Opioid Overdose Problems in United States: Insights from Prescribing and Overdose Death Rates

William Qian

2023-10-20

Contents

Introduction

The opioid crisis has been a pressing public health concern in the United States for the past few decades, leaving an indelible mark on countless families and communities. Between 2015 and 2021, the country witnessed significant shifts in opioid use patterns, influenced by legislative changes, public awareness campaigns, and the evolving nature of drug trafficking. The impact of these shifts is most palpably seen in two critical metrics: opioid prescribing rates and overdose death rates. This report will provide a comprehensive examination of opioid use in the U.S. during this seven-year span. Through an in-depth analysis, we aim to furnish valuable insights into the nature of the crisis, exploring the hidden relationship between related factors.

Questions Answered by this Analysis

To make the report more easily to read, I will make all the questions listed here, and of course, there will be a link to the part of that question in the report.

1. **What kind of opioids is mainly responsible for drug overdose deaths?**
2. **Which states holds the highest number of drug overdose deaths?**
3. **Which state's opioid overdose problem is the most serious?**
4. **What if people want to get rid of opioid?**

5. **Are those opioid treatment programs effective?**
6. **What can we do in order to decrease opioid death rate?**

Librarys used in this Analysis

Package	Description
tidyverse	The 'tidyverse' is a set of packages that work in harmony because they share common data representations and 'API' design.
ggplot2	A system for 'declaratively' creating graphics, based on "The Grammar of Graphics". You provide the data, tell 'ggplot2' how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details.
gtsummary	Creates presentation-ready tables summarizing data sets, regression models, and more. The code to create the tables is concise and highly customizable. Data frames can be summarized with any function, e.g. mean(), median(), even user-written functions. Regression models are summarized and include the reference rows for categorical variables.
ggmap	A collection of functions to visualize spatial data and models on top of static maps from various online sources [e.g Google Maps and Stamen Maps]. It includes tools common to those tasks, including functions for geolocation and routing.
maps	Draw Geographical Maps
usmap	Obtain United States map data frames of varying region types (e.g. county, state). The map data frames include Alaska and Hawaii conveniently placed to the bottom left, as they appear in most maps of the US. Convenience functions for plotting choropleths and working with FIPS codes are also provided.
GGally	The R package 'ggplot2' is a plotting system based on the grammar of graphics. 'GGally' extends 'ggplot2' by adding several functions to reduce the complexity of combining geometric objects with transformed data. Some of these functions include a pairwise plot matrix, a two group pairwise plot matrix, a parallel coordinates plot, a survival plot, and several functions to plot networks.
tidycensus	An integrated R interface to several United States Census Bureau APIs and the US Census Bureau's geographic boundary files. Allows R users to return Census and ACS data as tidyverse-ready data frames, and optionally returns a list-column with feature geometry for mapping and spatial analysis.

```
library("tidyverse")
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library("ggplot2")
library("gtsummary")
```

```
## #BlackLivesMatter
```

```r
library("ggmap")
```

```
## The legacy packages maptools, rgdal, and rgeos, underpinning the sp package,
## which was just loaded, were retired in October 2023.
## Please refer to R-spatial evolution reports for details, especially
## https://r-spatial.org/r/2023/05/15/evolution4.html.
## It may be desirable to make the sf package available;
## package maintainers should consider adding sf to Suggests:.
## i Google's Terms of Service: <https://mapsplatform.google.com>
## i Please cite ggmap if you use it! Use `citation("ggmap")` for details.
```

```r
library("maps")
```

```
##
## Attaching package: 'maps'
##
## The following object is masked from 'package:purrr':
##
##     map
```

```r
library("usmap")
library("GGally")
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
library("tidycensus")
```

```
## Warning: package 'tidycensus' was built under R version 4.3.1
```

```r
library("knitr")
library("scales")
```

```
##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
census_api_key("5663548f2a81838e1d10729754708ca96f2f9a53", install = TRUE, overwrite = TRUE)
```

```
## Your original .Renviron will be backed up and stored in your R HOME directory if needed.
## Your API key has been stored in your .Renviron and can be accessed by Sys.getenv("CENSUS_API_KEY").
## To use now, restart R or run `readRenviron("~/.Renviron")`
```

```
## [1] "5663548f2a81838e1d10729754708ca96f2f9a53"
```

A Glance at Data

Medicaid Opioid Prescribing Rates - by Geography

The Medicaid Opioid Prescribing Rates by Geography dataset provides information on state comparisons of the number and percentage of Medicaid opioid prescriptions.

- **Fee-for-Service (FFS)**:
  In a Fee-for-Service model, healthcare providers are paid for each service or procedure provided to a patient. Each service or procedure is billed separately, and the cost is determined by the type and number of services provided. This model can potentially incentivize providers to offer more services to receive more payment, even if those services may not be necessary for the patient's care.
- **Managed Care (MC):**
  Managed Care models aim to provide better coordinated care by utilizing a network of healthcare providers to deliver services to patients at a lower cost. In this model, healthcare providers are usually paid a fixed amount per patient (per member per month) regardless of the number of services provided. Managed Care organizations often focus on preventive care and coordination of services to keep individuals healthy and to manage the costs of care.

Variable Name	Definition
Year	Identifies the data year.
Geo_Lvl	Identifies the level of geography that the data in the row has been aggregated. A value of "National" indicates the data in the row is aggregated across all states and the District of Columbia. A value of "State" indicates the data in the row is aggregated to the state of the beneficiary.
Geo_Cd	For the state-level data, the state FIPS code that is associated with state of the beneficiary. Restrictions: States are restricted to the 50 U.S. States and the District of Columbia.
Geo_Desc	Data aggregated at the National level are identified by "National". Data aggregated at the State level list the state associated with the beneficiary. The values include the 50 United States and the District of Columbia.
Plan_Type	Identifies the plan type of the data in the row. A value of "All" indicates the data in the row includes both Managed Care and Fee-for-Service claims. A value of "FFS" indicates the data in the row is for a Fee-for-Service claim. A value of "MC" indicates the data in the row is for a Managed Care claim.
Tot_Opioid_Clms	The number of opioid prescriptions include any opioid prescription for which Medicaid paid a portion of the claim, as well as those opioid prescriptions for which Medicaid paid the claim in full.
Tot_Clms	The number of prescriptions include any prescription for which Medicaid paid a portion of the claim, as well as those prescriptions for which Medicaid paid the claim in full.
Opioid_Prscrbng_Rate	The number of Opioid Claims divided by the Overall Claims and multiplied by 100.
Opioid_Prscrbng_Rate_5Y_Chg	The percentage point difference in the rate from five years previous to the data year, which is calculated by subtracting the rate five years previous from the rate in the data year. The change in the prescribing rate is displayed as an increase, decrease, or no change. An increase reflects a percentage point difference of at least 0.10 and a decrease reflects a difference of at least -0.10.
Opioid_Prscrbng_Rate_1Y_Chg	The percentage point difference in the rate from one year previous to the data year, which is calculated by subtracting the rate one year previous from the rate in the data year. The change in the prescribing rate is displayed as an increase, decrease, or no change. An increase reflects a percentage point difference of at least 0.10 and a decrease reflects a difference of at least -0.10.
LA_Tot_Opioid_Clms	The number of long-acting opioid prescriptions include any long-acting opioid prescription for which Medicaid paid a portion of the claim, as well as those long-acting opioid prescriptions for which Medicaid paid the claim in full.
LA_Opioid_Prscrbng_Rate	The number of Long-Acting Opioid Claims divided by the Opioid Claims and multiplied by 100.

Variable Name	Definition
LA_Opioid_Prscrbng_Rate_5Y_Chg	The percentage point difference in the rate from five years previous to the data year, which is calculated by subtracting the rate five years previous from the rate in the data year. The change in the prescribing rate is displayed as an increase, decrease, or no change. An increase reflects a percentage point difference of at least 0.10 and a decrease reflects a difference of at least -0.10.
LA_Opioid_Prscrbng_Rate_1Y_Chg	The percentage point difference in the rate from one year previous to the data year, which is calculated by subtracting the rate one year previous from the rate in the data year. The change in the prescribing rate is displayed as an increase, decrease, or no change. An increase reflects a percentage point difference of at least 0.10 and a decrease reflects a difference of at least -0.10.

```r
medicaid_opioid_prescribing_rates <- read.csv("./OMT_MDCD_R23_P11_V10_YTD21_GEO.csv")
kable(medicaid_opioid_prescribing_rates[1:10, 1:8], caption = "medicaid_opioid_prescribing_rates")
```

Table 3: medicaid_opioid_prescribing_rates

Year	Geo_Lvl	Geo_Cd	Geo_Desc	Plan_Type	Tot_Opioid_Clms	Tot_Clms	Opioid_Prscrbng_Rate
2021	National	NA	National	All	21654225	686625295	3.15
2021	National	NA	National	FFS	5084859	180712324	2.81
2021	National	NA	National	MC	16569366	505912971	3.28
2021	State	1	Alabama	All	175237	7525456	2.33
2021	State	1	Alabama	FFS	175237	7525456	2.33
2021	State	1	Alabama	MC	0	0	NA
2021	State	2	Alaska	All	58330	1436383	4.06
2021	State	2	Alaska	FFS	58330	1436383	4.06
2021	State	2	Alaska	MC	0	0	NA
2021	State	4	Arizona	All	512306	14333371	3.57

Opioid Treatment Program Providers

The Opioid Treatment Program (OTP) Providers dataset provides information on Providers who have enrolled in Medicare under the Opioid Treatment Program. It contains provider's name, National Provider Identifier (NPI), address, phone number and the effective enrollment date.

Variable Name	Definition
NPI	National Provider Identifier (NPI) number of the Provider
Provider Name	Name of the Provider
Address Line 1	Provider's Street Address
Address Line 2	Provider's Street Address
City	Provider's City
State	Provider's State Abbreviation
Zip	Provider's Zip Code
Medicare ID Effective Date	The date when the Provider's Medicare ID becomes effective
Phone	Provider's Phone Number

```r
opioid_treatment_program_providers <- read.csv("./OPIOID_TREATMENT_PROGRAM_PROVIDERS_10102023.csv")
kable(opioid_treatment_program_providers[1:10, 1:8], caption = "opioid_treatment_program_providers")
```

Table 5: opioid_treatment_program_providers

NPI	PROVIDER.NAME	ADDRESS.LINE	ADDRESS.LINE.2	CITY	STATE	ZIP	MEDICARE.ID.EFFECTIVE.
1003081399 1013055110	BAART BEHAVIORAL HEALTH SERVICES IN	617 COM-STOCK RD	STE 5	BERLIN	VT	05602-8498	1/1/2020
1003150004	AMS OF WISCONSIN LLC	9532 E 16 FRONTAGE RD	STE 100	ONALASKA	WI	54650-6742	1/1/2020
1003362484	BHG XLII LLC	5715 PRINCESS ANNE RD		VIRGINIA BEACH	VA	23462-3222	1/1/2020
1003368945	RTS EDGEWOOD	2205 PULASKI HIGHWAY		EDGEWOOD	MD	21040	10/13/2020
1003571647	METRO TREATMENT OF FLORIDA LP	1241 BLANDING BLVD, STE 5	NEW SEASON TREATMENT CENTER 21	ORANGE PARK	FL	32065-5908	1/1/2020
1003581174 1326713314	PREMIER CARE OF OHIO, LLC	2632 WOODMAN CENTER CT		KETTERING	OH	45420-1477	1/1/2020
1003583733	AFFINITY HEALTHCARE GROUP CHERRY HI	1305 KINGS HWY N		CHERRY HILL	NJ	08034-1919	9/8/2022
1003947193	WEST TEXAS COUNSELING & REHABILITAT	1108 DOBIE DR STE 102	WTCR PLANO, INC.	PLANO	TX	75074-5391	1/1/2020
1003953548	ALLIANCE RECOVERY CENTER	1116 E PONCE DE LEON AVE		DECATUR	GA	30030-2711	1/1/2020
1003953548	ALLIANCE RECOVERY CENTER	119 SYCAMORE DR		ATHENS	GA	30606-3462	1/1/2020

VSRR Provisional Drug Overdose Death Counts

This data presents provisional counts for drug overdose deaths based on a current flow of mortality data in the National Vital Statistics System. Counts for the most recent final annual data are provided for comparison. National provisional counts include deaths occurring within the 50 states and the District of Columbia as of the date specified and may not include all deaths that occurred during a given time period. Provisional counts are often incomplete and causes of death may be pending investigation resulting in an underestimate relative to final counts. To address this, methods were developed to adjust provisional counts for reporting delays by generating a set of predicted provisional counts.

Several data quality metrics, including the percent completeness in overall death reporting, percentage of deaths with cause of death pending further investigation, and the percentage of drug overdose deaths with specific drugs or drug classes reported are included to aid in interpretation of provisional data as these measures are related to the accuracy of provisional counts. Reporting of the specific drugs and drug classes involved in drug overdose deaths varies by jurisdiction, and comparisons of death rates involving specific drugs across selected jurisdictions should not be made. Provisional data presented will be updated on a monthly basis as additional records are received.

Variable Name	Definition
State	The specific state within the U.S. for which the data is presented.
Year	The specific year for which the data is presented.
Month	The specific month for which the data is presented.
Period	Time frame or duration for which the data is relevant (e.g., a specific month, quarter, or year).
Indicator	The specific metric or measure being reported (e.g., drug overdose deaths, specific drug involved).
Data Value	The actual numerical value or count associated with the indicator for the specified state, year, and month.
Percent Complete	Percentage completeness of all death reports for the specified period.
Percent Pending Investigation	Percentage of deaths for the specified period where the cause is still under further investigation.
State Name	Full name of the state for which the data is presented.
Footnote	Additional notes or clarifications related to the data for the specified state, year, and month.
Footnote Symbol	Symbol or marker indicating the presence of a footnote or the type of footnote provided.
Predicted Value	Predicted or adjusted value for the indicator, compensating for potential underestimation due to reporting delays.

```
vssr_provisional_drug_overdose_death_counts <- read.csv("./VSRR_Provisional_Drug_Overdose_Death_Counts.
kable(vssr_provisional_drug_overdose_death_counts[1:10, 1:8],
      caption = "vssr_provisional_drug_overdose_death_counts")
```

Table 7: vssr_provisional_drug_overdose_death_counts

State	Year	Month	Period	Indicator	Data.Value	Percent.Complete	Percent.Pending.Investigation
AK	2015	April	12 month-ending	Percent with drugs specified	88.09524	100	0
AK	2015	April	12 month-ending	Heroin (T40.1)	NA	100	0
AK	2015	April	12 month-ending	Opioids (T40.0-T40.4,T40.6)	NA	100	0
AK	2015	April	12 month-ending	Natural & semi-synthetic opioids, incl. methadone (T40.2, T40.3)	NA	100	0
AK	2015	April	12 month-ending	Methadone (T40.3)	NA	100	0
AK	2015	April	12 month-ending	Psychostimulants with abuse potential (T43.6)	NA	100	0
AK	2015	April	12 month-ending	Number of Drug Overdose Deaths	126.00000	100	0

State	Year	Month	Period	Indicator	Data.Value	Percent.Complete	Percent.Pending.Investigation
AK	2015	April	12 month-ending	Natural & semi-synthetic opioids (T40.2)	NA	100	0
AK	2015	April	12 month-ending	Natural, semi-synthetic, & synthetic opioids, incl. methadone (T40.2-T40.4)	NA	100	0
AK	2015	April	12 month-ending	Synthetic opioids, excl. methadone (T40.4)	NA	100	0

Diving into Data

A Brief look at Medicaid Opioid Prescribing Rates Data

First, we are going to summarize the Medicaid Opioid Prescribing Rates data in a whole view. We may find that the data is categorized based on geographical levels. The data also hints at various types of health or insurance plans.

When delving into the opioid data, we observe that the total number of opioid claims ranged massively, from none at all to almost 38 million. In comparison, the total claims, encompassing more than just opioids, reached up to a staggering 704 million. The average opioid prescribing rate stands at around 5%, but this varies, with some places having a rate as high as 29.44%.

A notable aspect of the data is the focus on the evolution of these rates. Over a 5-year span, the opioid prescribing rate has seen both significant increases and decreases, with the most drastic 5-year change being an increase of 16.19%. Yearly changes also exhibit variability, with the most pronounced shift being a 15.31% increase.

The dataset doesn't just limit itself to general opioids; it pays specific attention to long-acting (LA) opioids. While the total number of these specific claims reached up to around 4.7 million, their prescribing rate compared to all prescriptions was, on average, lower at 0.73%.

Statistic	N	Mean	St. Dev.	Min	Max
Year	1,404	2,017.000	2.583	2,013	2,021
Geo_Cd	1,377	28.961	15.683	1	56
Tot_Opioid_Clms	1,386	770,834.800	3,049,434.000	0	37,964,067
Tot_Clms	1,402	16,592,302.000	64,943,294.000	0	704,296,772
Opioid_Prscrbng_Rate	1,272	5.014	2.784	0.000	29.440
Opioid_Prscrbng_Rate_5Y_Chg	552	-2.829	2.421	-10.420	16.190
Opioid_Prscrbng_Rate_1Y_Chg	1,119	-0.447	1.039	-4.100	15.310
LA_Tot_Opioid_Clms	1,368	80,656.540	347,496.000	0	4,672,903
LA_Opioid_Prscrbng_Rate	1,248	10.050	10.935	0.000	97.470
LA_Opioid_Prscrbng_Rate_5Y_Chg	532	3.046	14.252	-14.260	84.250
LA_Opioid_Prscrbng_Rate_1Y_Chg	1,093	0.560	5.139	-12.330	92.650

```
summary(medicaid_opioid_prescribing_rates)
```

```
##       Year        Geo_Lvl              Geo_Cd         Geo_Desc
##  Min.   :2013   Length:1404        Min.   : 1.00   Length:1404
##  1st Qu.:2015   Class :character   1st Qu.:16.00   Class :character
##  Median :2017   Mode  :character   Median :29.00   Mode  :character
##  Mean   :2017                      Mean   :28.96
##  3rd Qu.:2019                      3rd Qu.:42.00
```

```
##  Max.   :2021                      Max.   :56.00
##                                    NA's   :27
##   Plan_Type          Tot_Opioid_Clms       Tot_Clms          Opioid_Prscrbng_Rate
##  Length:1404         Min.   :       0   Min.   :         0   Min.   : 0.000
##  Class :character    1st Qu.:   41961   1st Qu.:   1096489   1st Qu.: 3.090
##  Mode  :character    Median :  182083   Median :   4402720   Median : 4.640
##                      Mean   :  770835   Mean   :  16592302   Mean   : 5.014
##                      3rd Qu.:  570217   3rd Qu.:  11378235   3rd Qu.: 6.420
##                      Max.   :37964067   Max.   :704296772   Max.   :29.440
##                      NA's   :18         NA's   :2           NA's   :132
##  Opioid_Prscrbng_Rate_5Y_Chg Opioid_Prscrbng_Rate_1Y_Chg LA_Tot_Opioid_Clms
##  Min.   :-10.420             Min.   :-4.1000             Min.   :      0
##  1st Qu.: -3.910             1st Qu.:-0.8200             1st Qu.:   3646
##  Median : -3.065             Median :-0.4800             Median :  14459
##  Mean   : -2.829             Mean   :-0.4474             Mean   :  80657
##  3rd Qu.: -2.237             3rd Qu.:-0.1800             3rd Qu.:  46538
##  Max.   : 16.190             Max.   :15.3100             Max.   :4672903
##  NA's   :852                 NA's   :285                 NA's   :36
##  LA_Opioid_Prscrbng_Rate LA_Opioid_Prscrbng_Rate_5Y_Chg
##  Min.   : 0.000          Min.   :-14.260
##  1st Qu.: 5.598          1st Qu.: -1.500
##  Median : 8.250          Median : -0.220
##  Mean   :10.050          Mean   :  3.046
##  3rd Qu.:10.355          3rd Qu.:  1.225
##  Max.   :97.470          Max.   : 84.250
##  NA's   :156             NA's   :872
##  LA_Opioid_Prscrbng_Rate_1Y_Chg
##  Min.   :-12.3300
##  1st Qu.: -0.5700
##  Median : -0.0600
##  Mean   :  0.5597
##  3rd Qu.:  0.4600
##  Max.   : 92.6500
##  NA's   :311
```

```r
# use tbl_summary to obtain the summary of medicaid_opioid_prescribing_rates
tbl_summary(medicaid_opioid_prescribing_rates %>% filter(Year >= 2015),
            include = -c("Geo_Desc"),
            by = "Year",
            statistic = list(
              all_continuous() ~ "Min: {min}, Max: {max}, Mean: {mean}, SD: {sd} ",
              all_categorical() ~ "{n} / {N} ({p}%)"
            )
) %>%
  as_gt() %>%
  gt::gtsave(
    filename = "medicaid_opioid_prescribing_rates.png"
  )
```

```
## Warning: There were 2 warnings in `mutate()`.
## The first warning was:
## i In argument: `sd = (function (x, na.rm = FALSE) ...`.
## Caused by warning:
## ! There were 6 warnings in `summarise()`.
## The first warning was:
```

```
## i In argument: `min = .Primitive("min")(variable)`.
## i In group 1: `by = 2015`.
## Caused by warning:
## ! no non-missing arguments to min; returning Inf
## i Run `dplyr::last_dplyr_warnings()` to see the 5 remaining warnings.
## i Run `dplyr::last_dplyr_warnings()` to see the 1 remaining warning.
```

We may find that, for columns like `Opioid_Prscrbng_Rate_5Y_Chg`, there are a lot of `NA` values. Since those rows take up a large proportion of the whole dataset, we may not want to drop them. Instead, we can remove those columns from our analysis. And then, we can simply replace the `NA` values in other columns with 0.

```
medicaid_opioid_prescribing_rates <- medicaid_opioid_prescribing_rates %>%
  select(-c(Opioid_Prscrbng_Rate_5Y_Chg, LA_Opioid_Prscrbng_Rate_5Y_Chg))

medicaid_opioid_prescribing_rates[is.na(medicaid_opioid_prescribing_rates)] <- 0
```

```
summary(medicaid_opioid_prescribing_rates)
```

```
##       Year        Geo_Lvl               Geo_Cd         Geo_Desc
##  Min.   :2013   Length:1404        Min.   : 0.00   Length:1404
##  1st Qu.:2015   Class :character   1st Qu.:15.75   Class :character
##  Median :2017   Mode  :character   Median :28.50   Mode  :character
##  Mean   :2017                      Mean   :28.40
##  3rd Qu.:2019                      3rd Qu.:41.25
##  Max.   :2021                      Max.   :56.00
##   Plan_Type         Tot_Opioid_Clms       Tot_Clms          Opioid_Prscrbng_Rate
##  Length:1404        Min.   :       0   Min.   :        0   Min.   : 0.000
##  Class :character   1st Qu.:   38192   1st Qu.:  1065782   1st Qu.: 2.627
##  Mode  :character   Median :  177328   Median :  4380402   Median : 4.300
##                     Mean   :  760952   Mean   : 16568666   Mean   : 4.543
##                     3rd Qu.:  563518   3rd Qu.: 11378235   3rd Qu.: 6.270
##                     Max.   :37964067   Max.   :704296772   Max.   :29.440
##  Opioid_Prscrbng_Rate_1Y_Chg LA_Tot_Opioid_Clms LA_Opioid_Prscrbng_Rate
##  Min.   :-4.1000             Min.   :      0    Min.   : 0.000
##  1st Qu.:-0.7000             1st Qu.:   2888    1st Qu.: 4.380
##  Median :-0.3250             Median :  13849    Median : 7.830
##  Mean   :-0.3566             Mean   :  78588    Mean   : 8.933
##  3rd Qu.: 0.0000             3rd Qu.:  44809    3rd Qu.: 9.920
##  Max.   :15.3100             Max.   :4672903    Max.   :97.470
##  LA_Opioid_Prscrbng_Rate_1Y_Chg
##  Min.   :-12.3300
##  1st Qu.: -0.3900
##  Median :  0.0000
##  Mean   :  0.4357
##  3rd Qu.:  0.2625
##  Max.   : 92.6500
```

Let's look at the distribution of the opioid prescribing rate.

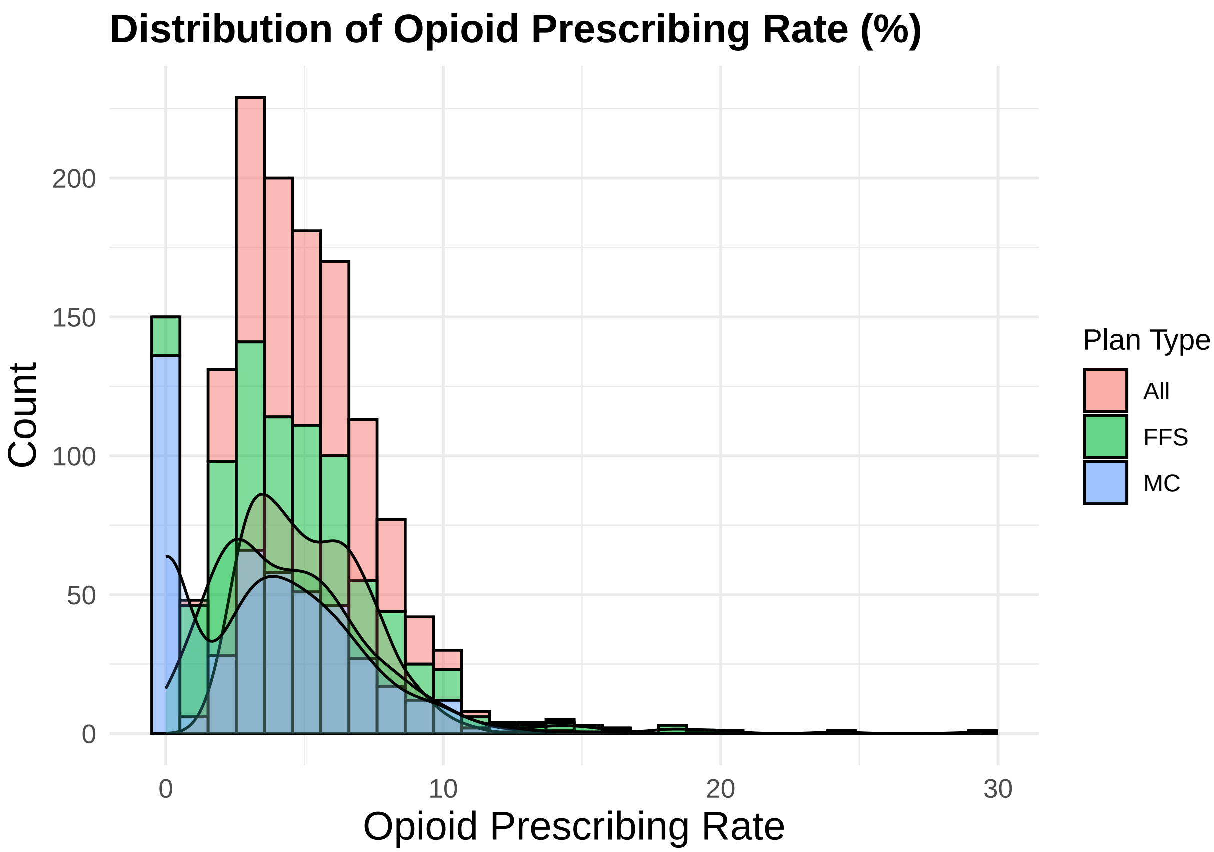
```
options(repr.plot.width = 10, repr.plot.height = 10)

# geom_histogram and geom_density for the distribution
ggplot(medicaid_opioid_prescribing_rates, aes(x = Opioid_Prscrbng_Rate, fill = Plan_Type)) +
  geom_histogram(bins = 30, color = "black", alpha = 0.5) +
  geom_density(aes(y = after_stat(count), fill = Plan_Type, group = Plan_Type), alpha = 0.2) +
  labs(
```

Characteristic	**2015**, N = 156[1]	**2016**, N = 156[1]	**2017**, N = 156[1]	**2018**, N = 156[1]	**2019**, N = 156[1]	**2020**, N = 156[1]	**2021**, N 156[1]
Geo_Lvl							
National	3 / 156 (1.9%)	3 / 156 (1.9%)	3 / 156 (1.9%)	3 / 156 (1.9%)	3 / 156 (1.9%)	3 / 156 (1.9%)	3 / 156 (1.9%)
State	153 / 156 (98%)	153 / 156 (98%)	153 / 156 (98%)	153 / 156 (98%)	153 / 156 (98%)	153 / 156 (98%)	153 / 15 (98%)
Geo_Cd	Min: 1, Max: 56, Mean: 29, SD: 16	Min: 1, Max: 56, Mean: 29, SD: 16	Min: 1, Max: 56, Mean: 29, SD: 16	Min: 1, Max: 56, Mean: 29, SD: 16	Min: 1, Max: 56, Mean: 29, SD: 16	Min: 1, Max: 56, Mean: 29, SD: 16	Min: 1, M 56, Mea 29, SD:
Unknown	3	3	3	3	3	3	3
Plan_Type							
All	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 15 (33%)
FFS	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 15 (33%)
MC	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 156 (33%)	52 / 15 (33%)
Tot_Opioid_Clms	Min: 0, Max: 37,964,067, Mean: 973,438, SD: 3,758,597	Min: 0, Max: 36,902,273, Mean: 952,082, SD: 3,716,422	Min: 0, Max: 32,240,821, Mean: 835,342, SD: 3,260,899	Min: 0, Max: 25,287,108, Mean: 655,127, SD: 2,554,512	Min: 0, Max: 22,058,928, Mean: 578,881, SD: 2,254,238	Min: 0, Max: 21,159,978, Mean: 548,523, SD: 2,167,363	Min: 0, M 21,654,2 Mean: 562,165, 2,231,26
Unknown	0	2	2	2	4	2	2
Tot_Clms	Min: 0, Max: 641,988,059, Mean: 16,461,232, SD: 63,693,614	Min: 0, Max: 685,821,751, Mean: 17,585,173, SD: 68,736,795	Min: 0, Max: 704,296,772, Mean: 18,260,601, SD: 71,511,896	Min: 0, Max: 685,892,618, Mean: 17,586,990, SD: 69,162,256	Min: 0, Max: 679,767,097, Mean: 17,429,926, SD: 68,638,715	Min: 0, Max: 653,713,145, Mean: 16,761,876, SD: 66,019,277	Min: 0, M 686,625,2 Mean: 17,605,7 SD: 69,547,3
Unknown	0	0	2	0	0	0	0
Opioid_Prscrbng_Rate	Min: 0.00, Max: 19.68, Mean: 6.23, SD: 2.39	Min: 1.20, Max: 18.36, Mean: 5.71, SD: 2.21	Min: 0.85, Max: 18.49, Mean: 4.87, SD: 2.05	Min: 0.00, Max: 14.99, Mean: 4.07, SD: 1.91	Min: 0.53, Max: 17.78, Mean: 3.60, SD: 2.20	Min: 0.04, Max: 16.19, Mean: 3.47, SD: 2.46	Min: 0.0 Max: 29. Mean: 3. SD: 3.1
Unknown	14	15	14	14	15	14	13
Opioid_Prscrbng_Rate_5Y_Chg	Min: Inf, Max: -Inf, Mean: NA, SD: NA	Min: Inf, Max: -Inf, Mean: NA, SD: NA	Min: Inf, Max: -Inf, Mean: NA, SD: NA	Min: -9.21, Max: 5.93, Mean: -2.92, SD: 1.92	Min: -9.48, Max: 13.84, Mean: -3.22, SD: 2.44	Min: -10.42, Max: 16.19, Mean: -2.73, SD: 2.92	Min: -9. Max: 8.9 Mean: -2. SD: 2.2
Unknown	156	156	156	19	19	17	17
Opioid_Prscrbng_Rate_1Y_Chg	Min: -3.40, Max: 9.06, Mean: -0.55, SD: 1.15	Min: -2.37, Max: 4.26, Mean: -0.58, SD: 0.67	Min: -2.86, Max: 2.28, Mean: -0.82, SD: 0.55	Min: -3.50, Max: 7.00, Mean: -0.79, SD: 1.03	Min: -2.37, Max: 7.02, Mean: -0.49, SD: 0.93	Min: -3.49, Max: 15.31, Mean: -0.10, SD: 1.47	Min: -2.7 Max: 13.2 Mean: -0. SD: 1.2
Unknown	16	16	16	16	16	16	16
LA_Tot_Opioid_Clms	Min: 0, Max: 2,885,869, Mean: 73,997, SD: 282,979	Min: 0, Max: 2,783,521, Mean: 72,833, SD: 277,805	Min: 0, Max: 2,412,650, Mean: 63,267, SD: 241,115	Min: 0, Max: 2,406,827, Mean: 63,189, SD: 242,996	Min: 0, Max: 3,119,928, Mean: 82,008, SD: 351,902	Min: 0, Max: 4,008,483, Mean: 108,155, SD: 485,199	Min: 0, M 4,672,90 Mean: 122,886, 565,89
Unknown	0	4	4	4	4	8	4
LA_Opioid_Prscrbng_Rate	Min: 0.0, Max: 26.1, Mean: 8.4, SD: 3.6	Min: 2.0, Max: 47.9, Mean: 8.8, SD: 5.1	Min: 1.6, Max: 35.6, Mean: 9.1, SD: 4.8	Min: 2.0, Max: 86.0, Mean: 10.9, SD: 11.5	Min: 0.0, Max: 91.9, Mean: 11.3, SD: 14.0	Min: 0.0, Max: 94.2, Mean: 12.6, SD: 17.5	Min: 0.0 Max: 97. Mean: 12 SD: 18.
Unknown	15	17	16	17	15	20	15
LA_Opioid_Prscrbng_Rate_5Y_Chg	Min: Inf, Max: -Inf, Mean: NA, SD: NA	Min: Inf, Max: -Inf, Mean: NA, SD: NA	Min: Inf, Max: -Inf, Mean: NA, SD: NA	Min: -14, Max: 76, Mean: 2, SD: 11	Min: -13, Max: 82, Mean: 3, SD: 13	Min: -10, Max: 84, Mean: 4, SD: 16	Min: -11 Max: 81 Mean: 3, 16
Unknown	156	156	156	25	23	23	21
LA_Opioid_Prscrbng_Rate_1Y_Chg	Min: -6.59, Max: 6.49, Mean: 0.17, SD: 1.29	Min: -4.21, Max: 40.49, Mean: 0.44, SD: 3.75	Min: -12.31, Max: 8.88, Mean: 0.28, SD: 2.01	Min: -7.96, Max: 50.32, Mean: 1.82, SD: 8.89	Min: -8.01, Max: 24.63, Mean: 0.58, SD: 3.90	Min: -12.33, Max: 92.65, Mean: 1.25, SD: 9.13	Min: -3.0 Max: 19. Mean: 0. SD: 3.0
Unknown	20	18	18	18	18	20	20

[1] n / N (%); Min: Minimum, Max: Maximum, Mean: Mean, SD: SD

Figure 1: medicaid_opioid_prescribing_rates

```
    title = "Distribution of Opioid Prescribing Rate (%)",
    x = "Opioid Prescribing Rate",
    y = "Count",
    fill = "Plan Type"
  ) +
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 15, face = "bold"),
    axis.title = element_text(size = 15),
    axis.text = element_text(size = 10)
  )
```



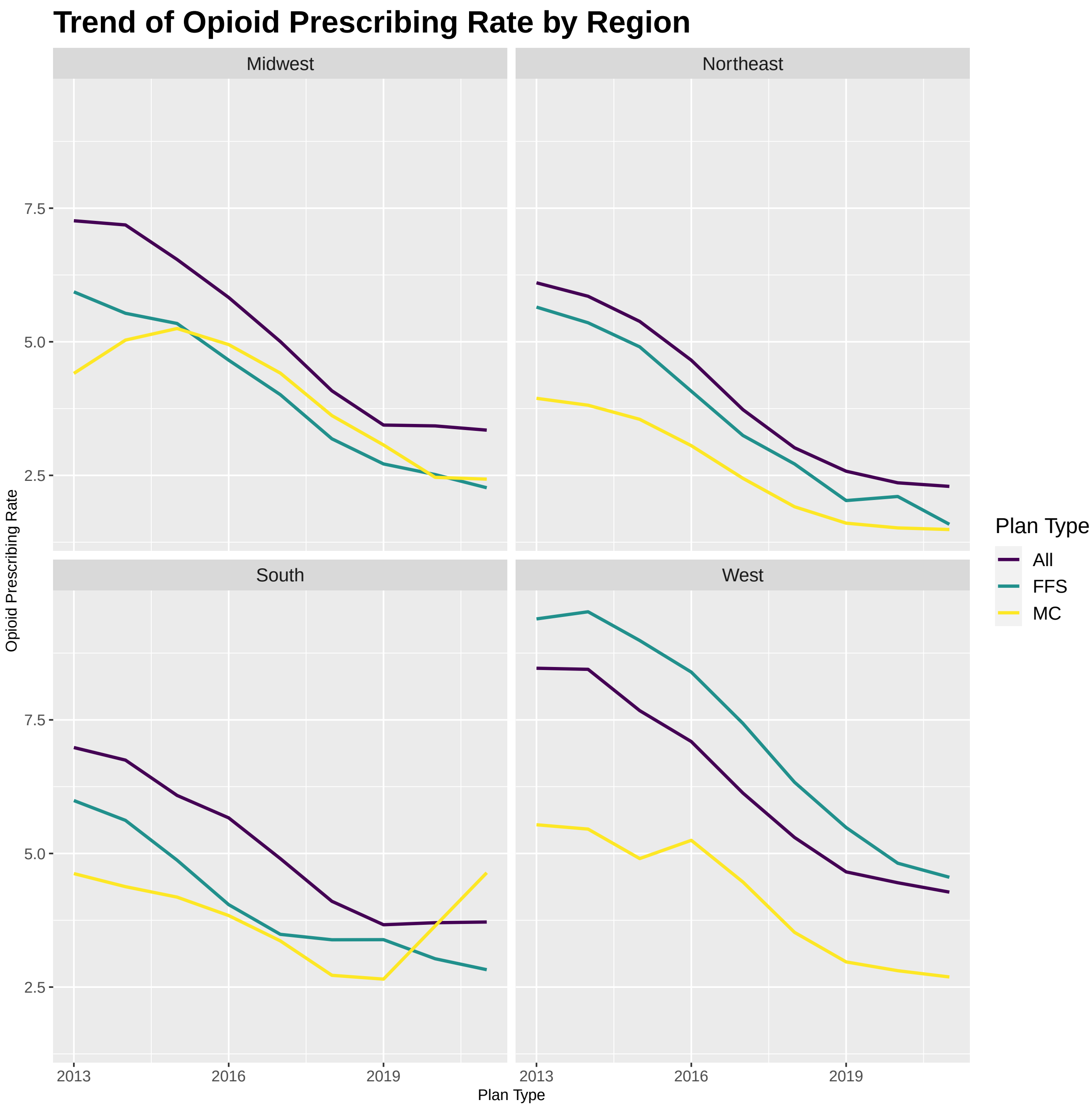
- All (Red) Plan Type: This plan type has a peak frequency between approximately 5% and 10% on the Opioid Prescribing Rate scale, with a curve that suggests a somewhat normal distribution. The distribution slightly skews to the right.
- FFS (Green) Plan Type: The distribution for this plan type peaks slightly earlier than the "All" plan type and seems to have a lower frequency. The curve for FFS is broader and flatter compared to the other two.
- MC (Blue) Plan Type:The distribution for this plan type starts at a higher frequency at the lower end of the scale and then decreases steadily. The MC curve begins with a sharp rise and then steadily declines, differentiating it from the other curves.

Next, let's look at the distribution of the Opioid_Prscrbng_Rate by state. The distribution of the data is very interesting for some of the states. For example, Arizona's FFS plan type has a very high opioid prescribing rate, while its MC plan type has a very low opioid prescribing rate. Nevertheless, some of the states have the opposite situation. But in general, the opioid prescribing rate is decreasing over years for both plan types.

```r
# classify states into regions (Northeast, Midewest, South, West) first, then plot them
medicaid_opioid_prescribing_rates %>%
  filter(Geo_Desc != "National") %>%
  mutate(region = case_when(
    Geo_Desc %in% c("Maine", "New Hampshire", "Vermont",
                    "Massachusetts", "Rhode Island", "Connecticut",
                    "New York", "Pennsylvania", "New Jersey",
                    "District of Columbia") ~ "Northeast",
    Geo_Desc %in% c("Ohio", "Indiana", "Illinois",
                    "Michigan", "Wisconsin", "Missouri",
                    "North Dakota", "South Dakota", "Nebraska",
                    "Kansas", "Minnesota", "Iowa") ~ "Midwest",
    Geo_Desc %in% c("Delaware", "Maryland", "Virginia",
                    "West Virginia", "North Carolina", "South Carolina",
                    "Georgia", "Florida", "Kentucky",
                    "Tennessee", "Alabama", "Mississippi",
                    "Arkansas", "Louisiana", "Oklahoma", "Texas") ~ "South",
    Geo_Desc %in% c("Idaho", "Montana", "Wyoming",
                    "Nevada", "Utah", "Colorado",
                    "Arizona", "New Mexico", "Alaska",
                    "California", "Hawaii", "Oregon", "Washington") ~ "West",
    TRUE ~ "Other"
  )) %>%
  group_by(Year, Plan_Type, region) %>%
  summarise(Opioid_Prscrbng_Rate = mean(Opioid_Prscrbng_Rate)) %>%
  ggplot(aes(x = Year, y = Opioid_Prscrbng_Rate, color = Plan_Type)) +
  geom_line(size = 1) +
  labs(
    title = "Trend of Opioid Prescribing Rate by Region",
    x = "Year",
    y = "Opioid Prescribing Rate",
    color = "Plan Type"
  ) +
  facet_wrap(~region, ncol = 2) +
  scale_color_viridis_d() +
  scale_x_continuous(breaks = seq(min(medicaid_opioid_prescribing_rates$Year),
                                  max(medicaid_opioid_prescribing_rates$Year),
                                  by = 3),
                     name = "Plan Type") +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 20, face = "bold"),
    plot.subtitle = element_text(size = 16),
    axis.title = element_text(size = 10),
    axis.text = element_text(size = 10),
    legend.title = element_text(size = 14),
    legend.text = element_text(size = 12),
    strip.text = element_text(size = 12)
  )
```

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

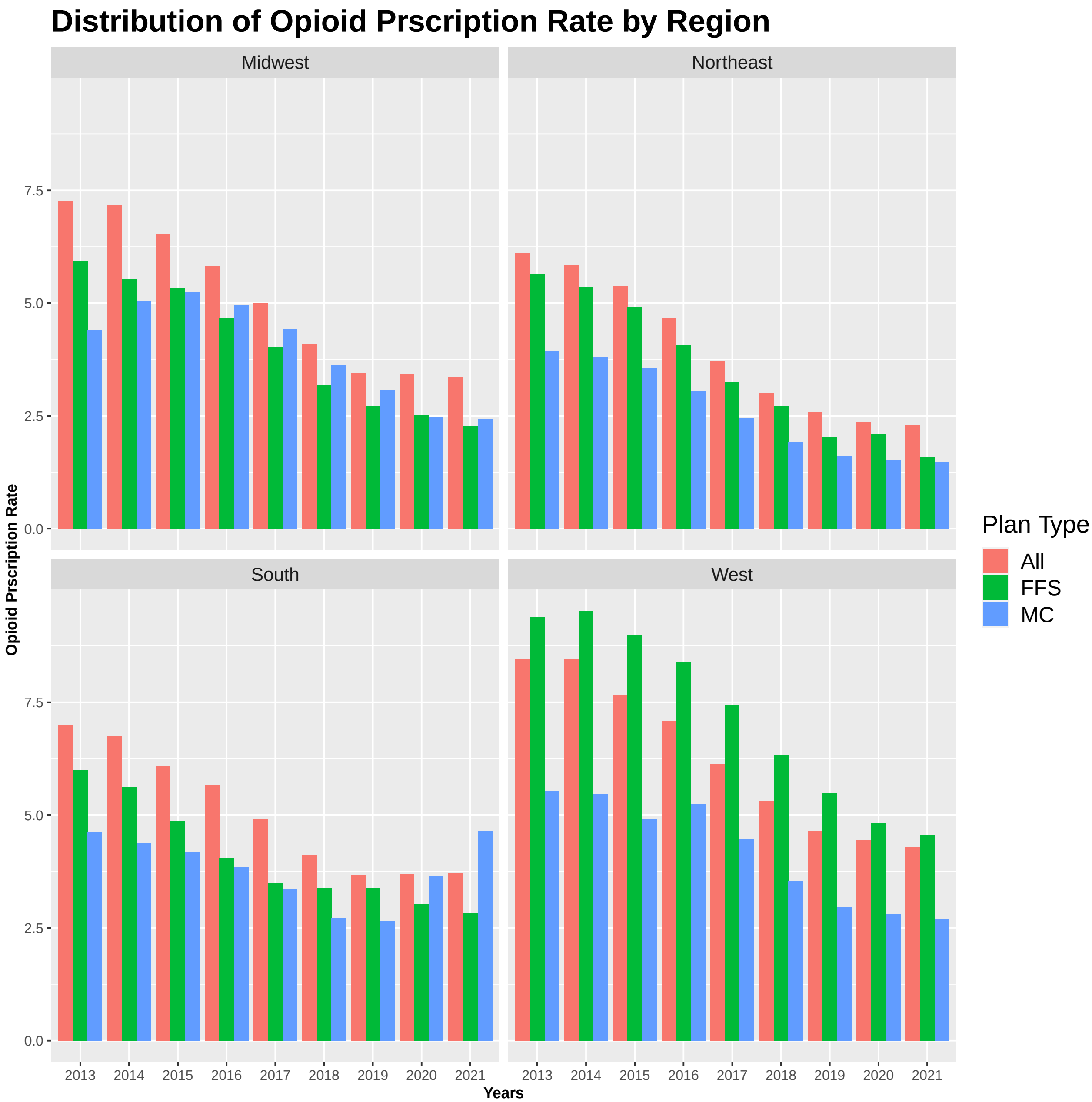
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```




```r
# classify states into regions (Northeast, Midewest, South, West) first, then plot them
medicaid_opioid_prescribing_rates %>%
  filter(Geo_Desc != "National") %>%
  mutate(region = case_when(
    Geo_Desc %in% c("Maine", "New Hampshire", "Vermont",
                    "Massachusetts", "Rhode Island", "Connecticut",
                    "New York", "Pennsylvania", "New Jersey",
                    "District of Columbia") ~ "Northeast",
```

```r
    Geo_Desc %in% c("Ohio", "Indiana", "Illinois",
                    "Michigan", "Wisconsin", "Missouri",
                    "North Dakota", "South Dakota", "Nebraska",
                    "Kansas", "Minnesota", "Iowa") ~ "Midwest",
    Geo_Desc %in% c("Delaware", "Maryland", "Virginia",
                    "West Virginia", "North Carolina", "South Carolina",
                    "Georgia", "Florida", "Kentucky",
                    "Tennessee", "Alabama", "Mississippi",
                    "Arkansas", "Louisiana", "Oklahoma", "Texas") ~ "South",
    Geo_Desc %in% c("Idaho", "Montana", "Wyoming",
                    "Nevada", "Utah", "Colorado",
                    "Arizona", "New Mexico", "Alaska",
                    "California", "Hawaii", "Oregon", "Washington") ~ "West",
    TRUE ~ "Other"
  )) %>%
  group_by(Year, Plan_Type, region) %>%
  summarise(Opioid_Prscrbng_Rate = mean(Opioid_Prscrbng_Rate)) %>%
  ggplot(aes(x = as.factor(Year),
             y = Opioid_Prscrbng_Rate,
             fill = as.factor(Plan_Type))) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Distribution of Opioid Prscription Rate by Region",
       x = "Years",
       y = "Opioid Prscription Rate",
       fill = "Plan Type") +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 20, face = "bold"),
    axis.title = element_text(size = 10, face = "bold"),
    legend.title = element_text(size = 16),
    legend.text = element_text(size = 14),
    strip.text = element_text(size = 12)
  ) +
  facet_wrap(~region, ncol = 2)
```

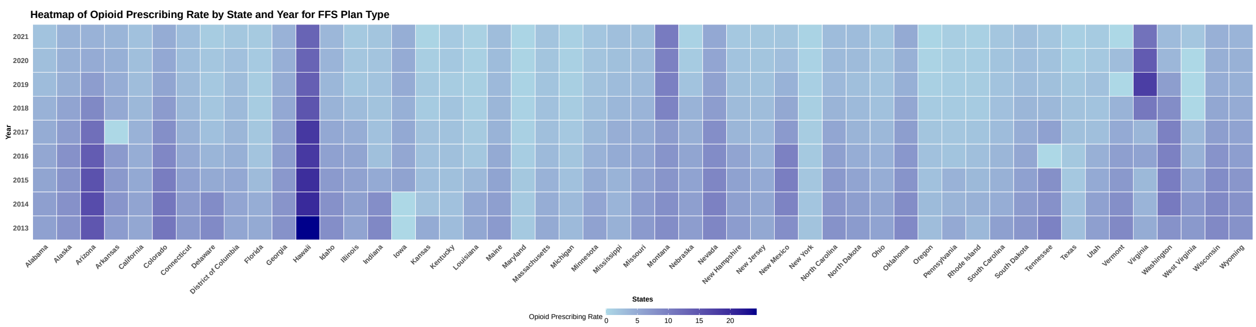
```
## `summarise()` has grouped output by 'Year', 'Plan_Type'. You can override using
## the `.groups` argument.
```



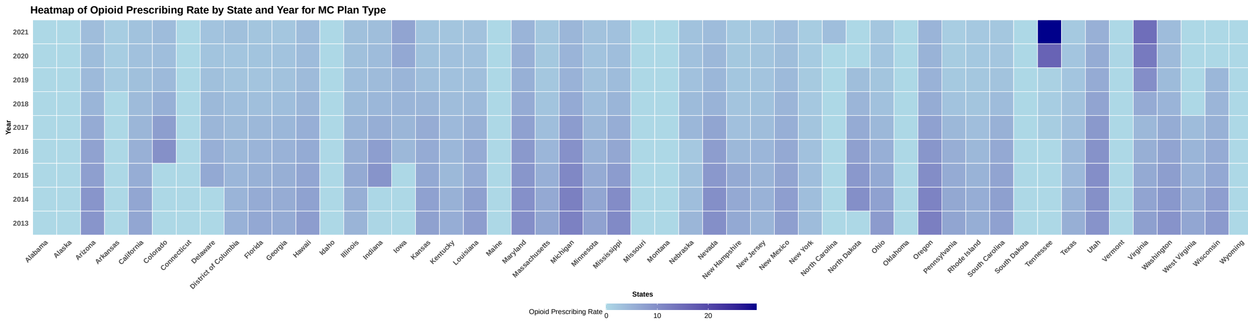
Next, we are going to draw a heatmap for `Opioid_Prscrbng_Rate`.

```
# Heatmap for plan type FFS
medicaid_opioid_prescribing_rates %>%
  filter(Plan_Type == "FFS" & Geo_Desc != "National") %>%
  ggplot(aes(y = as.factor(Year), x = Geo_Desc, fill = Opioid_Prscrbng_Rate)) +
  geom_tile(color = "white") +
  scale_y_discrete(position = "left", name = "Year") +
  scale_x_discrete(name = "States") +
  scale_fill_gradient(low = "lightblue",
                      high = "darkblue",
                      guide = guide_colourbar(barwidth = 30)) +
  labs(
    fill = "Opioid Prescribing Rate"
```

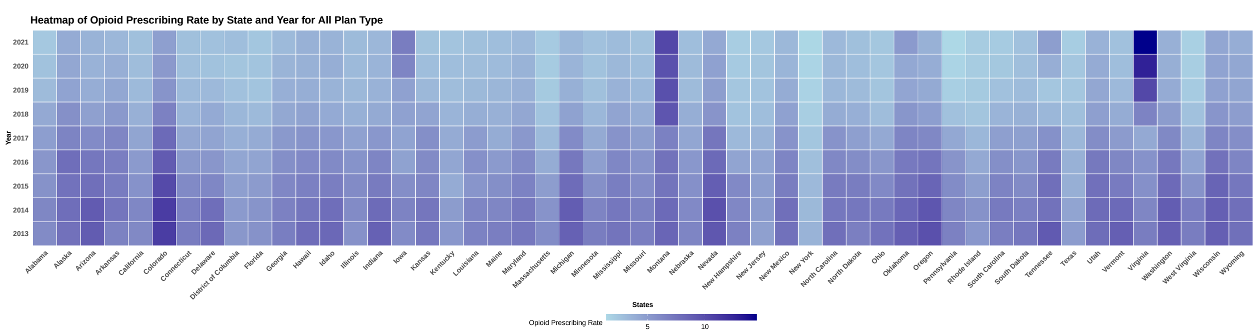
```
  ) +
  coord_equal() +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    legend.title = element_text(size = 20),
    legend.text = element_text(size = 20),
    plot.title = element_text(size = 30, face = "bold"),
    axis.title = element_text(size = 20, face = "bold"),
    axis.text = element_text(size = 20, face = "bold"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.grid = element_blank()
  ) +
  ggtitle("Heatmap of Opioid Prescribing Rate by State and Year for FFS Plan Type")
```




```
# Heatmap for plan type MC
medicaid_opioid_prescribing_rates %>%
  filter(Plan_Type == "MC" & Geo_Desc != "National") %>%
  ggplot(aes(y = as.factor(Year), x = Geo_Desc, fill = Opioid_Prscrbng_Rate)) +
  geom_tile(color = "white") +
  scale_y_discrete(position = "left", name = "Year") +
  scale_x_discrete(name = "States") +
  scale_fill_gradient(low = "lightblue",
                      high = "darkblue",
                      guide = guide_colourbar(barwidth = 30)) +
  labs(
    fill = "Opioid Prescribing Rate"
  ) +
  coord_equal() +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    legend.title = element_text(size = 20),
    legend.text = element_text(size = 20),
    plot.title = element_text(size = 30, face = "bold"),
    axis.title = element_text(size = 20, face = "bold"),
    axis.text = element_text(size = 20, face = "bold"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.grid = element_blank()
  ) +
  ggtitle("Heatmap of Opioid Prescribing Rate by State and Year for MC Plan Type")
```




```r
# Heatmap for plan type All
medicaid_opioid_prescribing_rates %>%
  filter(Plan_Type == "All" & Geo_Desc != "National") %>%
  ggplot(aes(y = as.factor(Year), x = Geo_Desc, fill = Opioid_Prscrbng_Rate)) +
  geom_tile(color = "white") +
  scale_y_discrete(position = "left", name = "Year") +
  scale_x_discrete(name = "States") +
  scale_fill_gradient(low = "lightblue",
                      high = "darkblue",
                      guide = guide_colourbar(barwidth = 30)) +
  labs(
    fill = "Opioid Prescribing Rate"
  ) +
  coord_equal() +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    legend.title = element_text(size = 20),
    legend.text = element_text(size = 20),
    plot.title = element_text(size = 30, face = "bold"),
    axis.title = element_text(size = 20, face = "bold"),
    axis.text = element_text(size = 20, face = "bold"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.grid = element_blank()
  ) +
  ggtitle("Heatmap of Opioid Prescribing Rate by State and Year for All Plan Type")
```



A Brief Look at Opioid Treatment Program Providers Data

```r
summary(opioid_treatment_program_providers)
```

```
##       NPI            PROVIDER.NAME       ADDRESS.LINE.1      ADDRESS.LINE.2
##   Length:1431        Length:1431        Length:1431        Length:1431
```

```
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##      CITY              STATE               ZIP
##  Length:1431        Length:1431        Length:1431
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##  MEDICARE.ID.EFFECTIVE.DATE    PHONE
##  Length:1431                Length:1431
##  Class :character           Class :character
##  Mode  :character           Mode  :character
```

For this dataset, what we're going to do is to group the data by state and count the number of providers. The rest of the dataset is not very useful for our analysis.

```r
# group the data by state and count the number of providers
provider_count <- opioid_treatment_program_providers %>%
  group_by(STATE) %>%
  summarise(Count = n()) %>%
  arrange(desc(Count))
kable(provider_count[1:10,], caption = "provider_count")
```

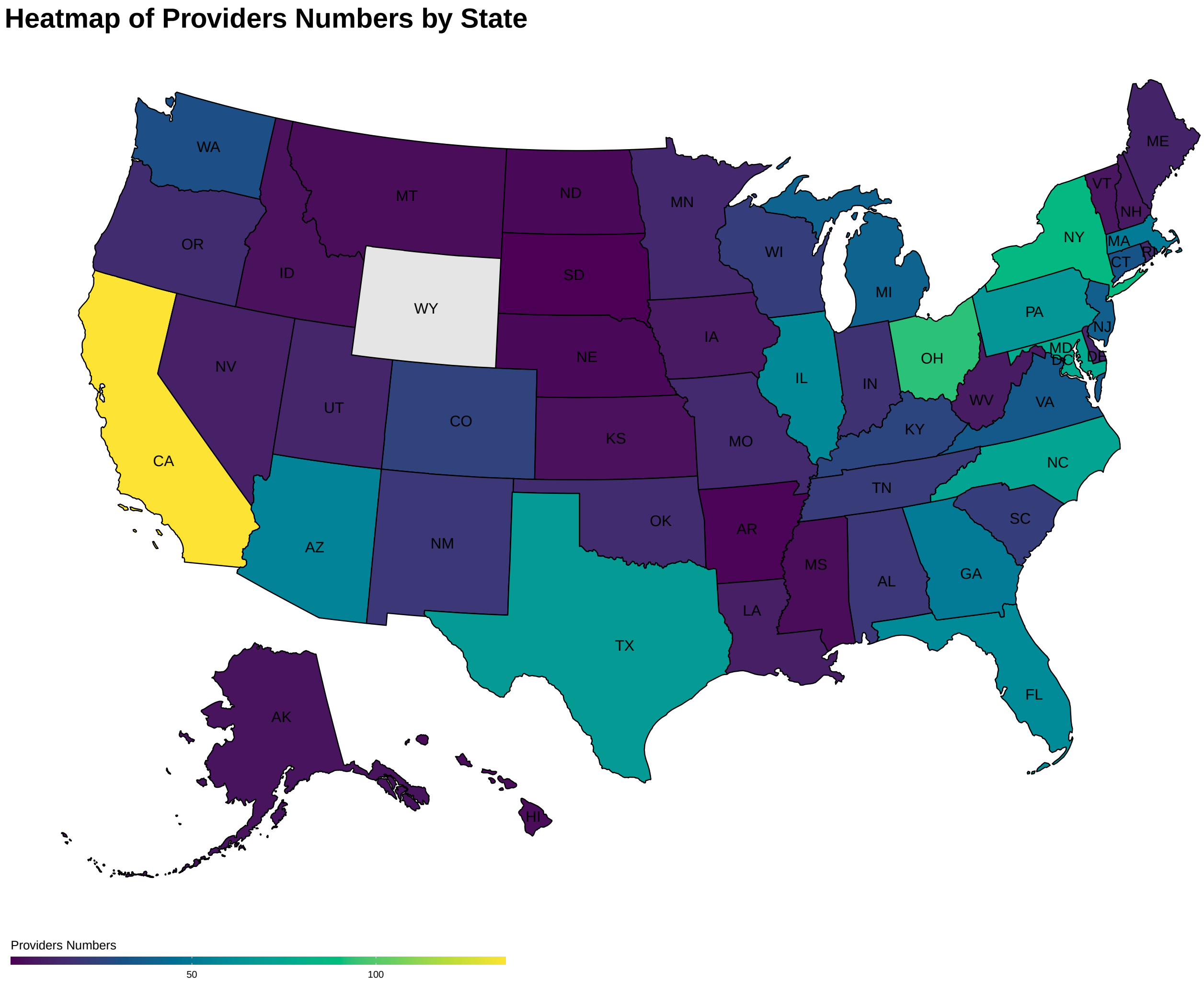
Table 9: provider_count

STATE	Count
CA	135
OH	93
NY	87
MD	76
NC	73
TX	68
PA	65
FL	59
IL	58
AZ	55

Next, we are going to plot the number of providers with a geo heatmap. Interesting, WY doesn't have any providers, while CA has the most providers.

```r
provider_count$state <- provider_count$STATE

# plot the number of providers with a geo heatmap
plot_usmap(data = provider_count, values = "Count", labels = TRUE) +
  scale_fill_gradientn(
    colours = hcl.colors(10), na.value = "grey90",
    guide = guide_colourbar(
      barwidth = 25, barheight = 0.4,
      title.position = "top"
    )
  ) +
  labs(fill = "Providers Numbers") +
  theme(
    legend.position = "bottom",
    plot.title = element_text(size = 20, face = "bold")
  ) +
```

```r
  ggtitle("Heatmap of Providers Numbers by State")
```



A Brief look at VSRR Provisional Drug Overdose Death Counts Data

This is the summary of drug overdose death dataset.

Variable	Min	1st Qu.	Median	Mean	3rd Qu.	Max	NA's
Year	2015	2017	2019	2019	2021	2023	-
Data.Value	10	96	315	13,334	1,270	3,538,076	11,033
Percent.Complete	99.5	100.0	100.0	100.0	100.0	100.0	-
Percent.Pending.Investigation	0.00000	0.01702	0.05266	0.11703	0.15430	1.74989	-
Predicted.Value	10	100	318	1,446	859	112,024	21,450

```r
summary(vssr_provisional_drug_overdose_death_counts)
```

```
##     State                Year         Month              Period
##  Length:60600       Min.   :2015   Length:60600       Length:60600
##  Class :character   1st Qu.:2017   Class :character   Class :character
##  Mode  :character   Median :2019   Mode  :character   Mode  :character
##                     Mean   :2019
```

```
##                     3rd Qu.:2021
##                     Max.   :2023
##
##   Indicator           Data.Value      Percent.Complete
##  Length:60600       Min.   :     10   Min.   : 99.5
##  Class :character   1st Qu.:     96   1st Qu.:100.0
##  Mode  :character   Median :    315   Median :100.0
##                     Mean   :  13334   Mean   :100.0
##                     3rd Qu.:   1270   3rd Qu.:100.0
##                     Max.   :3538076   Max.   :100.0
##                     NA's   :11033
##  Percent.Pending.Investigation  State.Name          Footnote
##  Min.   :0.00000              Length:60600       Length:60600
##  1st Qu.:0.01702              Class :character   Class :character
##  Median :0.05266              Mode  :character   Mode  :character
##  Mean   :0.11703
##  3rd Qu.:0.15430
##  Max.   :1.74989
##
##  Footnote.Symbol    Predicted.Value
##  Length:60600       Min.   :    10
##  Class :character   1st Qu.:   100
##  Mode  :character   Median :   318
##                     Mean   :  1446
##                     3rd Qu.:   859
##                     Max.   :112024
##                     NA's   :21450
```

```
# use tbl_summary to summarize vssr_provisional_drug_overdose_death_counts
tbl_summary(vssr_provisional_drug_overdose_death_counts,
            include = -c("Month", "State", "State.Name"),
            by = "Year",
            statistic = list(
              all_continuous() ~ "Min: {min}, Max: {max}, Mean: {mean}, SD: {sd} ",
              all_categorical() ~ "{n} / {N} ({p}%)"
            )
) %>%
  as_gt() %>%
  gt::gtsave(
    filename = "vssr_provisional_drug_overdose_death_counts.png"
  )
```

According to the description of the origin dataset, the column `Indicator` shows the specific metric or measure being reported (e.g., drug overdose deaths, specific drug involved). We can see that there are 12 different indicators in the dataset. The symbols like "T40.5" are codes from the International Classification of Diseases (ICD). Specifically, these codes are from the ICD-10 (10th revision) coding system, which is used worldwide for morbidity and mortality statistics, reimbursement systems, and automated decision support in health care.

The meaning of `Indicator` code: - T40.0: Opium - T40.1: Heroin - T40.2: Natural & semi-synthetic opioids - T40.3: Methadone - T40.4: Synthetic opioids, excluding methadone - T40.5: Cocaine - T43.6: Psychostimulants with abuse potential

Characteristic	**2015**, N = 7,200[1]	**2016**, N = 7,200[1]	**2017**, N = 7,200[1]	**2018**, N = 7,200[1]	**2019**, N = 7,200[1]	**2020**, N = 7,200[1]	**2021**, N = 7,200[1]	**2022**, N = 7,200[1]	**2023**, 3,00
Period									
12 month-ending	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	3,00 3,00 (100
Indicator									
Cocaine (T40.5)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Heroin (T40.1)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Methadone (T40.3)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Natural & semi-synthetic opioids (T40.2)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Natural & semi-synthetic opioids, incl. methadone (T40.2, T40.3)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Natural, semi-synthetic, & synthetic opioids, incl. methadone (T40.2-T40.4)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Number of Deaths	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	265 3,00 (8.8
Number of Drug Overdose Deaths	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	265 3,00 (8.8
Opioids (T40.0-T40.4,T40.6)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Percent with drugs specified	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	636 / 7,200 (8.8%)	265 3,00 (8.8
Psychostimulants with abuse potential (T43.6)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Synthetic opioids, excl. methadone (T40.4)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	588 / 7,200 (8.2%)	245 3,00 (8.2
Data.Value	Min: 10, Max: 2,729,315, Mean: 15,263, SD: 141,984	Min: 10, Max: 2,749,864, Mean: 13,994, SD: 135,521	Min: 10, Max: 2,820,034, Mean: 14,027, SD: 136,770	Min: 10, Max: 2,855,774, Mean: 13,245, SD: 134,054	Min: 10, Max: 2,861,523, Mean: 11,854, SD: 126,170	Min: 10, Max: 3,390,278, Mean: 12,481, SD: 134,281	Min: 10, Max: 3,538,076, Mean: 13,683, SD: 148,683	Min: 10, Max: 3,477,160, Mean: 13,226, SD: 143,923	Min: Ma 3,208, Mea 12,7 SD 135,5
Unknown	2,736	2,246	2,052	1,622	956	658	380	218	16
Percent.Complete									
99.5	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	12 / 7,200 (0.2%)	0 / 3, (0%
100	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,188 / 7,200 (100%)	3,00 3,00 (100
Percent.Pending.Investigation	Min: 0.00, Max: 1.41, Mean: 0.11, SD: 0.20	Min: 0.00, Max: 0.91, Mean: 0.11, SD: 0.15	Min: 0.00, Max: 0.90, Mean: 0.12, SD: 0.17	Min: 0.00, Max: 0.70, Mean: 0.11, SD: 0.14	Min: 0.00, Max: 0.71, Mean: 0.13, SD: 0.16	Min: 0.00, Max: 0.71, Mean: 0.11, SD: 0.14	Min: 0.00, Max: 0.57, Mean: 0.10, SD: 0.11	Min: 0.00, Max: 0.65, Mean: 0.11, SD: 0.13	Min: 0 Max: Mea 0.23, 0.2
Footnote									
	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	4,080 / 7,200 (57%)	0 / 3, (0%
Data suppressed (<10).	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	120 / 7,200 (1.7%)	0 / 3, (0%
Numbers may differ from published reports using final data. See Technical Notes.	4,464 / 7,200 (62%)	4,954 / 7,200 (69%)	5,148 / 7,200 (72%)	5,578 / 7,200 (77%)	6,244 / 7,200 (87%)	6,542 / 7,200 (91%)	6,820 / 7,200 (95%)	0 / 7,200 (0%)	0 / 3, (0%
Numbers may differ from published reports using final data. See Technical Notes. Data not shown due to low data quality.	2,449 / 7,200 (34%)	1,990 / 7,200 (28%)	1,843 / 7,200 (26%)	1,423 / 7,200 (20%)	767 / 7,200 (11%)	495 / 7,200 (6.9%)	232 / 7,200 (3.2%)	0 / 7,200 (0%)	0 / 3, (0%
Numbers may differ from published reports using final data. See Technical Notes. Data suppressed (<10).	99 / 7,200 (1.4%)	77 / 7,200 (1.1%)	90 / 7,200 (1.3%)	137 / 7,200 (1.9%)	155 / 7,200 (2.2%)	145 / 7,200 (2.0%)	146 / 7,200 (2.0%)	0 / 7,200 (0%)	0 / 3, (0%
Numbers may differ from published reports using final data. See Technical Notes. Data suppressed (<10). Data not shown due to low data quality.	188 / 7,200 (2.6%)	179 / 7,200 (2.5%)	119 / 7,200 (1.7%)	62 / 7,200 (0.9%)	34 / 7,200 (0.5%)	18 / 7,200 (0.3%)	2 / 7,200 (<0.1%)	0 / 7,200 (0%)	0 / 3, (0%
Underreported due to incomplete data.	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	2,902 / 7,200 (40%)	2,83 3,00 (95
Underreported due to incomplete data. Data not shown due to low data quality.	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	27 / 7,200 (0.4%)	71 / 3, (2.4
Underreported due to incomplete data. Data suppressed (<10).	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	71 / 7,200 (1.0%)	93 / 3, (3.1
Underreported due to incomplete data. Data suppressed (<10). Data not shown due to low data quality.	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	1 / 3, (<0.1
Footnote.Symbol									
	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	4,200 / 7,200 (58%)	0 / 3, (0%
*	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	0 / 7,200 (0%)	3,000 / 7,200 (42%)	3,00 3,00 (100
**	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	7,200 / 7,200 (100%)	0 / 7,200 (0%)	0 / 3, (0%
Predicted.Value	Min: 10, Max: 53,356, Mean: 1,006, SD: 4,080	Min: 10, Max: 64,932, Mean: 1,130, SD: 4,628	Min: 10, Max: 71,653, Mean: 1,339, SD: 5,564	Min: 10, Max: 71,006, Mean: 1,279, SD: 5,418	Min: 10, Max: 72,151, Mean: 1,203, SD: 5,145	Min: 10, Max: 93,655, Mean: 1,458, SD: 6,353	Min: 10, Max: 109,179, Mean: 1,750, SD: 7,801	Min: 10, Max: 110,759, Mean: 1,830, SD: 8,313	Min: Ma 112,0 Mea 1,872, 8,5
Unknown	3,896	3,512	3,312	2,868	2,207	1,920	1,649	1,471	61

[1] n / N (%); Min: Minimum, Max: Maximum, Mean: Mean, SD: SD

Figure 2: vssr_provisional_drug_overdose_death_counts

What kind of opioids is mainly responsible for drug overdose deaths?

We may found that T40.4, which is Synthetic opioids, excl. methadone is the leading cause of death among the indicators, accounting for 39.34%. Psychostimulants with abuse potential (T43.6) take the second place, and the next one is Cocaine (T40.5). Methadone (T40.3) represent the smallest percentage of deaths at 3.15%.

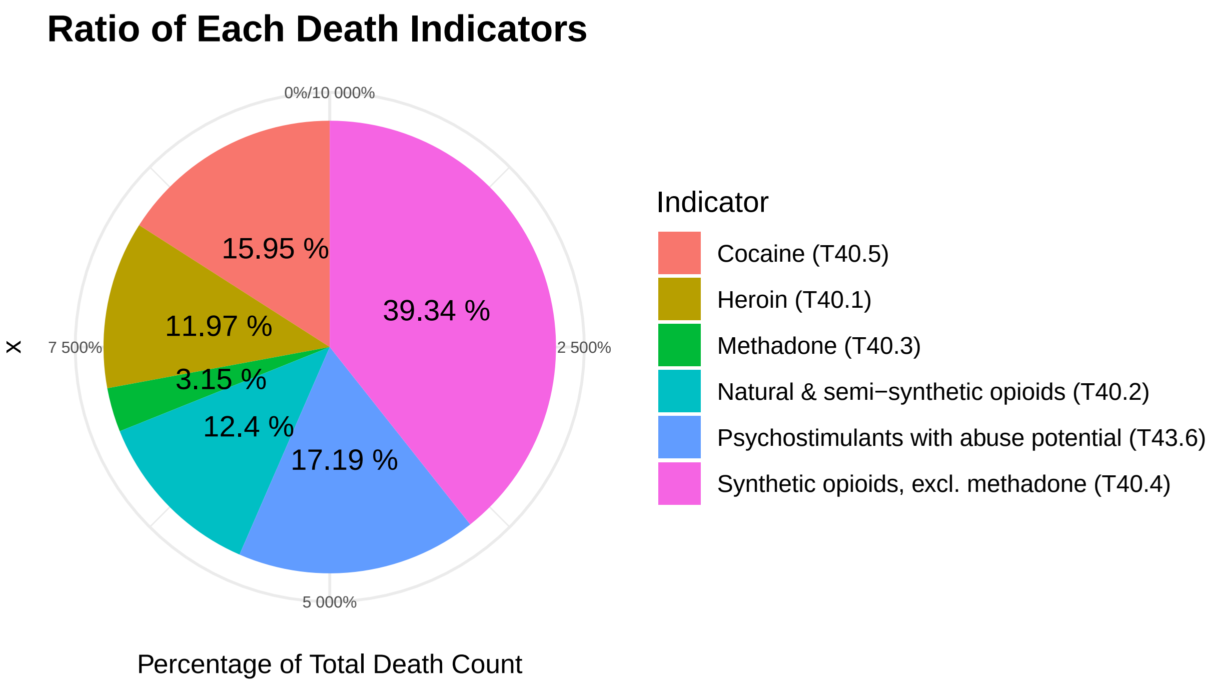
```
# replace all NA values with 0
vssr_provisional_drug_overdose_death_counts <- vssr_provisional_drug_overdose_death_counts %>%
  replace(is.na(.), 0) %>%
  mutate(State = ifelse(State == "YC", "NY", State))

kable(vssr_provisional_drug_overdose_death_counts %>%
        group_by(Indicator) %>%
        summarise(Count = sum(Predicted.Value)),
      caption = "vssr_provisional_drug_overdose_death_counts")
```

Table 11: vssr_provisional_drug_overdose_death_counts

Indicator	Count
Cocaine (T40.5)	2991135
Heroin (T40.1)	2244816
Methadone (T40.3)	591111
Natural & semi-synthetic opioids (T40.2)	2326269
Natural & semi-synthetic opioids, incl. methadone (T40.2, T40.3)	2804446
Natural, semi-synthetic, & synthetic opioids, incl. methadone (T40.2-T40.4)	9061513
Number of Deaths	0
Number of Drug Overdose Deaths	15886267
Opioids (T40.0-T40.4,T40.6)	10102520
Percent with drugs specified	0
Psychostimulants with abuse potential (T43.6)	3224112
Synthetic opioids, excl. methadone (T40.4)	7378065

```
# pie charts for indicators
vssr_provisional_drug_overdose_death_counts %>%
  filter(Indicator %in% c("Heroin (T40.1)",
                          "Natural & semi-synthetic opioids (T40.2)",
                          "Methadone (T40.3)",
                          "Synthetic opioids, excl. methadone (T40.4)",
                          "Cocaine (T40.5)",
                          "Psychostimulants with abuse potential (T43.6)")) %>%
  group_by(Indicator) %>%
  summarise(Count = sum(Predicted.Value)) %>%
  mutate(Percentage = Count / sum(Count) * 100) %>%
  ggplot(aes(x = "", y = Percentage, fill = Indicator)) +
  geom_bar(width = 1, stat = "identity") +
  geom_text(aes(label = paste(round(Percentage, 2), "%")),
            position = position_stack(vjust = 0.5)) +
  coord_polar("y", start = 0) +
  labs(
    title = "Ratio of Each Death Indicators",
    y = "Percentage of Total Death Count"
  ) +
  scale_y_continuous(labels = percent_format()) +
```

```
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 14, face = "bold"),
    axis.title = element_text(size = 10),
    axis.text = element_text(size = 6)
  )
```

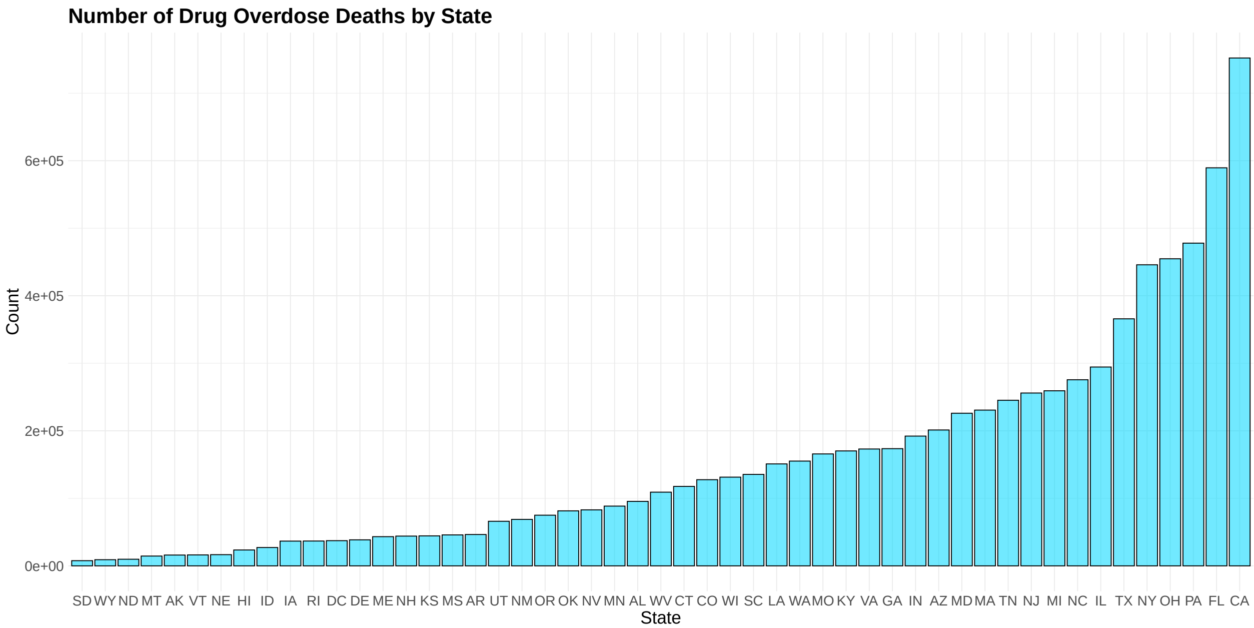


Which states holds the highest number of drug overdose deaths?

We can easily tell from the following graph that California, Florida, Pennsylvania, Ohio, and New York have the highest number of drug overdose deaths. However, we may also find those states have the largest population among U.S. from the census data. So, having the highest number of drug overdose deaths cannot simply proves that these states have the most serious drug problem. A further research is required.

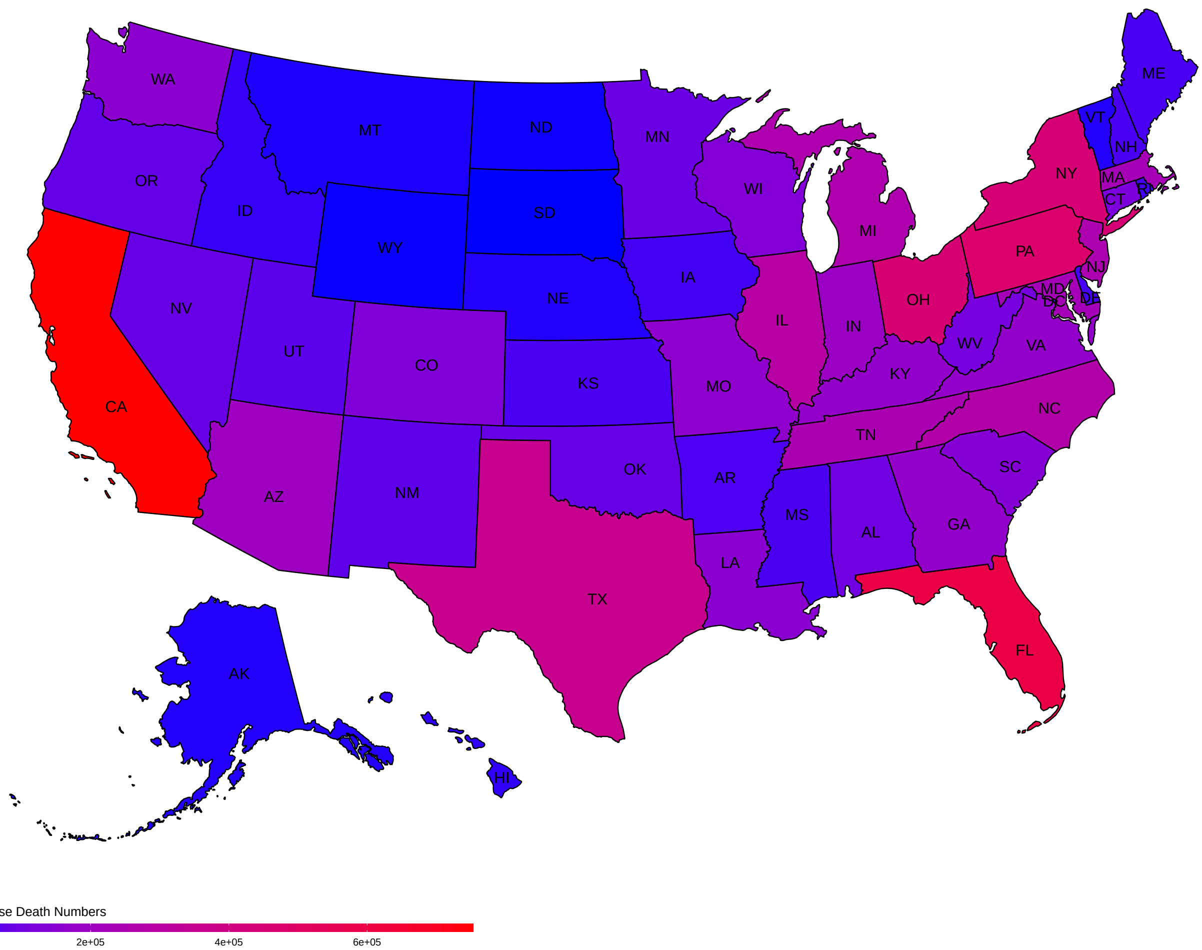
```
# Bar plot for Drug Overdose Deaths (ordered)
vssr_provisional_drug_overdose_death_counts %>%
  filter(Indicator == "Number of Drug Overdose Deaths" & State != "US") %>%
  group_by(State) %>%
  summarise(Count = sum(Predicted.Value)) %>%
  arrange(desc(Count)) %>%
  ggplot(aes(x = reorder(State, Count), y = Count)) +
  geom_bar(stat = "identity", fill = "#00d9ff8e", color = "black") +
  labs(
    title = "Number of Drug Overdose Deaths by State",
    x = "State",
    y = "Count"
  ) +
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 24, face = "bold"),
```

```
    axis.title = element_text(size = 20),
    axis.text = element_text(size = 16),
    legend.title = element_blank(),
    legend.text = element_blank()
  )
```




```
state_deaths <- vssr_provisional_drug_overdose_death_counts %>%
  filter(Indicator == "Number of Drug Overdose Deaths" & State != "US") %>%
  group_by(State) %>%
  summarise(Count = sum(Predicted.Value)) %>%
  mutate(state = State)

# Drug Overdose Deaths heatmap
plot_usmap(data = state_deaths, values = "Count", labels = TRUE) +
  scale_fill_gradientn(
    colours = c("blue", "red"), na.value = "grey90",
    guide = guide_colourbar(
      barwidth = 25, barheight = 0.4,
      title.position = "top"
    )
  ) +
  labs(fill = "Overdose Death Numbers") +
  theme(
    legend.position = "bottom",
    plot.title = element_text(size = 20, face = "bold")
  ) +
  ggtitle("Heatmap of Overdose Death Numbers by State")
```

Heatmap of Overdose Death Numbers by State



The following code will obtain census data from cencus data (2015 - 2021). The data will consist of the population of each state as well as the median of household income and poverty rate of each state.

```
# Obtain census data, including population, income and poverty rate
census_data <- data.frame()

for (year in 2015:2021) {
  population_data <- get_acs(
    geography = "state",
    variables = "B01003_001",
    year = year
  )

  income_data <- get_acs(
    geography = "state",
    variables = "B19013_001",
    year = year
  )

  poverty_data <- get_acs(
    geography = "state",
    variables = "S1701_C03_001",
```

```r
    year = year
  )

  # combine data together
  combined_data <- merge(population_data, income_data, by = c("GEOID", "NAME")) %>%
    merge(poverty_data, by = c("GEOID", "NAME")) %>%
    rename(POPULATION = estimate.x, INCOME = estimate.y, POVERTYRATE = estimate) %>%
    mutate(YEAR = year) %>%
    filter(GEOID != "72") %>%
    select(-c(moe.x, moe.y, moe, variable.x, variable.y, variable))

  census_data <- rbind(census_data, combined_data)
}
```

```
## Getting data from the 2011-2015 5-year ACS
## Getting data from the 2011-2015 5-year ACS
## Getting data from the 2011-2015 5-year ACS

## Using the ACS Subject Tables

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

## Using the ACS Subject Tables

## Getting data from the 2013-2017 5-year ACS
## Getting data from the 2013-2017 5-year ACS
## Getting data from the 2013-2017 5-year ACS

## Using the ACS Subject Tables

## Getting data from the 2014-2018 5-year ACS
## Getting data from the 2014-2018 5-year ACS
## Getting data from the 2014-2018 5-year ACS

## Using the ACS Subject Tables

## Getting data from the 2015-2019 5-year ACS
## Getting data from the 2015-2019 5-year ACS
## Getting data from the 2015-2019 5-year ACS

## Using the ACS Subject Tables

## Getting data from the 2016-2020 5-year ACS
## Getting data from the 2016-2020 5-year ACS
## Getting data from the 2016-2020 5-year ACS

## Using the ACS Subject Tables

## Getting data from the 2017-2021 5-year ACS
## Getting data from the 2017-2021 5-year ACS
## Getting data from the 2017-2021 5-year ACS

## Using the ACS Subject Tables
```

```r
state_info <- data.frame(
  STATE = state.abb,
  STATENAME = state.name
)
state_info <- rbind(state_info, c("DC", "District of Columbia"))
```

```r
census_data <- left_join(census_data, state_info, by = c("NAME" = "STATENAME")) %>%
  rename(STATENAME = NAME)

kable(census_data[1:10,], caption = "Census Data")
```

Table 12: Census Data

GEOID	STATENAME	POPULATION	INCOME	POVERTYRATE	YEAR	STATE
01	Alabama	4830620	43623	18.8	2015	AL
02	Alaska	733375	72515	10.2	2015	AK
04	Arizona	6641928	50255	18.2	2015	AZ
05	Arkansas	2958208	41371	19.3	2015	AR
06	California	38421464	61818	16.3	2015	CA
08	Colorado	5278906	60629	12.7	2015	CO
09	Connecticut	3593222	70331	10.5	2015	CT
10	Delaware	926454	60509	12.0	2015	DE
11	District of Columbia	647484	70848	18.0	2015	DC
12	Florida	19645772	47507	16.5	2015	FL

We shall re-order the cesus data by population to check which states have the biggest population in 2021.

```r
kable(census_data %>%
        filter(YEAR == 2021) %>%
        arrange(desc(POPULATION)) %>%
        head(10), caption = "Census Data (2021) ordered by Population")
```

Table 13: Census Data (2021) ordered by Population

GEOID	STATENAME	POPULATION	INCOME	POVERTYRATE	YEAR	STATE
06	California	39455353	84097	12.3	2021	CA
48	Texas	28862581	67321	14.0	2021	TX
12	Florida	21339762	61777	13.1	2021	FL
36	New York	20114745	75157	13.5	2021	NY
42	Pennsylvania	12970650	67587	11.8	2021	PA
17	Illinois	12821813	72563	11.8	2021	IL
39	Ohio	11769923	61938	13.4	2021	OH
13	Georgia	10625615	65030	13.9	2021	GA
37	North Carolina	10367022	60516	13.7	2021	NC
26	Michigan	10062512	63202	13.3	2021	MI

Which state's opioid overdose problem is the most serious?

A more reasonable way comparing to the total death counts is to use the death rate to demonstrate the level of effect by opioid. With the help of the census, we can easily get the death rate (death per 100000 people).

We may find that the `Indicator` *of* `Number of Drug Overdose Deaths` *in overdose deaths dataset is roughly equals to the summation of T40.0-T40.5,T43.6. In order to make the analysis more easy, we will use* `Number of Drug Overdose Deaths` *to substite the rest of the indicators.*

It seems that West Virginia and District of Columbia has the extreme high death rate. And the states following them are Tennessee, Kentucky, and Louisiana. It seems that a lot of states that have the highest death rates is concentrate in the mid east of the United States. Nevertheless, California, Florida, Pennsylvania, Ohio, and New York, which are the states we found in the previous step, are no longer significant. The result is by no means the same as what we get from the total death counts analysis.

```
overdose_death_year_state <- vssr_provisional_drug_overdose_death_counts %>%
  filter(Indicator == "Number of Drug Overdose Deaths" & State != "US") %>%
  filter(Year != 2023) %>%
  group_by(State, Year) %>%
  summarise(DEATHCOUNT = sum(Predicted.Value)) %>%
  rename("STATE" = State, "YEAR" = Year)
```

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

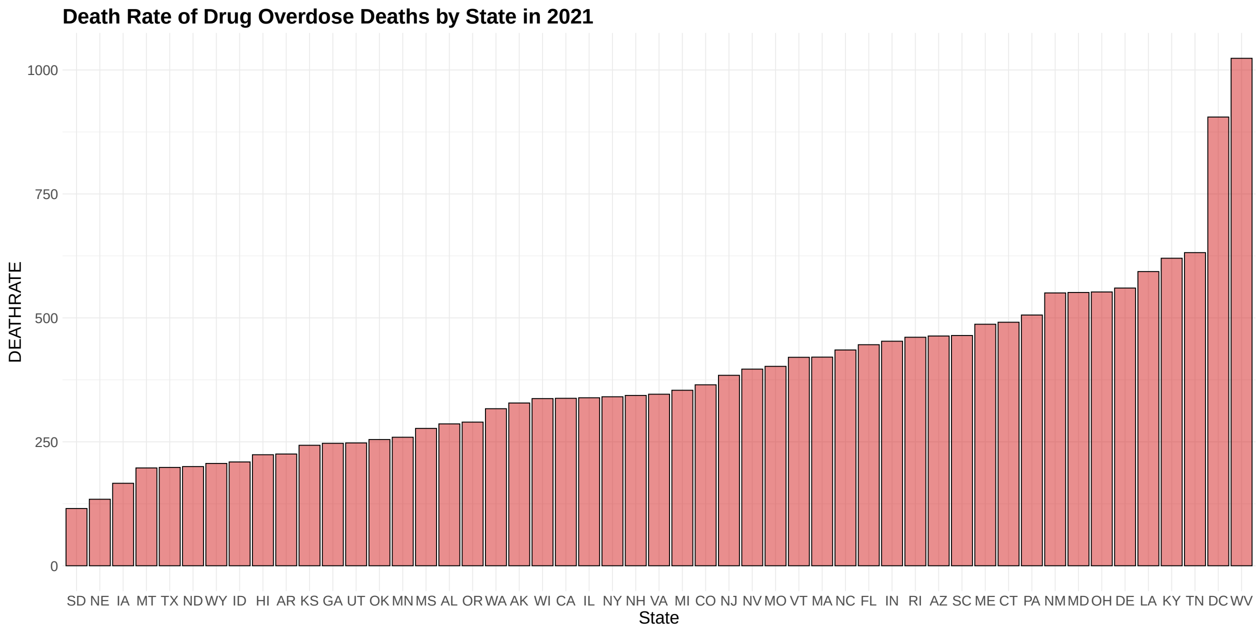
```
overdose_death_pop <- left_join(overdose_death_year_state, census_data, by = c("STATE", "YEAR"))

# Calculate the death rate
overdose_death_pop <- overdose_death_pop %>%
  mutate(DEATHRATE = DEATHCOUNT / POPULATION * 100000)
overdose_death_pop <- na.omit(overdose_death_pop)

kable(overdose_death_pop[1:10,], caption = "Death Rate")
```

Table 14: Death Rate

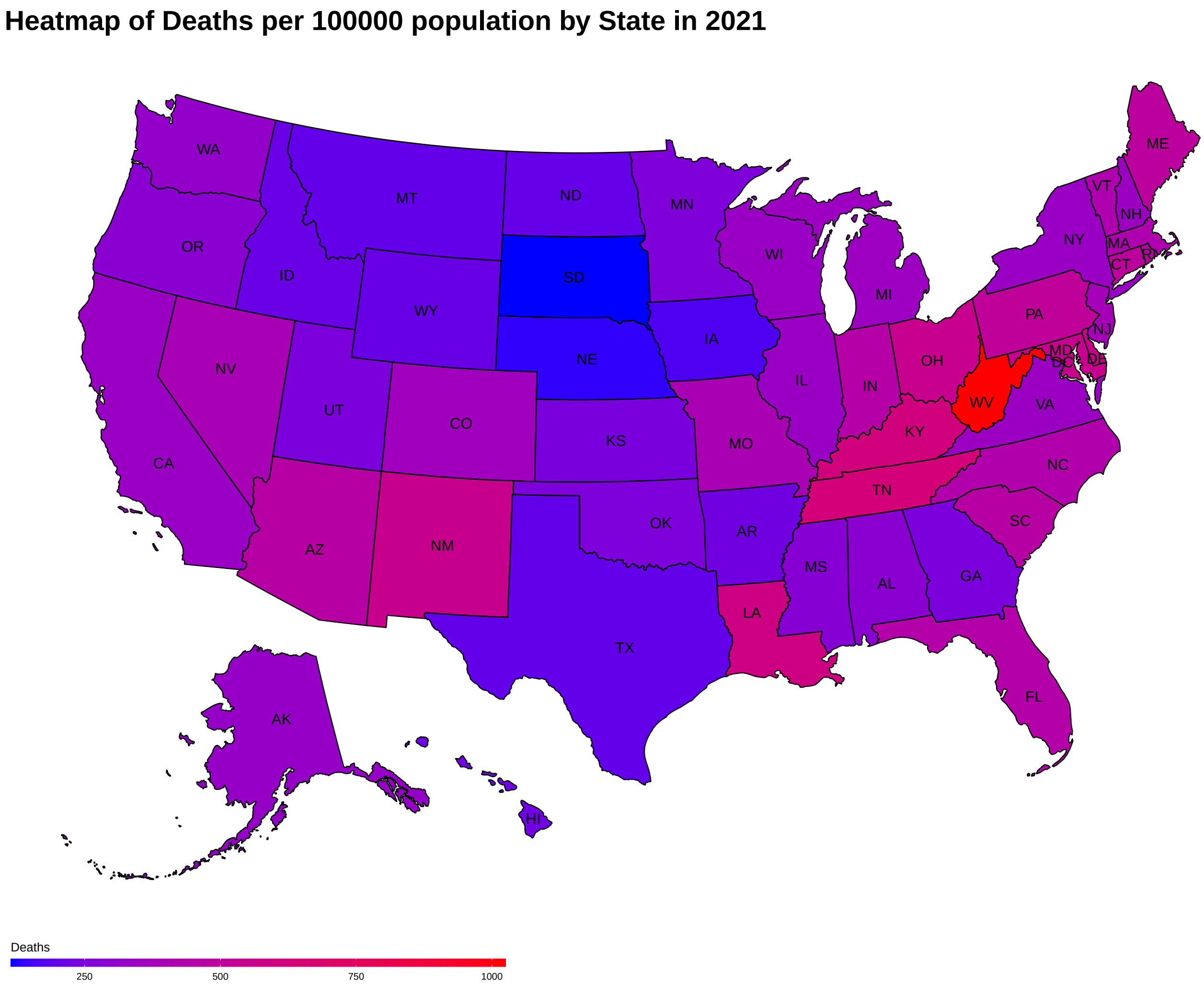
STATE	YEAR	DEATHCOUNT	GEOID	STATENAME	POPULATION	INCOME	POVERTYRATE	DEATHRATE
AK	2015	1472	02	Alaska	733375	72515	10.2	200.7159
AK	2016	1598	02	Alaska	736855	74444	10.1	216.8676
AK	2017	1579	02	Alaska	738565	76114	10.2	213.7930
AK	2018	1442	02	Alaska	738516	76715	10.8	195.2564
AK	2019	1544	02	Alaska	737068	77640	10.7	209.4786
AK	2020	1591	02	Alaska	736990	77790	10.3	215.8781
AK	2021	2416	02	Alaska	735951	80287	10.4	328.2827
AL	2015	9132	01	Alabama	4830620	43623	18.8	189.0441
AL	2016	8578	01	Alabama	4841164	44758	18.4	177.1888
AL	2017	9748	01	Alabama	4850771	46472	18.0	200.9577

```
# Bar plot for overdose death rate (ordered)
overdose_death_pop %>%
  filter(YEAR == 2021) %>%
  arrange(desc(DEATHRATE)) %>%
  ggplot(aes(x = reorder(STATE, DEATHRATE), y = DEATHRATE)) +
  geom_bar(stat = "identity", fill = "#d212127a", color = "black") +
  labs(
    title = "Death Rate of Drug Overdose Deaths by State in 2021",
    x = "State",
    y = "DEATHRATE"
  ) +
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 24, face = "bold"),
    axis.title = element_text(size = 20),
    axis.text = element_text(size = 16),
    legend.title = element_blank(),
```

```
    legend.text = element_blank()
  )
```




```
overdose_death_pop_2021 <- overdose_death_pop %>%
  filter(YEAR == 2021) %>%
  rename(state = STATE)

# Death rate heatmap
plot_usmap(data = overdose_death_pop_2021, values = "DEATHRATE", labels = TRUE) +
  scale_fill_gradientn(
    colours = c("blue", "red"), na.value = "grey90",
    guide = guide_colourbar(
      barwidth = 25, barheight = 0.4,
      title.position = "top"
    )
  ) +
  labs(fill = "Deaths") +
  theme(
    legend.position = "bottom",
    plot.title = element_text(size = 20, face = "bold")
  ) +
  ggtitle("Heatmap of Deaths per 100000 population by State in 2021")
```



What if people want to get rid of opioid? - Provdiers Rate by States

To help people that have opioid issues, the government of the United States has established a series of treatment programs. The dataset `Opioid Treatment Program Providers` included those information.

First of all, we need to find the starting year of each providers and then accumulate them.

```
# Transform the format of time
opioid_treatment_program_providers$YEAR <- format(
  as.POSIXct(opioid_treatment_program_providers$MEDICARE.ID.EFFECTIVE.DATE,
             format = "%m/%d/%Y"), "%Y")

# find the starting year of each providers and then accumulate them
provider_count <- opioid_treatment_program_providers %>%
  group_by(STATE, YEAR) %>%
  summarise(Count = n())
```

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

```
provider_sum <- expand.grid(
  STATE = unique(provider_count$STATE),
  YEAR = as.character(2015:2021)
```

```r
)

provider_sum <- provider_sum %>%
  left_join(provider_count, by = c("STATE", "YEAR"))

provider_sum <- provider_sum %>%
  arrange(STATE, YEAR) %>%
  group_by(STATE) %>%
  replace_na(list(Count = 0)) %>%
  mutate(Count = cumsum(Count)) %>%
  ungroup()

provider_sum$YEAR <- as.integer(provider_sum$YEAR)

kable(provider_sum[1:10,], caption = "Providers by States")
```

Table 15: Providers by States

STATE	YEAR	Count
AK	2015	0
AK	2016	0
AK	2017	0
AK	2018	0
AK	2019	0
AK	2020	5
AK	2021	5
AL	2015	0
AL	2016	0
AL	2017	0

Actually, it's not hard to find that the providers were not documented until 2020. To compensate that, we set the `Count` to 0 by default. Furthermore, data in Wyoming is not recorded.

To better measure how easy it is to get access to opioid treatment, we are going to calculate the number of providers per 100,000 people.

```r
# calculate the number of providers per 100,000 people.
census_provider <- left_join(provider_sum, census_data,
                             by = c("STATE" = "STATE", "YEAR" = "YEAR"))

census_provider <- census_provider %>%
  mutate(PROVIDERPER100000 = Count / POPULATION * 100000)

kable(census_provider[1:10,], caption = "Providers' data with Census")
```

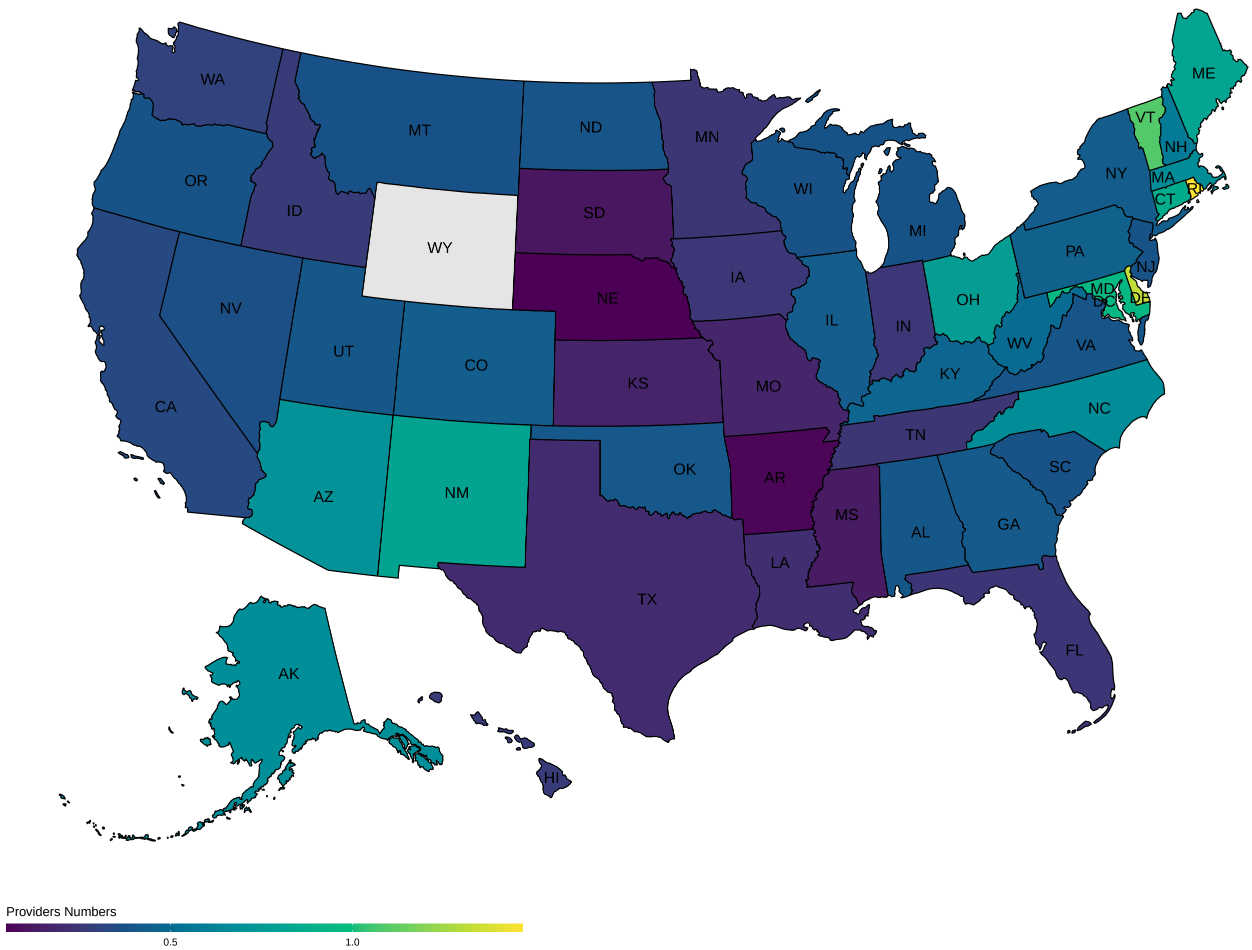
Table 16: Providers' data with Census

STATE	YEAR	Count	GEOID	STATENAME	POPULATION	INCOME	POVERTYRATE	PROVIDERPER100000
AK	2015	0	02	Alaska	733375	72515	10.2	0.0000000
AK	2016	0	02	Alaska	736855	74444	10.1	0.0000000
AK	2017	0	02	Alaska	738565	76114	10.2	0.0000000
AK	2018	0	02	Alaska	738516	76715	10.8	0.0000000
AK	2019	0	02	Alaska	737068	77640	10.7	0.0000000

STATE	YEAR	Count	GEOID	STATENAME	POPULATION	INCOME	POVERTYRATE	PROVIDERPER100000
AK	2020	5	02	Alaska	736990	77790	10.3	0.6784353
AK	2021	5	02	Alaska	735951	80287	10.4	0.6793931
AL	2015	0	01	Alabama	4830620	43623	18.8	0.0000000
AL	2016	0	01	Alabama	4841164	44758	18.4	0.0000000
AL	2017	0	01	Alabama	4850771	46472	18.0	0.0000000

```r
options(repr.plot.width = 20, repr.plot.height = 20)
provider_per_sum <- census_provider %>%
  filter(YEAR == 2021) %>%
  rename(state = STATE)

# provider rate heatmap
plot_usmap(data = provider_per_sum, values = "PROVIDERPER100000", labels = TRUE) +
  scale_fill_gradientn(
    colours = hcl.colors(10), na.value = "grey90",
    guide = guide_colourbar(
      barwidth = 25, barheight = 0.4,
      title.position = "top"
    )
  ) +
  labs(fill = "Providers Numbers") +
  theme(
    legend.position = "bottom",
    plot.title = element_text(size = 20, face = "bold")
  ) +
  ggtitle("Heatmap of Providers Numbers per 100000 population by State in 2021")
```

Heatmap of Providers Numbers per 100000 population by State in 2021



In the year of 2021, we may find that most of the states have already have several opioid treatment providers. And it is good to know that New England area has the highest rate of providers per 100000 population. But the number of providers per 100000 population is still very low in some states, like California and Florida. Which means people have opioid issues in those states may have trouble finding a opioid treatment provider.

Are those opioid treatment programs effective? - Death Rate VS Provider Rate

What people care about most is weather those treatment program helpful. The following analysis will use the death rate as the indicator of effectiveness. We will explore the relationship between the death rate and the treatment provider rate.

```r
# Joint data together, having a comprehesive data
prescribing_rates <- medicaid_opioid_prescribing_rates %>%
  filter(Geo_Desc != "National" & Year >= 2015) %>%
  select(Geo_Desc, Year, Plan_Type, Opioid_Prscrbng_Rate) %>%
  pivot_wider(names_from = Plan_Type, values_from = Opioid_Prscrbng_Rate) %>%
  rename(
    "YEAR" = Year,
    "STATENAME" = Geo_Desc
  )

joint_data <- left_join(prescribing_rates, overdose_death_pop,
                        by = c("STATENAME" = "STATENAME", "YEAR" = "YEAR")) %>%
```

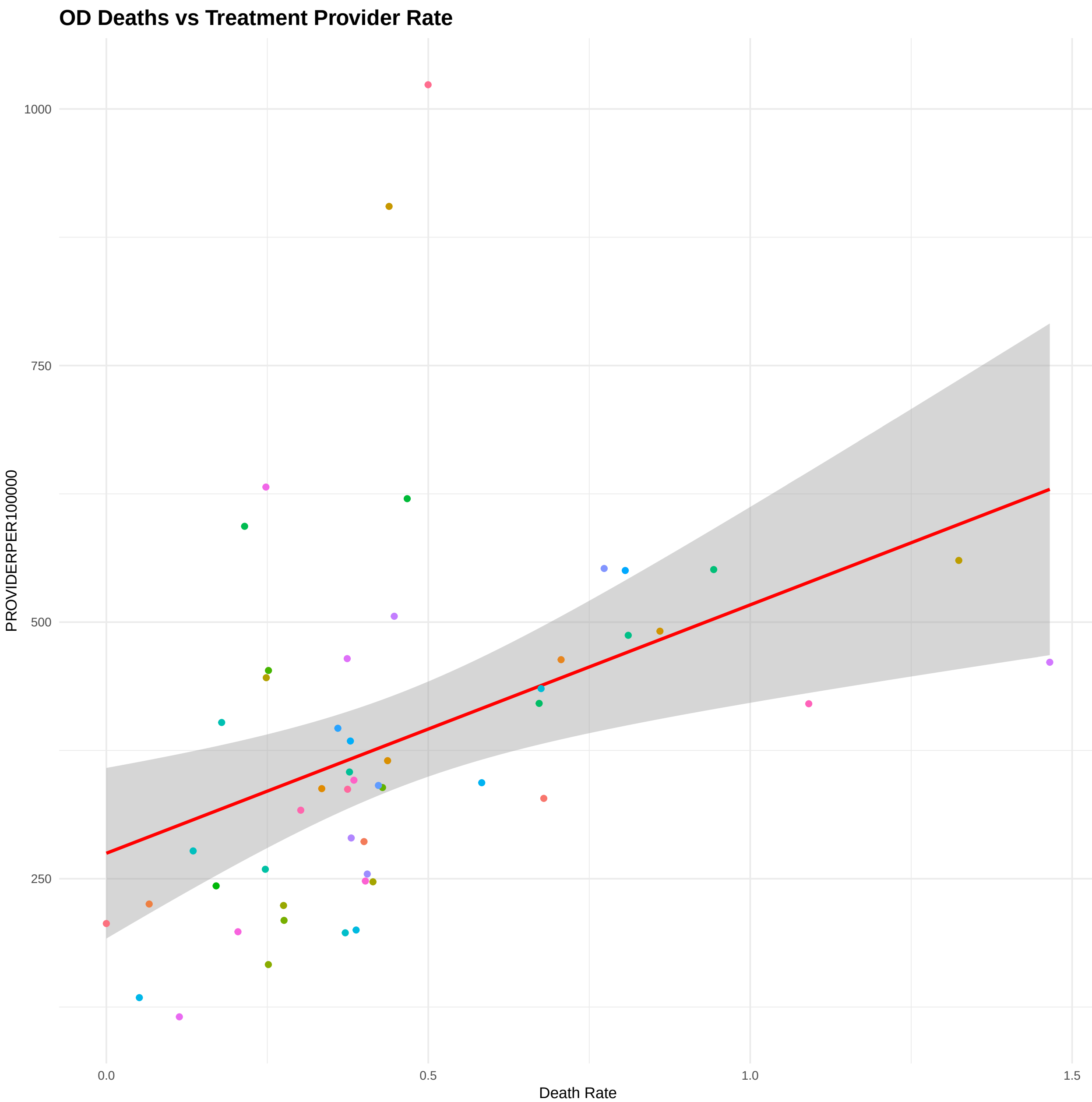
```r
  left_join(census_provider,
            by = c("STATENAME" = "STATENAME", "YEAR" = "YEAR"))

joint_data <- joint_data %>%
  select(-c("STATE.y", "POPULATION.y", "Count",
            "GEOID.x", "GEOID.y", "INCOME.y", "POVERTYRATE.y")) %>%
  rename("STATE" = STATE.x, "POPULATION" = POPULATION.x,
         "INCOME" = INCOME.x, "POVERTYRATE" = POVERTYRATE.x)

joint_data$PROVIDERPER100000[is.na(joint_data$PROVIDERPER100000)] <- 0

kable(joint_data %>%
        select(STATE, DEATHRATE, INCOME,
               POPULATION, POVERTYRATE, PROVIDERPER100000) %>%
        head(10), caption = "Joint Data")
```

Table 17: Joint Data

STATE	DEATHRATE	INCOME	POPULATION	POVERTYRATE	PROVIDERPER100000
AL	286.1931	54943	4997675	15.8	0.4001861
AK	328.2827	80287	735951	10.4	0.6793931
AZ	463.4137	65913	7079203	13.5	0.7062942
AR	225.3594	52123	3006309	16.0	0.0665268
CA	337.7970	84097	39455353	12.3	0.3345554
CO	365.0246	80184	5723176	9.6	0.4368204
CT	491.1894	83572	3605330	10.0	0.8598381
DE	560.1431	72724	981892	11.4	1.3239745
DC	905.0668	93547	683154	15.4	0.4391396
FL	445.8719	61777	21339762	13.1	0.2483627

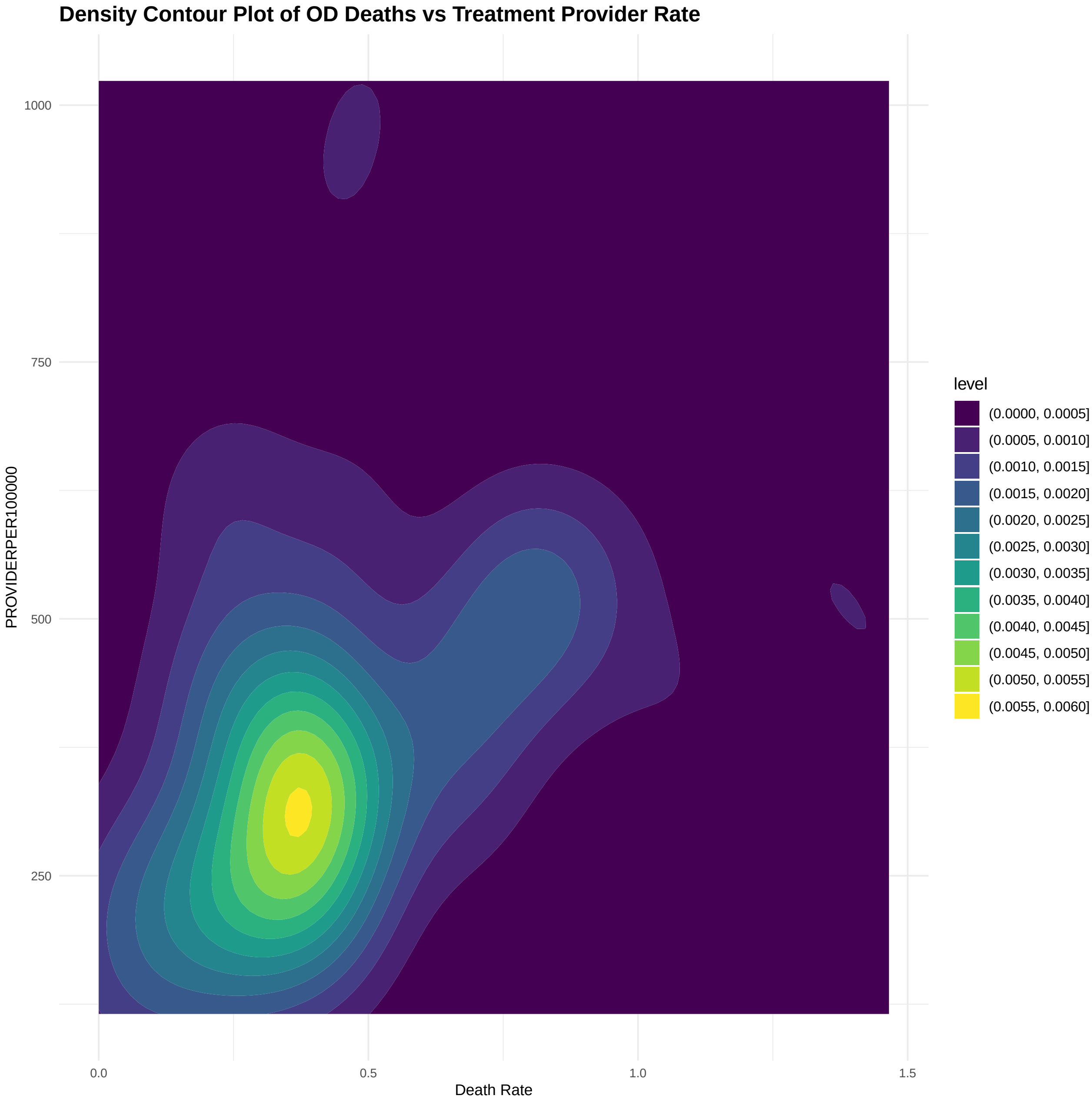
```r
# OD Deaths vs Treatment Provider Rate
joint_data %>%
  filter(YEAR == 2021) %>%
  ggplot(aes(x = PROVIDERPER100000, y = DEATHRATE, color = STATE)) +
  geom_smooth(method = "lm",
              se = TRUE,
              color = "red") +
  geom_point() +
  labs(
    title = "OD Deaths vs Treatment Provider Rate",
    x = "Death Rate",
    y = "PROVIDERPER100000"
  ) +
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 14, face = "bold"),
    axis.title = element_text(size = 10),
    axis.text = element_text(size = 8),
    legend.position = "none"
  )
```

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




```r
# Density Contour Plot of OD Deaths vs Treatment Provider Rate
joint_data %>%
  filter(YEAR == 2021) %>%
  ggplot(aes(x = PROVIDERPER100000, y = DEATHRATE)) +
  geom_density_2d_filled() +
  labs(
    title = "Density Contour Plot of OD Deaths vs Treatment Provider Rate",
    x = "Death Rate",
    y = "PROVIDERPER100000"
  ) +
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 14, face = "bold"),
```

```r
    axis.title = element_text(size = 10),
    axis.text = element_text(size = 8)
  )
```




```r
# correlation test
joint_data_2021 <- joint_data %>%
  filter(YEAR == 2021)

cor.test(joint_data_2021$DEATHRATE, joint_data_2021$PROVIDERPER100000)
```

```
##
##  Pearson's product-moment correlation
##
## data:  joint_data_2021$DEATHRATE and joint_data_2021$PROVIDERPER100000
```

```
## t = 3.1767, df = 49, p-value = 0.002578
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1553564 0.6184046
## sample estimates:
##       cor
## 0.4132474
```

Very interestingly, we find that the death rate and the provider rate are positively correlated ($cor = 0.413$, statistically significant). That is very counter intuitive because those providers are set to help people that have opioid use disorder (OUD) problems. They should be negatively correlated, which means with the help from those treatment providers, the death rate should be dropping. Yet, the analysis get the opposite result.

We may have a guess on the reasons: 1. It is not because providers are causing the overdose deaths, instead, overdose deaths are leading to more providers. 2. It is the providers that causing more overdose deaths: An introduction from Medicare.gov shows that medicare drug coverage (Part D) also covers drugs like buprenorphine (to treat opioid use disorders) and methadone (when prescribed for pain). That means those treatment programs are using another kind of opioid to treat the current opioid addiction, which may caused new addiction.

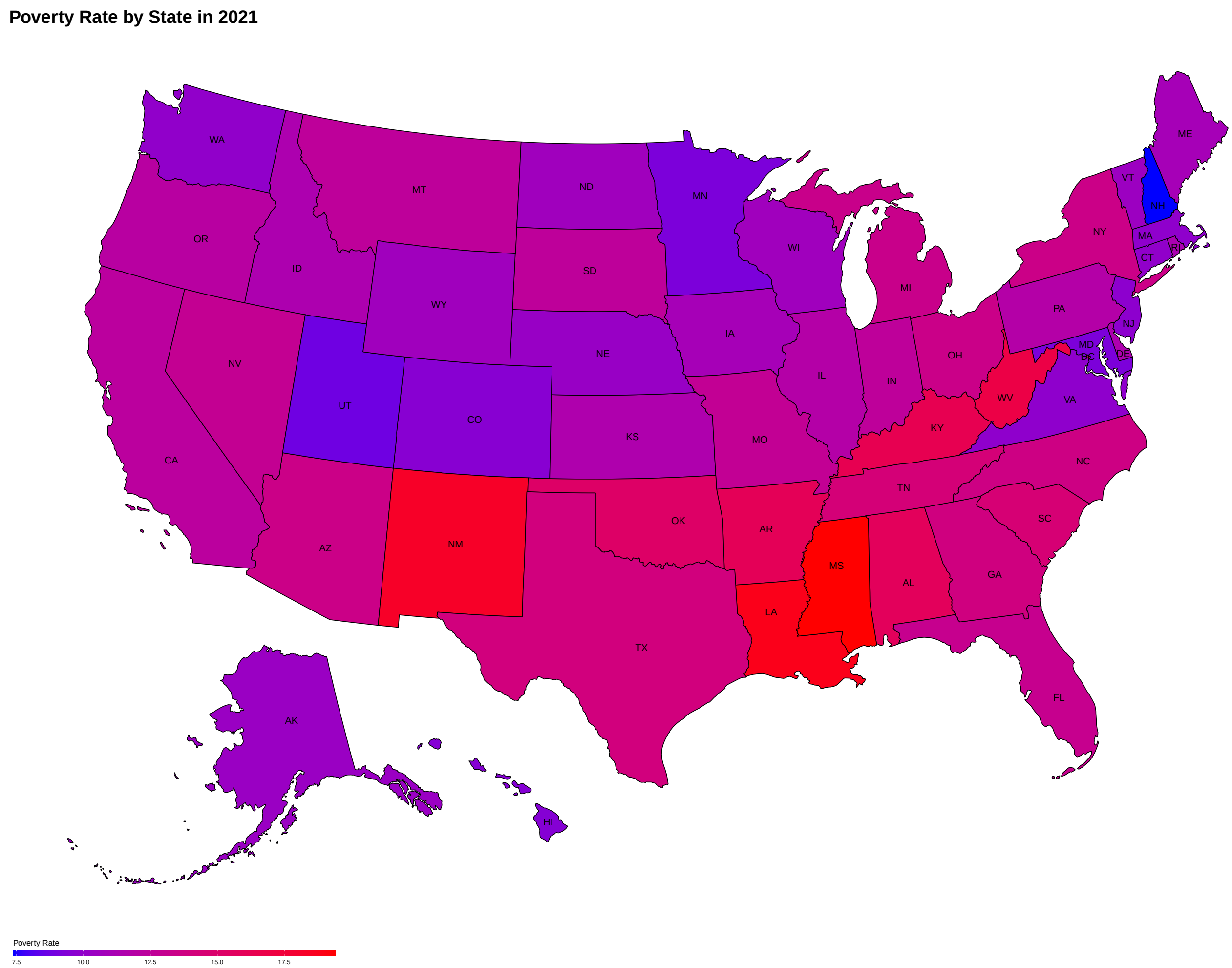
In conclusion, from current analysis, we can not judge if treatment programs are effective or not. Further analysis is needed.

What can we do in order to decrease opioid death rate?

Insight from Local Economic Data Combining with practical experience, it is reasonable to suggest that areas with lower average household income will have more people that have opioid use disorder (OUD) problems. Furthermore, areas with economic problems may not having enough treatment providers, casuing more overdose deaths.

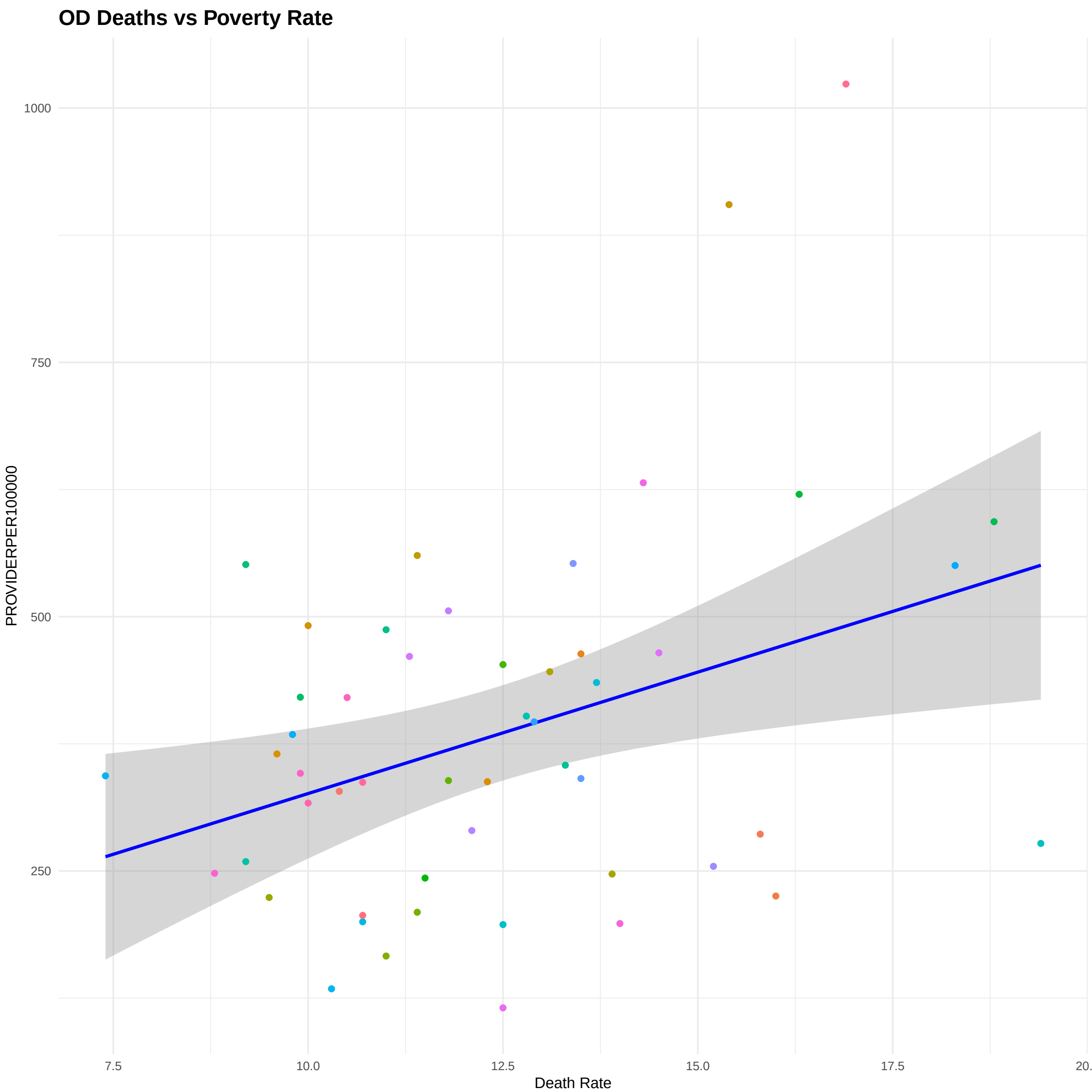
In the following part, we will examine the correlation between the income data and the death rate, also, local poverty rate will be introduced to make the analysis more reliable.

```r
income_data_2021 <- joint_data %>%
  filter(YEAR == 2021) %>%
  rename(state = STATE)

# Heatmap for poverty rate
plot_usmap(data = income_data_2021, values = "POVERTYRATE", labels = TRUE) +
  scale_fill_gradientn(
    colours = c("blue", "red"), na.value = "grey90",
    guide = guide_colourbar(
      barwidth = 25, barheight = 0.4,
      title.position = "top"
    )
  ) +
  labs(fill = "Poverty Rate") +
  theme(
    legend.position = "bottom",
    plot.title = element_text(size = 20, face = "bold")
  ) +
  ggtitle("Poverty Rate by State in 2021")
```




```r
# OD Deaths vs Poverty Rate
joint_data %>%
  filter(YEAR == 2021) %>%
  ggplot(aes(x = POVERTYRATE, y = DEATHRATE, color = STATE)) +
  geom_smooth(method = "lm",
              se = TRUE,
              color = "blue") +
  geom_point() +
  labs(
    title = "OD Deaths vs Poverty Rate",
    x = "Death Rate",
    y = "PROVIDERPER100000"
  ) +
  theme_minimal() +
  theme(
    plot.background = element_rect(fill = "white", colour = NA),
    plot.title = element_text(size = 14, face = "bold"),
    axis.title = element_text(size = 10),
    axis.text = element_text(size = 8),
    legend.position = "none"
  )
```

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



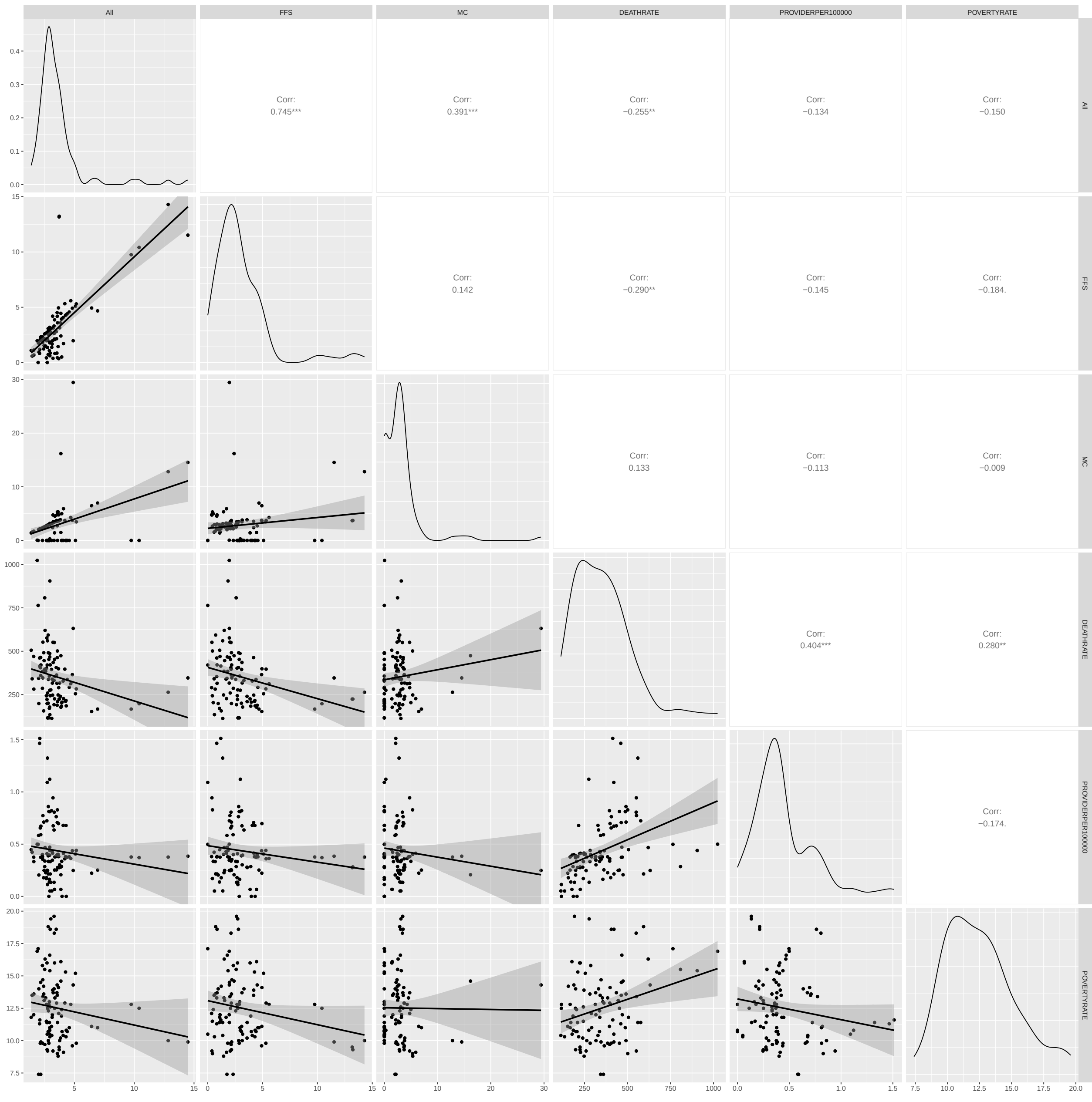

```r
# correlation test
cor.test(joint_data_2021$DEATHRATE, joint_data_2021$POVERTYRATE)
```

```
##
##  Pearson's product-moment correlation
##
## data:  joint_data_2021$DEATHRATE and joint_data_2021$POVERTYRATE
## t = 2.7031, df = 49, p-value = 0.009416
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.09397278 0.57839252
## sample estimates:
##       cor
## 0.3602275
```

As we can see, poverty rate and death rate are positively correlated ($cor = 0.36$, statistically significant), which meet our hypothesis. That means, the higher the poverty rate of a state, the higher overdose death rate it will have.

Insight from the Whole Picture In the next part, we will use pair plots to show the correlation of each variables in our dataset. By doing that, we may have a better view for our analysis.

```r
# pair plot for whole data
ggpairs(joint_data %>% filter(YEAR %in% c(2021, 2020)),
        columns = c("All", "FFS", "MC", "DEATHRATE", "PROVIDERPER100000", "POVERTYRATE"),
        lower = list(continuous = "smooth"))
```



Here is the summary table for the analysis data.

```r
tbl_summary(joint_data,
            by = YEAR, include = c("DEATHCOUNT", "DEATHRATE", "POPULATION", "INCOME",
                                   "POVERTYRATE", "All", "FFS", "MC", "PROVIDERPER100000"),
            statistic = list(
              DEATHCOUNT ~ "Mean: {mean}, Sum: {sum}, SD: {sd}",
              DEATHRATE ~ "Mean: {mean}, SD: {sd}",
              POPULATION ~ "Sum: {sum}, Mean: {mean}",
              INCOME ~ "Median: {median}, Mean: {mean}, SD: {sd}",
              POVERTYRATE ~ "Median: {median}, Mean: {mean}, SD: {sd}",
              All ~ "Mean: {mean}, SD: {sd}",
              FFS ~ "Mean: {mean}, SD: {sd}",
              MC ~ "Mean: {mean}, SD: {sd}",
              PROVIDERPER100000 ~ "Mean: {mean}, SD: {sd}"
            )
) %>%
  as_gt() %>%
  gt::gtsave(
    filename = "table_image.png"
  )
```

Characteristic	**2015**, N = 51[1]	**2016**, N = 51[1]	**2017**, N = 51[1]	**2018**, N = 51[1]	**2019**, N = 51[1]	**2020**, N = 51[1]	**2021**, N = 51[1]
DEATHCOUNT	Mean: 11,979, Sum: 610,938, SD: 11,784	Mean: 13,883, Sum: 708,036, SD: 13,949	Mean: 16,451, Sum: 839,009, SD: 17,394	Mean: 16,419, Sum: 837,371, SD: 16,741	Mean: 16,356, Sum: 834,135, SD: 16,728	Mean: 19,797, Sum: 1,009,644, SD: 20,795	Mean: 24,203 Sum: 1,234,342, SD: 25,910
DEATHRATE	Mean: 206, SD: 75	Mean: 235, SD: 95	Mean: 271, SD: 132	Mean: 268, SD: 123	Mean: 268, SD: 119	Mean: 319, SD: 149	Mean: 384, SD: 177
POPULATION	Sum: 316,515,021, Mean: 6,206,177	Sum: 318,558,162, Mean: 6,246,238	Sum: 321,004,407, Mean: 6,294,204	Sum: 322,903,030, Mean: 6,331,432	Sum: 324,697,795, Mean: 6,366,623	Sum: 326,569,308, Mean: 6,403,320	Sum: 329,725,481, Mean: 6,465,206
INCOME	Median: 52,997, Mean: 54,636, SD: 9,157	Median: 54,384, Mean: 56,031, SD: 9,406	Median: 56,570, Mean: 58,236, SD: 9,850	Median: 59,116, Mean: 60,621, SD: 10,297	Median: 61,439, Mean: 63,098, SD: 10,715	Median: 63,015, Mean: 65,045, SD: 11,052	Median: 66,644, Mean: 68,872 SD: 11,471
POVERTYRATE	Median: 15.20, Mean: 14.85, SD: 3.17	Median: 14.90, Mean: 14.53, SD: 3.12	Median: 14.20, Mean: 14.08, SD: 3.03	Median: 13.70, Mean: 13.65, SD: 2.93	Median: 13.10, Mean: 13.13, SD: 2.83	Median: 12.40, Mean: 12.60, SD: 2.72	Median: 12.10 Mean: 12.45, SD: 2.67
All	Mean: 6.46, SD: 1.32	Mean: 5.87, SD: 1.26	Mean: 5.01, SD: 1.15	Mean: 4.19, SD: 1.25	Mean: 3.65, SD: 1.55	Mean: 3.57, SD: 1.84	Mean: 3.49, SD: 2.11
FFS	Mean: 6.04, SD: 3.19	Mean: 5.30, SD: 3.18	Mean: 4.57, SD: 3.02	Mean: 3.96, SD: 2.76	Mean: 3.50, SD: 3.14	Mean: 3.18, SD: 2.74	Mean: 2.89, SD: 2.62
MC	Mean: 4.49, SD: 3.45	Mean: 4.30, SD: 3.09	Mean: 3.71, SD: 2.56	Mean: 2.98, SD: 2.12	Mean: 2.63, SD: 2.04	Mean: 2.74, SD: 2.99	Mean: 3.00, SD: 4.48
PROVIDERPER100000	Mean: 0.00, SD: 0.00	Mean: 0.00, SD: 0.00	Mean: 0.00, SD: 0.00	Mean: 0.00, SD: 0.00	Mean: 0.00, SD: 0.00	Mean: 0.42, SD: 0.28	Mean: 0.45, SD: 0.30

[1] Mean: Mean, Sum: Sum, SD: SD; Mean: Mean, SD: SD; Sum: Sum, Mean: Mean; Median: Median, Mean: Mean, SD: SD

Figure 3: table_output

Now we can tell the correlation between variables:

Variables Compared	Correlation (Corr)	Description
DEATHRATE vs. All	-0.255**	Mild positive correlation.
DEATHRATE vs. FFS	-0.290**	Mild negative correlation.
DEATHRATE vs. MC	0.113	Very weak positive correlation.
DEATHRATE vs. PROVIDERS	0.404***	Strong positive correlation.
DEATHRATE vs. POVERTYRATE	0.280**	Mild positive correlation.

The result suggest that, if we want to decrease the overdose death rate, we may want to increase the prescription rate of `All` and `FFS` plan. Also, by a more direct way, setting more opioid treatment providers as well as improve state's economic status will both help decrease overdose death rate.

Conclusion

In this analysis, we first imported three datasets, namely Medicaid Opioid Prescribing Rates, Opioid Treatment Program Providers, and VSRR Provisional Drug Overdose Death Counts. Then we cleaned the datasets, which included removing missing values, eliminating unnecessary columns, and performing some basic data processing. Next, we conducted some simple exploratory analyses on the datasets, including an analysis of prescribing rates, an analysis of prescription providers, and an analysis of death rates. Finally, we carried out some basic statistical analyses of the datasets, including a correlation analysis and some visual analyses.

During the exploratory analysis, we found that the leading cause of opioid-related deaths nationwide is Synthetic opioids. We also discovered that California and Florida have the highest number of opioid-related deaths, but the highest death rates are in West Virginia and Washington DC. In analyzing prescription providers, we observed that the New England region has the most prescription providers. However, in populous states like California and Florida, the number of prescription providers is notably low. In our death rate analysis, we noticed a positive correlation between death rates and the number of prescription providers, which is counterintuitive to our initial assumptions. After conducting further statistical analyses, we identified a positive correlation between death rates and poverty rates, aligning with our expectations. Finally, by constructing a correlation matrix to analyze the relationships between various variables, we concluded that if we aim to reduce opioid-related death rates, we can achieve this by increasing the number of prescription providers, raising prescription rates, and improving the economic conditions of the state.