Tutorial 2 - Linear models cont. (Solutions).

Christopher A Oldnall

Welcome to the third tutorial of the Regression and Simulation methods module. This is the next script in developing your skills in R, whilst learning how to perform analysis on linear regression models. Throughout this notebook we will be looking to get information about linear models we fit, as well as how we can consider non-linear models.

Exercise 0

Throughout remember we will need tidyverse, go ahead and do this as your first task.

Your Answer:

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

In 2004, the state of North Carolina released a large data set containing information on births recorded in this state. This data set is useful to researchers studying the relation between habits and practices of expectant mothers and the birth of their children. We will work with a random sample of observations from this data set. For a full descriptions of what each variable is, head to https://www.openintro.org/data/index.php?data=ncbirths

First, let's load the nc data set into our workspace. Notice that this is a slightly different way of loading the data in than what we did before. This is because the file we are using is an R data structure file and not a csv file - this way, we can have more information saved in the data set and need less pre-processing.

```
nc <- readRDS("/Users/chrisoldnall/Library/Mobile Documents/com~apple~CloudDocs/Teaching/SMSTC_RegAndSi
```

Exercise 1

How many cases are there in our sample? Determine whether each variable in this dataset is numerical or categorical. Following this create some plot to visually explore the distribution of mothers' smoking habits and baby weight - explain your findings. Hint: a histogram is good at showing whether something is normally distributed or not.

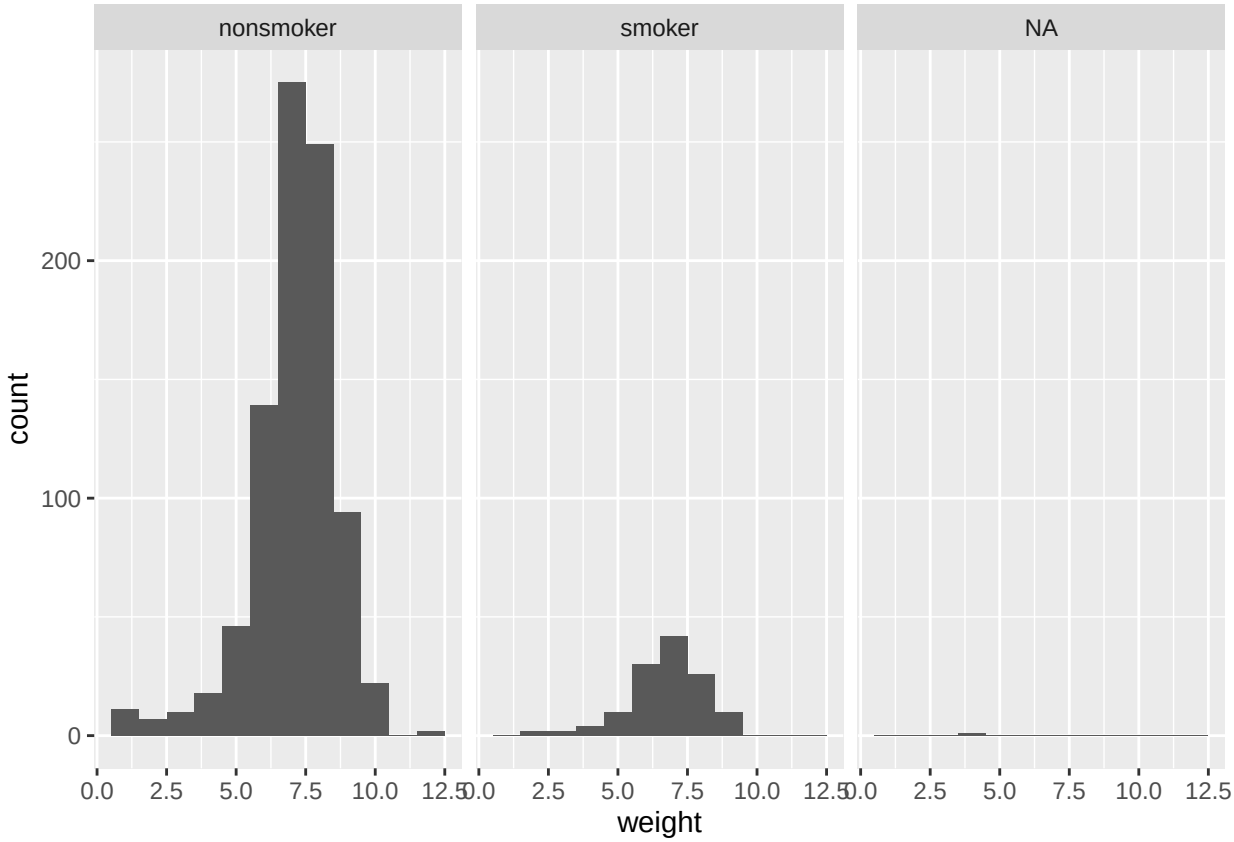
Your Answer:

```
str(nc)
```

```
## tibble [1,000 x 13] (S3: tbl_df/tbl/data.frame)
##  $ fage          : int [1:1000] NA NA 19 21 NA NA 18 17 NA 20 ...
##  $ mage          : int [1:1000] 13 14 15 15 15 15 15 15 16 16 ...
##  $ mature        : Factor w/ 2 levels "mature mom","younger mom": 2 2 2 2 2 2 2 2 2 2 ...
##  $ weeks         : int [1:1000] 39 42 37 41 39 38 37 35 38 37 ...
##  $ premie        : Factor w/ 2 levels "full term","premie": 1 1 1 1 1 1 1 2 1 1 ...
##  $ visits        : int [1:1000] 10 15 11 6 9 19 12 5 9 13 ...
##  $ marital       : Factor w/ 2 levels "married","not married": 1 1 1 1 1 1 1 1 1 1 ...
##  $ gained        : int [1:1000] 38 20 38 34 27 22 76 15 NA 52 ...
##  $ weight        : num [1:1000] 7.63 7.88 6.63 8 6.38 5.38 8.44 4.69 8.81 6.94 ...
##  $ lowbirthweight: Factor w/ 2 levels "low","not low": 2 2 2 2 2 1 2 1 2 2 ...
##  $ gender        : Factor w/ 2 levels "female","male": 2 2 1 2 1 2 2 2 2 1 ...
##  $ habit         : Factor w/ 2 levels "nonsmoker","smoker": 1 1 1 1 1 1 1 1 1 1 ...
##  $ whitemom      : Factor w/ 2 levels "not white","white": 1 1 2 2 1 1 1 1 2 2 ...
```

```
mothersmokeweight_plot <- nc %>%
    ggplot(aes(x = weight)) +
    geom_histogram(binwidth = 1) +
    facet_wrap(~ habit)

mothersmokeweight_plot
```



We see here there are 1000 cases and 13 variables within our sample; 'fage', 'mage', 'weeks', 'visits', 'gained' and 'weight' are numerical variables (with 'weight' being floating point) whilst the rest are categorical. Following this a histogram is created per smoking habit group and we see a normal distribution being formed in each group.

Exercise 2

Now go ahead and run a linear model between the weight and weeks data, save it as 'linear_model'. You can use the function 'summary()' on whatever you save your linear model as to get a range of analyses related information. Write the model it has fitted in markdown/latex and report the relevant information about the model fit (RSS). Hint: You can use the function 'anova()' to find out the RSS.

Your Answer:

```
linear_model <- nc %>%
    lm(formula = weight ~ weeks)

summary(linear_model)
```

```
##
## Call:
## lm(formula = weight ~ weeks, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5775 -0.7048 -0.0235  0.7022  4.4165
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.09529    0.46464  -13.12   <2e-16 ***
## weeks        0.34433    0.01209   28.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.119 on 996 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.449,  Adjusted R-squared:  0.4485
## F-statistic: 811.7 on 1 and 996 DF,  p-value: < 2.2e-16
```

```
anova(linear_model)
```

```
## Analysis of Variance Table
##
## Response: weight
##            Df Sum Sq Mean Sq F value    Pr(>F)
## weeks       1 1015.9 1015.86  811.74 < 2.2e-16 ***
## Residuals 996 1246.5    1.25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here we see that the model has been fitted as:

$$y_{\text{weight}} = -6.10 + 0.34 x_{\text{weeks}}.$$

The residual sum of squares is 0.449.

Exercise 3

Use the 'geom_smooth(method='lm')' command to plot the linear model of weight against weeks. Consider could there be a confounding variable in play and should we stratify our data? Fit another linear model which now depends on weeks and this other variable and see if the coefficient estimate for weeks varies from the first linear model.

Your Answer:

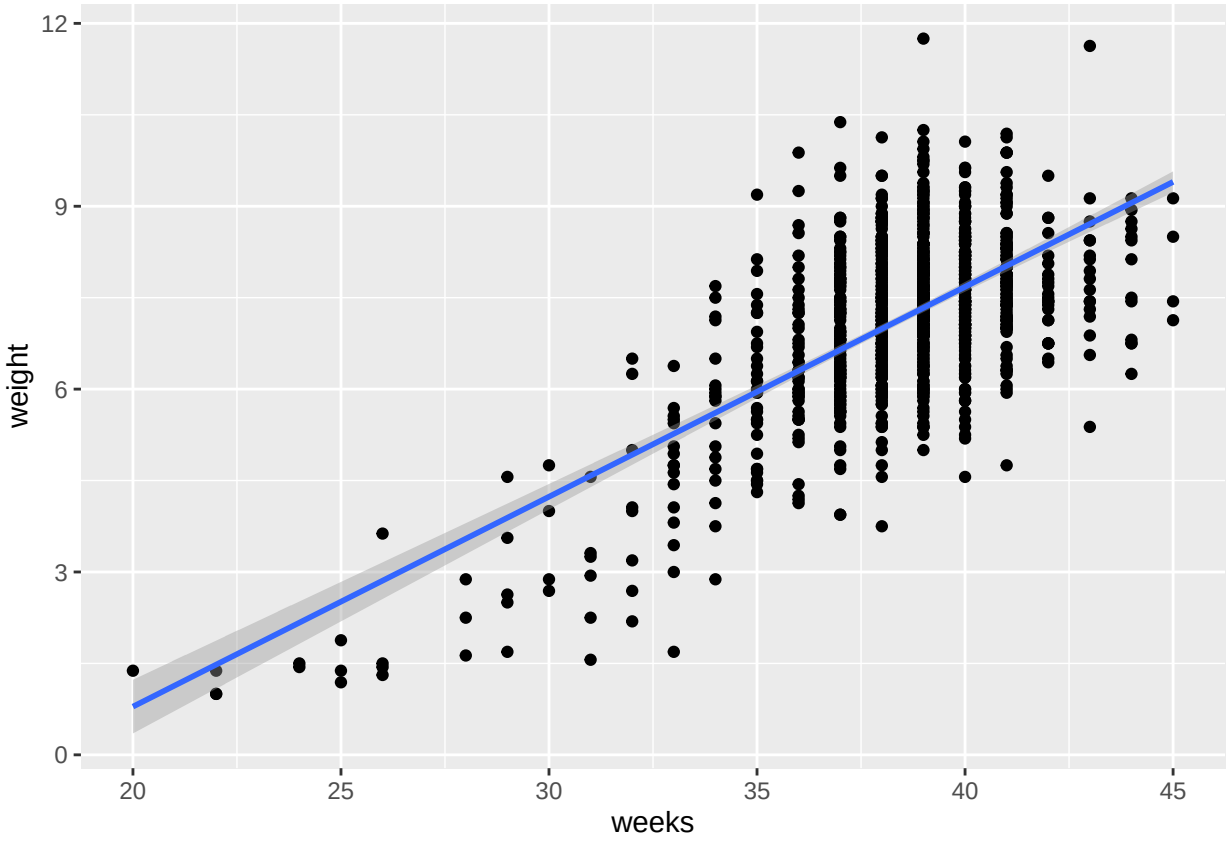
```
linearmodel_plot <- nc %>%
  ggplot(aes(x = weeks, y = weight)) +
  geom_point() +
  geom_smooth(method='lm')

linearmodel_plot
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```

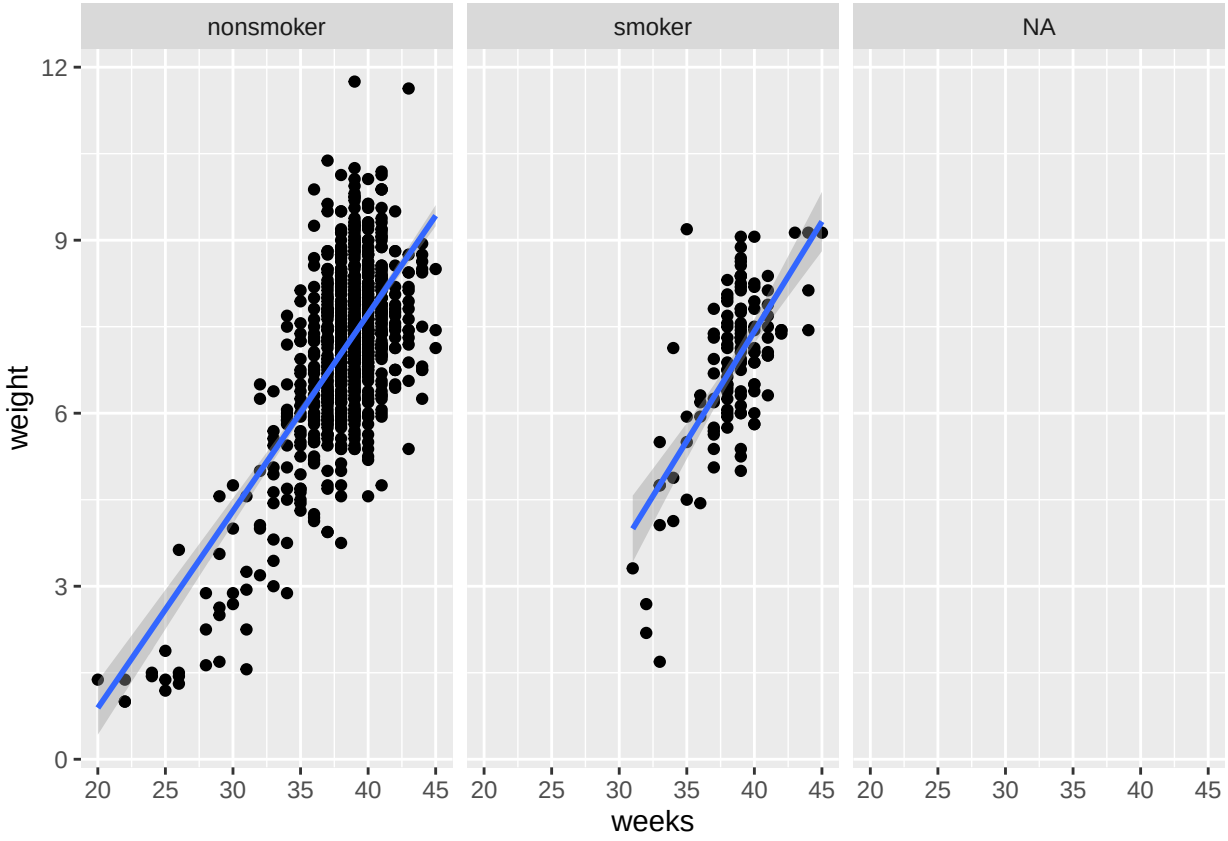



```
linearmodel_plot_faceted <- nc %>%
  ggplot(aes(x = weeks, y = weight)) +
  geom_point() +
  geom_smooth(method='lm') +
  facet_wrap(~habit)

linearmodel_plot_faceted
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_smooth()`).
## Removed 2 rows containing missing values (`geom_point()`).
```




```
linear_model2 <- nc %>%
    lm(formula = weight ~ weeks + habit)

summary(linear_model2)
```

```
##
## Call:
## lm(formula = weight ~ weeks + habit, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3788 -0.6893 -0.0344  0.7157  4.3708
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.07220    0.46228 -13.135  < 2e-16 ***
## weeks        0.34491    0.01202  28.685  < 2e-16 ***
## habitsmoker -0.35882    0.10607  -3.383 0.000746 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.113 on 995 degrees of freedom
##   (2 observations deleted due to missingness)
```

```
## Multiple R-squared:  0.4553, Adjusted R-squared:  0.4542
## F-statistic: 415.8 on 2 and 995 DF,  p-value: < 2.2e-16
```

Exercise 4

Complete the following code to calculate the F statistic between your two models and to obtain a p-value through the code provided. What does the p-value mean in this case?

```
RSS0 <- sum(residuals(linear_model)^2)
RSS1 <- sum(residuals(linear_model2)^2)
FObs <- ((RSS0 - RSS1) / 995-996) / (RSS1 / (1000 - 995))
p_value <- 1 - pf(FObs, 996, 995)

p_value
```

```
## [1] 1
```

Challenge

Take the weight and weeks data and fit a non-linear log model. Hint: You will need to use the function 'nls()' instead of 'lm()' as we are no longer fitting a linear model.

This is the end of this tutorial. Consider the following exercises if you have finished, and also for doing during this week.

From 3-1: Using the poisons data (available on the SMSTC website), find a transformation for which the model assumptions, when checked by the standard residuals plots, are reasonable.

From 3-4: A set of data on brown onions is available (on the SMSTC website). Fit a similar type of model as that of the white onions presented within the SMSTC notes. Comment on your findings.

From 1-1: Return to the data-frame cats in the MASS package from Tutorial 1. Plot and model the relationship between bwt and hwt using a non-linear function.