HPAM 7660 Data Assignment 4

Aidan Clements

February 22, 2024

1: The three essential components of data visualization: data, geom, and aes. Data are the set of variables we are looking at. Geom is short for the geometric object. This is the type of object we can see in a plot i.e. points, lines, or bars. Finally Aes refers to the aesthetic attributes of the geometric object. How are the points mapped into the dataset with positions, color, shape, etc.

2a: I am including the code for the parish_rates and cancer_alley_rates data frame here:

```
library(nycflights13)
library(readr)
library(ggplot2)
la_mort <-
  read_csv("https://www.dropbox.com/scl/fi/fzsnhfd3lq80v2o3sag6c/la_mort.csv?rlkey=h1vyjm2b8ppgejgsg3e8
```

```
## Rows: 642696 Columns: 29
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): stocr, strsd, stbrth, brthr, sex, marstat, ucod
## dbl (22): restatus, cntyocr, popcntyocr, cntyrsd, popcntyresd, educ1989, edu...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
la_mort$cancer_parish <- ifelse(la_mort$cntyrsd %in% c(5, 33, 47, 51, 71, 89, 93, 95, 121), 1, 0)

la_mort$cancer_parish <- ifelse(la_mort$cntyrsd %in% c(5, 33, 47, 51, 71, 89, 93, 95, 121), 1, 0)

la_mort$cancer39 <- ifelse(la_mort$ucr39 %in% c(5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15), 1, 0)

la_mort$cancer113 <- ifelse(la_mort$ucr113 %in% c(20:44), 1,0)

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##     filter, lag
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
parish_count <- la_mort%>%
  group_by(cntyrsd, cancer_parish, year) %>%
  summarize(cancer39 = sum(cancer39, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'cntyrsd', 'cancer_parish'. You can
## override using the `.groups` argument.
```

```r
summary(parish_count$cancer39)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     3.0    42.0    74.0   144.5   159.0   992.0
```

```r
la_pop <-
  read_csv("https://www.dropbox.com/scl/fi/650k1obpczky6bwa19ex6/la_county_pop.csv?rlkey=0aokd9m76q7mxw
```

```
## Rows: 24320 Columns: 23
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): stname, ctyname, agegrp
## dbl (20): state, county, year, tot_pop, tot_male, tot_female, wa_male, wa_fe...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
parish_count <- parish_count %>%
  rename(county = cntyrsd)
la_joined <- parish_count %>%
  inner_join(la_pop, by = c("county", "year"))
la_joined_all <- subset(la_joined, agegrp == "all")
la_joined_all$cancer_rate_total <- (la_joined_all$cancer39) / (la_joined_all$tot_pop)
summary(la_joined_all$cancer_rate_total)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0001157 0.0018691 0.0021703 0.0021985 0.0024863 0.0039361
```

```r
la_joined_all$cancer_rate_total <- ((la_joined_all$cancer39) / (la_joined_all$tot_pop / 100000))
parish_cancer_2019 <- subset(la_joined_all, year == 2019)
library(knitr)


la_mort$cancer_parish <- ifelse(la_mort$cntyrsd %in% c(5, 33, 47, 51, 71, 89, 93, 95, 121), 1, 0)


la_mort$cancer39 <- ifelse(la_mort$ucr39 %in% c(5:15), 1, 0)

library(dplyr)
la_mort_age <- la_mort %>%
  filter(age != 9999)
la_mort_age$age <- ifelse(la_mort_age$age < 2000, la_mort_age$age - 1000, 0)
```

```
age_breaks <- c(0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, Inf)
age_labels <- c("0_4", "5_9", "10_14", "15_19", "20_24", "25_29", "30_34", "35_39",
                "40_44", "45_49", "50_54", "55_59", "60_64", "65_69", "70_74",
                "75_79", "80_84", "85+")
la_mort_age$agegrp <- as.character(cut(la_mort_age$age, breaks = age_breaks, labels = age_labels, right

parish_count_age <- la_mort_age %>%
  group_by(cntyrsd, cancer_parish, agegrp, year) %>%
  summarize(cancer39 = sum(cancer39, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'cntyrsd', 'cancer_parish', 'agegrp'. You
## can override using the `.groups` argument.
```

```
la_pop <-
  read_csv("https://www.dropbox.com/scl/fi/650k1obpczky6bwa19ex6/la_county_pop.csv?rlkey=0aokd9m76q7mxw
```

```
## Rows: 24320 Columns: 23
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): stname, ctyname, agegrp
## dbl (20): state, county, year, tot_pop, tot_male, tot_female, wa_male, wa_fe...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
la_joined <- parish_count_age %>%
  inner_join(la_pop, by = c("cntyrsd" = "county", "year", "agegrp"))

stnrd_pop <-
  read_csv("https://www.dropbox.com/scl/fi/xzd2o5lza237so6vamqwb/stnrd_pop.csv?rlkey=zp90au2tuq6eptvi1y
```

```
## Rows: 18 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): agegrp
## dbl (1): stnrd_pop
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
la_joined_stnrd <- la_joined %>%
  inner_join(stnrd_pop, by = "agegrp")

la_joined_stnrd$stnrd_pop_weight <- (la_joined_stnrd$stnrd_pop) / (sum(stnrd_pop$stnrd_pop))

la_joined_stnrd$cancer_rate_adj <- ((la_joined_stnrd$cancer39) / (la_joined_stnrd$tot_pop / 100000)) * 1

parish_rates <- la_joined_stnrd %>%
  group_by(cntyrsd, cancer_parish, year) %>%
  summarize(cancer_rate_adj = sum(cancer_rate_adj, na.rm = TRUE), cancer39 = sum(cancer39), tot_pop =
              sum(tot_pop))
```

```
## `summarise()` has grouped output by 'cntyrsd', 'cancer_parish'. You can
## override using the `.groups` argument.
```

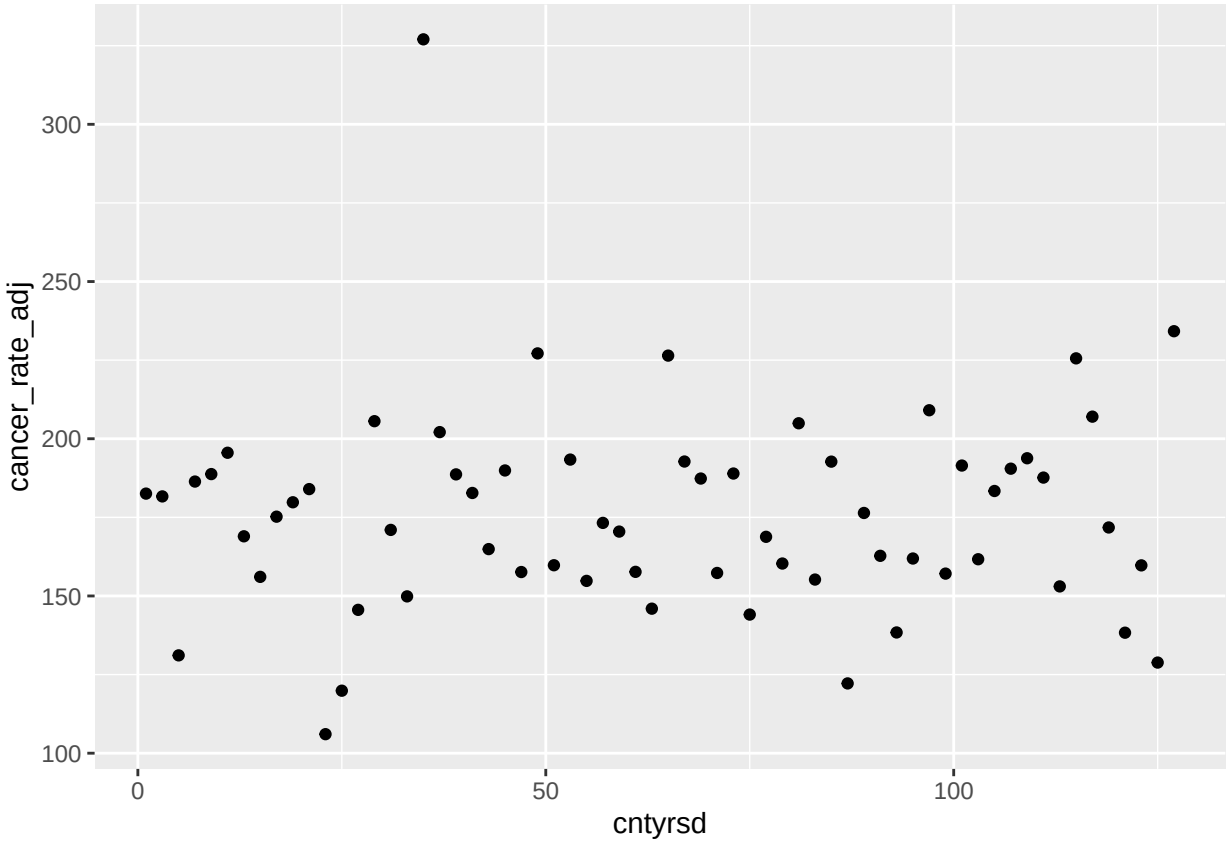
```
parish_rates$cancer_rate_crude <- (parish_rates$cancer39) / (parish_rates$tot_pop / 100000)

parish_cancer_2019 <- subset(la_joined_all, year == 2019)
library(knitr)

parish_cancer_2019 <- subset(parish_rates, year == 2019)

parish_rates$pop_weight <- (parish_rates$cancer_rate_adj) * (parish_rates$tot_pop)
cancer_alley_rates <- parish_rates %>%
  group_by(cancer_parish, year) %>%
  summarize(cancer_rate_adj_wt = sum(pop_weight) / sum(tot_pop))
```

```
## `summarise()` has grouped output by 'cancer_parish'. You can override using the
## `.groups` argument.
```

```
cancer_alley <-
  subset(cancer_alley_rates, cancer_parish == 1, select = c(cancer_rate_adj_wt, year)) %>%
  rename(cancer_alley_rate = cancer_rate_adj_wt)
no_cancer_alley <-
  subset(cancer_alley_rates, cancer_parish == 0, select = c(cancer_rate_adj_wt, year)) %>%
  rename(no_cancer_alley_rate = cancer_rate_adj_wt)
cancer_alley_table <- cancer_alley %>%
  inner_join(no_cancer_alley, by = "year")
cancer_alley_table <- cancer_alley_table[,c("year", "cancer_alley_rate", "no_cancer_alley_rate")]
```

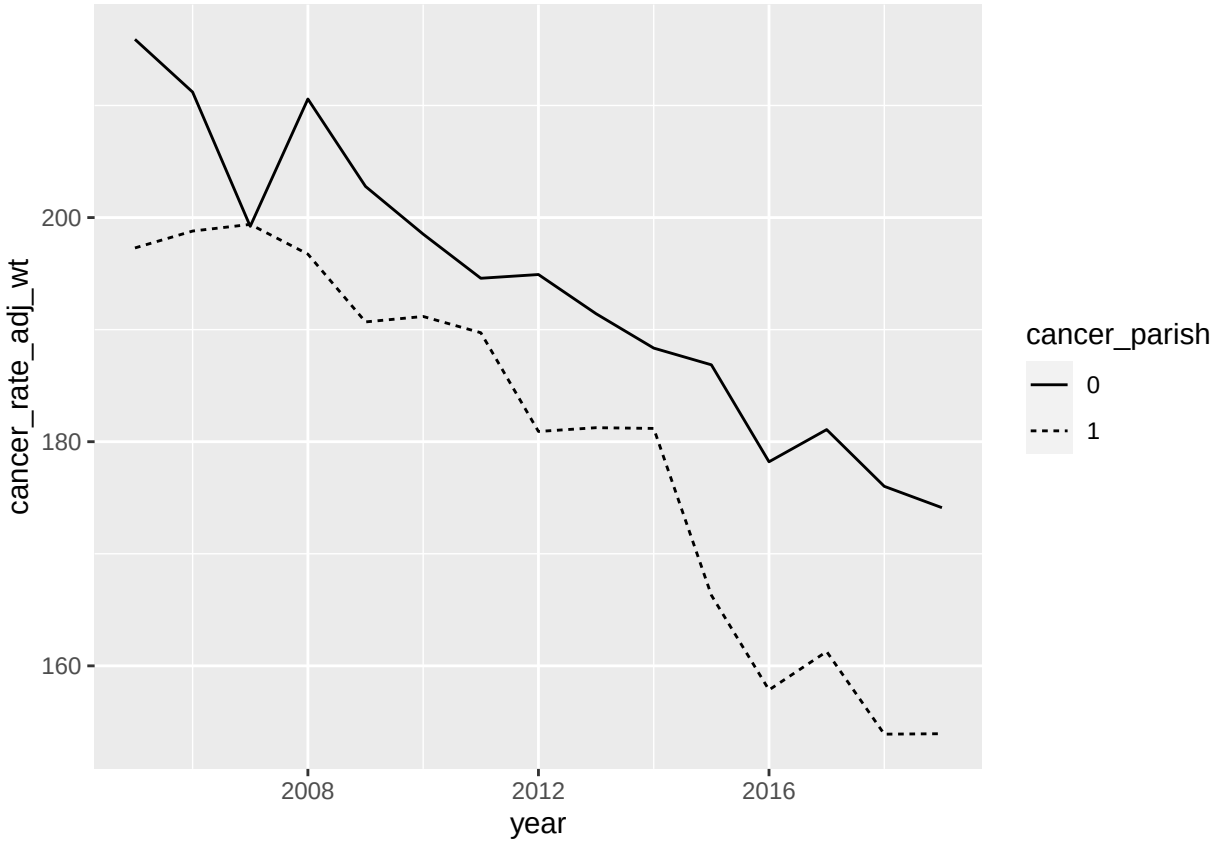
3: Now I will begin to use ggplot to create a scatterplot with county FIPS code on the x-axis and cancer mortality rates on the y-axis

```
    library(dplyr)
    library(knitr)
    library(ggplot2)
ggplot(data = parish_cancer_2019, year = 2019, mapping = aes(x = cntyrsd, y = cancer_rate_adj)) +
  geom_point()
```



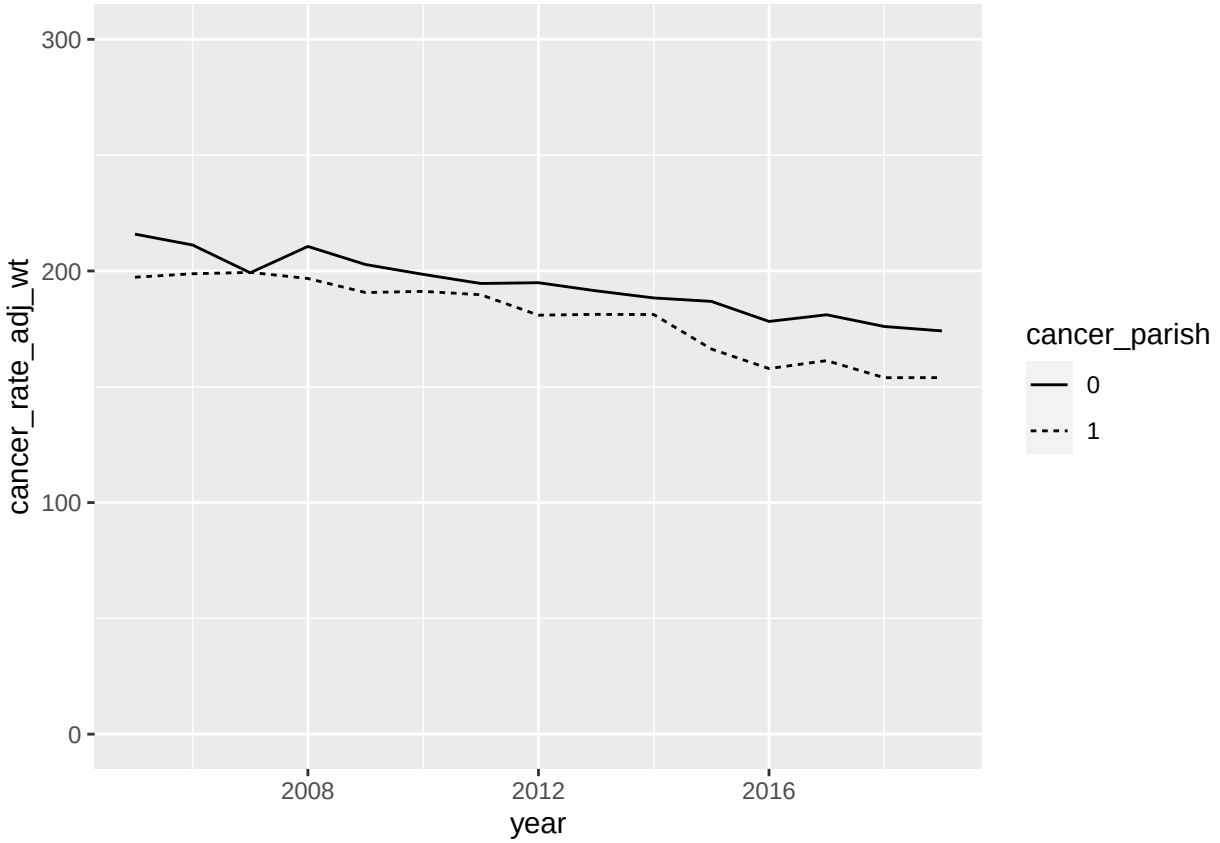
4: Create a linegraph that displays cancer mortality rates from 2005-2019 for both cancer alley and non-cancer alley parishes

```
cancer_alley_rates$cancer_parish <- factor(cancer_alley_rates$cancer_parish)
ggplot(data = cancer_alley_rates,
       mapping = aes(x = year, y = cancer_rate_adj_wt, group = cancer_parish)) +
         geom_line(aes(linetype= cancer_parish))
```



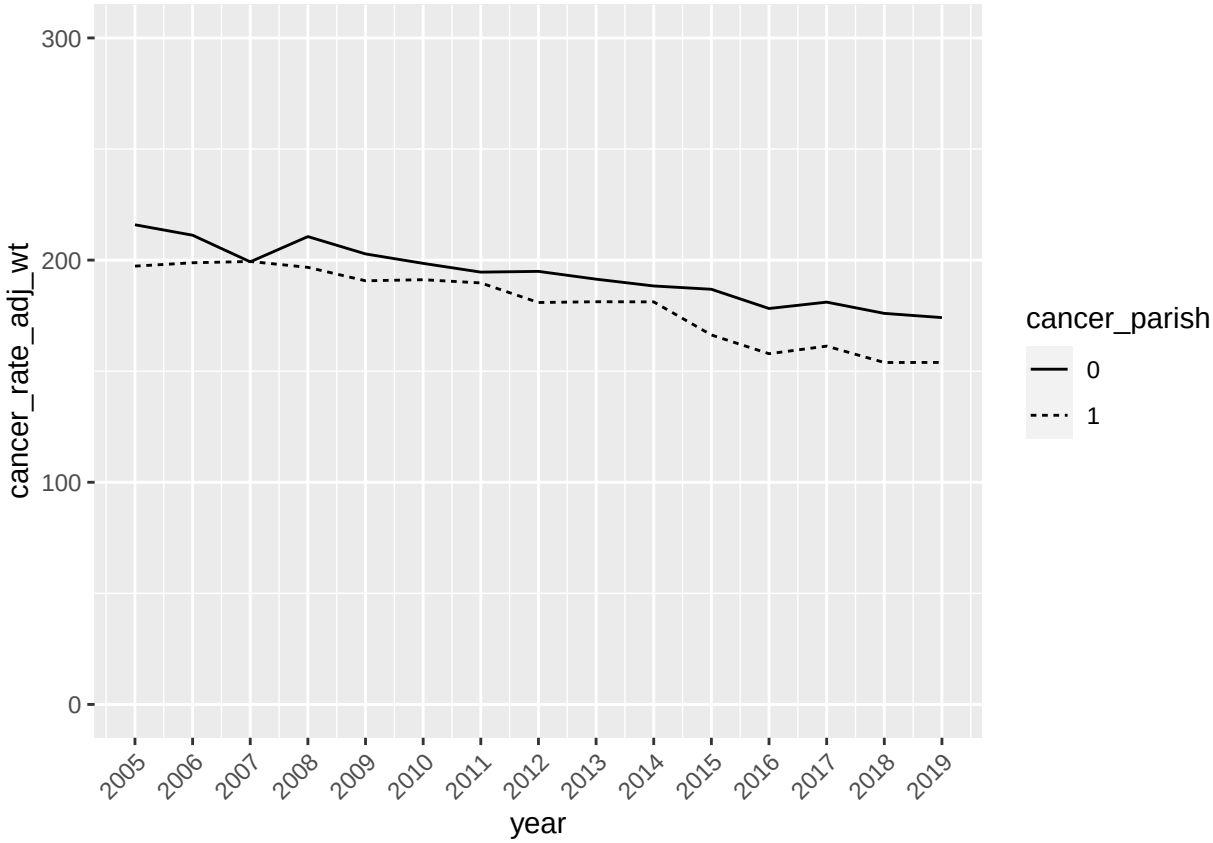
5: Now we will adjust the scale to change the y-axis values

```
cancer_alley_rates$cancer_parish <- factor(cancer_alley_rates$cancer_parish)
ggplot(data = cancer_alley_rates,
       mapping = aes(x = year, y = cancer_rate_adj_wt, group = cancer_parish)) +
         geom_line(aes(linetype= cancer_parish)) +
  scale_y_continuous(limits = c(0, 300))
```



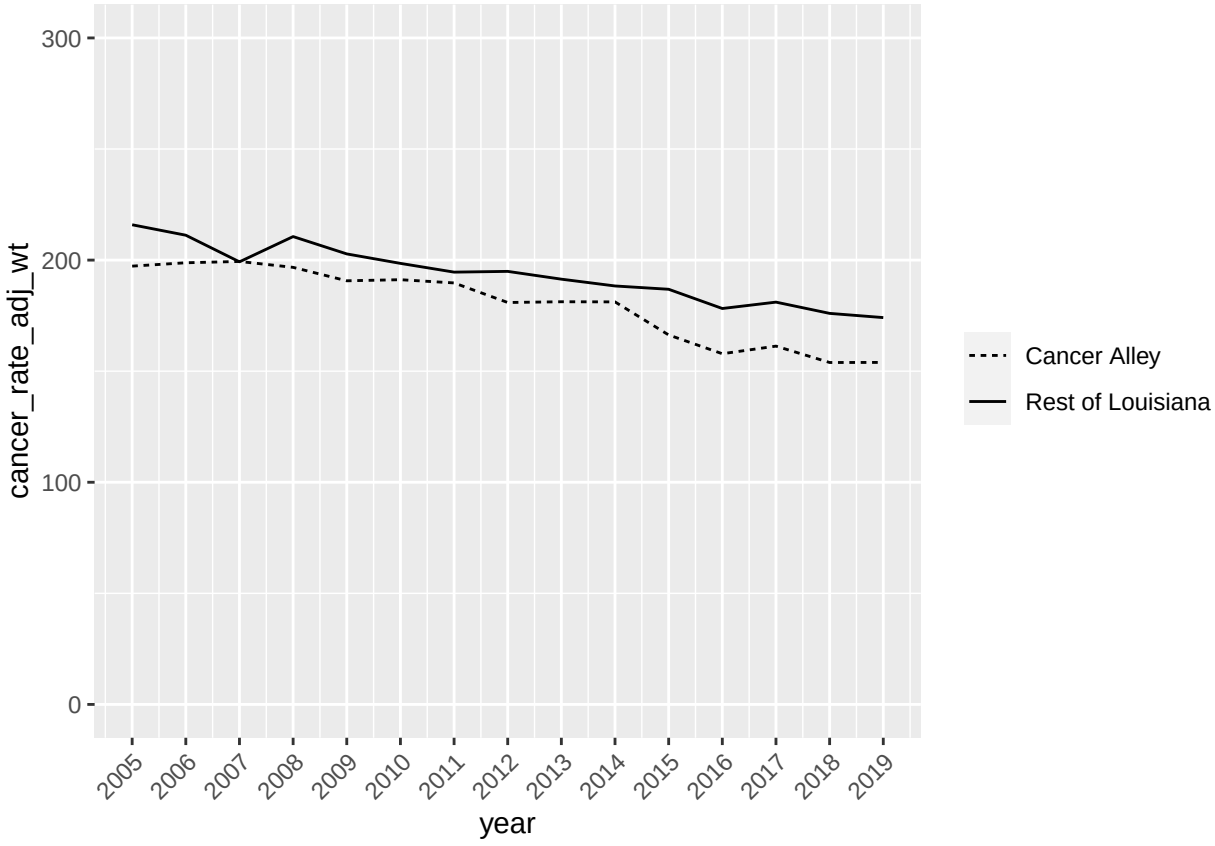
6: Now we will adjust the x-axis for values that fit our data

```
cancer_alley_rates$cancer_parish <- factor(cancer_alley_rates$cancer_parish)
ggplot(data = cancer_alley_rates,
       mapping = aes(x = year, y = cancer_rate_adj_wt, group = cancer_parish)) +
         geom_line(aes(linetype= cancer_parish))+
  scale_y_continuous(limits = c(0, 300)) +
  scale_x_continuous(breaks = seq(2005, 2019, by = 1)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



7: Now we will change the legend to display which is cancer alley and which is not

```
cancer_alley_rates$cancer_parish <- factor(cancer_alley_rates$cancer_parish)
ggplot(data = cancer_alley_rates,
       mapping = aes(x = year, y = cancer_rate_adj_wt, group = cancer_parish)) +
         geom_line(aes(linetype= cancer_parish))+
  scale_y_continuous(limits = c(0, 300)) +
  scale_x_continuous(breaks = seq(2005, 2019, by = 1)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_linetype_discrete(name = NULL, labels = c("Rest of Louisiana","Cancer Alley"), guide = guide_leg
```

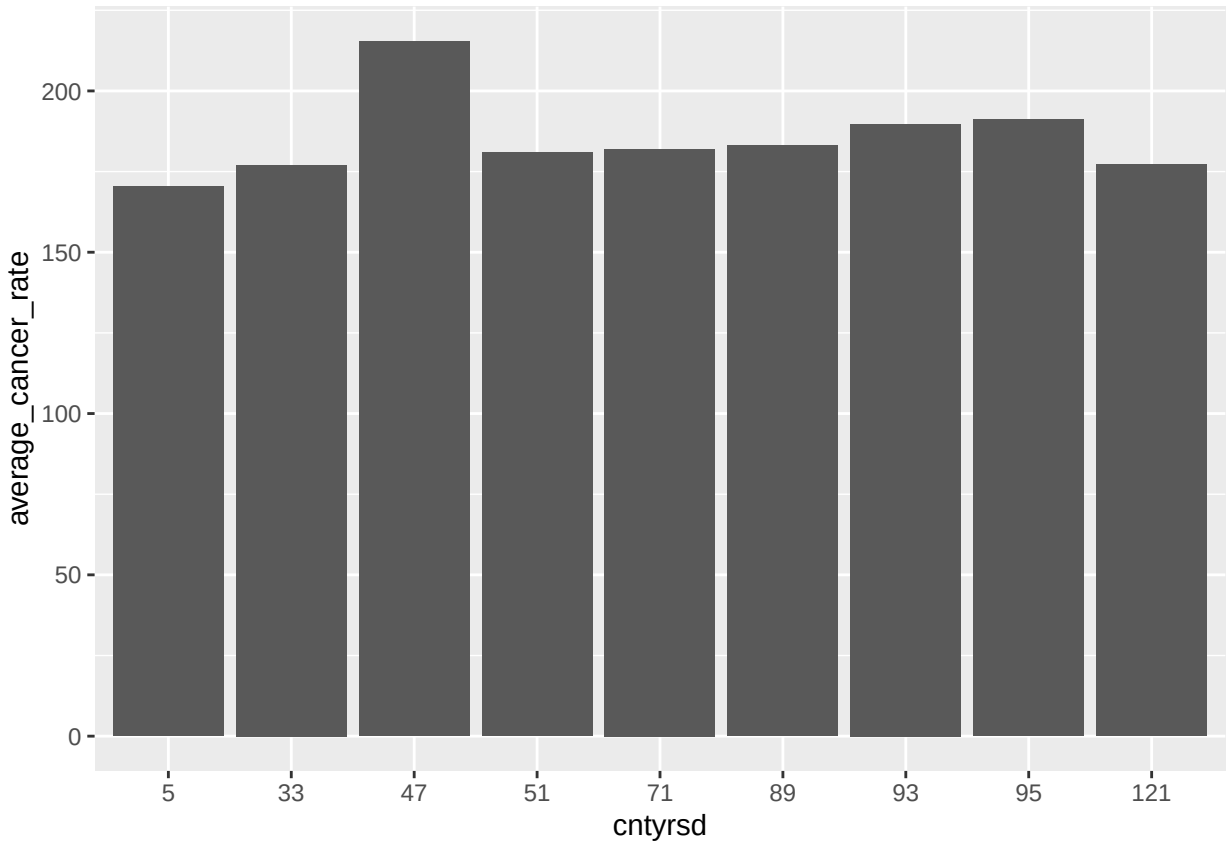


8: Now we will add a title to our plot

```
cancer_alley_rates$cancer_parish <- factor(cancer_alley_rates$cancer_parish)
ggplot(data = cancer_alley_rates,
       mapping = aes(x = year, y = cancer_rate_adj_wt, group = cancer_parish)) +
         geom_line(aes(linetype= cancer_parish))+
  scale_y_continuous(limits = c(0, 300)) +
  scale_x_continuous(breaks = seq(2005, 2019, by = 1)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_linetype_discrete(name = NULL, labels = c("Rest of Louisiana","Cancer Alley"), guide = guide_leg
  labs(title = "Cancer Mortality Rate Comparison", y = "Cancer Deaths per 100,000 Residents", x = NULL)
```

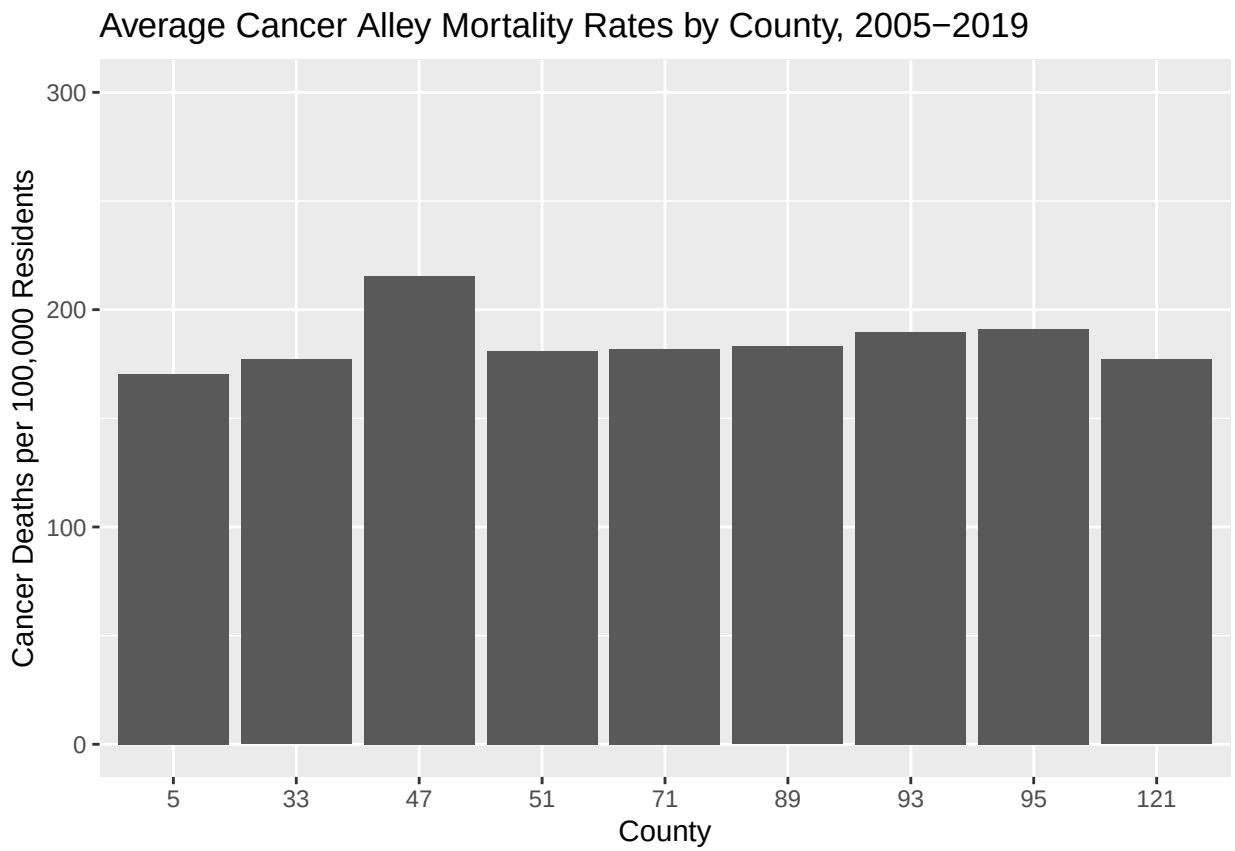
9: Now we will make a bar chart to look at just the differences in Cancer Alley parishes in terms of mortality rates

```
parish_rates$cntyrsd <- as.factor(parish_rates$cntyrsd)
parish_rates_cancer_alley <- subset(parish_rates, cancer_parish == 1)
averaged_rates <- parish_rates_cancer_alley %>%
  group_by(cntyrsd) %>%
  summarize(average_cancer_rate=mean(cancer_rate_adj, na.rm = TRUE))

ggplot(data = averaged_rates, mapping = aes(x = cntyrsd, y = average_cancer_rate)) +
  geom_col()
```



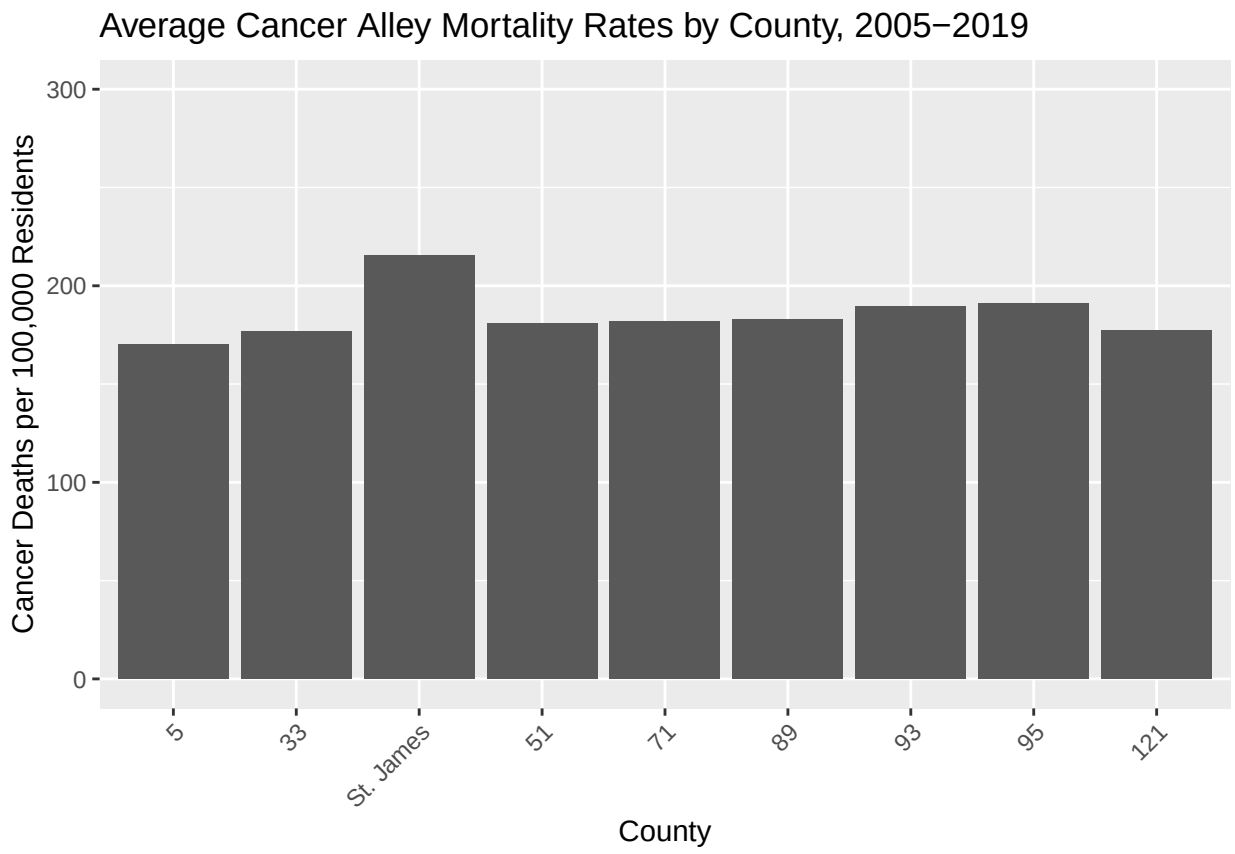
10: Change the y-scale to fit our data better and add title, x and y variable titles

```
ggplot(data = averaged_rates, mapping = aes(x = cntyrsd, y = average_cancer_rate)) +
  geom_col() +
  scale_y_continuous(limits = c(0, 300)) +
                       labs(title = "Average Cancer Alley Mortality Rates by County, 2005-2019", y = "C
```



11: Now we will change the x-axis to the county names instead of the FIPS code

```
ggplot(data = averaged_rates, mapping = aes(x = cntyrsd, y = average_cancer_rate)) +
  geom_col() +
  scale_y_continuous(limits = c(0, 300)) +
                       labs(title = "Average Cancer Alley Mortality Rates by County, 2005-2019", y = "Ca
  scale_x_discrete(labels = c("3" = "Acension",
                              "17" = "East Baton Rouge",
                              "24" = "Iberville",
                              "26" = "Jefferson",
                              "36" = "Orleans",
                              "45" = "St. Charles",
                              "47" = "St. James",
                              "48" = "St. John the Baptist",
                              "61" = "West Baton Rouge")) +
theme(axis.text.x = element_text(angle = 45, hjust = 1))
```


Average Cancer Alley Mortality Rates by County, 2005−2019
Cancer Deaths per 100,000 Residents
300
200
100
0
5
33
St. James
51
71
89
93
95
121
County