STAT 210
Applied Statistics and Data Analysis
Multiple Linear Regression 6
Polynomial Regression and Quantitative Regressors

Joaquin Ortega

Polynomial Regression

Polynomial Regression

In polynomial regression, the regressors associated with a predictor X_i form a polynomial in X_i with degree d.

In the case of only one regressor, the model is

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_d X^d + \epsilon$$

A particular case that occurs frequently is quadratic regression, where the polynomial has degree 2.

We have already seen that the `residualPlots()` function performs a curvature test to determine whether second-order terms should be included in the model.

Polynomial Regression

To illustrate polynomial regression, we follow an example in Fox and Weisberg, *An R Companion to Applied Regression*, SAGE (2019).

We use the data set `SLID` in the `carData` package '*which contains data for the province of Ontario from the 1994 wave of the Survey of Labour and Income Dynamics, a panel study of the Canadian labor force conducted by Statistics Canada.*'

```
str(SLID)
```

```
## 'data.frame':    7425 obs. of  5 variables:
##  $ wages    : num  10.6 11 NA 17.8 NA ...
##  $ education: num  15 13.2 16 14 8 16 12 14.5 15 10 ...
##  $ age      : int  40 19 49 46 71 50 70 42 31 56 ...
##  $ sex      : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 1 1 1 2 1 ...
##  $ language : Factor w/ 3 levels "English","French",..: 1 1 3 3 1 1 1 1 1 1
```

Polynomial Regression

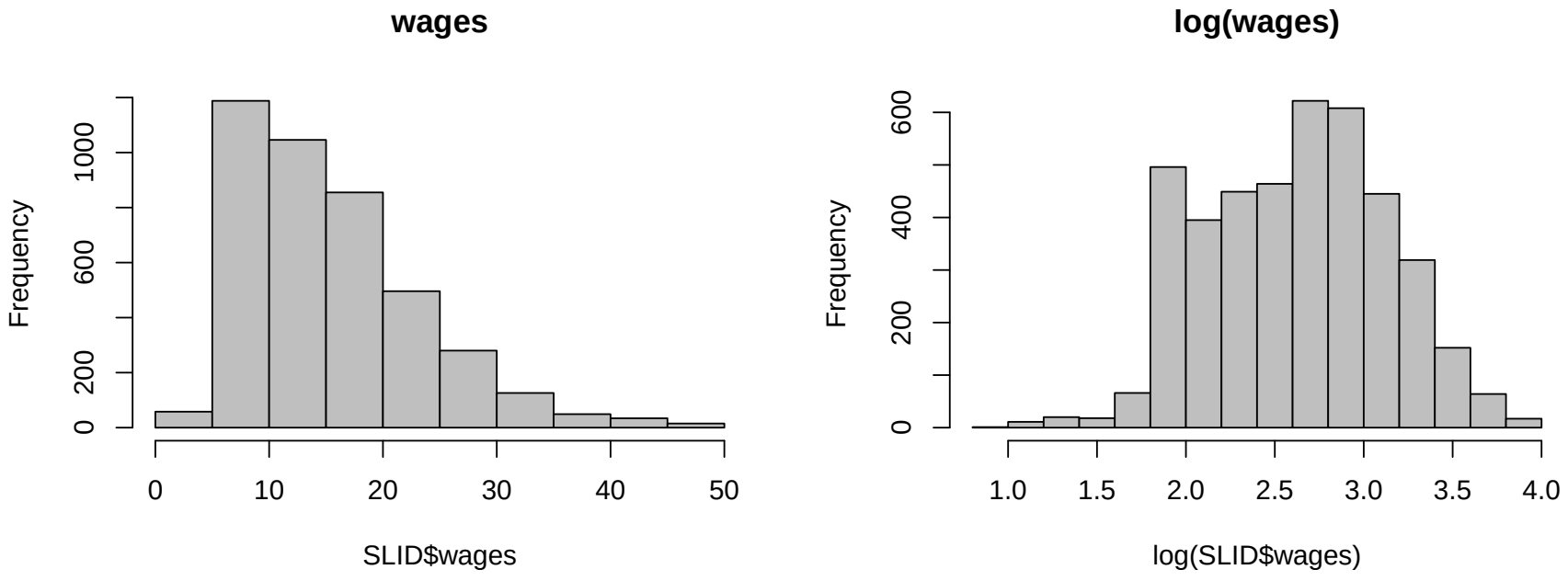
```
summary(SLID)
```

```
##      wages            education          age       
##  Min.   : 2.300   Min.   : 0.00   Min.   :16.00  
##  1st Qu.: 9.235   1st Qu.:10.30   1st Qu.:30.00  
##  Median :14.090   Median :12.10   Median :41.00  
##  Mean   :15.553   Mean   :12.50   Mean   :43.98  
##  3rd Qu.:19.800   3rd Qu.:14.53   3rd Qu.:57.00  
##  Max.   :49.920   Max.   :20.00   Max.   :95.00  
##  NA's   :3278     NA's   :249                    
##      sex           language  
##  Female:3880   English:5716  
##  Male  :3545   French : 497  
##                Other  :1091  
##                NA's   : 121  
##                              
##                              
##
```

NAs are not considered in the regression.

Polynomial Regression

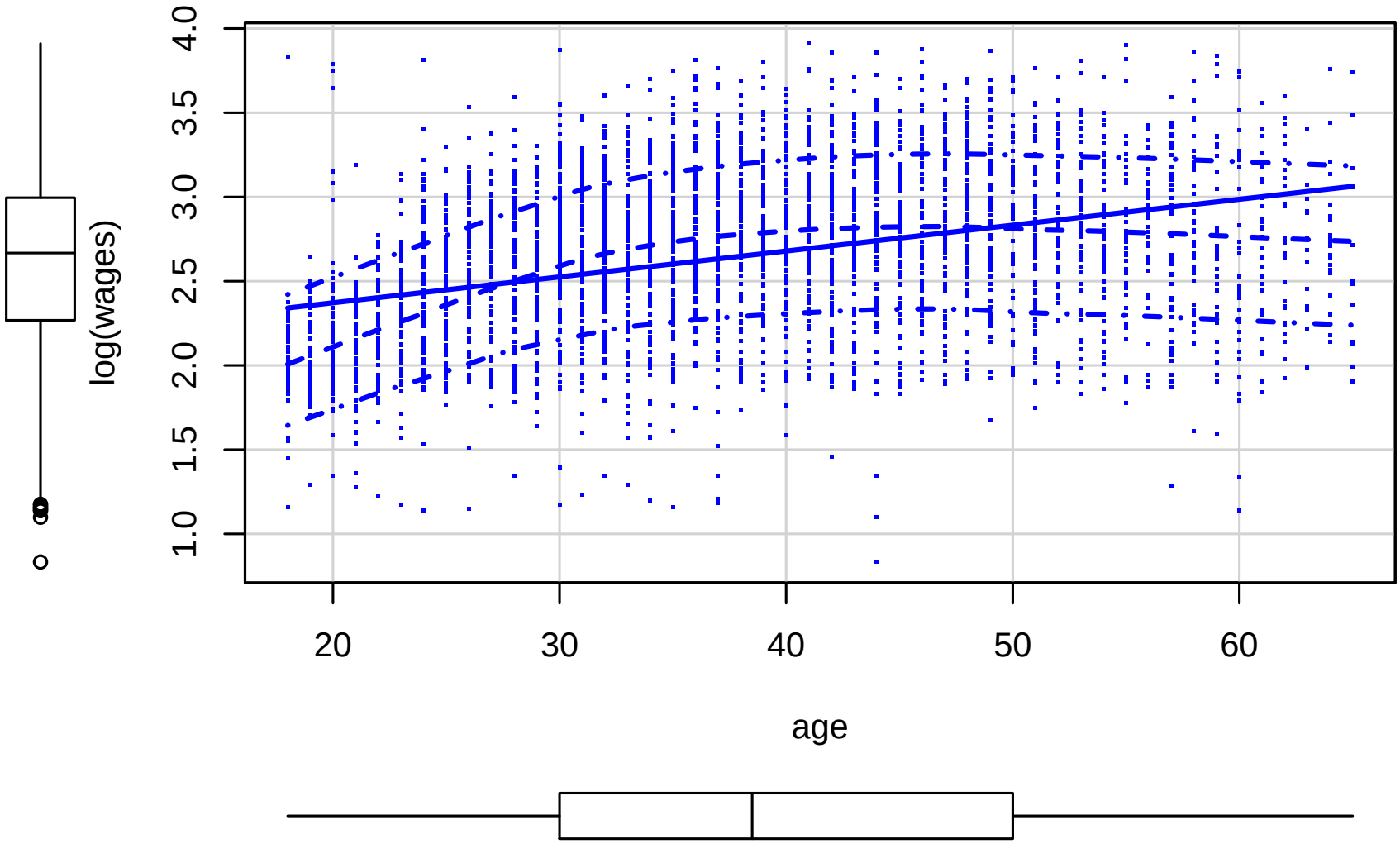
```
par(mfrow = c(1,2))
hist(SLID$wages, col='gray75', main='wages')
hist(log(SLID$wages), col='gray75', main='log(wages)')
```



We will consider `log(wages)` as the output variable and want to regress it on age

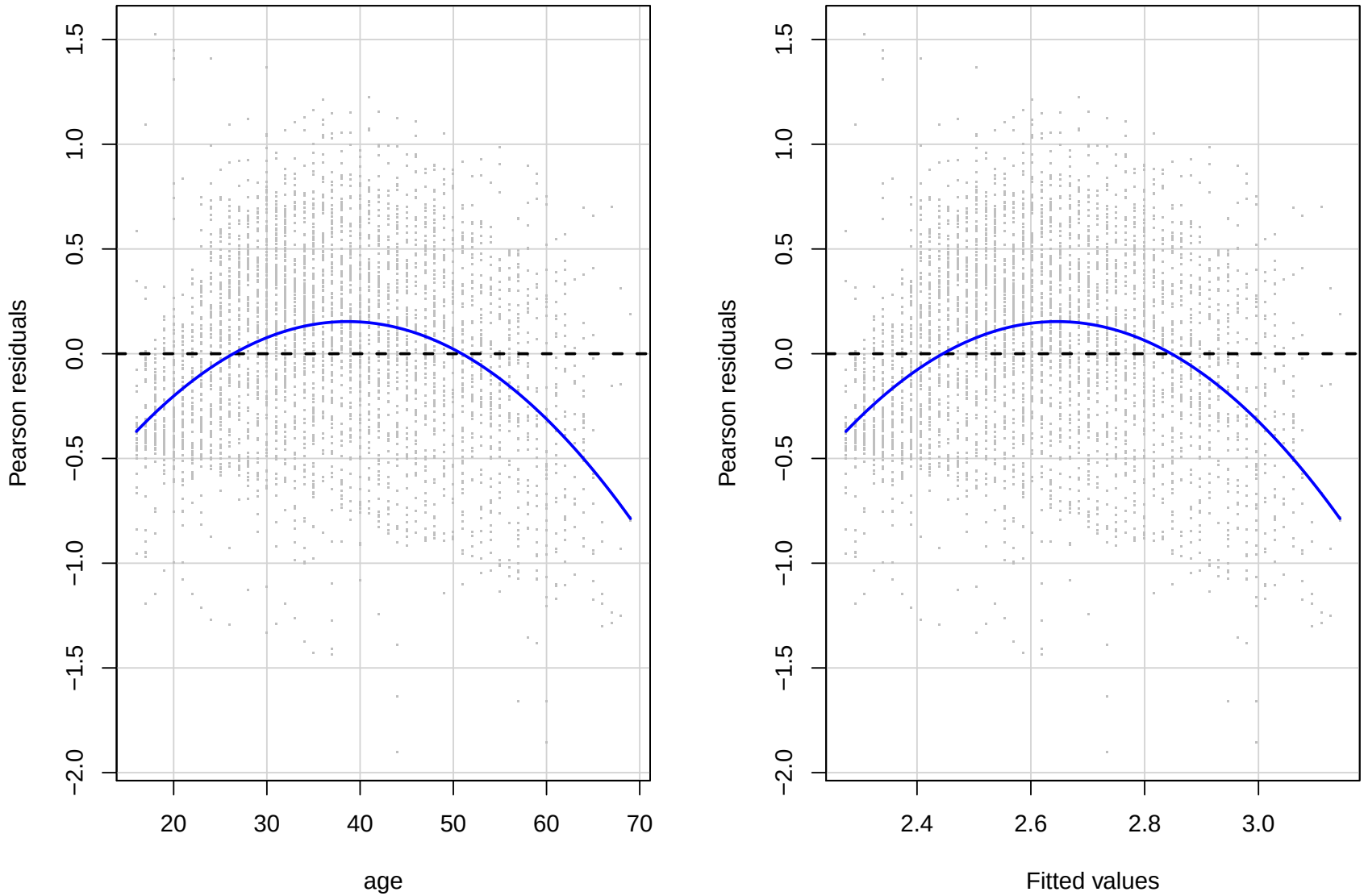
Polynomial Regression

```
slid.m <- lm(log(wages)~age, data = SLID)
scatterplot(log(wages)~age, data = SLID, pch='.',
            subset = age>= 18 & age <= 65)
```



Polynomial Regression

```
residualPlots(slid.m, pch ='.', col=gray(0.75))
```




```
##                 Test stat Pr(>|Test stat|)
```

Polynomial Regression

```
residualPlots(slid.m, plot = FALSE)
```

```
##             Test stat Pr(>|Test stat|)
## age           -24.389        < 2.2e-16 ***
## Tukey test    -24.389        < 2.2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Polynomial Regression

The quadratic model can be fit in at least three equivalent ways:

Adding the square of the regressor with the function `I()`,

```
slid.m1 <- update(slid.m, ~ age + I(age^2))
brief(slid.m1)
```

```
##             (Intercept)     age  I(age^2)
## Estimate         0.6394 0.09538 -1.02e-03
## Std. Error       0.0603 0.00329  4.19e-05
##
##  Residual SD = 0.433 on 4144 df, R-squared = 0.262
```

Polynomial Regression

using the `poly()` function with the option `raw = TRUE`,

```
slid.m2 <- lm(log(wages) ~ poly(age,2,raw = TRUE), data=SLID,
                subset = age>= 18 & age <= 65)
brief(slid.m2)
```

```
##             (Intercept) poly(age, 2, raw = TRUE)1
## Estimate         0.6515                   0.09486
## Std. Error       0.0697                   0.00375
##             poly(age, 2, raw = TRUE)2
## Estimate                    -1.02e-03
## Std. Error                   4.73e-05
##
##  Residual SD = 0.436 on 4010 df, R-squared = 0.222
```

Polynomial Regression

or using the `poly()` function with no `raw` option

```
slid.m3 <- lm(log(wages) ~ poly(age,2), data=SLID,
                subset = age>= 18 & age <= 65)
brief(slid.m3)
```

```
##             (Intercept) poly(age, 2)1 poly(age, 2)2
## Estimate         2.5388         -3.18        -29.11
## Std. Error       0.0122          1.53          1.35
##
##  Residual SD = 0.436 on 4010 df, R-squared = 0.222
```

Polynomial Regression

The first and second give approximately the same coefficients. The third option fits orthogonal polynomials, and the coefficients are different but fitted values are the same.

```
Anova(slid.m1)
```

```
## Anova Table (Type II tests)
##
## Response: log(wages)
##             Sum Sq   Df F value    Pr(>F)
## age         158.14    1  842.39 < 2.2e-16 ***
## I(age^2)    111.66    1  594.81 < 2.2e-16 ***
## Residuals 777.95 4144
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Polynomial Regression

```
Anova(slid.m2)
```

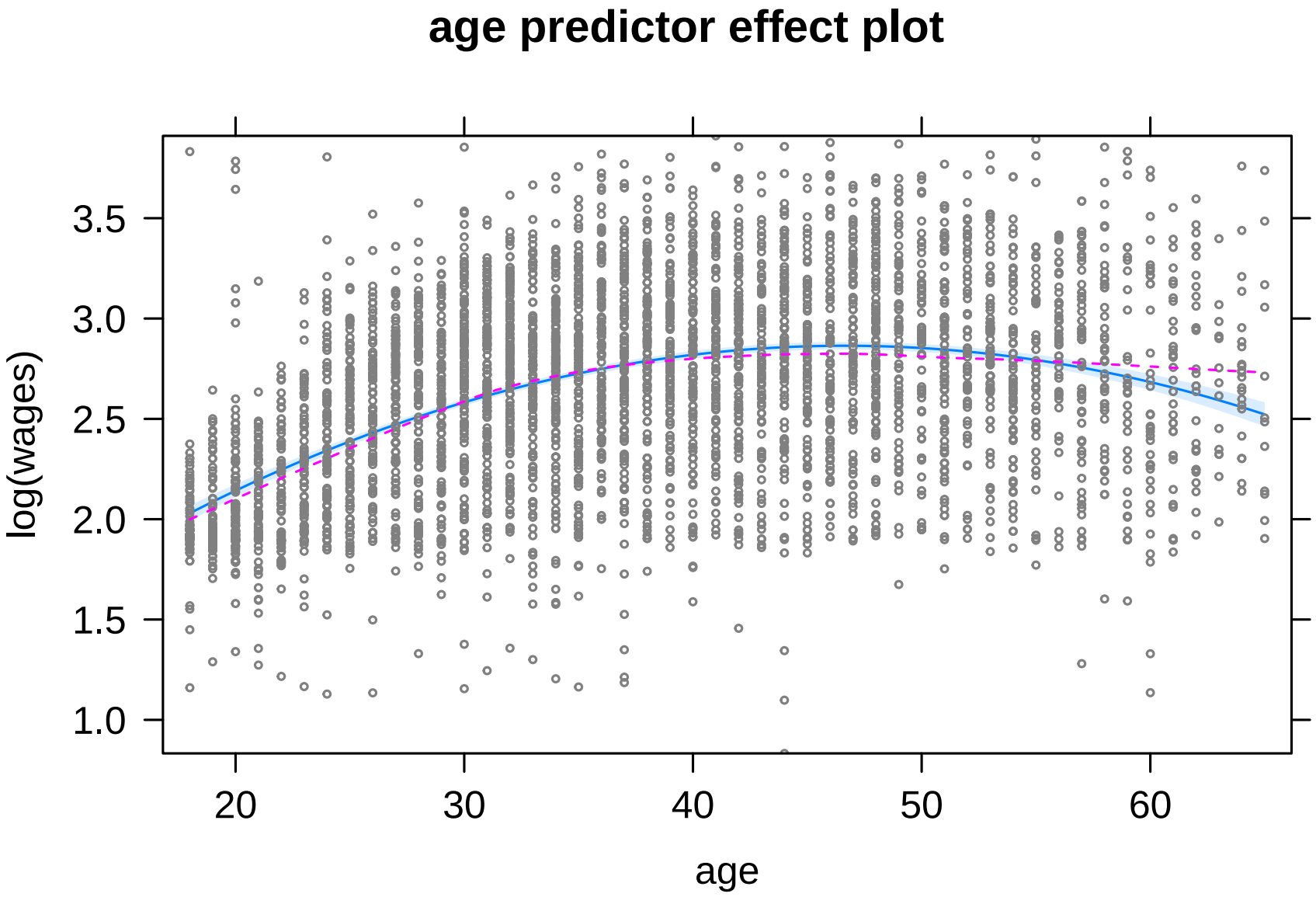
```
## Anova Table (Type II tests)
##
## Response: log(wages)
##                             Sum Sq   Df F value    Pr(>F)
## poly(age, 2, raw = TRUE) 216.83    2  570.62 < 2.2e-16 ***
## Residuals                761.88 4010
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Anova(slid.m3)
```

```
## Anova Table (Type II tests)
##
## Response: log(wages)
##               Sum Sq   Df F value    Pr(>F)
## poly(age, 2) 216.83    2  570.62 < 2.2e-16 ***
## Residuals    761.88 4010
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

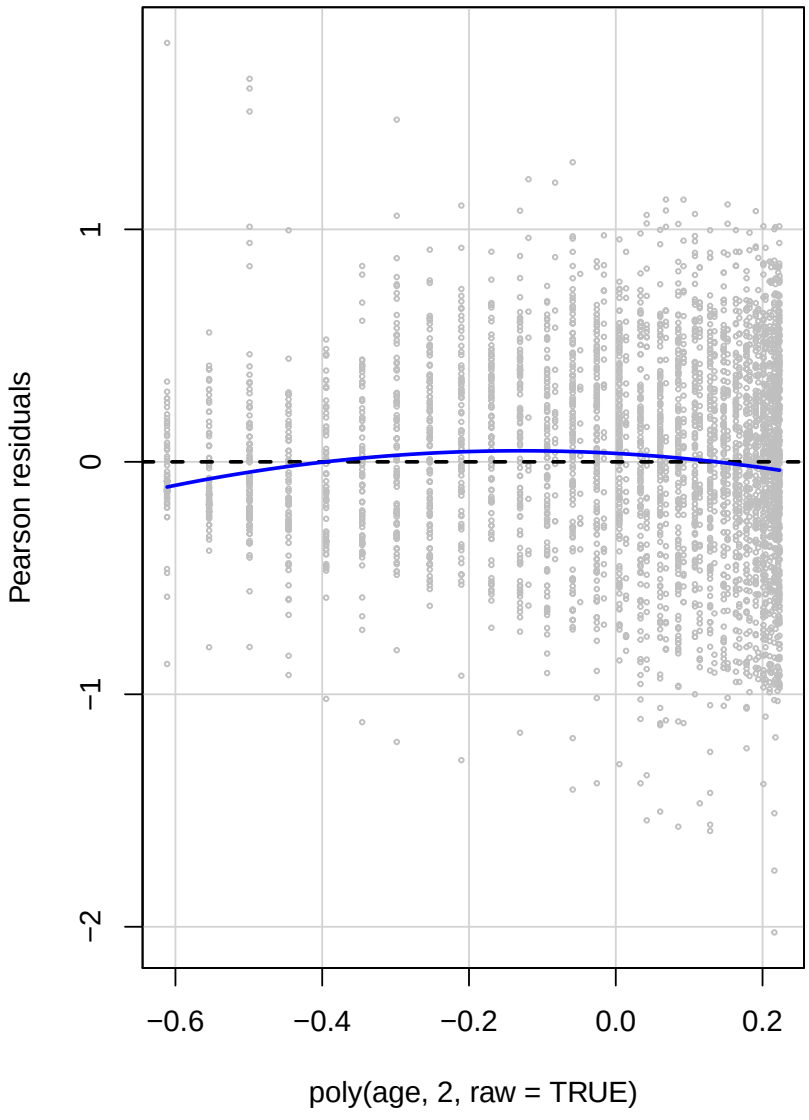
Polynomial Regression

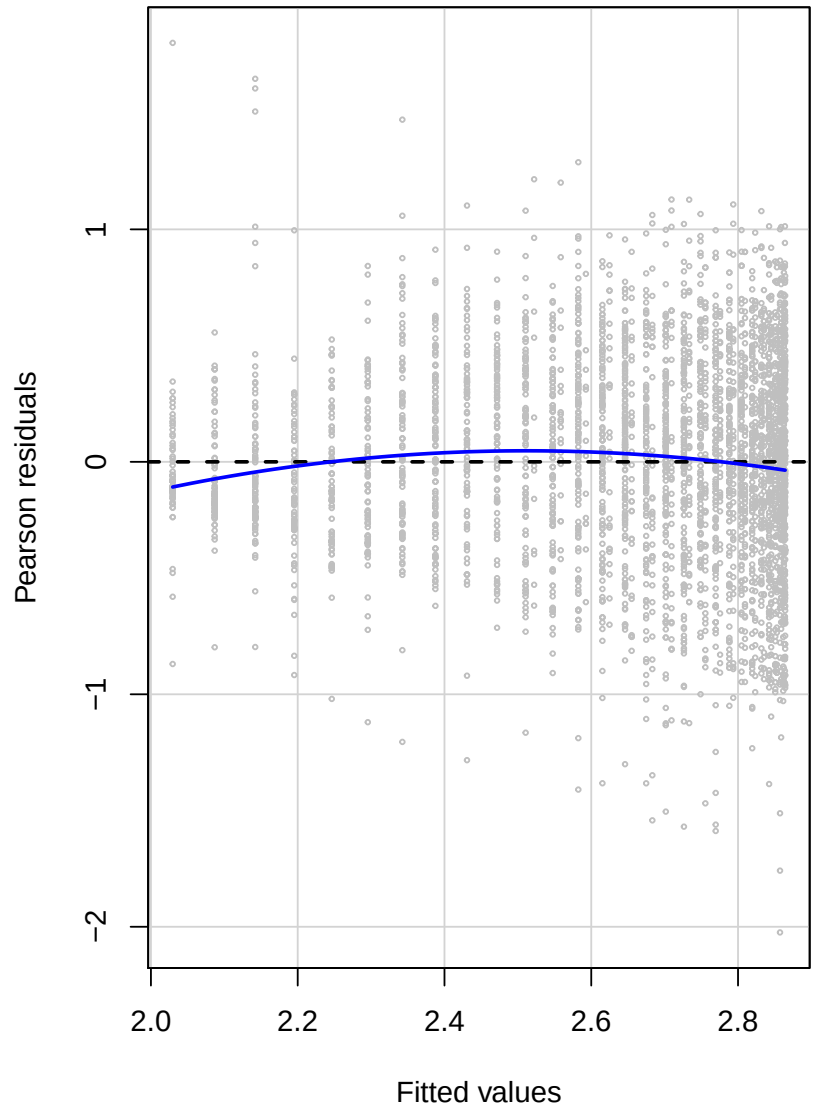
```
plot(predictorEffects(slid.m2, residuals = TRUE),
     partial.residuals = list(cex=0.35, col=gray(0.5), lty = 2))
```



Polynomial Regression

```
residualPlots(slid.m2, cex=0.35, col=gray(0.75), tests = FA
```





Qualitative Predictors

Qualitative Predictors

Up to this point, only quantitative (continuous) predictor variables have been used in regression models.

Regression using quantitative variables can be generalized to qualitative variables with the use of **dummy** variables.

A dummy variable is any variable in a regression model that takes on a finite number of values to identify different categories of a nominal variable.

Provided the regression model has an intercept, one must define $k - 1$ dummy variables to define a qualitative variable with k categories.

Qualitative Predictors

There are many ways to define the $k - 1$ dummy variables. R uses treatment contrasts by default to define qualitative variables (factors).

To see the values R uses to define a qualitative variable with four levels, enter

```
contr.treatment(4)
```

```
##   2 3 4
## 1 0 0 0
## 2 1 0 0
## 3 0 1 0
## 4 0 0 1
```

The rows of this matrix (4×3) are the levels of the qualitative predictor, and the columns are the dummy variables. R assigns levels to a qualitative variable in alphabetical order by default.

Qualitative Predictors

As an example, consider again the `iris` data set and the variable `species` that takes three values.

To include this variable in a regression, we have a dummy variable with the following values

```
contrasts(iris$Species)
```

```
##            versicolor virginica
## setosa              0         0
## versicolor          1         0
## virginica           0         1
```

Qualitative Predictors

The simplest situation where dummy variables might be used in a regression model is when the qualitative predictor has only two levels.

The regression model for a single quantitative predictor (X) and a dummy variable (D) is written

$$Y = \beta_0 + \beta_1 X + \beta_2 D + \beta_3 DX + \epsilon \tag{1}$$

where

$$D = \begin{cases} 0 & \text{for the first level} \\ 1 & \text{for the second level} \end{cases}$$

The model in (1) when D has two levels will yield one of four possible scenarios, as shown in the figure on the next slide.

Qualitative Predictors

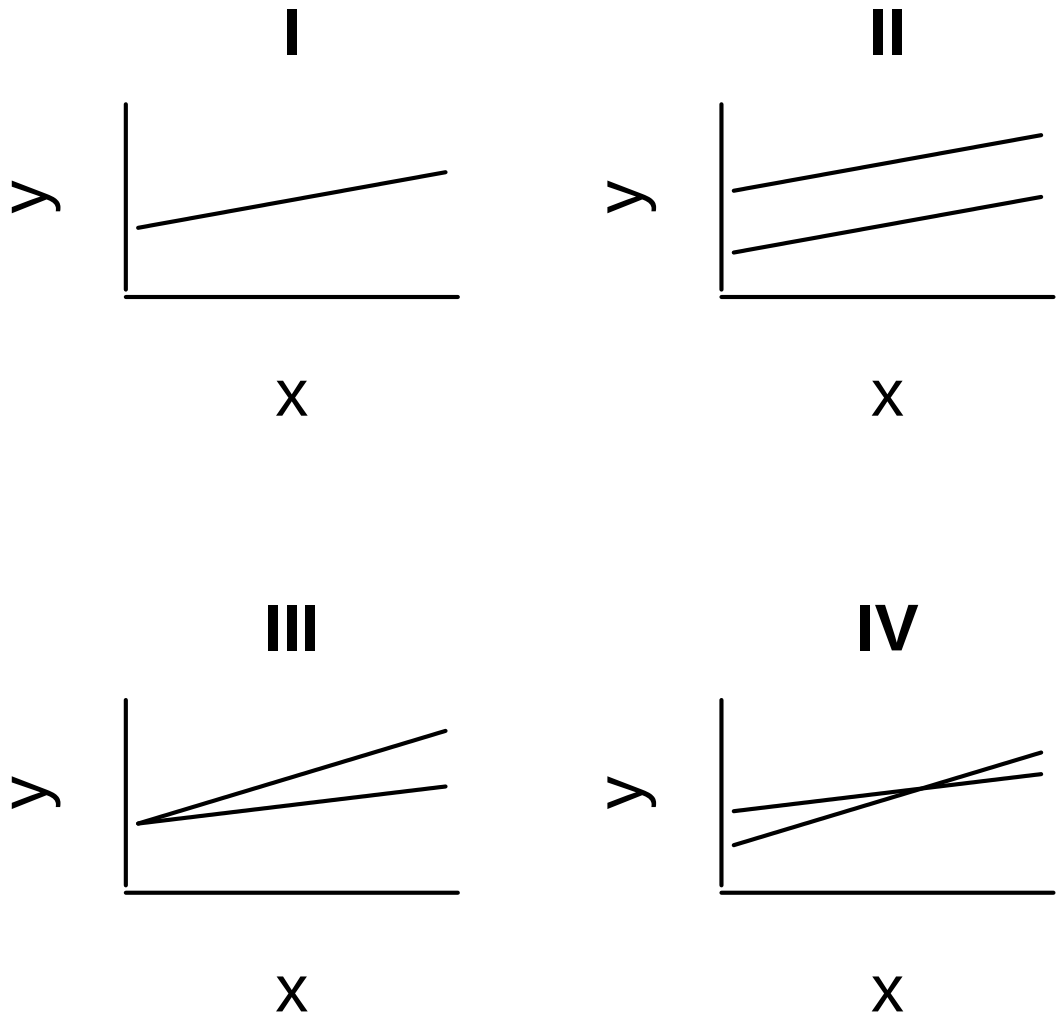


Figure 1: Effect of categorical variables in simple linear regression

Qualitative Predictors

This type of model requires the user to answer three basic questions:

(1) Are the lines the same?

(2) Are the slopes the same?

(3) Are the intercepts the same?

To address basic question (1), the null hypothesis $H_0 : \beta_2 = \beta_3 = 0$ must be tested.

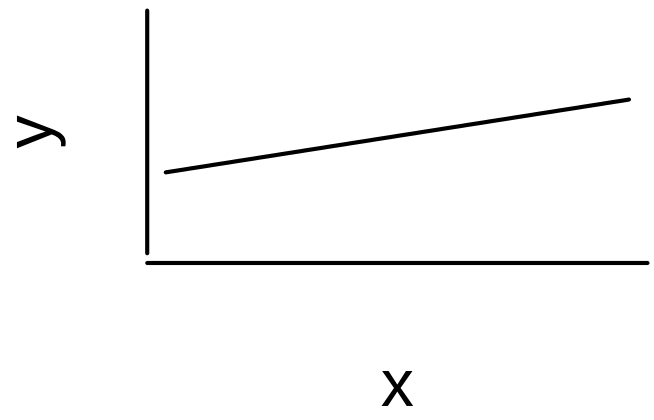
One way to perform the test is to use the general linear test statistic based on the full model found in (1) and the reduced model

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Qualitative Predictors

If the null hypothesis is not rejected, the interpretation is that there is one line present (the intercept and the slope are the same for both levels of the dummy variable).

I



This is the case for graph I. If the null hypothesis is rejected, either the slopes, the intercepts, or possibly both the slope and the intercept are different for the different levels of the dummy variable, as seen in graphs II, III, and IV.

Qualitative Predictors

To answer basic question (2), the null hypothesis $H_0 : \beta_3 = 0$ must be tested.

If the null hypothesis is not rejected, the two lines have the same slope, but different intercepts, as shown in graph II.

The two parallel lines that result when $\beta_3 = 0$ are

$$Y = \beta_0 + \beta_1 X \text{ (for } D = 0)$$

and

$$Y = (\beta_0 + \beta_2) + \beta_1 X \text{ (for } D = 1).$$

When $H_0 : \beta_3 = 0$ is rejected, one concludes that the two fitted lines are not parallel, as in graphs III and IV.

Qualitative Predictors

To answer basic question (3), the null hypothesis $H_0 : \beta_2 = 0$ for model (1) must be tested.

The reduced model for this test is

$$Y = \beta_0 + \beta_1 X + \beta_3 DX + \epsilon.$$

If the null hypothesis is not rejected, the two fitted lines have the same intercept but different slopes:

$$Y = \beta_0 + \beta_1 X \text{ (for } D = 0)$$

and

$$Y = \beta_0 + (\beta_1 + \beta_3) X \text{ (for } D = 1)$$

Qualitative Predictors

Graph III represents this situation. If the null hypothesis is rejected, one concludes that the two lines have different intercepts, as in graphs II and IV.

Let's go back to the `crabs` example we examined at the beginning of our consideration on Simple Linear Regression. There we looked at the relation between the variables `CL` and `FL`.

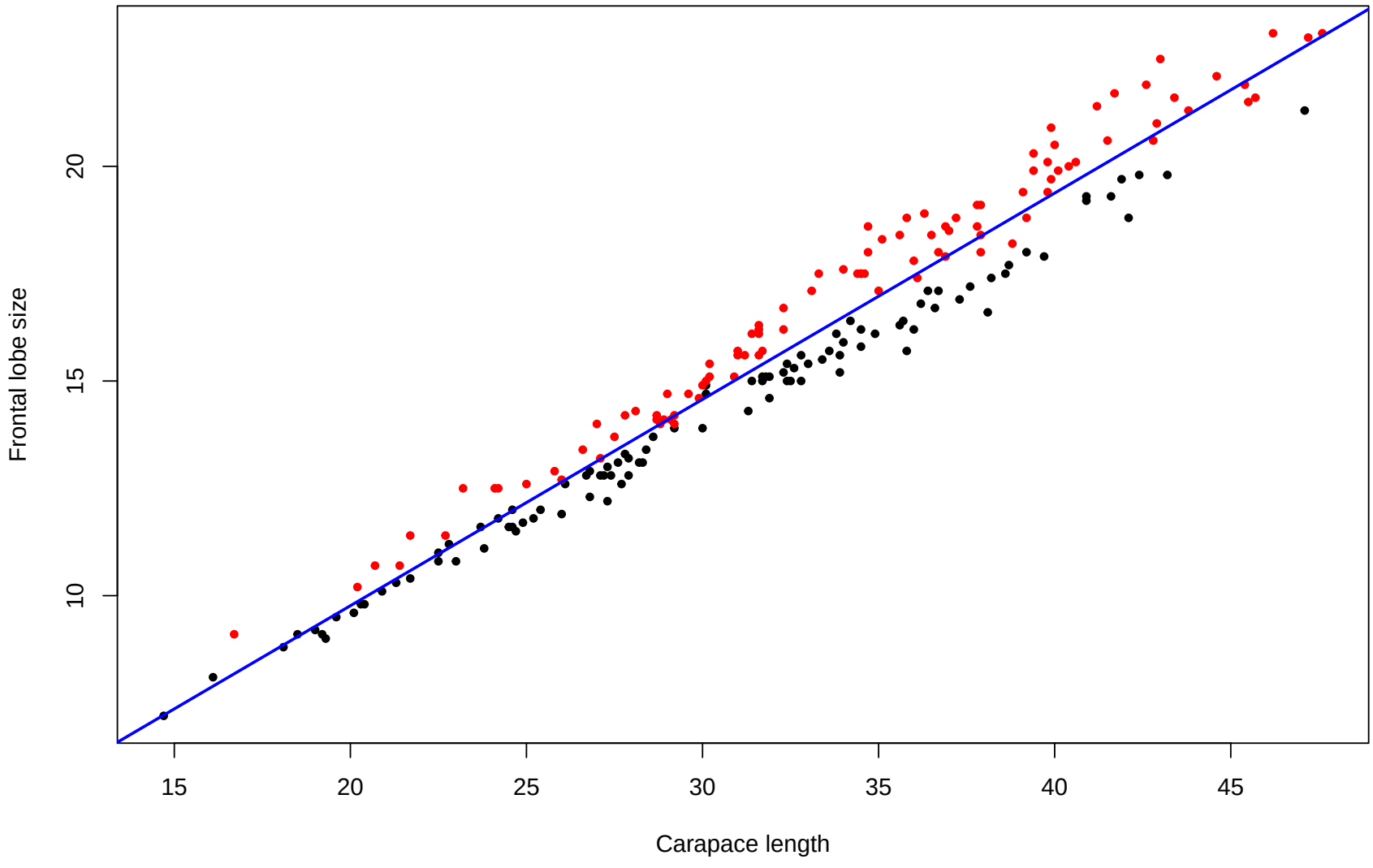
Example: Crabs

```
library(MASS); attach(crabs)
lmSimple <- lm(FL~CL); summary(lmSimple)
```

```
##
## Call:
## lm(formula = FL ~ CL)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.86395 -0.51746 -0.02826  0.50456  1.77009
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.15316    0.23477   0.652    0.515
## CL           0.48060    0.00714  67.313   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.717 on 198 degrees of freedom
## Multiple R-squared:  0.9581, Adjusted R-squared:  0.9579
## F-statistic:  4531 on 1 and 198 DF,  p-value: < 2.2e-16
```

Example: Crabs

```
plot(CL,FL, pch=20, xlab='Carapace length', ylab='Frontal l
abline(lmSimple, lw=2, col='blue')
```



Example: Crabs

This corresponds to fitting the simple model

$$FL = \beta_0 + \beta_1 CL + \epsilon$$

We now consider a second model including a dummy variable D for species, which gives

$$FL = \beta_0 + \beta_1 CL + \beta_2 D + \beta_3 CL \cdot D + \epsilon \tag{2}$$

```
fsp <- as.factor(sp)
contrasts(fsp)
```

```
##   O
## B 0
## O 1
```

Example: Crabs

The new vector `fsp` is a `factor` with values 0 for the blue (`B`) species and 1 for the orange (`O`).

We now fit the complete model (2).

```
lmComplete <-  lm(FL~CL+fsp+CL:fsp)
brief(lmComplete)
```

```
##             (Intercept)      CL   fspO CL:fspO
## Estimate          0.971 0.43531 -0.209 0.04335
## Std. Error        0.185 0.00599  0.282 0.00855
##
##  Residual SD = 0.411 on 196 df, R-squared = 0.986
```

Example: Crabs

To compare these models, we use an anova with the two models

```
anova(lmSimple,lmComplete)
```

```
## Analysis of Variance Table
##
## Model 1: FL ~ CL
## Model 2: FL ~ CL + fsp + CL:fsp
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1    198 101.793
## 2    196  33.139  2    68.654 203.03 < 2.2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Example: Crabs

The small p-value says that at least one of the two parameters β_2, β_3 is not zero.

To see if the lines have different slopes, we want to test

$$H_0 : \beta_3 = 0 \qquad \textit{vs.} \qquad H_1 : \beta_3 \neq 0.$$

Looking at the `summary()` for the `lmComplete` model

Example: Crabs

```
summary(lmComplete)
```

```
##
## Call:
## lm(formula = FL ~ CL + fsp + CL:fsp)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.13437 -0.23131 -0.01476  0.23612  1.22817
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.971315   0.184593   5.262 3.72e-07 ***
## CL           0.435315   0.005987  72.711  < 2e-16 ***
## fspO        -0.209274   0.281608  -0.743    0.458
## CL:fspO      0.043354   0.008554   5.068 9.25e-07 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4112 on 196 degrees of freedom
## Multiple R-squared:  0.9864, Adjusted R-squared:  0.9862
## F-statistic:  4728 on 3 and 196 DF,  p-value: < 2.2e-16
```

Example: Crabs

We see that the term `CL:fsp0` has a small p-value and therefore is statistically significant at the usual levels.

This means that the slopes are different: when $D = 0$ (blue species) the slope is $\beta_1 = 0.435315$ and when $D = 1$ (orange species) the slope is $\beta_1 + \beta_3 = 0.435315 + 0.043354 = 0.478669$.

Finally, the variable `fsp0` is the dummy variable, and the p value associated with it is large (0.458), which means it is not significant, and therefore there is no difference in the intercepts. The final model, then, is an intermediate model:

Example: Crabs

```
lmInter <-  lm(FL~CL+CL:fsp)
summary(lmInter)
```

```
##
## Call:
## lm(formula = FL ~ CL + CL:fsp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1232 -0.2509 -0.0102  0.2441  1.2255
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.881395   0.139246    6.33 1.62e-09 ***
## CL          0.438158   0.004600   95.26  < 2e-16 ***
## CL:fspO     0.037147   0.001843   20.16  < 2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4107 on 197 degrees of freedom
## Multiple R-squared:  0.9863, Adjusted R-squared:  0.9862
## F-statistic:  7108 on 2 and 197 DF,  p-value: < 2.2e-16
```

Example: Crabs

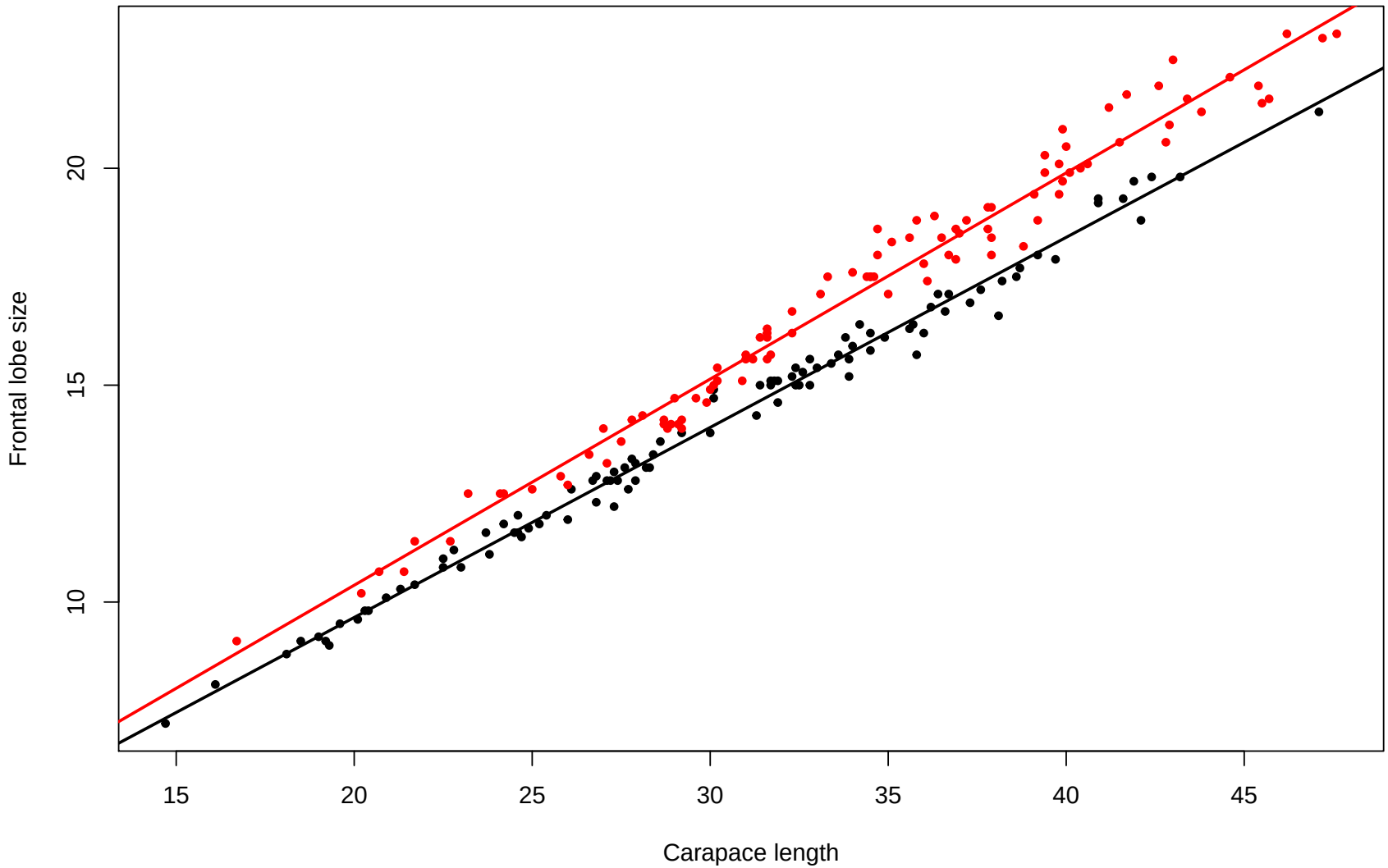
Observe that if we compare this model with the complete model through an anova table, we get the same test as in the summary:

```
anova(lmInter,lmComplete)
```

```
## Analysis of Variance Table
##
## Model 1: FL ~ CL + CL:fsp
## Model 2: FL ~ CL + fsp + CL:fsp
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    197 33.232
## 2    196 33.139  1  0.093374 0.5523 0.4583
```

Example: Crabs

```
plot(CL,FL, pch=20, xlab='Carapace length', ylab='Frontal lobe size', col=sp)
beta <- coef(lmInter)
abline(beta[1], beta[2],lwd=2)
abline(beta[1], sum(beta[-1]),lwd=2, col='red')
```



Example: Crabs

The final model is

$$FL = 0.881395 + 0.438158 \times CL + 0.037147 \times CL \times fsp.$$