solution_1

2024-09-02

Part 1: Explanatory analysis of the dataset

Assumptions:

- linearity of covariate effects
- homescedasticity of error variance –> error variacnce same for different dependent variables
- uncorrelated error
- additivity of errors

Diagnostics plots:

```r
#install.packages("car")
library(car)
```

```
## Loading required package: carData
```

```r
library(GGally)
```

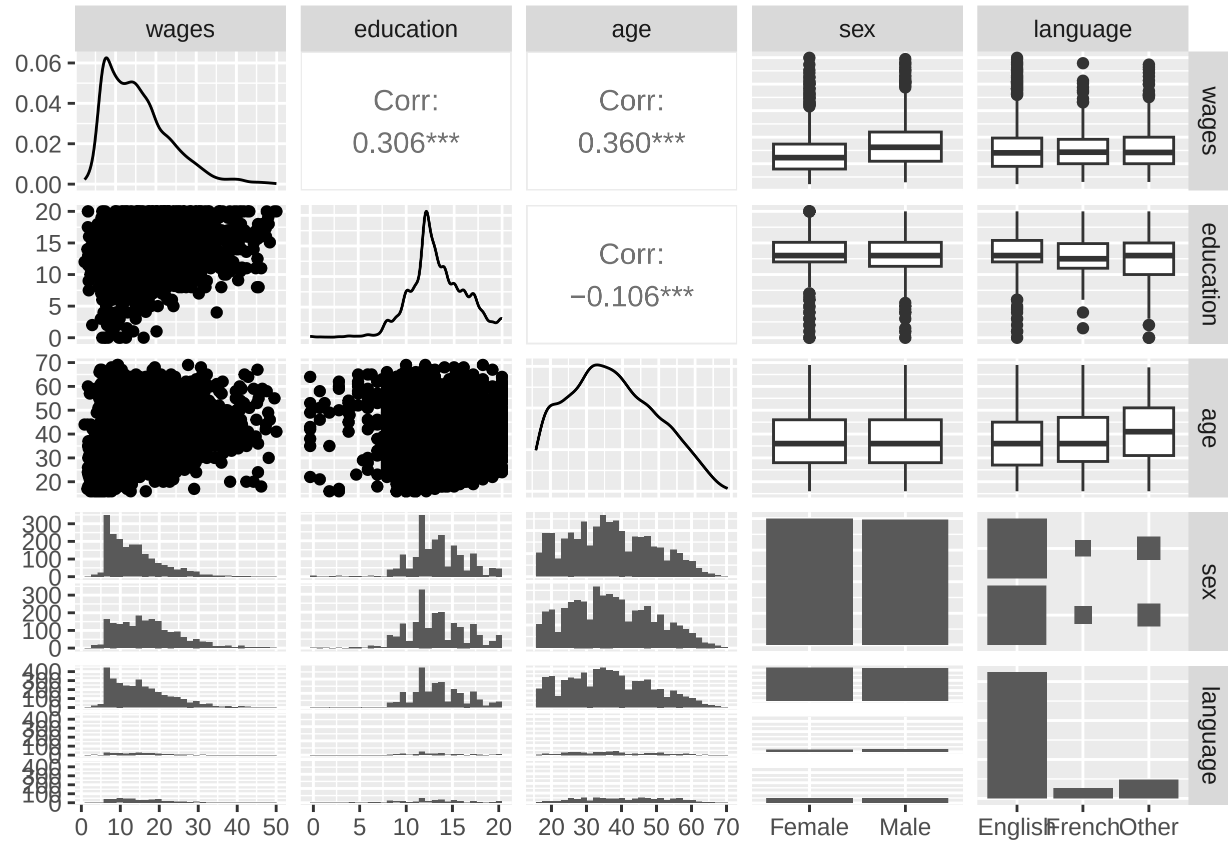
```
## Loading required package: ggplot2
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
data(SLID, package = "carData")
SLID <- SLID[complete.cases(SLID), ]
ggpairs(SLID)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Relations: - medium strong (Cohen) , positive correlation between education and wages and age and wages –> good if we want to predict wages, bad if not, because then it shows multicollinearity - weak (Cohen), negative correlation between age and education –> again good for prediction of one of them, but bad if not - there seems to be a small difference in wages between the sexes (males gaining more than females) - no difference in wages regarding the language level - about same distribution of education and age regarding sex

Part 2: Linear regression with the mylm package

Questions:

What is the interpretation of the parameter estimates? - The parameter estimation shows the influence or strength of influence of the covariate on the dependent variable. - The z- and p-values show if this effect in the linear regression is significant or not, so if it would show as well, if we would repeat the experiment. - The intercept shows the value of the dependent varaible if all covariatees are 0.

```
#install.packages("car")
#install.packages("mylm")
library(mylm)
```

```
##
## Attaching package: 'mylm'

## The following object is masked from 'package:carData':
##
##     SLID
```

```
library(car)
library(GGally)
library(stringr)
```

```r
data(SLID, package = "carData")
SLID <- SLID[complete.cases(SLID), ]

# comparison simple model
model1 <- mylm(wages ~ education, data = SLID)
model1b <- lm(wages ~ education, data = SLID)

# print
print.mylm(model1)
```

```
## Call:
## mylm(formula = wages ~ education, data = SLID)
##
## Coefficients:
## (Intercept) : 4.9717
## education : 0.7923
```

```r
# summary
summary.mylm(model1)
```

```
## Call:
## mylm(formula = wages ~ education, data = SLID)
##
## Residuals:
## Min       1Q        Median    3Q        Max
## -17.688   -5.822    -1.039    4.148     34.190
##
## Coefficients:
##                 Estimate     Std. Error   z value      Pr(>|z|)
## (Intercept)     4.971691     0.5344       9.3040       0.0000 ***
## education       0.7923091    0.0391       20.2816      0.0000 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  7.493 on  3984 degrees of freedom
## Multiple R-squared:  0.09359 ,  Adjusted R-squared:  0.09313
## F-statistic:  411.3 on 1 and 3984 DF, p-value: 0.000
```

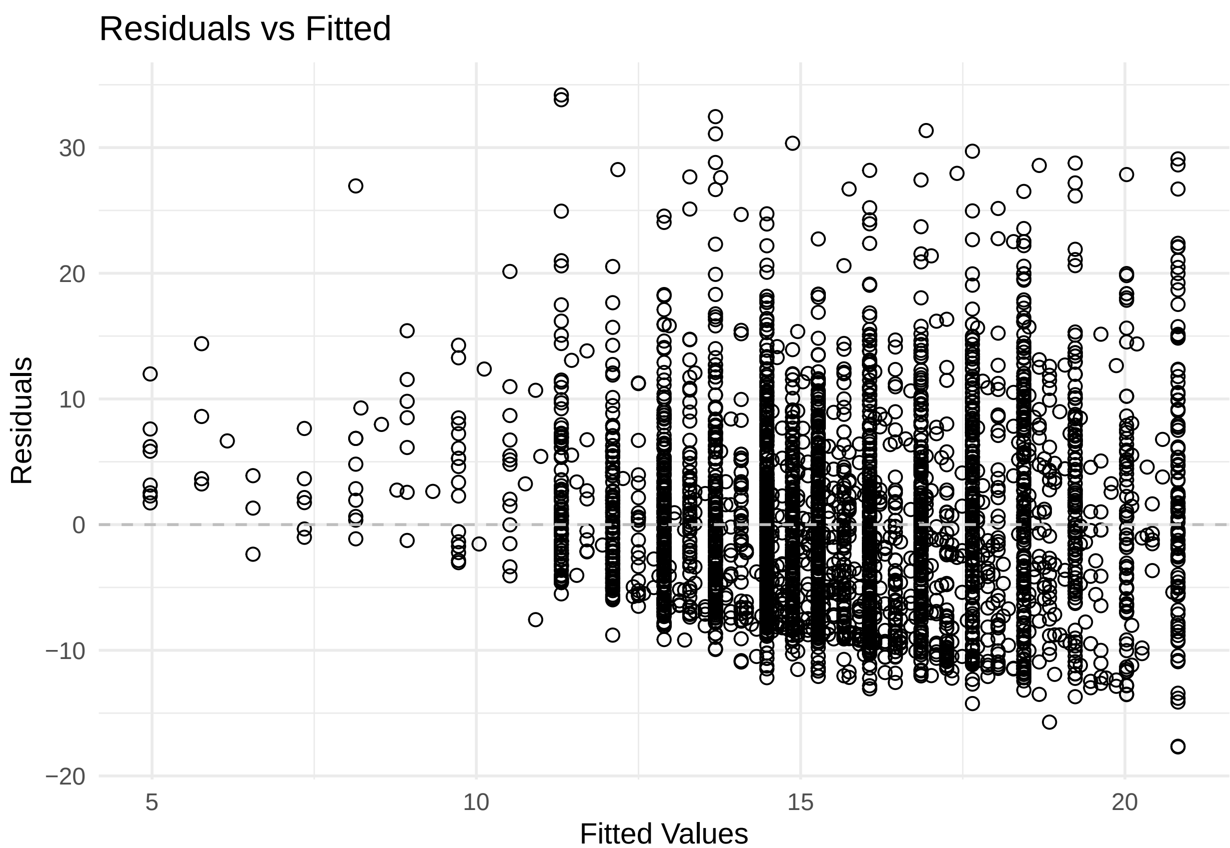
```r
summary(model1b)
```

```
##
## Call:
## lm(formula = wages ~ education, data = SLID)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.688  -5.822  -1.039   4.148  34.190
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.97169    0.53429   9.305   <2e-16 ***
## education    0.79231    0.03906  20.284   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.492 on 3985 degrees of freedom
```

```
## Multiple R-squared:  0.09359,    Adjusted R-squared:  0.09336
## F-statistic: 411.4 on 1 and 3985 DF,  p-value: < 2.2e-16
```

Plot

```r
library(mylm)
model1 <- mylm(wages ~ education, data = SLID)
plot.mylm(model1)
```



Comments:

- Heteroscadicity, so for larger fitted values the prediction gets less precise and the errors are more widely spread –> This indicates that there might be an error in the way the covariates are modelled.
- We have only little data for the higher values of wages, so this might make the prediction harder as there is less to fit/ learn from.
- Also, the distribution for the positive residuals has a higher variance than for the negative one.

ANOVA

-What is the residual sum of squares (SSE) and the degrees of freedom for this model? See output. -What is total sum of squares (SST) for this model? Test the significance of the regression using a χ^2-test. 23096 (Sum Sq education) + 223694 (sum sq residuals) = 246790. -What is the relationship between the χ^2- and z-statistic in simple linear regression? Find the critical value(s) for both tests. As there is only one beta to test in simple linear regression the χ^2 for the whole model should be the squared z statistic for education, which is true for our values $\chi^2 = 411$ and z = 20.284 (20.284ˆ2 = 411.44).

```r
library(mylm)
library(stringr)
model1 <- mylm(wages ~ education, data = SLID)
anova.mylm(model1)
```

```
## [1] "wages~"
## Analysis of Variance Table
## Response: wages
##               Df         Sum Sq     Mean Sq    Chi^2      Pr(>Chi^2)
## education     1          23096      23096      411.447    0 ***
## Residuals     3984       223694     56
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Total Sum SQ: 246790
## Chi-statistic:  411.3 on 3984 DF, p-value: 0.000
```

Comments:

- The covariate education gets a significant χ^2 value which means there is a high probability that it explains variance of the dependent variable.

Part 3: Multiple Linear Regression

-What are the estimates and standard errors of the intercepts and regression coefficients for this model? -Test the significance of the coefficients using a z-test. -What is the interpretation of the parameters?

```r
library(mylm)
library(stringr)
model1d <- mylm(wages ~ education + age, data = SLID)
summary.mylm(model1d)
```

```
## Call:
## mylm(formula = wages ~ education + age, data = SLID)
##
## Residuals:
## Min       1Q        Median    3Q        Max
## -24.303   -4.495    -0.807    3.674     37.628
##
## Coefficients:
##               Estimate     Std. Error   z value      Pr(>|z|)
## (Intercept)   -6.021653    0.6190       -9.7280      0.0000 ***
## education     0.9014644    0.0358       25.2057      0.0000 ***
## age           0.2570898    0.0090       28.7176      0.0000 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  6.821 on  3983 degrees of freedom
## Multiple R-squared:  0.2491 ,  Adjusted R-squared:  0.2485
## F-statistic:  660.5 on 2 and 3983 DF, p-value: 0.000
```

What is the interpretation of the parameters? - The intercept is negative, probably because all data points (persons) have an age higher than 0. - Education seems to have a higher influence than age as the parameter estimate is higher. - Both parameters are significant, so should have an influence on wages.

Comparison simple linear regression and multiple linear regression

```r
library(mylm)
model1 <- mylm(wages ~ education, data = SLID)
summary.mylm(model1)
```

```
## Call:
## mylm(formula = wages ~ education, data = SLID)
##
## Residuals:
## Min       1Q        Median    3Q        Max
## -17.688   -5.822    -1.039    4.148     34.190
##
## Coefficients:
##                 Estimate     Std. Error   z value      Pr(>|z|)
## (Intercept)     4.971691     0.5344       9.3040       0.0000 ***
## education       0.7923091    0.0391       20.2816      0.0000 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  7.493 on  3984 degrees of freedom
## Multiple R-squared:  0.09359 ,  Adjusted R-squared:  0.09313
## F-statistic:  411.3 on 1 and 3984 DF, p-value: 0.000
```

```r
model1c <- mylm(wages ~ age , data = SLID)
summary.mylm(model1c)
```

```
## Call:
## mylm(formula = wages ~ age, data = SLID)
##
## Residuals:
## Min       1Q        Median    3Q        Max
## -17.747   -4.847    -1.507    3.914     35.063
##
## Coefficients:
##                 Estimate     Std. Error   z value      Pr(>|z|)
## (Intercept)     6.890901     0.3741       18.4202      0.0000 ***
## age             0.2331079    0.0096       24.3222      0.0000 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  7.344 on  3984 degrees of freedom
## Multiple R-squared:  0.1293 ,  Adjusted R-squared:  0.1289
## F-statistic:  591.6 on 1 and 3984 DF, p-value: 0.000
```

```r
model2 <- mylm(wages ~ education + age, data = SLID)
summary.mylm(model2)
```

```
## Call:
## mylm(formula = wages ~ education + age, data = SLID)
##
## Residuals:
## Min       1Q        Median    3Q        Max
## -24.303   -4.495    -0.807    3.674     37.628
##
## Coefficients:
##                 Estimate     Std. Error   z value      Pr(>|z|)
## (Intercept)     -6.021653    0.6190       -9.7280      0.0000 ***
```

```
## education     0.9014644    0.0358       25.2057      0.0000 ***
## age           0.2570898    0.0090       28.7176      0.0000 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  6.821 on  3983 degrees of freedom
## Multiple R-squared:  0.2491 ,  Adjusted R-squared:  0.2485
## F-statistic:  660.5 on 2 and 3983 DF, p-value: 0.000
```

```r
test <- mylm(age ~ education, data = SLID)
summary.mylm(test)
```

```
## Call:
## mylm(formula = age ~ education, data = SLID)
##
## Residuals:
## Min         1Q         Median     3Q         Max
## -25.9116   -8.9982    -0.6658    8.9097     33.8817
##
## Coefficients:
##                Estimate      Std. Error    z value       Pr(>|z|)
## (Intercept)   42.76072      0.8609        49.6726       0.0000 ***
## education     -0.4245804    0.0629        -6.7464       2e-11 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  12.07 on  3984 degrees of freedom
## Multiple R-squared:  0.0113 ,  Adjusted R-squared:  0.0108
## F-statistic:  45.5 on 1 and 3984 DF, p-value: 2e-11
```

Why (and when) does the parameter estimates found (the two simple and the one multiple) differ? - They should not differ, if the covariates so age and education are independent (no correlation). - They differ because age and education are correlated. Explain/show how you can use mylm to find these values. –> Use the function on the model.

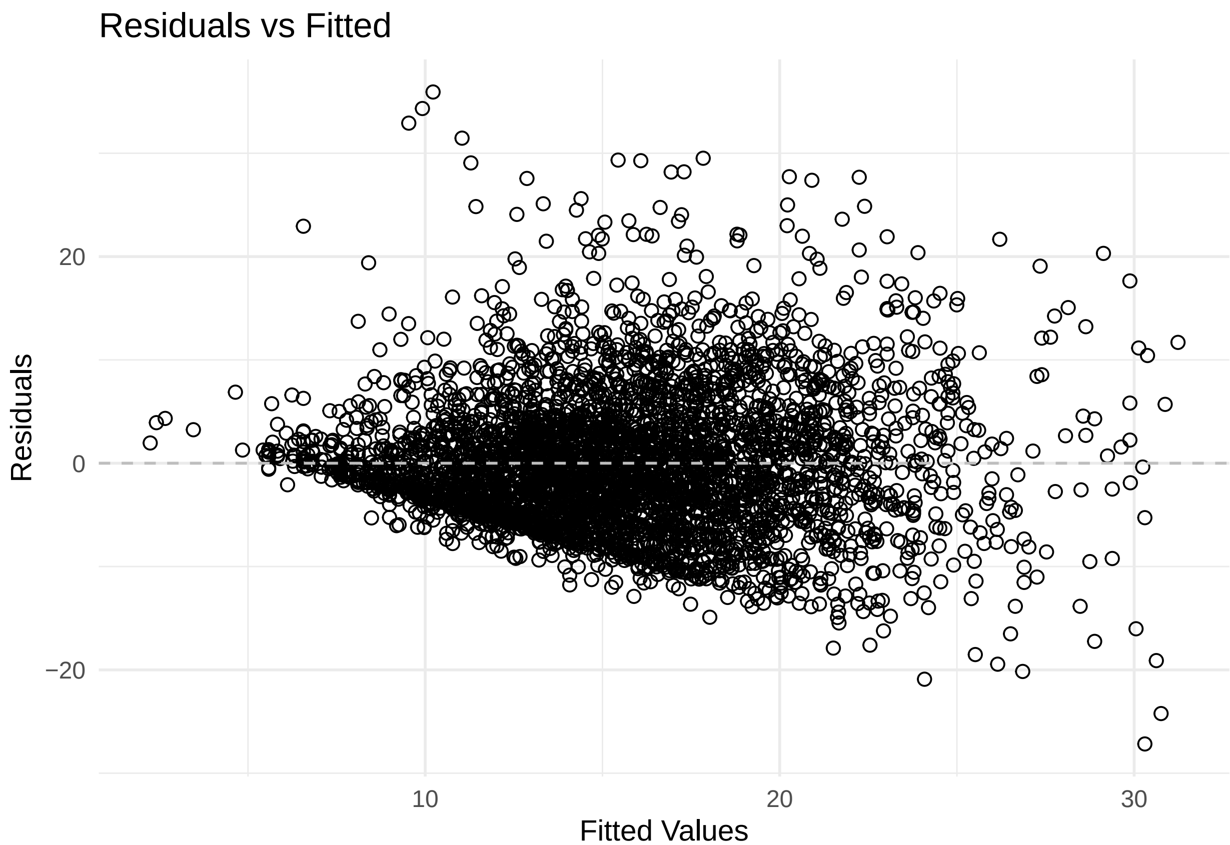
Part 4: Testing mylm

```r
library(mylm)
model3 <- mylm(wages ~ sex + age + language + I(education^2), data = SLID)
summary.mylm(model3)
```

```
## Call:
## mylm(formula = wages ~ sex + age + language + I(education^2),
##     data = SLID)
##
## Residuals:
## Min         1Q         Median     3Q         Max
## -27.1712   -4.2762    -0.7631    3.2176     35.9289
##
## Coefficients:
##                   Estimate       Std. Error     z value        Pr(>|z|)
## (Intercept)      -1.875531      0.4404         -4.2587        2e-05 ***
## sexMale          3.4087         0.2084         16.3529        0.0000 ***
## age              0.248625       0.0087         28.6973        0.0000 ***
## languageFrench   -0.07553202    0.4252         -0.1776        0.8590
## languageOther    -0.1345402     0.3232         -0.4163        0.6772
```

```
## I(education^2)   0.03481515     0.0013        26.9873       0.0000 ***
##
## Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
## Residual standard error:  6.578 on  3980 degrees of freedom
## Multiple R-squared:  0.3022 ,  Adjusted R-squared:  0.3012
## F-statistic:  344.8 on 5 and 3980 DF, p-value: 0.000
```

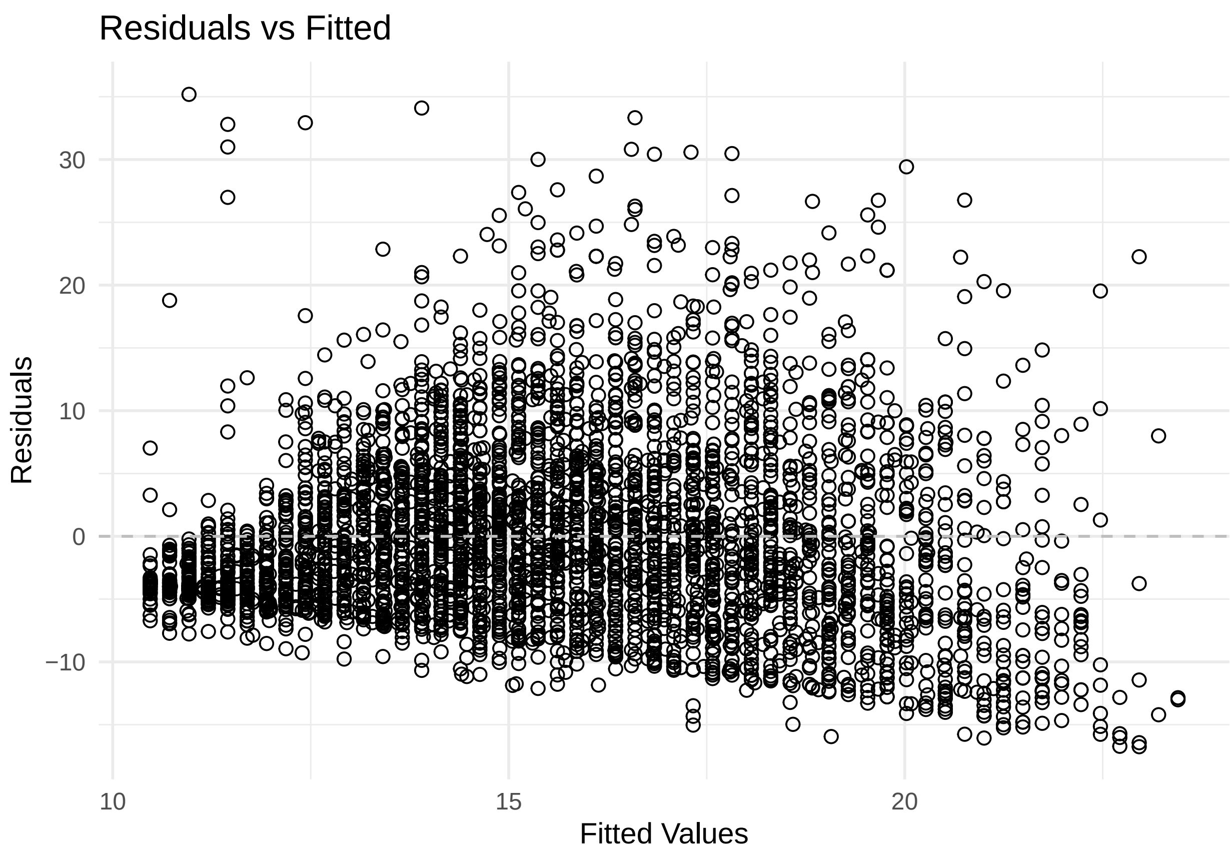
```r
plot.mylm(model3)
```



That sex/gender has a significant influence on the wages as well as age. Language has no significant effect and the squared education has an effect but a really small one. The model explains about 30% of the variance. We could leave out the language because it is unnecessary.

```
## Call:
## mylm(formula = wages ~ language + age + language * age, data = SLID)
##
## Residuals:
## Min       1Q        Median    3Q        Max
## -16.751   -4.832    -1.412    3.938     35.187
##
## Coefficients:
##                        Estimate       Std. Error     z value        Pr(>|z|)
## (Intercept)            6.555794       0.4107         15.9612        0.0000 ***
## languageFrench         2.860625       1.5963         1.7921         0.0731 .
## languageOther          0.8486213      1.2353         0.6870         0.4921
## age                    0.2448516      0.0107         22.9072        0.0000 ***
## languageFrench:age     -0.08392752    0.0405         -2.0743        0.0381 *
```

```
## languageOther:age    -0.03701381    0.0293        -1.2614       0.2072
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  7.34 on  3980 degrees of freedom
## Multiple R-squared:  0.1312 ,  Adjusted R-squared:  0.1299
## F-statistic:  120.2 on 5 and 3980 DF, p-value: 0.000
```



Language is still insignificant, age has an effect on wages. Also the interaction of languageFrench and age is significant. But the interaction term has only a small estimate. The model explains about 13% of the variance. In this model, I probably would add sex again, because it seems to explain a large proportion of variance.

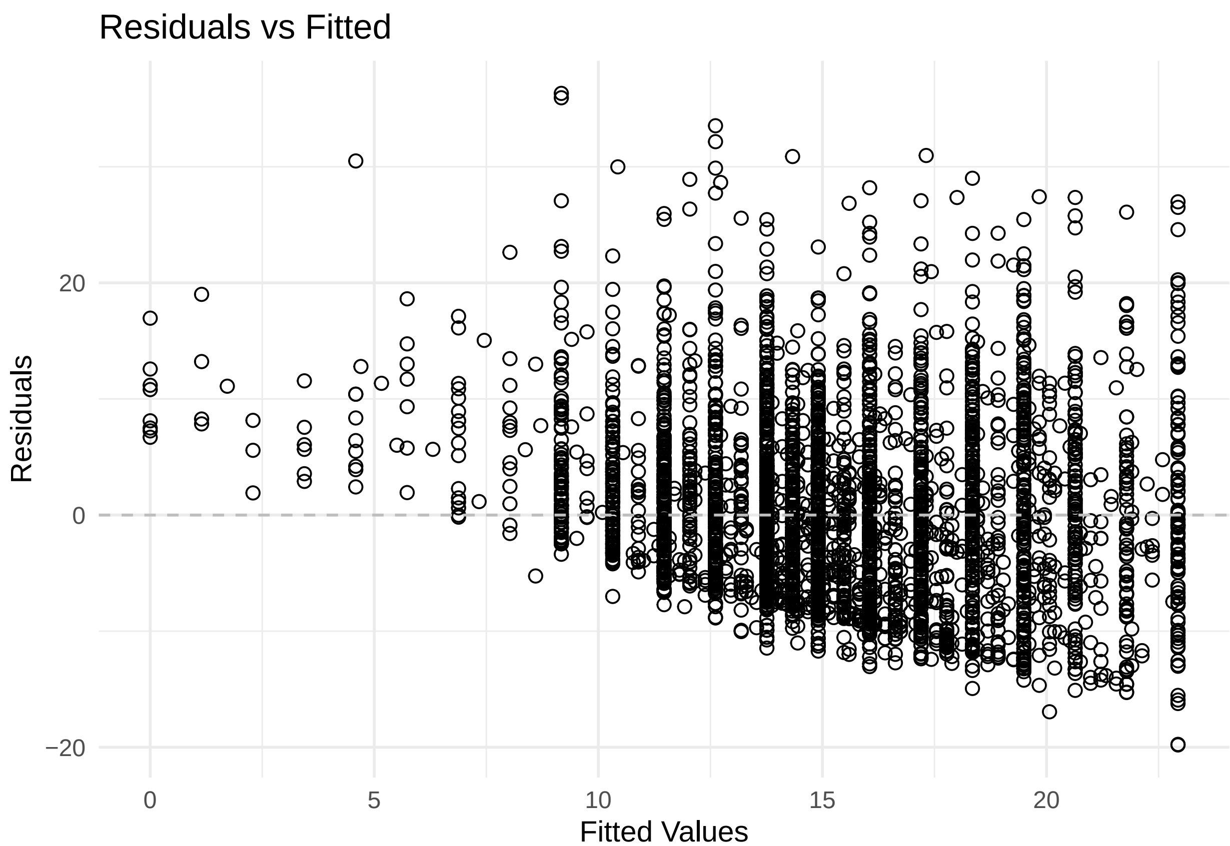
```r
library(mylm)
model5 <- mylm(wages ~ education - 1, data = SLID)
summary.mylm(model5)
```

```
## Call:
## mylm(formula = wages ~ education - 1, data = SLID)
##
## Residuals:
## Min        1Q         Median     3Q         Max
## -19.8039   -5.3421    -0.6624    4.4646     36.3264
##
## Coefficients:
##              Estimate    Std. Error   z value     Pr(>|z|)
## education    1.146697    0.0088       130.7776    0.0000 ***
```

```
## Warning in pf(F_stat, df1 = object$rank - 1, df2 = object$dof_residuals): NaNs
```

```
## produced
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error:  7.573 on  3985 degrees of freedom
## Multiple R-squared:  0.07389 ,  Adjusted R-squared:  0.07366
## F-statistic:  Inf on 0 and 3985 DF, p-value: NaN
```

```
plot.mylm(model5)
```



As there is no intercept the for a education of zero there is also a wage of zero now. The parameter for education is greater than with the intercept, as it has to compensate for the missing intercept. Also, the R^2 is smaller. We would recommend adding the intercept.

```
# Select Build, Build and reload to build and lode into the R-session.

library(stringr)

mylm <- function(formula, data = list(), contrasts = NULL, ...){
  # Extract model matrix & responses
  mf <- model.frame(formula = formula, data = data)
  X  <- model.matrix(attr(mf, "terms"), data = mf, contrasts.arg = contrasts)
  y  <- model.response(mf)
  terms <- attr(mf, "terms")


  # Add code here to calculate coefficients, residuals, fitted values, etc...
  # coefficients
```

```r
  coeff <- solve(t(X) %*% X) %*% t(X) %*% y
  coeff_list <- as.list(coeff[,1])

  # Assign names to coefficients
  names(coeff_list) <- colnames(X)

  # fitted values
  fitted_values <- X %*% coeff
  #residuals
  residuals <- y - fitted_values


  TSS <- sum((y-mean(y))^2)

  # and store the results in the list est
  est <- list(terms = terms, model = mf)

  # Store call and formula used
  est$call <- match.call()
  est$formula <- formula
  est$coeff <- coeff_list
  est$rank <- length(colnames(X))
  est$fitted_values <- fitted_values
  est$residuals <- residuals

  est$dof_residuals <- nrow(X) - length(colnames(X)) - 1
  est$data_matrix <- X
  est$TSS <- TSS


  # Set class name. This is very important!
  class(est) <- 'mylm'

  # Return the object with all results
  return(est)
}

print.mylm <- function(object, ...){
  # Code here is used when print(object) is used on objects of class "mylm"
  # Useful functions include cat, print.default and format
  variable_names = all.vars(object$formula)

  cat('Call:\n')
  print(object$call)
  cat('\nCoefficients:\n')
  for (name in names(object$coeff)) {
    cat(name, ': ')
    cat(format(object$coeff[[name]], digits = 4, nsmall = 4), '\n')
  }
}

summary.mylm <- function(object, ...){
  # Code here is used when summary(object) is used on objects of class "mylm"
```

```r
# Useful functions include cat, print.default and format

#
X <- object$data_matrix
RSS <- sum(object$residuals^2)
sigma2 <- RSS/object$dof_residuals
XtX_inv <- solve(t(X)%*%X)
cov_matrix <- sigma2*XtX_inv
stderr <- sqrt(diag(cov_matrix))


# z and p values
z <- as.numeric(object$coeff) / as.numeric(stderr)
p <- 2 * (1 - pnorm(abs(z)))

# significance levels
sig_level <- list()
for (value in p) {
  # Determine the significance level and append to the list
  if (value < 0.001) {
    sig_level[[length(sig_level) + 1]] <- '***'
  } else if (value < 0.01) {
    sig_level[[length(sig_level) + 1]] <- '**'
  } else if (value < 0.05) {
    sig_level[[length(sig_level) + 1]] <- '*'
  } else if (value < 0.1) {
    sig_level[[length(sig_level) + 1]] <- '.'
  } else {
    sig_level[[length(sig_level) + 1]] <- ' '
  }
}


## Call
cat('Call:\n')
print(object$call)

## Residuals
# set up values
summary_residuals <- c(
  Min = min(object$residuals),
  Q1 = quantile(object$residuals, 0.25),
  Median = median(object$residuals),
  Q3 = quantile(object$residuals, 0.75),
  Max = max(object$residuals)
)
formatted_residuals <- format(summary_residuals, digits = 4, nsmall = 3,justify = "right", trim = TRU
max_width <- max(nchar(formatted_residuals))
formatted_residuals <- format(summary_residuals, digits = 4, nsmall = 3, justify = "right", trim = TRU

# printing
cat('\nResiduals:\n')
```

```
  cat(str_pad("Min", max_width+2, side = 'right'),
      str_pad("1Q", max_width+2, side = 'right'),
      str_pad("Median", max_width+2, side = 'right'),
      str_pad("3Q", max_width+2, side = 'right'),
      str_pad("Max", max_width+2, side = 'right'), "\n")
  cat(str_pad(formatted_residuals[1], max_width+2, side = 'right'),
      str_pad(formatted_residuals[2], max_width+2, side = 'right'),
      str_pad(formatted_residuals[3], max_width+2, side = 'right'),
      str_pad(formatted_residuals[4], max_width+2, side = 'right'),
      str_pad(formatted_residuals[5], max_width+2, side = 'right'), "\n")




  cat('\nCoefficients:\n')
  max_name = max(nchar(names(object$coeff)))
  formatted_coeff<- format(object$coeff, nsmall = 4,justify = "right", trim = TRUE)
  max_width <- max(nchar(formatted_coeff))
  formatted_coeff <- format(object$coeff, nsmall = 4, justify = "right", trim = TRUE)


  cat(strrep(" ", max_name+2),
      str_pad('Estimate', max_width+3, 'right'),
      str_pad('Std. Error', max_width+3, 'right'),
      str_pad("z value", max_width+3, 'right'),
      str_pad( "Pr(>|z|)", max_width+3, 'right'), '\n')
  i <- 1
  for (name in names(object$coeff)) {
    cat(str_pad(name, max_name+3, 'right'))
    cat(
        str_pad(formatted_coeff[[name]], max_width+3, 'right'),
        str_pad(format(stderr[i], digits = 1, nsmall = 4, justify = "right", trim = TRUE), max_width+3,
        str_pad(format(z[i], digits = 1, nsmall = 4, justify = "right", trim = TRUE), max_width+3, 'rig
        str_pad(paste(format(p[i], digits = 1, nsmall = 4, justify = "right", trim = TRUE), sig_level[i
        '\n')
    i <- i+1
  }

  R_sqrd <- 1-(RSS/object$TSS)
  F_stat <- ((object$TSS-RSS)/(object$rank-1))/(RSS/object$dof_residuals )
  p_value_f <- 1 - pf(F_stat, df1 = object$rank - 1, df2 = object$dof_residuals)

  cat('\nSignif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1\n')

  cat('Residual standard error: ', format(sqrt(sigma2), digits=4), 'on ', object$dof_residuals,'degrees
  cat('Multiple R-squared: ', format(R_sqrd, digits=4), ', ', 'Adjusted R-squared: ', format(1-((1-R_sq
  cat('F-statistic: ', format(F_stat,digits = 1, nsmall = 1), 'on',object$rank-1, 'and',object$dof_resi

}

plot.mylm <- function(object, ...){
  # Code here is used when plot(object) is used on objects of class "mylm"
```

```r
  library(ggplot2)
  # ggplot requires that the data is in a data.frame, this must be done here
  data_plot <- data.frame(Fitted = object$fitted_values, Residuals=object$residuals)

    ggplot(data_plot, aes(x=Fitted, y=Residuals)) + geom_point(shape = 1, size = 2) +
      ggtitle('Residuals vs Fitted') +
      labs(x = 'Fitted Values', y = 'Residuals') +
      geom_hline(yintercept = 0, linetype = "dashed", color = "grey") +
      theme_minimal()

  # if you want the plot to look nice, you can e.g. use "labs" to add labels, and add colors in the geo

}

anova.mylm <- function(object, ...){
  # Code here is used when anova(object) is used on objects of class "mylm"

  # Components to test
  comp <- attr(object$terms, "term.labels")

  # Name of response
  response <- deparse(object$terms[[2]])

  # Total Sum of Squares (TSS)
  TSS <- sum((object$model[[response]] - mean(object$model[[response]]))^2)

  # Fit the sequence of models
  txtFormula <- paste(response, "~", sep = "") # for the formula for lm
  print(txtFormula)
  model <- list()
  RSS <- numeric(length(comp) + 1)  # empty to store RSS
  df <- numeric(length(comp) + 1)   # empty to store df

  # First model (only intercept)
  RSS[1] <- TSS
  df[1] <- nrow(object$model) - 1


  # Fit the sequence of models
  txtFormula <- paste(response, "~", sep = "")
  model <- list()
  for(numComp in 1:length(comp)){
    if(numComp == 1){
      txtFormula <- paste(txtFormula, comp[numComp])
    }
    else{
      txtFormula <- paste(txtFormula, comp[numComp], sep = "+")
    }
    formula <- formula(txtFormula)
    model[[numComp]] <- lm(formula = formula, data = object$model)
    # Fit the new model and calculate RSS
    model[[numComp]] <- lm(formula = formula, data = object$model)
    RSS[numComp + 1] <- sum(model[[numComp]]$residuals^2)
```

```r
  df[numComp + 1] <- model[[numComp]]$df.residual
}
# empty list to store values
anova_table <- list()

# Loop through the models and calculate stats
for (numComp in 1:length(comp)) {
  # Calculate difference in RSS
  SS_diff <- RSS[numComp] - RSS[numComp + 1]
  df_diff <- df[numComp] - df[numComp + 1]
  MS_diff <- SS_diff / df_diff  # Mean square for the model
  MS_residual <- RSS[numComp + 1] / df[numComp + 1]  # for Chi^2 statistic

  # Chi^2 statistic
  chi_sq <- SS_diff / MS_residual

  # P-value from Chi-squared distribution
  p_value <- pchisq(chi_sq, df_diff, lower.tail = FALSE)

  # Store the values in the list
  anova_table[[numComp]] <- c(df_diff, SS_diff, MS_diff, chi_sq, p_value)
}

# Convert the list to df
anova_df <- as.data.frame(do.call(rbind, anova_table))
colnames(anova_df) <- c("Df", "Sum_Sq", "Mean_Sq", "Chi^2", "Pr(>Chi)")

anova_df[["Sum_Sq"]] <- round(anova_df[["Sum_Sq"]], 0)
anova_df[['Mean_Sq']] <- round(anova_df[['Mean_Sq']], 0)
anova_df[['Chi^2']] <- round(anova_df[['Chi^2']], 3)
anova_df[['Pr(>Chi)']] <- round(anova_df[['Pr(>Chi)']], 3)


# Define the significance function
get_significance <- function(p_value) {
  if (p_value < 0.001) {
    return("***")
  } else if (p_value < 0.01) {
    return("**")
  } else if (p_value < 0.05) {
    return("*")
  } else if (p_value < 0.1) {
    return(".")
  } else {
    return(" ")
  }
}
anova_df$Signif <- sapply(as.numeric(anova_df[['Pr(>Chi)']]), get_significance)

# Find max for formatting
max_lengths <- sapply(anova_df, function(col) max(nchar(as.character(col))))
max_width <- max(max_lengths)
max_name = max(nchar(names(object$coeff)))
```

```r
  # Prepare to print ANOVA table with formatted widths
  cat('Analysis of Variance Table\n')
  cat(c('Response: ', response, '\n'), sep = '')

  cat(strrep(" ", max_name+2),
      str_pad('Df', max_width+3, 'right'),
      str_pad('Sum Sq', max_width+3, 'right'),
      str_pad("Mean Sq", max_width+3, 'right'),
      str_pad("Chi^2", max_width+3, 'right'),
      str_pad( "Pr(>Chi^2)", max_width+3, 'right'), '\n')
  i <- 1
  for (i in 1:nrow(anova_df)) {
    cat(str_pad(comp[i], max_name+3, 'right'))
    cat(
      str_pad(anova_df[i, "Df"], max_width+3, 'right'),
      str_pad(anova_df[i, "Sum_Sq"], max_width+3, 'right'),
      str_pad(anova_df[i, "Mean_Sq"], max_width+3, 'right'),
      str_pad(anova_df[i, "Chi^2"], max_width+3, 'right'),
      str_pad(paste(anova_df[i, "Pr(>Chi)"],anova_df$Signif), max_width+3, 'right'),
      '\n')
    i <- i+1
  }

  residual_SumSq <- RSS[1] - sum(as.numeric(anova_df$Sum_Sq))
  residual_Mean_Sq <- residual_SumSq/object$dof_residuals
  cat(str_pad('Residuals', max_name+2, 'right'),
      str_pad(object$dof_residuals, max_width+3, 'right'),
      str_pad(round(residual_SumSq), max_width+3, 'right'),
      str_pad(round(residual_Mean_Sq), max_width+3, 'right'))
  cat('\nSignif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1\n')
  cat('Total Sum SQ:', round(object$TSS), '\n')

  # chi squared test
  chi2_stat <- (object$TSS - residual_SumSq) / (residual_SumSq / object$dof_residuals)
  p_value_chi2 <- 1 - pchisq(chi2_stat, df = object$rank - 1)
  cat('Chi-statistic: ', format(chi2_stat,digits = 1, nsmall = 1), 'on', object$dof_residuals ,'DF, p-va

  #return(model)

}
```