Opioid Distributional Analysis

Scott C. Ganz

02 November, 2018

Introduction

This project distributes national estimates of the costs of the opioid epidemic among U.S. states and counties. Our aggregate cost measures come from the CEA report "The Underestimate Cost of the Opioid Crisis" (2017). CEA breaks down their $504 billion estimate for 2015 into non-mortality and mortality related costs as follows:

Cost Type	Cost	Estimate for 2015 (CEA)
Mortality	Opioid-related Death Costs	$431.7 billion
Non-Mortality	Health Care Costs	$36.6 billion
Non-Mortality	Productivity Costs	$25.9 billion
Non-Mortality	Criminal Justice Costs	$9.7 billion

We allocate these costs across states and counties based on variation in opioid-related deaths (mortality costs), opioid-related hospitalization costs (non-mortality health care costs), local work characteristics and opioid addiction rates (non-mortality productivity costs), and per-capita criminal justice costs (non-mortality criminal justice costs).

Data Structure

- Opioid deaths
- Medical costs
- Opioid addiction
- Criminal justice costs
- Productivity (income)

Data Collection/Cleaning

County list

```r
# read in county and states from the choroplethrMaps package
data(county.regions)
data(state.regions)
```

Opioid Deaths by County

Data on local opioid/general drug-related deaths come from the CDC WONDER Underlying Cause of Death database, as well as the Multiple Cause of Death database. Relevant variables include:

- Opioid Deaths (Multiple Cause of Death)
- Drug-related Deaths (Underlying Cause of Death)
- Drug-related Deaths (Multiple Cause of Death)

add state/county missing/observed breakdown

```
# Multiple cause of death data: opioid-related deaths
opioid_deaths <- read_tsv("../data/Opioid Deaths 2016 - MCD.txt") %>%
  select(-Notes) %>%
  filter(!is.na(County)) %>% # keep non-missing counties

# Only variables are county code and number of deaths
  mutate(region = `County Code` %>% as.numeric,
         opioid_deaths = Deaths) %>%
  select(region, opioid_deaths)
```

```
## Parsed with column specification:
## cols(
##   Notes = col_character(),
##   County = col_character(),
##   `County Code` = col_character(),
##   Deaths = col_integer(),
##   Population = col_integer(),
##   `Crude Rate` = col_character()
## )

## Warning in rbind(names(probs), probs_f): number of columns of result is not
## a multiple of vector length (arg 1)

## Warning: 89 parsing failures.
## row # A tibble: 5 x 5 col     row col   expected  actual    file                                   ex
## ... ................. ... ........................................................................ .
## See problems(...) for more details.
```

Overall Drug-related Deaths

```
# Underlying cause of death data: drug deaths
drug_deaths_ucd <- read_tsv("../data/Drug Deaths 2016 - UCD.txt") %>%
  filter(!is.na(`County Code`)) %>%
  mutate(region = `County Code` %>% as.numeric,
         drug_deaths_ucd = Deaths) %>%
  select(region, drug_deaths_ucd)
```

```
## Parsed with column specification:
## cols(
##   Notes = col_character(),
##   County = col_character(),
##   `County Code` = col_character(),
##   Deaths = col_integer(),
##   Population = col_integer(),
##   `Crude Rate` = col_character()
## )

## Warning in rbind(names(probs), probs_f): number of columns of result is not
## a multiple of vector length (arg 1)

## Warning: 80 parsing failures.
## row # A tibble: 5 x 5 col     row col   expected  actual    file                                expo
## ... ................. ... ...................................................................... ...
## See problems(...) for more details.
```

```r
# Multiple cause of death data: drug deaths
drug_deaths_mcd <- read_tsv("../data/Drug Deaths 2016 - MCD.txt") %>%
  filter(!is.na(`County Code`)) %>%
  mutate(region = `County Code` %>% as.numeric,
         drug_deaths_mcd = Deaths) %>%
  select(region, drug_deaths_mcd)
```

```
## Parsed with column specification:
## cols(
##   Notes = col_character(),
##   County = col_character(),
##   `County Code` = col_character(),
##   Deaths = col_integer(),
##   Population = col_integer(),
##   `Crude Rate` = col_character()
## )

## Warning in rbind(names(probs), probs_f): number of columns of result is not
## a multiple of vector length (arg 1)

## Warning: 126 parsing failures.
## row # A tibble: 5 x 5 col     row col         expected   actual     file
## ... ................. ... ..........................................................................
## See problems(...) for more details.
```

```r
# Combine drug death data
drug_deaths <- county.regions %>% select(region) %>%
  left_join(drug_deaths_ucd) %>%
  left_join(drug_deaths_mcd)
```

```
## Joining, by = "region"

## Joining, by = "region"
```

GPCI by County

Data on health care costs come from the Centers for Medicare & Medicaid Services. We use the 2018 geographic practice cost index (GPCI) for services covered by the Medicare Physician Fee Schedule. Relevant variables include:

- Medical cost (GPCI)

```r
# read in GPCI geographic crosswalk
gpci_xwalk <- read_excel("../data/gpci_county_xwalk.xlsx", skip = 1) %>%
  filter(!is.na(`Carrier Number`)) %>%
  filter(!row_number() == n()) %>% #drop last row
  fill(State) %>% #fill states forward
  mutate(`MAC LOCALITY` = str_c(`Carrier Number`, `Locality Number`) %>% as.numeric)

# GPCI data
gpci_level <- read_csv("../data/gpci_2018.csv") %>%
  select(`MAC LOCALITY`, `GPCI WORK`, `GPCI PE`, `GPCI MP`) %>%
  mutate(GPCI = 4.5 * `GPCI WORK` + 1.42 * `GPCI PE` + 0.38 * `GPCI MP`) %>%
  right_join(gpci_xwalk) %>%
  mutate(State = ifelse(State == "HAWAII/GUAM", "HAWAII", State)) %>%
  unique
```

```
## Warning: Duplicated column names deduplicated: 'PROC STAT' => 'PROC
## STAT_1' [14], 'PCTC' => 'PCTC_1' [37], 'CONV FACT' => 'CONV FACT_1' [49],
## 'NOT USED FOR MEDICARE' => 'NOT USED FOR MEDICARE_1' [50]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   `MAC LOCALITY` = col_integer(),
##   `GPCI WORK` = col_double(),
##   `GPCI PE` = col_double(),
##   `GPCI MP` = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
## Joining, by = "MAC LOCALITY"
```

```
# Code 99291: RVU WORK = 4.5, RVU PE = 1.42. RVU MP = 0.38 FROM CMS

# function to get medical cost values
find_gpci = function(county, state) {
  out <- gpci_level %>% filter(str_detect(State %>% tolower, state),
                               str_detect(Counties %>% tolower, county))
  if (nrow(out) == 1) { # if the county exists
     return(out %>% pull(GPCI))
  } else { # otherwise pull all counties/other counties value
    return(gpci_level %>% filter(State %>% tolower == state,
                                 str_detect(Counties, "ALL COUNTIES|ALL OTHER COUNTIES")) %>%
             pull(GPCI))
  }
}

# for each county in the US, attempt to read GPCI
medical_cost_index <- county.regions %>%
  select(region, county.name, state.name) %>%
  rowwise() %>%
  mutate(medical_cost = find_gpci(county.name, state.name)) %>%
  ungroup %>%
  select(region, medical_cost)
```

Opioid Addiction Rates

Opioid addiction rate data come from the Substance Abuse and Mental Health Services Administration (SAMHSA). We use non-medical use of pain relievers data at the substate/metro area level from 2012-2014. The 2012-2014 NSDUH Substate Region Estimates are available here. The data for our analysis come from Table 8. Relevant variables include:

- Fraction of individuals using pain relievers for nonmedical use

The following code maps substate areas to counties;

```
# This crosswalk maps substate regions to county fips codes
addiction_xwalk1 <- read.sas7bdat("../data/substate_county121314.sas7bdat") %>% tbl_df %>%
  select(sbst14n, state, county) %>%
  mutate(county_fips = state * 1000 + county)

# This crosswalk maps census tracts to county fips codes
```

```r
# if > 1 tract in a county using the tract that is most representative
addiction_xwalk2 <- read.sas7bdat("../data/substate_tract121314.sas7bdat") %>% tbl_df %>%
  select(sbst14n, state, county) %>%
  group_by(sbst14n, state, county) %>%
  summarise(n_tracts = n()) %>%
  arrange(state, county, -n_tracts) %>%
  group_by(state, county) %>%
  filter(row_number() == 1) %>%
  ungroup %>%
  select(-n_tracts) %>%
  mutate(county_fips = state * 1000 + county)

# This final addiction crosswalk combines census tract and substate regions
addiction_xwalk <- addiction_xwalk1 %>% bind_rows(addiction_xwalk2) %>% unique
```

```
## Warning in bind_rows_(x, .id): Unequal factor levels: coercing to character

## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector

## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector
```

```r
# Merge non-medical opioid use data to state crosswalk
nonmedical_use <- read_excel("../data/NonMed Use of Opioid.xlsx") %>%
  mutate(state_name = State %>% tolower,
         nonmedical_use_pct = `Small \r\nArea Estimate`,
         sbst14n = `Substate Region`) %>%
  select(state_name, nonmedical_use_pct, sbst14n) %>%
  left_join(state.regions %>% rename(state_name = region, state_fips = fips.numeric) %>%
              select(state_name, state_fips)) %>%
  filter(!is.na(state_fips))
```

```
## Joining, by = "state_name"
```

```r
# Merge county-level crosswalk to nonmedical_use data
# correcting error in crosswalk
nonmedical_use <- addiction_xwalk %>% rename(state_fips = state) %>%
  mutate(sbst14n = case_when(sbst14n == "Trillium Health Resources 1" ~ "Trillium Healthcare Resources 1
                             sbst14n == "Trillium Health Resources 2" ~ "Trillium Healthcare Resources 2
                             TRUE ~ sbst14n)) %>%
  left_join(nonmedical_use) %>%
  select(county_fips, nonmedical_use_pct) %>%
  rename(region = county_fips) %>%
  filter(!is.na(region))
```

```
## Joining, by = c("sbst14n", "state_fips")
```

```r
# merge in the rest of the counties
nonmedical_use <- county.regions %>% select(region, state.fips.character) %>% left_join(nonmedical_use)
```

```
## Joining, by = "region"
```

```r
# NOTE: Ask what this function does
knitr:::knit_code$get()
```

```
## $`unnamed-chunk-1`
## [1] "  # pdf_document:"   "  #   fig_height: 6" "  #   fig_width: 10"
## [4] "  # html_document:"  "  #   fig_height: 6" "  #   fig_width: 10"
## [7] "  # word_document:"  "  #   fig_height: 6" "  #   fig_width: 10"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$include
## [1] FALSE
##
## attr(,"chunk_opts")$label
## [1] "unnamed-chunk-1"
##
##
## $setup
##  [1] "require(mosaic)   # Load additional packages here "
##  [2] "require(dplyr)"
##  [3] "require(readr)"
##  [4] "require(readxl)"
##  [5] "require(tidyr)"
##  [6] "require(stringr)"
##  [7] "require(choroplethr)"
##  [8] "require(choroplethrMaps)"
##  [9] "require(sas7bdat)"
## [10] "require(tidycensus)"
## [11] "require(lme4)"
## [12] "require(glmmTMB)"
## [13] "require(ggplot2)"
## [14] "library(RColorBrewer)"
## [15] ""
## [16] "census_api_key('69d9f9e5add214b53a97d01d26fbec5051720528')"
## [17] ""
## [18] "# this function calls the \"process_hospitalizations.R\" code, which cleans the hospitalization
## [19] ""
## [20] "source('process_hospitalizations.R')"
## [21] "# Some customization.  You can alter or delete as desired (if you know what you are doing)."
## [22] "# trellis.par.set(theme=theme.mosaic()) # change default color scheme for lattice"
## [23] "knitr::opts_chunk$set("
## [24] "  tidy=FALSE,     # display code as typed"
## [25] "  size=\"small\")   # slightly smaller font for code"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "setup"
##
## attr(,"chunk_opts")$include
## [1] FALSE
##
##
## $`unnamed-chunk-2`
## [1] "costdt <- read_excel(\"../data/CEA_Estimates.xlsx\")"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$include
## [1] FALSE
##
## attr(,"chunk_opts")$label
## [1] "unnamed-chunk-2"
##
```

```
##
## $county.list
## [1] "# read in county and states from the choroplethrMaps package"
## [2] "data(county.regions)"
## [3] "data(state.regions)"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "county.list"
##
##
## $opioid.deaths
## [1] "# Multiple cause of death data: opioid-related deaths"
## [2] "opioid_deaths <- read_tsv(\"../data/Opioid Deaths 2016 - MCD.txt\") %>%"
## [3] "  select(-Notes) %>% "
## [4] "  filter(!is.na(County)) %>% # keep non-missing counties"
## [5] "  "
## [6] "# Only variables are county code and number of deaths"
## [7] "  mutate(region = `County Code` %>% as.numeric,"
## [8] "         opioid_deaths = Deaths) %>%"
## [9] "  select(region, opioid_deaths) "
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "opioid.deaths"
##
##
## $overall.drug.deaths
##  [1] "# Underlying cause of death data: drug deaths"
##  [2] "drug_deaths_ucd <- read_tsv(\"../data/Drug Deaths 2016 - UCD.txt\") %>%"
##  [3] "  filter(!is.na(`County Code`)) %>%"
##  [4] "  mutate(region = `County Code` %>% as.numeric,"
##  [5] "         drug_deaths_ucd = Deaths) %>%"
##  [6] "  select(region, drug_deaths_ucd)"
##  [7] ""
##  [8] "# Multiple cause of death data: drug deaths"
##  [9] "drug_deaths_mcd <- read_tsv(\"../data/Drug Deaths 2016 - MCD.txt\") %>%"
## [10] "  filter(!is.na(`County Code`)) %>%"
## [11] "  mutate(region = `County Code` %>% as.numeric,"
## [12] "         drug_deaths_mcd = Deaths) %>%"
## [13] "  select(region, drug_deaths_mcd)"
## [14] ""
## [15] "# Combine drug death data"
## [16] "drug_deaths <- county.regions %>% select(region) %>%"
## [17] "  left_join(drug_deaths_ucd) %>%"
## [18] "  left_join(drug_deaths_mcd)"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "overall.drug.deaths"
##
##
## $gpci
##  [1] "# read in GPCI geographic crosswalk"
##  [2] "gpci_xwalk <- read_excel(\"../data/gpci_county_xwalk.xlsx\", skip = 1) %>%"
##  [3] "  filter(!is.na(`Carrier Number`)) %>%"
##  [4] "  filter(!row_number() == n()) %>% #drop last row "
##  [5] "  fill(State) %>% #fill states forward"
##  [6] "  mutate(`MAC LOCALITY` = str_c(`Carrier Number`, `Locality Number`) %>% as.numeric)"
##  [7] ""
##  [8] "# GPCI data"
##  [9] "gpci_level <- read_csv(\"../data/gpci_2018.csv\") %>%"
## [10] "  select(`MAC LOCALITY`, `GPCI WORK`, `GPCI PE`, `GPCI MP`) %>%"
## [11] "  mutate(GPCI = 4.5 * `GPCI WORK` + 1.42 * `GPCI PE` + 0.38 * `GPCI MP`) %>%"
## [12] "  right_join(gpci_xwalk) %>%"
## [13] "  mutate(State = ifelse(State == \"HAWAII/GUAM\", \"HAWAII\", State)) %>%"
## [14] "  unique"
## [15] ""
## [16] ""
## [17] "# Code 99291: RVU WORK = 4.5, RVU PE = 1.42. RVU MP = 0.38 FROM CMS"
## [18] ""
## [19] "# function to get medical cost values"
## [20] "find_gpci = function(county, state) {"
## [21] "  out <- gpci_level %>% filter(str_detect(State %>% tolower, state),"
## [22] "                        str_detect(Counties %>% tolower, county))"
## [23] "  if (nrow(out) == 1) { # if the county exists"
## [24] "     return(out %>% pull(GPCI))"
## [25] "  } else { # otherwise pull all counties/other counties value"
## [26] "    return(gpci_level %>% filter(State %>% tolower == state,"
## [27] "                                 str_detect(Counties, \"ALL COUNTIES|ALL OTHER COUNTIES\")) %>%
## [28] "             pull(GPCI))"
## [29] "  }"
## [30] "}"
## [31] ""
## [32] "# for each county in the US, attempt to read GPCI"
## [33] "medical_cost_index <- county.regions %>%"
## [34] "  select(region, county.name, state.name) %>%"
## [35] "  rowwise() %>% "
## [36] "  mutate(medical_cost = find_gpci(county.name, state.name)) %>%"
## [37] "  ungroup %>%"
## [38] "  select(region, medical_cost)"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "gpci"
##
##
## $opioid.addiction
##  [1] "# This crosswalk maps substate regions to county fips codes"
##  [2] "addiction_xwalk1 <- read.sas7bdat(\"../data/substate_county121314.sas7bdat\") %>% tbl_df %>%"
##  [3] "  select(sbst14n, state, county) %>%"
##  [4] "  mutate(county_fips = state * 1000 + county)"
##  [5] ""
##  [6] "# This crosswalk maps census tracts to county fips codes"
##  [7] "# if > 1 tract in a county using the tract that is most representative"
##  [8] "addiction_xwalk2 <- read.sas7bdat(\"../data/substate_tract121314.sas7bdat\") %>% tbl_df %>%"
##  [9] "  select(sbst14n, state, county) %>% "
## [10] "  group_by(sbst14n, state, county) %>%"
## [11] "  summarise(n_tracts = n()) %>%"
## [12] "  arrange(state, county, -n_tracts) %>%"
## [13] "  group_by(state, county) %>%"
## [14] "  filter(row_number() == 1) %>%"
## [15] "  ungroup %>%"
## [16] "  select(-n_tracts) %>%"
## [17] "  mutate(county_fips = state * 1000 + county)"
## [18] ""
## [19] "# This final addiction crosswalk combines census tract and substate regions"
## [20] "addiction_xwalk <- addiction_xwalk1 %>% bind_rows(addiction_xwalk2) %>% unique"
## [21] ""
## [22] "# Merge non-medical opioid use data to state crosswalk "
## [23] "nonmedical_use <- read_excel(\"../data/NonMed Use of Opioid.xlsx\") %>%"
## [24] "  mutate(state_name = State %>% tolower,"
## [25] "         nonmedical_use_pct = `Small \\r\\nArea Estimate`,"
## [26] "         sbst14n = `Substate Region`) %>%"
## [27] "  select(state_name, nonmedical_use_pct, sbst14n) %>%"
## [28] "  left_join(state.regions %>% rename(state_name = region, state_fips = fips.numeric) %>%"
## [29] "              select(state_name, state_fips)) %>%"
## [30] "  filter(!is.na(state_fips))"
## [31] ""
## [32] "# Merge county-level crosswalk to nonmedical_use data"
## [33] "# correcting error in crosswalk"
## [34] "nonmedical_use <- addiction_xwalk %>% rename(state_fips = state) %>% "
## [35] "  mutate(sbst14n = case_when(sbst14n == \"Trillium Health Resources 1\" ~ \"Trillium Healthcar
## [36] "                            sbst14n == \"Trillium Health Resources 2\" ~ \"Trillium Healthcare
## [37] "                            TRUE ~ sbst14n)) %>% "
## [38] "  left_join(nonmedical_use) %>%"
## [39] "  select(county_fips, nonmedical_use_pct) %>%"
## [40] "  rename(region = county_fips) %>%"
## [41] "  filter(!is.na(region))"
## [42] ""
## [43] "# merge in the rest of the counties"
## [44] "nonmedical_use <- county.regions %>% select(region, state.fips.character) %>% left_join(nonmedi
## [45] "# NOTE: Ask what this function does"
## [46] "knitr:::knit_code$get()"
## [47] ""
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "opioid.addiction"
##
##
## $criminal.justice
##  [1] "# read in criminal justice costs"
##  [2] "criminal_costs <- read_csv(\"../data/criminal_justice_expenditures.csv\","
##  [3] "              skip = 25, col_names = F) %>%"
##  [4] "  na.omit %>%"
##  [5] "  filter(X2 != \"-\") %>%"
##  [6] "  select(X1, X2, X4) %>%"
##  [7] "  mutate(state = X1, criminal_cost_per_capita = X4/(X2 %>% as.numeric * 1000)) %>% # create pr
##  [8] "  select(state, criminal_cost_per_capita)"
##  [9] ""
## [10] "# data are at the state level-- merge in corresponding counties"
## [11] "criminal_costs <- county.regions %>%"
## [12] "  select(region, state.name) %>%"
## [13] "  left_join(criminal_costs %>% mutate(state.name = state %>% tolower) %>%"
## [14] "              select(state.name, criminal_cost_per_capita)) %>%"
## [15] "  select(region, criminal_cost_per_capita)"
## [16] ""
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "criminal.justice"
##
##
## $hospitalization.costs
##  [1] "# read in hospitalization cost data-- these are already cleaned with the \"process_hospitaliza
##  [2] "hospitalization_costs <- read_rds(\"../data/hospitalization_costs.rds\")"
##  [3] ""
##  [4] "# county-level data"
##  [5] "county_hosp_costs <- hospitalization_costs %>% filter(region_name != \"State Total\") %>%"
##  [6] "  mutate(county.name = region_name %>% tolower %>% str_trim,"
##  [7] "         state.name = state_name %>% tolower %>% str_trim) %>%"
##  [8] "  inner_join(county.regions) %>%"
##  [9] "  select(region, discharges, mean_costs, per_capita_costs, total_costs)"
## [10] ""
## [11] "# state-level data"
## [12] "state_hosp_costs <- hospitalization_costs %>% filter(region_name == \"State Total\") %>%"
## [13] "  mutate(region = state_name %>% tolower %>% str_trim,"
## [14] "         state_total_costs = total_costs,"
## [15] "         state_discharges = discharges,"
## [16] "         state_mean_costs = mean_costs) %>%"
## [17] "  select(region, state_total_costs, state_discharges, state_mean_costs)"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "hospitalization.costs"
##
##
## $county.demographics
##   [1] "# population by county"
##   [2] "population <- get_acs(geography = \"county\","
##   [3] "             variables = c(population = \"B01003_001\"),"
##   [4] "             survey = \"acs5\","
##   [5] "             year = 2016) %>%"
##   [6] "  mutate(region = GEOID %>% as.numeric,"
##   [7] "         total_population = estimate) %>%"
##   [8] "  select(region, total_population)"
##   [9] ""
##  [10] "# working age population by county"
##  [11] "workers <- get_acs(geography = \"county\","
##  [12] "             variables = c(m2534 = \"B15001_011\", # get males and females aged 25 through 64
##  [13] "                           m3544 = \"B15001_019\", "
##  [14] "                           m4564 = \"B15001_027\","
##  [15] "                           f2534 = \"B15001_052\","
##  [16] "                           f3544 = \"B15001_060\","
##  [17] "                           f4564 = \"B15001_068\"),"
##  [18] "             survey = \"acs5\","
##  [19] "             year = 2016) %>%"
##  [20] "  group_by(GEOID) %>%"
##  [21] "  summarise(working_population = sum(estimate, na.rm = T)) %>% #sum population by GEOID (coun
##  [22] "  mutate(region = GEOID %>% as.numeric) %>% "
##  [23] "  select(region, working_population) %>% ungroup"
##  [24] ""
##  [25] "# educational attainment by county"
##  [26] "educational_attainment <- get_acs(\"county\","
##  [27] "                                  variables = c(\"B06009_001\", \"B06009_002\", \"B06009_003\
##  [28] "  group_by(GEOID) %>%"
##  [29] "  summarise(pct_no_hs = estimate[2]/estimate[1],"
##  [30] "            pct_hs = estimate[3]/estimate[1],"
##  [31] "            pct_college = 1 - pct_no_hs - pct_hs) %>%"
##  [32] "  rename(region = GEOID) %>%"
##  [33] "  mutate(region = region %>% as.numeric)"
##  [34] ""
##  [35] "# Urban/rural status by county"
##  [36] "urban_rural <- read_excel(\"../data/percent_rural.xlsx\") %>%"
##  [37] "  mutate(region = GEOID %>% as.numeric,"
##  [38] "         percent_rural = `2010 Census \\r\\nPercent Rural`) %>%"
##  [39] "  select(region, percent_rural)"
##  [40] ""
##  [41] "# Racial breakdown by county"
##  [42] "race <- get_acs(geography = \"county\","
##  [43] "             variables = c(total_population = \"B03002_001\","
##  [44] "                           black_alone_not_hispanic = \"B03002_004\","
##  [45] "                           hispanic_all_races = \"B03002_012\"),"
##  [46] "             survey = \"acs5\", year = 2016) %>%"
##  [47] "  group_by(GEOID) %>%"
##  [48] "  summarise(percent_black = estimate[2]/estimate[1] * 100,"
##  [49] "            percent_hispanic = estimate[3]/estimate[1] * 100) %>%"
##  [50] "  mutate(region = GEOID %>% as.numeric) %>%"
##  [51] "  select(region, percent_black, percent_hispanic)"
##  [52] "  "
##  [53] "# med_income <- get_acs(geography = \"county\","
##  [54] "#                   variables = c(med_income_per_capita = \"B06011_001\"),"
##  [55] "#                   survey = \"acs5\", year = 2016) %>%"
##  [56] "#   mutate(region = GEOID %>% as.numeric,"
##  [57] "#          med_income_per_capita = estimate) %>%"
##  [58] "#   select(region, med_income_p  er_capita)"
##  [59] ""
##  [60] "# mean_income <- get_acs(geography = \"county\","
##  [61] "#                   variables = c(income_per_capita = \"B19301_001\"),"
##  [62] "#                   survey = \"acs5\", year = 2016) %>%"
##  [63] "#   mutate(region = GEOID %>% as.numeric,"
##  [64] "#          mean_income_per_capita = estimate) %>%"
##  [65] "#   select(region, mean_income_per_capita)"
##  [66] ""
##  [67] "# Population by state"
##  [68] "st_population <- get_acs(geography = \"state\","
##  [69] "             variables = c(population = \"B01003_001\"),"
##  [70] "             survey = \"acs5\","
##  [71] "             year = 2016) %>%"
##  [72] "  mutate(region = NAME %>% tolower,"
##  [73] "         total_population = estimate) %>%"
##  [74] "  select(region, total_population)"
##  [75] ""
##  [76] "# Workers by state"
##  [77] "st_workers <- get_acs(geography = \"state\","
##  [78] "             variables = c(m2534 = \"B15001_011\","
##  [79] "                           m3544 = \"B15001_019\","
##  [80] "                           m4564 = \"B15001_027\","
##  [81] "                           f2534 = \"B15001_052\","
##  [82] "                           f3544 = \"B15001_060\","
##  [83] "                           f4564 = \"B15001_068\"),"
##  [84] "             survey = \"acs5\","
##  [85] "             year = 2016) %>%"
##  [86] "  group_by(GEOID) %>%"
##  [87] "  summarise(working_population = sum(estimate, na.rm = T)) %>%"
##  [88] "  rename(region = GEOID) %>% ungroup"
##  [89] ""
##  [90] "# st_income <- get_acs(geography = \"state\","
##  [91] "#                   variables = c(income_per_capita = \"B06011_001\"),"
##  [92] "#                   survey = \"acs5\", year = 2016) %>%"
##  [93] "#   mutate(region = NAME %>% tolower,"
##  [94] "#          income_per_capita = estimate) %>%"
##  [95] "#   select(region, income_per_capita)"
##  [96] ""
##  [97] "# Income by county"
##  [98] "income <- read_excel(\"../data/est16all.xls\", skip = 3) %>% select(`State FIPS Code`, `County
##  [99] ""
## [100] "# Income by state"
## [101] "med_income <- income %>% mutate(med_income = `Median Household Income` %>% as.numeric,"
## [102] "                                region = str_c(`State FIPS Code`, `County FIPS Code`) %>% as.
## [103] "  right_join(county.regions) %>% select(region, med_income)"
## [104] ""
## [105] "# County fips code 000 corresponds to state-level stats"
## [106] "st_income <- income %>% filter(`County FIPS Code` == \"000\") %>%"
## [107] "  mutate(st_med_income = `Median Household Income` %>% as.numeric,"
## [108] "         fips.character = `State FIPS Code`) %>%"
## [109] "  right_join(state.regions) %>%"
## [110] "  select(region, st_med_income)"
## [111] ""
## [112] "# merge county-level demographic data"
## [113] "demographics <- county.regions %>% "
## [114] "  select(region) %>% "
## [115] "  left_join(population) %>%"
## [116] "  left_join(workers) %>%"
## [117] "  left_join(race) %>% "
## [118] "  left_join(med_income) %>%"
## [119] "  left_join(urban_rural) %>%"
## [120] "  left_join(educational_attainment) %>% tbl_df"
## [121] ""
## [122] "# merge state-level demographic data"
## [123] "st_demographics <- state.regions %>%"
## [124] "  select(region, fips.character) %>%"
## [125] "  left_join(st_income) %>%"
## [126] "  left_join(st_population) %>%"
## [127] "  left_join(st_workers)"
## [128] ""
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "county.demographics"
##
```

```
##
## $`unnamed-chunk-3`
## [1] "hosp_regressors = c(\"Log of Total Population\","
## [2] "                    \"Log of Opioid Deaths\","
## [3] "                    \"Critical Care Fee Schedule\","
## [4] "                    \"Log of Median Per-Capita Income\""
## [5] "                    )"
## [6] "hosp_reg_table <- data.frame(Response = c(\"Log of Hospitalization Costs\", rep(\"\", times = l
## attr(,"chunk_opts")
## attr(,"chunk_opts")$include
## [1] FALSE
##
## attr(,"chunk_opts")$label
## [1] "unnamed-chunk-3"
##
##
## $health.costs
##   [1] "# County-level hospitalization costs"
##   [2] "county_hosp_costs <- hospitalization_costs %>% filter(region_name != \"State Total\") %>%"
##   [3] "  mutate(county.name = region_name %>% tolower %>% str_trim,"
##   [4] "         state.name = state_name %>% tolower %>% str_trim) %>%"
##   [5] "  right_join(county.regions)"
##   [6] ""
##   [7] "# State-level hospitalization costs"
##   [8] "state_hosp_costs <- state_hosp_costs %>% right_join(state.regions) %>% rename(state.name = reg
##   [9] ""
##  [10] "# Merge data NOTE: do we need this? I can't find it referenced later in the code"
##  [11] "# hosp_costs <- county_hosp_costs %>% left_join(state_hosp_costs) %>% left_join(demographics)
##  [12] "#   left_join(st_demographics %>% rename(state.name = region, total_st_population = total_popu
##  [13] ""
##  [14] "# data for county-level hospitalization cost regressions"
##  [15] "hosp_costs_model <- county_hosp_costs %>%  "
##  [16] "  left_join(medical_cost_index) %>%"
##  [17] "  left_join(nonmedical_use) %>%"
##  [18] "  left_join(demographics) %>%"
##  [19] "  left_join(opioid_deaths) %>%"
##  [20] "  left_join(drug_deaths) %>%"
##  [21] "  mutate(log_total_hosp_costs = log(total_costs),"
##  [22] "         log_population = log(total_population),"
##  [23] "         log_nonmedical_use_count = log(total_population/1000 * nonmedical_use_pct),"
##  [24] "         log_opioid_deaths = log(opioid_deaths),"
##  [25] "         log_drug_deaths_mcd = log(drug_deaths_mcd))"
##  [26] ""
##  [27] "# mixed-effects linear regression with state random effects--counties with opioid-related dea
##  [28] "h1 <- lmer(log_total_hosp_costs ~ medical_cost + log_nonmedical_use_count + log_population + l
##  [29] ""
##  [30] "# adjustment factor NEED TO ASK ABOUT THIS"
##  [31] "h1_adj = sum(exp(resid(h1)))/(length(resid(h1)) - 7)"
##  [32] ""
##  [33] "# mixed-effects linear regression with state random effects--counties with only drug deaths"
##  [34] "h2 <- lmer(log_total_hosp_costs ~ medical_cost + log_nonmedical_use_count + log_population + l
##  [35] ""
##  [36] "h2_adj = sum(exp(resid(h2)))/(length(resid(h2)) - 6)"
##  [37] ""
## [38] "# mixed-effects linear regression with state random effects--counties without opioid/general 
## [39] "h3 <- lmer(log_total_hosp_costs ~ medical_cost + log_nonmedical_use_count + log_population + 
## [40] ""
## [41] "h3_adj = sum(exp(resid(h3)))/(length(resid(h3)) - 5)"
## [42] ""
## [43] "### State-level hospitalization cost data"
## [44] "# State-level opioid death data-- for use in linear regression"
## [45] "state_opioid_deaths <- read_tsv(\"../data/State Opioid Deaths 2016 - MCD.txt\") %>%"
## [46] "  select(-Notes) %>%"
## [47] "  filter(!is.na(State)) %>%"
## [48] "  mutate(state.fips.character = `State Code`,"
## [49] "         state_opioid_deaths = Deaths) %>%"
## [50] "  select(state.fips.character, state_opioid_deaths)"
## [51] ""
## [52] "# opioid medical cost data -- outcome of interest in regression"
## [53] "state_medical_costs <- medical_cost_index %>% left_join(demographics) %>%"
## [54] "  left_join(county.regions %>% select(region, state.fips.character)) %>%"
## [55] "  group_by(state.fips.character) %>%  "
## [56] "  summarise(medical_cost = sum(total_population/sum(total_population, na.rm = T) * medical_co
## [57] ""
## [58] "# data for state-level regressions"
## [59] "state_hosp_costs_model <- state_hosp_costs %>%"
## [60] "  left_join(st_demographics) %>%"
## [61] "  left_join(state_opioid_deaths %>% rename(fips.character = state.fips.character)) %>%"
## [62] "  left_join(state_medical_costs %>% rename(fips.character = state.fips.character)) %>%"
## [63] "  mutate(log_hospital_costs = log(state_total_costs),"
## [64] "         log_total_population = log(total_population),"
## [65] "         log_mean_cost = log(state_mean_costs),"
## [66] "         log_opioid_deaths = log(state_opioid_deaths),"
## [67] "         log_income = log(st_med_income))"
## [68] ""
## [69] "# linear regression: state hospitalization costs"
## [70] "sh1 <- lm(log_hospital_costs ~ log_total_population +"
## [71] "             log_opioid_deaths + medical_cost + log_income, data = state_hosp_costs_model)"
## [72] ""
## [73] "# This is correcting non-normality of the residuals?"
## [74] "sh1_adj <- sum(exp(resid(sh1)))/26"
## [75] ""
## [76] "# add predicted values for state hospitalization costs, generate per-capita hospitalization c
## [77] "# this code fills in predicted values if state costs are missing, otherwise it returns origin
## [78] "state_hosp_costs <- state_hosp_costs_model %>%"
## [79] "  mutate(state_est_hosp_costs = ifelse(is.na(state_total_costs), exp(predict(sh1, state_hosp_
## [80] "        state_costs_per_capita = state_est_hosp_costs/total_population) %>%"
## [81] "  select(fips.character, state_est_hosp_costs, state_costs_per_capita)"
## [82] ""
## [83] "### county-level hospitalization costs"
## [84] "# for each county, get predicted hospitalization costs based on which independent variables a
## [85] "d1_hosp <- hosp_costs_model %>% filter(!is.na(medical_cost + log_nonmedical_use_count + log_p
## [86] ""
## [87] "d1_hosp <- d1_hosp %>% mutate(predicted_hospitalization_costs1 = exp(predict(h1, d1_hosp, all
## [88] ""
## [89] "d2_hosp <- hosp_costs_model %>% filter(!is.na(medical_cost + log_nonmedical_use_count + log_p
## [90] ""
## [91] "d2_hosp <- d2_hosp %>% mutate(predicted_hospitalization_costs2 = exp(predict(h2, d2_hosp, all
##  [92] ""
##  [93] "d3_hosp <- hosp_costs_model %>% filter(!is.na(medical_cost + log_nonmedical_use_count + log_po
##  [94] ""
##  [95] "d3_hosp <- d3_hosp %>% mutate(predicted_hospitalization_costs3 = exp(predict(h3, d3_hosp, allo
##  [96] ""
##  [97] "# assign county-level hospital costs according to hierarchy of data available: if the real dat
##  [98] "proj_hospital_costs <- hosp_costs_model %>% select(region, state.fips.character, total_costs)
##  [99] "  left_join(state_hosp_costs %>% rename(state.fips.character = fips.character)) %>%"
## [100] "  left_join(d1_hosp %>% select(region, predicted_hospitalization_costs1)) %>% "
## [101] "  left_join(d2_hosp %>% select(region, predicted_hospitalization_costs2)) %>% "
## [102] "  left_join(d3_hosp %>% select(region, predicted_hospitalization_costs3)) %>%"
## [103] "   mutate(est_total_hosp_costs = case_when("
## [104] "    !is.na(total_costs) ~ total_costs,"
## [105] "    !is.na(predicted_hospitalization_costs1) ~ predicted_hospitalization_costs1,"
## [106] "    !is.na(predicted_hospitalization_costs2) ~ predicted_hospitalization_costs2,"
## [107] "    !is.na(predicted_hospitalization_costs3) ~ predicted_hospitalization_costs3)) %>%"
## [108] "  group_by(state.fips.character) %>%"
## [109] "  mutate(total_est_costs = sum(est_total_hosp_costs, na.rm = T)) %>%"
## [110] "  mutate(infl_factor = state_est_hosp_costs/total_est_costs,"
## [111] "         est_total_hosp_costs = est_total_hosp_costs * infl_factor) %>% ungroup"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "health.costs"
##
##
## $opioid_deaths
##  [1] "d <- county.regions %>% tbl_df %>% dplyr::select(region, state.fips.character) %>%"
##  [2] "  left_join(opioid_deaths) %>%"
##  [3] "  left_join(drug_deaths) %>%"
##  [4] "  left_join(nonmedical_use) %>%"
##  [5] "  left_join(demographics) %>%"
##  [6] "  left_join(medical_cost_index) %>%"
##  [7] "  left_join(county_hosp_costs) %>%"
##  [8] "  mutate(nonmedical_use_count = nonmedical_use_pct * total_population/1000,"
##  [9] "         drug_deaths_mcd = drug_deaths_mcd,"
## [10] "         opioid_deaths = opioid_deaths %>% as.numeric) %>%"
## [11] "  mutate(log_drug_deaths_mcd = log(drug_deaths_mcd),"
## [12] "         log_nonmedical_use_count = log(nonmedical_use_count),"
## [13] "         log_population = log(total_population),"
## [14] "         log_income = log(med_income),"
## [15] "         log_med_income = log(med_income),"
## [16] "         log_opioid_deaths = log(opioid_deaths))"
## [17] ""
## [18] "m1 <- glmmTMB(opioid_deaths ~ log_drug_deaths_mcd + log_nonmedical_use_count + log_population +
## [19] "               percent_hispanic + pct_no_hs +"
## [20] "                (1|state.fips.character), data = d, family = nbinom2(link = \"log\"), verbose =
## [21] ""
## [22] "m2 <- glmmTMB(opioid_deaths ~ log_nonmedical_use_count + log_population + log_med_income + perc
## [23] "               percent_hispanic + pct_no_hs +"
## [24] "                (1|state.fips.character), data = d, family = nbinom2(link = \"log\"), verbose =
## [25] ""
## [26] "d1_deaths <- d %>% filter(!is.na(log_drug_deaths_mcd + log_nonmedical_use_count + log_populatio
## [27] "               percent_hispanic + pct_no_hs))"
## [28] ""
## [29] "d1_deaths <- d1_deaths %>% mutate(predicted_opioid_deaths1 = predict(m1, d1_deaths, allow.new.l
## [30] ""
## [31] "d2_deaths <- d %>% filter(!is.na(log_nonmedical_use_count + log_population + log_med_income + p
## [32] "                percent_hispanic + pct_no_hs))"
## [33] ""
## [34] "d2_deaths <- d2_deaths %>% mutate(predicted_opioid_deaths2 = predict(m2, d2_deaths, allow.new.l
## [35] ""
## [36] "proj_opioid_deaths <- d %>% select(region, opioid_deaths, total_population) %>% "
## [37] "  left_join(d1_deaths %>% select(region, predicted_opioid_deaths1)) %>% "
## [38] "  left_join(d2_deaths %>% select(region, predicted_opioid_deaths2)) %>% "
## [39] "  select(region, total_population, opioid_deaths, predicted_opioid_deaths1, predicted_opioid_de
## [40] ""
## [41] "proj_opioid_deaths <- county.regions %>% select(region, state.fips.character, state.abb) %>% "
## [42] "  left_join(proj_opioid_deaths) %>% "
## [43] "  left_join(state_opioid_deaths) %>%"
## [44] "  filter(!is.na(total_population))"
## [45] ""
## [46] "proj_opioid_deaths <- proj_opioid_deaths %>%"
## [47] "  mutate(est_opioid_deaths = case_when("
## [48] "    !is.na(opioid_deaths) ~ opioid_deaths,"
## [49] "    !is.na(predicted_opioid_deaths1) ~ predicted_opioid_deaths1,"
## [50] "    !is.na(predicted_opioid_deaths2) ~ predicted_opioid_deaths2)"
## [51] "  ) %>%"
## [52] "  group_by(state.fips.character) %>%"
## [53] "  mutate(est_state_opioid_deaths = sum(est_opioid_deaths, na.rm = T),"
## [54] "         infl_factor = state_opioid_deaths/est_state_opioid_deaths,"
## [55] "         est_opioid_deaths = est_opioid_deaths * infl_factor) %>% ungroup"
## [56] ""
## [57] "projection_summary1 <- proj_opioid_deaths %>%"
## [58] " select(est_opioid_deaths, opioid_deaths, total_population, state.fips.character) %>%"
## [59] " mutate(modeled = ifelse(is.na(opioid_deaths), 1, 0)) %>%"
## [60] " group_by(state.fips.character) %>%"
## [61] " summarise(opioid_deaths = 100 * sum(est_opioid_deaths, na.rm = T)/sum(total_population, na.rm
## [62] "           modeled_pct = sum(total_population * modeled, na.rm = T)/sum(total_population, na.rm
## [63] ""
## [64] "projection_summary2 <- proj_hospital_costs %>% left_join(demographics) %>%"
## [65] " select(total_costs, est_total_hosp_costs, total_population, state.fips.character) %>%"
## [66] " mutate(modeled = ifelse(is.na(total_costs), 1, 0)) %>%"
## [67] " group_by(state.fips.character) %>% "
## [68] " summarise(per_capita_costs = sum(est_total_hosp_costs, na.rm = T)/sum(total_population, na.rm
## [69] "           modeled_pct = sum(total_population * modeled, na.rm = T)/sum(total_population, na.rm
## [70] ""
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "opioid_deaths"
##
##
## $build.model
##  [1] "fatal_costs <- 431.7"
##  [2] "total_nonfatal_costs <- 72.3"
##  [3] "nonfatal_health_costs <- total_nonfatal_costs * 0.5068966"
##  [4] "total_criminal_justice_costs <- total_nonfatal_costs * 0.1344828"
##  [5] "total_productivity_costs <- total_nonfatal_costs * 0.3586207 "
##  [6] ""
##  [7] ""
##  [8] "d <- county.regions %>% tbl_df %>% "
##  [9] "  left_join(proj_opioid_deaths %>% select(region, est_opioid_deaths)) %>%"
## [10] "  left_join(proj_hospital_costs %>% select(region, est_total_hosp_costs)) %>%"
## [11] "  left_join(nonmedical_use) %>%"
## [12] "  left_join(medical_cost_index) %>%"
## [13] "  left_join(criminal_costs) %>%"
## [14] "  left_join(demographics)"
## [15] ""
## [16] "d <- d %>% filter(!is.na(total_population), !is.na(med_income)) %>% "
## [17] "  mutate(death_wt = est_opioid_deaths/sum(est_opioid_deaths, na.rm = T),"
## [18] "         death_cost = death_wt * fatal_costs * 10^9,"
## [19] "         death_cost_pc = death_cost/total_population) %>%"
## [20] "  mutate(health_wt = est_total_hosp_costs/sum(est_total_hosp_costs),"
## [21] "         health_cost = health_wt * nonfatal_health_costs * 10^9,"
## [22] "         health_cost_pc = health_cost/total_population) %>%"
## [23] "  mutate(criminal_wt = (nonmedical_use_pct * total_population * criminal_cost_per_capita)/sum(
## [24] "         criminal_cost = criminal_wt * total_criminal_justice_costs * 10^9,"
## [25] "         criminal_cost_pc = criminal_cost/total_population) %>%"
## [26] "  mutate(productivity_wt = (med_income * 0.175 * nonmedical_use_pct * working_population)/sum(
## [27] "         productivity_cost = productivity_wt * total_productivity_costs * 10^9,"
## [28] "         productivity_cost_pc = productivity_cost/total_population)"
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "build.model"
##
##
## $make.county.maps
##  [1] ""
##  [2] "o <- d %>% "
##  [3] "  filter(total_population > 2500, !(state.name %in% c(\"alaska\", \"hawaii\"))) %>%"
##  [4] "  mutate(per_capita_nf_cost = (criminal_cost + health_cost + productivity_cost)/total_populati
##  [5] "         per_capita_total_cost = (criminal_cost + health_cost + productivity_cost + death_cost
##  [6] ""
##  [7] "write_csv(o, \"../out/county_data.csv\")"
##  [8] ""
##  [9] "### NON-FATAL COSTS"
## [10] ""
## [11] "map1 <- o %>% select(region, per_capita_nf_cost) %>%"
## [12] "  mutate(value = per_capita_nf_cost) %>%"
## [13] "  select(region, value)"
## [14] ""
## [15] "q <- c(56, 129, 153, 183, 231, 727)"
## [16] ""
## [17] "c = CountyChoropleth$new(map1)"
## [18] "c$title = \"Non-Fatal Opioid Costs Per Capita by County in 2015\""
## [19] "c$set_num_colors(5)"
## [20] "c$set_zoom(state.regions$region[-c(1, 12)])"
## [21] "c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, \"Oranges\"),"
## [22] "                                   name = \"Per Capita Cost\","
## [23] "                                   labels = c(str_c(\"$\", q[1] %>% round, \" to $\", q[2] %>%
## [24] "                                              str_c(\"$\", q[2] %>% round, \" to $\", q[3] %>%
## [25] "                                              str_c(\"$\", q[3] %>% round, \" to $\", q[4] %>%
## [26] "                                              str_c(\"$\", q[4] %>% round, \" to $\", q[5] %>%
## [27] "                                              str_c(\"$\", q[5] %>% round, \" to $\", q[6] %>%
## [28] "                                   na.value = \"black\","
## [29] "                                   drop = F)"
## [30] ""
## [31] "c$render() + "
## [32] "  theme(text=element_text(size=14,  family=\"Times\")) +"
## [33] "  theme(plot.title = element_text(hjust = 0.5),"
## [34] "        plot.caption = element_text(size = 8),"
## [35] "        legend.title = element_text(size = 12),"
## [36] "        legend.text = element_text(size = 10))"
## [37] ""
## [38] "### TOTAL COSTS"
## [39] ""
## [40] "map2 <- o %>% select(region, per_capita_total_cost) %>%"
## [41] "  mutate(value = per_capita_total_cost) %>%"
## [42] "  select(region, value)"
## [43] ""
## [44] "q <- c(160, 824, 1153, 1585, 2232, 8734)"
## [45] ""
## [46] "c = CountyChoropleth$new(map2)"
## [47] "c$title = \"Total Opioid Costs Per Capita by County in 2015\""
## [48] "c$set_num_colors(5)"
## [49] "c$set_zoom(state.regions$region[-c(1, 12)])"
## [50] "c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, \"Oranges\"),"
## [51] "                                   name = \"Per Capita Cost\","
## [52] "                                   na.value = \"black\","
## [53] "                                   labels = c(str_c(\"$\", q[1] %>% round, \" to $\", q[2] %>%
## [54] "                                              str_c(\"$\", q[2] %>% round, \" to $\", q[3] %>%
## [55] "                                              str_c(\"$\", q[3] %>% round, \" to $\", q[4] %>%
## [56] "                                              str_c(\"$\", q[4] %>% round, \" to $\", q[5] %>%
## [57] "                                              str_c(\"$\", q[5] %>% round, \" to $\", q[6] %>%
## [58] "                                   drop = F)"
## [59] ""
## [60] "c$render() + "
## [61] "  theme(text=element_text(size=14,  family=\"Times\")) +"
## [62] "  theme(plot.title = element_text(hjust = 0.5),"
## [63] "        plot.caption = element_text(size = 8),"
## [64] "        legend.title = element_text(size = 12),"
## [65] "        legend.text = element_text(size = 10))"
## [66] ""
## [67] ""
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "make.county.maps"
##
##
## $make.state.maps
##  [1] ""
##  [2] "o2 <- d %>% group_by(state.name) %>%"
##  [3] "  summarise(criminal_cost = sum(criminal_cost),"
##  [4] "         health_cost = sum(health_cost),"
##  [5] "         productivity_cost = sum(productivity_cost),"
##  [6] "         death_cost = sum(death_cost),"
##  [7] "         total_cost = sum(criminal_cost + health_cost + productivity_cost),"
##  [8] "         total_population = sum(total_population),"
##  [9] "         health_cost_pc = sum(health_cost)/sum(total_population),"
## [10] "         productivity_cost_pc = sum(productivity_cost)/sum(total_population),"
## [11] "         death_cost_pc = sum(death_cost)/sum(total_population)) %>%"
## [12] "  ungroup %>%"
## [13] "  mutate(per_capita_nf_cost = (criminal_cost + health_cost + productivity_cost)/total_populati
## [14] "         per_capita_total_cost = (criminal_cost + health_cost + productivity_cost + death_cost
## [15] "         region = state.name)"
## [16] ""
## [17] "write_csv(o2, \"../out/state_data.csv\")"
## [18] ""
## [19] "### NON-FATAL COSTS"
## [20] ""
## [21] "map3 <- o2 %>% select(region, per_capita_nf_cost) %>%"
## [22] "  mutate(value = per_capita_nf_cost) %>%"
## [23] "  select(region, value)"
## [24] ""
## [25] "q <- c(118, 160, 195, 226, 289, 493)"
## [26] ""
## [27] "c = StateChoropleth$new(map3)"
## [28] "c$title = \"Non-Fatal Opioid Costs Per Capita by State in 2015\""
## [29] "c$set_num_colors(5)"
## [30] "c$set_zoom(state.regions$region[-c(1, 12)])"
## [31] "c$show_labels = FALSE"
## [32] "c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, \"Oranges\"),"
## [33] "                                   name = \"Per Capita Cost\","
## [34] "                                   na.value = \"black\","
## [35] "                                   labels = c(str_c(\"$\", q[1] %>% round, \" to $\", q[2] %>%
## [36] "                                              str_c(\"$\", q[2] %>% round, \" to $\", q[3] %>%
## [37] "                                              str_c(\"$\", q[3] %>% round, \" to $\", q[4] %>%
## [38] "                                              str_c(\"$\", q[4] %>% round, \" to $\", q[5] %>%
## [39] "                                              str_c(\"$\", q[5] %>% round, \" to $\", q[6] %>%
## [40] "                                   drop = F)"
## [41] ""
## [42] "c$render() + "
## [43] "  theme(text=element_text(size=14,  family=\"Times\")) +"
## [44] "  theme(plot.title = element_text(hjust = 0.5),"
## [45] "        plot.caption = element_text(size = 8),"
## [46] "        legend.title = element_text(size = 12),"
## [47] "        legend.text = element_text(size = 10))"
## [48] ""
## [49] "### TOTAL COSTS"
## [50] ""
## [51] "map4 <- o2 %>% select(region, per_capita_total_cost) %>%"
## [52] "  mutate(value = per_capita_total_cost) %>%"
## [53] "  select(region, value)"
## [54] ""
## [55] "q <- c(394, 907, 1385, 1827, 2530, 4378)"
## [56] ""
## [57] "c = StateChoropleth$new(map4)"
## [58] "c$title = \"Total Opioid Costs Per Capita by State in 2015\""
## [59] "c$set_num_colors(5)"
## [60] "c$set_zoom(state.regions$region[-c(1, 12)])"
## [61] "c$show_labels = FALSE"
## [62] "c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, \"Oranges\"),"
## [63] "                                   name = \"Per Capita Cost\","
## [64] "                                   labels = c(str_c(\"$\", q[1] %>% round, \" to $\", q[2] %>%
## [65] "                                              str_c(\"$\", q[2] %>% round, \" to $\", q[3] %>%
## [66] "                                              str_c(\"$\", q[3] %>% round, \" to $\", q[4] %>%
## [67] "                                              str_c(\"$\", q[4] %>% round, \" to $\", q[5] %>%
## [68] "                                              str_c(\"$\", q[5] %>% round, \" to $\", q[6] %>%
## [69] "                                   na.value = \"black\","
## [70] "                                   drop = F)"
## [71] ""
## [72] "c$render() + "
## [73] "  theme(text=element_text(size=14,  family=\"Times\")) +"
## [74] "  theme(plot.title = element_text(hjust = 0.5),"
## [75] "        plot.caption = element_text(size = 8),"
## [76] "        legend.title = element_text(size = 12),"
## [77] "        legend.text = element_text(size = 10))"
## [78] ""
## attr(,"chunk_opts")
## attr(,"chunk_opts")$label
## [1] "make.state.maps"
```

Criminal Justice Costs

Criminal justice cost data are available from the Bureau of Justice. We use the 2012 preliminary file for our analysis. The data are at the state level– in order to estimate county-level criminal justice costs associated with the opioid crisis, we weight counties proportionally to the number of individuals who use pain relievers for nonmedical use. This code can be found in the section DEVELOP COUNTY-LEVEL ESTIMATES. Relevant variables include:

- State-level criminal costs per capita.

```
# read in criminal justice costs
criminal_costs <- read_csv("../data/criminal_justice_expenditures.csv",
              skip = 25, col_names = F) %>%
  na.omit %>%
  filter(X2 != "-") %>%
  select(X1, X2, X4) %>%
  mutate(state = X1, criminal_cost_per_capita = X4/(X2 %>% as.numeric * 1000)) %>% # create pre capita
  select(state, criminal_cost_per_capita)
```

```
## Parsed with column specification:
## cols(
##   X1 = col_character(),
##   X2 = col_character(),
##   X3 = col_integer(),
##   X4 = col_integer(),
##   X5 = col_double(),
##   X6 = col_integer(),
##   X7 = col_double(),
##   X8 = col_integer(),
##   X9 = col_double(),
##   X10 = col_integer(),
##   X11 = col_double()
## )
```

```r
# data are at the state level-- merge in corresponding counties
criminal_costs <- county.regions %>%
  select(region, state.name) %>%
  left_join(criminal_costs %>% mutate(state.name = state %>% tolower) %>%
              select(state.name, criminal_cost_per_capita)) %>%
  select(region, criminal_cost_per_capita)
```

```
## Joining, by = "state.name"
```

Hospitalization Costs

Hospitalization cost data come from the Agency for Healthcare Reserach and Quality. The data can be accessed through the "Community" statistics link when creating a custom data table. Our state-level data are for year 2014, and are at the county level. We focus on the "Stays for Alcohol and Other Drugs" diagnosis. Relevant variables include:

- total hospitalization costs: stays for alcohol/other drugs
- total discharges: stays for alcohol/other drugs
- mean costs: stays for alcohol/other drugs

```r
# read in hospitalization cost data-- these are already cleaned with the "process_hospitalizations.R" c
hospitalization_costs <- read_rds("../data/hospitalization_costs.rds")

# county-level data
county_hosp_costs <- hospitalization_costs %>% filter(region_name != "State Total") %>%
  mutate(county.name = region_name %>% tolower %>% str_trim,
         state.name = state_name %>% tolower %>% str_trim) %>%
  inner_join(county.regions) %>%
  select(region, discharges, mean_costs, per_capita_costs, total_costs)
```

```
## Joining, by = c("county.name", "state.name")
```

```r
# state-level data
state_hosp_costs <- hospitalization_costs %>% filter(region_name == "State Total") %>%
  mutate(region = state_name %>% tolower %>% str_trim,
         state_total_costs = total_costs,
         state_discharges = discharges,
         state_mean_costs = mean_costs) %>%
  select(region, state_total_costs, state_discharges, state_mean_costs)
```

County/State Covariates

We derive our county- and state- level income and demographic controls from the American Community Survey. We use data from the 2016 5-year survey. Relevant variables include:

- Population
- Working age population
- Educational attainment
- Urban/rural status
- Race
- Income

```r
# population by county
population <- get_acs(geography = "county",
             variables = c(population = "B01003_001"),
             survey = "acs5",
             year = 2016) %>%
  mutate(region = GEOID %>% as.numeric,
         total_population = estimate) %>%
  select(region, total_population)
```

```
## Getting data from the 2012-2016 5-year ACS
```

```
# working age population by county
workers <- get_acs(geography = "county",
             variables = c(m2534 = "B15001_011", # get males and females aged 25 through 64
                           m3544 = "B15001_019",
                           m4564 = "B15001_027",
                           f2534 = "B15001_052",
                           f3544 = "B15001_060",
                           f4564 = "B15001_068"),
             survey = "acs5",
             year = 2016) %>%
  group_by(GEOID) %>%
  summarise(working_population = sum(estimate, na.rm = T)) %>% #sum population by GEOID (county)
  mutate(region = GEOID %>% as.numeric) %>%
  select(region, working_population) %>% ungroup
```

```
## Getting data from the 2012-2016 5-year ACS
```

```
# educational attainment by county
educational_attainment <- get_acs("county",
                                  variables = c("B06009_001", "B06009_002", "B06009_003"),  year = 2016
  group_by(GEOID) %>%
  summarise(pct_no_hs = estimate[2]/estimate[1],
            pct_hs = estimate[3]/estimate[1],
            pct_college = 1 - pct_no_hs - pct_hs) %>%
  rename(region = GEOID) %>%
  mutate(region = region %>% as.numeric)
```

```
## Getting data from the 2012-2016 5-year ACS
```

```
# Urban/rural status by county
urban_rural <- read_excel("../data/percent_rural.xlsx") %>%
  mutate(region = GEOID %>% as.numeric,
         percent_rural = `2010 Census \r\nPercent Rural`) %>%
  select(region, percent_rural)
```

```
## Warning in function_list[[k]](value): NAs introduced by coercion
```

```
# Racial breakdown by county
race <- get_acs(geography = "county",
             variables = c(total_population = "B03002_001",
                           black_alone_not_hispanic = "B03002_004",
                           hispanic_all_races = "B03002_012"),
             survey = "acs5", year = 2016) %>%
  group_by(GEOID) %>%
  summarise(percent_black = estimate[2]/estimate[1] * 100,
            percent_hispanic = estimate[3]/estimate[1] * 100) %>%
  mutate(region = GEOID %>% as.numeric) %>%
  select(region, percent_black, percent_hispanic)
```

```
## Getting data from the 2012-2016 5-year ACS
```

```r
# med_income <- get_acs(geography = "county",
#                    variables = c(med_income_per_capita = "B06011_001"),
#                    survey = "acs5", year = 2016) %>%
#   mutate(region = GEOID %>% as.numeric,
#          med_income_per_capita = estimate) %>%
#   select(region, med_income_p  er_capita)

# mean_income <- get_acs(geography = "county",
#                    variables = c(income_per_capita = "B19301_001"),
#                    survey = "acs5", year = 2016) %>%
#   mutate(region = GEOID %>% as.numeric,
#          mean_income_per_capita = estimate) %>%
#   select(region, mean_income_per_capita)

# Population by state
st_population <- get_acs(geography = "state",
             variables = c(population = "B01003_001"),
             survey = "acs5",
             year = 2016) %>%
  mutate(region = NAME %>% tolower,
         total_population = estimate) %>%
  select(region, total_population)
```

```
## Getting data from the 2012-2016 5-year ACS
```

```r
# Workers by state
st_workers <- get_acs(geography = "state",
             variables = c(m2534 = "B15001_011",
                           m3544 = "B15001_019",
                           m4564 = "B15001_027",
                           f2534 = "B15001_052",
                           f3544 = "B15001_060",
                           f4564 = "B15001_068"),
             survey = "acs5",
             year = 2016) %>%
  group_by(GEOID) %>%
  summarise(working_population = sum(estimate, na.rm = T)) %>%
  rename(region = GEOID) %>% ungroup
```

```
## Getting data from the 2012-2016 5-year ACS
```

```r
# st_income <- get_acs(geography = "state",
#                    variables = c(income_per_capita = "B06011_001"),
#                    survey = "acs5", year = 2016) %>%
#   mutate(region = NAME %>% tolower,
#          income_per_capita = estimate) %>%
#   select(region, income_per_capita)

# Income by county
income <- read_excel("../data/est16all.xls", skip = 3) %>% select(`State FIPS Code`, `County FIPS Code`

# Income by state
med_income <- income %>% mutate(med_income = `Median Household Income` %>% as.numeric,
                                region = str_c(`State FIPS Code`, `County FIPS Code`) %>% as.numeric) %>
  right_join(county.regions) %>% select(region, med_income)
```

```
## Warning in function_list[[k]](value): NAs introduced by coercion
```

```
## Joining, by = "region"
```

```
# County fips code 000 corresponds to state-level stats
st_income <- income %>% filter(`County FIPS Code` == "000") %>%
  mutate(st_med_income = `Median Household Income` %>% as.numeric,
         fips.character = `State FIPS Code`) %>%
  right_join(state.regions) %>%
  select(region, st_med_income)
```

```
## Joining, by = "fips.character"
```

```
# merge county-level demographic data
demographics <- county.regions %>%
  select(region) %>%
  left_join(population) %>%
  left_join(workers) %>%
  left_join(race) %>%
  left_join(med_income) %>%
  left_join(urban_rural) %>%
  left_join(educational_attainment) %>% tbl_df
```

```
## Joining, by = "region"
```

```
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
```

```
# merge state-level demographic data
st_demographics <- state.regions %>%
  select(region, fips.character) %>%
  left_join(st_income) %>%
  left_join(st_population) %>%
  left_join(st_workers)
```

```
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
```

Constructing State and County Weights

Health Cost Model

In order to estimate the geographic variation in the hospitalization costs associated with the opioid crisis, we calculate state- and county-level weights to distribute the CEA (2017) aggregate cost estimate of $36.6 billion. The original hospitalization cost data are at the state level– we obseve data for 32 states with in the US. We predict opioid-related hospitalization costs for the remaining 18 states and the District of Columbia using a linear regression model. We estimate the following model:

$$Y = X\beta + \epsilon$$

Response	Regressors
Log of Hospitalization Costs	Log of Total Population

Response	Regressors
	Log of Opioid Deaths Critical Care Fee Schedule Log of Median Per-Capita Income

```
# County-level hospitalization costs
county_hosp_costs <- hospitalization_costs %>% filter(region_name != "State Total") %>%
  mutate(county.name = region_name %>% tolower %>% str_trim,
         state.name = state_name %>% tolower %>% str_trim) %>%
  right_join(county.regions)
```

```
## Joining, by = c("county.name", "state.name")
```

```
# State-level hospitalization costs
state_hosp_costs <- state_hosp_costs %>% right_join(state.regions) %>% rename(state.name = region)
```

```
## Joining, by = "region"
```

```
# Merge data NOTE: do we need this? I can't find it referenced later in the code
# hosp_costs <- county_hosp_costs %>% left_join(state_hosp_costs) %>% left_join(demographics) %>%
#   left_join(st_demographics %>% rename(state.name = region, total_st_population = total_population))

# data for county-level hospitalization cost regressions
hosp_costs_model <- county_hosp_costs %>%
  left_join(medical_cost_index) %>%
  left_join(nonmedical_use) %>%
  left_join(demographics) %>%
  left_join(opioid_deaths) %>%
  left_join(drug_deaths) %>%
  mutate(log_total_hosp_costs = log(total_costs),
         log_population = log(total_population),
         log_nonmedical_use_count = log(total_population/1000 * nonmedical_use_pct),
         log_opioid_deaths = log(opioid_deaths),
         log_drug_deaths_mcd = log(drug_deaths_mcd))
```

```
## Joining, by = "region"
```

```
## Joining, by = c("region", "state.fips.character")
```

```
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
```

```
# mixed-effects linear regression with state random effects--counties with opioid-related deaths
h1 <- lmer(log_total_hosp_costs ~ medical_cost + log_nonmedical_use_count + log_population + log_opioid_

# adjustment factor NEED TO ASK ABOUT THIS
h1_adj = sum(exp(resid(h1)))/(length(resid(h1)) - 7)

# mixed-effects linear regression with state random effects--counties with only drug deaths
h2 <- lmer(log_total_hosp_costs ~ medical_cost + log_nonmedical_use_count + log_population + log_drug_de

h2_adj = sum(exp(resid(h2)))/(length(resid(h2)) - 6)

# mixed-effects linear regression with state random effects--counties without opioid/general drug death
h3 <- lmer(log_total_hosp_costs ~ medical_cost + log_nonmedical_use_count + log_population + (1|state.fi
```

```r
h3_adj = sum(exp(resid(h3)))/(length(resid(h3)) - 5)

### State-level hospitalization cost data
# State-level opioid death data-- for use in linear regression
state_opioid_deaths <- read_tsv("../data/State Opioid Deaths 2016 - MCD.txt") %>%
  select(-Notes) %>%
  filter(!is.na(State)) %>%
  mutate(state.fips.character = `State Code`,
         state_opioid_deaths = Deaths) %>%
  select(state.fips.character, state_opioid_deaths)
```

```
## Parsed with column specification:
## cols(
##   Notes = col_character(),
##   State = col_character(),
##   `State Code` = col_character(),
##   Deaths = col_integer(),
##   Population = col_integer(),
##   `Crude Rate` = col_double()
## )

## Warning in rbind(names(probs), probs_f): number of columns of result is not
## a multiple of vector length (arg 1)

## Warning: 59 parsing failures.
## row # A tibble: 5 x 5 col     row col   expected  actual    file
## ... ................. ... .........................................................................
## See problems(...) for more details.
```

```r
# opioid medical cost data -- outcome of interest in regression
state_medical_costs <- medical_cost_index %>% left_join(demographics) %>%
  left_join(county.regions %>% select(region, state.fips.character)) %>%
  group_by(state.fips.character) %>%
  summarise(medical_cost = sum(total_population/sum(total_population, na.rm = T) * medical_cost, na.rm =
```

```
## Joining, by = "region"
## Joining, by = "region"
```

```r
# data for state-level regressions
state_hosp_costs_model <- state_hosp_costs %>%
  left_join(st_demographics) %>%
  left_join(state_opioid_deaths %>% rename(fips.character = state.fips.character)) %>%
  left_join(state_medical_costs %>% rename(fips.character = state.fips.character)) %>%
  mutate(log_hospital_costs = log(state_total_costs),
         log_total_population = log(total_population),
         log_mean_cost = log(state_mean_costs),
         log_opioid_deaths = log(state_opioid_deaths),
         log_income = log(st_med_income))
```

```
## Joining, by = "fips.character"

## Joining, by = "fips.character"
## Joining, by = "fips.character"
```

```r
# linear regression: state hospitalization costs
sh1 <- lm(log_hospital_costs ~ log_total_population +
            log_opioid_deaths + medical_cost + log_income, data = state_hosp_costs_model)
```

```
# This is correcting non-normality of the residuals?
sh1_adj <- sum(exp(resid(sh1)))/26

# add predicted values for state hospitalization costs, generate per-capita hospitalization costs
# this code fills in predicted values if state costs are missing, otherwise it returns original state c
state_hosp_costs <- state_hosp_costs_model %>%
  mutate(state_est_hosp_costs = ifelse(is.na(state_total_costs), exp(predict(sh1, state_hosp_costs_model
         state_costs_per_capita = state_est_hosp_costs/total_population) %>%
  select(fips.character, state_est_hosp_costs, state_costs_per_capita)

### county-level hospitalization costs
# for each county, get predicted hospitalization costs based on which independent variables are availab
d1_hosp <- hosp_costs_model %>% filter(!is.na(medical_cost + log_nonmedical_use_count + log_population +

d1_hosp <- d1_hosp %>% mutate(predicted_hospitalization_costs1 = exp(predict(h1, d1_hosp, allow.new.leve

d2_hosp <- hosp_costs_model %>% filter(!is.na(medical_cost + log_nonmedical_use_count + log_population +

d2_hosp <- d2_hosp %>% mutate(predicted_hospitalization_costs2 = exp(predict(h2, d2_hosp, allow.new.leve

d3_hosp <- hosp_costs_model %>% filter(!is.na(medical_cost + log_nonmedical_use_count + log_population))

d3_hosp <- d3_hosp %>% mutate(predicted_hospitalization_costs3 = exp(predict(h3, d3_hosp, allow.new.leve

# assign county-level hospital costs according to hierarchy of data available: if the real data are ava
proj_hospital_costs <- hosp_costs_model %>% select(region, state.fips.character, total_costs) %>%
  left_join(state_hosp_costs %>% rename(state.fips.character = fips.character)) %>%
  left_join(d1_hosp %>% select(region, predicted_hospitalization_costs1)) %>%
  left_join(d2_hosp %>% select(region, predicted_hospitalization_costs2)) %>%
  left_join(d3_hosp %>% select(region, predicted_hospitalization_costs3)) %>%
   mutate(est_total_hosp_costs = case_when(
    !is.na(total_costs) ~ total_costs,
    !is.na(predicted_hospitalization_costs1) ~ predicted_hospitalization_costs1,
    !is.na(predicted_hospitalization_costs2) ~ predicted_hospitalization_costs2,
    !is.na(predicted_hospitalization_costs3) ~ predicted_hospitalization_costs3)) %>%
  group_by(state.fips.character) %>%
  mutate(total_est_costs = sum(est_total_hosp_costs, na.rm = T)) %>%
  mutate(infl_factor = state_est_hosp_costs/total_est_costs,
         est_total_hosp_costs = est_total_hosp_costs * infl_factor) %>% ungroup
```

```
## Joining, by = "state.fips.character"
```

```
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
```

BUILD OPIOID DEATH MODEL

```
d <- county.regions %>% tbl_df %>% dplyr::select(region, state.fips.character) %>%
  left_join(opioid_deaths) %>%
  left_join(drug_deaths) %>%
  left_join(nonmedical_use) %>%
```

```r
  left_join(demographics) %>%
  left_join(medical_cost_index) %>%
  left_join(county_hosp_costs) %>%
  mutate(nonmedical_use_count = nonmedical_use_pct * total_population/1000,
         drug_deaths_mcd = drug_deaths_mcd,
         opioid_deaths = opioid_deaths %>% as.numeric) %>%
  mutate(log_drug_deaths_mcd = log(drug_deaths_mcd),
         log_nonmedical_use_count = log(nonmedical_use_count),
         log_population = log(total_population),
         log_income = log(med_income),
         log_med_income = log(med_income),
         log_opioid_deaths = log(opioid_deaths))
```

```
## Joining, by = "region"
## Joining, by = "region"

## Joining, by = c("region", "state.fips.character")

## Joining, by = "region"
## Joining, by = "region"

## Joining, by = c("region", "state.fips.character")
```

```r
m1 <- glmmTMB(opioid_deaths ~ log_drug_deaths_mcd + log_nonmedical_use_count + log_population + log_med_
               percent_hispanic + pct_no_hs +
                (1|state.fips.character), data = d, family = nbinom2(link = "log"), verbose = F)

m2 <- glmmTMB(opioid_deaths ~ log_nonmedical_use_count + log_population + log_med_income + percent_rural
               percent_hispanic + pct_no_hs +
                (1|state.fips.character), data = d, family = nbinom2(link = "log"), verbose = F)

d1_deaths <- d %>% filter(!is.na(log_drug_deaths_mcd + log_nonmedical_use_count + log_population + log_m
               percent_hispanic + pct_no_hs))

d1_deaths <- d1_deaths %>% mutate(predicted_opioid_deaths1 = predict(m1, d1_deaths, allow.new.levels=TRU

d2_deaths <- d %>% filter(!is.na(log_nonmedical_use_count + log_population + log_med_income + percent_ru
               percent_hispanic + pct_no_hs))

d2_deaths <- d2_deaths %>% mutate(predicted_opioid_deaths2 = predict(m2, d2_deaths, allow.new.levels=TRU

proj_opioid_deaths <- d %>% select(region, opioid_deaths, total_population) %>%
  left_join(d1_deaths %>% select(region, predicted_opioid_deaths1)) %>%
  left_join(d2_deaths %>% select(region, predicted_opioid_deaths2)) %>%
  select(region, total_population, opioid_deaths, predicted_opioid_deaths1, predicted_opioid_deaths2)
```

```
## Joining, by = "region"

## Joining, by = "region"
```

```r
proj_opioid_deaths <- county.regions %>% select(region, state.fips.character, state.abb) %>%
  left_join(proj_opioid_deaths) %>%
  left_join(state_opioid_deaths) %>%
  filter(!is.na(total_population))
```

```
## Joining, by = "region"

## Joining, by = "state.fips.character"
```

```
proj_opioid_deaths <- proj_opioid_deaths %>%
  mutate(est_opioid_deaths = case_when(
    !is.na(opioid_deaths) ~ opioid_deaths,
    !is.na(predicted_opioid_deaths1) ~ predicted_opioid_deaths1,
    !is.na(predicted_opioid_deaths2) ~ predicted_opioid_deaths2)
  ) %>%
  group_by(state.fips.character) %>%
  mutate(est_state_opioid_deaths = sum(est_opioid_deaths, na.rm = T),
         infl_factor = state_opioid_deaths/est_state_opioid_deaths,
         est_opioid_deaths = est_opioid_deaths * infl_factor) %>% ungroup

projection_summary1 <- proj_opioid_deaths %>%
 select(est_opioid_deaths, opioid_deaths, total_population, state.fips.character) %>%
 mutate(modeled = ifelse(is.na(opioid_deaths), 1, 0)) %>%
 group_by(state.fips.character) %>%
 summarise(opioid_deaths = 100 * sum(est_opioid_deaths, na.rm = T)/sum(total_population, na.rm = T),
           modeled_pct = sum(total_population * modeled, na.rm = T)/sum(total_population, na.rm = T))

projection_summary2 <- proj_hospital_costs %>% left_join(demographics) %>%
 select(total_costs, est_total_hosp_costs, total_population, state.fips.character) %>%
 mutate(modeled = ifelse(is.na(total_costs), 1, 0)) %>%
 group_by(state.fips.character) %>%
 summarise(per_capita_costs = sum(est_total_hosp_costs, na.rm = T)/sum(total_population, na.rm = T),
           modeled_pct = sum(total_population * modeled, na.rm = T)/sum(total_population, na.rm = T))
```

```
## Joining, by = "region"
```

DEVELOP COUNTY-LEVEL ESTIMATES

```
fatal_costs <- 431.7
total_nonfatal_costs <- 72.3
nonfatal_health_costs <- total_nonfatal_costs * 0.5068966
total_criminal_justice_costs <- total_nonfatal_costs * 0.1344828
total_productivity_costs <- total_nonfatal_costs * 0.3586207


d <- county.regions %>% tbl_df %>%
  left_join(proj_opioid_deaths %>% select(region, est_opioid_deaths)) %>%
  left_join(proj_hospital_costs %>% select(region, est_total_hosp_costs)) %>%
  left_join(nonmedical_use) %>%
  left_join(medical_cost_index) %>%
  left_join(criminal_costs) %>%
  left_join(demographics)
```

```
## Joining, by = "region"
## Joining, by = "region"
```

```
## Joining, by = c("region", "state.fips.character")
```

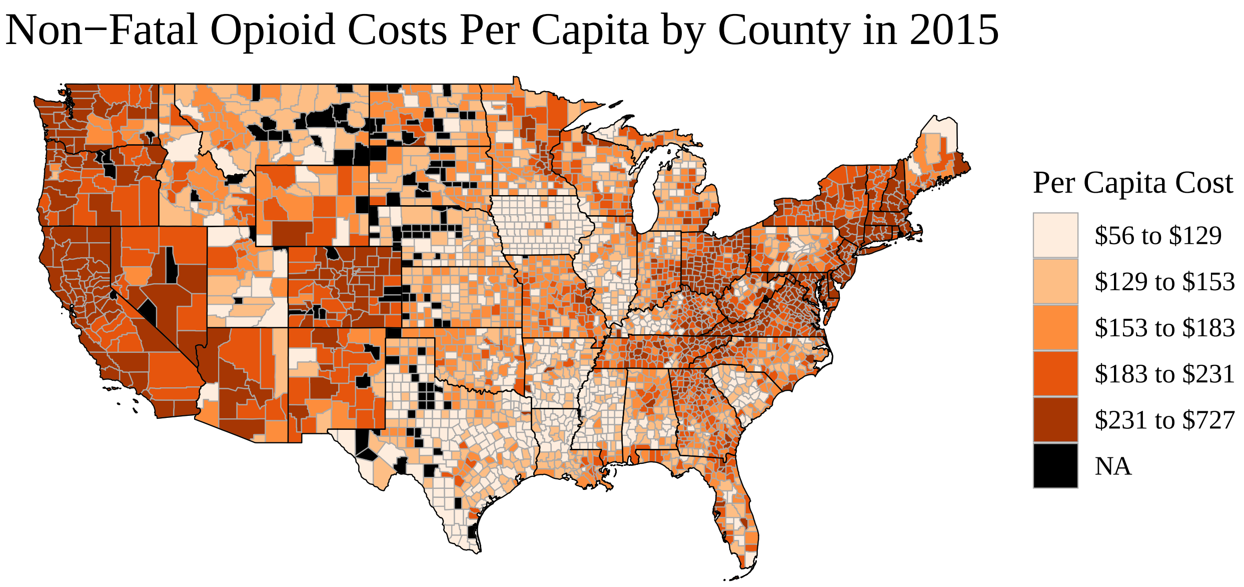
```
## Joining, by = "region"
## Joining, by = "region"
## Joining, by = "region"
```

```
d <- d %>% filter(!is.na(total_population), !is.na(med_income)) %>%
  mutate(death_wt = est_opioid_deaths/sum(est_opioid_deaths, na.rm = T),
         death_cost = death_wt * fatal_costs * 10^9,
         death_cost_pc = death_cost/total_population) %>%
  mutate(health_wt = est_total_hosp_costs/sum(est_total_hosp_costs),
         health_cost = health_wt * nonfatal_health_costs * 10^9,
         health_cost_pc = health_cost/total_population) %>%
  mutate(criminal_wt = (nonmedical_use_pct * total_population * criminal_cost_per_capita)/sum(nonmedical
         criminal_cost = criminal_wt * total_criminal_justice_costs * 10^9,
         criminal_cost_pc = criminal_cost/total_population) %>%
  mutate(productivity_wt = (med_income * 0.175 * nonmedical_use_pct * working_population)/sum(med_income
         productivity_cost = productivity_wt * total_productivity_costs * 10^9,
         productivity_cost_pc = productivity_cost/total_population)
```

MAKE MAPS FOR COUNTIES

```
o <- d %>%
  filter(total_population > 2500, !(state.name %in% c("alaska", "hawaii"))) %>%
  mutate(per_capita_nf_cost = (criminal_cost + health_cost + productivity_cost)/total_population,
         per_capita_total_cost = (criminal_cost + health_cost + productivity_cost + death_cost)/total_po

write_csv(o, "../out/county_data.csv")

### NON-FATAL COSTS

map1 <- o %>% select(region, per_capita_nf_cost) %>%
  mutate(value = per_capita_nf_cost) %>%
  select(region, value)

q <- c(56, 129, 153, 183, 231, 727)

c = CountyChoropleth$new(map1)
c$title = "Non-Fatal Opioid Costs Per Capita by County in 2015"
c$set_num_colors(5)
c$set_zoom(state.regions$region[-c(1, 12)])
c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, "Oranges"),
                                   name = "Per Capita Cost",
                                   labels = c(str_c("$", q[1] %>% round, " to $", q[2] %>% round),
                                              str_c("$", q[2] %>% round, " to $", q[3] %>% round),
                                              str_c("$", q[3] %>% round, " to $", q[4] %>% round),
                                              str_c("$", q[4] %>% round, " to $", q[5] %>% round),
                                              str_c("$", q[5] %>% round, " to $", q[6] %>% round)),
                                   na.value = "black",
                                   drop = F)

c$render() +
  theme(text=element_text(size=14,  family="Times")) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(size = 8),
        legend.title = element_text(size = 12),
        legend.text = element_text(size = 10))
```

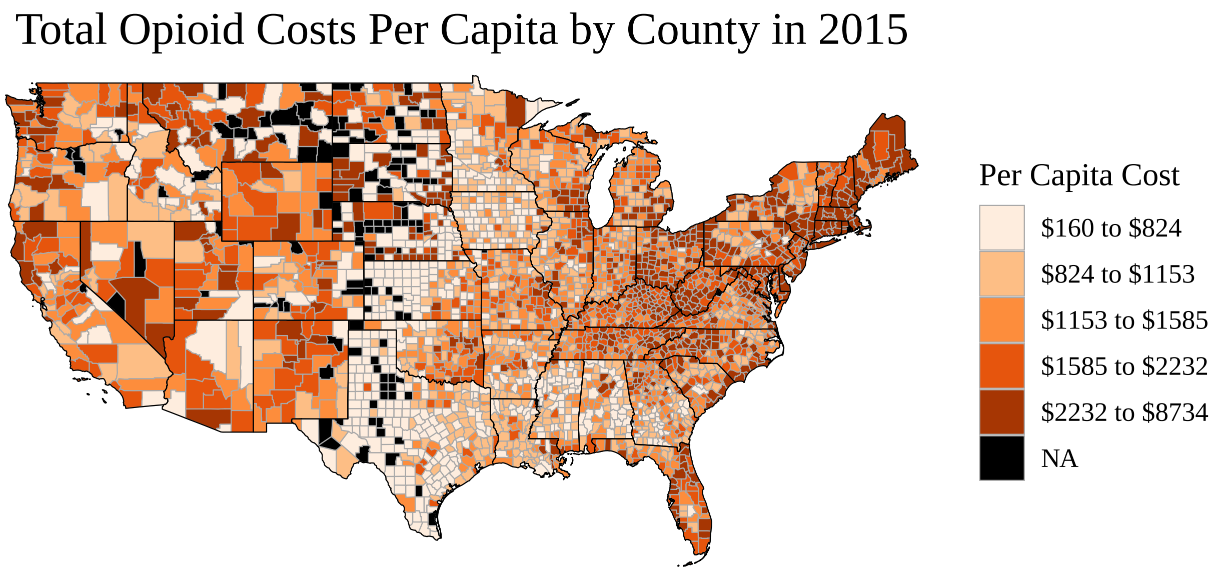
```
## Warning in self$bind(): The following regions were missing and are being
## set to NA: 13239, 31149, 40025, 48045, 30055, 38027, 38037, 38039, 38043,
## 38047, 38065, 48431, 30109, 31007, 31057, 31071, 31075, 35021, 31115,
## 41021, 48447, 49009, 48033, 46063, 46069, 46075, 46097, 46113, 30019,
## 20187, 21201, 13265, 48301, 48327, 53023, 51091, 38013, 38087, 38095,
## 49033, 20203, 8079, 46049, 46107, 46119, 32011, 35011, 30075, 31005, 31069,
## 31077, 48173, 48269, 8033, 51515, 20025, 20033, 20101, 48101, 48109, 48125,
## 48345, 56027, 48235, 48243, 8115, 28055, 38023, 38033, 31165, 30079, 30103,
## 41055, 48443, 31015, 46089, 31049, 46021, 29227, 30011, 30033, 30045,
## 31091, 31103, 38001, 26083, 48311, 48261, 48155, 31113, 46073, 38083,
## 20199, 20071, 20083, 30037, 31171, 31183, 32009, 30051, 30059, 30069,
## 30107, 31009, 31073, 31085, 31117, 38007, 38091, 46017, 46055, 46095,
## 46111, 41069, 48359, 48393, 48433, 48011, 8053, 48137, 8057, 48263, 8111,
## 49031, 6003, 8017, 8061, 16025, 16033
```



```
### TOTAL COSTS

map2 <- o %>% select(region, per_capita_total_cost) %>%
  mutate(value = per_capita_total_cost) %>%
  select(region, value)

q <- c(160, 824, 1153, 1585, 2232, 8734)

c = CountyChoropleth$new(map2)
c$title = "Total Opioid Costs Per Capita by County in 2015"
c$set_num_colors(5)
c$set_zoom(state.regions$region[-c(1, 12)])
```

```r
c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, "Oranges"),
                                   name = "Per Capita Cost",
                                   na.value = "black",
                                   labels = c(str_c("$", q[1] %>% round, " to $", q[2] %>% round),
                                              str_c("$", q[2] %>% round, " to $", q[3] %>% round),
                                              str_c("$", q[3] %>% round, " to $", q[4] %>% round),
                                              str_c("$", q[4] %>% round, " to $", q[5] %>% round),
                                              str_c("$", q[5] %>% round, " to $", q[6] %>% round)),
                                   drop = F)

c$render() +
  theme(text=element_text(size=14,  family="Times")) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(size = 8),
        legend.title = element_text(size = 12),
        legend.text = element_text(size = 10))
```

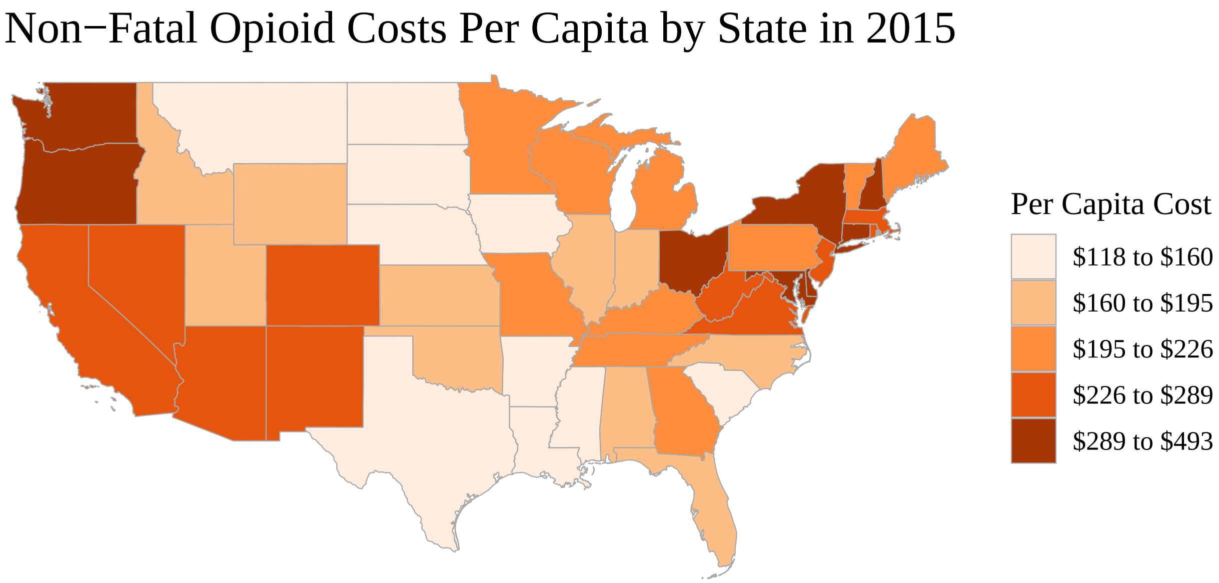
```
## Warning in self$bind(): The following regions were missing and are being
## set to NA: 13239, 31149, 40025, 48045, 30055, 38027, 38037, 38039, 38043,
## 38047, 38065, 48431, 30109, 31007, 31057, 31071, 31075, 35021, 31115,
## 41021, 48447, 49009, 48033, 46063, 46069, 46075, 46097, 46113, 30019,
## 20187, 21201, 13265, 48301, 48327, 53023, 51091, 38013, 38087, 38095,
## 49033, 20203, 8079, 46049, 46107, 46119, 32011, 35011, 30075, 31005, 31069,
## 31077, 48173, 48269, 8033, 51515, 20025, 20033, 20101, 48101, 48109, 48125,
## 48345, 56027, 48235, 48243, 8115, 28055, 38023, 38033, 31165, 30079, 30103,
## 41055, 48443, 31015, 46089, 31049, 46021, 29227, 30011, 30033, 30045,
## 31091, 31103, 38001, 26083, 48311, 48261, 48155, 31113, 46073, 38083,
## 20199, 20071, 20083, 30037, 31171, 31183, 32009, 30051, 30059, 30069,
## 30107, 31009, 31073, 31085, 31117, 38007, 38091, 46017, 46055, 46095,
## 46111, 41069, 48359, 48393, 48433, 48011, 8053, 48137, 8057, 48263, 8111,
## 49031, 6003, 8017, 8061, 16025, 16033
```



MAKE MAPS FOR STATES

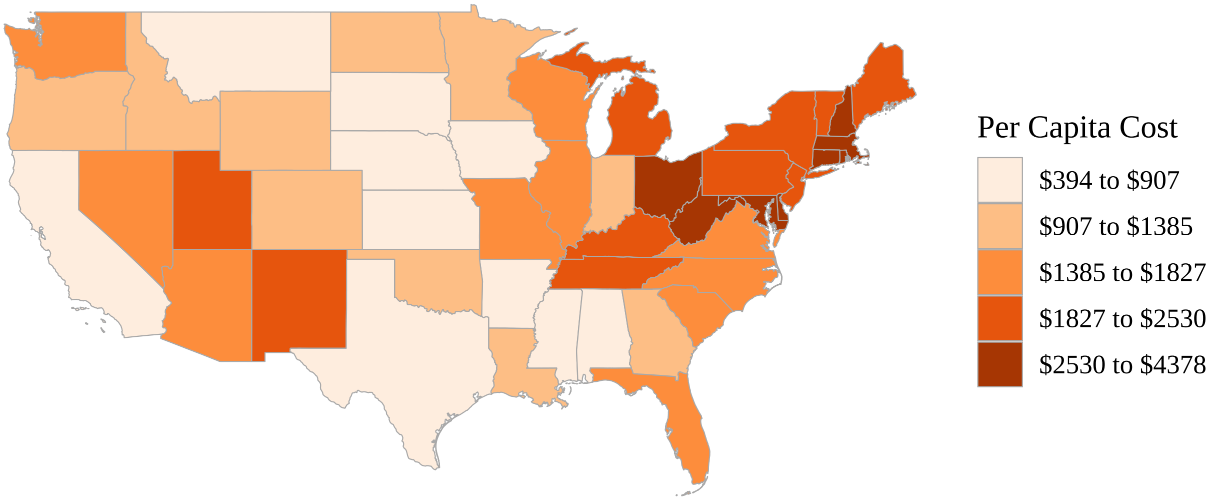
```
o2 <- d %>% group_by(state.name) %>%
  summarise(criminal_cost = sum(criminal_cost),
         health_cost = sum(health_cost),
         productivity_cost = sum(productivity_cost),
         death_cost = sum(death_cost),
         total_cost = sum(criminal_cost + health_cost + productivity_cost),
         total_population = sum(total_population),
         health_cost_pc = sum(health_cost)/sum(total_population),
         productivity_cost_pc = sum(productivity_cost)/sum(total_population),
         death_cost_pc = sum(death_cost)/sum(total_population)) %>%
  ungroup %>%
  mutate(per_capita_nf_cost = (criminal_cost + health_cost + productivity_cost)/total_population,
         per_capita_total_cost = (criminal_cost + health_cost + productivity_cost + death_cost)/total_p
         region = state.name)

write_csv(o2, "../out/state_data.csv")

### NON-FATAL COSTS

map3 <- o2 %>% select(region, per_capita_nf_cost) %>%
  mutate(value = per_capita_nf_cost) %>%
  select(region, value)
```

```
q <- c(118, 160, 195, 226, 289, 493)

c = StateChoropleth$new(map3)
c$title = "Non-Fatal Opioid Costs Per Capita by State in 2015"
c$set_num_colors(5)
c$set_zoom(state.regions$region[-c(1, 12)])
c$show_labels = FALSE
c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, "Oranges"),
                                   name = "Per Capita Cost",
                                   na.value = "black",
                                   labels = c(str_c("$", q[1] %>% round, " to $", q[2] %>% round),
                                              str_c("$", q[2] %>% round, " to $", q[3] %>% round),
                                              str_c("$", q[3] %>% round, " to $", q[4] %>% round),
                                              str_c("$", q[4] %>% round, " to $", q[5] %>% round),
                                              str_c("$", q[5] %>% round, " to $", q[6] %>% round)),
                                   drop = F)

c$render() +
  theme(text=element_text(size=14,  family="Times")) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(size = 8),
        legend.title = element_text(size = 12),
        legend.text = element_text(size = 10))
```



```
### TOTAL COSTS

map4 <- o2 %>% select(region, per_capita_total_cost) %>%
  mutate(value = per_capita_total_cost) %>%
  select(region, value)

q <- c(394, 907, 1385, 1827, 2530, 4378)

c = StateChoropleth$new(map4)
c$title = "Total Opioid Costs Per Capita by State in 2015"
c$set_num_colors(5)
c$set_zoom(state.regions$region[-c(1, 12)])
c$show_labels = FALSE
c$ggplot_scale = scale_fill_manual(values = brewer.pal(5, "Oranges"),
                                   name = "Per Capita Cost",
                                   labels = c(str_c("$", q[1] %>% round, " to $", q[2] %>% round),
                                              str_c("$", q[2] %>% round, " to $", q[3] %>% round),
                                              str_c("$", q[3] %>% round, " to $", q[4] %>% round),
                                              str_c("$", q[4] %>% round, " to $", q[5] %>% round),
                                              str_c("$", q[5] %>% round, " to $", q[6] %>% round)),
                                   na.value = "black",
                                   drop = F)

c$render() +
  theme(text=element_text(size=14,  family="Times")) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.caption = element_text(size = 8),
        legend.title = element_text(size = 12),
        legend.text = element_text(size = 10))
```

Total Opioid Costs Per Capita by State in 2015


Per Capita Cost
$394 to $907
$907 to $1385
$1385 to $1827
$1827 to $2530
$2530 to $4378