

Weekly Summary Template

Brady Miller

Table of contents

Tuesday, Jan 24	1
Thursday, Jan 26	4

Tuesday, Jan 24

! TIL

Include a *very brief* summary of what you learnt in this class here.

Today, I learnt the following concepts in class:

1. How to read in csv's through functions such as read.csv, read.table or read.delim & also how to hard code a data frame
2. Learned about what dplyr is and some of its verbs and their functions
3. Learned about ggplot, some of its geoms and how to create plots using its aesthetics.

```
library(tidyverse)
```

```
-- Attaching packages ----- tidyverse 1.3.2 --
v ggplot2 3.4.0      v purrr   1.0.1
v tibble  3.1.8      v dplyr   1.0.10
v tidyr   1.3.0      v stringr 1.5.0
v readr   2.1.3      v forcats 0.5.2
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library(knitr)
```

In class we learned how to hard code in a data frame and that the kable function can help 'clean up' tables (make them look better)

```
data_hard_code <- data.frame(  
  Name = c("Jake", "Ryan", "Emma"),  
  Age = c(25,44,31),  
  Height = c(5.10,6.1,5.5)  
)  
  
data_hard_code %>%  
  knitr::kable()
```

Name	Age	Height
Jake	25	5.1
Ryan	44	6.1
Emma	31	5.5

```
# data <- read.csv("file path.csv") is used to read in a csv file
```

Some of the dplyr verbs that we learned about and some examples of using them are below

```
# Select  
head(mpg,10) %>%  
  select(c(model,drv, cty)) %>%  
  knitr::kable()
```

model	drv	cty
a4	f	18
a4	f	21
a4	f	20
a4	f	21
a4	f	16
a4	f	18
a4	f	18

model	drv	cty
a4 quattro	4	18
a4 quattro	4	16
a4 quattro	4	20

```
# Mutate
iris %>%
  mutate(sepal_area = Sepal.Length * Sepal.Width) %>%
  head(10) %>%
  knitr::kable()
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species	sepal_area
5.1	3.5	1.4	0.2	setosa	17.85
4.9	3.0	1.4	0.2	setosa	14.70
4.7	3.2	1.3	0.2	setosa	15.04
4.6	3.1	1.5	0.2	setosa	14.26
5.0	3.6	1.4	0.2	setosa	18.00
5.4	3.9	1.7	0.4	setosa	21.06
4.6	3.4	1.4	0.3	setosa	15.64
5.0	3.4	1.5	0.2	setosa	17.00
4.4	2.9	1.4	0.2	setosa	12.76
4.9	3.1	1.5	0.1	setosa	15.19

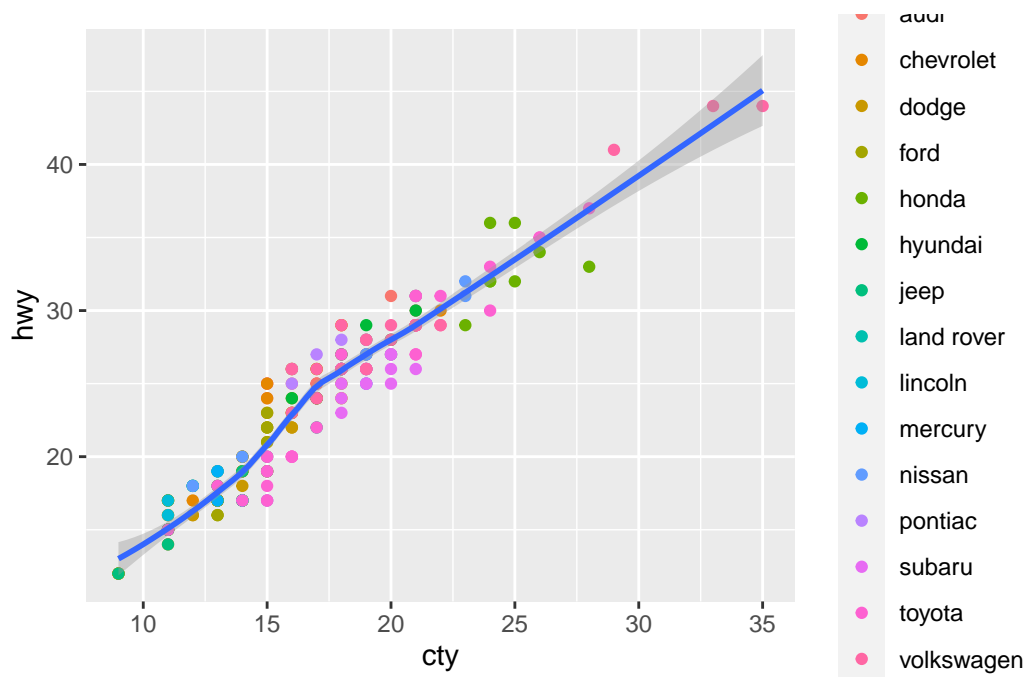
```
# Filter
head(mpg,10) %>%
  filter(hwy > 26) %>%
  head(., 10) %>%
  knitr::kable()
```

manufacturer	model	displ	year	cyl	trans	drv	cty	hwy	fl	class
audi	a4	1.8	1999	4	auto(l5)	f	18	29	p	compact
audi	a4	1.8	1999	4	manual(m5)	f	21	29	p	compact
audi	a4	2.0	2008	4	manual(m6)	f	20	31	p	compact
audi	a4	2.0	2008	4	auto(av)	f	21	30	p	compact
audi	a4	3.1	2008	6	auto(av)	f	18	27	p	compact
audi	a4	2.0	2008	4	manual(m6)	4	20	28	p	compact
	quattro									

Learned how to use ggplot with some basic geoms and aesthetics

```
# creates scatterplot of city miles/gallon vs hwy miles/gallon
# color encodes each point by its manufacturer
# adds a trend line between the city and highway mpg
plot <- ggplot(mpg)
plot <- plot +
  geom_point(aes(x = cty, y = hwy, color = manufacturer)) +
  geom_smooth(aes(x = cty, y = hwy))
plot
```

`geom_smooth()` using method = 'loess' and formula = 'y ~ x'



Thursday, Jan 26

! TIL

Include a *very brief* summary of what you learnt in this class here.
Today, I learnt the following concepts in class:

1. How to use ggThemeAssist

2. Some of the packages from the tidyverse library and what they can do
3. Discussed categorical variables/factors and how we can factor variables

Using `ggThemeAssist`, you can highlight the code below and add in all the extra theme features seen in the next code chunk

```
library(ggThemeAssist)
graph <- mpg %>%
  ggplot() +
  geom_point(aes(x = cty, y = hwy, color = class))

graph <- mpg %>%
  ggplot() +
  geom_point(aes(x = cty, y = hwy, color = class)) + theme(plot.title = element_text(face
panel.background = element_rect(linetype = "dashed"),
plot.background = element_rect(fill = "white",
  linetype = "dashed")) +labs(title = "City mpg vs Highway mpg",
  x = "City mpg", y = "Highway mpg")
```

Some packages in tidyverse that we discussed were `purrr`, `forcats` and `readr`

`Purrr` provides lots of functional programming tools

```
# creating an ixi matrix and computes the average of the elements

results = c()
for (i in 1:5) {
  M <- matrix(
    runif(i*i), nrow = i
  )
  results[i] <- mean(M)
}
results
```

```
[1] 0.8856155 0.7351595 0.5124889 0.4215193 0.5001513
```

```
# map() -> applies a function to each element of the matrix and returns the output as a li
map(
  1:5,
```

```

function(i){
  mean(
    matrix(
      runif(i*i), nrow = i
    )
  )
}
)

```

```

[[1]]
[1] 0.6384429

```

```

[[2]]
[1] 0.561755

```

```

[[3]]
[1] 0.4079518

```

```

[[4]]
[1] 0.5368825

```

```

[[5]]
[1] 0.5950676

```

Categorical Variables/Factors

We learned that by using `as.factor()`, we can break down a column containing categorical variables into its levels

```

var <- mpg$manufacturer
as.factor(var)

```

```

[1] audi      audi      audi      audi      audi      audi
[7] audi      audi      audi      audi      audi      audi
[13] audi      audi      audi      audi      audi      audi
[19] chevrolet chevrolet chevrolet chevrolet chevrolet chevrolet
[25] chevrolet chevrolet chevrolet chevrolet chevrolet chevrolet
[31] chevrolet chevrolet chevrolet chevrolet chevrolet chevrolet
[37] chevrolet dodge    dodge    dodge    dodge    dodge
[43] dodge     dodge     dodge     dodge     dodge     dodge
[49] dodge     dodge     dodge     dodge     dodge     dodge

```

[55]	dodge	dodge	dodge	dodge	dodge	dodge
[61]	dodge	dodge	dodge	dodge	dodge	dodge
[67]	dodge	dodge	dodge	dodge	dodge	dodge
[73]	dodge	dodge	ford	ford	ford	ford
[79]	ford	ford	ford	ford	ford	ford
[85]	ford	ford	ford	ford	ford	ford
[91]	ford	ford	ford	ford	ford	ford
[97]	ford	ford	ford	honda	honda	honda
[103]	honda	honda	honda	honda	honda	honda
[109]	hyundai	hyundai	hyundai	hyundai	hyundai	hyundai
[115]	hyundai	hyundai	hyundai	hyundai	hyundai	hyundai
[121]	hyundai	hyundai	jeep	jeep	jeep	jeep
[127]	jeep	jeep	jeep	jeep	land rover	land rover
[133]	land rover	land rover	lincoln	lincoln	lincoln	mercury
[139]	mercury	mercury	mercury	nissan	nissan	nissan
[145]	nissan	nissan	nissan	nissan	nissan	nissan
[151]	nissan	nissan	nissan	nissan	pontiac	pontiac
[157]	pontiac	pontiac	pontiac	subaru	subaru	subaru
[163]	subaru	subaru	subaru	subaru	subaru	subaru
[169]	subaru	subaru	subaru	subaru	subaru	toyota
[175]	toyota	toyota	toyota	toyota	toyota	toyota
[181]	toyota	toyota	toyota	toyota	toyota	toyota
[187]	toyota	toyota	toyota	toyota	toyota	toyota
[193]	toyota	toyota	toyota	toyota	toyota	toyota
[199]	toyota	toyota	toyota	toyota	toyota	toyota
[205]	toyota	toyota	toyota	volkswagen	volkswagen	volkswagen
[211]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen
[217]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen
[223]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen
[229]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen

15 Levels: audi chevrolet dodge ford honda hyundai jeep land rover ... volkswagen

Can use `fct_reorder()`, to reorder the values of a given categorical variable based on another variable.

```
# reorganizes the manufacturer levels based on hwy mpg in ascending order
fct_reorder(var, mpg$hwy, .desc = FALSE)
```

[1]	audi	audi	audi	audi	audi	audi
[7]	audi	audi	audi	audi	audi	audi
[13]	audi	audi	audi	audi	audi	audi
[19]	chevrolet	chevrolet	chevrolet	chevrolet	chevrolet	chevrolet

[25]	chevrolet	chevrolet	chevrolet	chevrolet	chevrolet	chevrolet
[31]	chevrolet	chevrolet	chevrolet	chevrolet	chevrolet	chevrolet
[37]	chevrolet	dodge	dodge	dodge	dodge	dodge
[43]	dodge	dodge	dodge	dodge	dodge	dodge
[49]	dodge	dodge	dodge	dodge	dodge	dodge
[55]	dodge	dodge	dodge	dodge	dodge	dodge
[61]	dodge	dodge	dodge	dodge	dodge	dodge
[67]	dodge	dodge	dodge	dodge	dodge	dodge
[73]	dodge	dodge	ford	ford	ford	ford
[79]	ford	ford	ford	ford	ford	ford
[85]	ford	ford	ford	ford	ford	ford
[91]	ford	ford	ford	ford	ford	ford
[97]	ford	ford	ford	honda	honda	honda
[103]	honda	honda	honda	honda	honda	honda
[109]	hyundai	hyundai	hyundai	hyundai	hyundai	hyundai
[115]	hyundai	hyundai	hyundai	hyundai	hyundai	hyundai
[121]	hyundai	hyundai	jeep	jeep	jeep	jeep
[127]	jeep	jeep	jeep	jeep	land rover	land rover
[133]	land rover	land rover	lincoln	lincoln	lincoln	mercury
[139]	mercury	mercury	mercury	nissan	nissan	nissan
[145]	nissan	nissan	nissan	nissan	nissan	nissan
[151]	nissan	nissan	nissan	nissan	pontiac	pontiac
[157]	pontiac	pontiac	pontiac	subaru	subaru	subaru
[163]	subaru	subaru	subaru	subaru	subaru	subaru
[169]	subaru	subaru	subaru	subaru	subaru	toyota
[175]	toyota	toyota	toyota	toyota	toyota	toyota
[181]	toyota	toyota	toyota	toyota	toyota	toyota
[187]	toyota	toyota	toyota	toyota	toyota	toyota
[193]	toyota	toyota	toyota	toyota	toyota	toyota
[199]	toyota	toyota	toyota	toyota	toyota	toyota
[205]	toyota	toyota	toyota	volkswagen	volkswagen	volkswagen
[211]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen
[217]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen
[223]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen
[229]	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen	volkswagen

15 Levels: land rover dodge lincoln ford mercury jeep chevrolet audi ... honda