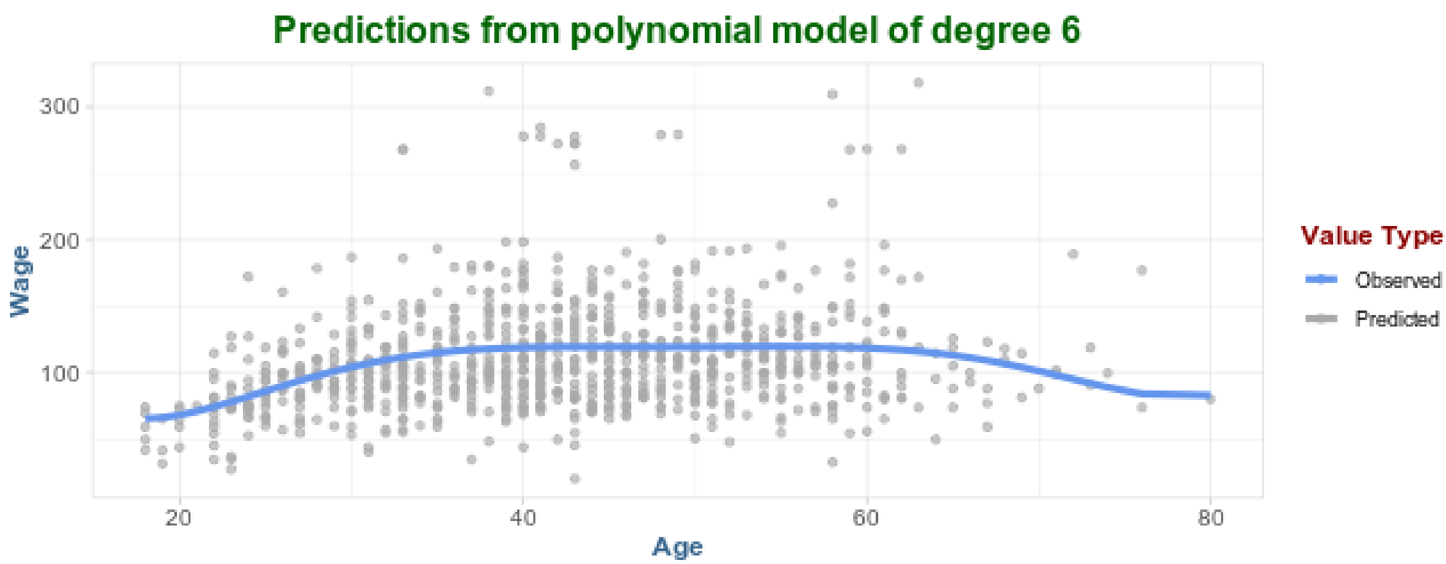
```
degree <- 20; folds = 10
cv.errors <- numeric(degree)

fold.size <- nrow(train) / folds

for(deg in 1:degree)
{
   # 10 fold cv
   errors <- numeric(folds)
   for(fold in 1:folds)
   {
      holdout <- seq((fold - 1) * fold.size, fold * fold.size)

      cv.train <- train[!holdout]
      cv.test <- train[holdout]
```

```
      fit <- lm(wage ~ poly(age, deg), data = cv.train)

      pred <- predict(fit, newdata = cv.test, type = "response")

      errors[fold] <- sqrt(mean((cv.test$wage - pred)^2))
   }
   cv.errors[deg] <- mean(errors)
}

lowest.error <- which.min(cv.errors)

cv.results <- data.table(degree = 1:degree, error = cv.errors)[, lowest := degree == lowest.err

ggplot(cv.results, aes(degree, error, fill = lowest)) +
   geom_bar(stat = "identity") +
   labs(title = "RMSE by Degree")
```




```
model <- lm(wage ~ poly(age, lowest.error), data = train)

test %>%
    mutate(predictions = predict(model, test)) %>%
    ggplot(aes(age, wage, col = 'darkgrey')) +
    geom_point(alpha = .65) +
    geom_line(aes(age, predictions, col = 'cornflowerblue'), size = 1.5) +
    scale_color_manual(name = 'Value Type',
                       labels = c('Observed', 'Predicted'),
                       values = c('cornflowerblue', 'darkgrey' )) +
    labs(x = 'Age', y = 'Wage',
```

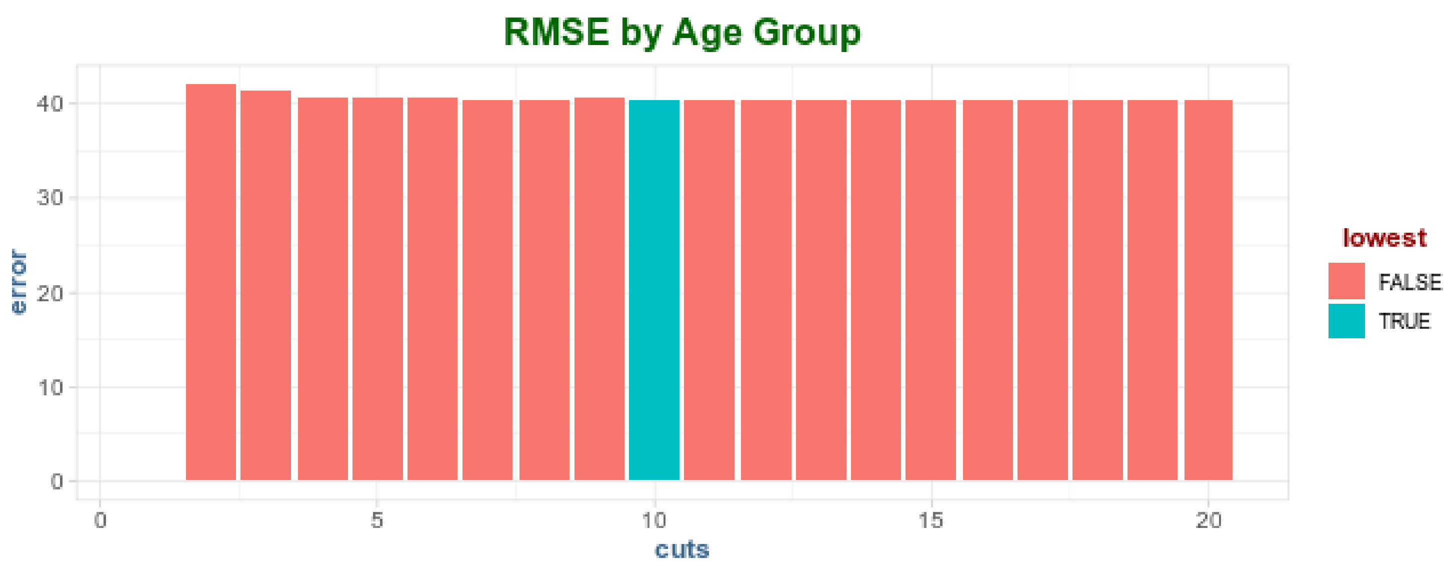
```
    title = paste0('Predictions from polynomial model of degree ', lowest.error))
```



b.) Fit a step function to predict wage using age, and perform cross-validation to choose the optimal number of cuts. Make a plot of the fit obtained.

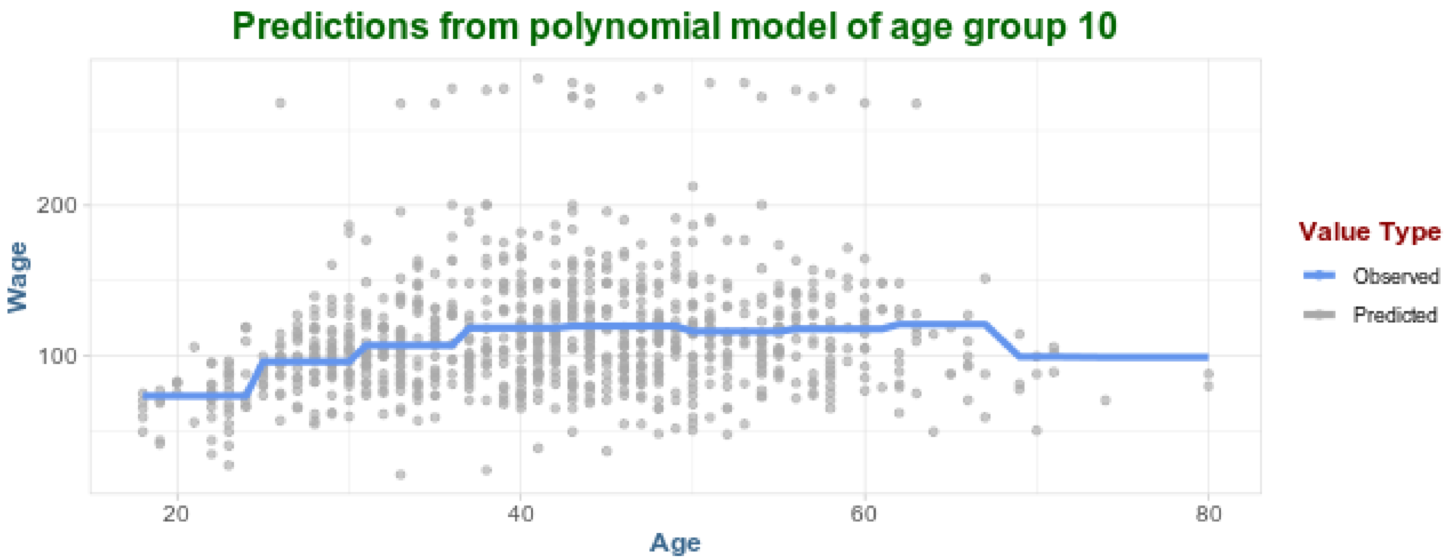
```
cuts <- 20; folds = 10
cv.errors <- numeric(degree)

fold.size <- nrow(train) / folds

for(cuts in 2:cuts)
{
   # 10 fold cv
   errors <- numeric(folds)

   # apply cut here so CV train/test have same levels
   train$AgeGroup <- cut(train$age, cuts)

   for(fold in 1:folds)
   {
      holdout <- seq((fold - 1) * fold.size, fold * fold.size)

      cv.train <- train[!holdout]
      cv.test <- train[holdout]

      fit <- lm(wage ~ I(AgeGroup), data = cv.train)

      pred <- predict(fit, newdata = cv.test, type = "response")
```

```
    errors[fold] <- sqrt(mean((cv.test$wage - pred)^2))
  }

  cv.errors[cuts] <- mean(errors)
}

lowest.error <- which.min(cv.errors[cv.errors != 0])

cv.results <- data.table(cuts = 1:cuts, error = cv.errors)[, lowest := cuts == lowest.error]

ggplot(cv.results, aes(cuts, error, fill = lowest)) +
  geom_bar(stat = "identity") +
  labs(title = "RMSE by Age Group")
```




```
wage.grouped <- wage
wage.grouped$AgeGroup <- cut(wage.grouped$age, lowest.error)

test.size <- .7
index <- sample(nrow(wage), nrow(wage) * test.size, replace = F)

train <- wage.grouped[index]
test <- wage.grouped[!index]

model <- lm(wage ~ I(AgeGroup), data = train)

test %>%
    mutate(predictions = predict(model, test)) %>%
    ggplot(aes(age, wage, col = 'darkgrey')) +
    geom_point(alpha = .65) +
```

```
  geom_line(aes(age, predictions, col = 'cornflowerblue'), size = 1.5) +
  scale_color_manual(name = 'Value Type',
                     labels = c('Observed', 'Predicted'),
                     values = c('cornflowerblue', 'darkgrey' )) +
  labs(x = 'Age', y = 'Wage',
       title = paste0('Predictions from polynomial model of age group ', lowest.error))
```



The wage data set contains a number of other features not explored in this chapter, such as marital status (*marit1*), job class (*jobclass*), and others. Explore the relationships between some of these other predictors and wage, and use non-linear fitting techniques in order to fit flexible models to the data. Create plots of the results obtained, and write a summary of your findings.