2007-2017 OASAS Admissions Analytics

Caitlin Kirkpatrick

```
# Documentation from data.world recommends package installation from GitHub
devtools::install_github("datadotworld/data.world-r", build_vignettes = TRUE)
```

```
## Using GitHub PAT from the git credential store.

## Skipping install of 'data.world' from a github remote, the SHA1 (a1fd7656) has not changed since las
##   Use 'force = TRUE' to force installation
```

```
# Load the requisite API token obtained from data.world advanced settings
token <- readLines('api')
saved_cfg <- data.world::save_config(token)
data.world::set_config(saved_cfg)

# From data.world R and RStudio integration:
library("data.world")
```

```
## Loading required package: dwapi

##
## Attaching package: 'dwapi'

## The following object is masked from 'package:usethis':
##
##     create_project

## The following object is masked from 'package:dplyr':
##
##     sql
```

```
sql_stmt <- data.world::qry_sql("SELECT * FROM chemical_dependence_treatment_program_admissions_beginni
admissions_df <- data.world::query(
  sql_stmt, "https://data.world/data-ny-gov/ngbt-9rwf")
```

```
## Rows: 72463 Columns: 7

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): county_of_program_location, program_category, service_type, age_gro...
## dbl (2): year, admissions
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Data Understanding

```
glimpse(admissions_df)
```

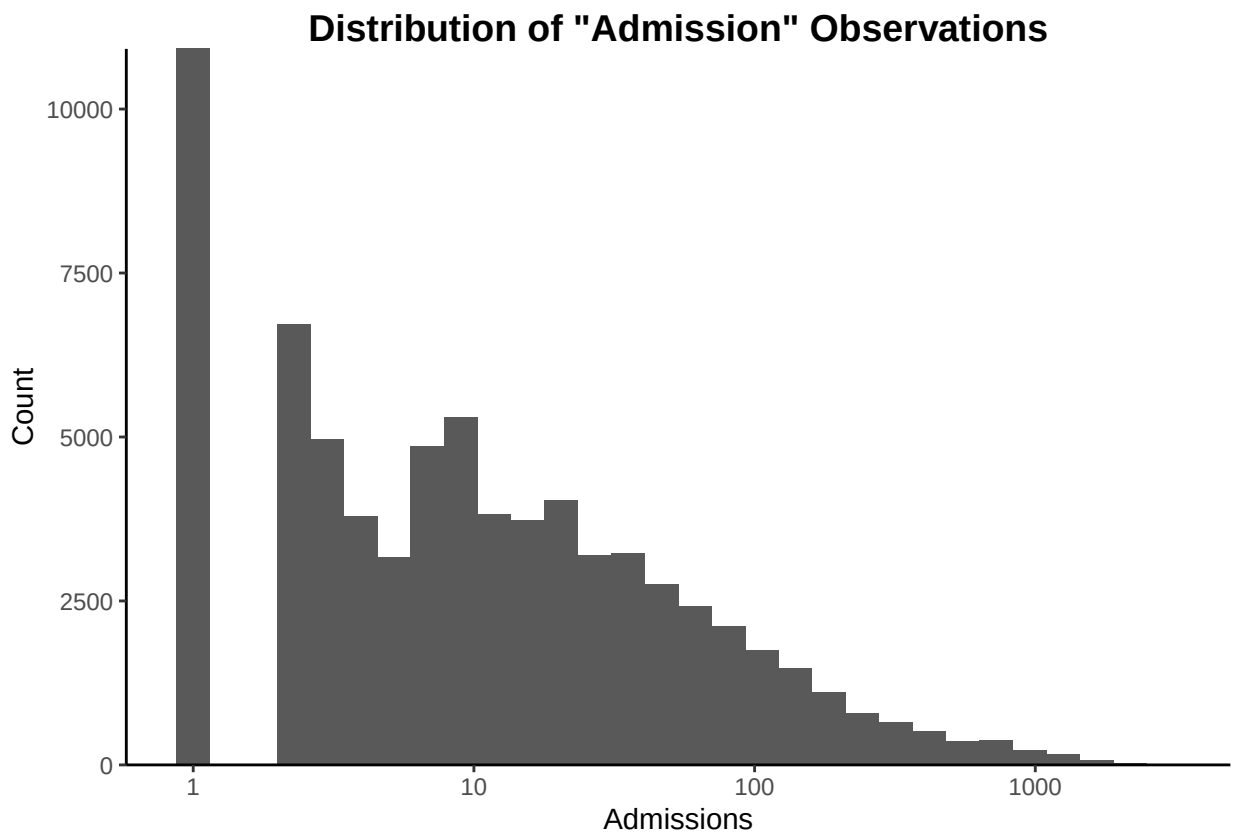
```
## Rows: 72,463
## Columns: 7
## $ year                        <dbl> 2017, 2017, 2017, 2017, 2017, 2017, 2017, 2~
## $ county_of_program_location <chr> "Albany", "Albany", "Albany", "Albany", "Al~
## $ program_category            <chr> "Crisis", "Crisis", "Crisis", "Crisis", "Cr~
## $ service_type                <chr> "Medically Managed Detoxification", "Medica~
## $ age_group                   <chr> "18 thru 24", "18 thru 24", "18 thru 24", "~
## $ primary_substance_group     <chr> "Alcohol", "All Others", "Cocaine incl Crac~
## $ admissions                  <dbl> 25, 7, 1, 64, 20, 140, 10, 3, 1, 41, 22, 26~
```

```
summary(admissions_df)
```

```
##       year      county_of_program_location program_category
##  Min.   :2007   Length:72463               Length:72463
##  1st Qu.:2009   Class :character           Class :character
##  Median :2012   Mode  :character           Mode  :character
##  Mean   :2012
##  3rd Qu.:2015
##  Max.   :2017
##  service_type        age_group         primary_substance_group
##  Length:72463       Length:72463       Length:72463
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##    admissions
##  Min.   :   1.00
##  1st Qu.:   3.00
##  Median :   8.00
##  Mean   :  44.62
##  3rd Qu.:  30.00
##  Max.   :2862.00
```

```
# Visualize outliers in the admissions column
ggplot(admissions_df, aes(x = admissions)) +
  geom_histogram() +
  labs(title = "Distribution of \"Admission\" Observations",
       x = "Admissions",
       y = "Count") +
  scale_x_log10() +
  scale_y_continuous(expand = c(0,0)) +
  theme_classic() +
  theme(
    plot.title = element_text(face = "bold", size = 14, hjust = 0.5, margin = margin(t = 5))
  )
```

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




```
# Column summaries
program_category_summary <- admissions_df %>%
  group_by(program_category) %>%
  summarize(
    min_admissions = min(admissions),
    median_admissions = median(admissions),
    mean_admissions = mean(admissions),
    max_admissions = max(admissions)
  )
print(program_category_summary)
```

```
## # A tibble: 5 x 5
##   program_category         min_admissions median_admissions mean_admissions
##   <chr>                             <dbl>              <dbl>           <dbl>
## 1 Crisis                                1                 15            67.2
## 2 Inpatient                             1                 15            43.8
## 3 Opioid Treatment Program              1                  9            55.4
## 4 Outpatient                            1                 11            56.1
## 5 Residential                           1                  3            11.7
## # i 1 more variable: max_admissions <dbl>
```

```
service_type_summary <- admissions_df %>%
  group_by(service_type) %>%
  summarize(
    min_admissions = min(admissions),
```

```
    median_admissions = median(admissions),
    mean_admissions = mean(admissions),
    max_admissions = max(admissions)
  )
print(service_type_summary)
```

```
## # A tibble: 28 x 5
##    service_type  min_admissions median_admissions mean_admissions max_admissions
##    <chr>                  <dbl>             <dbl>           <dbl>          <dbl>
##  1 Community Re~              1                 3            6.70            143
##  2 Inpatient Re~              1                15           43.8            1106
##  3 Intensive Re~              1                 7           25.0             516
##  4 Limited Outp~              1                 7           15.0             151
##  5 Long Term Re~              1                 2            4.79             31
##  6 Med Sup With~              1                14           70.6            2058
##  7 Med Sup With~              1                 8           32.1             341
##  8 Medically Ma~              1                19           95.1            2862
##  9 Medically Mo~              1                13           38.9            2516
## 10 Meth to Abst~              1                16           27.4              79
## # i 18 more rows
```

```
age_group_summary <- admissions_df %>%
  group_by(age_group) %>%
  summarize(
    min_admissions = min(admissions),
    median_admissions = median(admissions),
    mean_admissions = mean(admissions),
    max_admissions = max(admissions)
  )
print(age_group_summary)
```

```
## # A tibble: 6 x 5
##   age_group    min_admissions median_admissions mean_admissions max_admissions
##   <chr>                 <dbl>             <dbl>           <dbl>          <dbl>
## 1 18 thru 24                1                 8            31.3           1518
## 2 25 thru 34                1                13            52.9           1876
## 3 35 thru 44                1                10            53.1           2862
## 4 45 thru 54                1                 8            58.8           2716
## 5 55 and Older              1                 5            30.0           1277
## 6 Under 18                  1                 4            23.5            661
```

```
primary_substance_group_summary <- admissions_df %>%
  group_by(primary_substance_group) %>%
  summarize(
    min_admissions = min(admissions),
    median_admissions = median(admissions),
    mean_admissions = mean(admissions),
    max_admissions = max(admissions)
  )
print(primary_substance_group_summary)
```

```
## # A tibble: 6 x 5
```

```
##   primary_substance_group min_admissions median_admissions mean_admissions
##   <chr>                             <dbl>             <dbl>           <dbl>
## 1 Alcohol                               1                21           91.6
## 2 All Others                            1                 3            9.92
## 3 Cocaine incl Crack                    1                 7           29.3
## 4 Heroin                                1                13           55.0
## 5 Marijuana incl Hashish                1                 8           46.0
## 6 Other Opioids                         1                 6           15.4
## # i 1 more variable: max_admissions <dbl>
```

```r
# Re-classification of categorical variables
admissions_df <- admissions_df %>%
  mutate(
    program_category = as.factor(program_category),
    service_type = as.factor(service_type),
    age_group = as.factor(age_group),
    primary_substance_group = as.factor(primary_substance_group)
  )

# Manual outlier identification
admissions_outliers <- admissions_df %>%
  mutate(
    mean_admissions = mean(admissions, na.rm = TRUE),
    sd_admissions = sd(admissions, na.rm = TRUE)
  ) %>%
  # Relative to the mean, any values on the lower or upper bounds that are 3 times the standard deviati
  filter(admissions < mean_admissions - 3 * sd_admissions | admissions > mean_admissions + 3 * sd_admiss
  select(admissions)
admissions_outliers
```

```
## # A tibble: 1,380 x 1
##    admissions
##         <dbl>
##  1        526
##  2        468
##  3        515
##  4        501
##  5        752
##  6        496
##  7        566
##  8        442
##  9        564
## 10        469
## # i 1,370 more rows
```

```r
# Remove outliers
rmv_admissions_outliers <- admissions_df %>%
  mutate(
    mean_admissions = mean(admissions, na.rm = TRUE),
    sd_admissions = sd(admissions, na.rm = TRUE)
  ) %>%
  filter(!(admissions < mean_admissions - 3 * sd_admissions | admissions > mean_admissions + 3 * sd_admi
```

```r
# Note subtracted outliers from new dataframe
str(admissions_df)
```

```
## tibble [72,463 x 7] (S3: tbl_df/tbl/data.frame)
##  $ year                       : num [1:72463] 2017 2017 2017 2017 2017 ...
##  $ county_of_program_location: chr [1:72463] "Albany" "Albany" "Albany" "Albany" ...
##  $ program_category           : Factor w/ 5 levels "Crisis","Inpatient",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ service_type               : Factor w/ 28 levels "Community Residential",..: 8 8 8 8 8 8 8 8 8 8 .
##  $ age_group                  : Factor w/ 6 levels "18 thru 24","25 thru 34",..: 1 1 1 1 1 2 2 4 4 4
##  $ primary_substance_group    : Factor w/ 6 levels "Alcohol","All Others",..: 1 2 3 4 6 1 2 2 3 4 ...
##  $ admissions                 : num [1:72463] 25 7 1 64 20 140 10 3 1 41 ...
```

```r
str(rmv_admissions_outliers$admissions)
```

```
##  num [1:71083] 25 7 1 64 20 140 10 3 1 41 ...
```

- It may not be significant that admissions numbers are higher than usual at any given point in time

---

Data Preparation

- Data is structured relationally prior to analysis

```r
# Read in .csv created from https://www.dot.ny.gov/main/business-center/engineering/specifications/loca
county_codes <- read_csv("county_codes.csv")
```

```
## Rows: 62 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): county_of_program_location, county_code
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
county_codes
```

```
## # A tibble: 62 x 2
##    county_of_program_location county_code
##    <chr>                      <chr>
##  1 Albany                     AL
##  2 Cattaraugus                CA
##  3 Chenango                   CN
##  4 Delaware                   DE
##  5 Franklin                   FR
##  6 Hamilton                   HA
##  7 Lewis                      LE
##  8 Montgomery                 MG
##  9 Oneida                     ON
## 10 Orleans                    OL
## # i 52 more rows
```

```r
county <- admissions_df %>%
  select(county_of_program_location) %>%
  distinct() %>%
  # Join codes with corresponding counties
  left_join(county_codes, by = "county_of_program_location") %>%
  mutate(county_code = case_when(
      # Tagging counties with first two characters and "-NYC"
      county_of_program_location %in% c("Bronx", "Queens", "Kings") ~ paste(str_to_upper(str_sub(county
      county_of_program_location == "New York" ~ "NYC",
      # Handle to not treat as NA value, changed code to NS instead
      county_of_program_location == "Nassau" ~ "NS",
      # Handle to not treat as NA value, manually assigned SL
      county_of_program_location == "St Lawrence" ~ "SL",
      TRUE ~ county_code
    )
  )

# Note: The county "Hamilton" is included in the county_codes csv, but it is not found in admissions_df
county
```

```
## # A tibble: 61 x 2
##    county_of_program_location county_code
##    <chr>                      <chr>
##  1 Albany                     AL
##  2 Bronx                      BR-NYC
##  3 Broome                     BM
##  4 Dutchess                   DU
##  5 Erie                       ER
##  6 Kings                      KI-NYC
##  7 Monroe                     MO
##  8 Nassau                     NS
##  9 New York                   NYC
## 10 Niagara                    NI
## # i 51 more rows
```

- county contains county names corresponding to a county code

```r
# Define abbreviations
program_category_index <- c(
  "Crisis" = "C",
  "Inpatient" = "I",
  "Opioid Treatment Program" = "OTP",
  "Outpatient" = "O",
  "Residential" = "R"
)

# Add program_code column
admissions_df <- admissions_df %>%
  mutate(program_code = recode(program_category,
       "Crisis" = "C",
       "Inpatient" = "I",
       "Opioid Treatment Program" = "OTP",
       "Outpatient" = "O",
```

```
       "Residential" = "R"))

# Add program_category tibble
program_category <- admissions_df %>%
  distinct(program_category, .keep_all = TRUE) %>%
  select(program_code, program_category)

program_category
```

```
## # A tibble: 5 x 2
##   program_code program_category
##   <fct>        <fct>
## 1 C            Crisis
## 2 I            Inpatient
## 3 OTP          Opioid Treatment Program
## 4 O            Outpatient
## 5 R            Residential
```

- program_category contains identifiers assigned to each program category, duplicates

```
# Define index
primary_substance_group_index <- c(
  "Alcohol" = "A",
  "All Others" = "AO",
  "Cocaine incl Crack" = "CC",
  "Heroin" = "H",
  "Marijuana incl Hashish" = "MH",
  "Other Opioids" = "OO"
)

# Add substance_code column
admissions_df <- admissions_df %>%
  mutate(substance_code = recode(primary_substance_group,
        "Alcohol" = "A",
        "All Others" = "AO",
        "Cocaine incl Crack" = "CC",
        "Heroin" = "H",
        "Marijuana incl Hashish" = "MH",
        "Other Opioids" = "OO"))

# Create substance dataframe based on index
primary_substance_group <- admissions_df %>%
  distinct(primary_substance_group, .keep_all = TRUE) %>%
  select(substance_code, primary_substance_group)

primary_substance_group
```

```
## # A tibble: 6 x 2
##   substance_code primary_substance_group
##   <fct>          <fct>
## 1 A              Alcohol
## 2 AO             All Others
```

```
## 3 CC             Cocaine incl Crack
## 4 H              Heroin
## 5 OO             Other Opioids
## 6 MH             Marijuana incl Hashish
```

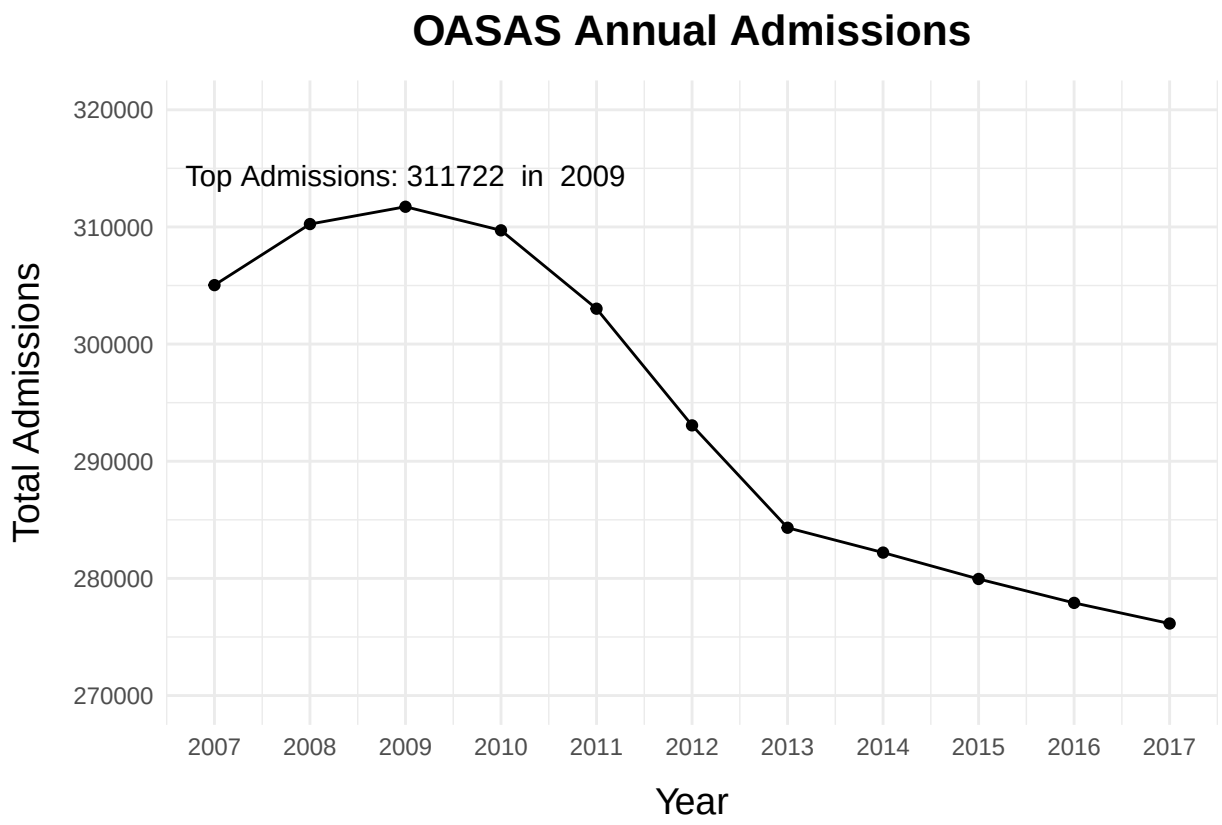
```
# Join county_code by county name
admissions_df <- admissions_df %>%
  full_join(county, by = "county_of_program_location")

# Final tibble: admissions_data_df
admissions_df <- admissions_df %>%
  select(
    year,
    county_code,
    program_code,
    service_type,
    age_group,
    primary_substance_group,
    substance_code,
    admissions
  )
admissions_df
```

```
## # A tibble: 72,463 x 8
##     year county_code program_code service_type  age_group primary_substance_gr~1
##    <dbl> <chr>       <fct>        <fct>         <fct>     <fct>
##  1  2017 AL          C            Medically Ma~ 18 thru ~ Alcohol
##  2  2017 AL          C            Medically Ma~ 18 thru ~ All Others
##  3  2017 AL          C            Medically Ma~ 18 thru ~ Cocaine incl Crack
##  4  2017 AL          C            Medically Ma~ 18 thru ~ Heroin
##  5  2017 AL          C            Medically Ma~ 18 thru ~ Other Opioids
##  6  2017 AL          C            Medically Ma~ 25 thru ~ Alcohol
##  7  2017 AL          C            Medically Ma~ 25 thru ~ All Others
##  8  2017 AL          C            Medically Ma~ 45 thru ~ All Others
##  9  2017 AL          C            Medically Ma~ 45 thru ~ Cocaine incl Crack
## 10  2017 AL          C            Medically Ma~ 45 thru ~ Heroin
## # i 72,453 more rows
## # i abbreviated name: 1: primary_substance_group
## # i 2 more variables: substance_code <fct>, admissions <dbl>
```

```
# This function uses aggregate() to sum the total admissions for every year in the admissions_data_df
# The max point is computed from the aggregated tibble and stored for later reference on the graph

annual_admissions <- function() {

  # Store aggregated data in separate tibble
  total_admissions <- aggregate(admissions_df$admissions,
            by = list(year = admissions_df$year),
            sum) %>%
    rename(total = x)

  # Line maximum
  max_point <- total_admissions[which.max(total_admissions$total), ]
```

```
  # Line graph
  ggplot(total_admissions, aes(year, total)) +
    geom_line() +
    geom_point() +
    scale_x_continuous(breaks = 2007:2017) +
    scale_y_continuous(limits = c(270000, 320000)) +
    labs(
      title = "OASAS Annual Admissions",
      x = "Year",
      y = "Total Admissions") +
    theme_minimal() +
    theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold", margin = margin(t = 5, r = 10, l =
    axis.title.x = element_text(size = 14, margin = margin(t = 10, b = 5)),
    axis.title.y = element_text(size = 14, margin = margin(r = 10, l = 5)),
    margin = margin(r = 10)
    ) +
    annotate("text",
             x = max_point$year,
             y = max_point$total,
             label = paste("Top Admissions:", max_point$total, " in ", max_point$year),
             vjust = -1)
}

annual_admissions()
```

```
## Warning in plot_theme(plot): The ‘margin‘ theme element is not defined in the
## element hierarchy.
```



- Admissions peak in 2009 with 311,722 patients
- A significant downward trend is observed from 2010 to 2013
- Admissions continue to gradually decline from 2013 onward

Data Evaluation

Admissions by County

```
# Total number of admission in NYS
total_admissions = sum(admissions_df$admissions)

# Calculate percentage of admissions in each county
admissions_by_county <- admissions_df %>%
  select(county_code, admissions) %>%
  group_by(county_code) %>%
  summarize(percentage = ((sum(admissions) / total_admissions * 100))) %>%
  arrange(desc(percentage))

admissions_by_county
```

```
## # A tibble: 61 x 2
```

```
##    county_code percentage
##    <chr>            <dbl>
##  1 NYC              15.3
##  2 QU-NYC            8.42
##  3 BR-NYC            8.03
##  4 KI-NYC            7.64
##  5 SU                7.57
##  6 ER                5.45
##  7 WE                5.08
##  8 MO                4.53
##  9 NS                3.66
## 10 OD                3.51
## # i 51 more rows
```

```
top_admissions <- admissions_by_county %>%
  slice_max(n = 5, percentage)

top_admissions
```

```
## # A tibble: 5 x 2
##   county_code percentage
##   <chr>            <dbl>
## 1 NYC              15.3
## 2 QU-NYC            8.42
## 3 BR-NYC            8.03
## 4 KI-NYC            7.64
## 5 SU                7.57
```

Rehab Facility Identification

```
rehab_df <- admissions_df %>%
  # Only show rehabilitation services
  filter(str_detect(service_type, regex("Rehab|Rehabilitation", ignore_case = TRUE))) %>%
  select(service_type, age_group, primary_substance_group, admissions)
rehab_df
```

```
## # A tibble: 17,319 x 4
##    service_type             age_group  primary_substance_group admissions
##    <fct>                    <fct>      <fct>                          <dbl>
##  1 Inpatient Rehabilitation 18 thru 24 Alcohol                           11
##  2 Inpatient Rehabilitation 18 thru 24 All Others                         2
##  3 Inpatient Rehabilitation 18 thru 24 Cocaine incl Crack                 4
##  4 Inpatient Rehabilitation 18 thru 24 Heroin                            21
##  5 Inpatient Rehabilitation 18 thru 24 Marijuana incl Hashish             6
##  6 Inpatient Rehabilitation 18 thru 24 Other Opioids                      5
##  7 Inpatient Rehabilitation 25 thru 34 Alcohol                           49
##  8 Inpatient Rehabilitation 25 thru 34 All Others                         7
##  9 Inpatient Rehabilitation 25 thru 34 Cocaine incl Crack                31
## 10 Inpatient Rehabilitation 25 thru 34 Heroin                           101
## # i 17,309 more rows
```

```
top_substance_df <- rehab_df %>%
  # Only interested in these combinations
  group_by(service_type, age_group, primary_substance_group) %>%
  # Take count to show how many admissions exist for each substance in each age group
  summarize(substance_count = sum(admissions)) %>%
  # Limit to age group
  group_by(age_group) %>%
  # Filter for the substances with the highest count
  filter(substance_count == max(substance_count)) %>%
  # Show relevant columns
  select(service_type, age_group, primary_substance_group, substance_count)
```

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

```
top_substance_df
```

```
## # A tibble: 6 x 4
## # Groups:   age_group [6]
##   service_type             age_group    primary_substance_group substance_count
##   <fct>                    <fct>        <fct>                              <dbl>
## 1 Inpatient Rehabilitation 18 thru 24   Heroin                             22705
## 2 Inpatient Rehabilitation 25 thru 34   Heroin                             37753
## 3 Inpatient Rehabilitation 35 thru 44   Alcohol                            50698
## 4 Inpatient Rehabilitation 45 thru 54   Alcohol                            69590
## 5 Inpatient Rehabilitation 55 and Older Alcohol                            30051
## 6 Res Rehab for Youth      Under 18     Marijuana incl Hashish             10643
```

- To identify the most prominent substance used in each age group, we first define a regular expression in a new dataframe that filters all services containing "Rehab" or "Rehabilitation" in the name
- top_substance_df is created to find the top substance per age group
  - The data is grouped by age_group and primary_substance_group because we are only interested in analyses in the context of these variables paired together

Admissions by Substance and Age Group

```
all_substance_df <- rehab_df %>%
  group_by(service_type, age_group, primary_substance_group) %>%
  # Take count to show how many admissions exist for each substance in each age group
  summarize(substance_count = sum(admissions)) %>%
  # Limit to age group
  group_by(age_group) %>%
  # Filter for the substances with the highest count
  mutate(substance_count == max(substance_count)) %>%
  # Show relevant columns
  select(service_type, age_group, primary_substance_group, substance_count)
```

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

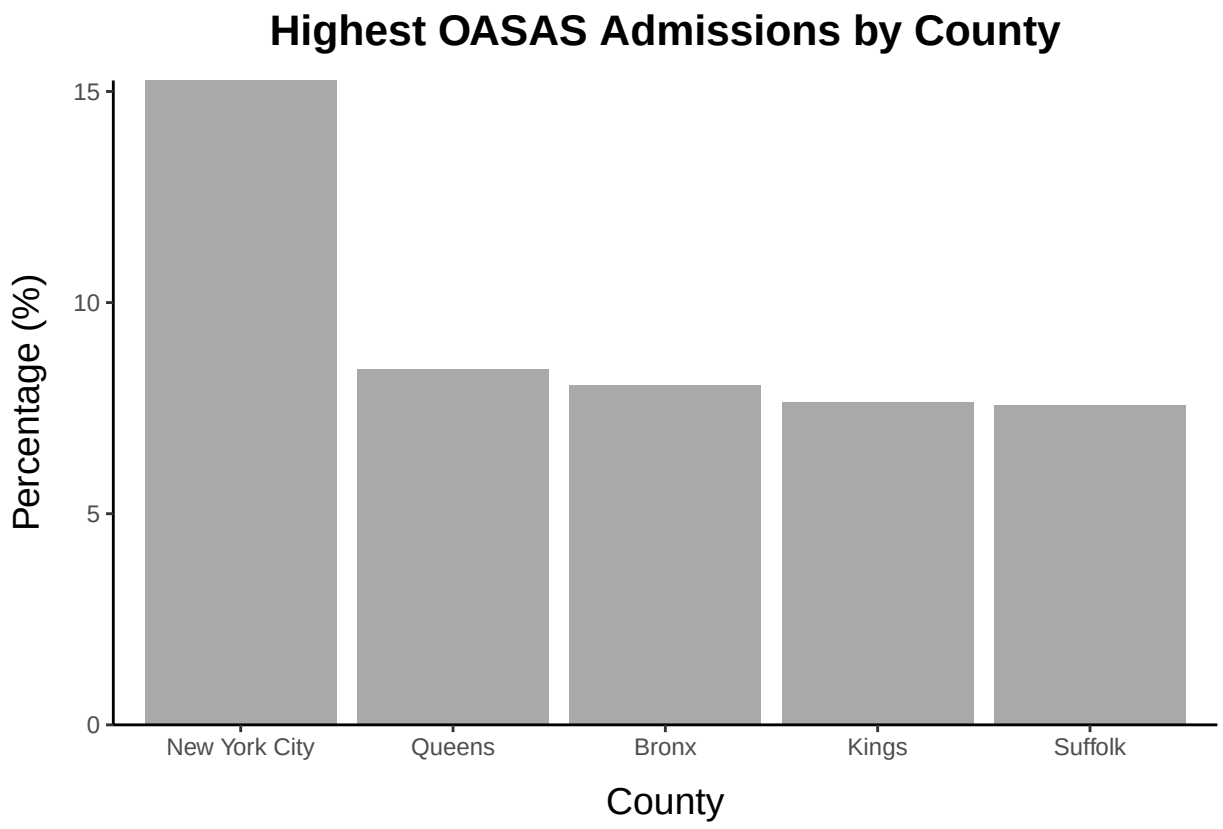
```
all_substance_df
```

```
## # A tibble: 236 x 4
## # Groups:   age_group [6]
##    service_type             age_group  primary_substance_group substance_count
##    <fct>                    <fct>      <fct>                                <dbl>
##  1 Inpatient Rehabilitation 18 thru 24 Alcohol                              10949
##  2 Inpatient Rehabilitation 18 thru 24 All Others                             3234
##  3 Inpatient Rehabilitation 18 thru 24 Cocaine incl Crack                     4583
##  4 Inpatient Rehabilitation 18 thru 24 Heroin                                22705
##  5 Inpatient Rehabilitation 18 thru 24 Marijuana incl Hashish                10209
##  6 Inpatient Rehabilitation 18 thru 24 Other Opioids                          8718
##  7 Inpatient Rehabilitation 25 thru 34 Alcohol                               32121
##  8 Inpatient Rehabilitation 25 thru 34 All Others                             5640
##  9 Inpatient Rehabilitation 25 thru 34 Cocaine incl Crack                    14559
## 10 Inpatient Rehabilitation 25 thru 34 Heroin                                37753
## # i 226 more rows
```

Data Visualization

Top Admissions by County

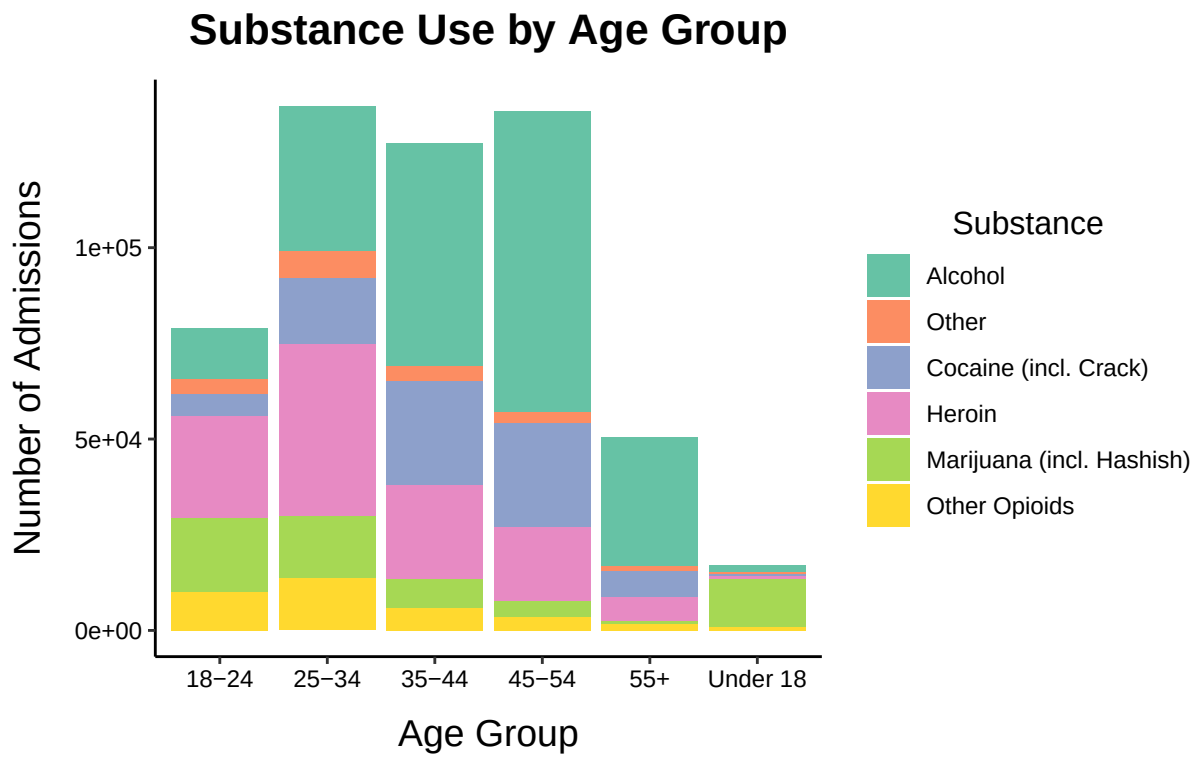
```
ggplot(top_admissions, aes(reorder(county_code, -percentage), percentage)) +
  geom_col(fill = "darkgray") +
  labs(x = "County",
       y= "Percentage (%)",
       title = "Highest OASAS Admissions by County") +
  scale_y_continuous(expand = c(0,0)) +
  scale_x_discrete(labels = c("NYC" = "New York City",
                              "QU-NYC" = "Queens",
                              "BR-NYC" = "Bronx",
                              "KI-NYC" = "Kings",
                              "SU" = "Suffolk")) +
  theme_classic() +
  theme(
    plot.title = element_text(size = 16, face = "bold", hjust = 0.5, margin = margin(t = 5, b = 10)),
    axis.title.x = element_text(size = 14, color = "black", margin = margin(t = 10, b = 5)),
    axis.title.y = element_text(size = 14, color = "black", margin = margin(r = 10, l = 5))
  )
```



Substance Use by Age Group

```
ggplot(all_substance_df, aes(age_group, substance_count, fill = primary_substance_group)) +
  geom_bar(stat = "identity") +
  labs(
    x = "Age Group",
    y = "Number of Admissions",
    title = "Substance Use by Age Group",
    caption = "Stacked bar showing substance use by age group from 2007-2017.",
    legend = "Substance"
  ) +
  scale_fill_brewer(
    name = "Substance",
    palette = "Set2",
    labels = c(
      "All Others" = "Other",
      "Cocaine incl Crack" = "Cocaine (incl. Crack)",
      "Marijuana incl Hashish" = "Marijuana (incl. Hashish)"
    )
  ) +
  scale_x_discrete(
    labels = c(
      "18 thru 24" = "18-24",
      "25 thru 34" = "25-34",
```

```
      "35 thru 44" = "35-44",
      "45 thru 54" = "45-54",
      "55 and Older" = "55+",
      "Under 18" = "Under 18"
    )
  ) +
  theme_classic() +
  theme(
    axis.title.x = element_text(size = 14, color = "black", margin = margin(t = 10, b = 5)),
    axis.title.y = element_text(size = 14, color = "black", margin = margin(r = 10, l = 10)),
    axis.text = element_text(color = "black"),
    legend.title = element_text(size = 12, color = "black", hjust = 0.5),
    plot.title = element_text(size = 16, face = "bold", margin = margin(b = 10, t = 10), hjust = 0.5),
    plot.caption = element_text(size = 11, hjust = 0.5, margin = margin(b = 5, t = 5))
  )
```

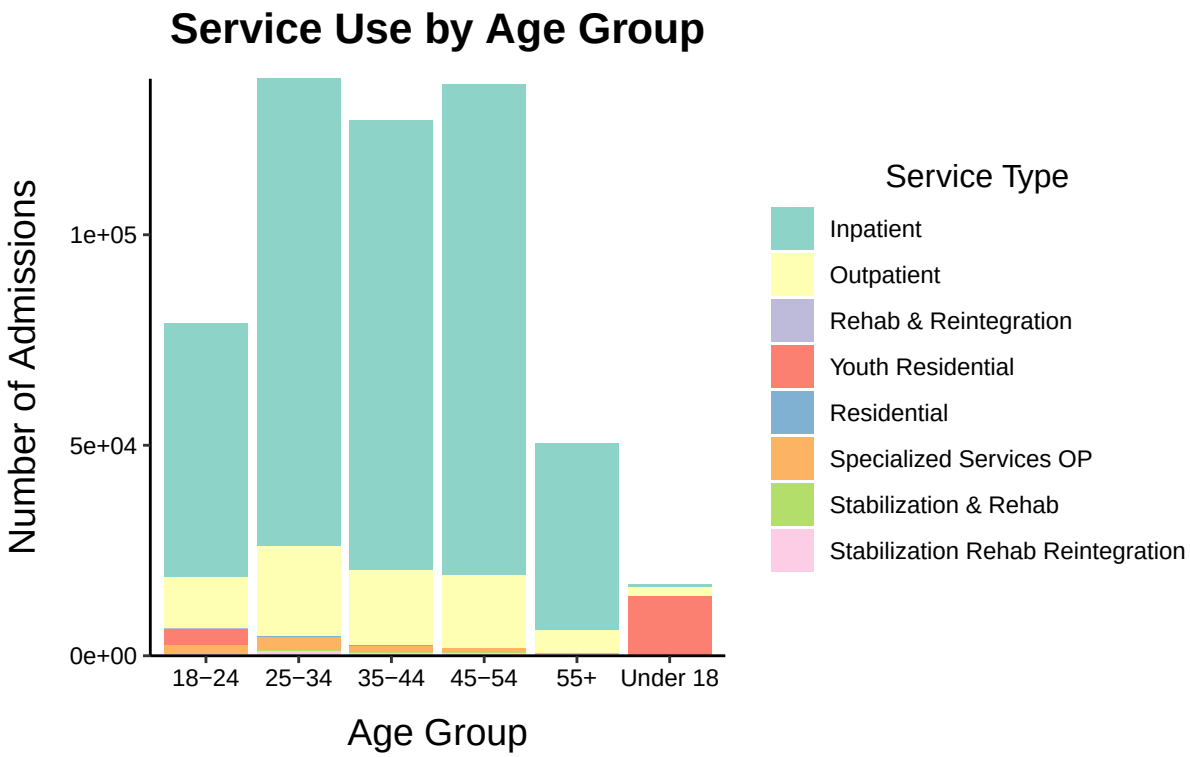


Stacked bar showing substance use by age group from 2007−2017.

Service Use by Age Group

```
ggplot(all_substance_df, aes(age_group, substance_count, fill = service_type)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(expand = c(0,0)) +
  labs(
    x = "Age Group",
```

```
    y = "Number of Admissions",
    title = "Service Use by Age Group",
    caption = "Stacked bar showing service use by age group from 2007-2017."
  ) +
  scale_fill_brewer(
    name = "Service Type",
    palette = "Set3",
    labels = c(
      "Inpatient Rehabilitation" = "Inpatient",
      "Outpatient Rehabilitation" = "Outpatient",
      "Rehab and Reintegration" = "Rehab & Reintegration",
      "Res Rehab for Youth" = "Youth Residential",
      "Residential Rehabilitation" = "Residential",
      "Specialized Services OP Rehab" = "Specialized Services OP",
      "Stabilization and Rehab" = "Stabilization & Rehab"
    )
  ) +
  scale_x_discrete(
    labels = c(
      "18 thru 24" = "18-24",
      "25 thru 34" = "25-34",
      "35 thru 44" = "35-44",
      "45 thru 54" = "45-54",
      "55 and Older" = "55+",
      "Under 18" = "Under 18"
    )
  ) +
  theme_classic() +
  theme(
    axis.title.x = element_text(size = 14, color = "black", margin = margin(t = 10, b = 5)),
    axis.title.y = element_text(size = 14, color = "black", margin = margin(r = 10, l = 10)),
    axis.text = element_text(color = "black"),
    legend.title = element_text(size = 12, color = "black", hjust = 0.5),
    plot.title = element_text(size = 16, face = "bold", margin = margin(b = 10, t = 10), hjust = 0.5),
    plot.caption = element_text(size = 11, hjust = 0.5, margin = margin(b = 5, t = 5))
  )
```

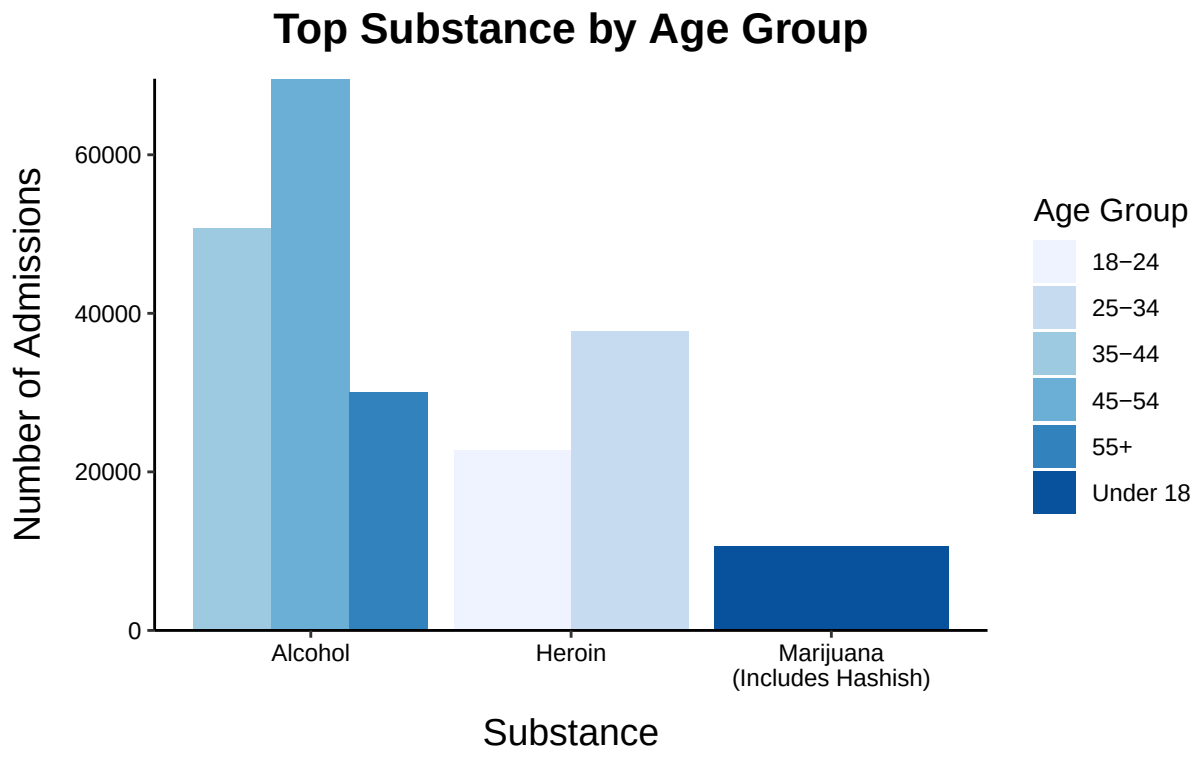


Stacked bar showing service use by age group from 2007−2017.

Top Substance Use by Age Group

```
ggplot(top_substance_df, aes(primary_substance_group, substance_count, fill = age_group)) +
  geom_bar(stat = "identity", position = "dodge") +
  scale_y_continuous(expand = c(0,0)) +
  labs(
    y = "Number of Admissions",
    title = "Top Substance by Age Group",
    caption = "Stacked bar showing the top substance used per age group from 2007-2017."
  ) +
  scale_x_discrete(
    name = "Substance",
    labels = c(
      "All Others" = "Other",
      "Cocaine incl Crack" = "Cocaine (Includes Crack)",
      "Marijuana incl Hashish" = "Marijuana\n(Includes Hashish)"
    )
  ) +
  scale_fill_brewer(
    name = "Age Group",
    labels = c(
      "18 thru 24" = "18-24",
      "25 thru 34" = "25-34",
      "35 thru 44" = "35-44",
```

```
    "45 thru 54" = "45-54",
    "55 and Older" = "55+",
    "Under 18" = "Under 18"
  )
) +
theme_classic() +
theme(
  axis.title.x = element_text(size = 14, color = "black", margin = margin(t = 10, b = 5)),
  axis.title.y = element_text(size = 14, color = "black", margin = margin(r = 10, l = 10)),
  axis.text = element_text(color = "black"),
  legend.title = element_text(size = 12, color = "black", hjust = 0.5),
  plot.title = element_text(size = 16, face = "bold", margin = margin(b = 10, t = 10), hjust = 0.5),
  plot.caption = element_text(size = 11, hjust = 0.5, margin = margin(b = 5, t = 5))
)
```



Stacked bar showing the top substance used per age group from 2007−2017.