Data Visualizations and Analysis

Grayson White

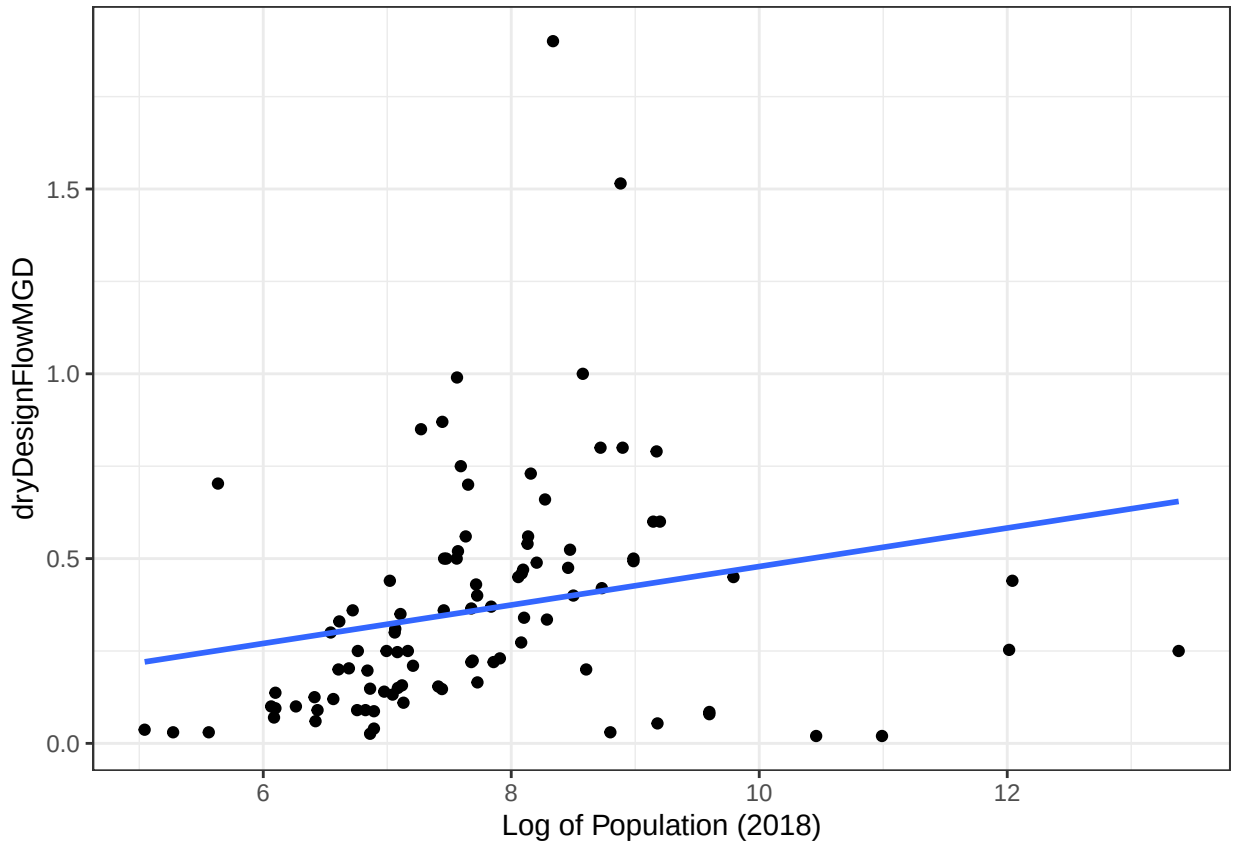
7/13/2020

```
ggplot(working_df, aes(x = log(pop_2018), y = dryDesignFlowMGD)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() +
  labs(x = "Log of Population (2018)")
```

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

```
## Warning: Removed 17 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 17 rows containing missing values (geom_point).
```

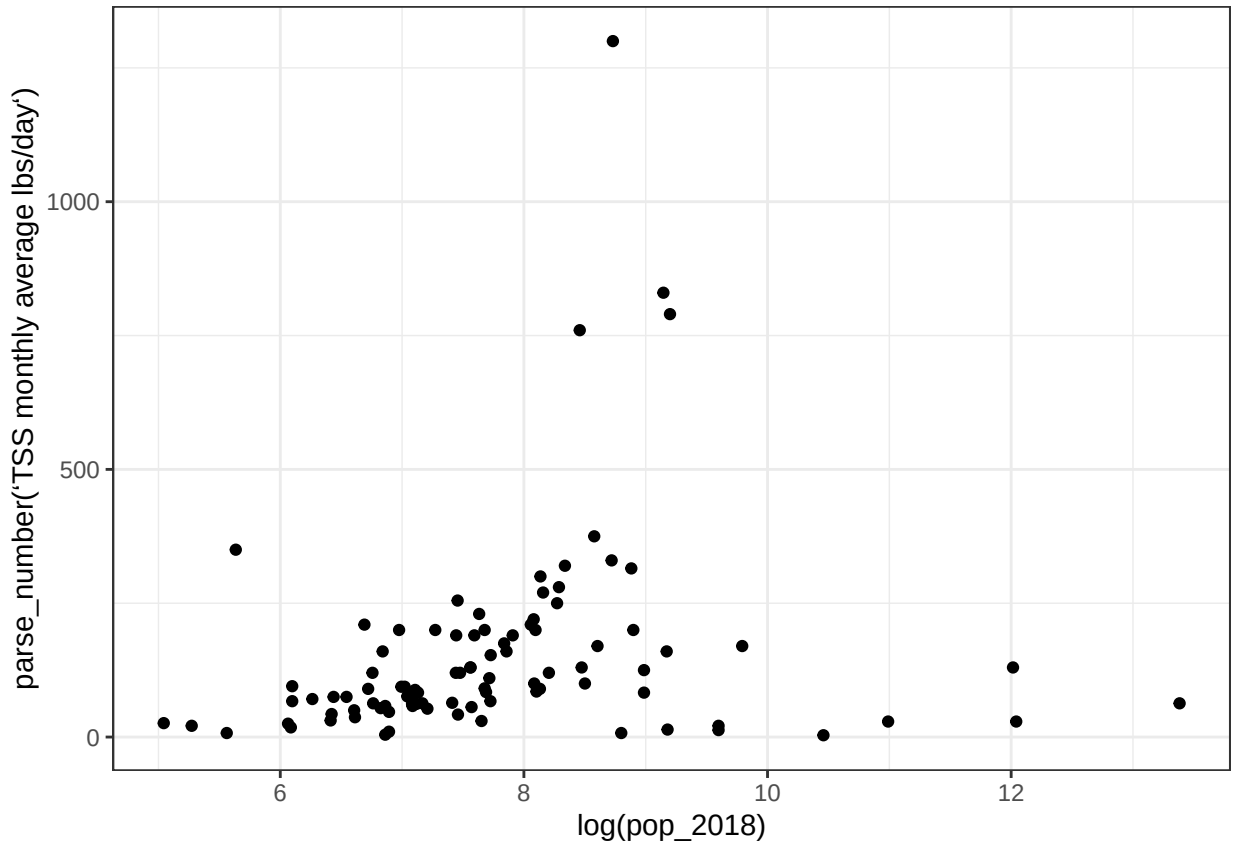


```
ggplot(working_df, aes(x = log(pop_2018),
                       y = parse_number(`TSS monthly average lbs/day`))) +
  geom_point() +
  theme_bw()
```

```
## Warning: 2 parsing failures.
## row col expected actual
##  13  -- a number     na
##  45  -- a number     na
```

```
## Warning: 2 parsing failures.
## row col expected actual
##  13  -- a number     na
##  45  -- a number     na
```

```
## Warning: Removed 19 rows containing missing values (geom_point).
```




```
working_df %>%
  filter(type1 %in% c("lagoons", "activated sludge")) %>%
  group_by(type1) %>%
  summarize(median = median(pop_2018, na.rm = TRUE))
```

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

```
## # A tibble: 2 x 2
##   type1            median
##   <chr>             <dbl>
## 1 activated sludge  1962.
## 2 lagoons           1718.
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(plotly)
```

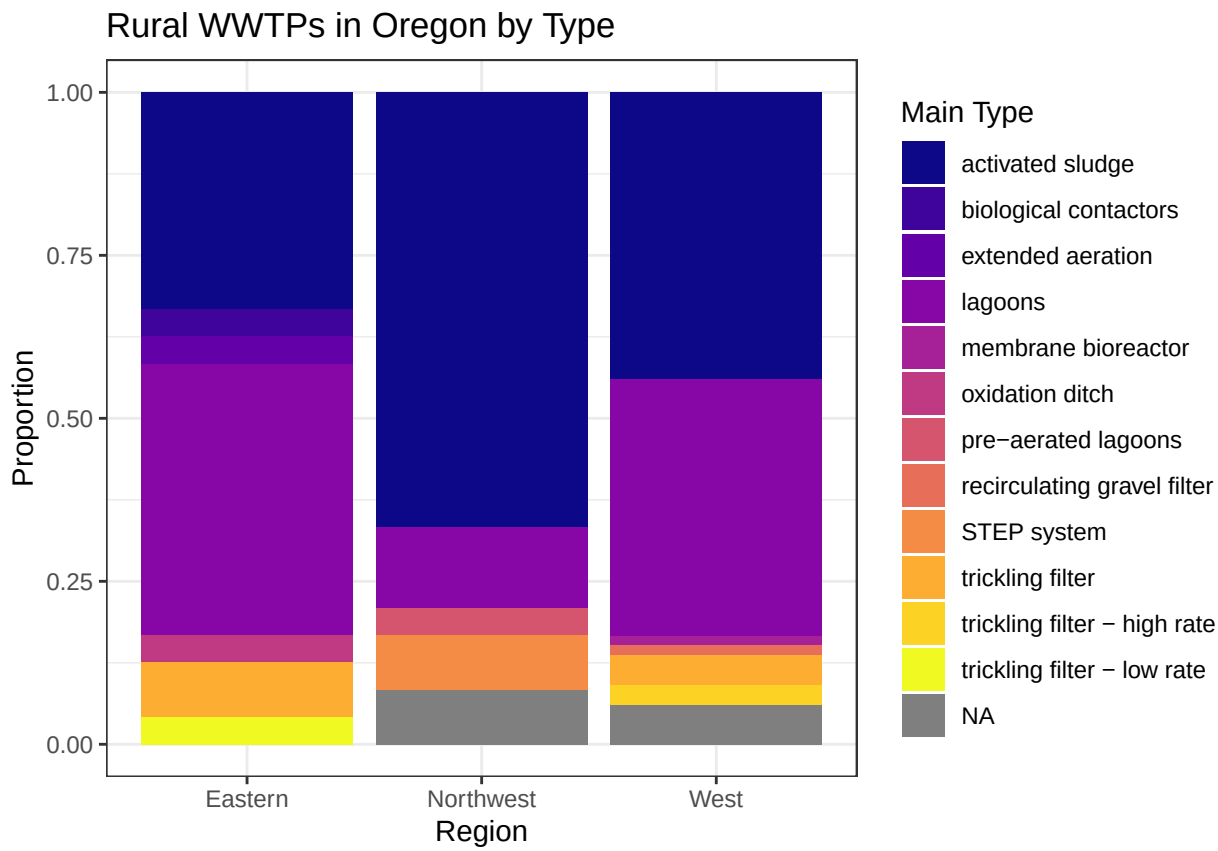
```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

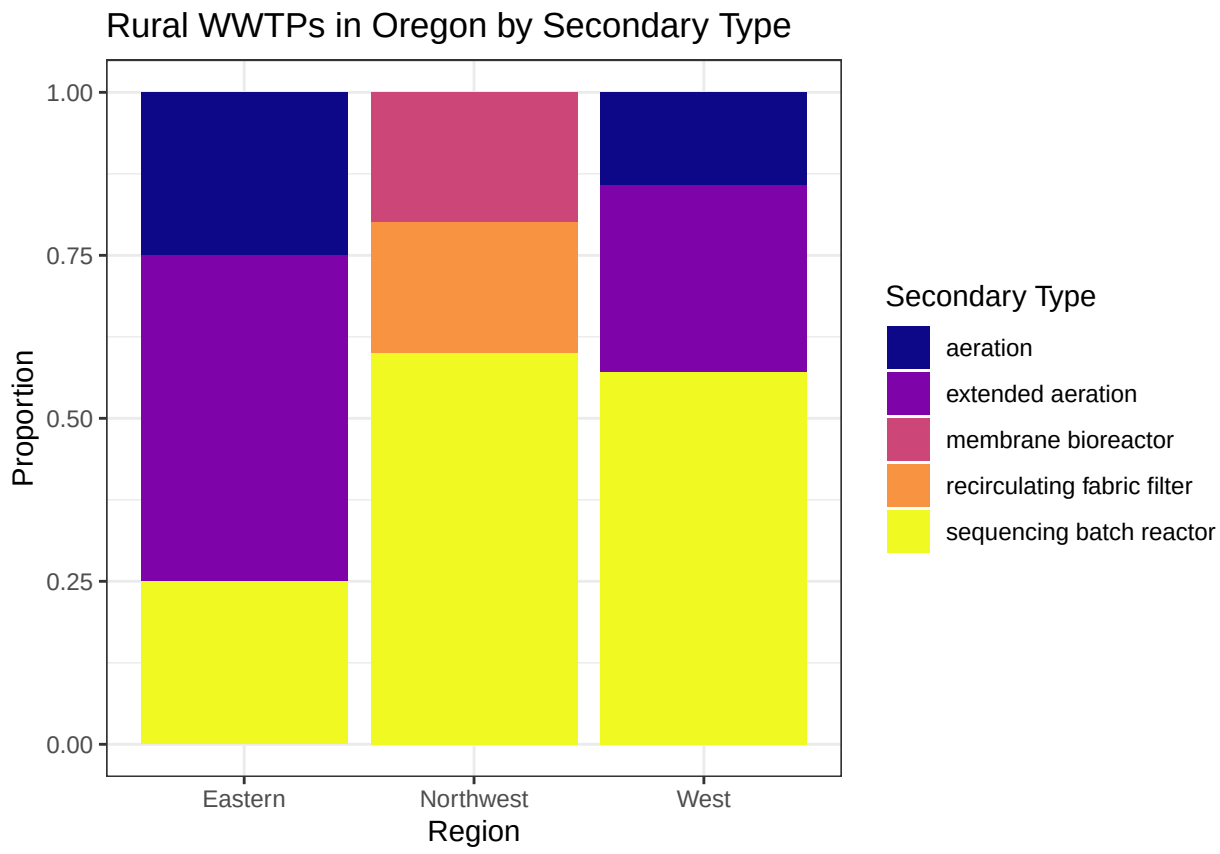
```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

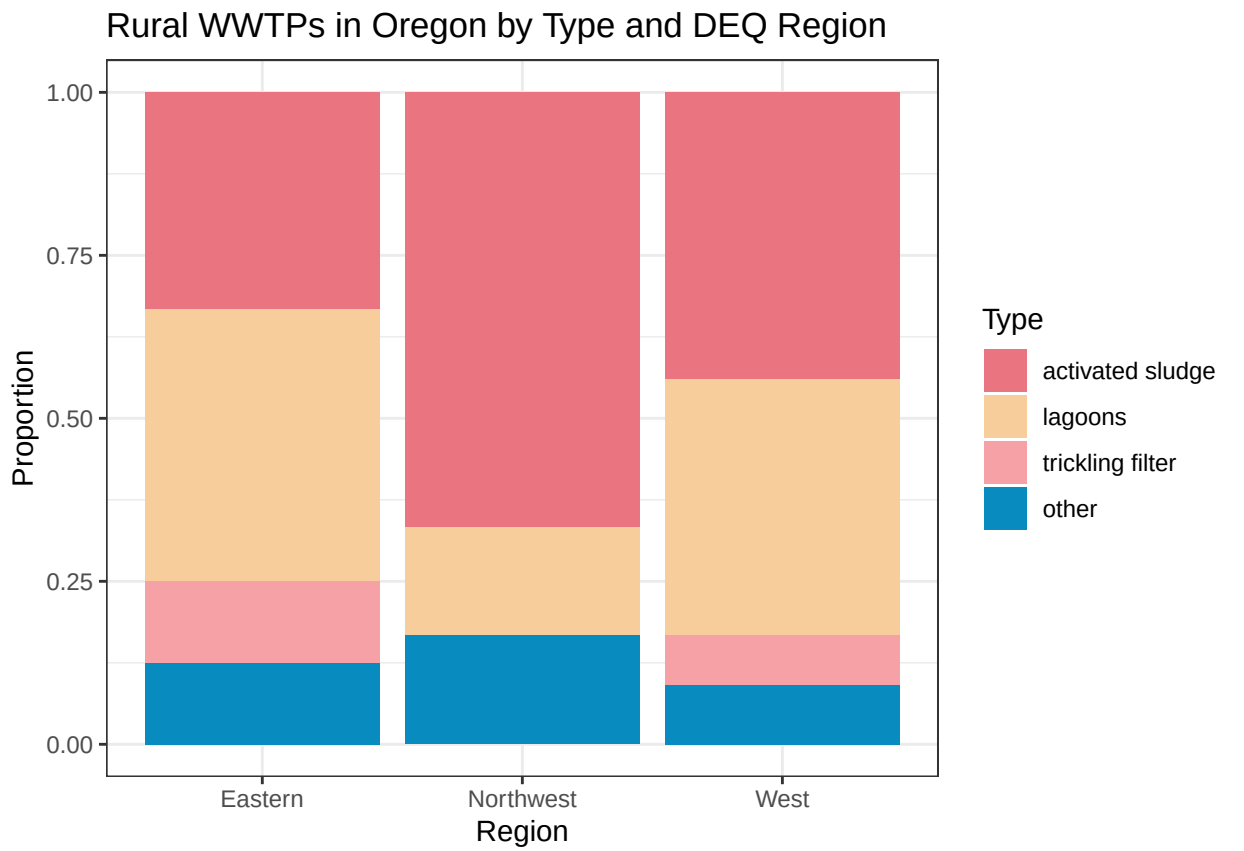
```
p <- working_df %>%
  ggplot(aes(x = Region.x,
             fill = type1)) +
  geom_bar(position = "fill") +
  scale_fill_viridis_d(option = "C", na.value = "grey50") +
  scale_x_discrete(labels=c("Eastern", "Northwest", "West")) +
  theme_bw() +
  labs(x = "Region",
       fill = "Main Type",
       y = "Proportion",
       title = "Rural WWTPs in Oregon by Type")
p
```




```
# ggplotly(p, tooltip = c("type1", "count"))

working_df %>%
  filter(type2 != c("na", NA)) %>%
  ggplot(aes(x = Region.x,
             fill = type2)) +
  geom_bar(position = "fill") +
  scale_fill_viridis_d(option = "C", na.value = "grey50") +
  scale_x_discrete(labels=c("Eastern", "Northwest", "West")) +
  theme_bw() +
  labs(x = "Region",
       fill = "Secondary Type",
       y = "Proportion",
       title = "Rural WWTPs in Oregon by Secondary Type")
```



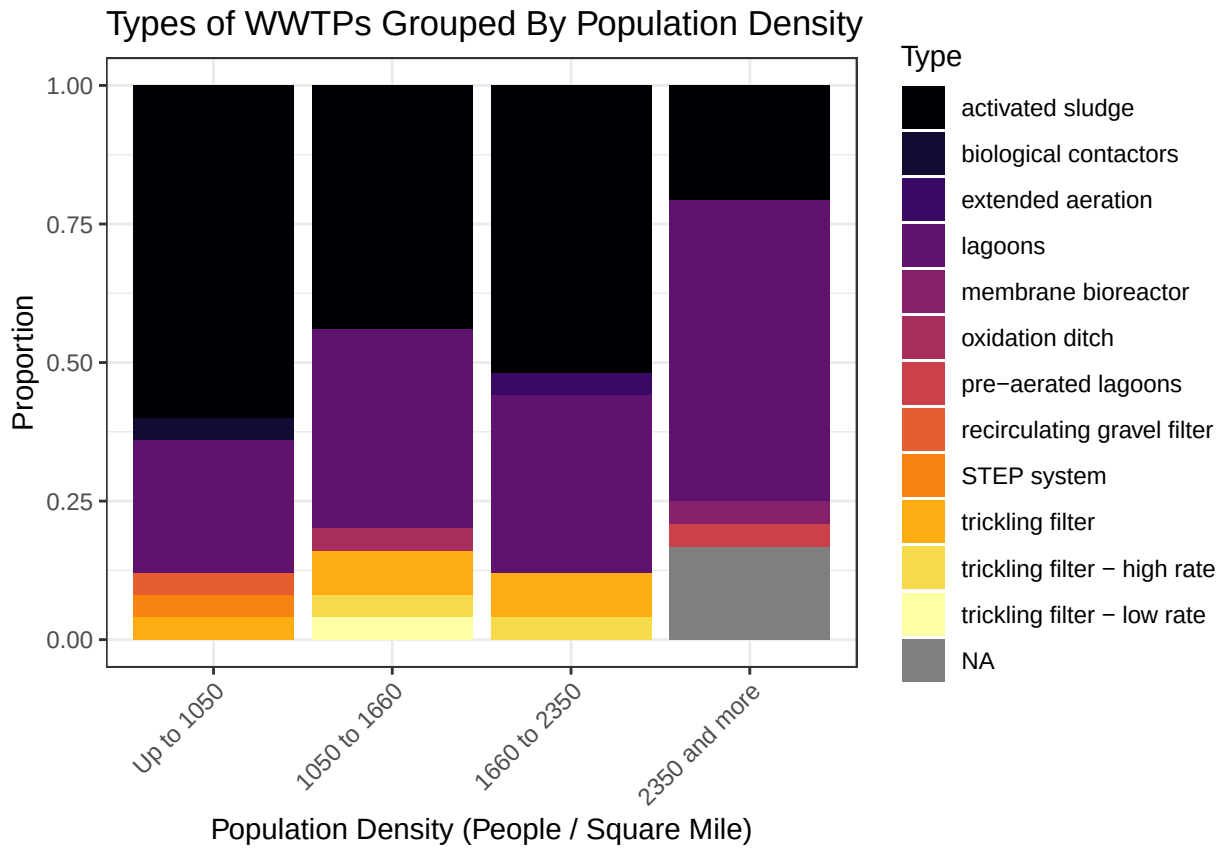

```r
library(LaCroixColoR)

p4 <- working_df %>%
  mutate(
    type_plot = case_when(
      type1 %in% c("lagoons", "pre-aerated lagoons") ~ "lagoons",
      type1 %in% c("trickling filter", "trickling filter - high rate",
                   "trickling filter - low rate") ~ "trickling filter",
      type1 %in% c("activated sludge") ~ "activated sludge",
      type1 %in% c("extended aeration", "membrane bioreactor", "recirculating gravel filter", NA,
                   "STEP system", "oxidation ditch", "biological contactors"
    ) ~ "other"
  )) %>%
  ggplot(aes(x = Region.x,
             fill = factor(type_plot, levels = c("activated sludge", "lagoons", "trickling filter", "ot
  geom_bar(position = "fill") +
  scale_fill_manual(values = lacroix_palette("Pamplemousse", type = "discrete")) +
  scale_x_discrete(labels=c("Eastern", "Northwest", "West")) +
  theme_bw() +
  labs(x = "Region",
       fill = "Type",
       y = "Proportion",
       title = "Rural WWTPs in Oregon by Type and DEQ Region")
  #theme(legend.position = "bottom")
p4
```



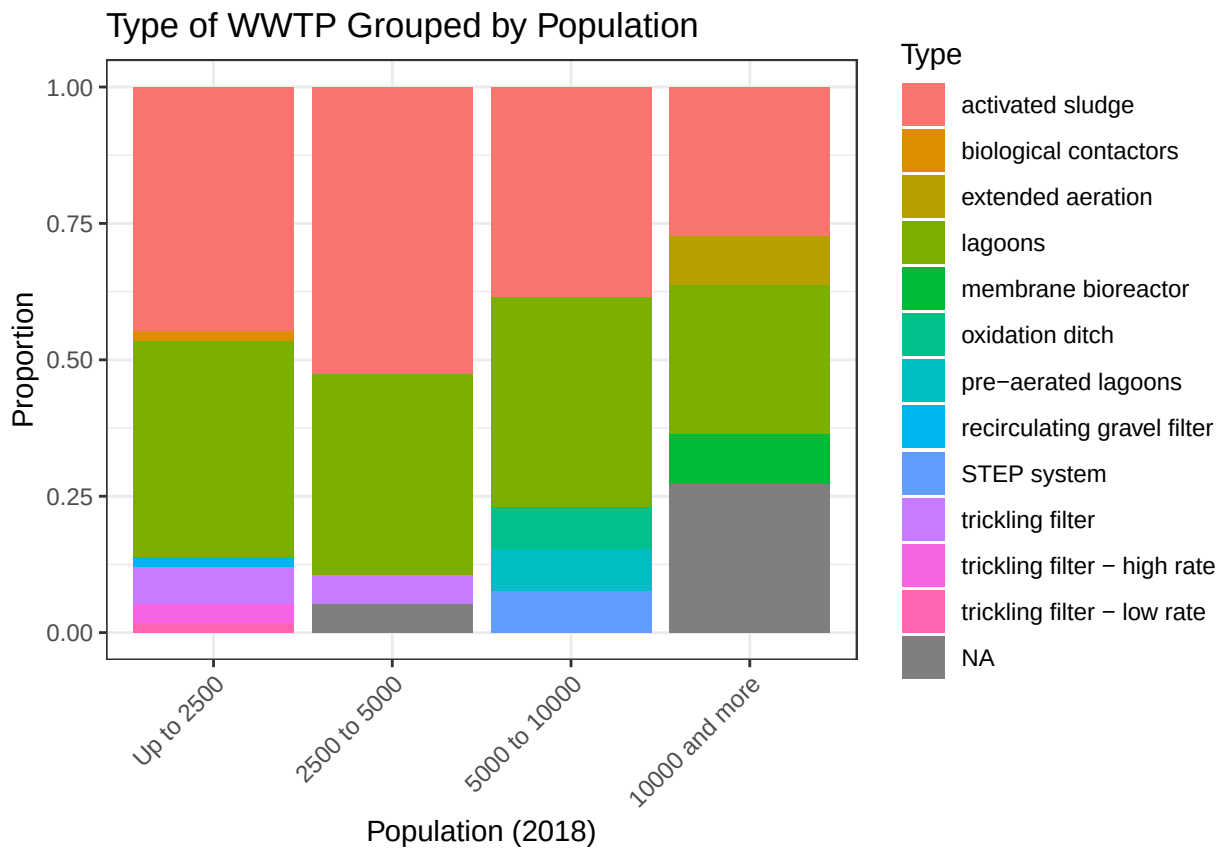

```
library(gtools)
```

```
## Warning: package 'gtools' was built under R version 4.0.1
```

```
working_df$quantile_density <- quantcut(working_df$pop_density_2018, q = 4)
ggplot(working_df[!is.na(working_df$quantile_density), ], aes(x = quantile_density,
                      fill = type1)) +
  geom_bar(position = "fill") +
  scale_x_discrete(labels=c("Up to 1050", "1050 to 1660", "1660 to 2350", "2350 and more")) +
  scale_fill_viridis_d(option = "B", na.value = "grey50") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1)) +
  labs(x = "Population Density (People / Square Mile)",
       y = "Proportion",
       fill = "Type",
       title = "Types of WWTPs Grouped By Population Density")
```




```
working_df$quantile_pop_2018 <- quantcut(working_df$pop_2018, q = 4)
working_df$cut_pop_2018 <- cut(working_df$pop_2018, breaks = c(0, 2500, 5000, 10000, 1e7))
p1 <- ggplot(working_df[!is.na(working_df$cut_pop_2018), ], aes(x = cut_pop_2018,
                        fill = type1)) +
  geom_bar(position = "fill") +
  scale_x_discrete(labels=c("Up to 2500", "2500 to 5000", "5000 to 10000", "10000 and more")) +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1)) +
  labs(x = "Population (2018)",
       y = "Proportion",
       title = "Type of WWTP Grouped by Population",
       fill = "Type")

p1
```




```
# ggplotly(p1)
```

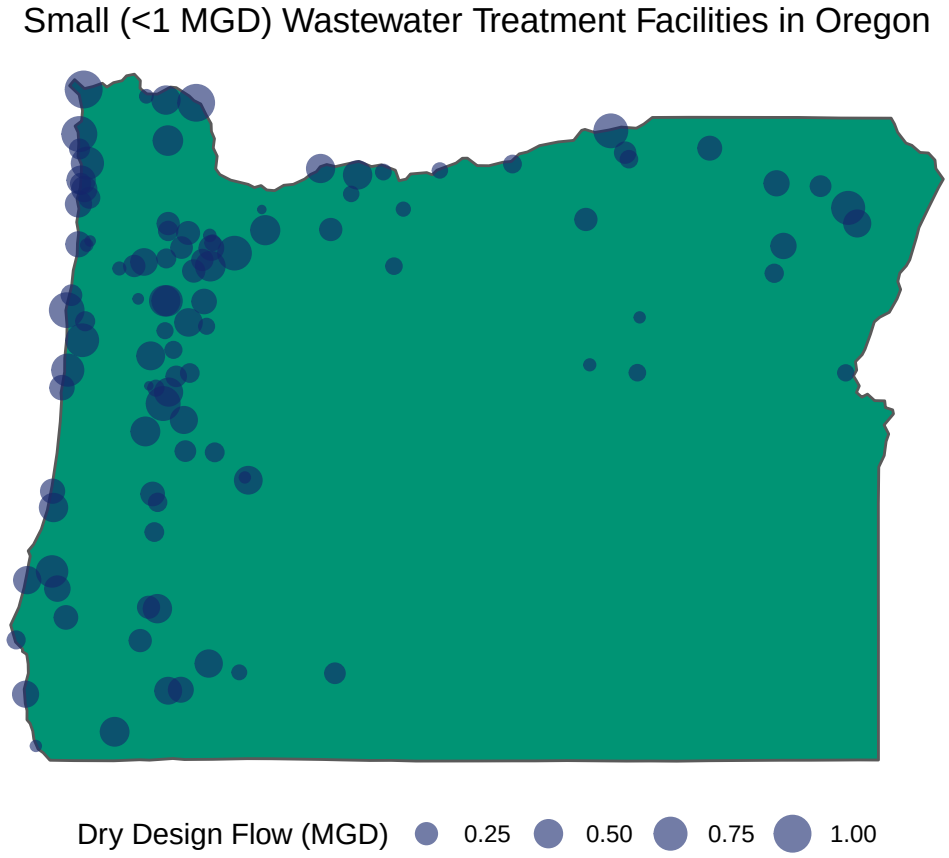
Maps:

```
library(sf)
```

```
## Linking to GEOS 3.7.2, GDAL 2.4.2, PROJ 5.2.0
```

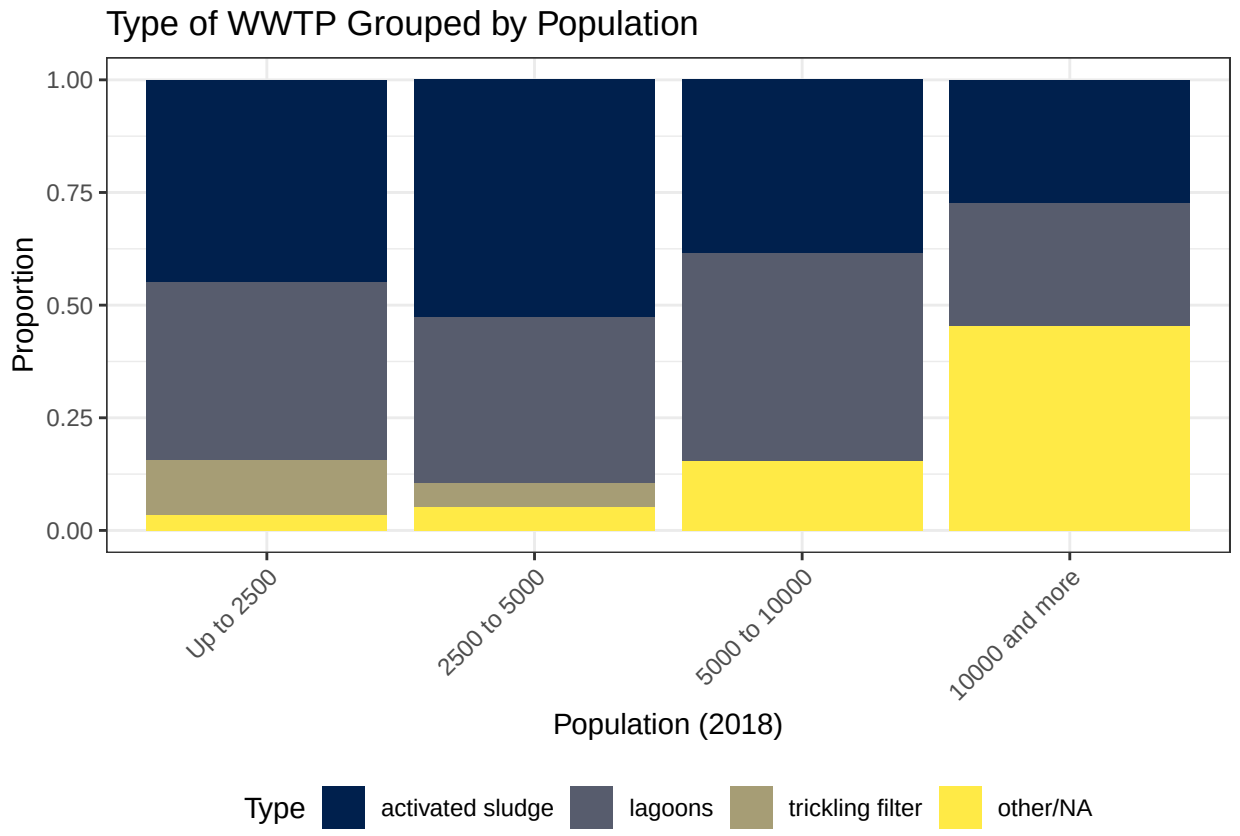
```
library(USAboundaries)
library(PNWColors)
df_sf <- st_as_sf(working_df, coords = c("Longitude", "Latitude"), crs = "+proj=longlat +datum=WGS84")
OR_sf <- us_boundaries(type = "state", states = "OR")


working_df_ <- working_df %>%
  filter(dryDesignFlowMGD <= 1)
p2 <- ggplot() +
  geom_sf(data = OR_sf, fill = "#009474") +
  geom_point(data = working_df_,
             aes(
                 x = Longitude,
                 y = Latitude,
                 size = dryDesignFlowMGD
                 ),
```

```
            alpha = 0.6,
            color = "#15266B") +
  coord_sf() +
  theme_void() +
  labs(title = "Small (<1 MGD) Wastewater Treatment Facilities in Oregon",
       size = "Dry Design Flow (MGD)") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.title.position = "plot",
        legend.position = "bottom")

p2
```




```
# pop la croix
working_df <- working_df %>%
  mutate(
    type_plot = case_when(
      type1 %in% c("lagoons", "pre-aerated lagoons") ~ "lagoons",
      type1 %in% c("trickling filter", "trickling filter - high rate",
                   "trickling filter - low rate") ~ "trickling filter",
      type1 %in% c("activated sludge") ~ "activated sludge",
      type1 %in% c("extended aeration", "membrane bioreactor", "recirculating gravel filter", NA,
                   "STEP system", "oxidation ditch", "biological contactors"
    ) ~ "other/NA"
  ))
ggplot(working_df[!is.na(working_df$cut_pop_2018), ], aes(x = cut_pop_2018,
```

```
                      fill = factor(type_plot, levels = c("activated sludge", "lagoons", "trickling fi
  geom_bar(position = "fill") +
  scale_x_discrete(labels=c("Up to 2500", "2500 to 5000", "5000 to 10000", "10000 and more")) +
  scale_fill_viridis_d(option = "E") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1),
        legend.position = "bottom") +
  labs(x = "Population (2018)",
       y = "Proportion",
       title = "Type of WWTP Grouped by Population",
       fill = "Type")
```

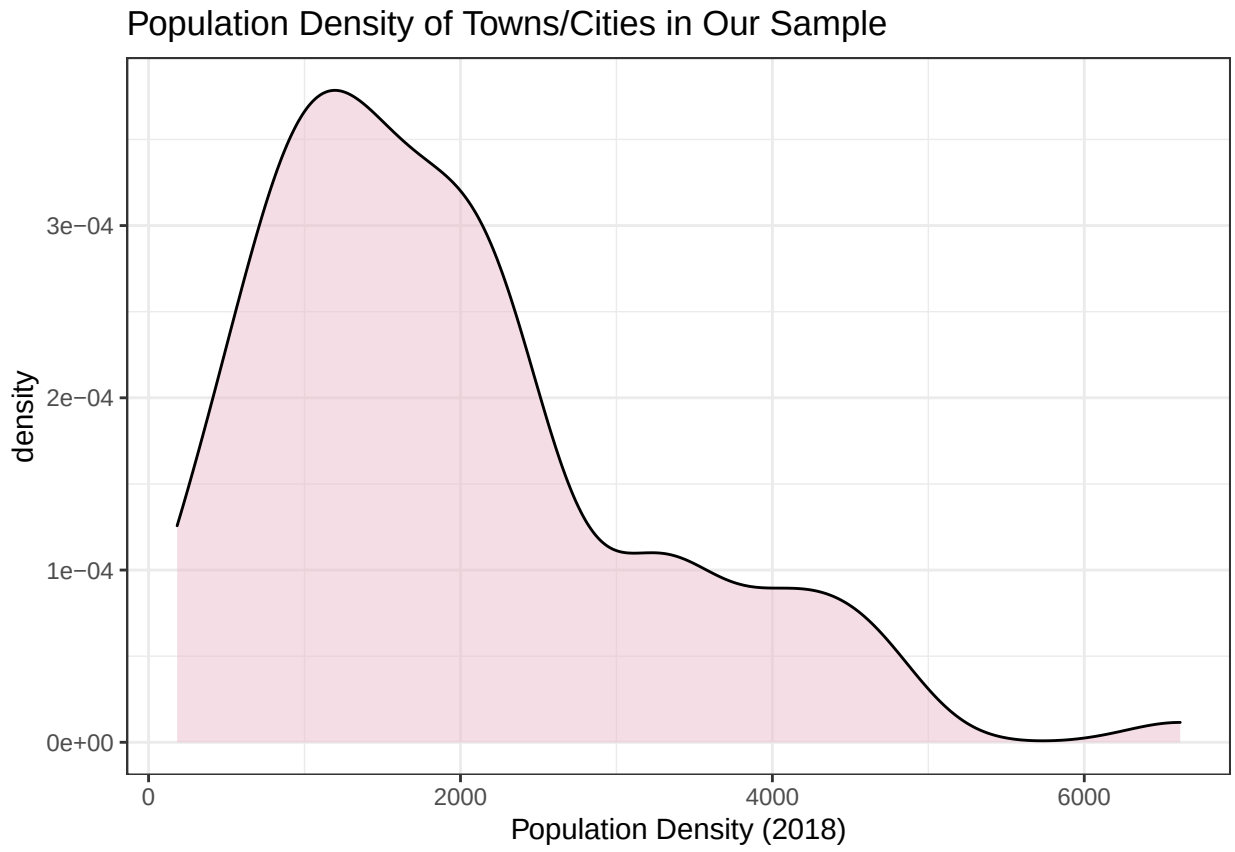



```
# do not run, does not work, crashes R
# try pop densities
# library(tidycensus)
# library(tigris)
#
# pop_block <- get_acs(
#   geography = "block group",
#   variables = "B01003_001",
#   state = "OR",
#   geometry = TRUE,
#   key = "abac0e1ca2aa3d3ebb31d6d2fcdbaf52d3e25f7d"
# )
#
```

```
# area_2017 <-
#   block_groups(year = 2017,
#                state = "OR",
#                class = "sf")
#
# area_2017 <- area_2017 %>%
#   mutate(area = ALAND / 2589988) %>%
#   select(area, geometry, GEOID)
#
# pop_block <- pop_block %>%
#   select(estimate, geometry, GEOID)
#
# class(pop_block) <- "data.frame"
# class(area_2017) <- "data.frame"
#
# density <- left_join(pop_block, area_2017, by = c("GEOID" = "GEOID"))
#
# density <- density %>%
#   mutate(pop_density = estimate/area)
#
# class(density) <- c("sf", "data.frame")
```

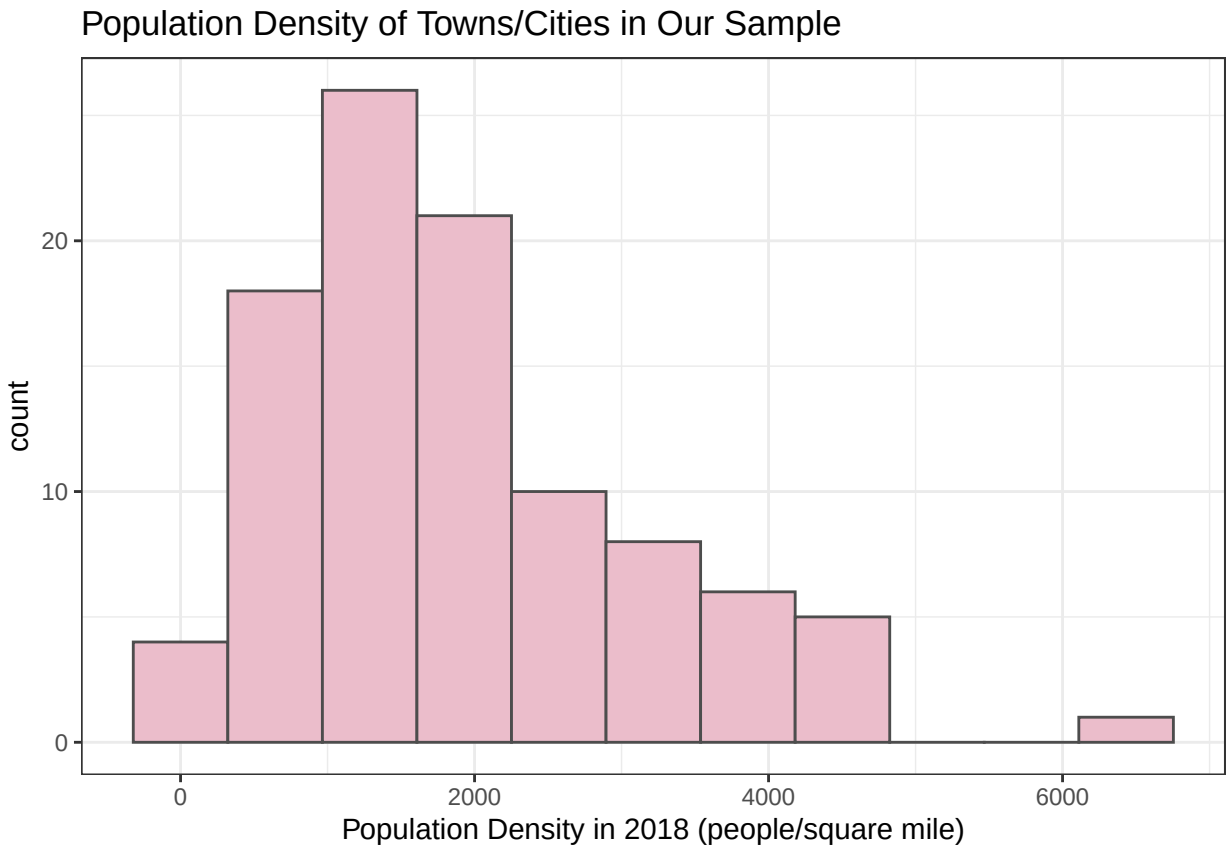
```
working_df %>%
  ggplot(aes(x = pop_density_2018)) +
  geom_density(fill = "#EBBDCB",
               alpha = 0.5) +
  theme_bw() +
  labs(x = "Population Density (2018)",
       title = "Population Density of Towns/Cities in Our Sample")
```

```
## Warning: Removed 15 rows containing non-finite values (stat_density).
```



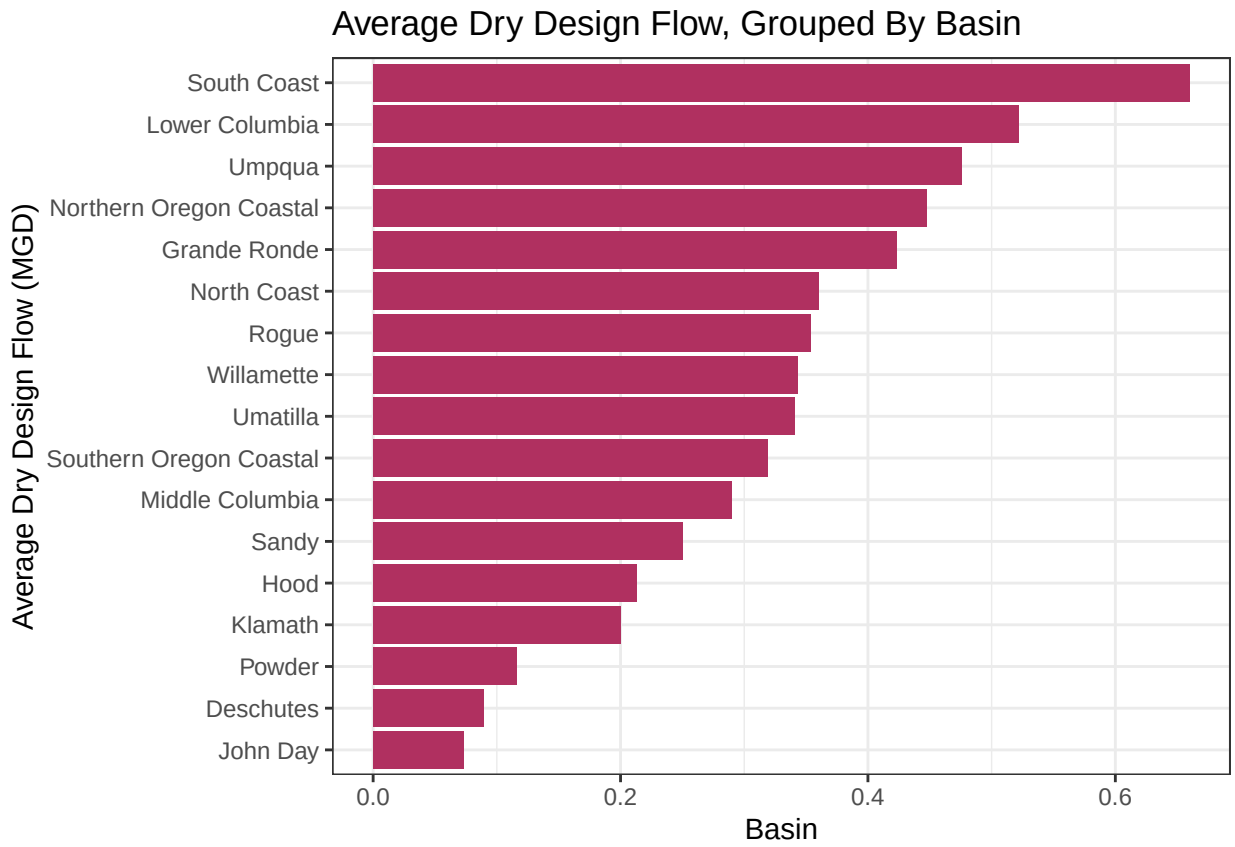

```
working_df %>%
  ggplot(aes(x = pop_density_2018)) +
  geom_histogram(bins = 11,
                 fill = "#EBBDCB",
                 color = "grey30") +
  theme_bw() +
  labs(x = "Population Density in 2018 (people/square mile)",
       title = "Population Density of Towns/Cities in Our Sample")
```

```
## Warning: Removed 15 rows containing non-finite values (stat_bin).
```



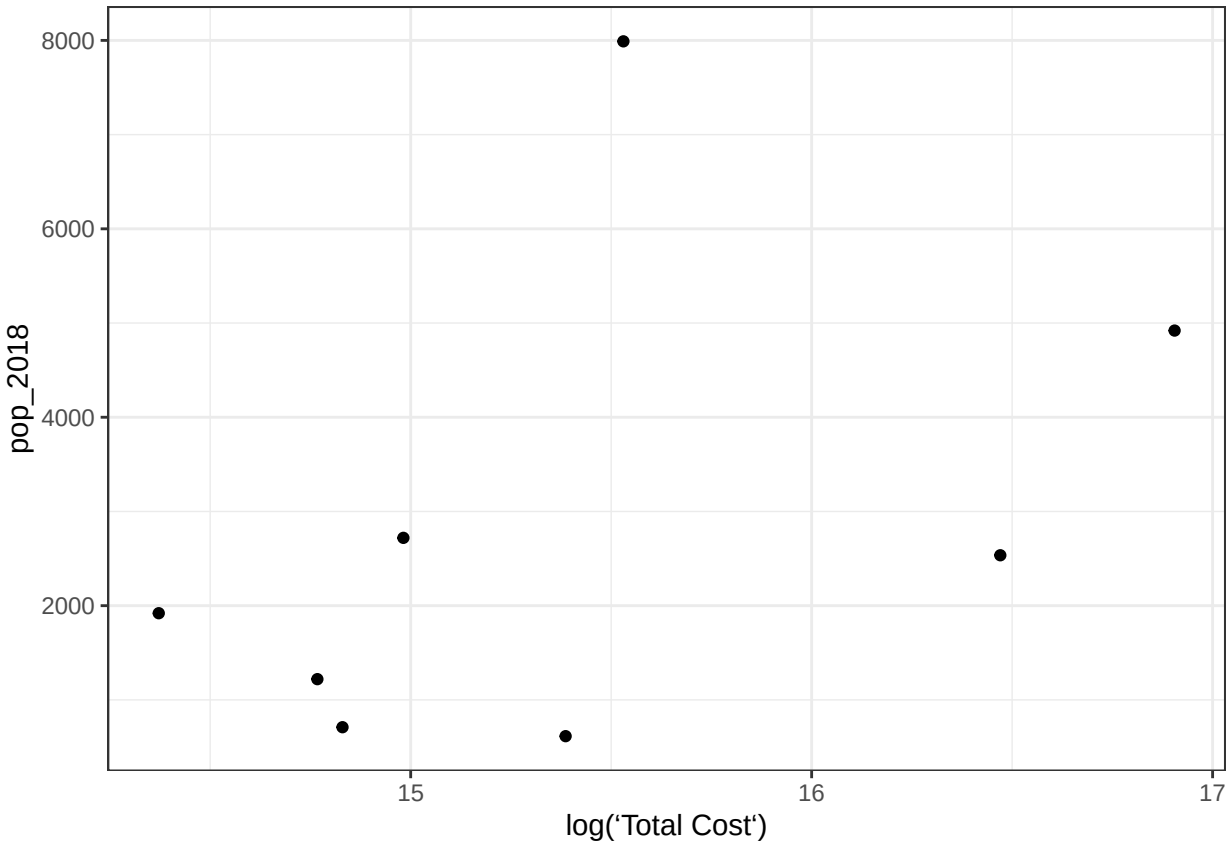

```
working_df %>%
  filter(!is.na(basin)) %>%
  group_by(basin) %>%
  summarize(mgd = mean(dryDesignFlowMGD, na.rm = TRUE)) %>%
  ggplot(
    aes(x = reorder(basin, mgd), y = mgd)
  ) +
  geom_col(fill = "maroon") +
  coord_flip() +
  theme_bw() +
  labs(
    x = "Average Dry Design Flow (MGD)",
    y = "Basin",
    title = "Average Dry Design Flow, Grouped By Basin"
  )
```

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

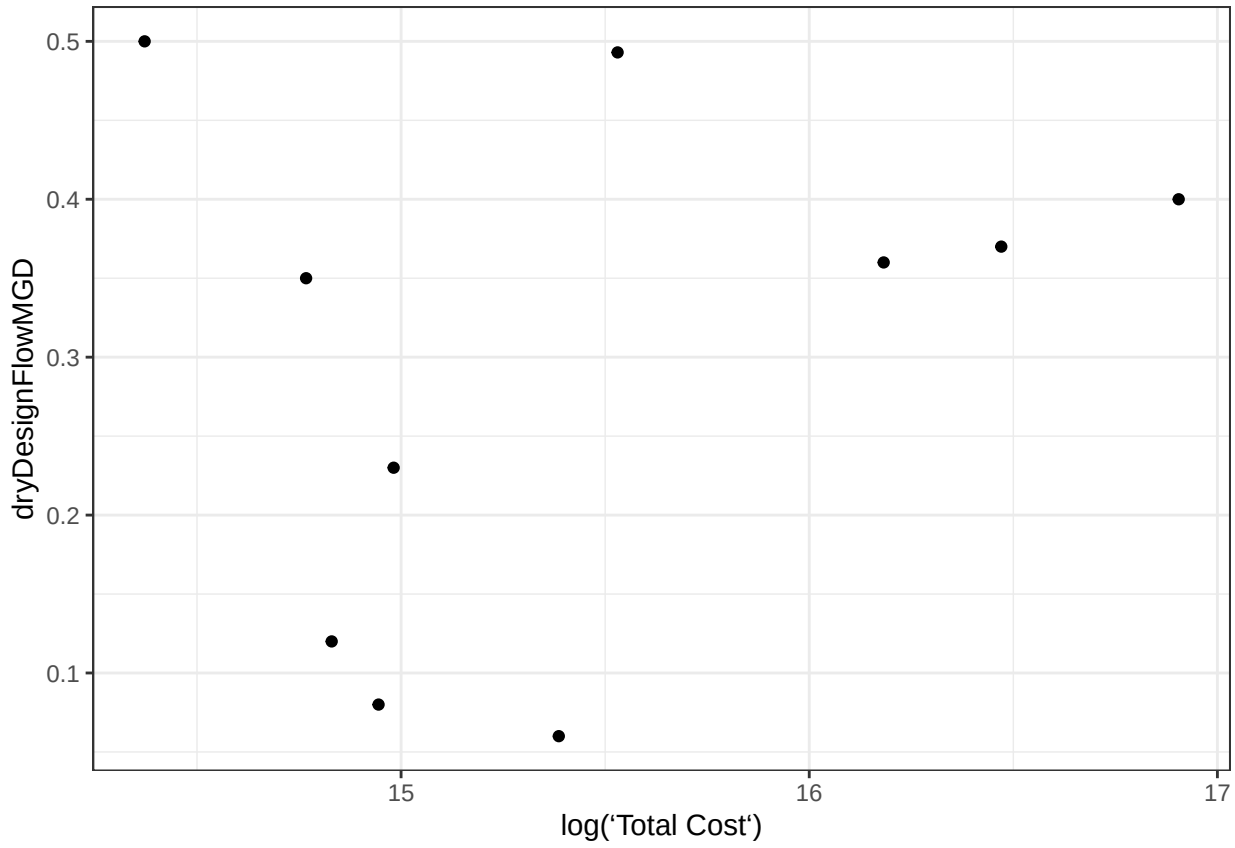



```
# cost
cost %>%
  ggplot(aes(x = log(`Total Cost`),
             y = pop_2018)) +
  geom_point() +
  theme_bw()
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```

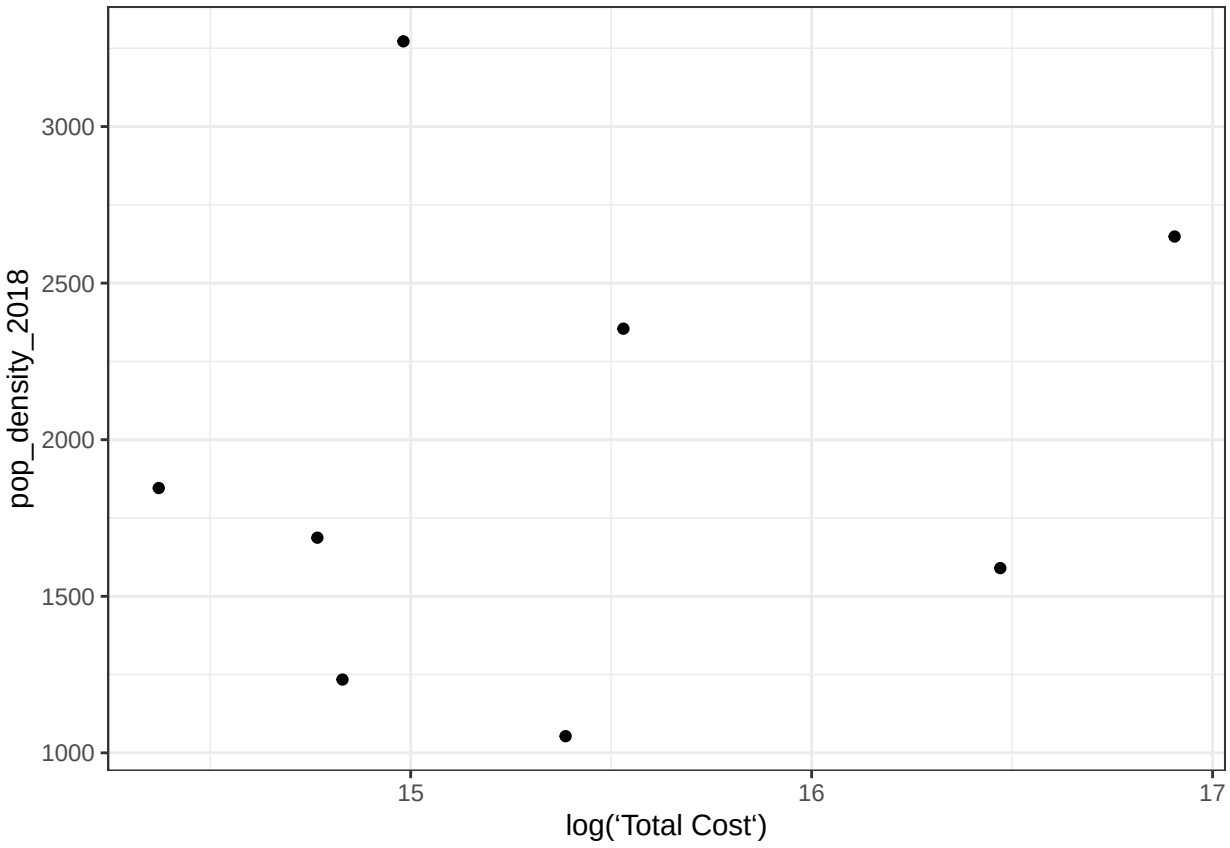



```
cost %>%
  ggplot(aes(x = log(‘Total Cost‘),
             y = dryDesignFlowMGD)) +
  geom_point() +
  theme_bw()
```




```
cost %>%
  ggplot(aes(x = log(‘Total Cost‘),
             y = pop_density_2018)) +
  geom_point() +
  theme_bw()
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```




```
m1 <- lm(log(‘Total Cost‘) ~ dryDesignFlowMGD + pop_2018 + pop_density_2018 + Region.x, data = cost)
summary(m1)
```

```
##
## Call:
## lm(formula = log(‘Total Cost‘) ~ dryDesignFlowMGD + pop_2018 +
##     pop_density_2018 + Region.x, data = cost)
##
## Residuals:
##        1        2        3        4        6        7        8       10
##  0.11565  0.14747 -0.06669  0.04150 -0.04896  0.23982 -0.42879  0.00000
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      17.1074264  0.6083830  28.119  0.00126 **
## dryDesignFlowMGD -8.5604098  2.3437494  -3.652  0.06746 .
## pop_2018          0.0008011  0.0001919   4.176  0.05285 .
## pop_density_2018 -0.0015676  0.0004210  -3.724  0.06515 .
## Region.xNWR       3.4327461  0.6600789   5.201  0.03504 *
## Region.xWR        2.7520890  0.7572573   3.634  0.06807 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.3775 on 2 degrees of freedom
##   (2 observations deleted due to missingness)
```

```
## Multiple R-squared:  0.9471, Adjusted R-squared:  0.8148
## F-statistic: 7.158 on 5 and 2 DF,  p-value: 0.1271
```

```
library(scales)
```

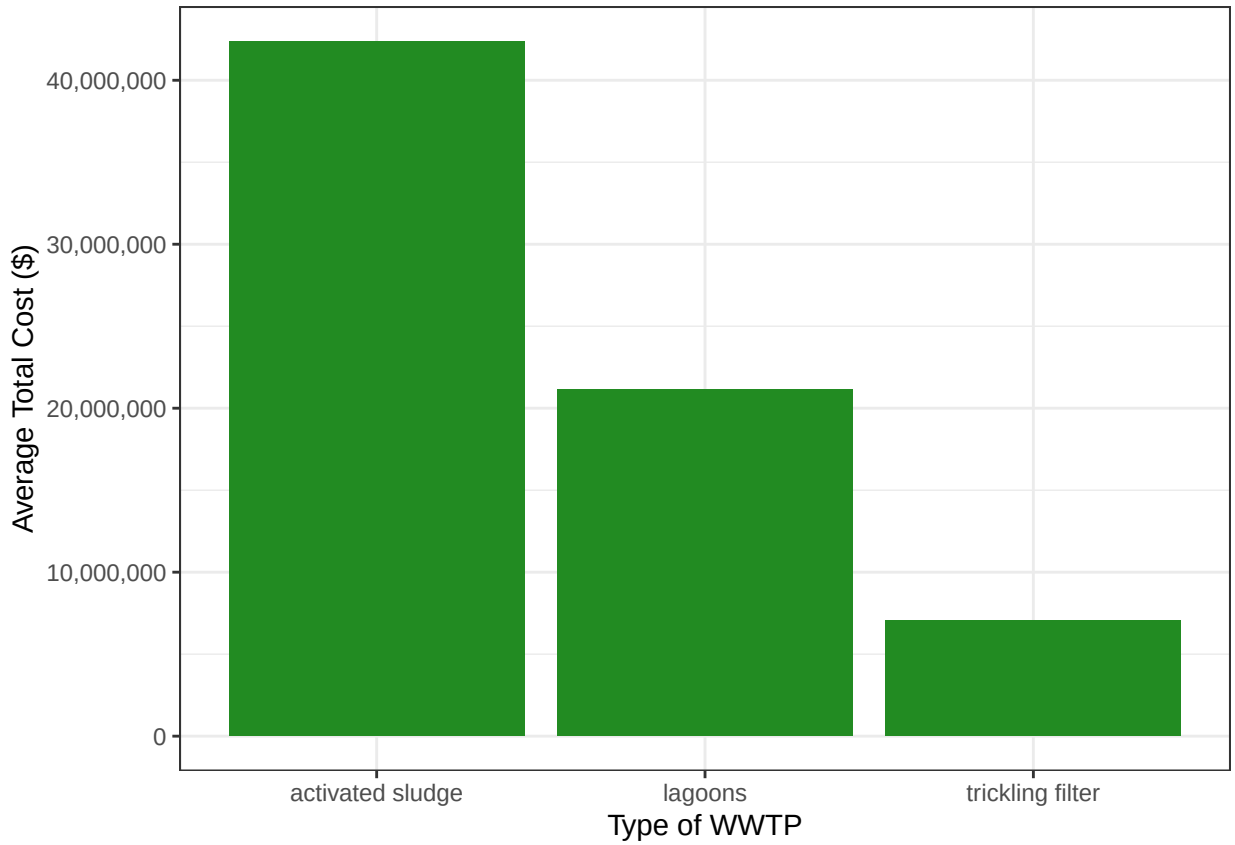
```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:viridis':
##
##     viridis_pal
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
point <- format_format(big.mark = ",", decimal.mark = ".", scientific = FALSE)
cost %>%
  ggplot(aes(x = type1,
             y = mean(`Total Cost`))) +
  geom_col(fill = "forest green") +
  labs(x = "Type of WWTP",
       y = "Average Total Cost ($)") +
  theme_bw() +
  scale_y_continuous(labels = point)
```




```
cost %>%
  group_by(type1) %>%
  summarize(n())
```

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

```
## # A tibble: 3 x 2
##   type1            `n()`
##   <chr>            <int>
## 1 activated sludge     6
## 2 lagoons              3
## 3 trickling filter     1
```