Lab II Stats II

OLS in R

Martin Arvidsson

2024-11-11

Outline

As last time…

- **Part 1** – *Practical* lecture in R
- **Part 2** – Assignment time.

Data, Question & Method

Data

- Same data as last time (**child-IQ data**)
  - `ppvt` — Child's test scores at age 3
  - `hs_degree` — Whether the mother has a high-school degree
  - `momage` — Mother's age at the time she gave birth

Question to explore today:

What is the relationship between *mom's age* and *child's IQ-score*?

Method

Ordinary Least Squares (OLS) Regression

Procedural steps

1. Import & inspect data
2. Plot relationship of interest (`momage` v `ppvt`)
3. Specify & Fit OLS: $ppvt = \beta_0 + \beta_1 momage + \epsilon$
4. Interpret and check model-fit
5. Consider extensions (and repeat 3-4)

(1) Importing & inspecting data

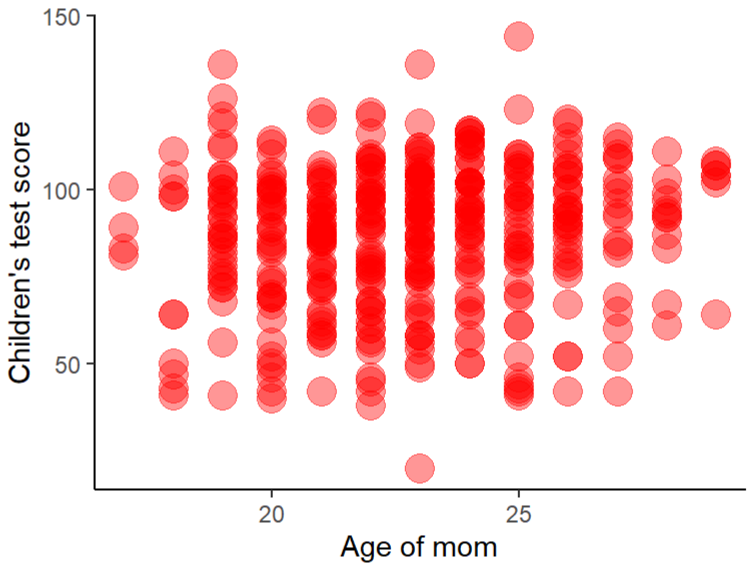
```r
# We're going to use ggplot2 later for plotting
library(ggplot2)

# Import data
df <- read.csv("../lab2/childiq2.csv")
```

```r
# Inspect data
str(df)
```

```
## 'data.frame':    400 obs. of  4 variables:
##  $ X        : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ ppvt     : int  120 89 78 42 115 97 94 68 103 94 ...
##  $ momage   : int  21 17 19 20 26 20 20 24 19 24 ...
##  $ hs_degree: int  1 0 1 0 1 0 0 1 1 1 ...
```

(2) Plot: **momage** v **ppvt**

```
ggplot(df,aes(x=momage,y=ppvt)) +
  geom_point(alpha=0.4, size=5,colour="red")+
  xlab("Age of mom")+
  ylab("Children's test score")+
  theme_classic()
```



No super-clear relationship. Perhaps weakly positive?

(3) Fit OLS

- Moving beyond plotting — **fit OLS** to find *the line* that minimizes distance between *line* and *data*
- Fitting an OLS regression in `R` is straightforward.
  - Use `lm()` — short for *linear model*
- Need to specify two arguments:
  - **Formula**: `output ~ predictors`
  - **Data**
- Recall interest: relationship between `momage` & `ppvt`

```
m1 <- lm(formula = ppvt ~ momage, data = df)
```

(4) Inspect model via `summary()`

```
summary(m1)
```

```
##
## Call:
## lm(formula = ppvt ~ momage, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.109 -11.798   2.971  14.860  55.210
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  67.7827     8.6880   7.802 5.42e-14 ***
## momage        0.8403     0.3786   2.219    0.027 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.34 on 398 degrees of freedom
## Multiple R-squared:  0.01223,    Adjusted R-squared:  0.009743
## F-statistic: 4.926 on 1 and 398 DF,  p-value: 0.02702
```

(5) Interpretation

- $\beta_0 = 67.8$:
  - When mom's age is 0, the predicted IQ-score is ~ 68
  - Of course, a mom of age 0 doesn't make sense…
- $\beta_1 = 0.85$:
  - A one-unit-increase in `momage` predicts an IQ-increase of 0.85
  - E.g. for a mom having her child at age 23, the model predicts an IQ score of $67.8 + 23 \times 0.85 = 87.35$
  - **However**: before we make this interpretation, we should consider the **uncertainity** of our estimate!
  - If β_1 is found **not significantly different from 0**, the *observed effect-size* is consistent with a *true effect-size* of 0 $\rightarrow$ careful with interpretation.

(6) Significance test for β_1

- `t-test` for the slope of `momage` ($\beta_1 = 0.84$):
  - $H_0 : \beta_1 = 0$
  - $H_1 : \beta_1 \neq 0$
- **p-value** = 0.027 — represents the probability of observing the observed (or a more extreme) effect given that H_0 (no effect) is true.
  - Using the common *significance level* 0.05, we **reject** H_0.
  - Logic: very *unlikely* to observe $\beta_1 = 0.84$ given H_0

$\rightarrow$ Conclude: β_1 is significantly different from 0

(7) Plot predicted values over data

```
# Data to predict for
x <- seq(from = min(df$momage), to = max(df$momage), by = 1)
print(x)
```

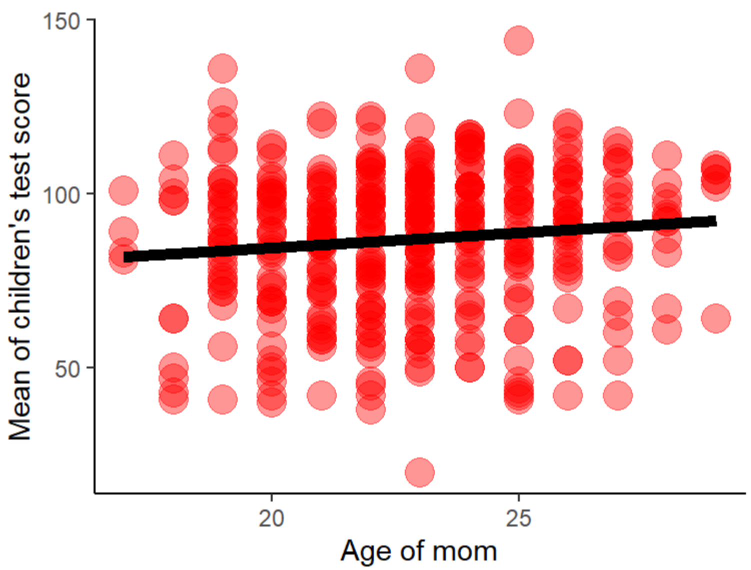
```
##  [1] 17 18 19 20 21 22 23 24 25 26 27 28 29
```

```
# Prediction: y_hat = b0 + b1 * x
line_data <- data.frame(x = x, y = m1$coefficients[1] +
                                   m1$coefficients[2] * x)
head(line_data,5)
```

```
##    x        y
## 1 17 82.06732
## 2 18 82.90759
## 3 19 83.74787
## 4 20 84.58814
## 5 21 85.42841
```

(7) Plot predicted values over data

```
x <- seq(from = min(df$momage), to = max(df$momage), by = 1)
line_data <- data.frame(x=x, y=m1$coefficients[1] + m1$coefficients[2]*x)
ggplot(unique(df,by="mean_ppvt_by_momage"),aes(x=momage,y=ppvt))+
  geom_point(alpha=0.4, size=5,colour="red")+
  xlab("Age of mom") +
  ylab("Mean of children's test score")+
  theme_classic() +
  geom_line(data=line_data,aes(x=x,y=y),linewidth=2,color="black")
```



(8) R^2

```
summary(m1)
```

```
##
## Call:
## lm(formula = ppvt ~ momage, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.109 -11.798   2.971  14.860  55.210
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  67.7827     8.6880   7.802 5.42e-14 ***
## momage        0.8403     0.3786   2.219    0.027 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.34 on 398 degrees of freedom
## Multiple R-squared:  0.01223,    Adjusted R-squared:  0.009743
## F-statistic: 4.926 on 1 and 398 DF,  p-value: 0.02702
```

- $R^2 = \frac{Explained\ variance}{Total\ variance} = \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2}$
- Interpretation: *proportion of variance explained*.

(9) Linear/Non-linear relationship?

- Although the plot is not very suggestive of this — suppose we're interested in exploring the possibility of a **non-linear** relationship between `ppvt` and `momage`
- To do this, we can:
  - Incorporate a **squared** version of `momage`: $momage^2$
  - Note: we add this using the `I()` function to inhibit R from interpretting `^` as a `formula`-specific-operator.

```
m2 <- lm(formula = ppvt ~ momage + I(momage^2), data = df)
```

(9) Inspect model `m2`

```
summary(m2)
```

```
##
## Call:
## lm(formula = ppvt ~ momage + I(momage^2), data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -66.500 -11.748   3.044  14.581  55.494
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 111.57981   63.77234   1.750   0.0809 .
## momage       -3.01631    5.57597  -0.541   0.5888
## I(momage^2)   0.08373    0.12079   0.693   0.4886
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.35 on 397 degrees of freedom
## Multiple R-squared:  0.01342,    Adjusted R-squared:  0.008449
## F-statistic:   2.7 on 2 and 397 DF,  p-value: 0.06844
```

(9) Inspect model `m2`

- We find:
  - Worse data-fit (lower **Adjusted** R^2): 0.0097 vs. 0.0084
  - Insignificant linear & squared-terms

$\longrightarrow$ **drop** *squared* term!

(10) What about education?

- For each mom we know whether she **finished high-school**
- Well-known: **positive correlation** between *education-level* and the *age at which people have kids*.
- Can we see this in our data?

```
aggregate(x = df$momage, by = list(df$hs_degree), FUN=mean)
```

```
##   Group.1        x
## 1       0 21.58824
## 2       1 23.11429
```

- Yes, on average, moms that **finished high-school** have children *1.5 years later*

$\longrightarrow$ **Q**: How does `momage` relate to `ppvt`, *cond. on* `hs_degree`?

(11) Adding **hs_degree** to model

- `hs_degree` is a categ-variable: let's make it into a **factor**

```
# Making hs_degree into a factor
df$hs_degree <- factor(df$hs_degree, levels = c(0,1))
# Fit model incorporating "hs_degree"
m3 <- lm(formula = ppvt ~ momage + hs_degree, data = df)
```

(11) Inspecting `m3`

```
summary(m3)
```

```
##
## Call:
## lm(formula = ppvt ~ momage + hs_degree, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.132 -12.630   1.818  14.833  58.809
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  67.9739     8.5290   7.970 1.70e-14 ***
## momage        0.4851     0.3821   1.270    0.205
## hs_degree1   10.0348     2.5093   3.999 7.58e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.97 on 397 degrees of freedom
## Multiple R-squared:  0.05048,    Adjusted R-squared:  0.04569
## F-statistic: 10.55 on 2 and 397 DF,  p-value: 3.427e-05
```

(11) Inspecting `m3`

- Including `hs_degree` leads to a big improvement in fit
  - Adjusted $R^2 = 0.046$ (vs 0.009)
  - The model now explains $\sim 4.5\%$ of the variance in the data (still quite low)
- β_2
  - Mom having a high-school degree predicts an increase in IQ score by ~ 10 points
- β_1
  - Controlling for `hs_degree`, the slope for `momage` is (i) reduced by approx. half, and (ii) no longer signficantly different from 0

(12) Could importance of age differ depending on education-level?

- If one thinks that `momage` could have a different effect on `ppvt` depending on the *education level* (`hs_degree`), one could additionally include a so-called **interaction** between `momage` and `hs_degree`:

```
# Fit model incorporating interaction effect between momage and hs_degree
m4 <- lm(formula = ppvt ~ momage + hs_degree + momage*hs_degree, data = df)
```

(12) Inspecting `m4`

```
summary(m4)
```

```
##
## Call:
## lm(formula = ppvt ~ momage + hs_degree + momage * hs_degree,
##     data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.696 -12.407   2.022  14.804  54.343
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       105.2202    17.6454   5.963 5.49e-09 ***
## momage             -1.2402     0.8113  -1.529   0.1271
## hs_degree1        -38.4088    20.2815  -1.894   0.0590 .
## momage:hs_degree1   2.2097     0.9181   2.407   0.0165 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.85 on 396 degrees of freedom
## Multiple R-squared:  0.06417,    Adjusted R-squared:  0.05708
## F-statistic: 9.051 on 3 and 396 DF,  p-value: 8.276e-06
```

(12) Inspecting `m4`

- Slightly improved fit
  - Adjusted $R^2 = 0.056$ (vs 0.046)
- Interpretation is difficult, e.g.
- $\beta_2 = -38.41$ (`hs_degree`):
  - Difference between the predicted IQ-scores for children whose mothers *finished with a hs-diploma* (and were of age 0), and children whose mothers *did not finish with a hs-diploma* (and were of *age 0*).
  - No mothers of age 0, and thus not interpretable.
- Let us instead plot the *predicted values* and compare!

(12) Plotting interactions

```
# Create data to predict for
pred_data <- expand.grid(momage=seq(from=min(df$momage),
                                    to=max(df$momage),by=1),
                         hs_degree = as.factor(c(0,1)))
head(pred_data,2)
```

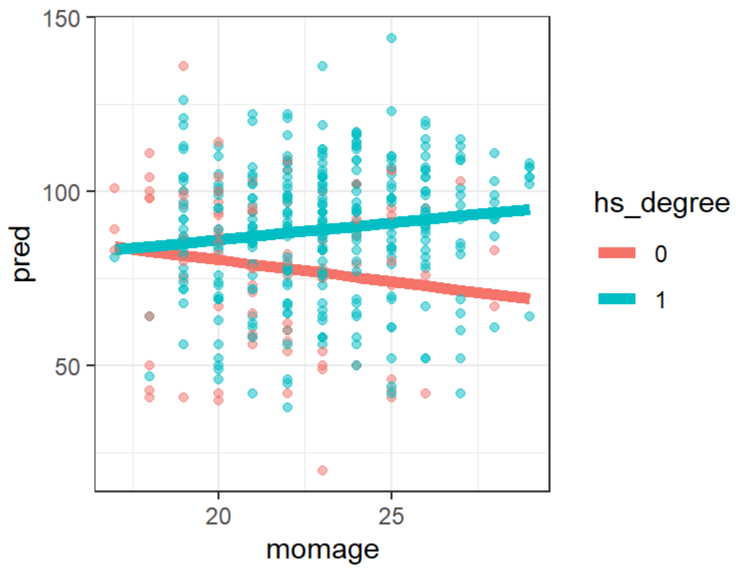
```
##   momage hs_degree
## 1     17         0
## 2     18         0
```

```
# Another way to predict
pred_data$pred <- predict(m4,newdata=pred_data)
head(pred_data,2)
```

```
##   momage hs_degree     pred
## 1     17         0 84.13726
## 2     18         0 82.89709
```

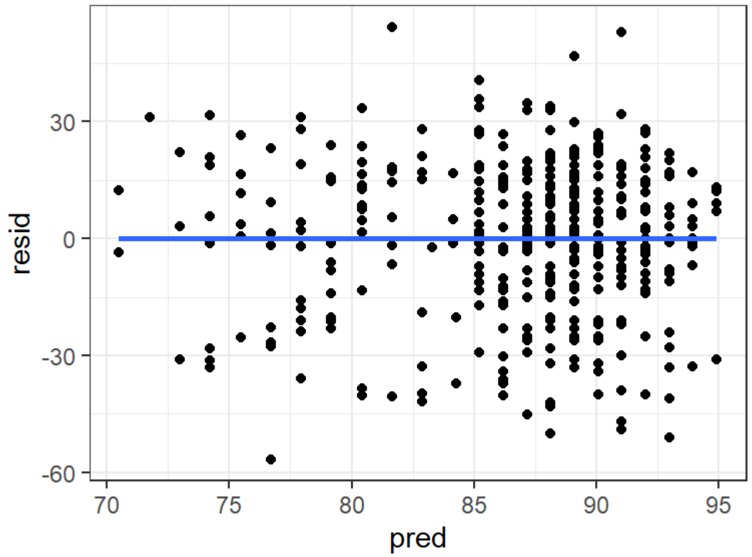
(12) Plotting interactions

```
pred_data <- expand.grid(momage=seq(min(df$momage),max(df$momage),1),
                         hs_degree = as.factor(c(0,1)))
pred_data$pred <- predict(m4,newdata=pred_data)
ggplot(pred_data,aes(x=momage,y=pred,group=hs_degree,color=hs_degree))+
geom_line(size=2)+
geom_point(data=df,aes(x=momage,y=ppvt,color=hs_degree),linewidth=1.5,alpha=0.5) +
theme_bw()
```



(13) Validity of `m4`? Check residuals.

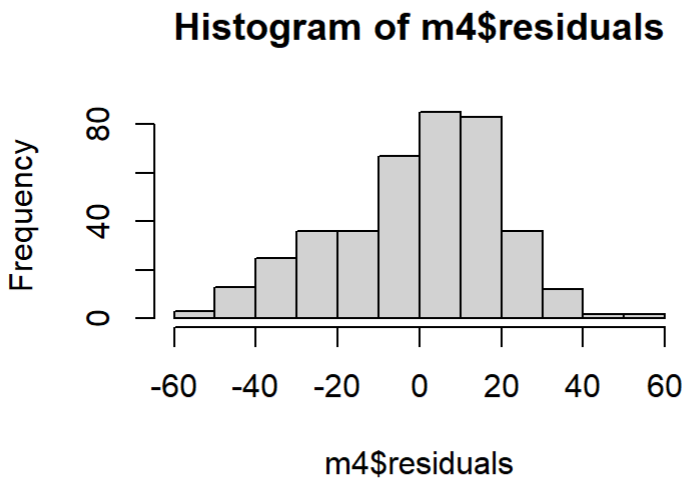
```
# (1) Check Heteroscedasticity (constant variance)
resid_df <- data.frame(pred=m4$fitted.values, resid=m4$residuals)
ggplot(resid_df,aes(x=pred,y=resid)) +
  geom_point() + stat_smooth(method = 'lm',se = FALSE) +
  theme_bw()
```



Looks good. Zero-centered with constant variance.

(13) Validity of `m4`? Check residuals.

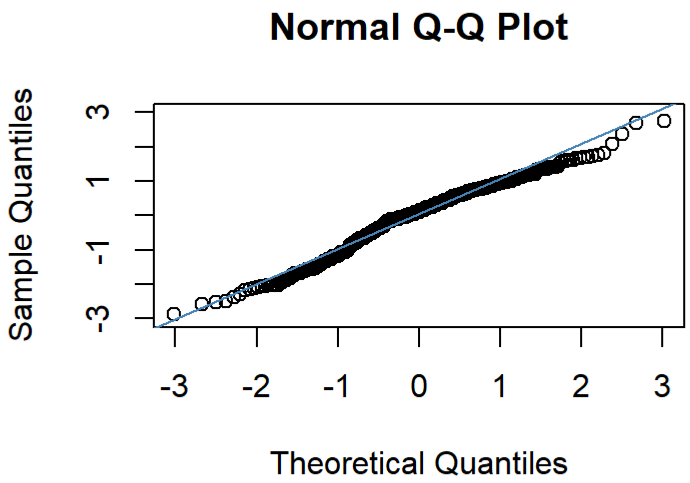
```
# (2) Are the residuals normally distributed?
# - Alt. 1: Histogram
hist(m4$residuals)
```



Looks OK. Pretty bell-shaped around 0.

(13) Validity of `m4`? Check residuals.

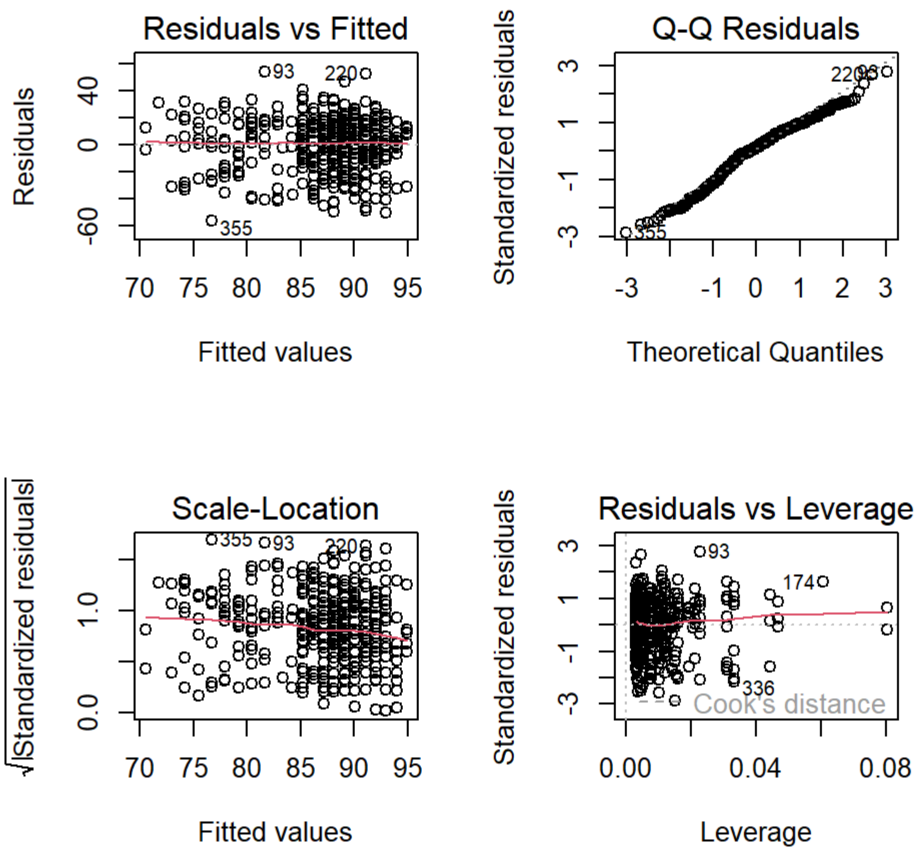
```
# (2) Are the residuals normally distributed?
# - Alt. 2: QQ-plot
std_residuals <- (m4$residuals - mean(m4$residuals)) / sd(m4$residuals)
qqnorm(std_residuals,ylim = c(-3,3))
qqline(std_residuals, col = 'steelblue')
```



Quantiles of *sample distribution* match quantiles of *Normal*.

(13) Validity of `m4`? Check residuals.

```
# Alternatively, use plot() function of model object
par(mfrow=c(2,2))
plot(m4)
```



Assignment time!