

MATH2349 Semester 2, 2018

[Code ▾](#)

Assignment 2

Meg Cuddihy (s3608125), Sam Holt (s3381728), Verity Miles (s3644459)

Setup

Install and load the necessary packages to reproduce the report here:

[Hide](#)

```
# Loading all the required packages
library(readr)
library(tidyr)
library(dplyr)
library(Hmisc)
library(outliers)
library(DescTools)
library(kableExtra)
library(mosaic)
```

WHO data

Importing the WHO data

[Hide](#)

```
who <- read_csv("Who.csv")
kable(head(who[,1:5]), caption = "WHO data preview") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

WHO data preview

country	iso2	iso3	year	new_sp_m014
Afghanistan	AF	AFG	1980	NA
Afghanistan	AF	AFG	1981	NA
Afghanistan	AF	AFG	1982	NA
Afghanistan	AF	AFG	1983	NA
Afghanistan	AF	AFG	1984	NA
Afghanistan	AF	AFG	1985	NA

Note: all code the same

Tidy Task 1:

Use tidyr functions to reshape/ gather the WHO data

[Hide](#)

```
who_tidy1 <- who %>% gather(code, value, 5:length(who))
kable(head(who_tidy1)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

country	iso2	iso3	year	code	value
Afghanistan	AF	AFG	1980	new_sp_m014	NA
Afghanistan	AF	AFG	1981	new_sp_m014	NA
Afghanistan	AF	AFG	1982	new_sp_m014	NA
Afghanistan	AF	AFG	1983	new_sp_m014	NA
Afghanistan	AF	AFG	1984	new_sp_m014	NA
Afghanistan	AF	AFG	1985	new_sp_m014	NA

Note: Sam's code

Tidy Task 2:

Splitting/ separating the code column to get closer to tidy data

Hide

```
who_tidy2 <- who_tidy1 %>%
  separate(code, c("new", "var", "sex"), sep = "_") %>%
  separate(sex, c("sex", "age"), sep = 1)
kable(head(who_tidy2)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

country	iso2	iso3	year	new	var	sex	age	value
Afghanistan	AF	AFG	1980	new	sp	m	014	NA
Afghanistan	AF	AFG	1981	new	sp	m	014	NA
Afghanistan	AF	AFG	1982	new	sp	m	014	NA
Afghanistan	AF	AFG	1983	new	sp	m	014	NA
Afghanistan	AF	AFG	1984	new	sp	m	014	NA
Afghanistan	AF	AFG	1985	new	sp	m	014	NA

Note: Verity's code

Tidy Task 3:

Spreading the data to get it into a tidy format

Hide

```
who_tidy3 <- who_tidy2 %>% spread(var, value)
kable(head(who_tidy3)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

country	iso2	iso3	year	new	sex	age	ep	rel	sn	sp
Afghanistan	AF	AFG	1980	new	m	014	NA	NA	NA	NA
Afghanistan	AF	AFG	1981	new	m	014	NA	NA	NA	NA
Afghanistan	AF	AFG	1982	new	m	014	NA	NA	NA	NA
Afghanistan	AF	AFG	1983	new	m	014	NA	NA	NA	NA
Afghanistan	AF	AFG	1984	new	m	014	NA	NA	NA	NA
Afghanistan	AF	AFG	1985	new	m	014	NA	NA	NA	NA

Note: all code the same

Tidy Task 4:

Mutating and factorising variables

Hide

```
who_tidy4 <- who_tidy3 %>% mutate(sex = factor(sex, levels = c("m", "f"), labels = c("m", "f")),
                                age = factor(age, levels = c("014", "1524", "2534", "3544", "4554",
"5564", "65"),
                                labels = c("<15", "15-24", "25-34", "35-44", "45-54",
"55-64", "65>="),
                                ordered = TRUE))

#Checking the class of sex and age
sapply(who_tidy4[c("sex", "age")], class)
```

```
$sex
[1] "factor"

$age
[1] "ordered" "factor"
```

Hide

```
kable(head(who_tidy4)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

country	iso2	iso3	year	new	sex	age	ep	rel	sn	sp
Afghanistan	AF	AFG	1980	new	m	<15	NA	NA	NA	NA
Afghanistan	AF	AFG	1981	new	m	<15	NA	NA	NA	NA
Afghanistan	AF	AFG	1982	new	m	<15	NA	NA	NA	NA
Afghanistan	AF	AFG	1983	new	m	<15	NA	NA	NA	NA
Afghanistan	AF	AFG	1984	new	m	<15	NA	NA	NA	NA
Afghanistan	AF	AFG	1985	new	m	<15	NA	NA	NA	NA

Note: Meg's code

Task 5: Filter & Select

Filtering and selecting the WHO data

Hide

```
countries <- c("Fiji", "Germany", "Korea")
who_subset <- who_tidy4 %>% select(-c(2, 5)) %>%
  filter(country %in% countries)
kable(head(who_subset)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

country	iso3	year	sex	age	ep	rel	sn	sp
Fiji	FJI	1980	m	<15	NA	NA	NA	NA
Fiji	FJI	1981	m	<15	NA	NA	NA	NA
Fiji	FJI	1982	m	<15	NA	NA	NA	NA
Fiji	FJI	1983	m	<15	NA	NA	NA	NA
Fiji	FJI	1984	m	<15	NA	NA	NA	NA
Fiji	FJI	1985	m	<15	NA	NA	NA	NA

Note: Verity's code

Species and Surveys data sets

Importing the species and surveys data

Hide

```
surveys <- read_csv("surveys.csv")
species <- read_csv("species.csv")
kable(head(surveys), caption = "Surveys data preview") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

Surveys data preview

record_id	month	day	year	species_id	sex	hindfoot_length	weight
1	7	16	1977	NL	M	32	NA
2	7	16	1977	NL	M	33	NA
3	7	16	1977	DM	F	37	NA
4	7	16	1977	DM	M	36	NA
5	7	16	1977	DM	M	35	NA
6	7	16	1977	PF	M	14	NA

Hide

```
kable(head(species), caption = "Species data preview") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

Species data preview

species_id	genus	species	taxa
AB	Amphispiza	bilineata	Bird
AH	Ammospermophilus	harrisi	Rodent
AS	Ammodramus	savannarum	Bird
BA	Baiomys	taylori	Rodent
CB	Campylorhynchus	brunneicapillus	Bird
CM	Calamospiza	melanocorys	Bird

Note: all code the same

Task 6: Join

Join the species and surveys data

Hide

```
surveys_combined <- surveys %>% left_join(species[, c("species_id", "genus", "species", "taxa")], by = "species_id")
kable(head(surveys_combined)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

record_id	month	day	year	species_id	sex	hindfoot_length	weight	genus	species	taxa
1	7	16	1977	NL	M	32	NA	Neotoma	albigula	Rodent
2	7	16	1977	NL	M	33	NA	Neotoma	albigula	Rodent
3	7	16	1977	DM	F	37	NA	Dipodomys	merriami	Rodent
4	7	16	1977	DM	M	36	NA	Dipodomys	merriami	Rodent
5	7	16	1977	DM	M	35	NA	Dipodomys	merriami	Rodent
6	7	16	1977	PF	M	14	NA	Perognathus	flavus	Rodent

Note: Verity's code

Task 7: Calculate

Calculating mean weight and hindfoot lengths

Hide

```
species_avg <- surveys_combined %>% filter(species == "fulviventer") %>%
  group_by(month) %>%
  summarise(mean_weights = mean(weight, na.rm = TRUE),
            mean_foot = mean(hindfoot_length, na.rm = TRUE))
kable(head(species_avg)) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

month	mean_weights	mean_foot
-------	--------------	-----------

month	mean_weights	mean_foot
1	48.22222	26.77778
2	56.50000	25.75000
3	58.00000	28.66667
4	50.50000	21.00000
5	54.66667	26.00000
6	38.50000	23.00000

Note: mainly Meg's code

Task 8: Missing Values

Dealing with missing values through imputing

Hide

```
surveys_combined_year <- surveys_combined %>% filter(year == "1990")
surveys_combined_year %>% group_by(species) %>% summarise(Missing = sum(is.na(weight)))
```

species <chr>	Missing <int>
albigula	0
bilineata	27
chlorurus	8
eremicus	2
flavus	0
fulvescens	0
fulviventer	0
gramineus	4
harrisi	22
hispidus	0
1-10 of 21 rows	Previous 1 2 3 Next

Hide

```
surveys_weight_imputed <- surveys_combined_year %>% group_by(species) %>% mutate(weight = ifelse(is.na(weight), mean(weight, na.rm = TRUE), weight))
```

Note: Sam's code

There are still missing values as there are some entire species that only have NA values for weight in 1990. The means of these species is NaN.

Task 9: Inconsistencies or Special Values

Checking for inconsistencies in the weight data and explaining why they have occurred.

[Hide](#)

```
sum(is.nan(surveys_weight_imputed$weight))
```

```
[1] 82
```

[Hide](#)

```
is.special <- function(x){
  if (is.numeric(x)) !is.finite(x) else is.na(x)
}
any(sapply(surveys_weight_imputed$weight, is.special))
```

```
[1] TRUE
```

[Hide](#)

```
#Checking one example
example <- first(which(is.na(surveys_weight_imputed$weight)))
species_eg <- surveys_weight_imputed[example,]$species
no_data_species <- filter(surveys_weight_imputed, species == species_eg)
favstats(no_data_species$weight) #Shows that all values for this species are NaN
```

	min <dbl>	Q1 <dbl>	median <dbl>	Q3 <dbl>	max <dbl>	mean <dbl>	sd <dbl>	n <int>	missing <int>
	NA	NA	NA	NA	NA	NaN	NA	0	27

```
1 row
```

[Hide](#)

```
surveys_weight_imputed %>% group_by(species) %>% summarise(Missing = sum(is.na(weight)), Mean = mean(w
eight))
```

species <chr>	Missing <int>	Mean <dbl>
albigula	0	154.275862
bilineata	27	NaN
chlorurus	8	NaN
eremicus	0	21.671233
flavus	0	7.117647
fulvescens	0	12.916667
fulviventor	0	52.611111
gramineus	4	NaN
harrisi	22	NaN
hispidus	0	56.875000

```
1-10 of 21 rows
```

```
Previous 1 2 3 Next
```

Note: combination of code

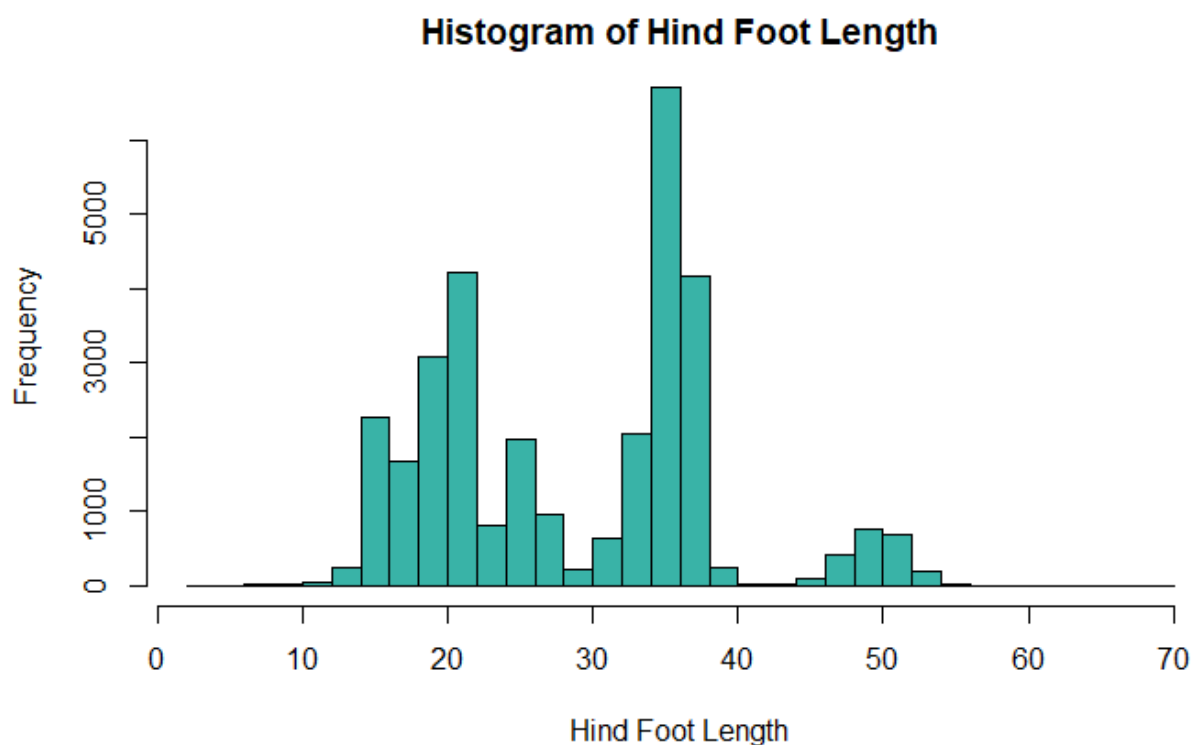
Explanation: Some species in the data frame have no values available for weight for any observation of that species. When calculating the mean by species, we have excluded NAs so any species with no weight values recorded will have no valid numbers available for calculation. Therefore, the final output includes NaN (not a number) results for the weights of species with no valid weight observation.

Task 10: Outliers

Checking for outliers in the hindfoot length data

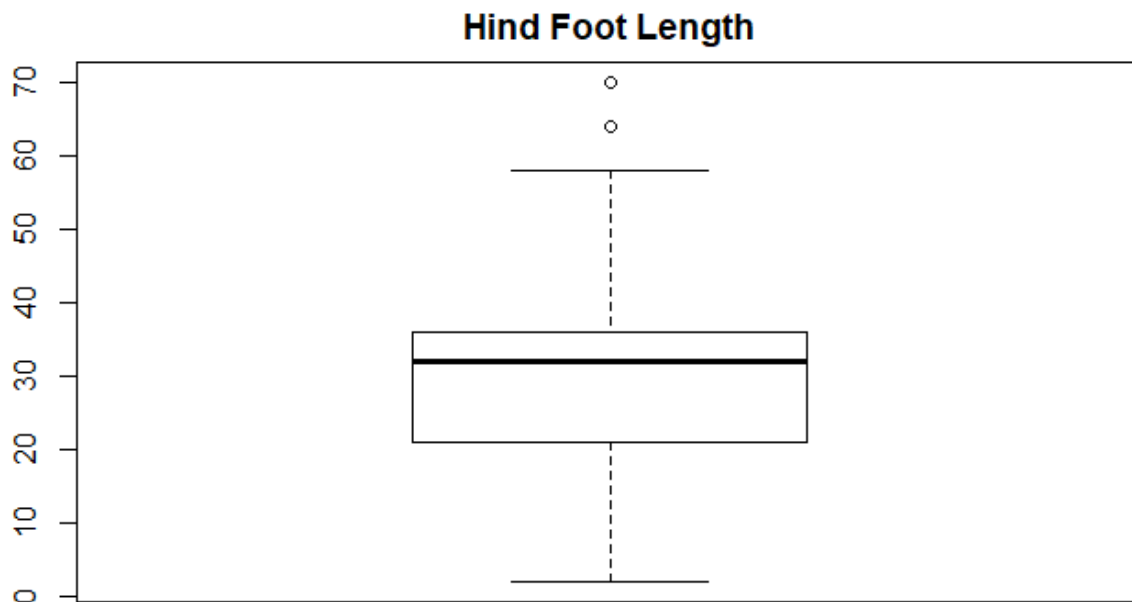
Hide

```
hist(surveys_combined$hindfoot_length, breaks = 30,  
     main = "Histogram of Hind Foot Length",  
     xlab = "Hind Foot Length",  
     col = "#39b3a7") #does not seem to be normally distributed
```



Hide

```
x <- boxplot(surveys_combined$hindfoot_length,  
             main = "Hind Foot Length") #hence the Tukey method is used
```

Hide

```
x$out
```

```
[1] 70 64
```

Hide

```
IQR(surveys_combined$hindfoot_length, na.rm = TRUE) * 3
```

```
[1] 45
```

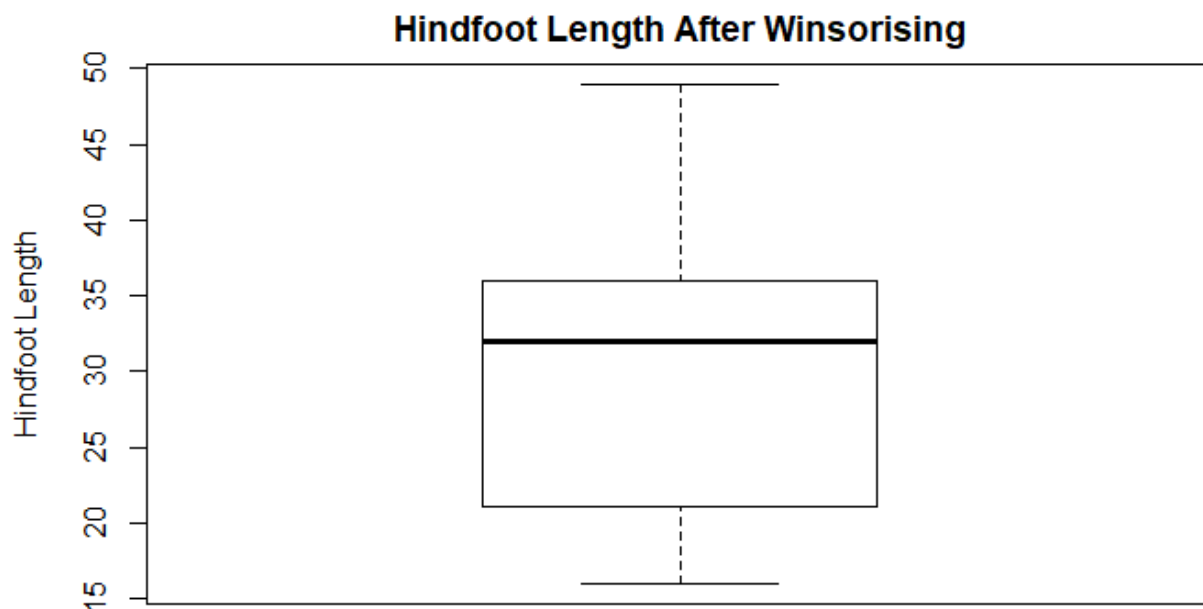
Hide

```
# Winsorising the outliers as they don't follow a normal distribution nor are they extreme (neither exceed 3*IQR)
# Winsorising with the Winsorize() function from the DescTools Package
surveys_combined_capped <- Winsorize(surveys_combined$hindfoot_length, na.rm = TRUE)
summary(surveys_combined_capped)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
16.00	21.00	32.00	29.24	36.00	49.00	4111

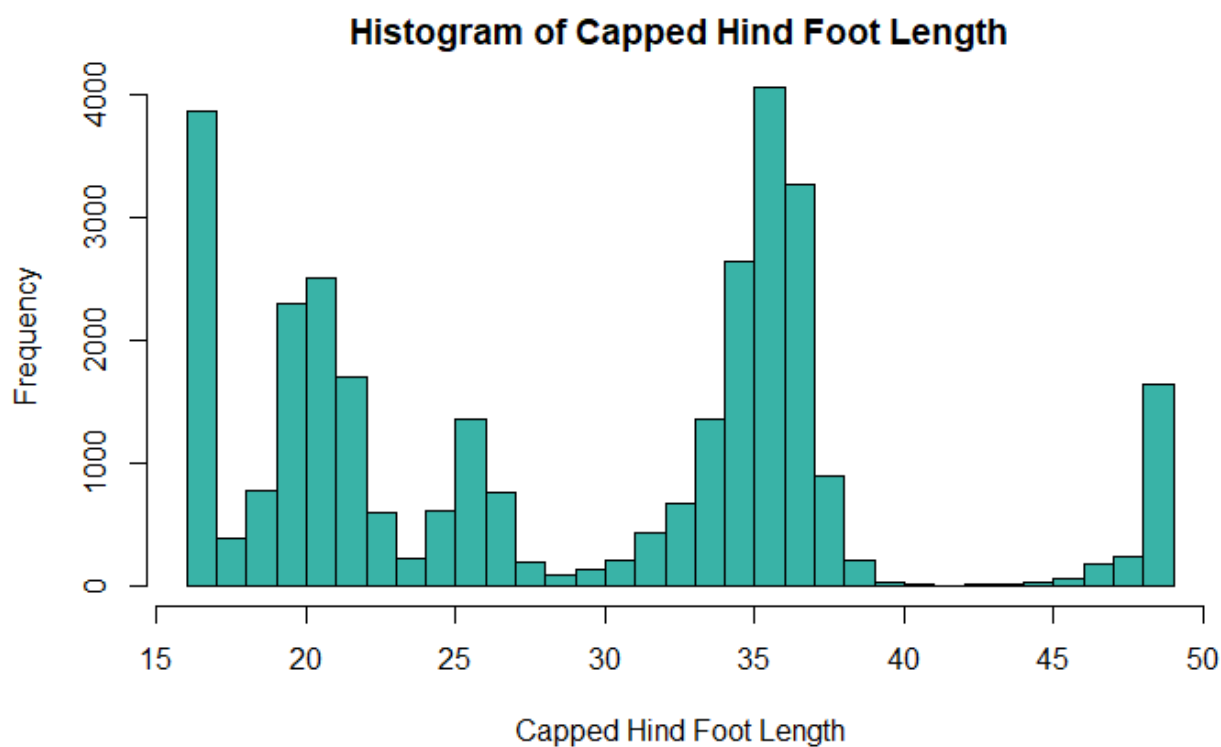
Hide

```
boxplot(surveys_combined_capped,
        main = 'Hindfoot Length After Winsorising',
        ylab = 'Hindfoot Length')
```



Hide

```
hist(surveys_combined_capped, breaks = 30,  
     main = "Histogram of Capped Hind Foot Length",  
     xlab = "Capped Hind Foot Length",  
     col = "#39b3a7") #show that there are now no outliers but more values at each edge of the data
```



Note: Sam's code

The hindfoot length data is not normally distributed as shown in the initial histogram. The boxplot indicates that there are two outliers present. The Tukey method was used to transform the data. Rather than ignoring the outliers or setting them to the mean value, we chose to cap/ winsorise them. We did this as they don't follow a normal distribution nor are they

very extreme. The second histogram shows that the outliers were dealt with as the spread of the data is not as wide and there are more values in the first and last bins than in the original histogram.