Week 3 Exercises

Andrew Perfors

## Day 1

### Tables

1. Create a summary table of the foods in Gladly’s data set and put it in a a variable called foodtable. Once you got it to work, knit this document to Word.

# Jeg er litt trøtt  
  
colours <- c("grey", "brown","navy","white","white","pink","pink","brown","yellow")  
# now make a table out of it  
  
colourtable <- table (colours)  
  
# print the table  
colourtable

## colours  
## brown grey navy pink white yellow   
## 2 1 1 2 2 1

# ny table  
# ny column <- verdier  
  
genders <- c("female","male","female","male","female","nb","female","female","male")  
  
# kjønn table  
gendertable <- table(genders)  
  
#yes  
gendertable

## genders  
## female male nb   
## 5 3 1

#crosstabulation  
table(colours,genders)

## genders  
## colours female male nb  
## brown 1 1 0  
## grey 1 0 0  
## navy 1 0 0  
## pink 1 0 1  
## white 1 1 0  
## yellow 0 1 0

1. Make a cross tab of age by ageCat. What do you think ageCat reflects? Do you think it is correctly calculated? [The latter two questions are thought questions asking you to think about and look at the data, no code necessary!]
2. Create a summary table of foods by species in the dataset. It will be kind of big.

### Pipes

1. The following code generates 100 random numbers from a normal distribution and calculates the mean of those numbers. Rewrite it using pipes and assign the output to a variable called meanNumsPipe.

# old code  
nums <- rnorm(n=100)  
meanNums <- mean(nums)  
  
# new code put here  
N <- 100

1. The following code generates 50 random numbers from a normal distribution with mean 100 and standard deviation 15. It then calculates the trimmed mean of those numbers. Rewrite it using pipes. Assign it to a new variable at the end. (A trimmed mean is the mean calculated on the dataset with a certain proportion of the ends removed. In this example, 20% is trimmed off each end. This is often more robust to extreme outliers than just taking the mean is.)

# old code  
nums <- rnorm(n=50,mean=100,sd=15)  
trMeanNums <- mean(nums,trim=0.2)  
trMeanNums

## [1] 100.1056

# new code  
N <- 50

### Grouping

1. Calculate the mean age of each species in Gladly’s dataset and assign the answer to a variable called ageTibble. Make sure ageTibble is visible when you knit your document.
2. Group Gladly’s data by species, and for each species calculate two variables: mnCarrot (which gives the mean answer to the carrot question) and sumCarrot (which gives the sum of everyone of that species answers to the carrot question). Note: you’ll have to use na.rm because there are some missing datapoints. Which of these do you think is more informative about how each species answers the question, and why?
3. What is the difference, if any, between group\_by(species,gender) and group\_by(gender,species)?
4. Group Gladly’s data by gender, and for each gender calculate lowQn (the 25th quantile), hiQn (the 75th quantile), and median (the 50th quantile) for age.
5. Challenge: in the last slide we calculated stdErr using the summary statistics calculated in a previous group\_by() command. It’s possible to do this all in one set of pipes (i.e., without having to save the intermediate tibble). See if you can figure out how to do this. NOTE: if you can’t do this, don’t worry! I just want you to practice playing around for yourself, but it’s not necessarily intuitively obvious.