Rmd File

Learn Against The Machine

November 14, 2018

Instructions

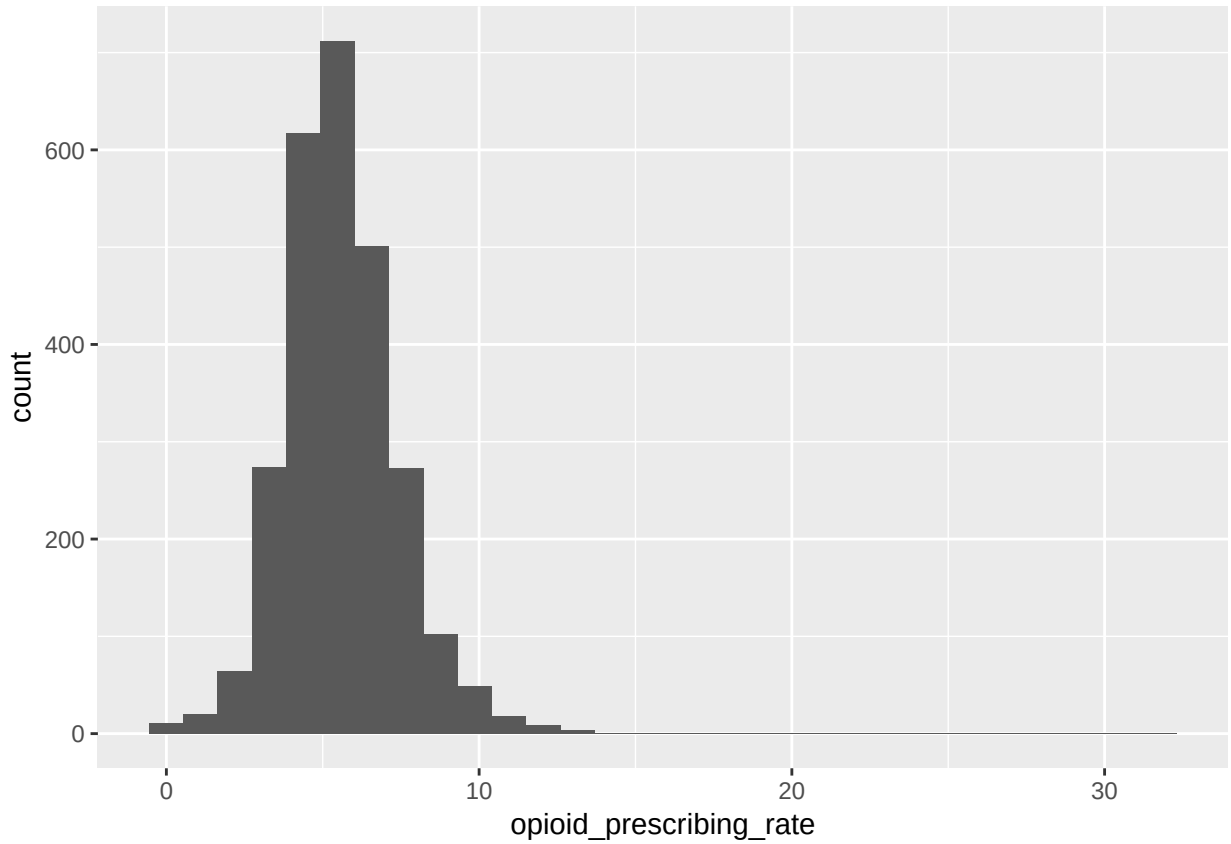
Attempt to implement at least 3 versions of your team's highest priority method as outlined in the revised proposal. This could look like:

If you are focusing on regression, you could: Run a few bivariate regression models Try a few subset selection methods Experiment with ridge regression and lasso

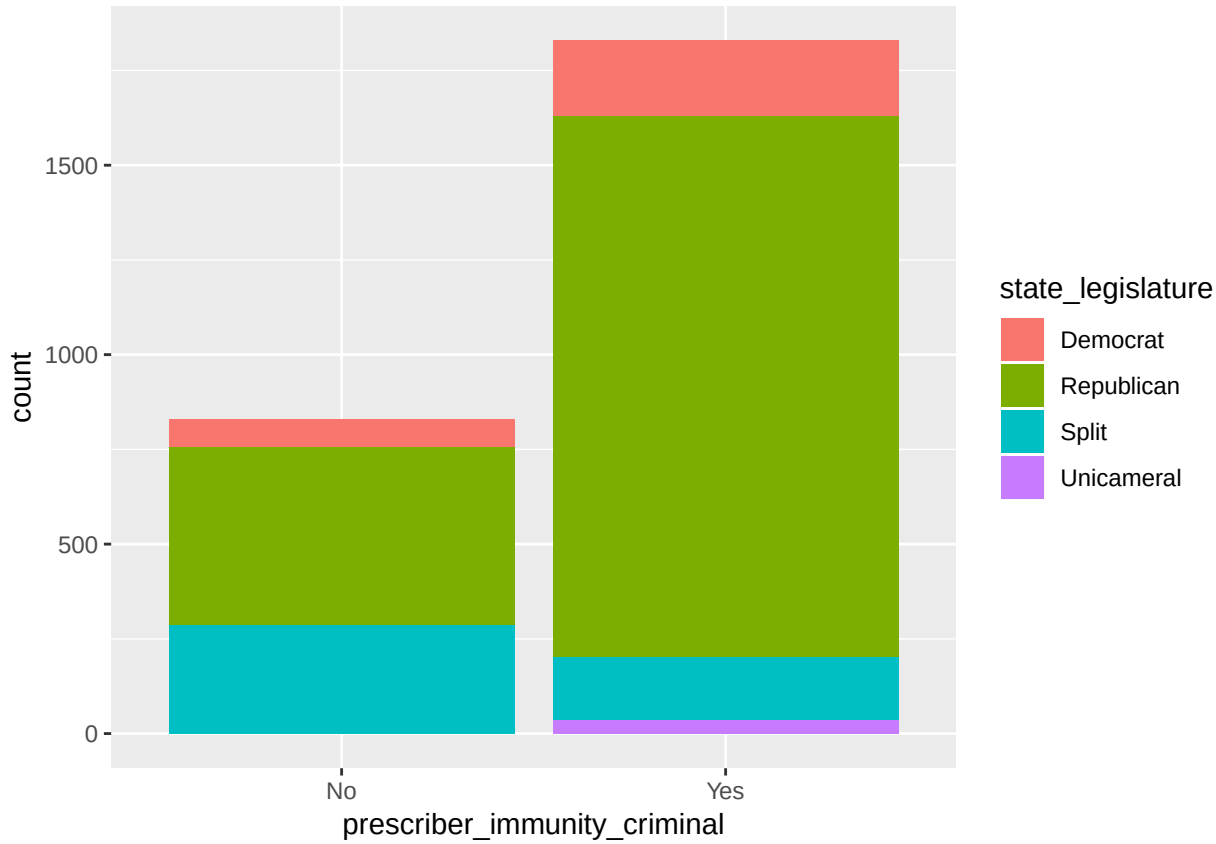
If you are prioritizing classification, you could try kNN, QDA, LDA, and Logistic regression (and comparing the assumptions violated in each) If your plan was to do classification and then regression, could the class on November 12 provide inspiration? Leverage cross-validation with different number of folds to tune various hyper-parameters (such as penalties or the k in kNN).

```
## reading in data files, joining to access prescribing behavior (response vars)
library(tidyverse)
library(corrplot)
County_Drug <- read_csv("County_Drug.csv")
prescribing_behavior <- read_csv("293 COUNTY DATA/prescribing_behavior.csv") %>%
  mutate(county_id = paste0("05000US", FIPS)) %>%
  subset(select = -c(`State Name`, `State Abbreviation`, `County Name`, `FIPS`))
colnames(prescribing_behavior) <- c("part_d_prescribers",
                                    "part_d_opioid_prescribers",
                                    "opioid_claims",
                                    "extended_release_opioid_claims",
                                    "overall_claims",
                                    "opioid_prescribing_rate",
                                    "extended_release_prescription_rate",
                                    "change_in_rate",
                                    "county_id")

## joining data sets
working_data <- County_Drug %>%
  inner_join(prescribing_behavior, by = "county_id") %>%
  na.omit() %>%
  subset(select = -c(X1))
```

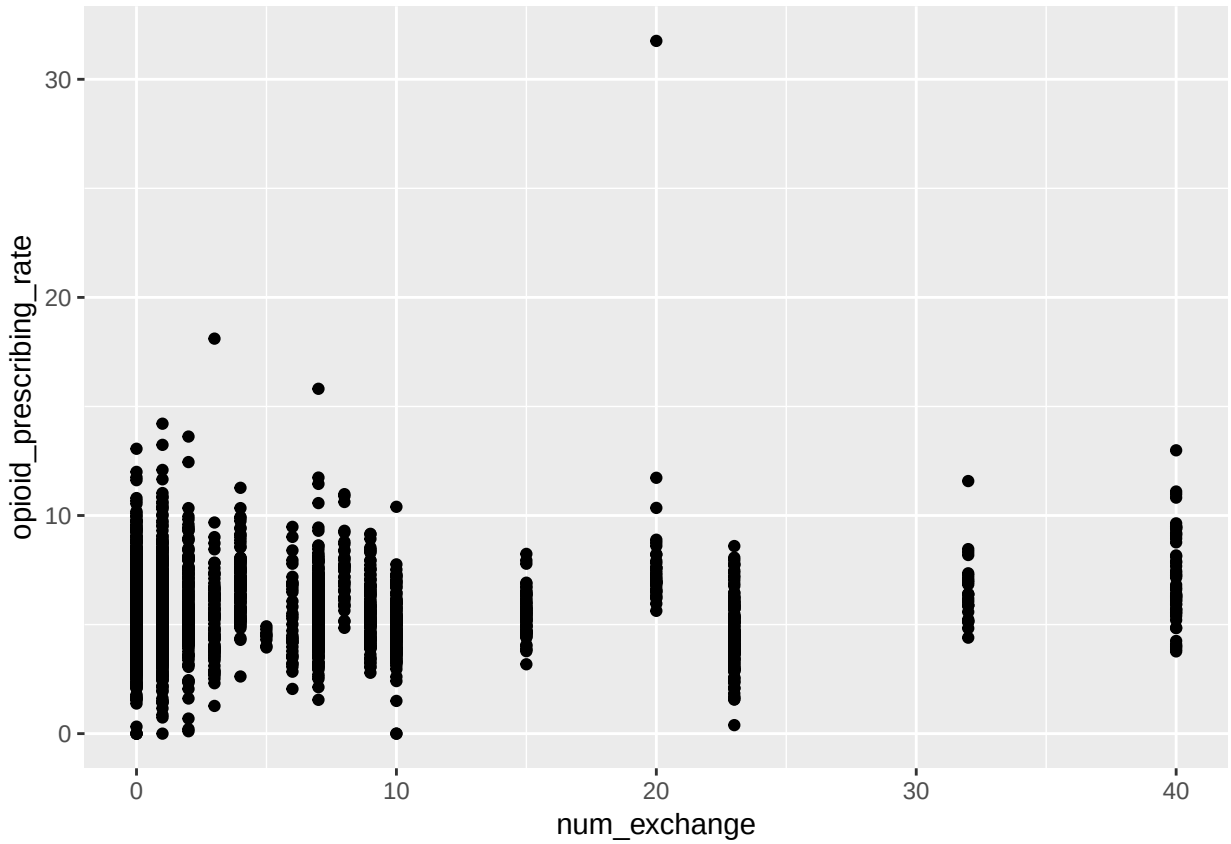
```
## some visualizations

ggplot(working_data, aes(x = opioid_prescribing_rate)) +
  geom_histogram()
```



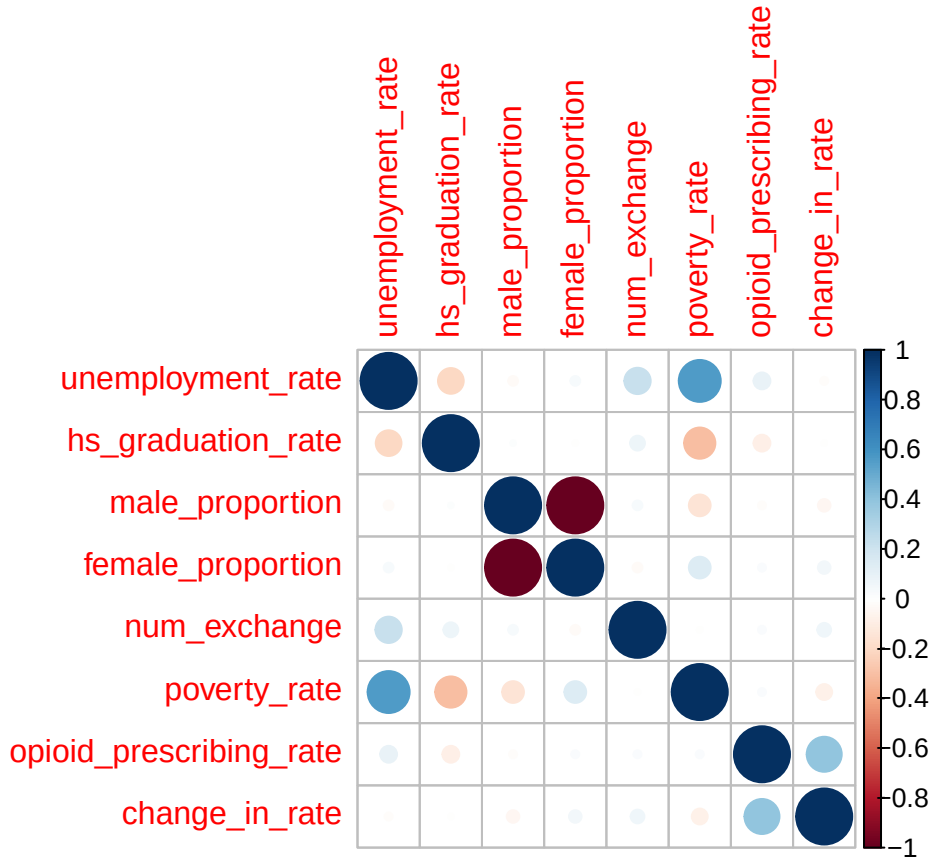

```
ggplot(working_data, aes(x = prescriber_immunity_criminal, fill = state_legislature)) +
  geom_bar()
```



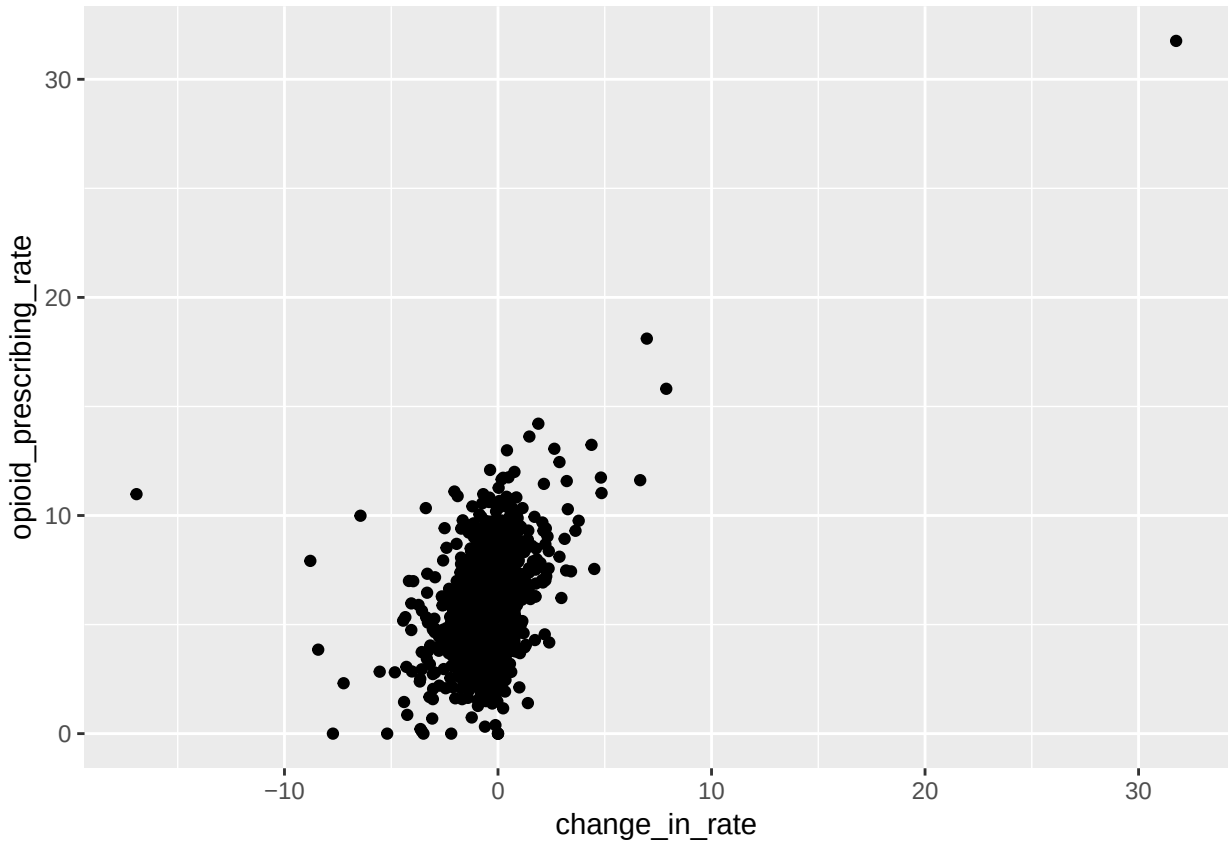

```
ggplot(working_data, aes(x = num_exchange, y = opioid_prescribing_rate)) +
  geom_point()
```




```
numeric_only <- working_data[, sapply(working_data, is.numeric)] %>%
  select(unemployment_rate,
         hs_graduation_rate,
         male_proportion,
         female_proportion,
         num_exchange,
         poverty_rate,
         opioid_prescribing_rate,
         change_in_rate)
cor_data <- cor(numeric_only) %>%
  round(2)
corrplot(cor_data, method = "circle")
```




```
ggplot(working_data, aes(x = change_in_rate, y = opioid_prescribing_rate)) +
  geom_point()
```



Plot 1: Checking normality of response variable (opioid prescription rate)

The first graph in this series of initial visualizations is meant to help determine if the values of our response variable are distributed normally among the many counties (n > 2,000) that we will be using for analysis. We observe from this graph that the values are centered between 5% and 6%, and that they do not show any notable skew towards higher or lower values. If we decide to pursue regression, this is a huge positive.

Plot 2: Examining makeup of state legislatures versus state harm reduction policies

We wanted to get a look at whether our initial intuition–that democrat-leaning states would represent a greater proportion of the states with comprehensive harm reduction policies–holds up within this data set. Actually, we note that in general, republican-leaning counties make up a larger proportion of our sample, and while they do make up the largest proportion of states without comprehensive HR policies, they also make up the largest proportion of states that do have these measures in place. This leads us to consider the results we might observe in our future analysis, which is based around the intution that more conservative states/counties will be less progressive in their harm reduction policy and have higher opioid prescription rates.

Plot 3: Comparing harm reduction policy with opioid-prescription rate (state-level)

With this plot, we wanted to get a sense of whether basic harm reduction initatives, like needle exchanges, showed any obvious relationship with opioid prescribing rate (the intuition being that states implementing

HR policies may also be taking steps to curb their opioid prescribing rate). We actually saw that there is a slight positive association between the number of needle exchanges and the prescribing rate, which in a way also makes a certain amount of sense. However, this association is relatively weak.

Plot 4: Correlation of variables of interest (numeric)

As we are limited to calculating a correlation matrix for numeric variables, this is what we decided to do. We omitted variables associated with race because they would have made the table too large, and this is only a preliminary look. We observed that high correlation ($0.5 < |\text{cor}|$) is only really occurring where it is to be expected, between the proportions of male and female residents, as well as between the poverty rate and the unemployment rate. We might consider evaluating this second set of variables for relevance; perhaps one should be removed to lessen the potential for multicolinearity to interfere with model fitting.

Plot 5: Scatterplot of opioid prescription rate versus change in that rate (since 2013)

This final scatterplot was created in order to investigate the relationship between prescription rate and the change in this rate since 2013. We wanted to evaluate whether high prescribing rates were coupled with significant decreases or increases in those rates over the temporal span between 2013 and 2015; we observed that the changes in rates are primarily clustered in the range [-5, 5] and that these changes guide prescription rates to within [1, 10], primarily. There is a single highly unusual county indicated as having both a prescribing rate and change in rate of 31.76; after looking specifically at this county, we have determined that perhaps prescription rate data was not available for this area in 2013, and so the "change in rate" simply jumped when data were eventually added. We will consider whether to include this case if change in prescription rate becomes a variable we'd like to explore further.

```
## numerical summaries
library(mosaic)
library(Hmisc)

favstats(working_data$population)
```

```
##   min       Q1  median    Q3      max     mean       sd    n missing
##  1084 16461.25 33573.5 85146 10150558 120263.3 357044.8 2658       0
```

```
describe(working_data$opioid_prescribing_rate)
```

```
## working_data$opioid_prescribing_rate
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     2658        0      734        1    5.593    1.974    3.015    3.590
##      .25      .50      .75      .90      .95
##    4.393    5.430    6.600    7.833    8.682
##
## lowest :  0.00  0.11  0.21  0.32  0.39, highest: 13.62 14.21 15.81 18.11 31.76
```

```
working_data %>%
  group_by(state_legislature) %>%
  summarise(mean = mean(opioid_prescribing_rate))
```

```
## # A tibble: 4 x 2
##   state_legislature  mean
##   <chr>             <dbl>
## 1 Democrat           5.65
## 2 Republican         5.69
```

```
## 3 Split              5.22
## 4 Unicameral         4.86
```

```
working_data %>%
  group_by(state) %>%
  summarise(avg_rate = mean(opioid_prescribing_rate)) %>%
  arrange(avg_rate) %>%
  tail(5)
```

```
## # A tibble: 5 x 2
##   state      avg_rate
##   <chr>         <dbl>
## 1 Oregon         7.27
## 2 Colorado       7.38
## 3 Utah           8.10
## 4 Washington     8.10
## 5 Nevada         8.29
```

```
working_data %>%
  group_by(state) %>%
  summarise(avg_rate = mean(opioid_prescribing_rate)) %>%
  arrange(avg_rate) %>%
  head(5)
```

```
## # A tibble: 5 x 2
##   state         avg_rate
##   <chr>            <dbl>
## 1 Rhode Island      3.66
## 2 New York          4.02
## 3 North Dakota      4.08
## 4 Massachusetts     4.37
## 5 New Jersey        4.42
```

Summary 1: Quartile distribution of county population sizes

We thought it might be pertinent to our investigation to have some sense of the range of population sizes represented in our sample. We see that the median size is 33573 while the mean is 120263, indicating that this distribution is likely skewed towards higher populations.

Summary 2: Highest and lowest opioid prescription rates

In order to get a better sense of what "normal" versus "extreme" might look like in the sense of opioid prescription rate, we took a look at the five lowest and five highest prescription rates. We wanted to see, first and foremost, if the highest rates were clustered closely together, or if there were some notable standouts. We were also interested in determining what these values tend to look like. We were able to observe that all but the highest prescribing rate fall below 19% (the highest is >30%). We were also able to determine that the lowest values do not share similar variation.

Summary 3: Mean opioid prescription rate by state legislature composition

We wanted to determine whether there might be a significant association between political party and prescribing rates. After running these numbers, we observed that Republican and Democratic legislatures have nearly identical mean rates of prescription, and split legislatures are only slightly lower. Nebraska is

a unicameral legislature and has an even lower mean, though we are chalking this up to it's status as the singular unicameral legislative body in the nation.

Summary 4: Highest prescribing states

As we began to wind down our preliminary visualizations, we decided to determine which 5 states had the highest average (mean) prescribing rates and which had the lowest. The highest states included Nevada, Washington, Utah, Colorado, and Oregon, while the lowest were Rhode Island, New York, North Dakota, Massachusetts, and New Jersey (both lists in descending order). This summary has helped us to better observe the presence of a regional divide in prescribing practices (east vs. west), and it presents some interesting ideas as we think about ways in which we may want to expand on our data (i.e. categorizing states by region to perform more aggregate analyses).

```
## methods set up
library(class)

working_data$over_avg_rx_rate = 0
working_data$over_avg_rx_rate[working_data$opioid_prescribing_rate >median(working_data$opioid_prescrib

## setting up testing & training datasets
working_data$id <- 1:nrow(working_data)
train = working_data %>% dplyr::sample_frac(.75)
test = dplyr::anti_join(working_data, train, by = 'id')

set.seed(1)
```

```
## logistic regression
glm_fit = glm(over_avg_rx_rate ~ unemployment_rate +
                hs_graduation_rate +
                average_age +
                population +
                male_proportion +
                num_exchange +
                poverty_rate,
                  data=working_data,
                  family = "binomial")
summary(glm_fit)
```

```
##
## Call:
## glm(formula = over_avg_rx_rate ~ unemployment_rate + hs_graduation_rate +
##     average_age + population + male_proportion + num_exchange +
##     poverty_rate, family = "binomial", data = working_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7687  -1.1398  -0.9326   1.1737   1.4788
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)         3.657e+00  1.222e+00   2.993  0.00276 **
## unemployment_rate   6.205e+00  2.082e+00   2.981  0.00288 **
## hs_graduation_rate -1.986e+00  4.556e-01  -4.359 1.31e-05 ***
## average_age        -1.950e-02  8.705e-03  -2.240  0.02506 *
```

```
## population          1.312e-08  1.167e-07   0.112  0.91047
## male_proportion    -3.144e-02  2.033e-02  -1.546  0.12200
## num_exchange       -3.582e-03  4.871e-03  -0.735  0.46211
## poverty_rate       -6.677e-01  8.415e-01  -0.793  0.42752
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3684.7  on 2657  degrees of freedom
## Residual deviance: 3638.2  on 2650  degrees of freedom
## AIC: 3654.2
##
## Number of Fisher Scoring iterations: 4
```

```
glm_party_fit <- glm(over_avg_rx_rate ~
                       unemployment_rate +
                       hs_graduation_rate +
                       average_age +
                       white_proportion +
                       black_proportion +
                       american_indian_proportion +
                       asian_proportion +
                       hawaiian_pacific_proportion +
                       interracial_proportion +
                       hispanic_proportion +
                       state_legislature,
                     data = working_data,
                     family = "binomial")

summary(glm_party_fit)
```

```
##
## Call:
## glm(formula = over_avg_rx_rate ~ unemployment_rate + hs_graduation_rate +
##     average_age + white_proportion + black_proportion + american_indian_proportion +
##     asian_proportion + hawaiian_pacific_proportion + interracial_proportion +
##     hispanic_proportion + state_legislature, family = "binomial",
##     data = working_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7228  -1.0912  -0.5594   1.1081   1.9409
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -7.072e+08  1.215e+09  -0.582 0.560561
## unemployment_rate           1.186e+01  1.958e+00   6.056 1.39e-09 ***
## hs_graduation_rate         -3.004e+00  5.337e-01  -5.629 1.81e-08 ***
## average_age                -3.296e-02  9.930e-03  -3.319 0.000902 ***
## white_proportion            7.072e+06  1.215e+07   0.582 0.560561
## black_proportion            7.072e+06  1.215e+07   0.582 0.560561
## american_indian_proportion  7.072e+06  1.215e+07   0.582 0.560561
## asian_proportion            7.072e+06  1.215e+07   0.582 0.560561
## hawaiian_pacific_proportion 7.072e+06  1.215e+07   0.582 0.560561
```

```
## interracial_proportion       7.072e+06  1.215e+07   0.582 0.560561
## hispanic_proportion         -5.116e-03  3.306e-03  -1.548 0.121681
## state_legislatureRepublican  6.288e-01  1.508e-01   4.169 3.06e-05 ***
## state_legislatureSplit      -2.565e-01  1.711e-01  -1.499 0.133952
## state_legislatureUnicameral -2.401e-01  4.268e-01  -0.563 0.573703
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3684.7  on 2657  degrees of freedom
## Residual deviance: 3405.9  on 2644  degrees of freedom
## AIC: 3433.9
##
## Number of Fisher Scoring iterations: 4
```

```
glm_sig_fit <- glm(over_avg_rx_rate ~ unemployment_rate + hs_graduation_rate + average_age + state_legis

glm_probs = data.frame(probs = predict(glm_sig_fit, newdata = test, type = "response"))
glm_pred <- glm_probs %>%
  mutate(pred = ifelse(probs > 0.5, 1, 0))

glm_pred <- cbind(test, glm_pred)
glm_pred %>%
  count(pred, over_avg_rx_rate) %>%
  spread(over_avg_rx_rate, n, fill = 0)
```

```
## # A tibble: 2 x 3
##    pred   `0`   `1`
##   <dbl> <dbl> <dbl>
## 1     0   178   123
## 2     1   158   205
```

```
glm_pred %>%
  summarise(score = mean(pred == over_avg_rx_rate),
            recip = mean(pred != over_avg_rx_rate))
```

```
##       score     recip
## 1 0.5768072 0.4231928
```

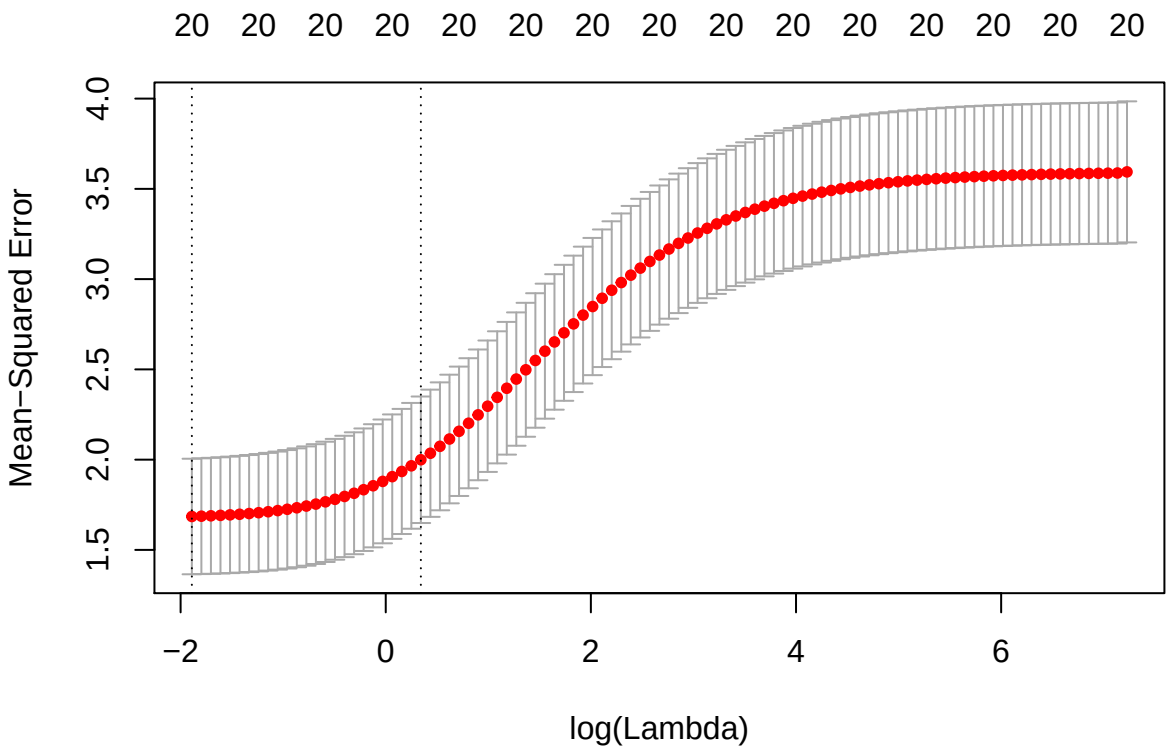
Method 1: Logistic Regression

For this first attempt, we considered unemployment rate, high school graduation rate, average age, population, male proportion, number of needle exchange programs, and poverty rate as predictors of opioid prescribing rate (above or below average, when looking at the country as a whole). Based on p-values, it looks like population, male proportion, number of needle exchanges, and poverty rate are unlikely to be strongly related to over-average prescription of opioids. The smallest p-value here is for high school graduation rate, and the negative coefficient suggests that as high school gradation rate in a county rises, the rate of opiod prescription is less likely to be greater than average. The p-values for both high school graduation rate and unemployment rate are very small, less than 0.01, so there's evidence that there is an association between them and prescribing rate. One concern for this step is it assumes the predictors are independent, and we actually found some correlation between high school graduation rate and unemployment rate in plot 4.

In our second attempt at Logistic Regression (glm_party_fit), we incorporated our variable of most interest, which is a four-level indicator of state legislature party dominance, as well as our assortment of

race and ethnicity-related variables (given as proportions of the counties' populations). The variable `state-legislature` reports party dominance (Republican or Democrat) in the following manner: "Democrat" indicates Democratic control of both houses, "Republican" represents Republican control of both houses, "Split" indicates Democratic control in one house and Republican control in the other, and "Unicameral" applies only to counties in Nebraska (the state has a single legislative house). Democratic control was set as our reference group. We observe in the summary output that Republican control of both houses is significantly associated with an increase in the liklihood that a county will have an opioid prescription rate above the median (p-value < 0.01). This is highly encouraging, as this is the result we expected to see, based on our intuition. The test error for this model is approximately 40%, which is better than random chance, but perhaps not ideal.

```
## Ridge regression for logistic regression
set.seed(1)
library(glmnet)
working_full <- na.omit(working_data) %>%
  select(unemployment_rate, hs_graduation_rate, average_age, population, male_proportion, white_proporti

working_full$over_avg_rx_rate = 0
working_full$over_avg_rx_rate[working_data$opioid_prescribing_rate >median(working_data$opioid_prescribi

train <- working_full %>%
  sample_frac(0.75)
test <- working_full %>%
  setdiff(train)

x_train <- model.matrix(opioid_prescribing_rate ~ ., train)[,-19]
x_test <- model.matrix(opioid_prescribing_rate ~ ., test)[,-19]

y_train <- train %>%
  select(opioid_prescribing_rate) %>%
  unlist() %>%
  as.numeric()

y_test <- test %>%
  select(opioid_prescribing_rate) %>%
  unlist() %>%
  as.numeric()

grid <- 10^seq(10, -2, length = 100)
cv_ridge <- cv.glmnet(x_train, y_train, alpha = 0)
plot(cv_ridge)
```



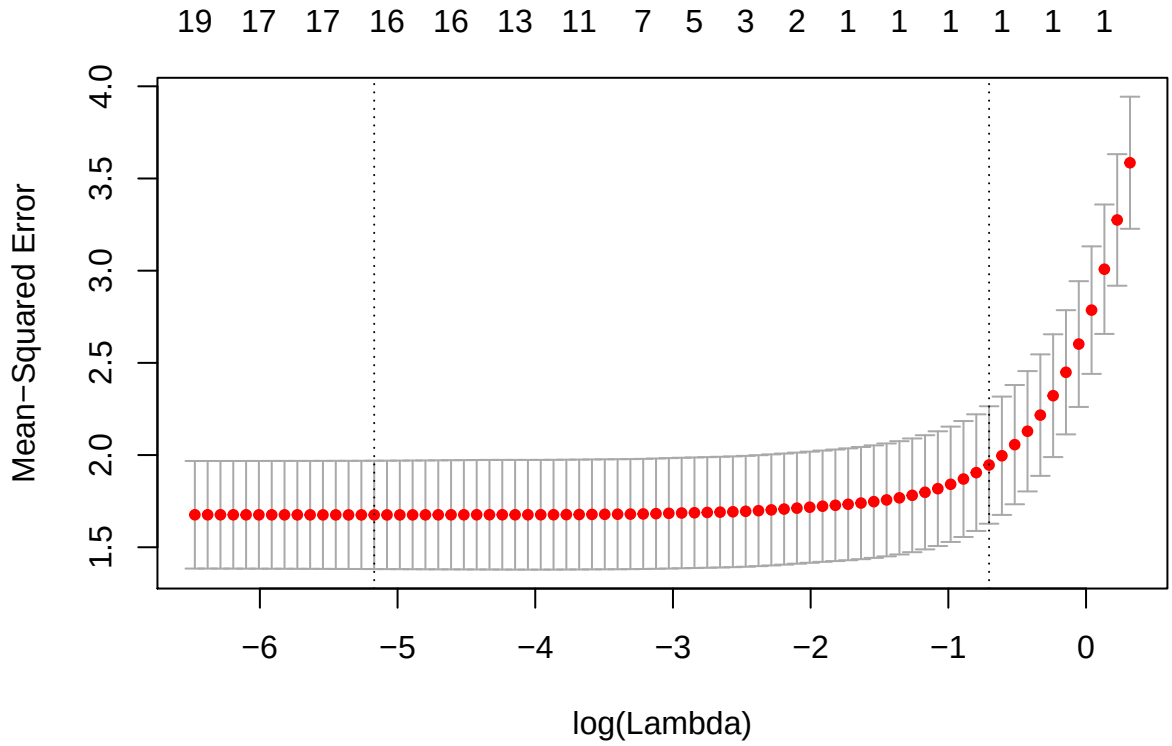

```
bestlam <- cv_ridge$lambda.min

ridge_pred <- predict(cv_ridge, s = bestlam, newx = x_test)
mean((ridge_pred - y_test)^2)
```

```
## [1] 1.225699
```

```
out <- glmnet(x_test, y_test, alpha = 0)
predict(out, type = "coefficients", s = bestlam)[1:20,]
```

```
##                      (Intercept)                      (Intercept)
##                     4.530186e+00                     0.000000e+00
##                unemployment_rate                hs_graduation_rate
##                     8.336178e+00                    -1.861613e+00
##                      average_age                       population
##                    -2.880057e-03                    -7.712540e-08
##                  male_proportion                  white_proportion
##                     9.265885e-03                     7.797521e-03
##                 black_proportion       american_indian_proportion
##                    -8.377829e-03                    -2.131598e-03
##                 asian_proportion     hawaiian_pacific_proportion
##                    -3.551200e-02                    -4.847956e-02
##           interracial_proportion              hispanic_proportion
##                     1.498295e-01                    -3.612942e-03
##     state_legislatureRepublican           state_legislatureSplit
##                    -1.470080e-02                    -2.629368e-01
##     state_legislatureUnicameral              syringe_exchangeYes
##                     2.074923e-01                    -1.296309e-02
##                    num_exchange prescriber_immunity_criminalYes
##                     3.193547e-03                    -8.078835e-02
```

```
###################################
## Lasso

lasso_cv <- cv.glmnet(x_train, y_train, alpha = 1)
plot(lasso_cv)
```




```
bestlam <- lasso_cv$lambda.min

lasso_mod <- glmnet(x_train, y_train, alpha = 1, lambda = grid)

lasso_pred <- predict(lasso_mod, s = bestlam, newx = x_test)
mean((lasso_pred - y_test)^2)
```

```
## [1] 1.227725
```

```
lasso_coef <- predict(lasso_mod, type = "coefficients", s = bestlam)[1:20,]
lasso_coef
```

```
##                      (Intercept)                      (Intercept)
##                    5.033545e+00                     0.000000e+00
##                unemployment_rate                hs_graduation_rate
##                    3.626397e+00                    -3.873673e-01
##                     average_age                       population
##                    -3.071478e-03                    -9.021040e-08
##                  male_proportion                  white_proportion
##                    -1.662913e-02                     2.471422e-03
##                  black_proportion      american_indian_proportion
##                    -6.204553e-03                     0.000000e+00
##                  asian_proportion     hawaiian_pacific_proportion
##                    -1.152377e-02                    -1.410375e-01
##           interracial_proportion             hispanic_proportion
```

```
##                    1.282987e-01                   -3.260596e-03
##     state_legislatureRepublican          state_legislatureSplit
##                    1.625233e-02                    0.000000e+00
##     state_legislatureUnicameral             syringe_exchangeYes
##                    0.000000e+00                    1.734203e-02
##                    num_exchange prescriber_immunity_criminalYes
##                    7.379746e-04                    0.000000e+00
```

```
lasso_coef[lasso_coef != 0]
```

```
##                  (Intercept)           unemployment_rate
##                 5.033545e+00                3.626397e+00
##           hs_graduation_rate                average_age
##                -3.873673e-01               -3.071478e-03
##                   population             male_proportion
##                -9.021040e-08               -1.662913e-02
##             white_proportion            black_proportion
##                 2.471422e-03               -6.204553e-03
##             asian_proportion hawaiian_pacific_proportion
##                -1.152377e-02               -1.410375e-01
##       interracial_proportion         hispanic_proportion
##                 1.282987e-01               -3.260596e-03
## state_legislatureRepublican         syringe_exchangeYes
##                 1.625233e-02                1.734203e-02
##                 num_exchange
##                 7.379746e-04
```

Method 2: We considered approaching ridge regression and lasso for determining regression predictors (especially since our opioid prescription rate can be given as either a continuous numeric variable or a categorial variable). For both methods, test MSE was greater than 2, with a slight improvement (-0.2) when using lasso to select predictors. This is not tremendously encouraging. Based on these results, and our desire to be able to provide clear and consise interpretations of coefficients and model fit (as is often the priority in public health research, when possible), we feel that fine-tuning a logistic regression model may be our best bet to be able to produce salient results with interpretability.

```
library(class)
## knn
knn_train = train %>%
  select(unemployment_rate, hs_graduation_rate)
knn_test = test %>%
  select(unemployment_rate, hs_graduation_rate)
knn_train_rx_rate = train %>%
  select(over_avg_rx_rate) %>%
  .$over_avg_rx_rate
knn_test_rx_rate = test %>%
  select(over_avg_rx_rate) %>%
  .$over_avg_rx_rate

knn_pred_1 = knn(knn_train,
                 knn_test,
                 knn_train_rx_rate,
                 k = 1)

table(knn_pred_1, knn_test_rx_rate)
```

```
##           knn_test_rx_rate
```

```
## knn_pred_1   0   1
##          0 178 174
##          1 148 164
```

```
mean(knn_pred_1 == knn_test_rx_rate)
```

```
## [1] 0.5150602
```

```
knn_pred_3 = knn(knn_train,
                 knn_test,
                 knn_train_rx_rate,
                 k = 3)

table(knn_pred_3, knn_test_rx_rate)
```

```
##           knn_test_rx_rate
## knn_pred_3   0   1
##          0 186 167
##          1 140 171
```

```
mean(knn_pred_3 == knn_test_rx_rate)
```

```
## [1] 0.5376506
```

```
knn_pred_5 = knn(knn_train,
                 knn_test,
                 knn_train_rx_rate,
                 k = 5)

table(knn_pred_5, knn_test_rx_rate)
```

```
##           knn_test_rx_rate
## knn_pred_5   0   1
##          0 181 162
##          1 145 176
```

```
mean(knn_pred_5 == knn_test_rx_rate)
```

```
## [1] 0.5376506
```

```
knn_pred_10 = knn(knn_train,
                  knn_test,
                  knn_train_rx_rate,
                  k = 10)

table(knn_pred_10, knn_test_rx_rate)
```

```
##            knn_test_rx_rate
## knn_pred_10   0   1
##           0 178 169
##           1 148 169
```

```
mean(knn_pred_10 == knn_test_rx_rate)
```

```
## [1] 0.5225904
```

Method 3: KNN

In this attempt, we used KNN with high school graduation rate and unemployment rate. In this case, after trying with several different values of k (1, 3, 5, 10), we found that we were predicting with around 50% accuracy, and that there wasn't a significant change in accuracy when we changed k, indicating that perhaps KNN isn't the right method here.