

Numerical Summaries

Summarizing data in R 1/2

- Have seen `summary` (5-number summary of each column). But what if we want:
 - ▶ a summary or two of just one column
 - ▶ a count of observations in each category of a categorical variable
 - ▶ summaries by group
 - ▶ a different summary of all columns (eg. SD)
- To do this, meet pipe operator `%>%`. This takes input data frame, does something to it, and outputs result. (Learn: Ctrl-Shift-M.)

Summarizing data in R 2/2

- Output from a pipe can be used as input to something else, so can have a sequence of pipes.
- Summaries include: `mean`, `median`, `min`, `max`, `sd`, `IQR`, `quantile` (for obtaining quartiles or any percentile), `n` (for counting observations).
- Use our Australian athletes data again.

Packages for this section

```
library(tidyverse)
```

The athletes

```
summary(athletes)
```

Sex	Sport	RCC	WCC	
Length:202	Length:202	Min. :3.800	Min. : 3.300	
Class :character	Class :character	1st Qu.:4.372	1st Qu.: 5.900	
Mode :character	Mode :character	Median :4.755	Median : 6.850	
		Mean :4.719	Mean : 7.109	
		3rd Qu.:5.030	3rd Qu.: 8.275	
		Max. :6.720	Max. :14.300	
Hc	Hg	Ferr	BMI	SSF
Min. :35.90	Min. :11.60	Min. : 8.00	Min. :16.75	Min. : 28.00
1st Qu.:40.60	1st Qu.:13.50	1st Qu.: 41.25	1st Qu.:21.08	1st Qu.: 43.85
Median :43.50	Median :14.70	Median : 65.50	Median :22.72	Median : 58.60
Mean :43.09	Mean :14.57	Mean : 76.88	Mean :22.96	Mean : 69.02
3rd Qu.:45.58	3rd Qu.:15.57	3rd Qu.: 97.00	3rd Qu.:24.46	3rd Qu.: 90.35
Max. :59.70	Max. :19.20	Max. :234.00	Max. :34.42	Max. :200.80
%Bfat	LBM	Ht	Wt	
Min. : 5.630	Min. : 34.36	Min. :148.9	Min. : 37.80	
1st Qu.: 8.545	1st Qu.: 54.67	1st Qu.:174.0	1st Qu.: 66.53	
Median :11.650	Median : 63.03	Median :179.7	Median : 74.40	
Mean :13.507	Mean : 64.87	Mean :180.1	Mean : 75.01	
3rd Qu.:18.080	3rd Qu.: 74.75	3rd Qu.:186.2	3rd Qu.: 84.12	
Max. :35.520	Max. :106.00	Max. :209.4	Max. :123.20	

Summarizing one column

- Mean height:

```
athletes %>% summarize(m=mean(Ht))
```

```
# A tibble: 1 x 1
      m
  <dbl>
1  180.
```

or to get mean and SD of BMI:

```
athletes %>% summarize(m = mean(BMI), s = sd(BMI))
```

```
# A tibble: 1 x 2
      m      s
  <dbl> <dbl>
1  23.0  2.86
```

A warning

This doesn't work:

```
mean(BMI)
```

Error: object 'BMI' not found

because R needs to know what *dataframe* BMI lives in.

Quartiles

- quantile calculates percentiles (“fractiles”), so we want the 25th and 75th percentiles:

```
athletes %>% summarize( Q1=quantile(Wt, 0.25),  
                        Q3=quantile(Wt, 0.75))
```

```
# A tibble: 1 x 2  
      Q1      Q3  
  <dbl> <dbl>  
1  66.5  84.1
```


Creating new columns

- These weights are in kilograms. Maybe we want to summarize the weights in pounds.
- Convert kg to lb by multiplying by 2.2.
- Create new column and summarize that:

```
athletes %>% mutate(wt_lb=Wt*2.2) %>%  
  summarize(Q1_lb=quantile(wt_lb, 0.25),  
            Q3_lb=quantile(wt_lb, 0.75))
```

```
# A tibble: 1 x 2  
  Q1_lb Q3_lb  
  <dbl> <dbl>  
1  146.  185.
```

Counting how many

for example, number of athletes in each sport:

```
athletes %>% count(Sport)
```

```
# A tibble: 10 x 2
```

	Sport	n
	<chr>	<int>
1	BBall	25
2	Field	19
3	Gym	4
4	Netball	23
5	Row	37
6	Swim	22
7	T400m	29
8	TSprnt	15
9	Tennis	11
10	WPolo	17

Counting how many, variation 2:

Another way (which will make sense in a moment):

```
athletes %>% group_by(Sport) %>%  
  summarize(count=n())
```

```
# A tibble: 10 x 2
```

	Sport	count
	<chr>	<int>
1	BBall	25
2	Field	19
3	Gym	4
4	Netball	23
5	Row	37
6	Swim	22
7	T400m	29
8	TSprnt	15
9	Tennis	11
10	WPolo	17

Summaries by group

- Might want separate summaries for each “group”, eg. mean and SD of height for males and females. Strategy is `group_by` (to define the groups) and then `summarize`:

```
athletes %>% group_by(Sex) %>%  
  summarize(mean_Ht = mean(Ht), sd_Ht = sd(Ht))
```

```
# A tibble: 2 x 3  
  Sex      mean_Ht sd_Ht  
  <chr>    <dbl> <dbl>  
1 female    175.  8.24  
2 male     186.  7.90
```

Count plus stats

- If you want number of observations per group plus some stats, you need to go the `n()` way:

```
athletes %>% group_by(Sex) %>%  
summarize(n = n(), mean_Ht = mean(Ht), sd_Ht = sd(Ht))
```

```
# A tibble: 2 x 4  
  Sex      n mean_Ht sd_Ht  
  <chr> <int>   <dbl> <dbl>  
1 female   100    175.   8.24  
2 male    102    186.   7.90
```

- This explains second variation on counting within group: “within each sport/Sex, how many athletes were there?”

Summarizing several columns 1/2

- Standard deviation of each (numeric) column:

```
athletes %>% summarize(across(where(is.numeric), \(x) sd(x)))
```

```
# A tibble: 1 x 11
```

	RCC	WCC	Hc	Hg	Ferr	BMI	SSF	`\${Bfat`	LBM	Ht	Wt
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	0.458	1.80	3.66	1.36	47.5	2.86	32.6	6.19	13.1	9.73	13.9

Summarizing several columns 2/2

- Median and IQR of all columns whose name starts with H:

```
athletes %>% summarize(across(starts_with("H"),  
                             list(med = \(x) median(x),  
                                  iqr = \(x) IQR(x))))
```

```
# A tibble: 1 x 6
```

	Hc_med	Hc_iqr	Hg_med	Hg_iqr	Ht_med	Ht_iqr
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	43.5	4.98	14.7	2.07	180.	12.2

Same thing by group

```
athletes %>%  
  group_by(Sex) %>%  
  summarize(across(starts_with("H"),  
                    list(med = \(\h) median(h),  
                        iqr = \(\h) IQR(h))))
```

```
# A tibble: 2 x 7
```

	Sex	Hc_med	Hc_iqr	Hg_med	Hg_iqr	Ht_med	Ht_iqr
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	female	40.6	4.03	13.5	1.60	175	8.68
2	male	45.5	2.57	15.5	0.975	186.	11.3

... another one

```
athletes %>%  
  group_by(Sex) %>%  
  summarize(across(ends_with("C"),  
                    list(med = \(\h) median(h),  
                         iqr = \(\h) IQR(h))))
```

A tibble: 2 x 7

	Sex	RCC_med	RCC_iqr	WCC_med	WCC_iqr	Hc_med	Hc_iqr
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	female	4.38	0.370	6.7	2.15	40.6	4.03
2	male	5.01	0.315	7.1	2.35	45.5	2.57