Team #: 4
Your name: Julia Campbell

Instructions:

First, make a copy of this document (go to File then Make a copy).

Name the document “project4 submission [your name here]”

This is an individual assignment. You will be making 6 graphs in total. Two maps (tutorials 11 & 12). Two bar graphs (tutorial 13). Two line graphs (tutorial 13).

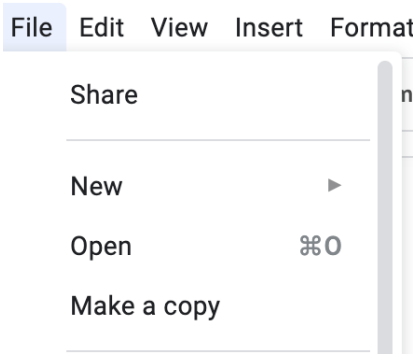
The tutorials walk you through how to create my choice of graph, but you’ll need to create your own version with *at least* one change from my version.

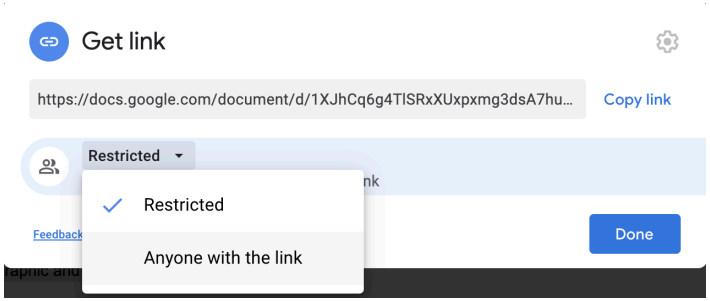
Learning goals: Geographic and temporal joining.

CCLE: When you are done with the deliverables,

1. go to the “Share” button on the top right of Google Doc.
2. Set the document so that “anyone with link” can view.
3. Upload the link on CCLE.

Please list your team number at the top!





Create a workplan for making the bar graph with both *county group and season* here

- Use “action” words like: create, join, match, append, summarise (you do not necessarily need to include R code)
- Add example tables along the way - you can copy and paste screenshots if needed.
- You can illustrate from left to right (as I’ve done here). Or, you can go from top to bottom over several pages.
- Work backward from the bar graph to start. What does the data look like?

Step 1: Import all raw data

Step 2: Create county boundaries in CA

Step 3: Match weather stations to counties -

	station	countyfip
1	USW00023188	73
2	USR0000CCOL	93
3	USR0000CLHO	65
4	USW00024259	93
5	USW00023225	61
6	USW00093209	79
7	USR0000CRIC	65
8	USR0000CFIS	25
9	USR0000CWOF	29

	station	name	geometry
1	USW00023188	SAN DIEGO INTERNATIONAL AIRPORT, CA US	c(-117.183, 32.7336)
2	USR0000CCOL	COLLINS BALDY CALIFORNIA, CA US	c(-122.95, 41.775)
3	USR0000CLHO	LOST HORSE CALIFORNIA, CA US	c(-116.188, 34.0178)
4	USW00024259	MONTAGUE SISKIYOU AIRPORT, CA US	c(-122.468, 41.7814)
5	USW00023225	BLUE CANYON AIRPORT, CA US	c(-120.71, 39.2774)
6	USW00093209	PASO ROBLES MUNICIPAL AIRPORT, CA US	c(-120.628, 35.6697)

Step 4: Find median average temperatures of all weather stations within each county

STATION	NAME	LATITUDE	LONGITUDE	ELEVATION	DATE	PRCP	TMAX	TMIN
USW00023188	SAN DIEGO INT	32.7336	-117.1831	4.6	1999-01-01	0	66	51
USW00023188	SAN DIEGO INT	32.7336	-117.1831	4.6	1999-01-02	0	71	46
USW00023188	SAN DIEGO INT	32.7336	-117.1831	4.6	1999-01-03	0	79	47
USW00023188	SAN DIEGO INT	32.7336	-117.1831	4.6	1999-01-04	0	73	47
USW00023188	SAN DIEGO INT	32.7336	-117.1831	4.6	1999-01-05	0	75	47

Step 5: Find death rates of each county by joining population data with deaths data

State	State Code	County	County Code	Yearly July 1st E:	Yearly July 1st E:	Population
California	6	Alameda County,	6001	1990	1990	1306166
California	6	Alameda County,	6001	1991	1991	1318543
California	6	Alameda County,	6001	1992	1992	1332208
California	6	Alameda County,	6001	1993	1993	1339189

County	County Code	Year	Year Code	Month	Month Code	Deaths
Alameda County,	6001	1999	1999	Jan., 1999	1999/01	961
Alameda County,	6001	1999	1999	Feb., 1999	1999/02	944
Alameda County,	6001	1999	1999	Mar., 1999	1999/03	886

Step 6: Join all counties with higher than average temperatures (“hot county”) and join all counties with lower than average temperatures (“cold county”)

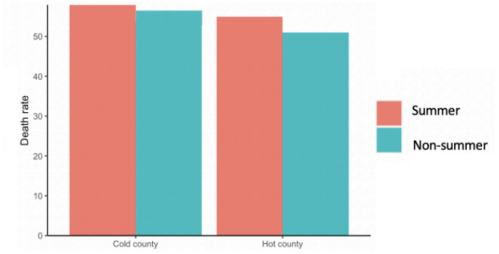
Step 7: Separate summer data from non-summer data for both counties

Step 8: Summarize the data (summer/non-summer in all counties with death rates, “cold” vs “hot”)

Step 9: Create the bar graph

Step 10: Design the bar graph (experiment with colors, label the axis and legend, etc)

Step 11: Bar graph

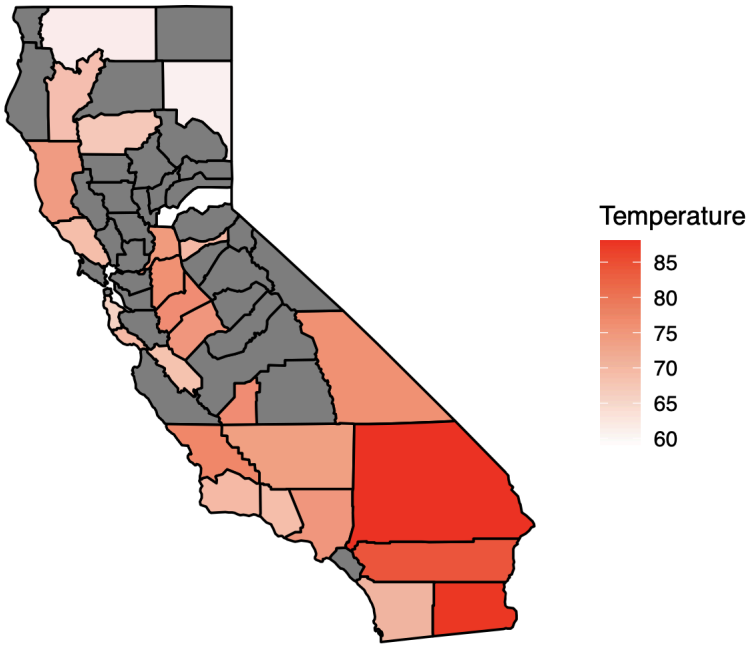
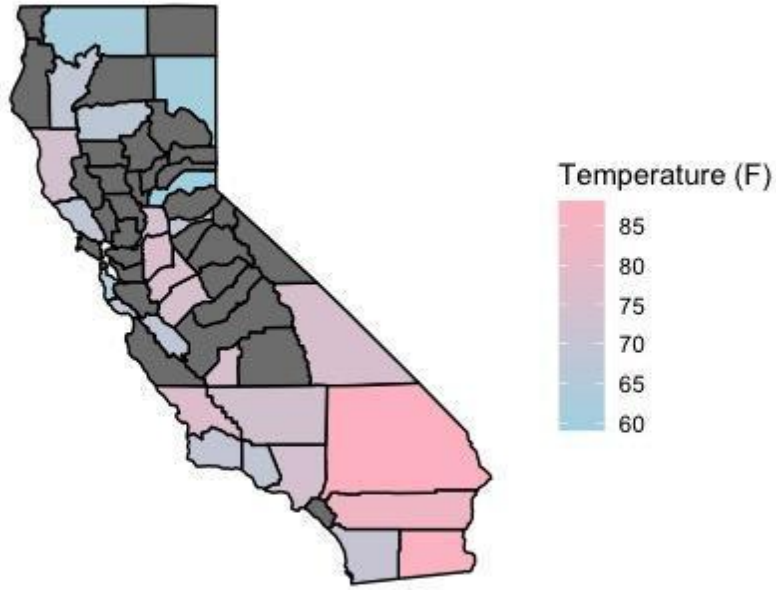


What you changed from my version to your version of the map: I changed the fill, outline, and point colors

My map	**Your map**

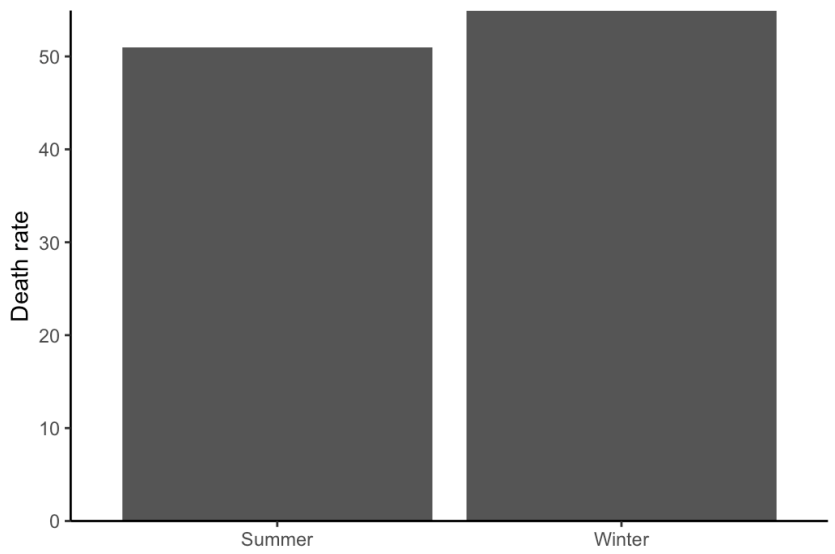
This space is intentionally blank

What you changed from my version to your version of the map: Changed the gradient to “light blue” - “pink”

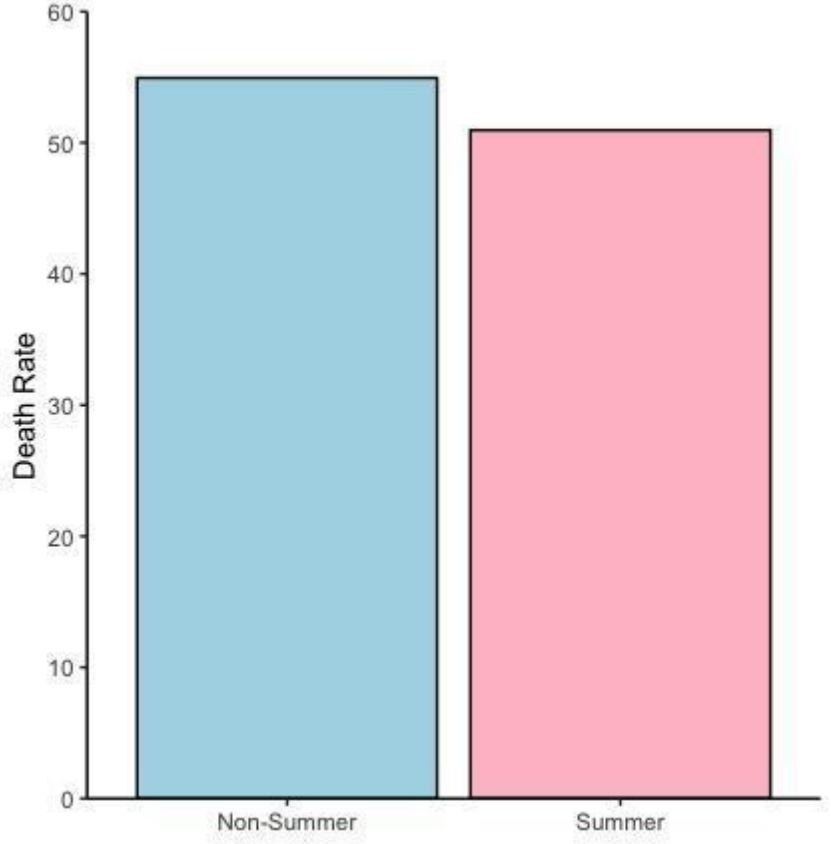
My map	**Your map**
	
This space is intentionally blank	

What you changed from my bar graph (hot counties only) to your bar graph: Made them separate colors (non-summer is blue, summer is pink) and put y scale limit to 60

My bar graph



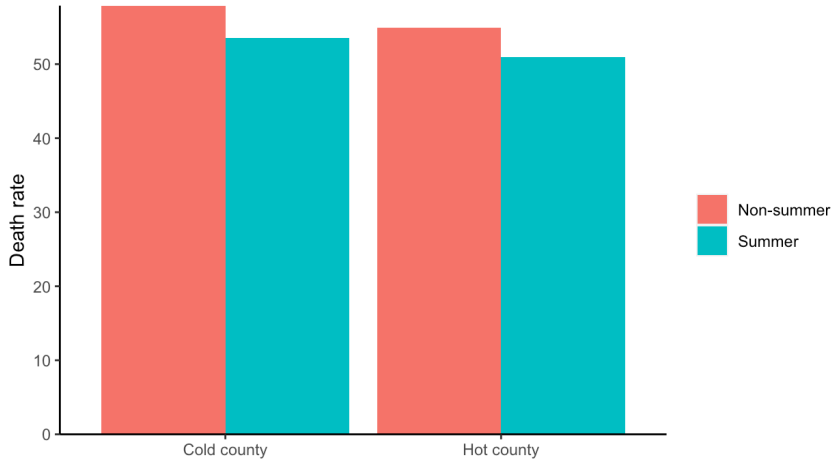
Your bar graph



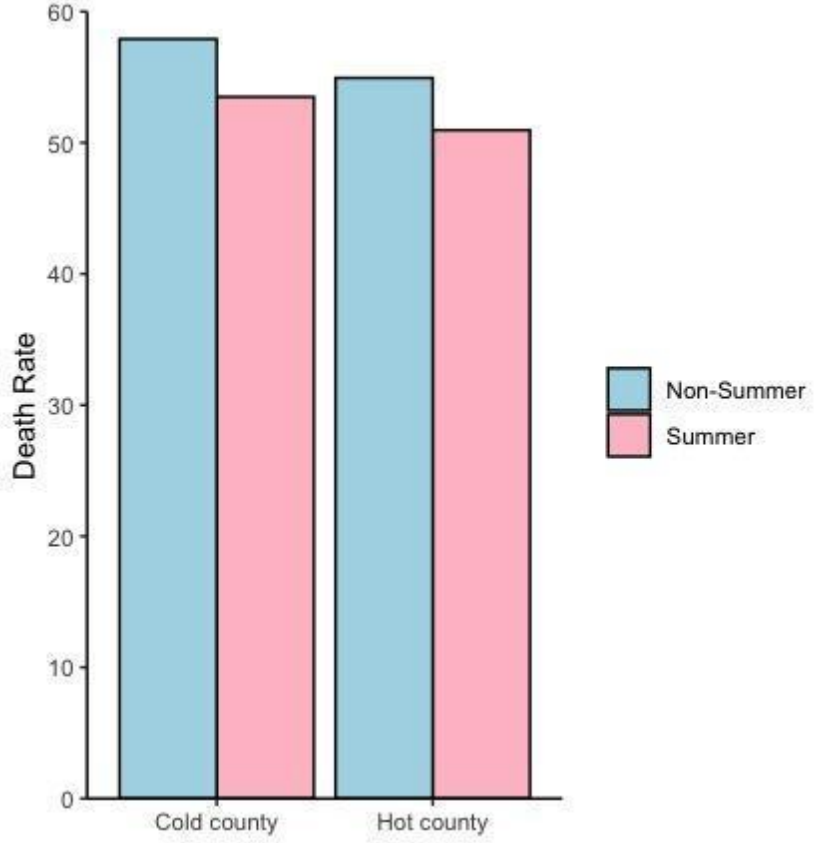
This space is intentionally blank

What you changed from my bar graph (both county groups) to your bar graph: Changed the colors to light blue and pink, outlined with black, and set y limit to 60

My bar graph



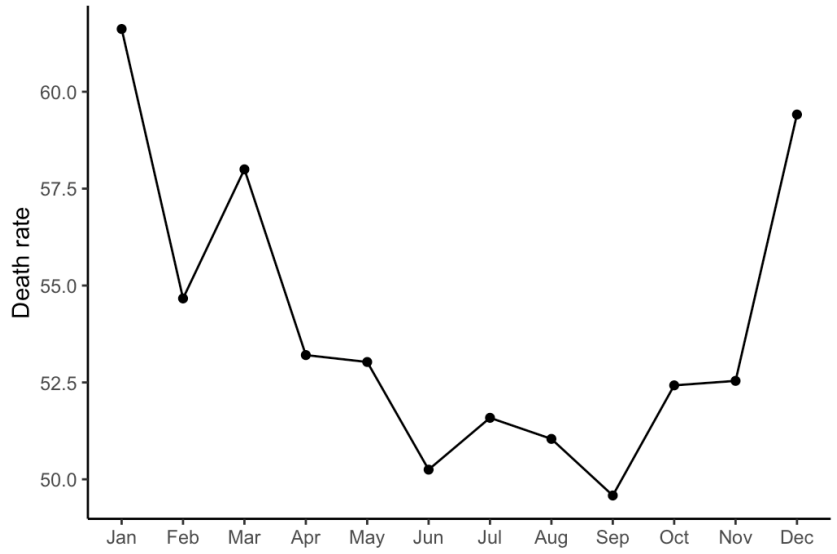
Your bar graph



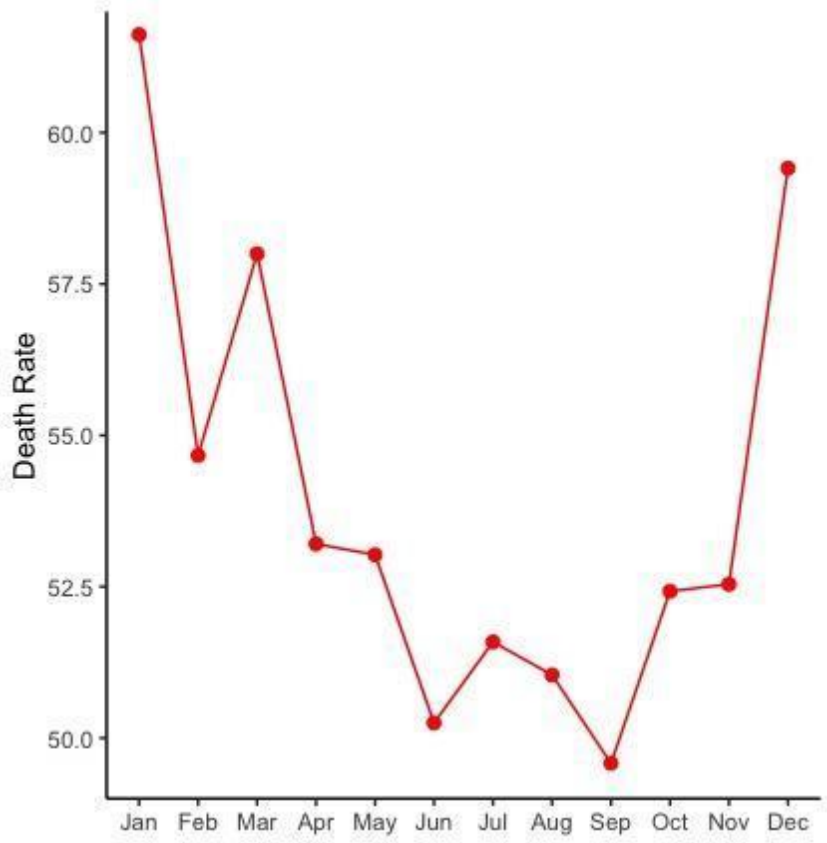
This space is intentionally blank

What you changed from my line graph (hot counties only) to your line graph: Changed the color of both the points and line to red and made the points slightly bigger

My line graph



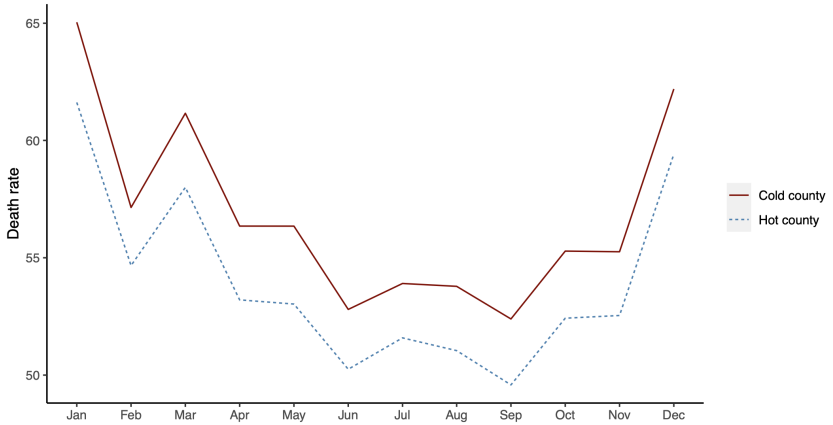
Your line graph



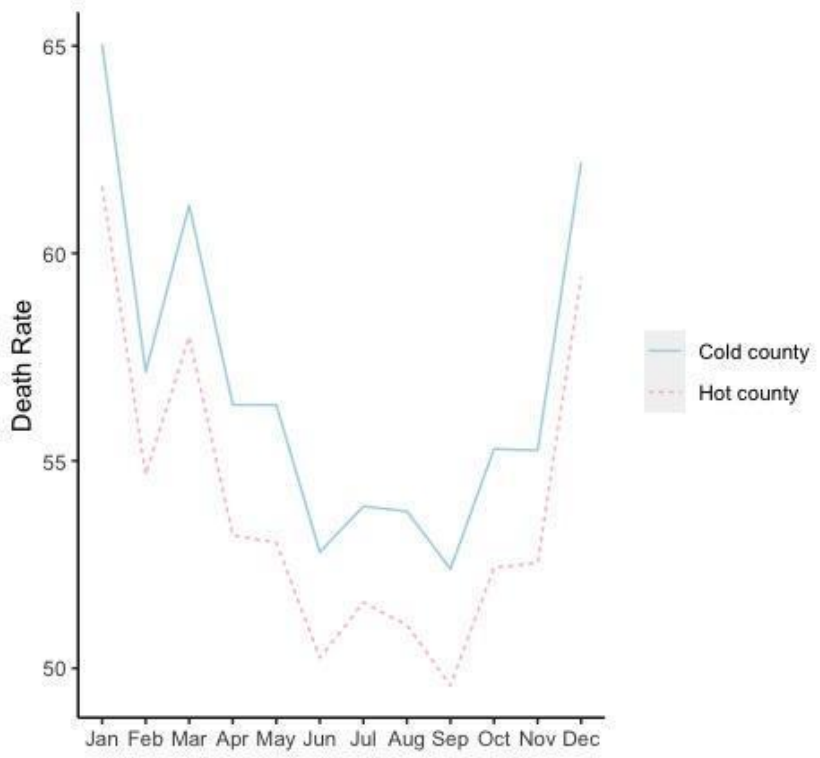
This space is intentionally blank

What you changed from my line graph (both county groups) to your line graph: Made cold county light blue and hot county pink

My line graph



Your line graph



This space is intentionally blank

Copy and paste your entire R script here.

```
# PROJECT 4
# MAY 20TH, 2020
# JULIA CAMPBELL


# LOAD LIBRARIES
library(ggplot2)
library(dplyr)
library(readr)
library(lubridate)
library(janitor)
library(maps)
library(mapdata)
library(ggmap)
library(sf)
library(spatialEco)
library(taRifx)

####################
# TUTORIAL 11 CODE #
####################

# IMPORT DATA
stations_california <- read_csv("~/Documents/big_enviro/data/ghcn/ghcn_stations_california.csv")
stations_california

# GET COUNTY BOUNDARIES
map_us_counties <- map_data("county")
map_us_counties <- rename(map_us_counties, state=region, county_name=subregion)
map_california_counties <- filter(map_us_counties, state=="california")
```

```
# STATION MAP
ggplot(map_california_counties, aes(y=lat, x=long, group=group)) +
  geom_polygon(fill="light blue", color="dark blue") +
  geom_point(data=stations_california, aes(y=latitude, x=longitude, group=station), size=0.75, color="red") +
  coord_fixed(1.3) +
  theme(panel.background = element_rect(fill="white"),
        axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.title.y=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank() )
ggsave("~/Documents/big_enviro/output/tutorials/map_stations_california.pdf")

# READ IN SHAPEFILE
counties_sf <- st_read("~/Documents/big_enviro/data/Tiger/tl_2019_us_county/tl_2019_us_county.shp")

# RESTRICT TO CALIFORNIA
counties_sf <- filter(counties_sf, STATEFP=="06")

# CONVERT STATION DATA INTO SF
stations_sf = st_as_sf(stations_california, coords = c("longitude", "latitude"))

# SET COORDINATES OF MONITOR DATA
st_crs(stations_sf) <- 4269

# COUNTIES CONTAINING STATIONS
county_station_within <- point.in.poly(stations_sf, counties_sf)

# CONVERT TO DATA FRAME
county_station_within <- as.data.frame(county_station_within)
```

```
# LOWERCASE NAMES
county_station_within <- clean_names(county_station_within)

# RENAME FIP VAR
county_station_within <- rename(county_station_within, countyfip=countyfp)

# KEEP ONLY NEEDED VARS
county_station_within <- select(county_station_within, station, countyfip)

# DESTRING FIP VAR
county_station_within <- mutate(county_station_within, countyfip=destring(countyfip))

# JOIN WITH DISTANCE MATRIX
ghcn_stn_within <- full_join(county_station_within, ghcn_california_1999_2018, by = "station")

#####################################################
# COLLAPSE WEATHER TO COUNTY-MONTH LEVEL AVERAGE #
#####################################################

# GROUP
ghcn_stn_within <- group_by(ghcn_stn_within, countyfip, year, month)

#SUMMARISE
ghcn_county_within <- summarise(ghcn_stn_within, tmax=mean(tmax, na.rm = TRUE))

#UNGROUP
ghcn_stn_within <- ungroup(ghcn_stn_within)

#########################
# CLEANING DATA PROCESS #
#########################
```

```
# IMPORT CDC DATA
cdc_california_1999_2018 <- read_delim("~/Documents/big_enviro/data/cdc_california_1999_2018.txt", "\t", escape_double =
FALSE, col_types = cols(`Month Code` = col_date(format = "%Y/%m")), trim_ws = TRUE)

# CLEAN NAMES FOR CDC
cdc_california_1999_2018 <- clean_names(cdc_california_1999_2018)

# ADD YEAR AND MONTH
cdc_california_1999_2018 <- mutate(cdc_california_1999_2018, year=year(month_code), month=month(month_code))

# SELECT RELEVANT DATA FOR CDC
cdc_california_1999_2018 <- select(cdc_california_1999_2018, county_code, deaths, year, month)

# IMPORT POPULATION DATA
pop_california_1990_2018 <- read_delim("Documents/big_enviro/data/pop_california_1990_2018.txt", "\t", escape_double = FALSE,
trim_ws = TRUE)

# CLEAN NAMES FOR POP
pop_california_1990_2018 <- clean_names(pop_california_1990_2018)

# RENAME
pop_california_1990_2018 <- rename(pop_california_1990_2018, year=yearly_july_1st_estimates)

# SELECT RELEVANT DATA FOR POP
pop_california_1990_2018 <- select(pop_california_1990_2018, county_code, year, population)

# ONLY YEARS 1999-2018
pop_california_1999_2018 <- filter(pop_california_1990_2018, year>=1999)

# JOIN POP AND CDC DATA
cdc_pop_1999_2018 <- full_join(pop_california_1999_2018, cdc_california_1999_2018, by = c("county_code", "year"))
```

```
# DESTRING
cdc_pop_1999_2018 <- mutate(cdc_pop_1999_2018, county_code=destring(county_code), population=destring(population),
deaths=destring(deaths))

# ADD COUNTY CODE
cdc_pop_1999_2018 <- mutate(cdc_pop_1999_2018, countyfip=county_code-6000)

# ADD DEATHRATES
cdc_pop_1999_2018 <- mutate(cdc_pop_1999_2018, deathrate=deaths/population*100000)

# VIEW DEATHRATES
summary(cdc_pop_1999_2018$deathrate)

# IMPORT GHCN DATA
ghcn_california_1999_2019 <- read_csv("~/Documents/big_enviro/data/ghcn/ghcn_california_1999_2019.csv", col_types =
cols(DATE = col_date(format = "%Y-%m-%d")))

# CLEAN NAMES FOR GHCN
ghcn_california_1999_2019 <- clean_names(ghcn_california_1999_2019)

# SELECT RELEVANT DATA FOR GHCN
ghcn_california_1999_2019 <- select(ghcn_california_1999_2019, -name, -latitude, -longitude, -elevation)

# ADD YEAR AND MONTH
ghcn_california_1999_2019 <- mutate(ghcn_california_1999_2019, year=year(date), month=month(date))

# ONLY YEARS 1999-2019
ghcn_california_1999_2018 <- filter(ghcn_california_1999_2019, year<2019)

# VIEW GHCN
summary(ghcn_california_1999_2018$tmax)
```

```
##############################################
# JOIN CDC + GHCN DATA AT COUNTY-YEAR-MONTH #
##############################################

cdc_ghcn_within <- full_join(ghcn_county_within, cdc_pop_1999_2018, by = c("countyfip", "year", "month"))

###################################
# SUMMARISE / COLLAPSE TO COUNTY #
###################################

#GROUP
cdc_ghcn_within <- group_by(cdc_ghcn_within, countyfip)

#SUMMARISE
cdc_ghcn_within_county_avg <- summarise(cdc_ghcn_within, tmax_avg=weighted.mean(tmax, population, na.rm = TRUE))

################
# PREPARE MAP #
################

# GET COUNTY BOUNDARIES
map_us_counties <- map_data("county")
map_us_counties <- rename(map_us_counties, state=region, county_name=subregion)
map_california_counties <- filter(map_us_counties, state=="california")

# READ IN COUNTY NAMES
fips_names_california <- read_csv("~/Documents/big_enviro/data/fips_names/fips_names_california.csv")

# JOIN THE NAMES W FIPS + NAMES
map_california_counties <- full_join(map_california_counties, fips_names_california, by = "county_name")
```

```
# FULL JOIN
final_map <- full_join(cdc_ghcn_within_county_avg, map_california_counties, by = "countyfip")

# TEMPERATURE MAP
ggplot(final_map, aes(y=lat, x=long, group=group, fill=tmax_avg)) +
  geom_polygon(color="black") +
  coord_fixed(1.3) +
  theme(panel.background = element_rect(fill="white"),
        axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.title.y=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank()) +
scale_fill_gradient(name = "Temperature (F)", low="light blue", high="pink", breaks = seq(60, 85, 5))
ggsave("~/Documents/big_enviro/output/tutorials/map_temperature_california.pdf")

# HOT/COLD BY MONTH
cdc_ghcn_within_county_avg <- mutate(cdc_ghcn_within_county_avg,
hotgroup=ifelse(tmax_avg>median(cdc_ghcn_within_county_avg$tmax_avg, na.rm=”TRUE”), “Hot county”, “Cold county”))

# JOIN HOT/COLD GROUP TO COUNTY
county_hotgroup_month <- full_join(cdc_ghcn_within, cdc_ghcn_within_county_avg, by = “countyfip”)

# GROUP BY COUNTY AND MONTH
county_hotgroup_month <- group_by(county_hotgroup_month, hotgroup, month)
final_county_hotgroup_month <- summarise(county_hotgroup_month, tmax = weighted.mean(tmax, population, na.rm = TRUE),
deathrate=weighted.mean(deathrate, population, na.rm = TRUE))
final_county_hotgroup_month <- na.omit(final_county_hotgroup_month, hotgroup)

# SUMMER/NON-SUMMER GROUP
county_hotgroup_seas <- mutate(county_hotgroup_month, season=ifelse(month >= 6 & month <= 8, “Summer”, “Non-Summer”))
```

```
# GROUP BY COUNTY AND MONTH
county_hotgroup_seas <- group_by(county_hotgroup_seas, season)

# SUMMARIZE
final_county_hotgroup_seas <- summarise(county_hotgroup_seas, tmax=weighted.mean(tmax, population, na.rm = TRUE),
deathrate=weighted.mean(deathrate, population, na.rm = TRUE))
final_county_hotgroup_seas <- na.omit(final_county_hotgroup_seas, season)

#################
# BAR GRAPHS #
#################

# COUNTY BY MONTH
final_county_hotgroup_month <- read_csv("~/Documents/big_enviro/data/project4/final_county_hotgroup_month.csv")

# COUNTY BY SEASON
final_county_hotgroup_seas <- read_csv("~/Documents/big_enviro/data/project4/final_county_hotgroup_seas.csv")

# HOT COUNTIES ONLY
hot_bar <- filter(final_county_hotgroup_seas, hotgroup=="Hot county")

# HOT COUNTY BAR GRAPH
ggplot(hot_bar, aes(fill=season, y=deathrate, x=season)) +
  geom_bar(stat="identity", color="black") +
  scale_fill_manual(values = c("light blue", "pink")) +
  labs(y ="Death Rate") +
  scale_y_continuous(expand = c(0,0), limits = c(0,60)) +
  theme(panel.background = element_rect(fill="white"),
        axis.title.x = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = "none" )
```

```
ggsave("~/Documents/big_enviro/output/tutorials/bar_deathrate_seas_hotgroup_california_1999_2018.pdf")

# BOTH COUNTIES BAR GRAPH
ggplot(final_county_hotgroup_seas, aes(fill=season, y=deathrate, x=hotgroup)) +
  geom_bar(position="dodge", stat="identity", color="black") +
  scale_fill_manual(values = c("light blue", "pink")) +
  labs(y = "Death Rate") +
  scale_y_continuous(expand = c(0,0), limits = c(0,60)) +
  theme(panel.background = element_rect(fill="white"),
        axis.title.x = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.title = element_blank() )
ggsave("~/Documents/big_enviro/output/tutorials/bar_deathrate_seas_bothgroup_california_1999_2018.pdf")

#################
# LINE GRAPHS #
#################

# HOT COUNTIES ONLY
hot_line <- filter(final_county_hotgroup_month, hotgroup=="Hot county")

# HOT COUNTY LINE GRAPH
ggplot(hot_line, aes(y=deathrate, x=month)) +
  geom_point(color="red", size=2) +
  geom_line(color="red") +
  labs(y = "Death Rate", x = "Month") +
  scale_y_continuous(expand = c(0,0), limits = c(49,62)) +
  scale_x_continuous(breaks = 1:12, labels = c('Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec')) +
  theme(panel.background = element_rect(fill="white"),
        legend.position = "none",
        axis.line = element_line(colour = "black"),
        axis.title.x=element_blank())
```

```
ggsave("~/Documents/big_enviro/output/tutorials/scat_deathrate_month_hotgroup_california_1999_2018.pdf")

# BOTH COUNTIES LINE GRAPH
ggplot(final_county_hotgroup_month, aes(x = month, y = deathrate)) +
  geom_line(aes(color = hotgroup, linetype=hotgroup)) +
  scale_color_manual(values = c("light blue", "pink")) +
  labs(y = "Death Rate") +
  scale_x_continuous(breaks = 1:12, labels = c('Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec')) +
  theme(panel.background = element_rect(fill="white"),
        axis.line = element_line(colour = "black"),
        axis.title.x=element_blank(),
        legend.title = element_blank())
ggsave("~/Documents/big_enviro/output/tutorials/scat_deathrate_month_bothgroup_california_1999_2018.pdf")
```