Report EDA

Christian Pascual

```r
library(tidyverse)
set.seed(1)

# Load the data and do some processing
cancer = read_csv("cancer_registry.csv") %>%
  # Split the geography variable
  separate(Geography, into = c("county", "state"), sep = ", ") %>%
  # Split up binnedInc into a lower and upper decile
  mutate(
    binnedInc = str_remove_all(binnedInc, "[(\\]]"),
    # also try to group states by region
    region = case_when(
      state %in% c("California", "Oregon", "Washington", "Nevada", "Idaho",
                   "Montana", "Wyoming", "Colorado", "Utah", "Alaska", "Hawaii") ~ "West",
      state %in% c("Arizona", "New Mexico", "Texas", "Oklahoma") ~ "Southwest",
      state %in% c("North Dakota", "South Dakota", "Nebraska", "Kansas",
                   "Minnesota", "Iowa", "Missouri", "Wisconsin", "Illinois",
                   "Indiana", "Ohio", "Michigan") ~ "Midwest",
      state %in% c("Arkansas", "Louisiana", "Mississippi", "Alabama", "Georgia",
                   "Florida", "South Carolina", "North Carolina", "Tennessee",
                   "Kentucky", "Virginia", "West Virginia", "District of Columbia",
                   "Delaware") ~ "Southeast",
      state %in% c("Maryland", "Pennsylvania", "New Jersey", "New York", "Rhode Island",
                   "Connecticut", "Massachusetts", "New Hampshire", "Vermont", "Maine") ~ "Northeast",
      TRUE ~ "Southwest" # Weird formatting means a single NM is NA in state
    )
  ) %>%
  separate(binnedInc, into = c("inc_dec_low", "inc_dec_high"), sep = ",") %>%
  janitor::clean_names() %>%  # Convert all column names to lowercase'
  mutate(
    high_college = pct_bach_deg25_over > median(pct_bach_deg25_over), # median(pct_bach_deg18_24),
    high_hs = pct_hs18_24 > median(pct_hs18_24)
  )
```

Missing Data

```r
# Get the number of rows with missing data for each column
cancer %>%
  select(everything()) %>%
  summarise_all(funs(sum(is.na(.)))) %>%
  select(which(colMeans(.) != 0))
```

```
## Warning: ‘funs()‘ is deprecated as of dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with ‘tibble::lst()‘:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_warnings()‘ to see where this warning was generated.
```

```
## # A tibble: 1 x 4
##   state pct_some_col18_24 pct_employed16_over pct_private_coverage_alone
##   <int>             <int>               <int>                      <int>
## 1     1              2285                 152                        609
```

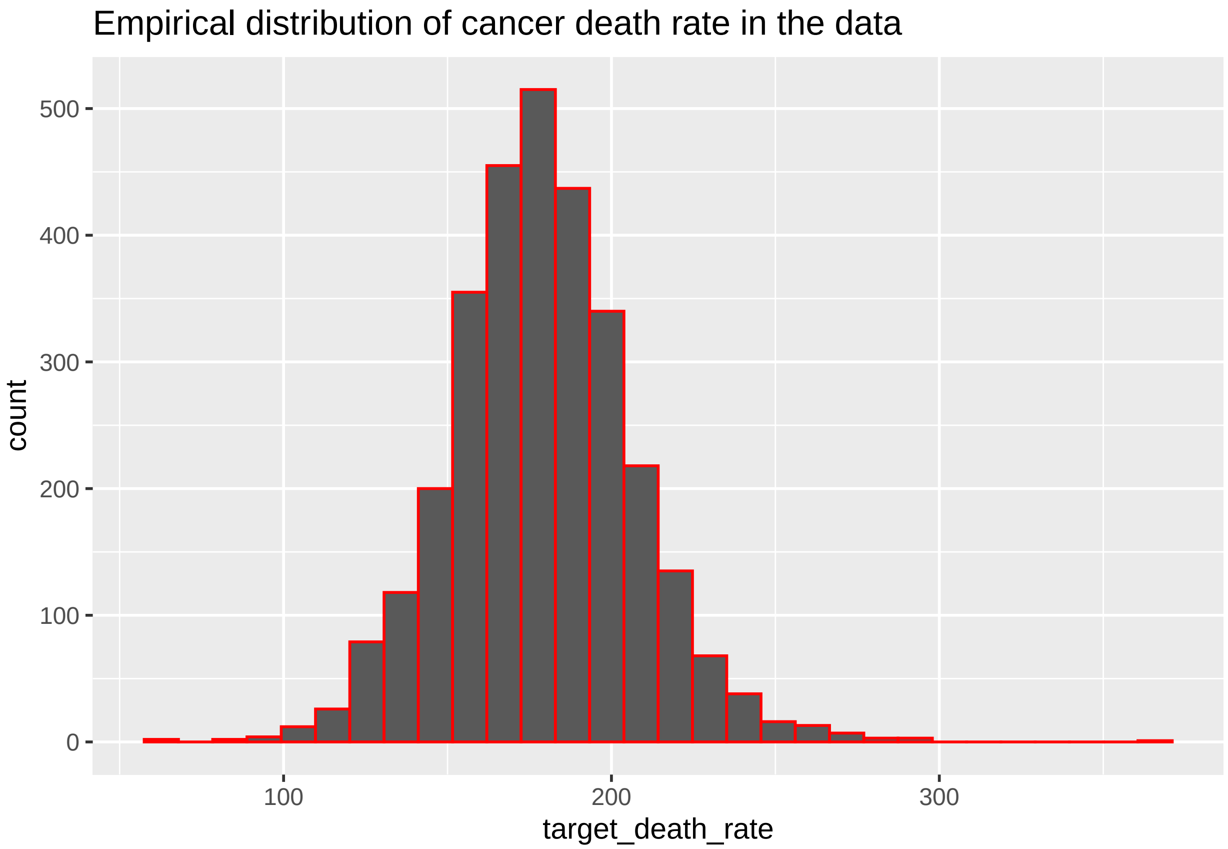
Three columns have missing data:

- pct_some_col18_24
- pct_employed16_over
- pct_private_coverage_alone

Distribution of outcome

```
cancer %>%
  ggplot(aes(x = target_death_rate)) +
  geom_histogram(color = "red") +
  ggtitle("Empirical distribution of cancer death rate in the data")
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

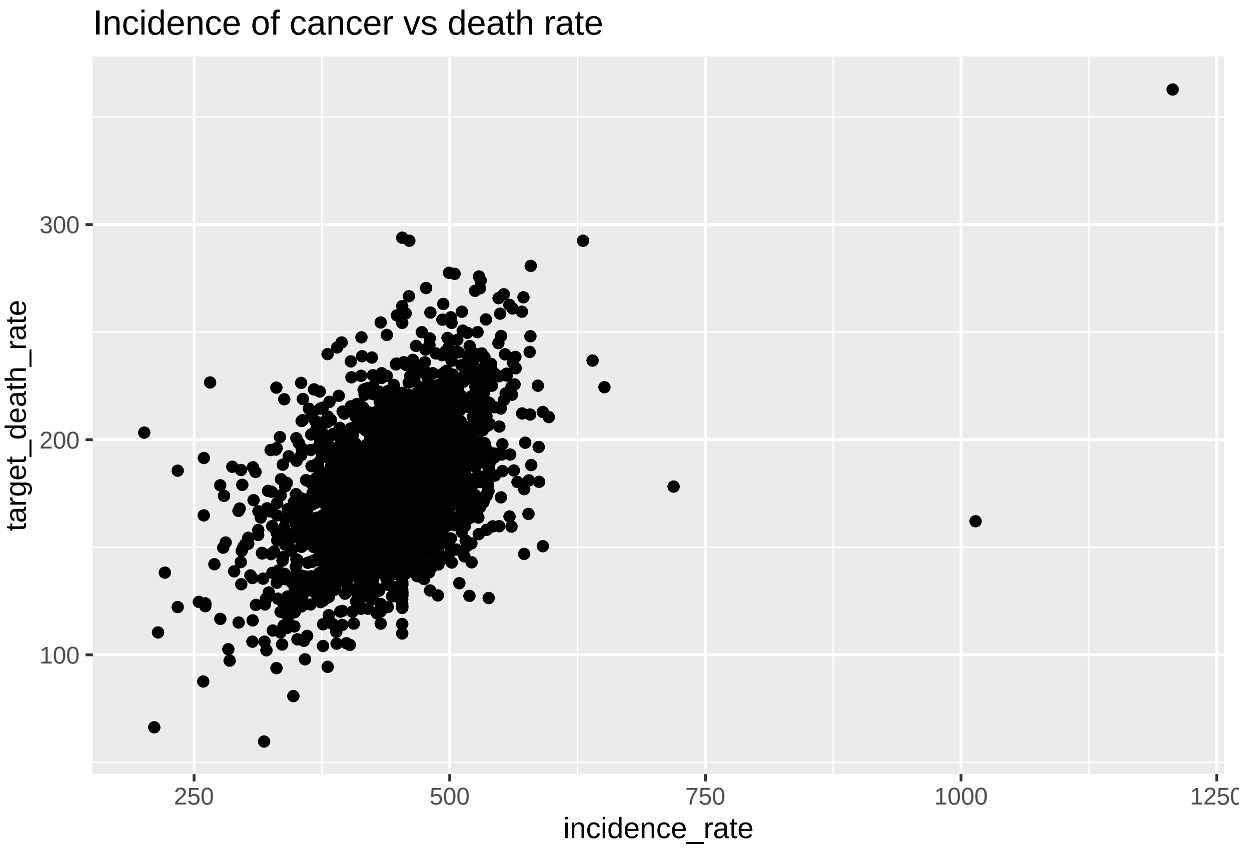


Outcome is reasonably normally distributed, which is nice.

Variable Trends

Incidence and cancer death rate

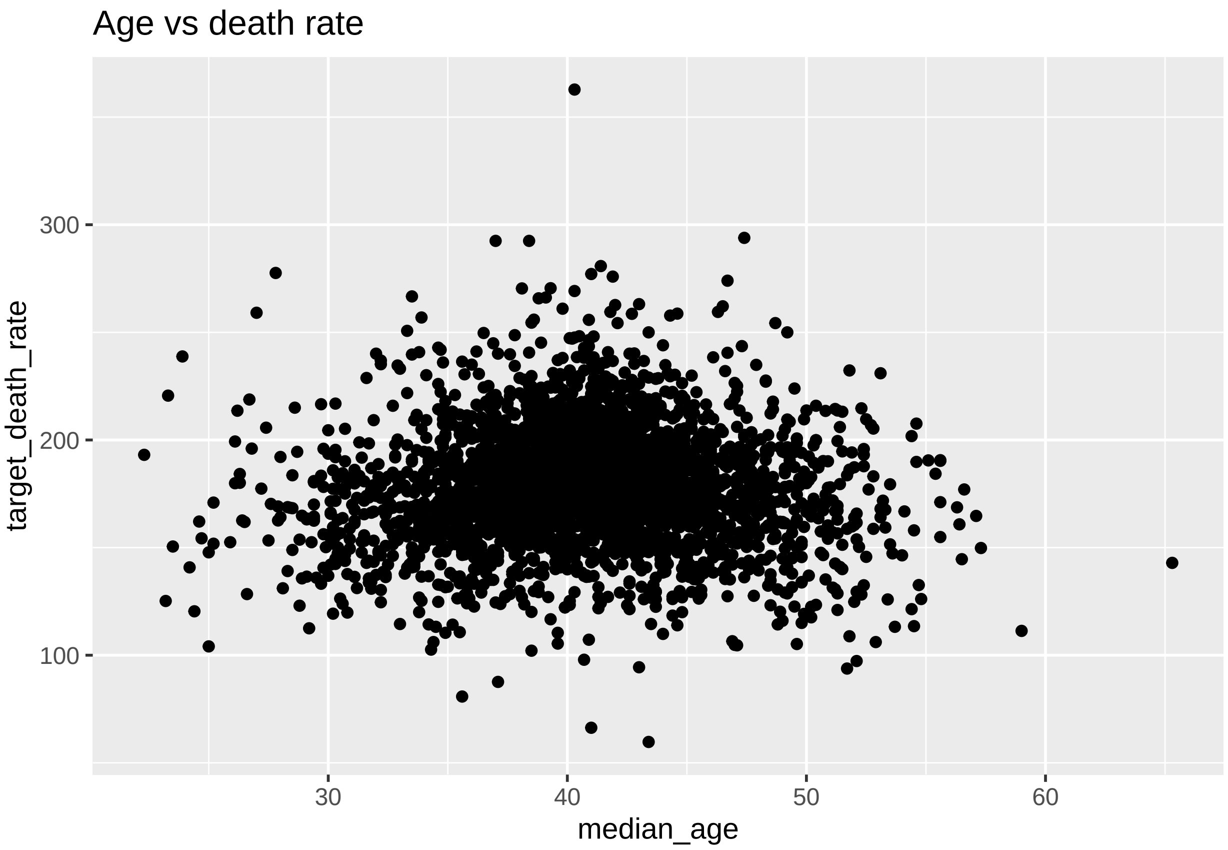
```
cancer %>%
  ggplot(aes(x = incidence_rate, y = target_death_rate)) +
  geom_point() +
  ggtitle("Incidence of cancer vs death rate")
```



Suggests a positive relationship between higher incidence and higher death rate. This is expected, so this should be controlled for in the model.

Age and death rate

```r
cancer %>%
  filter(median_age < 100) %>% # some weird values in the data
  ggplot(aes(x = median_age, y = target_death_rate)) +
  geom_point() +
  ggtitle("Age vs death rate")
```

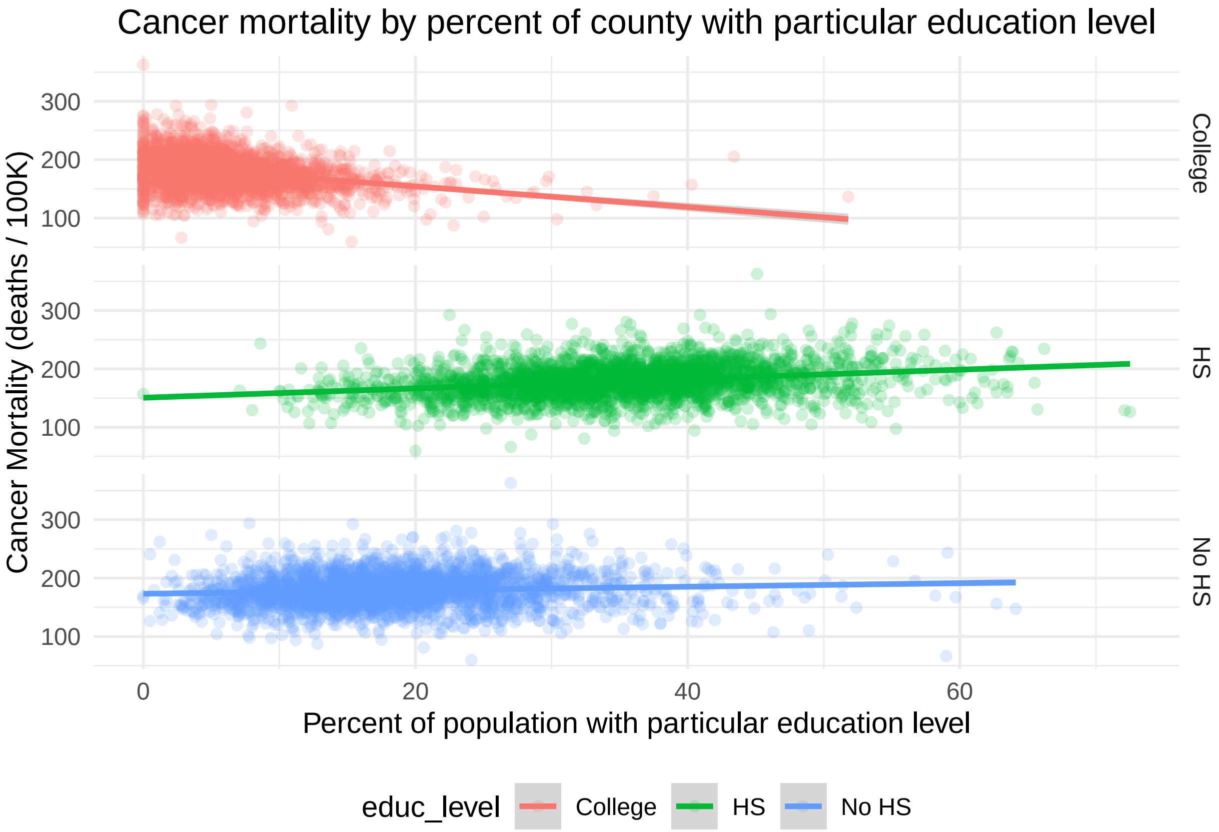


I thought there might be a relationship between age and death rate. Doesn't seem to show here. I know that the older you get, the more likely you are to get cancer, so this should still be controlled for in the analysis.

Education & Death Rate

```
cancer %>%
  pivot_longer(
    cols = c("pct_no_hs18_24", "pct_hs18_24", "pct_bach_deg18_24"),
    values_to = "pct",
    names_to = "educ_level"
  ) %>%
  mutate(
    educ_level = case_when(
      educ_level == "pct_no_hs18_24" ~ "No HS",
      educ_level == "pct_hs18_24" ~ "HS",
      educ_level == "pct_bach_deg18_24" ~ "College"
    )
  ) %>%
  ggplot(aes(x = pct, y = target_death_rate, color = educ_level)) +
  facet_grid(educ_level ~ .) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    plot.title = element_text(hjust = 0.5)) +
  geom_point(alpha = 0.2) +
  geom_smooth(method = "lm") +
  labs(
    title = "Cancer mortality by percent of county with particular education level",
    x = "Percent of population with particular education level",
    y = "Cancer Mortality (deaths / 100K)")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```

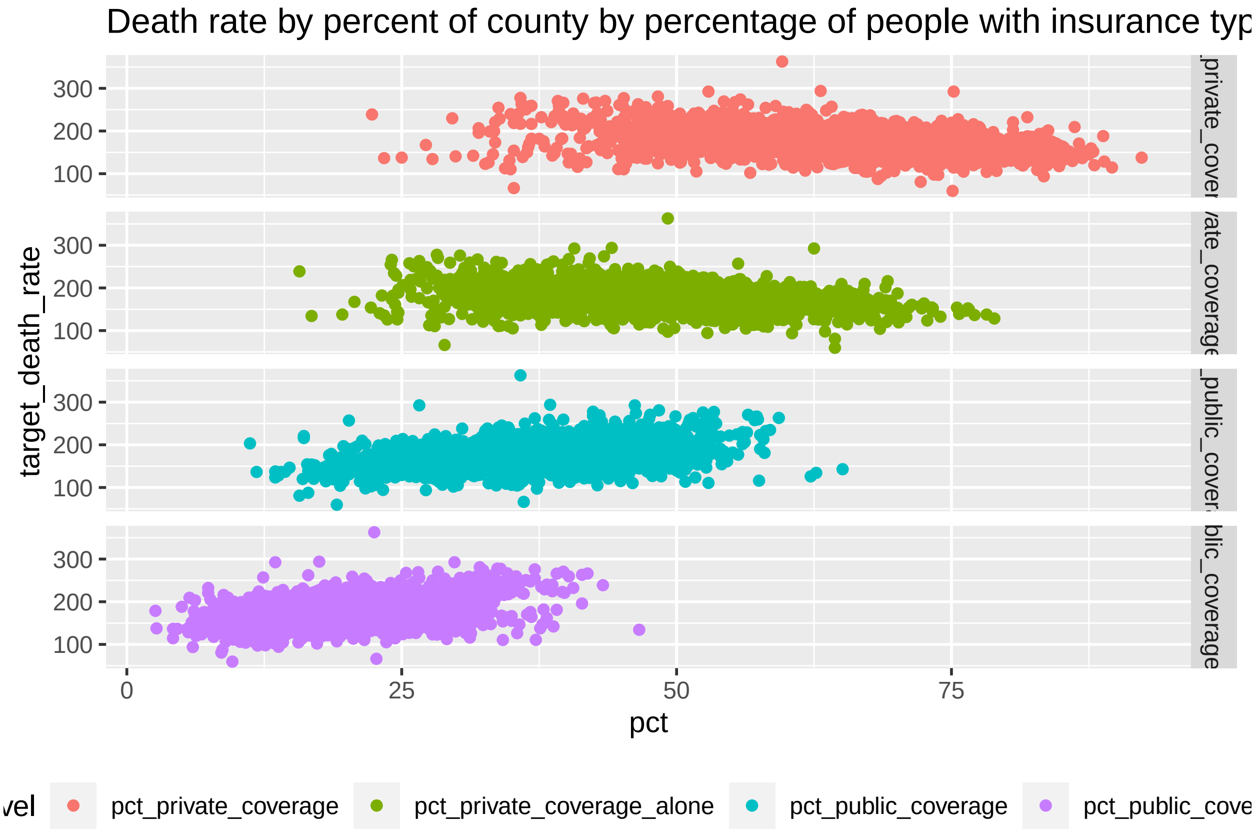


Hard to parse much here. It kind of looks like more education leads to slightly lower death rates (based on blue plot)? But this effect is not very pronounced.

Insurance Coverage & Death Rate

```
cancer %>%
  pivot_longer(
    cols = c("pct_private_coverage",
             "pct_private_coverage_alone",
             "pct_public_coverage",
             "pct_public_coverage_alone"),
    values_to = "pct",
    names_to = "ins_level"
  ) %>%
  ggplot(aes(x = pct, y = target_death_rate, color = ins_level)) +
  facet_grid(ins_level ~ .) +
  theme(legend.position = "bottom") +
  geom_point() +
  ggtitle("Death rate by percent of county by percentage of people with insurance types")
```

```
## Warning: Removed 609 rows containing missing values (geom_point).
```

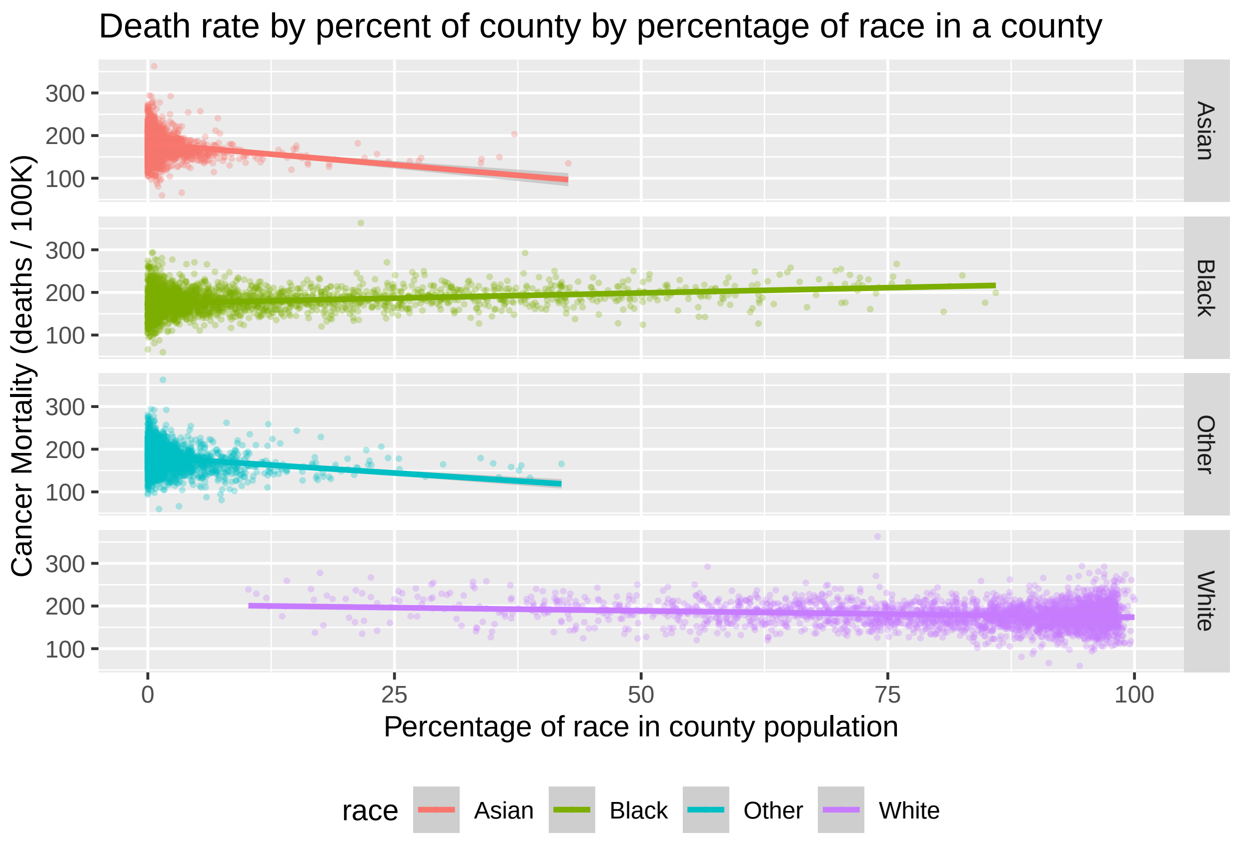


Interesting shift. The higher private coverage is, the lower the death rate. It's converse for public coverage. Perhaps this is a proxy for other economic factors? Maybe private insurance is better than public coverage? My thought was that more coverage in general would help improve death rate.

Race & Death Rate

```
cancer %>%
  pivot_longer(
    cols = c("pct_white",
             "pct_black",
             "pct_asian",
             "pct_other_race"),
    values_to = "pct",
    names_to = "race"
  ) %>%
  mutate(
    race = case_when(
      race == "pct_white" ~ "White",
      race == "pct_black" ~ "Black",
      race == "pct_asian" ~ "Asian",
      race == "pct_other_race" ~ "Other"
    )
  ) %>%
  ggplot(aes(x = pct, y = target_death_rate, color = race)) +
  geom_point(size = 0.5, alpha = 0.3) +
  geom_smooth(method = "lm") +
  facet_grid(race ~ .) +
  theme(legend.position = "bottom") +
  labs(
    title = "Death rate by percent of county by percentage of race in a county",
    x = "Percentage of race in county population",
    y = "Cancer Mortality (deaths / 100K)")
```

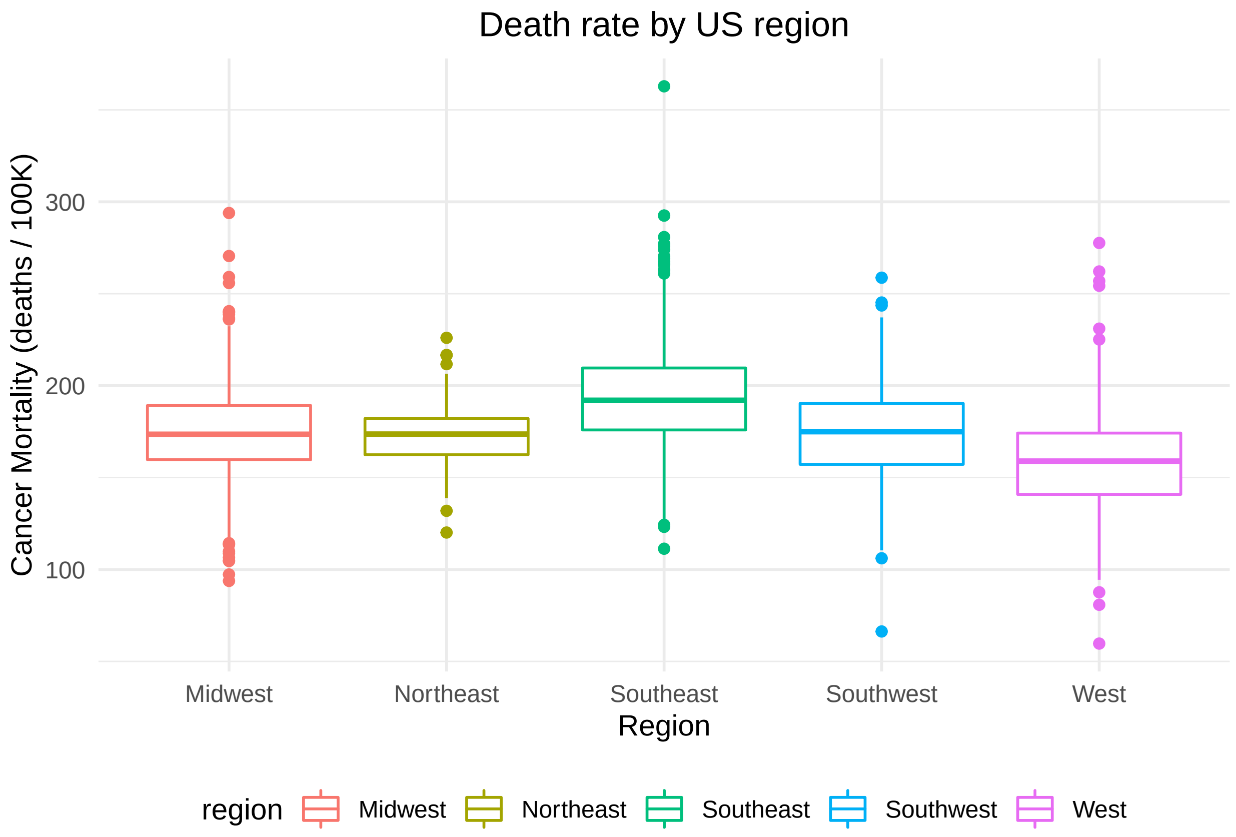
```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



Main takeaway here is that a higher percentage of African-Americans corresponds to higher death rates. This lines up with evidence of racial disparities in the healthcare system with regard to access and cost.

Region by Death Rate

```
cancer %>%
  ggplot(aes(x = region, y = target_death_rate, color = region)) +
  geom_boxplot() +
  theme_minimal() +
  labs(
    title = "Death rate by US region",
    x = "Region",
    y = "Cancer Mortality (deaths / 100K)") +
  theme(
    legend.position = "bottom",
    plot.title = element_text(hjust = 0.5)
  )
```

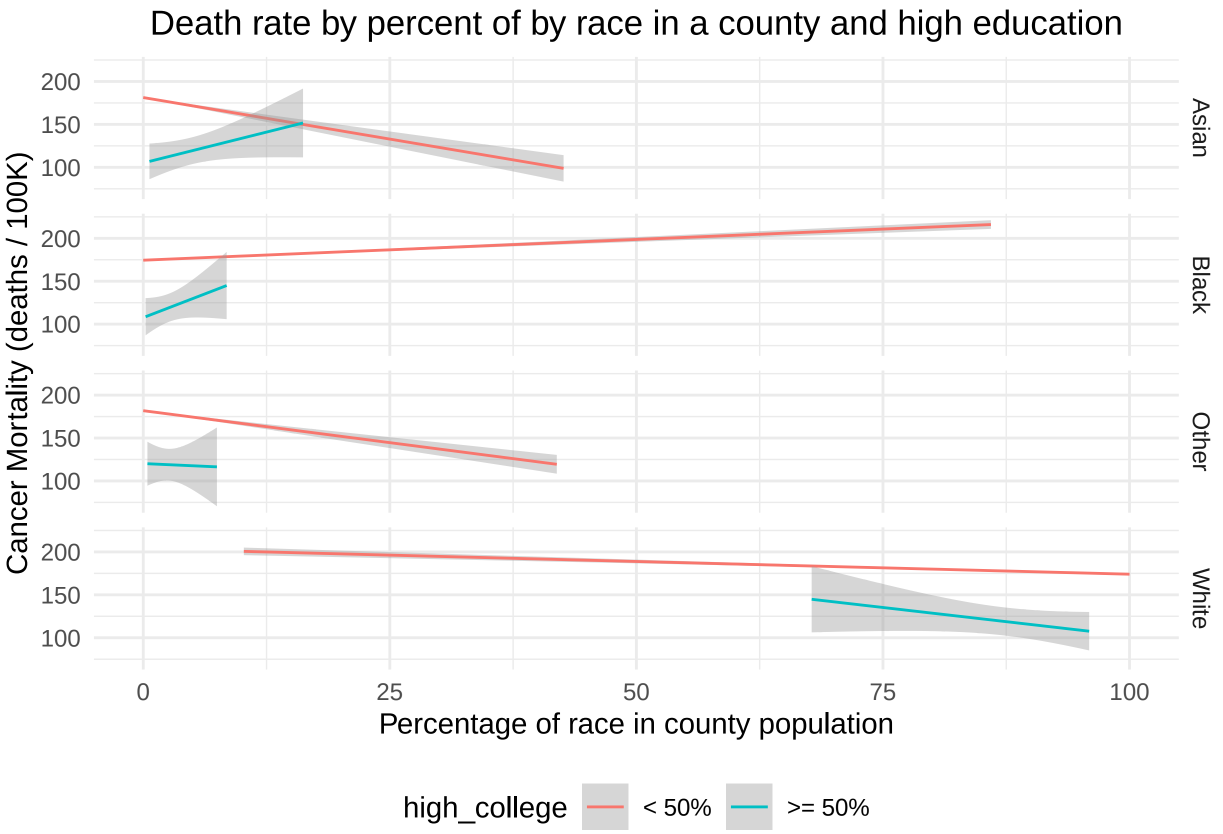


Interesting… Southeast seems to have a higher death rate than the other regions.

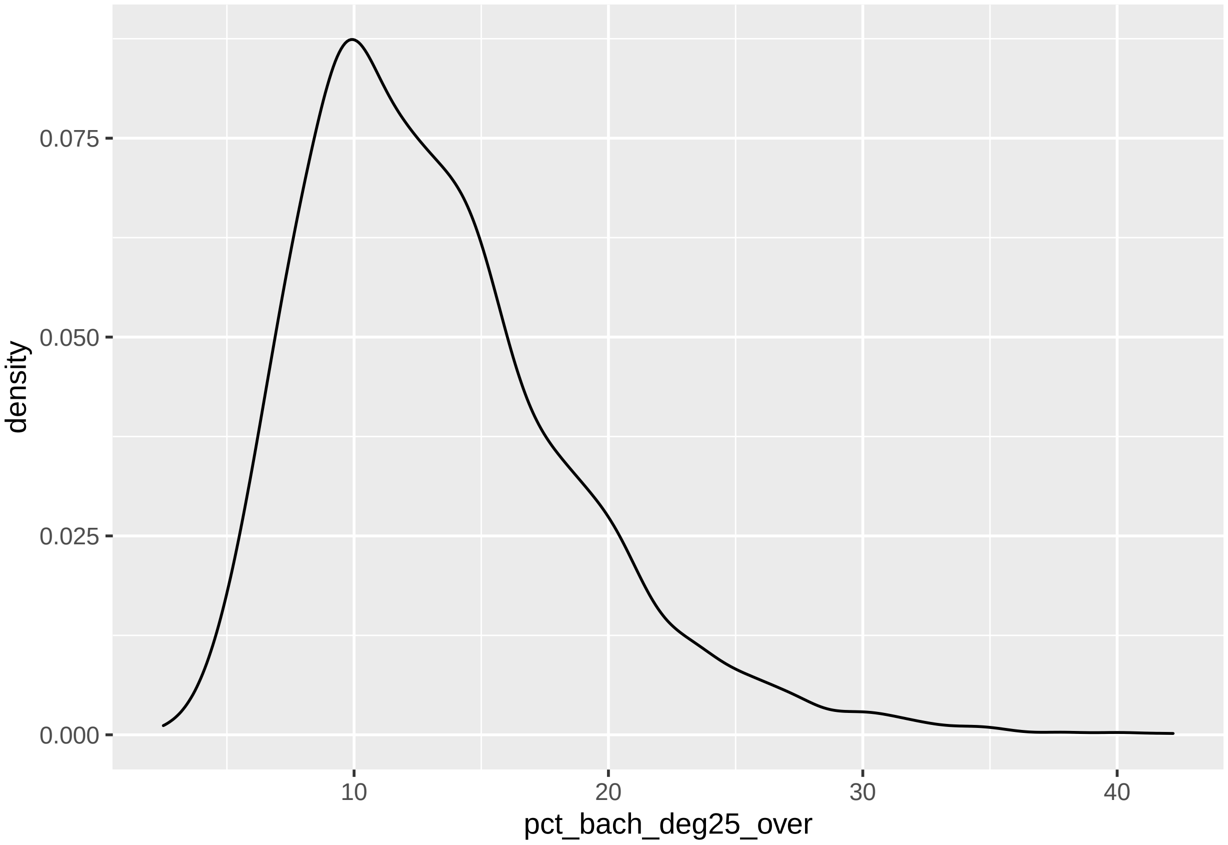
Interaction Between Education and Race?

```
cancer %>%
  mutate(
    high_college = pct_bach_deg25_over > 34, # median(pct_bach_deg18_24),
    high_hs = pct_hs18_24 > median(pct_hs18_24)
  ) %>%
  pivot_longer(
    cols = c("pct_white",
             "pct_black",
             "pct_asian",
             "pct_other_race"),
    values_to = "pct",
    names_to = "race"
  ) %>%
  mutate(
    race = case_when(
      race == "pct_white" ~ "White",
      race == "pct_black" ~ "Black",
      race == "pct_asian" ~ "Asian",
      race == "pct_other_race" ~ "Other"
    )
  ) %>%
  ggplot(aes(x = pct, y = target_death_rate, color = high_college)) +
  geom_smooth(method = "lm", size = 0.5) +
  facet_grid(race ~ .) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    plot.title = element_text(hjust = 0.5)
  ) +
  labs(
    title = "Death rate by percent of by race in a county and high education",
    x = "Percentage of race in county population",
    y = "Cancer Mortality (deaths / 100K)") +
  scale_color_discrete(labels = c("< 50%", ">= 50%"))
```

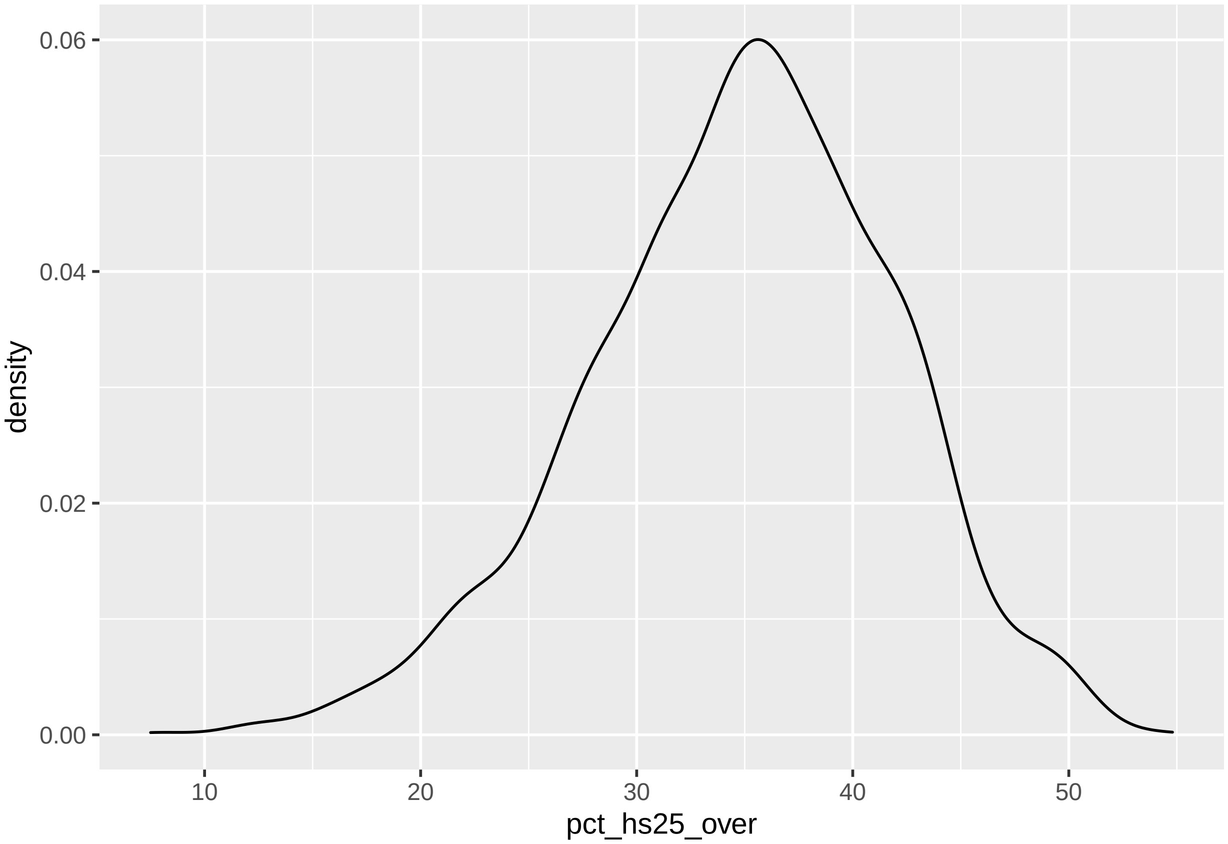
```
## `geom_smooth()` using formula 'y ~ x'
```



```
cancer %>%
  ggplot(aes(x = pct_bach_deg25_over)) +
  geom_density()
```



```r
cancer %>%
  ggplot(aes(x = pct_hs25_over)) +
  geom_density()
```



Inference Model

```
colnames(cancer)
```

```
##  [1] "avg_ann_count"              "avg_deaths_per_year"
##  [3] "target_death_rate"          "incidence_rate"
##  [5] "med_income"                 "pop_est2015"
##  [7] "poverty_percent"            "study_per_cap"
##  [9] "inc_dec_low"                "inc_dec_high"
## [11] "median_age"                 "median_age_male"
## [13] "median_age_female"          "county"
## [15] "state"                      "avg_household_size"
## [17] "percent_married"            "pct_no_hs18_24"
## [19] "pct_hs18_24"                "pct_some_col18_24"
## [21] "pct_bach_deg18_24"          "pct_hs25_over"
## [23] "pct_bach_deg25_over"        "pct_employed16_over"
## [25] "pct_unemployed16_over"      "pct_private_coverage"
## [27] "pct_private_coverage_alone" "pct_emp_priv_coverage"
## [29] "pct_public_coverage"        "pct_public_coverage_alone"
## [31] "pct_white"                  "pct_black"
## [33] "pct_asian"                  "pct_other_race"
## [35] "pct_married_households"     "birth_rate"
## [37] "region"                     "high_college"
## [39] "high_hs"
```

```
# Trying out fuill interaction
model = lm(target_death_rate ~ pct_no_hs18_24 + pct_no_hs18_24 + pct_some_col18_24 +
                               pct_bach_deg18_24 + pct_white + pct_black + pct_asian + pct_other_race +
                               pct_no_hs18_24*pct_white + pct_no_hs18_24*pct_black + pct_no_hs18_24*pct_
                               pct_no_hs18_24*pct_other_race + pct_hs18_24*pct_white + pct_hs18_24*pct_b
                               pct_hs18_24*pct_other_race + pct_some_col18_24*pct_white + pct_some_col18
                               pct_some_col18_24*pct_other_race + pct_bach_deg18_24*pct_white + pct_bach
                               pct_bach_deg18_24*pct_asian + pct_bach_deg18_24*pct_other_race +
                               incidence_rate + med_income + pop_est2015 + poverty_percent + median_age
                             data = cancer)

# Doesn't seem very good, will try indicators for high education and low education
summary(model)
```

```
##
## Call:
## lm(formula = target_death_rate ~ pct_no_hs18_24 + pct_no_hs18_24 +
##     pct_some_col18_24 + pct_bach_deg18_24 + pct_white + pct_black +
##     pct_asian + pct_other_race + pct_no_hs18_24 * pct_white +
##     pct_no_hs18_24 * pct_black + pct_no_hs18_24 * pct_asian +
##     pct_no_hs18_24 * pct_other_race + pct_hs18_24 * pct_white +
##     pct_hs18_24 * pct_black + pct_hs18_24 * pct_asian + pct_hs18_24 *
##     pct_other_race + pct_some_col18_24 * pct_white + pct_some_col18_24 *
##     pct_black + pct_some_col18_24 * pct_asian + pct_some_col18_24 *
##     pct_other_race + pct_bach_deg18_24 * pct_white + pct_bach_deg18_24 *
##     pct_black + pct_bach_deg18_24 * pct_asian + pct_bach_deg18_24 *
##     pct_other_race + incidence_rate + med_income + pop_est2015 +
##     poverty_percent + median_age, data = cancer)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -113.989  -10.849    0.195   11.260  103.229
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      2.039e+04  2.135e+04   0.955 0.339900
## pct_no_hs18_24                  -2.037e+02  2.131e+02  -0.956 0.339427
## pct_some_col18_24               -2.015e+02  2.137e+02  -0.943 0.345972
## pct_bach_deg18_24               -2.127e+02  2.129e+02  -0.999 0.318205
## pct_white                       -1.961e+02  2.205e+02  -0.889 0.374184
## pct_black                       -1.736e+02  2.356e+02  -0.737 0.461307
## pct_asian                       -2.949e+02  8.299e+02  -0.355 0.722385
## pct_other_race                  -1.866e+01  4.660e+02  -0.040 0.968076
## pct_hs18_24                     -2.030e+02  2.135e+02  -0.951 0.342084
## incidence_rate                   2.108e-01  1.484e-02  14.206  < 2e-16 ***
## med_income                      -3.960e-04  1.401e-04  -2.827 0.004834 **
## pop_est2015                      3.255e-07  2.218e-06   0.147 0.883357
## poverty_percent                  9.205e-01  2.760e-01   3.335 0.000896 ***
## median_age                      -1.702e-02  1.680e-02  -1.013 0.311433
## pct_no_hs18_24:pct_white         1.968e+00  2.201e+00   0.894 0.371530
## pct_no_hs18_24:pct_black         1.738e+00  2.352e+00   0.739 0.460077
## pct_no_hs18_24:pct_asian         2.850e+00  8.282e+00   0.344 0.730877
## pct_no_hs18_24:pct_other_race    2.092e-01  4.661e+00   0.045 0.964216
## pct_white:pct_hs18_24             1.964e+00  2.205e+00   0.890 0.373499
## pct_black:pct_hs18_24             1.740e+00  2.356e+00   0.738 0.460496
## pct_asian:pct_hs18_24             3.062e+00  8.307e+00   0.369 0.712511
## pct_other_race:pct_hs18_24        1.555e-01  4.661e+00   0.033 0.973393
## pct_some_col18_24:pct_white       1.945e+00  2.207e+00   0.881 0.378492
## pct_some_col18_24:pct_black       1.720e+00  2.357e+00   0.730 0.465827
## pct_some_col18_24:pct_asian       2.875e+00  8.301e+00   0.346 0.729199
## pct_some_col18_24:pct_other_race  1.613e-01  4.661e+00   0.035 0.972406
## pct_bach_deg18_24:pct_white       2.047e+00  2.200e+00   0.930 0.352432
## pct_bach_deg18_24:pct_black       1.877e+00  2.352e+00   0.798 0.425133
## pct_bach_deg18_24:pct_asian       3.106e+00  8.291e+00   0.375 0.708072
## pct_bach_deg18_24:pct_other_race  3.124e-01  4.642e+00   0.067 0.946359
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.9 on 732 degrees of freedom
##   (2285 observations deleted due to missingness)
## Multiple R-squared:  0.4594, Adjusted R-squared:  0.438
## F-statistic: 21.45 on 29 and 732 DF,  p-value: < 2.2e-16
```

```r
# Trying out different set of education vars
model2 = lm(target_death_rate ~ pct_hs25_over + pct_bach_deg25_over +
             pct_white + pct_black + pct_asian + pct_other_race +
             # Interactions
             pct_white*pct_hs25_over + pct_black*pct_hs25_over + pct_asian*pct_hs25_over + pct_other_rac
             pct_white*pct_bach_deg25_over + pct_black*pct_bach_deg25_over + pct_asian*pct_bach_deg25_ov
             # Confounders
             incidence_rate + med_income + pop_est2015 + poverty_percent + median_age,
           data = cancer)

# Looks good! Some interesting results here.
summary(model2)
```

```
##
## Call:
## lm(formula = target_death_rate ~ pct_hs25_over + pct_bach_deg25_over +
##     pct_white + pct_black + pct_asian + pct_other_race + pct_white *
##     pct_hs25_over + pct_black * pct_hs25_over + pct_asian * pct_hs25_over +
##     pct_other_race * pct_hs25_over + pct_white * pct_bach_deg25_over +
##     pct_black * pct_bach_deg25_over + pct_asian * pct_bach_deg25_over +
##     pct_other_race * pct_bach_deg25_over + incidence_rate + med_income +
##     pop_est2015 + poverty_percent + median_age, data = cancer)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -110.842  -11.397    0.188   11.385  116.656
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        -3.827e+01  4.561e+01  -0.839  0.40151
## pct_hs25_over                       2.870e+00  9.038e-01   3.175  0.00151 **
## pct_bach_deg25_over                 2.066e+00  1.650e+00   1.253  0.21047
## pct_white                           1.164e+00  4.627e-01   2.515  0.01194 *
## pct_black                           1.061e+00  5.204e-01   2.039  0.04151 *
## pct_asian                          -5.888e-01  1.984e+00  -0.297  0.76662    
## pct_other_race                     -1.507e+00  1.021e+00  -1.477  0.13978    
## incidence_rate                      1.985e-01  6.897e-03  28.784  < 2e-16 ***
## med_income                          1.238e-04  6.758e-05   1.832  0.06709 .  
## pop_est2015                        -1.708e-06  1.353e-06  -1.263  0.20684    
## poverty_percent                     1.198e+00  1.261e-01   9.504  < 2e-16 ***
## median_age                         -2.121e-03  7.921e-03  -0.268  0.78891    
## pct_hs25_over:pct_white            -2.554e-02  9.325e-03  -2.739  0.00620 ** 
## pct_hs25_over:pct_black            -3.198e-02  1.065e-02  -3.002  0.00270 ** 
## pct_hs25_over:pct_asian             1.715e-02  4.162e-02   0.412  0.68025    
## pct_hs25_over:pct_other_race        1.387e-02  2.396e-02   0.579  0.56279    
## pct_bach_deg25_over:pct_white      -4.068e-02  1.700e-02  -2.393  0.01675 *  
## pct_bach_deg25_over:pct_black      -2.232e-03  1.813e-02  -0.123  0.90204    
## pct_bach_deg25_over:pct_asian      -6.033e-03  4.941e-02  -0.122  0.90282    
## pct_bach_deg25_over:pct_other_race  2.918e-02  3.343e-02   0.873  0.38282    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.74 on 3027 degrees of freedom
## Multiple R-squared:  0.4973, Adjusted R-squared:  0.4942 
## F-statistic: 157.6 on 19 and 3027 DF,  p-value: < 2.2e-16
```

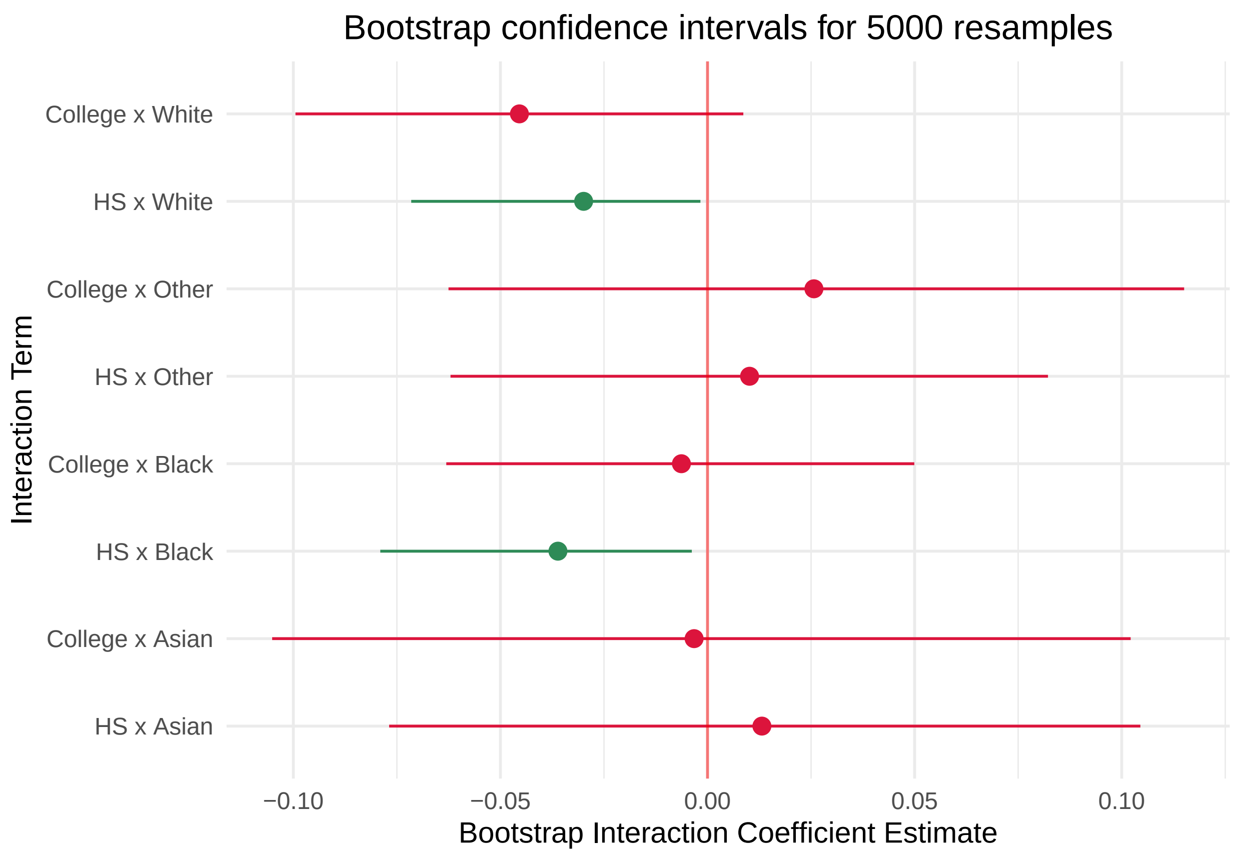
```
# Function for recreating the interaction model for bootstrapping
create_int_model = function(data) {

  # lm(target_death_rate ~ pct_no_hs18_24 + pct_no_hs18_24 + pct_some_col18_24 +
  #            pct_bach_deg18_24 + pct_white + pct_black + pct_asian + pct_other_race +
  #            pct_no_hs18_24*pct_white + pct_no_hs18_24*pct_black + pct_no_hs18_24*pct_asian +
  #            pct_no_hs18_24*pct_other_race + pct_hs18_24*pct_white + pct_hs18_24*pct_black + pct_hs18
  #            pct_hs18_24*pct_other_race + pct_some_col18_24*pct_white + pct_some_col18_24*pct_black +
  #            pct_some_col18_24*pct_other_race + pct_bach_deg18_24*pct_white + pct_bach_deg18_24*pct_b
  #            pct_bach_deg18_24*pct_asian + pct_bach_deg18_24*pct_other_race +
  #            incidence_rate + med_income + pop_est2015 + poverty_percent + median_age,
  #          data = data)


  lm(target_death_rate ~ pct_hs25_over + pct_bach_deg25_over +
           pct_white + pct_black + pct_asian + pct_other_race +
           # Interactions
           pct_white*pct_hs25_over + pct_black*pct_hs25_over + pct_asian*pct_hs25_over + pct_other_rac
           pct_white*pct_bach_deg25_over + pct_black*pct_bach_deg25_over + pct_asian*pct_bach_deg25_o
           # Confounders
           incidence_rate + med_income + pop_est2015 + poverty_percent + median_age,
         data = data)

}
```

```
# How many bootstrap datasets do I want
bs_n = 5000

# Terms to keep
# terms = model$coefficients %>% names %>% .[15:30]
terms = model2$coefficients %>% names %>% .[13:20]
```

```
# Create the bootstrap datasets and models
bs = tibble( idx = 1:bs_n ) %>%
  mutate(
    bs_data = map(idx, function(i) {
      sample_n(cancer, size = nrow(cancer), replace = TRUE)
    }),
    bs_model = map(bs_data, function(bsd) {
      create_int_model(bsd)
    }),
    bs_results = map(bs_model, broom::tidy)
  ) %>%
  select(idx, bs_results) %>%
  unnest(bs_results) %>%
  group_by(term) %>%
  summarize(
    n = n(),
    bs_mean = mean(estimate),
    bs_var = var(estimate),
    left_bound = quantile(estimate, 0.025),
    right_bound = quantile(estimate, 0.975),
  ) %>%
  filter(term %in% terms) %>%
  # Convert terms to factors for easier reordering
  mutate(
    # Model 1 formatting
    # term = factor(term,
    #                levels = c(
    #                  "pct_no_hs18_24:pct_asian", "pct_asian:pct_hs18_24",
    #                  "pct_some_col18_24:pct_asian", "pct_bach_deg18_24:pct_asian",
    #                  "pct_no_hs18_24:pct_black", "pct_black:pct_hs18_24",
    #                  "pct_some_col18_24:pct_black","pct_bach_deg18_24:pct_black",
    #                  "pct_no_hs18_24:pct_other_race", "pct_other_race:pct_hs18_24",
    #                  "pct_some_col18_24:pct_other_race", "pct_bach_deg18_24:pct_other_race",
    #                  "pct_no_hs18_24:pct_white", "pct_white:pct_hs18_24",
    #                  "pct_some_col18_24:pct_white", "pct_bach_deg18_24:pct_white"),
    #                labels = c(
    #                  "No HS x Asian", "Some HS x Asian",
    #                  "Some College x Asian", "College x Asian",
    #                  "No HS x Black", "Some HS x Black",
    #                  "Some College x Black", "College x Black",
    #                  "No HS x Other", "Some HS x Other",
    #                  "Some College x Other", "College x Other",
    #                  "No HS x White", "Some HS x White",
    #                  "Some College x White", "College x White"
    #                ))
    term = factor(term,
                  levels = c(
                    "pct_hs25_over:pct_asian", "pct_bach_deg25_over:pct_asian",
                    "pct_hs25_over:pct_black", "pct_bach_deg25_over:pct_black",
                    "pct_hs25_over:pct_other_race", "pct_bach_deg25_over:pct_other_race",
                    "pct_hs25_over:pct_white", "pct_bach_deg25_over:pct_white"),
                  labels = c(
                    "HS x Asian", "College x Asian",
                    "HS x Black", "College x Black",
                    "HS x Other", "College x Other",
                    "HS x White", "College x White"
                  ))
  )
```

```
## ‘summarise()‘ ungrouping output (override with ‘.groups‘ argument)
```

```
# Visualize the bootstrap confidence intervals
bs %>%
  ggplot(aes(x = term, y = bs_mean)) +
  geom_pointrange(aes(ymin = left_bound, ymax = right_bound,
                      color = if_else(left_bound > 0 | right_bound < 0, "y", "n"))
                  ) +
  geom_hline(yintercept = 0, color = "red", alpha = 0.5) +
  coord_flip() +
  labs(
    title = "Bootstrap confidence intervals for 5000 resamples",
    x = "Interaction Term",
    y = "Bootstrap Interaction Coefficient Estimate"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "none"
  ) +
  scale_color_manual(values = c("#DC143C", "#2E8B57"))
```



Wald Test

```r
# Create the contrast matrix for coefficients
C = matrix(integer(8 * 20), nrow = 8, ncol = 20)
C[1, 13] = C[2, 14] = C[3, 15] = C[4, 16] = C[5, 17] = C[6, 18] = C[7, 19] = C[8, 20] = 1
C[1, 14] = C[2, 15] = C[3, 16] = C[4, 17] = C[5, 18] = C[6, 19] = C[7, 20] = -1

# Constants for null distribution
q = nrow(C)
p = model2$coefficients %>% length

# Contrast on the model coefficients and get var-cov matrix of coefficients
g_beta = C %*% model2$coefficients
V = vcov(model2)

# Get Wald test statistic
W = (t(g_beta) %*% solve(C %*% V %*% t(C)) %*% g_beta) / q

# Calculate the p-value
pval = 1 - pchisq(W, df = nrow(C))
```

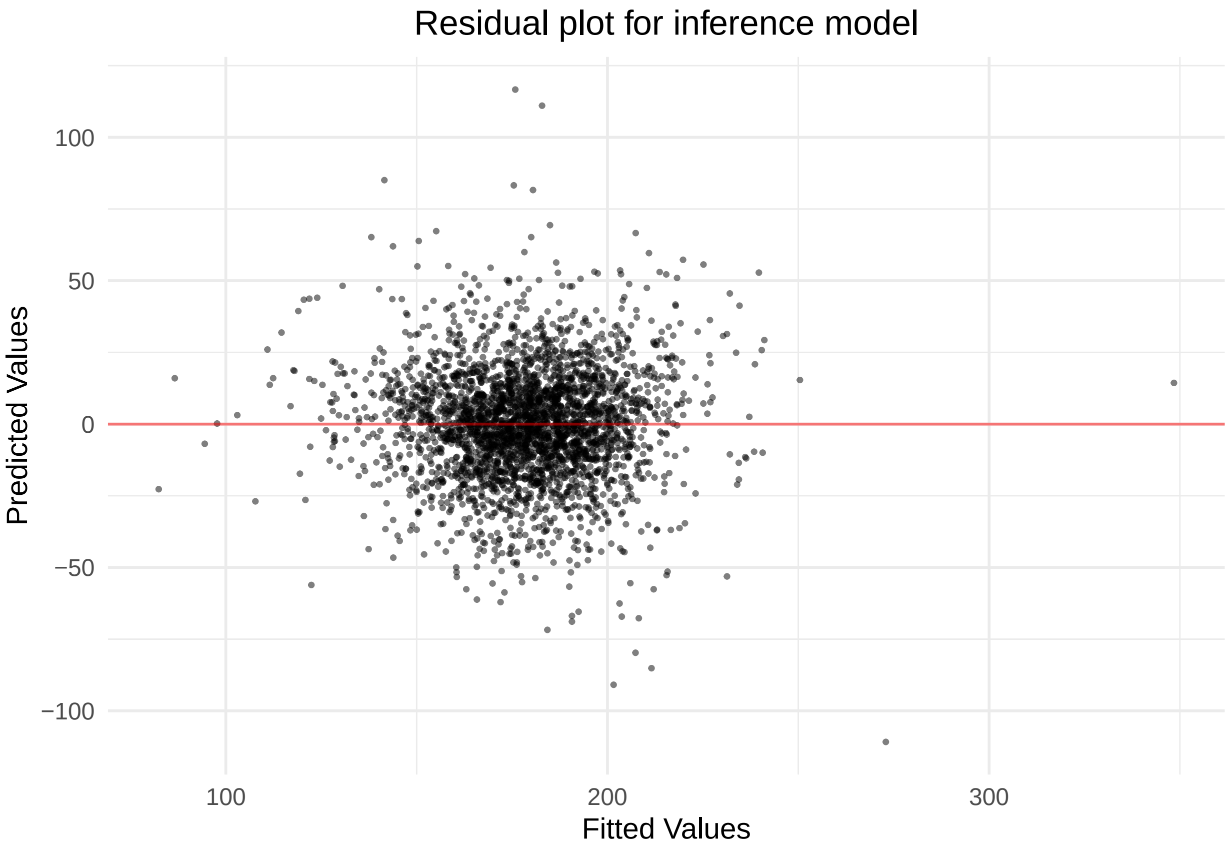
Bootstrap Wald

```r
# Use bootstrap to calculate the Wald Test
bs = tibble( idx = 1:bs_n ) %>%
  mutate(
    bs_data = map(idx, function(i) {
      sample_n(cancer, size = nrow(cancer), replace = TRUE)
    }),
    bs_model = map(bs_data, function(bsd) {
      create_int_model(bsd)
    }),
    bs_wald = map(bs_model, function(model) {
      q = nrow(C)
      g_beta = C %*% model$coefficients
      V = vcov(model)

      (t(g_beta) %*% solve(C %*% V %*% t(C)) %*% g_beta) / q
    })
  )

# How many of the resulting boostrap Wald tests are greater than the critical
# value under the null hypothesis
((bs %>% pull(bs_wald))  > qchisq(0.95, df = nrow(C))) %>% mean
```

```
## [1] 0.1626
```

Model Diagnostics

```r
cancer = cancer %>%
  mutate(
    fitted = model2$fitted.values,
    resid = model2$residuals
  )


# Residuals plot
cancer %>%
  ggplot(aes(x = fitted, y = resid)) +
  geom_point(size = 0.5, alpha = 0.5) +
  geom_hline(yintercept = 0, color = "red", alpha = 0.5) +
  theme_minimal() +
  labs(
    title = "Residual plot for inference model",
    x = "Fitted Values",
    y = "Predicted Values"
  ) +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "none"
  )
```



```
cancer %>% pull(resid) %>% mean
```

```
## [1] -5.436021e-16
```

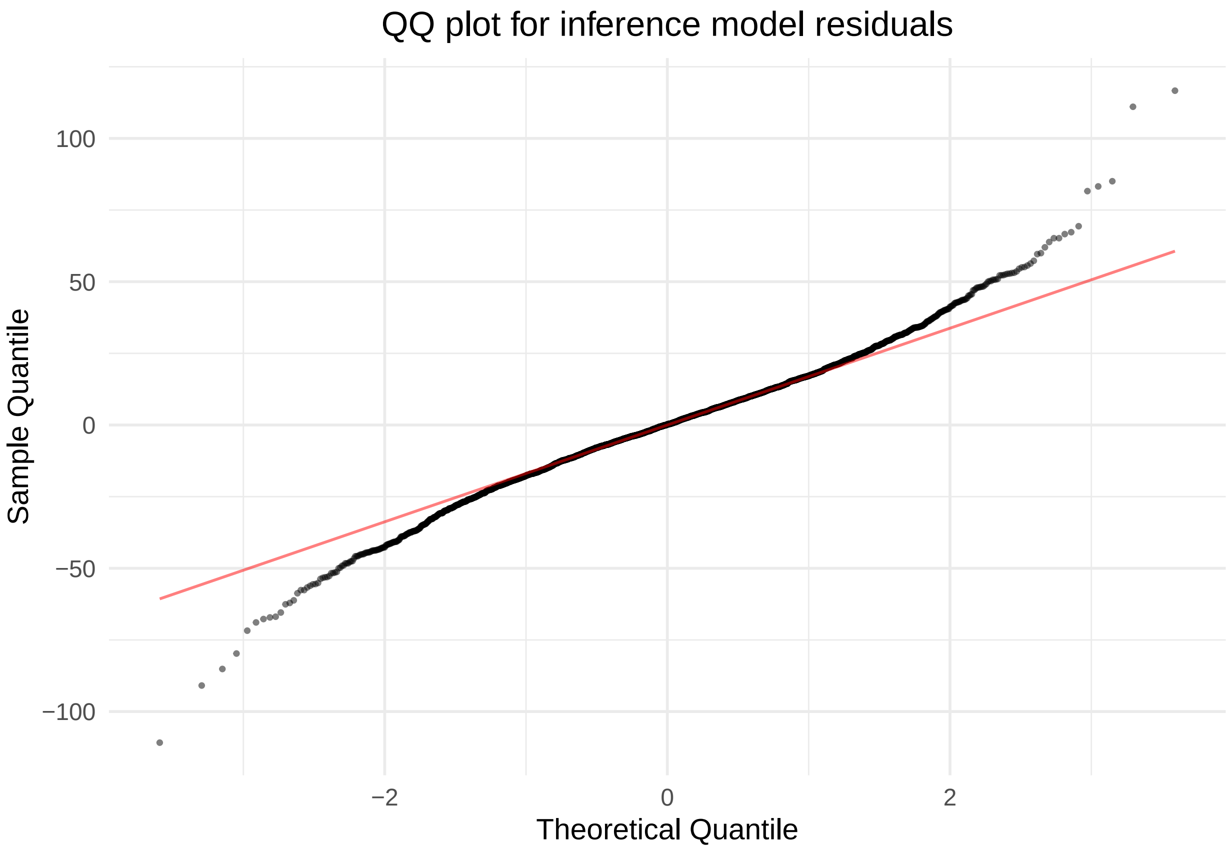
```
cancer %>% pull(resid) %>% var
```

```
## [1] 387.1482
```

```
# QQ-plot
theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "none"
  )
```

```
## List of 2
##  $ legend.position: chr "none"
##  $ plot.title     :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0.5
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##    ..$ margin       : NULL
##    ..$ debug        : NULL
##    ..$ inherit.blank: logi FALSE
##    ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi FALSE
##  - attr(*, "validate")= logi TRUE
```

```
# QQ plot demonstrates that errors have some heavy tails
# Suggests the errors are not normally distributed
cancer %>%
  ggplot(aes(sample = resid)) +
  stat_qq(size = 0.5, alpha = 0.5) +
  stat_qq_line(color = "red", alpha = 0.5) +
  theme_minimal() +
  labs(
    title = "QQ plot for inference model residuals",
    x = "Theoretical Quantile",
    y = "Sample Quantile"
  ) +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "none"
  )
```



QQ-plot suggests that the errors are not normally distributed, which means that the LSE estimates may not be normally distributed. This justifies the use of bootstrap to attempt to check the significance of the model estimates.