wk1 coding exercises

Christine Hawkes

1/12/2022

Contents

##Install R packages from CRAN

```
library("tidyverse")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

1. create a new data file with dimensions of 10x10

- (9x9 of data, 1 row for column headings and 1 column for sample names)

```
Cex <- read.csv("CodingExampleFile.csv", header=TRUE, row.names = 1)
```

2. rename a column (imagine you find a mistake in your file)

- for help see https://dplyr.tidyverse.org/reference/rename.html#examples

```
colnames(Cex)
```

```
## [1] "var1" "var2" "var3" "var4" "var5" "var6" "var7" "var8" "var9"
```

```
Cex<-dplyr::rename(Cex, var10=var9)
colnames(Cex)
```

```
## [1] "var1"  "var2"  "var3"  "var4"  "var5"  "var6"  "var7"  "var8"  "var10"
```

3. add a column with at least three categories (e.g., group=A,B,C) using baseR, tibble::add_column, or dplyr::mutate

- https://tibble.tidyverse.org/reference/add_column.html
- https://dplyr.tidyverse.org/reference/mutate.html

```
#base R
Cex2 <- Cex
Cex2$group <- c("A", "A", "A", "B", "B", "B", "C", "C", "C")
Cex2
```

```
##       var1 var2 var3 var4 var5 var6 var7 var8 var10 group
## samp1    0    8    9    1    3    5    8    1     2     A
## samp2   10    7    0    1    9   10    3   10     0     A
## samp3    2    3    3    5   10    2    1    1     5     A
## samp4    8    5    1    6    6    8    6   10     6     B
## samp5    6    7    7    1    9    6    9    0     7     B
## samp6    9    0    3    5    9    1    7    9     0     B
## samp7    6    4    4    8    0    0    6    7     8     C
## samp8    0    4    2    2    8    1    2    1     9     C
## samp9    0   10    2    4    0    9    2    2     6     C
```

```
#add_column
Cex3 <- as_tibble(Cex) #note that tibble loses rownames
Cex3 <- tibble::add_column(Cex3, group=c("A", "A", "A", "B", "B", "B", "C", "C", "C"))
Cex3
```

```
## # A tibble: 9 x 10
##    var1  var2  var3  var4  var5  var6  var7  var8 var10 group
##   <int> <int> <int> <int> <int> <int> <int> <int> <int> <chr>
## 1     0     8     9     1     3     5     8     1     2 A
## 2    10     7     0     1     9    10     3    10     0 A
## 3     2     3     3     5    10     2     1     1     5 A
## 4     8     5     1     6     6     8     6    10     6 B
## 5     6     7     7     1     9     6     9     0     7 B
## 6     9     0     3     5     9     1     7     9     0 B
## 7     6     4     4     8     0     0     6     7     8 C
## 8     0     4     2     2     8     1     2     1     9 C
## 9     0    10     2     4     0     9     2     2     6 C
```

```
#mutate
Cex4 <- dplyr::mutate(Cex, group=c("A", "A", "A", "B", "B", "B", "C", "C", "C"))
Cex4
```

```
##       var1 var2 var3 var4 var5 var6 var7 var8 var10 group
## samp1    0    8    9    1    3    5    8    1     2     A
## samp2   10    7    0    1    9   10    3   10     0     A
## samp3    2    3    3    5   10    2    1    1     5     A
## samp4    8    5    1    6    6    8    6   10     6     B
## samp5    6    7    7    1    9    6    9    0     7     B
## samp6    9    0    3    5    9    1    7    9     0     B
## samp7    6    4    4    8    0    0    6    7     8     C
## samp8    0    4    2    2    8    1    2    1     9     C
## samp9    0   10    2    4    0    9    2    2     6     C
```

4. repeat this on the original file by creating a data frame with the groups and combining data sets

```
# prepare files for merging
Cex5 <- rownames_to_column(Cex, var="Sample")
group <- c("A", "A", "A", "B", "B", "B", "C", "C", "C")
Sample <- Cex5$Sample
Cex6 <- data.frame(Sample, group)

#base R merge
Cex7 <- merge(Cex5, Cex6, by="Sample")

#dplyr
Cex8 <- dplyr::full_join(Cex5, Cex6, by="Sample")
```

5. summarize your data (sum, mean, sd) overall and by the categories in the new column

```
Cex_sums <- Cex8 %>%
  dplyr::group_by(group) %>%
  dplyr::summarise_if(is.numeric, sum, na.rm=TRUE)
View(Cex_sums)

Cex_avg <- Cex8 %>%
  dplyr::group_by(group) %>%
  dplyr::summarise_if(is.numeric, mean, na.rm=TRUE)
View(Cex_avg)

Cex_sd <- Cex8 %>%
  dplyr::group_by(group) %>%
  dplyr::summarise_if(is.numeric, sd, na.rm=TRUE)
View(Cex_sd)
```

6. if you found 1-5 very easy, try to convert from short format to long format data using tidyr::pivot_longer and then back again using tidyr::pivot_wider

- https://tidyr.tidyverse.org/reference/pivot_longer.html
- https://tidyr.tidyverse.org/reference/pivot_wider.html

```
Cex_long <- Cex8 %>%
  tidyr::pivot_longer(cols=starts_with("var"), names_to="var", values_to="count")

Cex_short <- Cex_long %>%
  tidyr::pivot_wider(names_from="var", values_from="count")
```

7. save at least one data file under a new name using the write.csv command and push to GitHub

```
write.csv(Cex_short, "CodingExampleFileUpdated.csv")
```

Session Info

```
sessionInfo()
```

```
## R version 4.1.2 (2021-11-01)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] forcats_0.5.1   stringr_1.4.0   dplyr_1.0.7     purrr_0.3.4
## [5] readr_2.1.1     tidyr_1.1.4     tibble_3.1.6    ggplot2_3.3.5
## [9] tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.1.1 xfun_0.28        haven_2.4.3      colorspace_2.0-2
##  [5] vctrs_0.3.8      generics_0.1.1   htmltools_0.5.2  yaml_2.2.1
##  [9] utf8_1.2.2       rlang_0.4.12     pillar_1.6.4     glue_1.5.0
## [13] withr_2.4.3      DBI_1.1.1        dbplyr_2.1.1     modelr_0.1.8
## [17] readxl_1.3.1     lifecycle_1.0.1  munsell_0.5.0    gtable_0.3.0
## [21] cellranger_1.1.0 rvest_1.0.2      evaluate_0.14    knitr_1.36
## [25] tzdb_0.2.0       fastmap_1.1.0    fansi_0.5.0      broom_0.7.10
## [29] Rcpp_1.0.7       scales_1.1.1     backports_1.4.0  jsonlite_1.7.2
## [33] fs_1.5.2         hms_1.1.1        digest_0.6.28    stringi_1.7.6
## [37] grid_4.1.2       cli_3.1.0        tools_4.1.2      magrittr_2.0.1
## [41] crayon_1.4.2     pkgconfig_2.0.3  ellipsis_0.3.2   xml2_1.3.3
## [45] reprex_2.0.1     lubridate_1.8.0  rstudioapi_0.13  assertthat_0.2.1
## [49] rmarkdown_2.11   httr_1.4.2       R6_2.5.1         compiler_4.1.2
```