Dec 5 Lab - regression assumptions and multiple regression

Bella Fascendini

2024-12-04

The focus of this lab is on evaluating data to check if linear regression assumptions are being met and fitting a multiple regression model to data.

We'll be working with the 'Parenthood' dataset, created by Dr. Danielle Navarro, which presents an interesting application of multiple regression.

The dataset captures 100 days of supposed observations of sleep patterns and mood. She quantified her daily grumpiness on a scale ranging from 0 (not at all grumpy) to 100 (extremely grumpy).

Each day, she recorded three key variables: her grumpiness level, the amount of sleep she got, and the amount of sleep her infant son got.

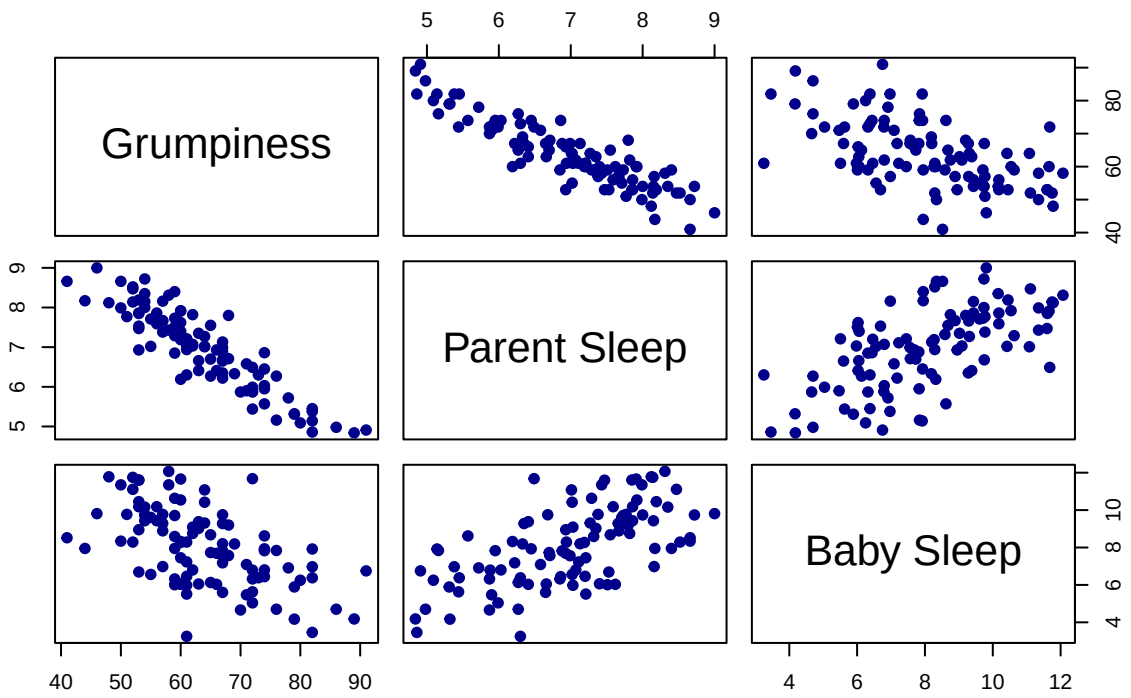
First, let's load the data

```
load("parenthood.Rdata")
names(parenthood)<- c("dan.sleep", "baby.sleep", "dan.grump", "day")
head(parenthood)
```

```
##   dan.sleep baby.sleep dan.grump day
## 1      7.59      10.18        56   1
## 2      7.91      11.66        60   2
## 3      5.14       7.92        82   3
## 4      7.71       9.61        55   4
## 5      6.68       9.75        67   5
## 6      5.99       5.04        72   6
```

To get a good idea of the dataset, it's worth visualizing it using multiple scatterplots.

```
pairs(parenthood[,c("dan.grump", "dan.sleep", "baby.sleep")],
      main = "Relationships between Sleep and Grumpiness",
      pch = 19,  # Solid circles for points
      col = "darkblue",  # Point color
      labels = c("Grumpiness", "Parent Sleep", "Baby Sleep"))
```



One possible explanation of Danielle's grumpiness is that it is influence by the amount of sleep she has had, and the amount of sleep that her baby has had. What would be a simple multiple regression model that captures this? Define and fit it to the data with lm()

(Hint: 2 predictors)

```
sleep_model <- lm(dan.grump ~ dan.sleep + baby.sleep, data = parenthood)
summary(sleep_model)
```

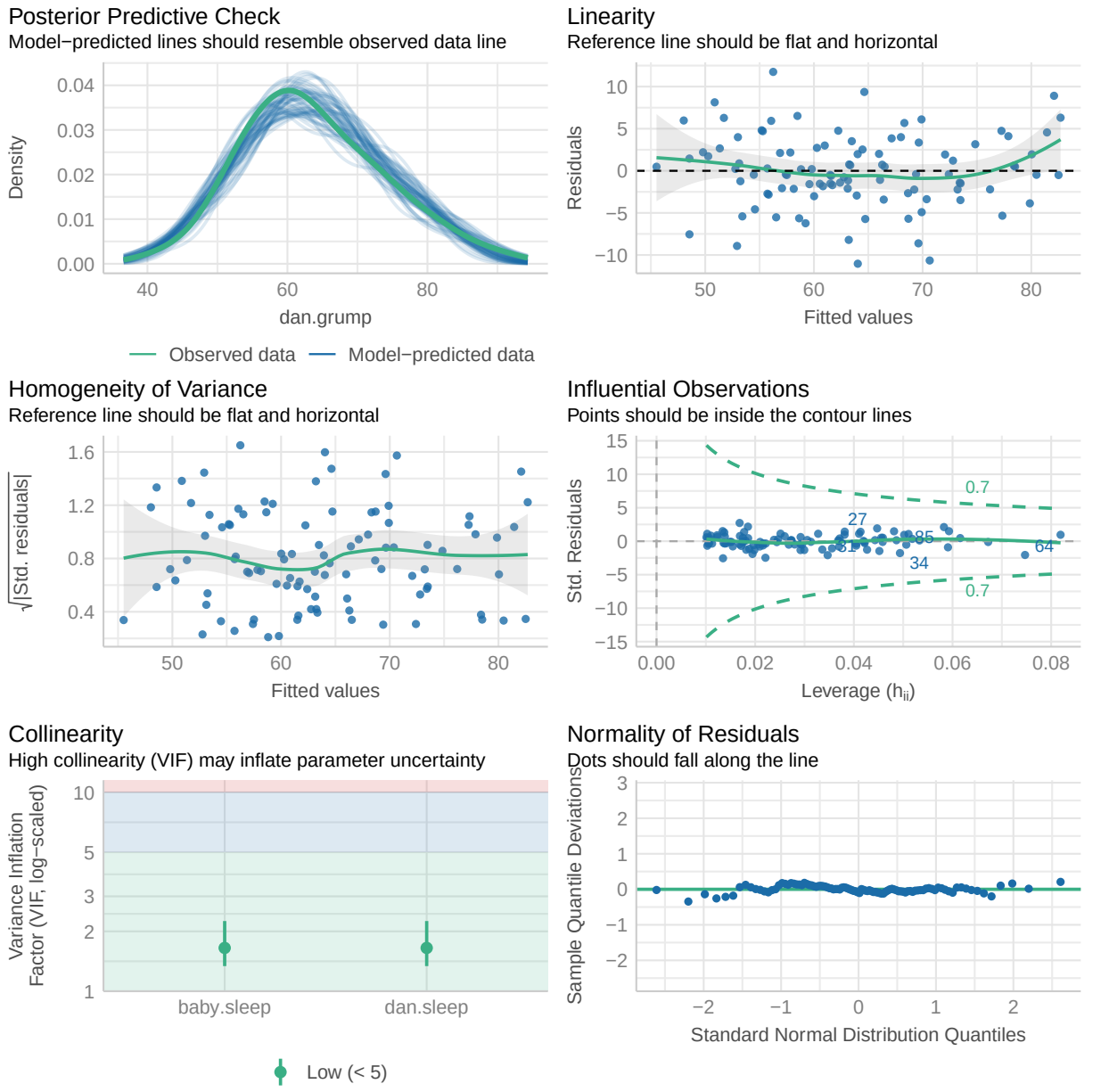
```
##
## Call:
## lm(formula = dan.grump ~ dan.sleep + baby.sleep, data = parenthood)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.0345  -2.2198  -0.4016   2.6775  11.7496
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 125.96557    3.04095  41.423   <2e-16 ***
## dan.sleep    -8.95025    0.55346 -16.172   <2e-16 ***
## baby.sleep    0.01052    0.27106   0.039    0.969
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.354 on 97 degrees of freedom
```

```
## Multiple R-squared:  0.8161, Adjusted R-squared:  0.8123
## F-statistic: 215.2 on 2 and 97 DF,  p-value: < 2.2e-16
```

Having fit the model to the data, we now have access to both, the fitted values and the residuals. This means that the regression assumptions can also be assessed :)

Use the model check function from the performance package to assess model assumptions visually. It takes as input the fitted model and the list of tests. Only giving the model as input however produces a more exhaustive lists of checks.

```
library(performance)
library(see)
check_model(sleep_model)
```



What do you observe? For any explanations of the observations make sure that you mention it in terms of the variables on the x and y axes of the plots that are generated.

```
#The model appears to be doing a reasonably good job because the blue predicted lines closely follow th
```

Note that while these visual model checks give you a qualitative idea of whether the assumptions are met, each of the regression assumptions also has corresponding quantitative tests. These tests are often based on appropriate hypotheses tests that are relevant for a given assumption.

E.g. a test for normality of residuals could be based on the null hypothesis that the residuals are indeed normal. But this gets rejected if the p-value generated by the test is lesser than our threshold (e.g. < 0.05). This is in fact the "Shapiro-Wilk test".

Run the shapiro test for normality of residuals. shapiro.test() takes as input the residuals of your model which can be found via the residuals() function or via relevant functions if you are using broom.

```
shapiro.test(residuals(sleep_model))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(sleep_model)
## W = 0.99228, p-value = 0.8414
```

What is the outcome of the Shapiro-Wilk normality test? Does it match you assessment (for normality) from the visual model check?

```
#The Shapiro-Wilk normality test result aligns well with what we observed in our visual check. The p-va
```

Similarly the Breusch-Pagan test – bptest() in the lmtest package, checks for the assumption of constant variance. Carry it out and see if it matches your assessment from the visual model check

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(zoo)
bptest(sleep_model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  sleep_model
## BP = 0.48868, df = 2, p-value = 0.7832
```

```
#It does match my assessment from the visual model check! The p-value of 0.7832 suggests that we fail r
```

The point to note is that there are several tests for testing assumptions as well, and sometimes the test you may want to use will depend on your modeling scenario.

Bonus Question: If the underlying research question Navarro intended to explore was whether her son's sleep patterns had any significant relationship with her grumpiness, beyond what could be explained by her own sleep patterns,how would you think of solving this question?

(Hint: this question suggests that she is thinking of two different models of the data generating process)

```
model1 <- lm(dan.grump ~ dan.sleep, data = parenthood)
model2 <- lm(dan.grump ~ dan.sleep + baby.sleep, data = parenthood)
anova(model1, model2)
```

```
## Analysis of Variance Table
##
## Model 1: dan.grump ~ dan.sleep
## Model 2: dan.grump ~ dan.sleep + baby.sleep
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     98 1838.7
## 2     97 1838.7  1  0.028576 0.0015 0.9691
```

```
#The RSS is almost identical between the models (both approximately 1838.7), and the p-value is very la
```

Having done this lab, I encourage you to go through chapter 15 of Learning Statistics with R in more detail, as it talks about many more modeling nuances.