

Tables

Introduction

First, load {tidyverse}, {papaja}, and {palmerpenguins}.

```
library(tidyverse)
library(papaja)
library(palmerpenguins)
```

Now let's build a data frame that will be our table.

```
(penguins_means <- penguins |>
  summarise(.by = c(island, species),
    bill_length = mean(bill_length_mm, na.rm = TRUE),
    bill_depth = mean(bill_depth_mm, na.rm = TRUE),
    flipper_length = mean(flipper_length_mm, na.rm = TRUE)) |>
  arrange(island))
```

```
## # A tibble: 5 x 5
##   island    species  bill_length bill_depth flipper_length
##   <fct>    <fct>      <dbl>      <dbl>      <dbl>
## 1 Biscoe  Adelie        39.0        18.4        189.
## 2 Biscoe  Gentoo        47.5        15.0        217.
## 3 Dream   Adelie        38.5        18.3        190.
## 4 Dream   Chinstrap     48.8        18.4        196.
## 5 Torgersen Adelie        39.0        18.4        191.
```

Tables by {knitr}

The `{knitr}` package uses the `kable()` function to format tables.

```
library(knitr)
kable(penguins_means)
```

island	species	bill_length	bill_depth	flipper_length
Biscoe	Adelie	38.97500	18.37045	188.7955
Biscoe	Gentoo	47.50488	14.98211	217.1870
Dream	Adelie	38.50179	18.25179	189.7321
Dream	Chinstrap	48.83382	18.42059	195.8235
Torgersen	Adelie	38.95098	18.42941	191.1961

Column and row names

You can control column names and row names with `col.names` and `row.names`.

```
kable(penguins_means,  
      col.names = c("Island", "Species", "Bill length (mm)",  
                    "Bill depth (mm)", "Flipper length (mm)"))
```

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.97500	18.37045	188.7955
Biscoe	Gentoo	47.50488	14.98211	217.1870
Dream	Adelie	38.50179	18.25179	189.7321
Dream	Chinstrap	48.83382	18.42059	195.8235
Torgersen	Adelie	38.95098	18.42941	191.1961

Column alignment

By default, character columns are left aligned and numeric columns are right aligned. You can set alignment manually with the `align` argument with `l` = left, `c` = center, and `r` = right. You can just pass a character string with a series of those letters.

```
kable(penguins_means,  
      col.names = c("Island", "Species", "Bill length (mm)",  
                    "Bill depth (mm)", "Flipper length (mm)"),  
      align = "rclcr")
```

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.97500	18.37045	188.7955
Biscoe	Gentoo	47.50488	14.98211	217.1870
Dream	Adelie	38.50179	18.25179	189.7321
Dream	Chinstrap	48.83382	18.42059	195.8235
Torgersen	Adelie	38.95098	18.42941	191.1961

Digit rounding

Round the digits for all numeric data columns with `digits` argument.

```
penguins_means |>
  kable(digits = 2)
```

island	species	bill_length	bill_depth	flipper_length
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

If you want different digits for different columns, you can pass a vector to the `digits` argument.

```
penguins_means |>
  kable(digits = c(0, 0, 1, 2, 3))
```

island	species	bill_length	bill_depth	flipper_length
Biscoe	Adelie	39.0	18.37	188.795
Biscoe	Gentoo	47.5	14.98	217.187
Dream	Adelie	38.5	18.25	189.732
Dream	Chinstrap	48.8	18.42	195.824
Torgersen	Adelie	39.0	18.43	191.196

Table 1

Penguin body measurements by island and species

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.97500	18.37045	188.7955
Biscoe	Gentoo	47.50488	14.98211	217.1870
Dream	Adelie	38.50179	18.25179	189.7321
Dream	Chinstrap	48.83382	18.42059	195.8235
Torgersen	Adelie	38.95098	18.42941	191.1961

Table titles

Add a title to the table with the `caption` argument. The good news is that we can cross-reference easily (Table 1). The bad news is that with captions, tables in PDFs are automatically placed at the top of the page. We'll see how to fix this later.

```
kable(penguins_means,
      col.names = c("Island", "Species", "Bill length (mm)",
                    "Bill depth (mm)", "Flipper length (mm)"),
      caption = "Penguin body measurements by island and species")
```

Supplementing kable with {kableExtra}

The `kable()` function is intentionally simple to use and therefore does not have a lot of additional functionality. The `{kableExtra}` package supplements the `kable()` functionality with additional formatting options by adding additional functions after the `kable()` function call using the `|>` pipe (a bit like how `ggplot()` works). Check out [Create Awesome LaTeX Table with knitr::kable and kableExtra](#).

```
# install.packages("kableExtra")
library(kableExtra)
```

General styling

The `kable_styling()` function formats a number of things such as font size, table width, and table alignment. I'll also add `latex_options = "hold_position"` to keep the table in the text. Otherwise, it puts it at the top of the page.

```
kable(penguins_means,
      caption = "Penguin body measurements by island and species",
      col.names = c("Island", "Species", "Bill length (mm)",
                    "Bill depth (mm)", "Flipper length (mm)"),
      digits = 2,
      booktabs = TRUE) |>
  kable_styling(font_size = 15,
                latex_options = "hold_position")
```

Table 2

Penguin body measurements by island and species

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

Labels spanning rows

If you want to label groups of rows, use `pack_rows()`. Let's get rid of the `island` column and label the islands explicitly.

```
penguins_means2 <- penguins_means |>
  select(-island)
```

```
kable(penguins_means2,
      digits = 2,
      booktabs = TRUE)
```

species	bill_length	bill_depth	flipper_length
Adelie	38.98	18.37	188.80
Gentoo	47.50	14.98	217.19
Adelie	38.50	18.25	189.73
Chinstrap	48.83	18.42	195.82
Adelie	38.95	18.43	191.20

```
kable(penguins_means2,
      digits = 2,
      booktabs = TRUE,
      col.names = c("", "Bill length (mm)", "Bill depth (mm)",
                    "Flipper length (mm)"))
) |>
pack_rows("Biscoe", start_row = 1, end_row = 2) |>
pack_rows("Dream", start_row = 3, end_row = 4) |>
pack_rows("Torgersen", start_row = 5, end_row = 5)
```

	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe			
Adelie	38.98	18.37	188.80
Gentoo	47.50	14.98	217.19
Dream			
Adelie	38.50	18.25	189.73
Chinstrap	48.83	18.42	195.82
Torgersen			
Adelie	38.95	18.43	191.20

Notice that we removed the first column name with "".

Labels spanning columns

You can label groups of columns with the `add_header_above()` function. Let's rearrange the data into wide format to illustrate this.

```
(wide_means <- penguins_means |>
  unite(island_species, island:species) |>
  pivot_wider(id_cols = !bill_depth:flipper_length,
              names_from = island_species,
              values_from = bill_length))
```

```
## # A tibble: 1 x 5
##   Biscoe_Adelie Biscoe_Gentoo Dream_Adelie Dream_Chinstrap Torgersen_Adelie
##           <dbl>         <dbl>         <dbl>         <dbl>         <dbl>
## 1           39.0           47.5           38.5           48.8           39.0
```

```
kable(wide_means,
      digits = 2,
      booktabs = TRUE)
```

Biscoe_Adelie	Biscoe_Gentoo	Dream_Adelie	Dream_Chinstrap	Torgersen_Adelie
38.98	47.5	38.5	48.83	38.95

Now that the data are in wide format, we can add the column names by specifying the species then add the headers.

```
kable(wide_means,
      digits = 2,
      booktabs = TRUE,
      col.names = c("Adelie", "Gentoo", "Adelie",
                    "Chinstrap", "Adelie")) |>
  add_header_above(c("Biscoe" = 2, "Dream" = 2, "Torgersen" = 1))
```

Biscoe		Dream		Torgersen
Adelie	Gentoo	Adelie	Chinstrap	Adelie
38.98	47.5	38.5	48.83	38.95

Table footnotes

Add table notes with the `footnote()` function.

```
kable(penguins_means,
      digits = 2,
      booktabs = TRUE,
      caption = "Penguin body measurements by island and species",
      col.names = c("Island*", "Species", "Bill length (mm)",
                    "Bill depth (mm)", "Flipper length (mm)")) |>
kable_styling(latex_options = "hold_position") |>
footnote(general = "Source: Gorman et al. (2014)",
        symbol = "Not all species are found on all islands.",
        footnote_as_chunk = TRUE)
```

Table 3

Penguin body measurements by island and species

Island*	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

Note: Source: Gorman et al. (2014)

* Not all species are found on all islands.

Landscape

Rotate wide tables with `landscape()` function.

```
kable(penguins_means,
      digits = 2,
      booktabs = TRUE,
      caption = "Penguin body measurements by island and species",
      col.names = c("Island*", "Species", "Bill length (mm)",
                    "Bill depth (mm)", "Flipper length (mm)")) |>
kable_styling(latex_options = "hold_position") |>
footnote(general = "Source: Gorman et al. (2014)",
        symbol = "Not all species are found on all islands.",
        footnote_as_chunk = TRUE) |>
landscape()
```

Table 4
Penguin body measurements by island and species

Island*	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

Note: Source: Gorman et al. (2014)

* Not all species are found on all islands.

Tables by {papaja}

The {papaja} package uses the `kable()` function to format tables in APA format with the `apa_table()` function. You can use many of the same arguments that are available in the `kable()` function. You can control where the table is placed (here, top, bottom) with the `placement` argument. You can add a general footnote with the `note` argument.

```
apa_table(penguins_means,
  caption = "Penguin body measurements by island and species",
  col.names = c("Island", "Species", "Bill length (mm)",
                "Bill depth (mm)", "Flipper length (mm)"),
  placement = "h",
  note = "Source: Gorman et al. (2014)")
```

Table 5

Penguin body measurements by island and species

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

Note. Source: Gorman et al. (2014)

Notice the alignment is different, with everything left aligned. Let's right align the means.

```
apa_table(penguins_means,
  caption = "Penguin body measurements by island and species",
  col.names = c("Island", "Species", "Bill length (mm)",
                "Bill depth (mm)", "Flipper length (mm)"),
  align = c("l", "l", "r", "r", "r"),
  placement = "h",
  note = "Source: Gorman et al. (2014)")
```

Table 6
Penguin body measurements by island and species

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

Note. Source: Gorman et al. (2014)

You can rotate to landscape orientation with the `landscape = TRUE` argument.

```
apa_table(penguins_means,
  caption = "Penguin body measurements by island and species",
  col.names = c("Island", "Species", "Bill length (mm)",
    "Bill depth (mm)", "Flipper length (mm)"),
  align = c("l", "l", "r", "r", "r"),
  placement = "h",
  note = "Source: Gorman et al. (2014)",
  landscape = TRUE)
```

Table 7
Penguin body measurements by island and species

Island	Species	Bill length (mm)	Bill depth (mm)	Flipper length (mm)
Biscoe	Adelie	38.98	18.37	188.80
Biscoe	Gentoo	47.50	14.98	217.19
Dream	Adelie	38.50	18.25	189.73
Dream	Chinstrap	48.83	18.42	195.82
Torgersen	Adelie	38.95	18.43	191.20

Note. Source: Gorman et al. (2014)

APA-formatted statistics by {papaja}

{papaja} also includes `apa_print()`, which extracts statistical values in APA format.

Linear regression

```
penguin_lm <- lm(bill_length_mm ~ sex, data = penguins)
summary(penguin_lm)
```

```
##
## Call:
## lm(formula = bill_length_mm ~ sex, data = penguins)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.2548  -4.7548   0.8452   4.3030  15.9030
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  42.0970     0.4003 105.152 < 2e-16 ***
## sexmale       3.7578     0.5636   6.667 1.09e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.143 on 331 degrees of freedom
## (11 observations deleted due to missingness)
## Multiple R-squared:  0.1184, Adjusted R-squared:  0.1157
## F-statistic: 44.45 on 1 and 331 DF, p-value: 1.094e-10
```

```
apa_print(penguin_lm)
```

```
## $estimate
## $estimate$Intercept
## [1] "$b = 42.10$, 95\\% CI $[41.31, 42.88]$"
##
## $estimate$sexmale
## [1] "$b = 3.76$, 95\\% CI $[2.65, 4.87]$"
##
## $estimate$modelfit
## $estimate$modelfit$r2
## [1] "$R^2 = .12$, 90\\% CI $[0.07, 0.18]$"
##
```

```
## $estimate$model$fit$r2_adj
## [1] "$R^2_{adj} = .12$"
##
## $estimate$model$fit$aic
## [1] "$\\mathrm{AIC} = 2,039.61$"
##
## $estimate$model$fit$bic
## [1] "$\\mathrm{BIC} = 2,051.03$"
##
##
##
## $statistic
## $statistic$Intercept
## [1] "$t(331) = 105.15$, $p < .001$"
##
## $statistic$sexmale
## [1] "$t(331) = 6.67$, $p < .001$"
##
## $statistic$model$fit
## $statistic$model$fit$r2
## [1] "$F(1, 331) = 44.45$, $p < .001$"
##
##
##
## $full_result
## $full_result$Intercept
## [1] "$b = 42.10$, 95\\% CI $[41.31, 42.88]$, $t(331) = 105.15$, $p < .001$"
##
## $full_result$sexmale
## [1] "$b = 3.76$, 95\\% CI $[2.65, 4.87]$, $t(331) = 6.67$, $p < .001$"
##
## $full_result$model$fit
## $full_result$model$fit$r2
## [1] "$R^2 = .12$, 90\\% CI $[0.07, 0.18]$, $F(1, 331) = 44.45$, $p < .001$"
##
##
##
## $table
## A data.frame with 6 labelled columns:
##
##      term estimate      conf.int statistic  df p.value
## 1 Intercept    42.10 [41.31, 42.88]   105.15 331  < .001
## 2 Sexmale      3.76  [2.65, 4.87]     6.67 331  < .001
##
## term      : Predictor
```

```
## estimate : $b$
## conf.int : 95\\% CI
## statistic: $t$
## df       : $\\mathit{df}$
## p.value  : $p$
## attr(,"class")
## [1] "apa_results" "list"
```

```
apa_table(apa_print(penguin_lm)$table,
  caption = "Linear regression results",
  placement = "h")
```

Table 8
Linear regression results

Predictor	<i>b</i>	95% CI	<i>t</i>	<i>df</i>	<i>p</i>
Intercept	42.10	[41.31, 42.88]	105.15	331	< .001
Sexmale	3.76	[2.65, 4.87]	6.67	331	< .001

Let's clean up those predictor names.

```
penguin_lm_table <- apa_print(penguin_lm)$table |>
  mutate(term = str_replace(term, "Sexmale", "Sex"))
apa_table(penguin_lm_table,
  caption = "Linear regression results",
  placement = "h")
```

Table 9

Linear regression results

term	<i>b</i>	95% CI	<i>t</i>	<i>df</i>	<i>p</i>
Intercept	42.10	[41.31, 42.88]	105.15	331	< .001
Sex	3.76	[2.65, 4.87]	6.67	331	< .001

How could we name the first column *Predictor* instead of *term*?

Other table packages

- `{gt}` RStudio's grammar of tables (logically like `ggplot2`)
- `{flextable}` Good Word output but a bit tricky to work with
- `{huxtable}` Very flexible but tricky to work with