Regression Report: Distance to nearest Syringe Services Program

Nicole Eberle

Contents

Distance to nearest Syringe Services Program **2**

EXECUTIVE SUMMARY **2**

INTRODUCTION **2**

DATA **3**

Summary Statistics . . . 3

Qualitative Predictor Variables . . . 4

Quantitative Predictor Variables . . . 6

Relationships Between Variables . . . 9

Variable Transformations . . . 14

ANALYSIS - Regression Models **16**

Basic Model - All predictors, No Transformations . . . 16

Additional Models - Target: Dist_ssp . . . 18

Additional Models - Target: sqrt_dist_ssp . . . 18

Interactions between predictor variables . . . 19

Interaction Plots: pct_unins . . . 20

Interaction Plots: opioid_rx_rate . . . 21

Interaction Model Results . . . 21

Final Regression Model - Predicting sqrt_dist_ssp using interaction variables between opioid_rx_rate and hiv_prevalence along with matero and pct_unins . . . 22

Check of Assumptions . . . 22

Outliers with Studentized Residuals . . . 22

Influential Observations and Leverage . . . 23

Assumption 1: Little to no multicollinearity between predictors . . . 25

Assumption 2: Errors (Residuals) are normally distributed . . . 26

Assumption 3: Homoscedasticity of errors (or, equal variance around the 0 across fitted y values). . . . 27

Assumption 4: Independence of the observations 30
Most Important Predictors 30
Conclusions 31

APPENDIX A: All Models **32**

APPENDIX B: The Code **33**

Distance to nearest Syringe Services Program

EXECUTIVE SUMMARY

This dataset examines the distance varying counties are from the nearest Syringe Services Program (SSP). The dataset provides the HIV Prevalence, Opioid Rx Rate, and Percentage Uninsured for each county, along with county categorical information such as state, region and metro/non-metro. The focus of the analysis was to predict the distance to the nearest Syringe Services Program based on county characterisitics.

Based on our diagnostics, the regression model predicting sqrt_dist_ssp using both interaction variables is the strongest model. This model still has room for improvement as seen by the adjusted R^2 of 20.4% and large residuals. The most important predictors in the model are pct_unins, opioid_rx_rate and metronon-metro. The final version of the regression equation can be seen here:

dist_ssp = (6.0097584 + -0.0102525(hiv_prevalence) + -0.0406647(opioid_rx_rate) + 0.3691861(pct_unins) + 4.134465(metronon-metro) + 1.0699818(region3North_south) + 1.5697619×10^{-4}(hiv_prevalence*opioid_rx_rate) + -0.1820985(pct_unins*metronon-metro))^2

Overall, this is not a very strong model and definitely still has room for improvement, despite the many variations tried. Additional data cleaning may be necessary, along with compare the model to a version created without the observations that have high leverage.

r.squared	adj.r.squared	sigma	statistic	p.value	df	df.residual	nobs
0.216	0.204	3.752	18.447	0	7	470	478

term	estimate	std.error	statistic	p.value
(Intercept)	6.00976	0.88760	6.77082	0.00000
hiv_prevalence	-0.01025	0.00298	-3.44566	0.00062
opioid_rx_rate	-0.04066	0.00880	-4.62119	0.00000
pct_unins	0.36919	0.06285	5.87373	0.00000
metronon-metro	4.13447	1.00722	4.10482	0.00005
region3North_south	1.06998	0.39724	2.69352	0.00732
hiv_prevalence:opioid_rx_rate	0.00016	0.00004	4.00183	0.00007
pct_unins:metronon-metro	-0.18210	0.08060	-2.25919	0.02433

INTRODUCTION

This dataset examines the distance varying counties are from the nearest Syringe Services Program (SSP). The dataset provides the HIV Prevalence, Opioid Rx Rate, and Percentage Uninsured for each county, along with categorical information such as state, region and metro/non-metro. The focus of the analysis was to predict the distance to the nearest Syringe Services Program based on county characterisitics.

DATA

The dataset provides the HIV Prevalence, Opioid Rx Rate, Percentage Uninsured, and distance from nearest SSP for each county, along with categorical information such as state, region and metro/non-metro. Here is a detailed description of each variable:

Qualitative:

- county: county name
- state: two-letter abbreviation for state
- region: geographical region of United States (Northeast, South, West, Midwest)
- metro: country is nonmetro(open countryside, rural towns, or smaller cities with up to 49,999) or metro

Quantitative:

- dist_ssp: distance in miles to nearest syringe services program
- hiv_prevalence: people age 13 and older diagnosed with HIV per 100,000
- opioid_rx_rate: number opioid perscriptions per 100 people
- pct_unins: percentage civilian noninstitutionalized population with no health insurance coverage

Summary Statistics

In our analysis, we will not being using the county or state variables. County is a unique identifier for each row and State has too many categories to be useful, therefore we will be using region instead.

The summary statistics for the variables being used in analysis are shown below:

Table 3: Data summary

Name	df_ops
Number of rows	500
Number of columns	6
Column type frequency:	
character	2
numeric	4
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
metro	0	1	5	9	0	2	0
region	0	1	4	9	0	4	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p50	p100	hist
dist_ssp	0	1	107.74	94.23	0.0	75.94	510.0	
hiv_prevalence	0	1	165.75	208.97	-1.0	101.15	2150.7	
opioid_rx_rate	0	1	68.33	36.81	0.2	62.40	345.1	
pct_unins	0	1	12.18	4.97	3.0	11.70	35.9	

In the summary statistics, notice that the min value for hiv_prevalence is -1. This value does not make sense given the definition of the variable. Let's look and see how many counties have an hiv_prevalence of -1 or 0.

hiv_prevalence	count
-1	70

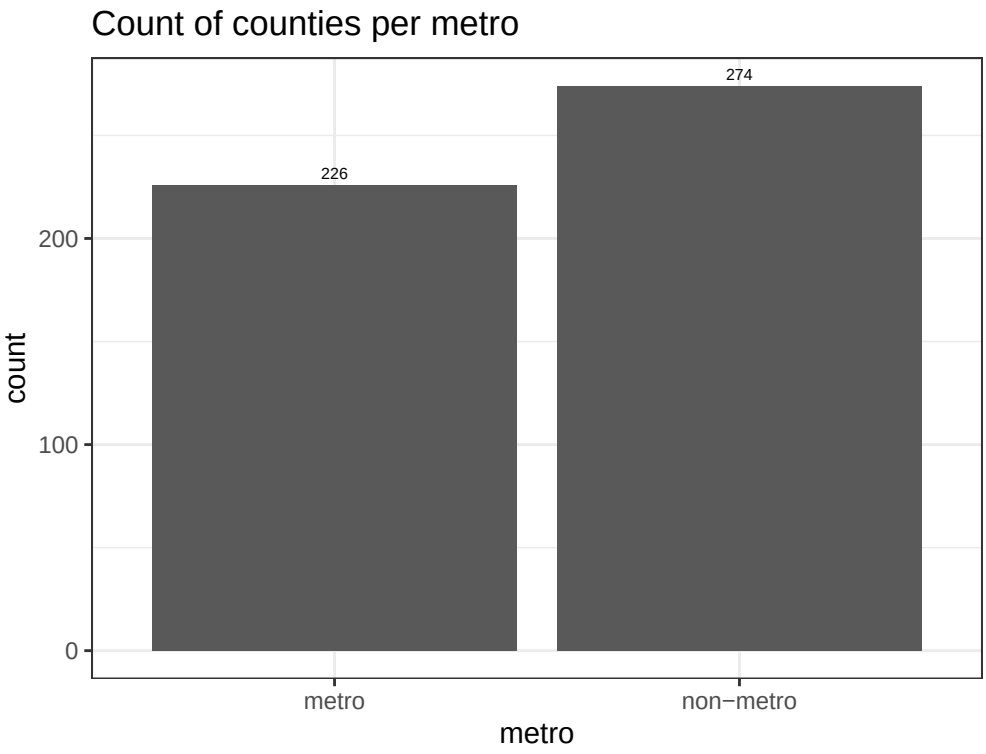
There are 70 counties that have -1 as their value for hiv_prevalence (14% of total counties). We can assume that -1 is a placeholder for unknown data in this instance. Having -1 as an input for hiv_prevalence in our model will likely lead to issues, particularly with trying to derive a coefficient for hiv_prevalence since the negative value would switch the sign of the coefficient. To avoid this, we are going to change all the values of -1 to be the mean of the other values of hiv_prevalence. Here are updated summary statistics for hiv_prevalence after replacing the -1 values:

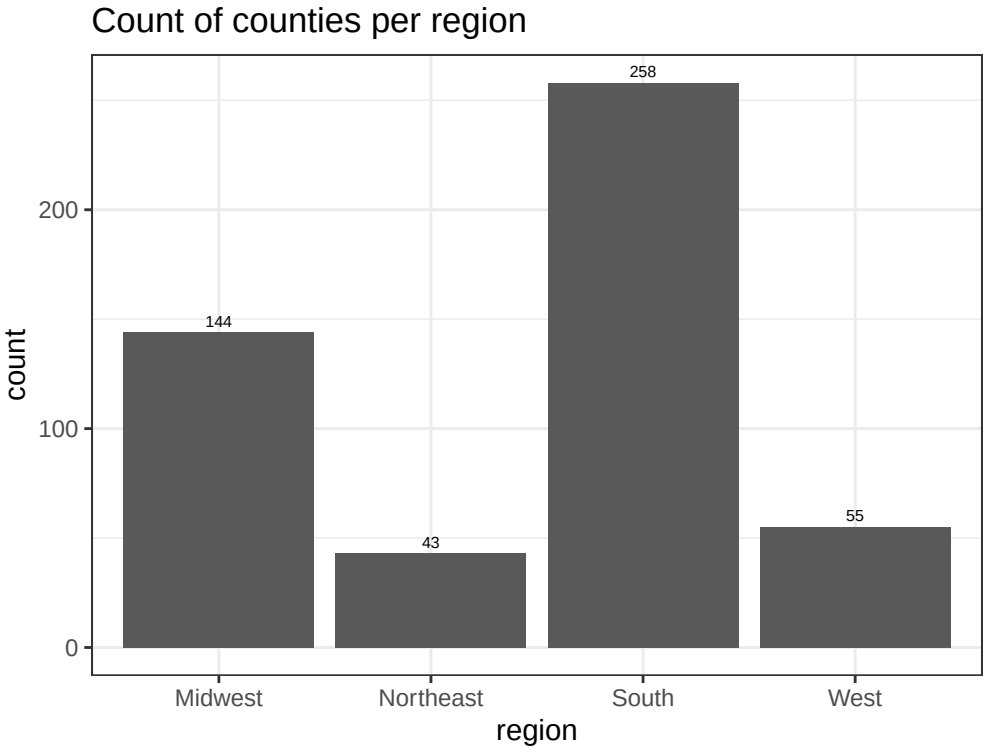
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  14.40   79.25  146.95  192.89  210.35 2150.70
```

We can now see that the minimum value for hiv_prevalence is 14.4. Next, let's look deeper at our variables to understand our data and see if any other cleaning/modifications need to take place.

Qualitative Predictor Variables

The qualitative predictor we are using are metro and region. Let's take a look at how many counties fall into each category for these variables.





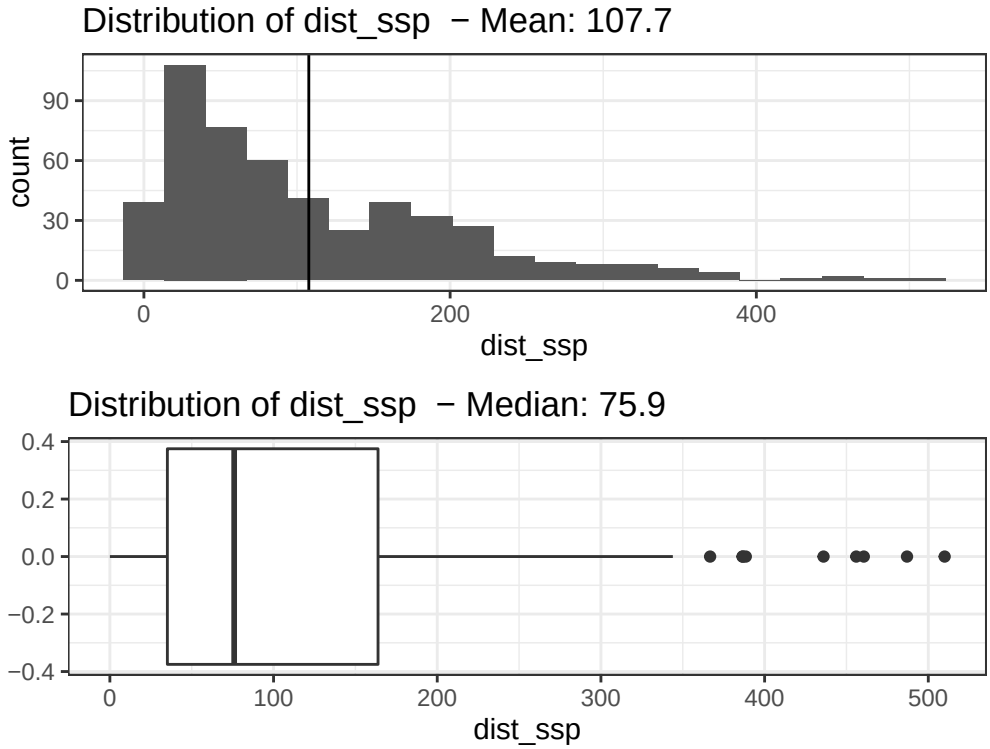
We can see the number of metro vs non-metro counties is fairly equal with 226 metro counties and 274 non-metro counties.

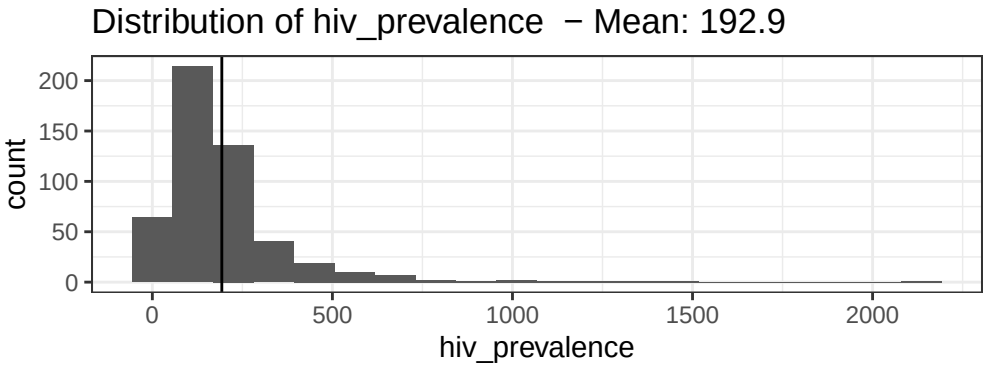
For region, the majority of counties fall in the South (258 counties) followed by the Midwest with 144, West

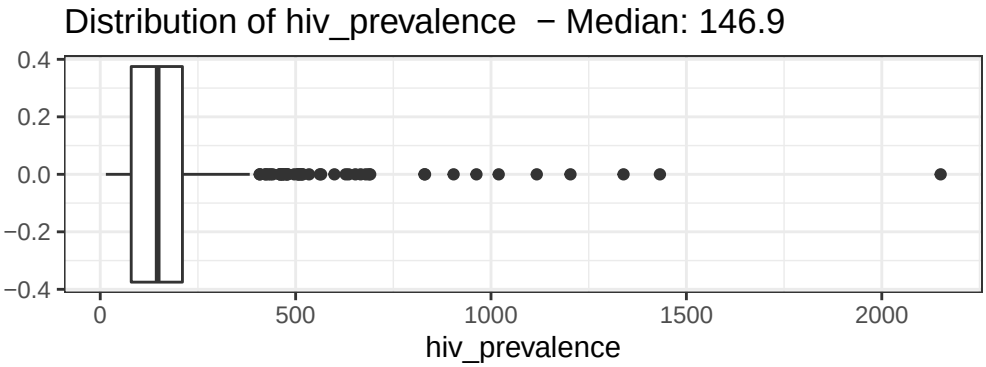
with 55 and Northeast with 43.

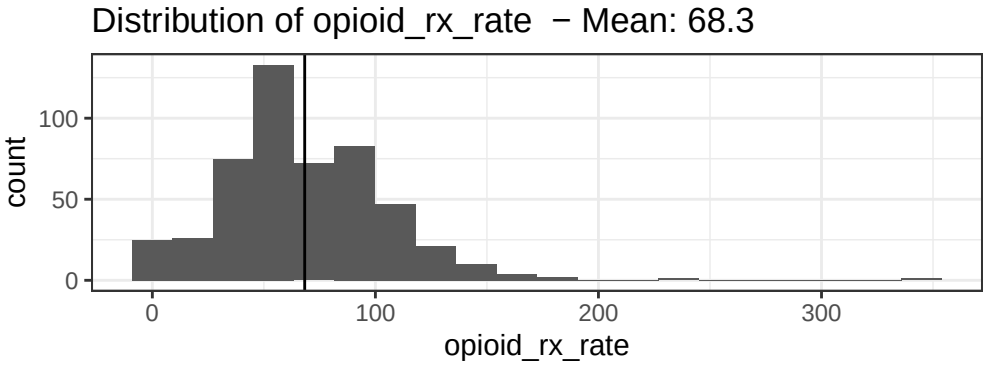
Quantitative Predictor Variables

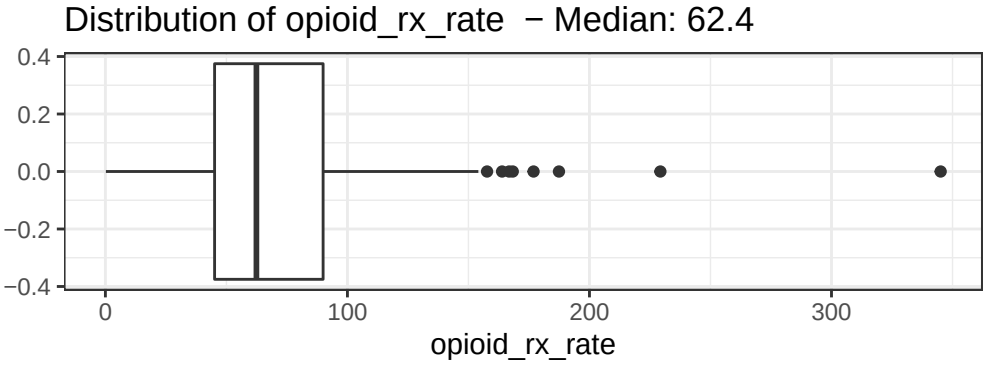
We have also examined our 3 quantitative predictors as well as our distance variable.

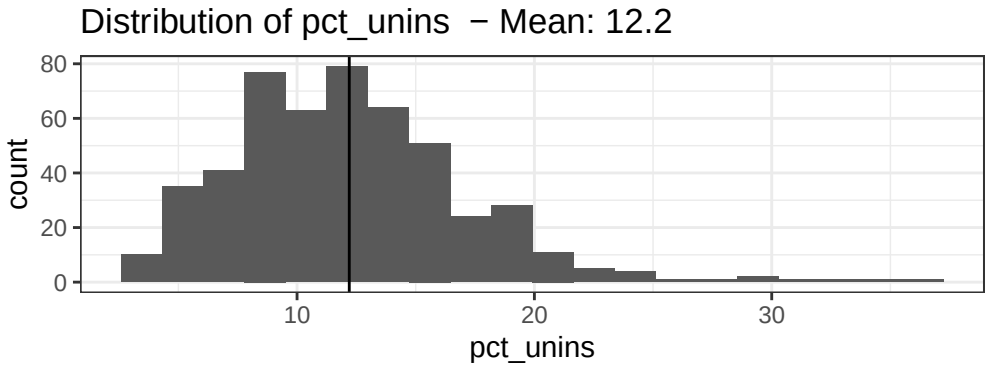


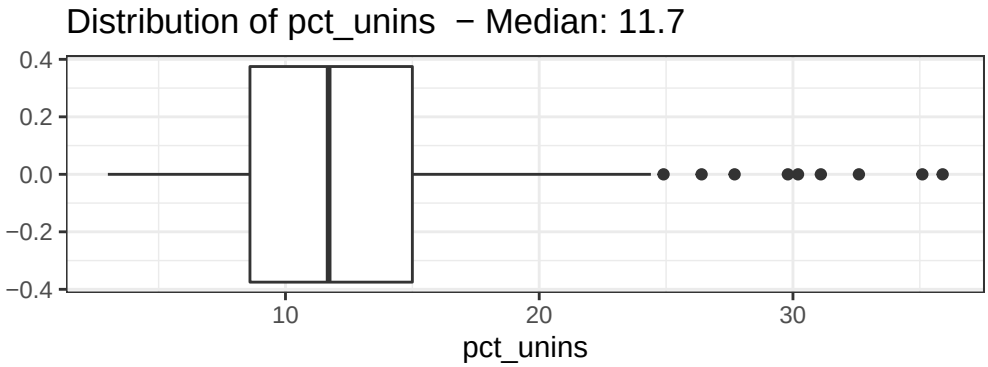

Distribution of hiv_prevalence − Mean: 192.9
count
200
150
100
50
0
0
500
1000
1500
2000
hiv_prevalence


Distribution of hiv_prevalence − Median: 146.9
0.4
0.2
0.0
−0.2
−0.4
0
500
1000
1500
2000
hiv_prevalence


Distribution of opioid_rx_rate − Mean: 68.3
count
100
50
0
0
100
200
300
opioid_rx_rate


Distribution of opioid_rx_rate − Median: 62.4
0.4
0.2
0.0
−0.2
−0.4
0
100
200
300
opioid_rx_rate





From the above graphs we can see:

RESPONSE VARIABLE:

1. dist_ssp is right-skewed. This is visible in the graphs and by the fact that the median (75.94) is less than the mean (107.74). This variable could be a candidate for transformation to normalize the distribution.

PREDICTOR VARIABLES

1. hiv_prevalence is right skewed. This is visible in the graphs and by the fact that the median (146.95) is less than the mean (192.89). This variable could be a candidate for transformation to normalize the distribution. The boxplot also highlights that there is an extreme outliers where hiv_prevalence is greater than 2000 that we may want to remove from out analysis.
2. opioid_rx_rate has a fairly normal distribution. This is visibile in the histogram and by the mean and median being similar (68.33 vs 62.4). The boxplot highlights that there are two more extreme outliers where opioid_rx_rate is greater than 200 that we may want to remove from out analysis.
3. opioid_rx_rate has a fairly normal distribution. This is visibile in the histogram and by the mean and median being similar (12.18 vs 11.7). There are a few outliers, but it is not surprising that there would be a couple counties with higher than average rates of uninsured.

Since some of our histogram and boxplots showed potential outliers, let's see how many values for each variable are more than 3 standard deviations from the mean.

Variable	Lower_bound	Count_lower	Upper_bound	Count_upper
dist_ssp	-174.9	0	390.3	5
hiv_prevalence	-400.5	0	786.3	10

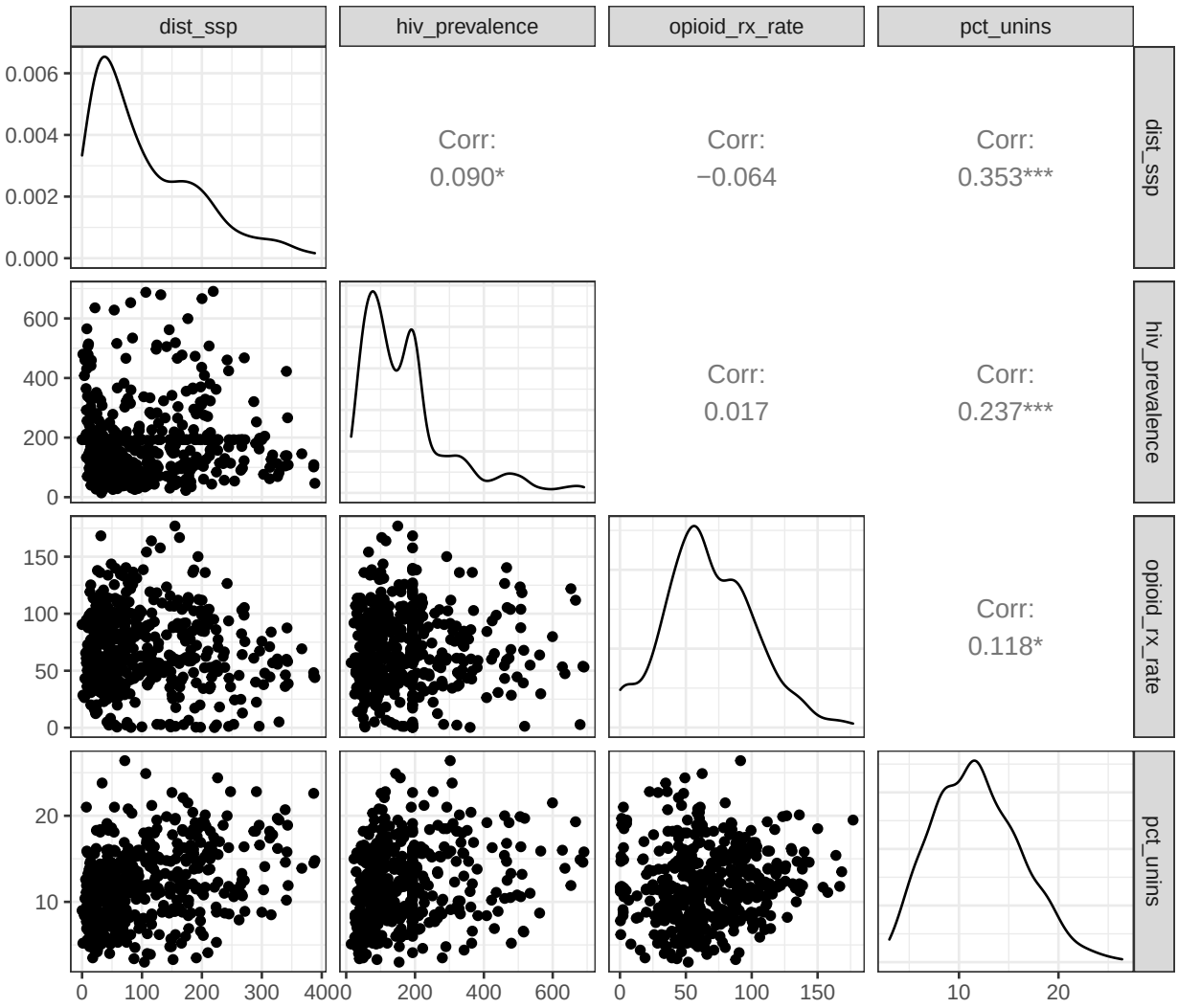
Variable	Lower_bound	Count_lower	Upper_bound	Count_upper
opioid_rx_rate	-42.1	0	178.7	3
pct_unins	-2.8	0	27.2	7

Based on the above table, we can remove any values that exceed the upper bound for each variable. We are removing outliers in this scenario because we want to build a model that can be used to predict the distance to the nearest SSP for the average county. Counties that have higher than average stats should be considered seperately in the future. This leaves us with 478 records in the dataset.

Relationships Between Variables

Now that we have examined our variables independently and removed a few outliers, we want to explore the relationships between the variables.

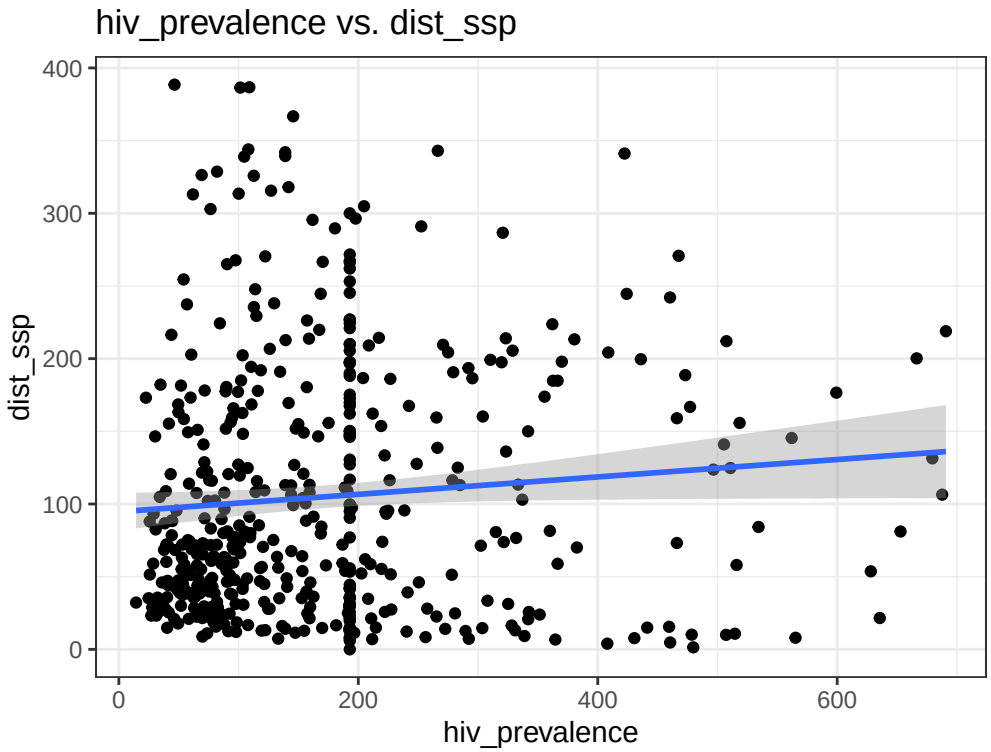
We can see these relationships between the qualitative variables in the following correlation matrix graph output:

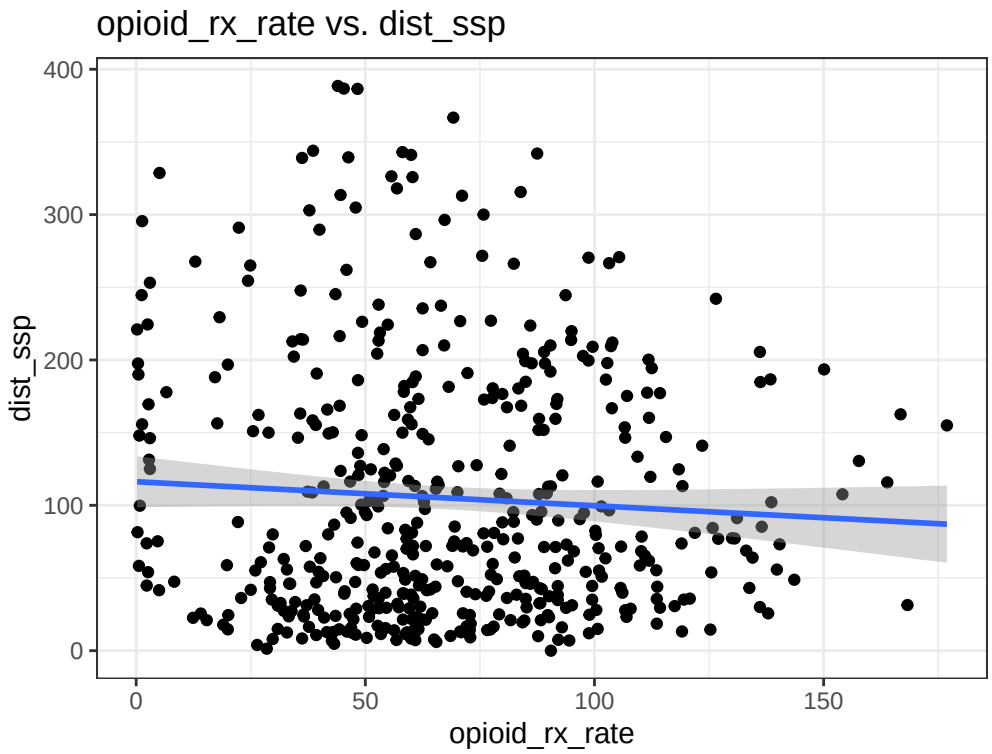


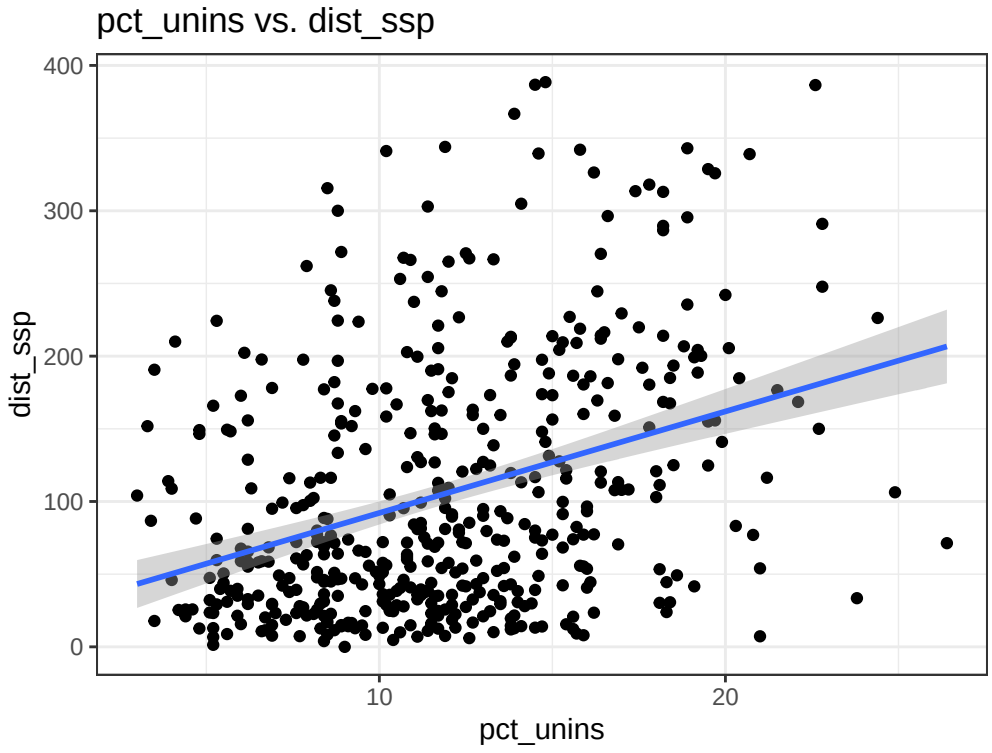
This correlation matrix shows that pct_unins may be a strong predictor for dist_ssp based of the correlation value of 0.353. The next highest correlation values are 0.237 between pct_unins and hiv_prevalence followed by 0.118 between pct_unins and opioid_rx_rate. These values are not too high so we may not need to be concerned with colinearity, but it is good to note that pct_unins does have some correlation with other predictor values. This will be important to keep in mind when checking the variance inflation factors on our models.

Here is a closer look at the scatterplot for each of the predictor variables vs the response variable. A line of best fit has been added to visualize any trends.

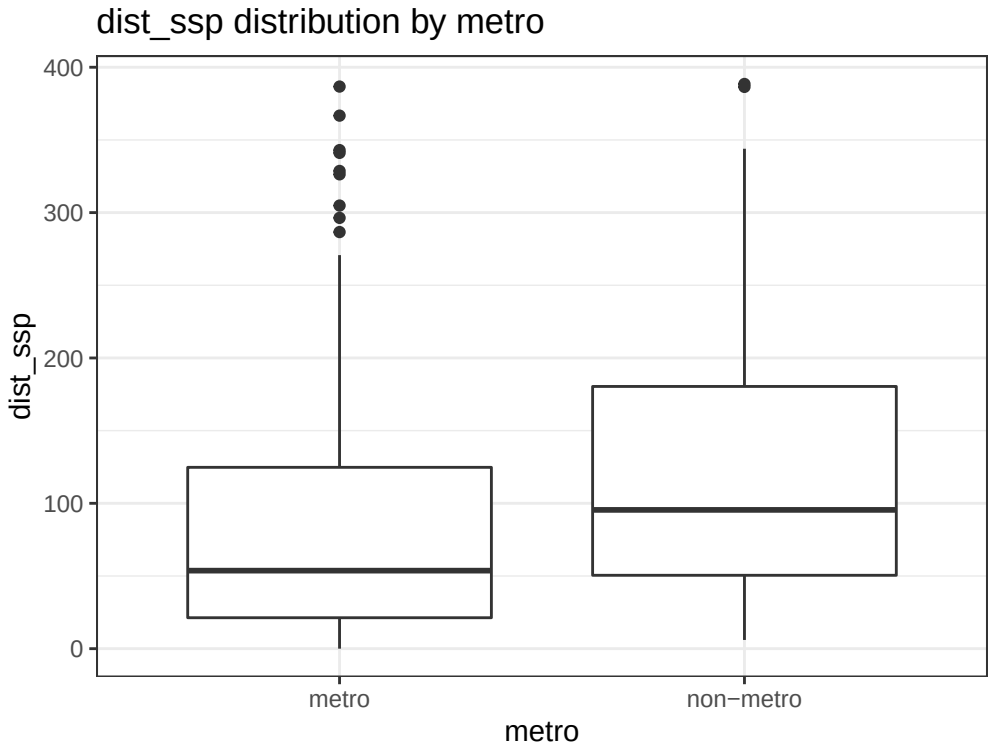
For hiv_prevalence, keep in mind that values of -1 were imputed with the mean, hence the noticeable vertical line of points near hiv_prevalence = 200.

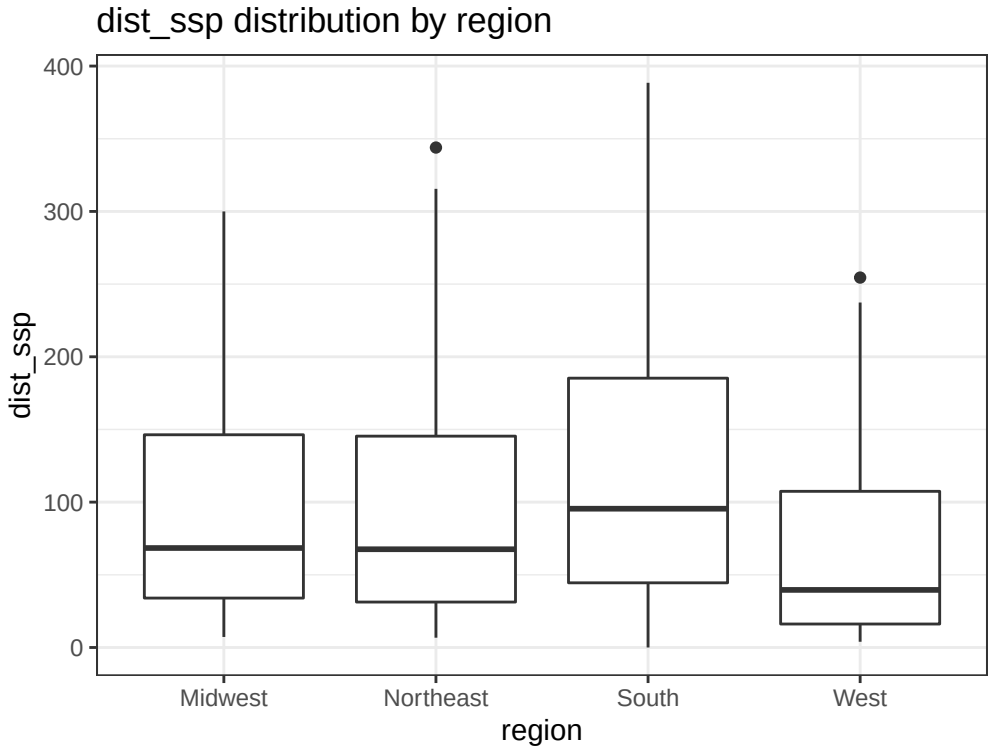






Now, let's look at the qualitative predictors. We can start by creating a box plot for each qualitative predictor to see if there is a difference in dist_ssp between the different groups.



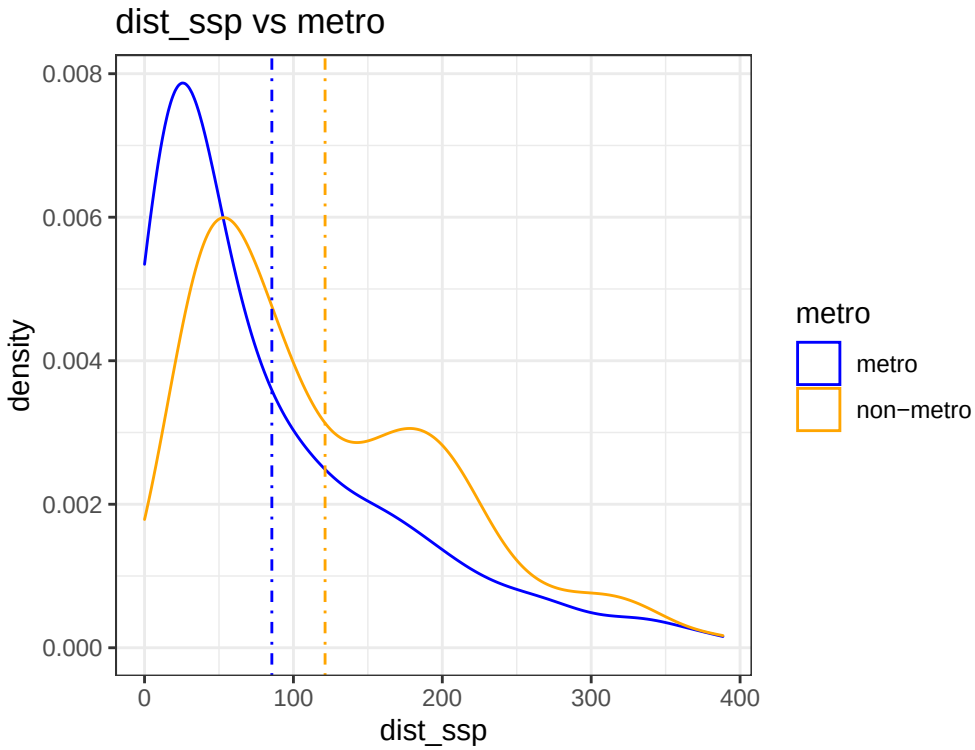


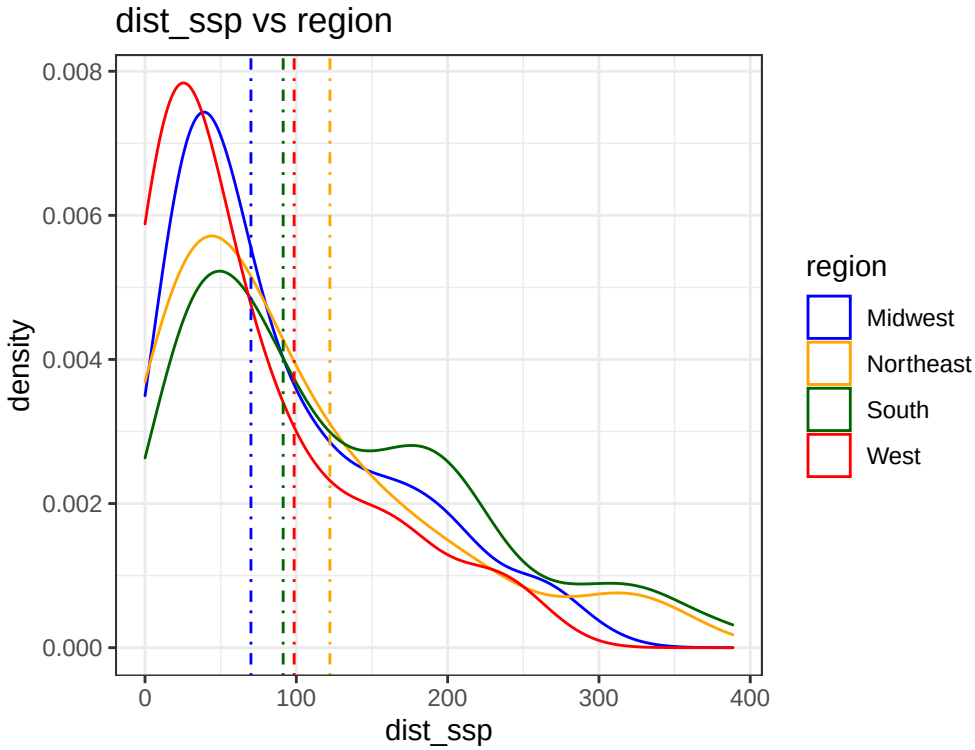
For metro, the box plot shows that counties in metro areas tend to be closer to SSPs than non-metro. This makes sense when you think about the differences between metro and non-metro areas.

For region, the median values for each region are fairly similar with South slightly above Midwest and

Northeast then followed by West.

We also can look at the qualitative variables in a density graph showing the distribution of dist_ssp by the variable.



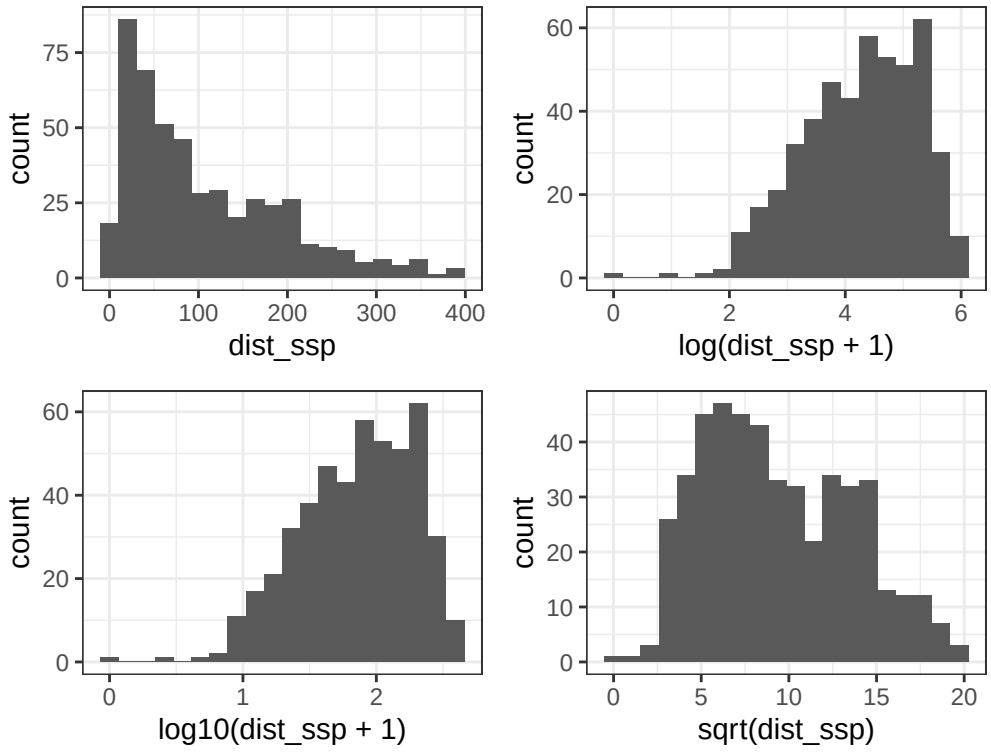


The findings from these density plots verify the observations made from the above box plots.

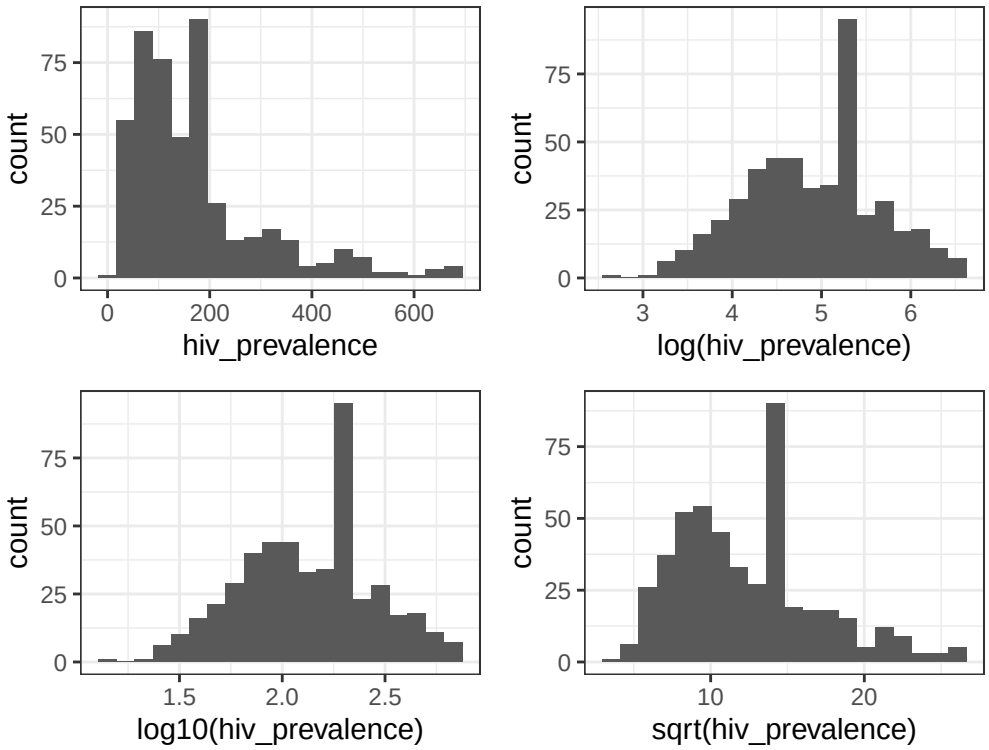
Variable Transformations

Based on the above EDA, dist_ssp and hiv_prevalence both are potential candidates for transformation in order to normalize the distributions. Let's look at the distribution of various transformation options for each variable.

For dist_ssp, when looking a log transformations we will be using (dist_ssp + 1) since some values of dist_ssp are 0, as shown in the summary statistics earlier in this report.

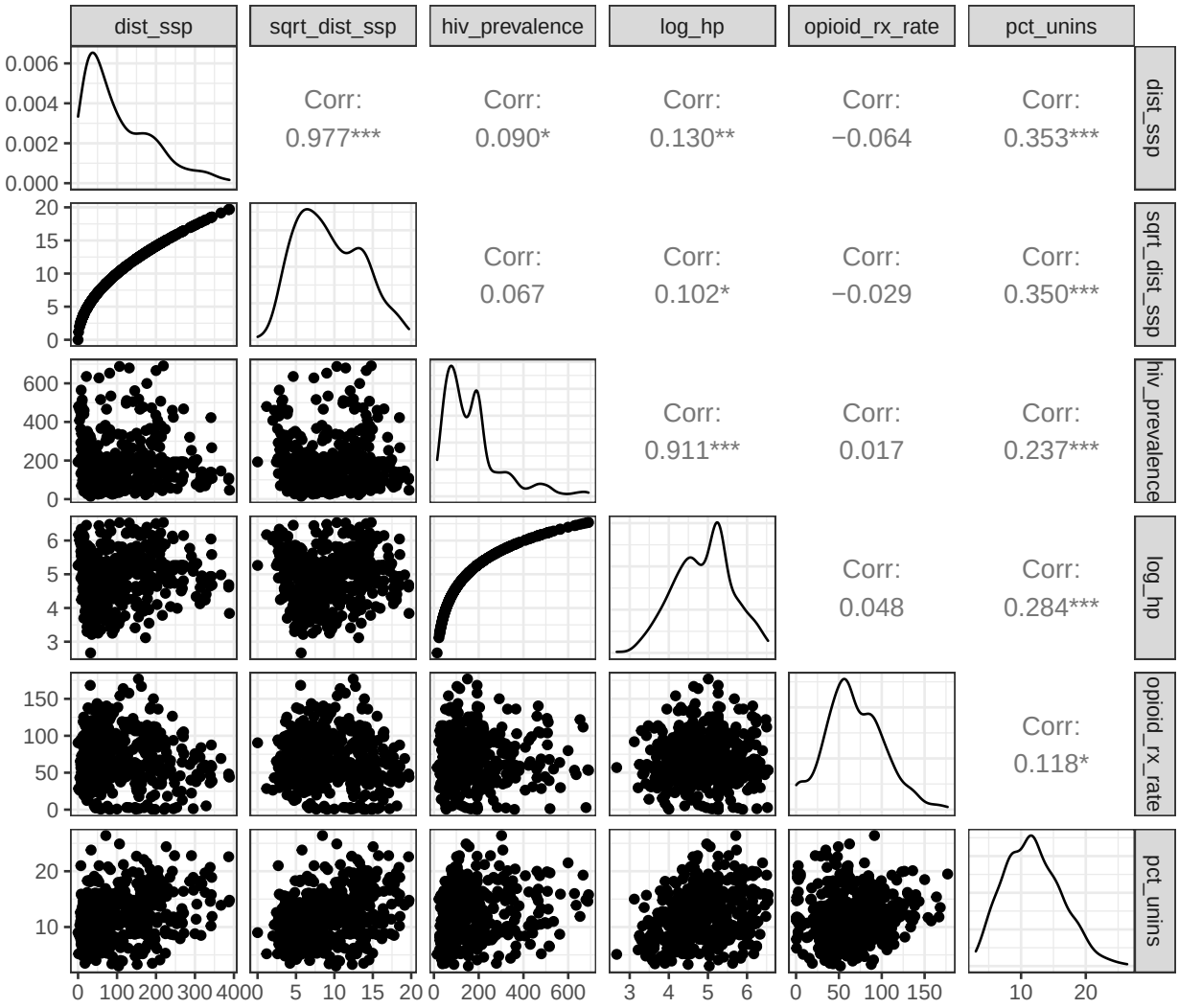


The top left plot is the original distribution of dist_ssp, while the other plots are various transformations. It is clear looking at these that the square root transformation normalizes the data best. We can add a column to our dataset for the sqrt_dist_ssp to try using while creating models.



The top left plot is the original distribution of hiv_prevalence, while the other plots are various transformations. Keep in mind that we imputed the mean value for 70 counties, hence the large spike in these histograms. While these transformations all appear to be more normally distributed than the original distribution, the log(hiv_prevalence) appears the most normally distributed. We can add a column to our dataset for the log(hiv_prevalence) to try using while creating models.

Before we begin our analysis, let's quickly look at how our new variables relate to the existing variables in a correlation matrix.



In this correlation matrix, all variables have a slightly stronger correlation to dist_ssp than sqrt_dist_ssp. This is interesting given that sqrt_dist_ssp has a more normal distribution. The new log_hp variable has a stronger correlation with the other predictor variables than hiv_prevalence.

ANALYSIS - Regression Models

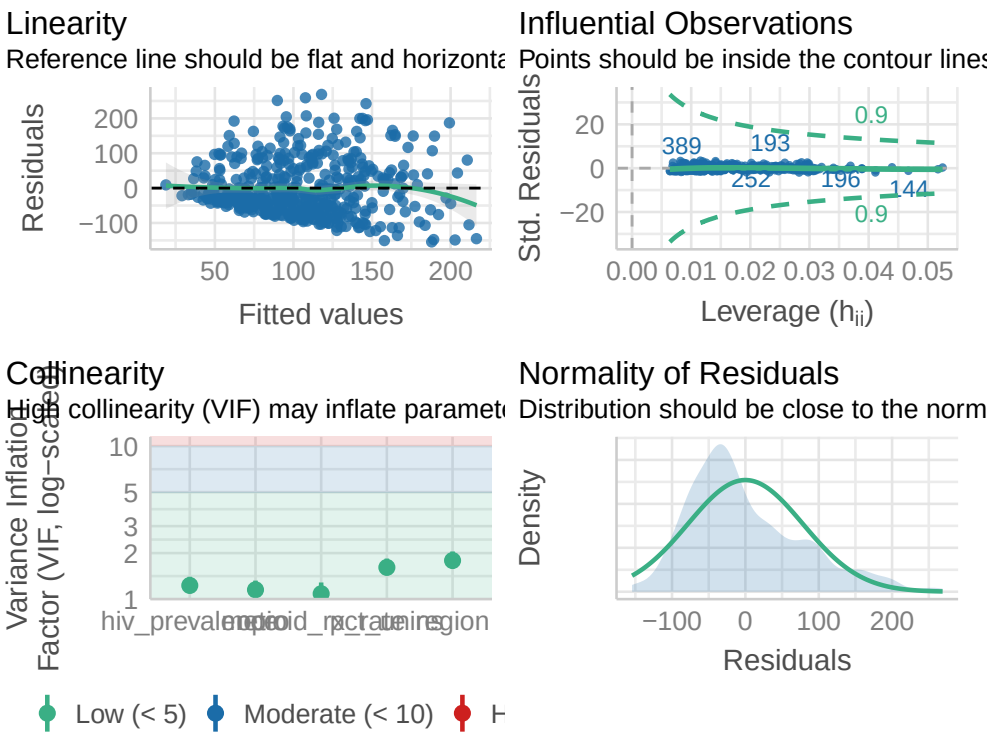
Next, we want to build regression models in order to predict dist_ssp. This report will walk through a few of the key models built and explain the results and what other variations were tried as a result. The full results for all models built will not be shown, but they will be summarized in tables.

Basic Model - All predictors, No Transformations

First, a model was created to predict dist_ssp using the all of original predictor variables (hiv_prevalence, opioid_rx_rate, pct_unins, metro, region). See the results of this model here:

r.squared	adj.r.squared	sigma	statistic	p.value	df	df.residual	nobs
0.184	0.172	79.045	15.182	0	7	470	478

term	estimate	std.error	statistic	p.value
(Intercept)	26.496	13.833	1.915	0.056
hiv_prevalence	0.017	0.031	0.565	0.572
opioid_rx_rate	-0.284	0.112	-2.536	0.012
pct_unins	6.838	1.046	6.535	0.000
metronon-metro	26.964	7.790	3.461	0.001
regionNortheast	18.229	14.272	1.277	0.202
regionSouth	3.218	10.501	0.306	0.759
regionWest	-36.240	13.237	-2.738	0.006



While the model is significant with a p-value of 0, it is currently only able to explain 17% of the variation in dist_ssp.

- Hiv_prevalence is insignificant with a p-value of 0.572. This is a sign to try some models with the transformed log_hp variable to see if it performs better.
- Two categories within the region variable are also insignificant - Northeast (p-value: 0.202) and South (p-value: 0.759). Based on this, it may be worth modifying the region category to combine the insignificant categories into one. A new variable region2 was created with the following categories: Northeast_south, West and Midwest. Region3 was also created combining the significant categories into one group and the insignificant into another. The categories in region3 are: Northeast_south and Midwest_west.

- In the linearity plot, there is evidence of heteroscedasticity based on the cone shape of the fitted values. It also is apparent that the residuals are not normally distributed in the normality of residuals plot. This is a sign to try some models with the target variables as the transformed sqrt_dist_ssp variable to see if it performs better.

Additional Models - Target: Dist_ssp

Based on these learnings, the following models were tried with the target variable of dist_ssp:

Predictors	AdjRSquare	MAE	Insig_Vars
hiv_prevalence+opioid_rx_rate+pct_unins+metro+region	0.172	62.362	hiv_prev, regionNortheast, regionSouth
hiv_prevalence+opioid_rx_rate+pct_unins+metro+region2	0.172	62.340	hiv_prev, region2North_South
hiv_prevalence+opioid_rx_rate+pct_unins+metro+region3	0.162	63.023	hiv_prev
hiv_prevalence+opioid_rx_rate+pct_unins+metro	0.154	63.245	hiv_prev
log_hp+opioid_rx_rate+pct_unins+metro	0.158	62.973	log_hp
opioid_rx_rate+pct_unins+metro	0.153	63.567	
opioid_rx_rate+pct_unins+metro+region3	0.163	63.145	
opioid_rx_rate+pct_unins+metro+region2	0.174	62.486	region2North_South

The only model here that improved upon the adjusted R^2 was when opioid_rx_rate, pct_unins, metro and region2 were used as predictor variables, but that model still had region2North_south as insignificant and had a slightly higher MAE than the original model. The two model versions where all variables were significant both had a lower adjusted r^2 and higher MAE than the original model. These models also had the same cone shaped fitted values vs residual plot as seen in the original model.

Additional Models - Target: sqrt_dist_ssp

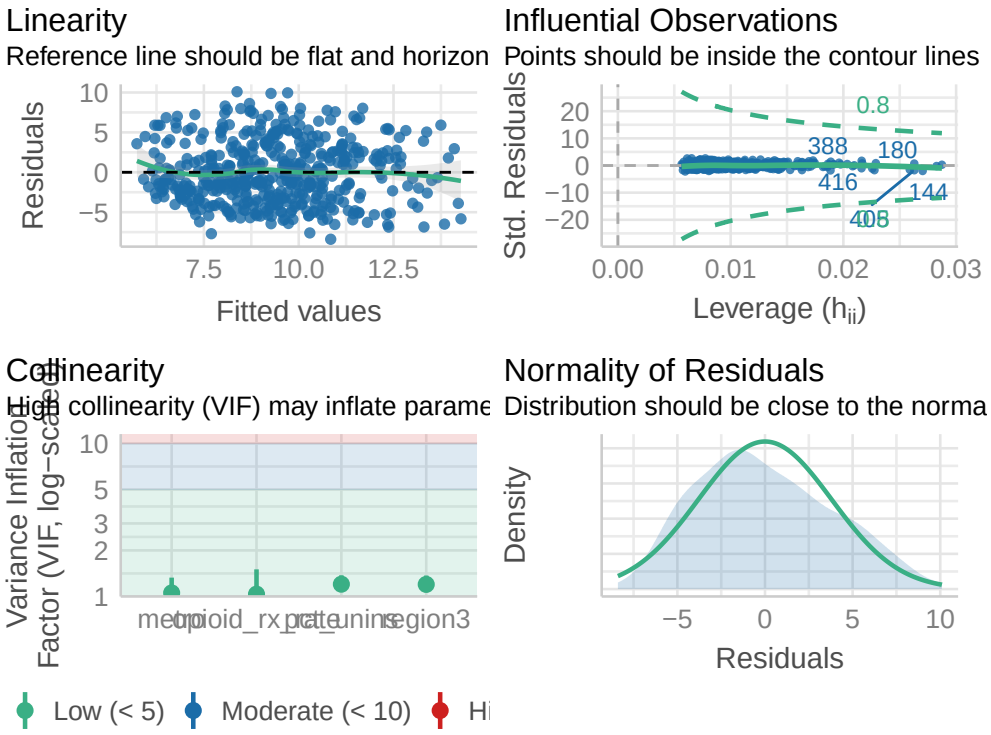
Next, models with similar predictor variable combinations were run, but this time the sqrt_dist_ssp variable was used as the target variable. Here are the results of those models:

Predictors	AdjRSquare	MAE	Insig_Vars
hiv_prevalence+opioid_rx_rate+pct_unins+metro+region	0.187	3.104	hiv_prev, regionNortheast, regionSouth, opioid_rx_rate
hiv_prevalence+opioid_rx_rate+pct_unins+metro+region2	0.187	3.104	hiv_prev, region2North_South
hiv_prevalence+opioid_rx_rate+pct_unins+metro+region3	0.172	3.152	hiv_prev
hiv_prevalence+opioid_rx_rate+pct_unins+metro	0.164	3.168	hiv_prev, opioid_rx_rate
log_hp+opioid_rx_rate+pct_unins+metro	0.166	3.156	log_hp, opioid_rx_rate
log_hp+opioid_rx_rate+pct_unins+metro+region2	0.188	3.095	log_hp,region2North_South
log_hp+opioid_rx_rate+pct_unins+metro+region3	0.173	3.142	log_hp
opioid_rx_rate+pct_unins+metro	0.164	3.182	opioid_rx_rate
opioid_rx_rate+pct_unins+metro+region3	0.173	3.156	

By using sqrt_dist_ssp as the target variable, the models produced tend to have a higher adjusted R^2 than the models run in the previous section. Using the sqrt_dist_ssp variable as the target also helped fix the heteroscedasticity issue. Here is the full output for the model using opiod_rx_rate, pct_unins, metro and region 3 to predict sqrt_dist_ssp.

r.squared	adj.r.squared	sigma	statistic	p.value	df	df.residual	nobs
0.18	0.173	3.823	26.026	0	4	473	478

term	estimate	std.error	statistic	p.value
(Intercept)	5.313	0.596	8.907	0.000
opioid_rx_rate	-0.012	0.005	-2.181	0.030
pct_unins	0.271	0.044	6.189	0.000
metronon-metro	1.884	0.361	5.223	0.000
region3North_south	0.994	0.389	2.556	0.011



When compared to the plots from the first model, this model is an improvement. The model is significant with a p-value of 0, but is still only able to explain 17.35% of the variation in dist_ssp. All of the predictor variables are significant and the linearity and normality of residual plots show that using the sqrt_dist_ssp solves the heteroscedasticity issue, as the distribution of residuals is much closer to normal now.

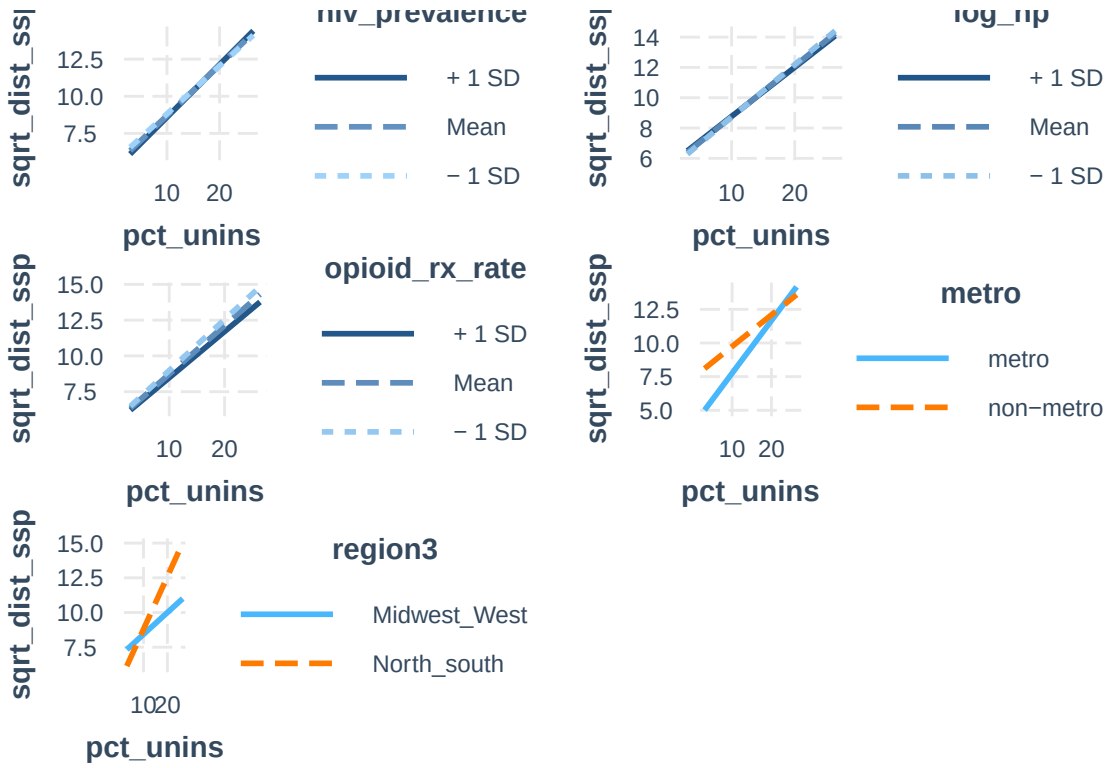
While this model is an improvement, the lower adjusted R^2 leaves room for growth.

Interactions between predictor variables

One potential way to improve this model is to look for interactions between the predictor variables. Since pct_unins had the highest correlation with the other predictor variables, let's check for interactions with pct_unins first.

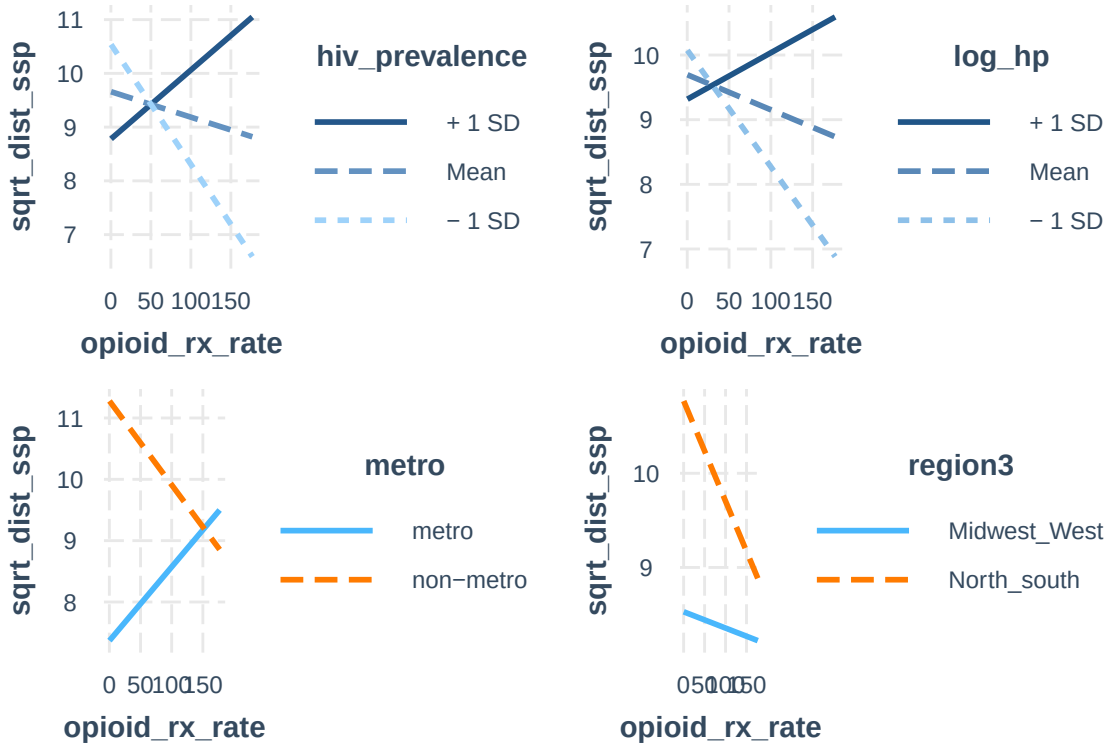
Interaction Plots: pct_unins

In the interaction plots, we are looking for when the lines cross. Lines going the same direction do not indicate an interaction and likely won't improve the model.



Based on the above plots, metro and region3 look like the best interactions to try with pct_unins. Let's also look at the interaction plots for opioid_rx_rate.

Interaction Plots: opioid_rx_rate



Based on the above plots, hiv_prevalence or log_hp look like the best interactions to try with opiod_rx_rate.

Interaction Model Results

Here is an overview of models run using various interaction

Predictors	AdjRSquare	MAE	Insig_Vars
log_hp+opioid_rx_rate+pct_unins+metro+ region3+log_hp*pct_unins	0.171	3.142	log_hp, pct_unins, log_hp*pct_unins
log_hp+opioid_rx_rate+pct_unins+metro+ region3+metro*pct_unins	0.179	3.120	log_hp
log_hp+opioid_rx_rate+pct_unins+metro+ region3+region3*pct_unins	0.180	3.131	log_hp, pct_unins,region3North_south
opioid_rx_rate+pct_unins+metro+ region3+region3*pct_unins	0.180	3.147	region3North_south
opioid_rx_rate+pct_unins+metro+region3+ metro*pct_unins	0.180	3.132	
hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region3+opioid_rx_rate*hiv_prevalence	0.197	3.109	
log_hp+opioid_rx_rate+pct_unins+metro+ region3+opioid_rx_rate*log_hp	0.190	3.105	
hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region3+opioid_rx_rate*hiv_prevalence*+metro*pct_unins	0.204	3.083	

The last three models in the above table are the best three models. Each model had significant p-values and the highest adjusted r^2s out of all models run so far. We will choose the model that uses an interaction

variable between opioid_rx_rate and hiv_prevalence along with and interaction variable between metro and pct_unins.

Final Regression Model - Predicting sqrt_dist_ssp using interaction variables between opioid_rx_rate and hiv_prevalence along with matero and pct_unins

While a summary of this model has already been shown in the above table, here is the models full output including the model significance, variable coefficients and checking the model assumptions.

r.squared	adj.r.squared	sigma	statistic	p.value	df	df.residual	nobs
0.216	0.204	3.752	18.447	0	7	470	478

term	estimate	std.error	statistic	p.value
(Intercept)	6.00976	0.88760	6.77082	0.00000
hiv_prevalence	-0.01025	0.00298	-3.44566	0.00062
opioid_rx_rate	-0.04066	0.00880	-4.62119	0.00000
pct_unins	0.36919	0.06285	5.87373	0.00000
metronon-metro	4.13447	1.00722	4.10482	0.00005
region3North_south	1.06998	0.39724	2.69352	0.00732
hiv_prevalence:opioid_rx_rate	0.00016	0.00004	4.00183	0.00007
pct_unins:metronon-metro	-0.18210	0.08060	-2.25919	0.02433

This model has a similar overall F test that shows that the model does help to predict distance with p-value of 0 and its Adjusted R-square tells that it can explain 20.38% of the variation in sqrt_dist_ssp.

Using the above coefficients, the equation for this model is:

sqrt_dist_ssp = 6.0097584 + -0.0102525(hiv_prevalence) + -0.0406647(opioid_rx_rate) + 0.3691861(pct_unins) + 4.134465(metronon-metro) + 1.0699818(region3North_south) + 1.5697619×10^{-4}(hiv_prevalence*opioid_rx_rate) + -0.1820985(pct_unins*metronon-metro)

Since it makes more sense to be predicting dist_ssp, rather than sqrt_dist_ssp, both sides of this equation can be squared to have the final equation of:

dist_ssp = (6.0097584 + -0.0102525(hiv_prevalence) + -0.0406647(opioid_rx_rate) + 0.3691861(pct_unins) + 4.134465(metronon-metro) + 1.0699818(region3North_south) + 1.5697619×10^{-4}(hiv_prevalence*opioid_rx_rate) + -0.1820985(pct_unins*metronon-metro))^2

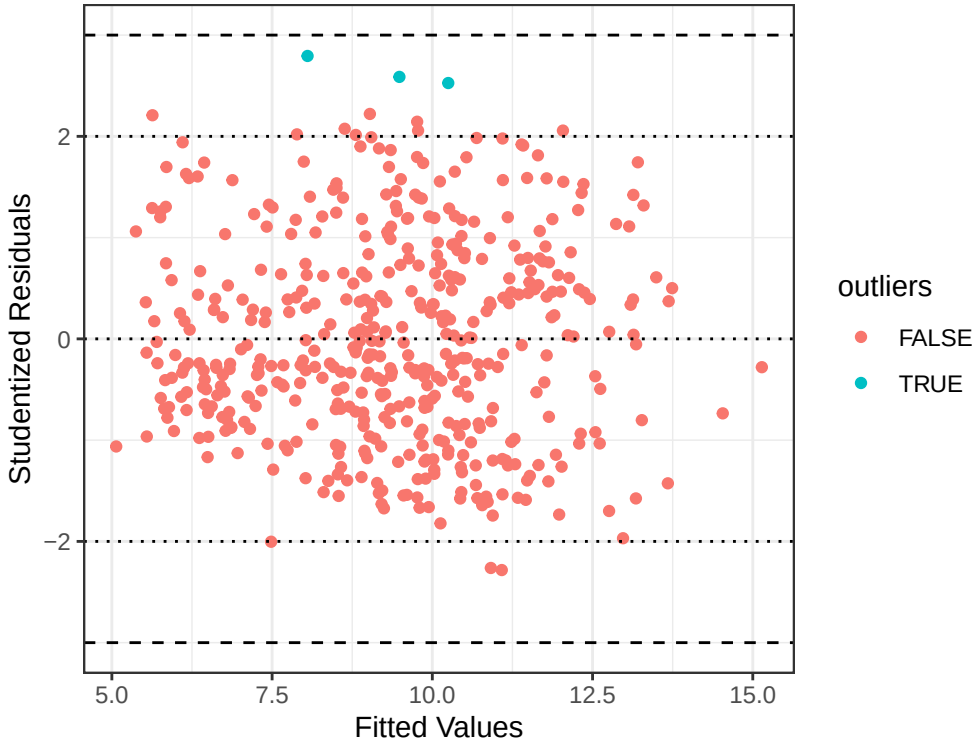
Check of Assumptions

Since this model is the best so far, we will perform a full check of assumptions.

Outliers with Studentized Residuals A value is defined as an outlier if the studentized residual is greater than 2.5. Based on this cut off, there are 3 values that are considered outliers. In this table, it is clear that the 3 outliers are scenarios where the model underestimated how close the nearest ssp is. It is also noticeable that all three of these outliers fall within metro counties. This may show that these three counties happen to be further from an SSP than one would expect for a metro area.

sqrt_dist_ssp	hiv_prevalence	opioid_rx_rate	pct_uninsu	metro	region3	.fitted	.resid	.std.resid	outliers
18.469	422.4	60.0	10.2	metro	North_south	8.053	10.416	2.794	TRUE
19.665	109.0	45.3	14.5	metro	North_south	10.248	9.417	2.527	TRUE
19.150	145.6	69.2	13.9	metro	North_south	9.486	9.663	2.587	TRUE

While we know there are only three points that are considered outliers, let's plot our residuals and include reference lines at +/- 2 and 3 studentized residuals to help visuals how far the model estimates are from the actual point. In this plot, outliers are shown in blue and all other points are shown in red. This helps us see that while those three points are being overestimated and considered outliers, they are barely outside the 2.5 studentized deviation cut off.



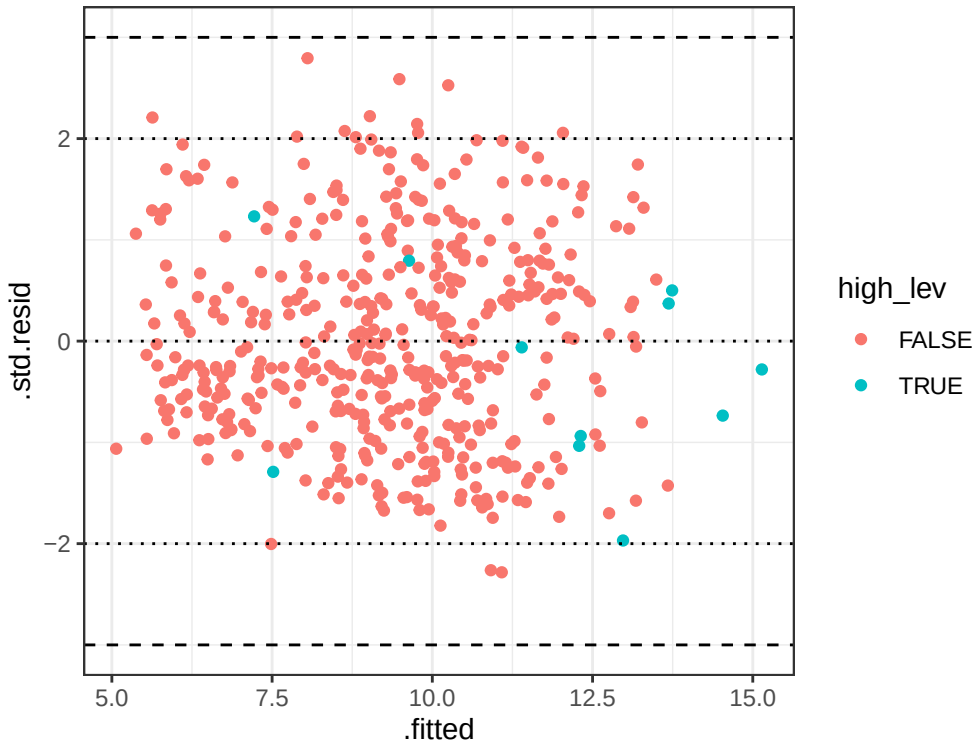
Influential Observations and Leverage Some observations may have more influence in the model than others. By checking for observations that have a leverage value that is 3 times greater than the average level value is one way to see how many values have a particularly strong influence on the model.The average leverage value (.hat) for observations in this data set is: 0.0167, therefore observations with a leverage value (.hat) greater than 0.0502 are considered influential.

This model has 11 influential points. While not included in this report, a next step could be to create a model without these points and see the impact that it has on model results and the coefficients.

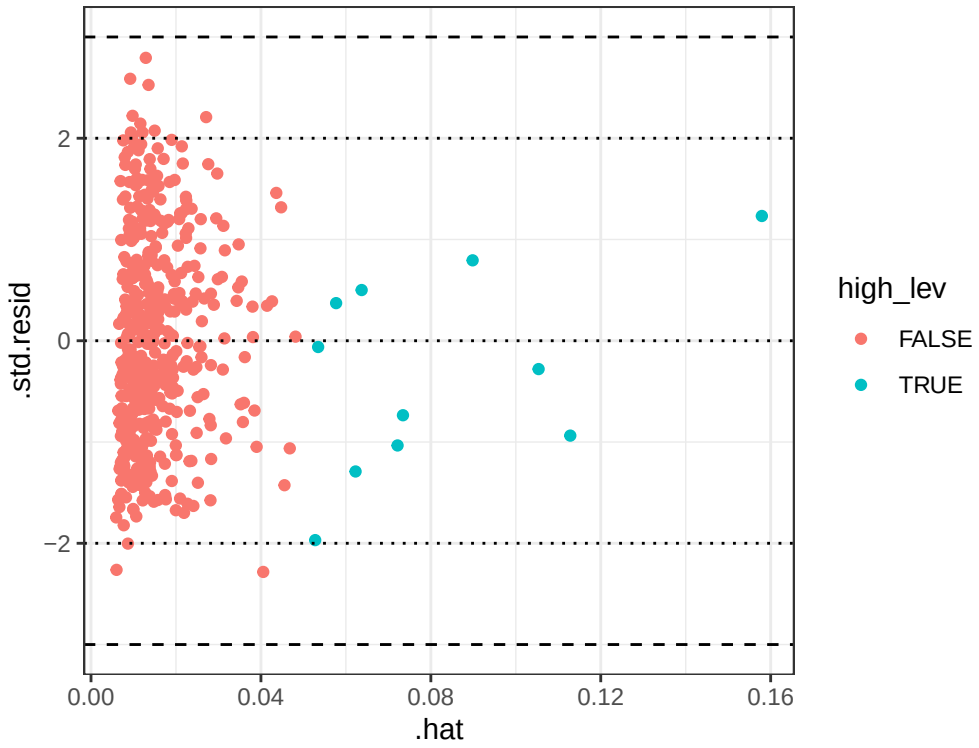
sqrt_dist_ssp	hiv_prevalence	opioid_rx_rate	pct_uninsu	metro	region3	.fitted	.resid	.hat	.std.resid	high_lev
15.560	460.4	126.5	20.0	metro	North_south	13.742	1.818	0.064	0.501	TRUE
15.042	157.2	49.3	24.4	metro	North_south	13.688	1.354	0.058	0.372	TRUE
11.170	510.8	118.4	13.2	metro	North_south	11.395	-0.225	0.053	-0.062	TRUE

sqrt_dist_ssp	hiv_prevalence	opioid_rx_rate	pct_uninsu	metro	region3	.fitted	.resid	.hat	.std.resid	high_lev
9.005	652.9	121.9	11.9	metro	North_south	12.316	-3.311	0.113	-0.937	TRUE
8.554	466.1	140.4	14.7	metro	North_south	12.291	-3.737	0.072	-1.034	TRUE
5.785	307.9	34.7	23.8	metro	North_south	12.976	-7.190	0.053	-1.969	TRUE
2.827	565.2	29.8	15.9	metro	Midwest_West	7.517	-4.691	0.062	-1.291	TRUE
14.149	666.4	111.8	19.3	non-metro	North_south	15.142	-0.992	0.105	-0.280	TRUE
12.480	518.5	1.3	19.7	non-metro	North_south	9.637	2.843	0.090	0.794	TRUE
11.874	505.4	123.5	19.9	non-metro	North_south	14.532	-2.657	0.073	-0.736	TRUE
11.463	679.5	2.8	14.9	non-metro	North_south	7.220	4.243	0.158	1.232	TRUE

Let's look at a plot examining where the high leverage values fall on the fitted vs the studentized residuals. Here it is clear that the high leverage values tend to be towards the upper end of predicted values, with a few exceptions.

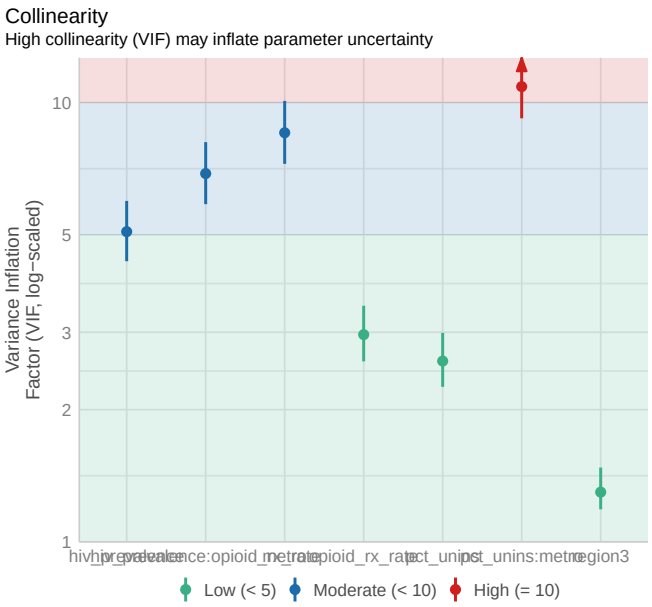


We also can look at how much leverage these points have in comparison to non-high leverage observations. This graph makes it very apparent that these observations have significantly more leverage than others. A next step for improve this model should be to try creating a model without these points to see how much it changes.



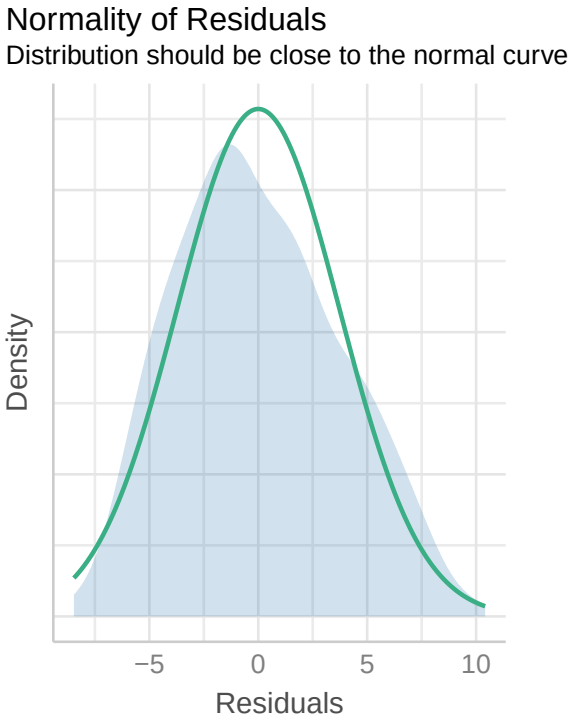
Assumption 1: Little to no multicollinearity between predictors It is imporant to check that the predictor variables do not have multicollinearity between each other. Since our final model contains interaction terms, we do expect to see multicollinearity for this model with the interaction variables.

Multicollinearity can be checked for by looking at the Variance Inflation Factor (VIF). The plot below shows the VIF values for each variable. We can see we have many that have a medium to high VIF, but these variables are all used within the interaction variables so that is to be expected. In all the model built prior to introducing interaction variables, the variables all how low VIF values (as seen in the prior models), therefore there is no concern about multicollinearity for this model.

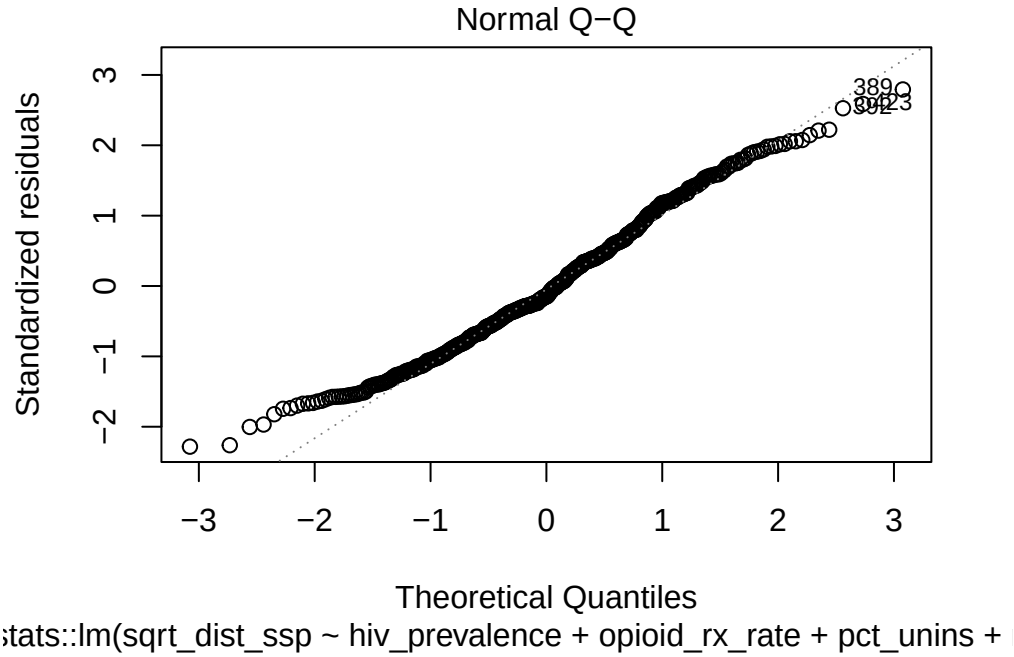


Assumption 2: Errors (Residuals) are normally distributed

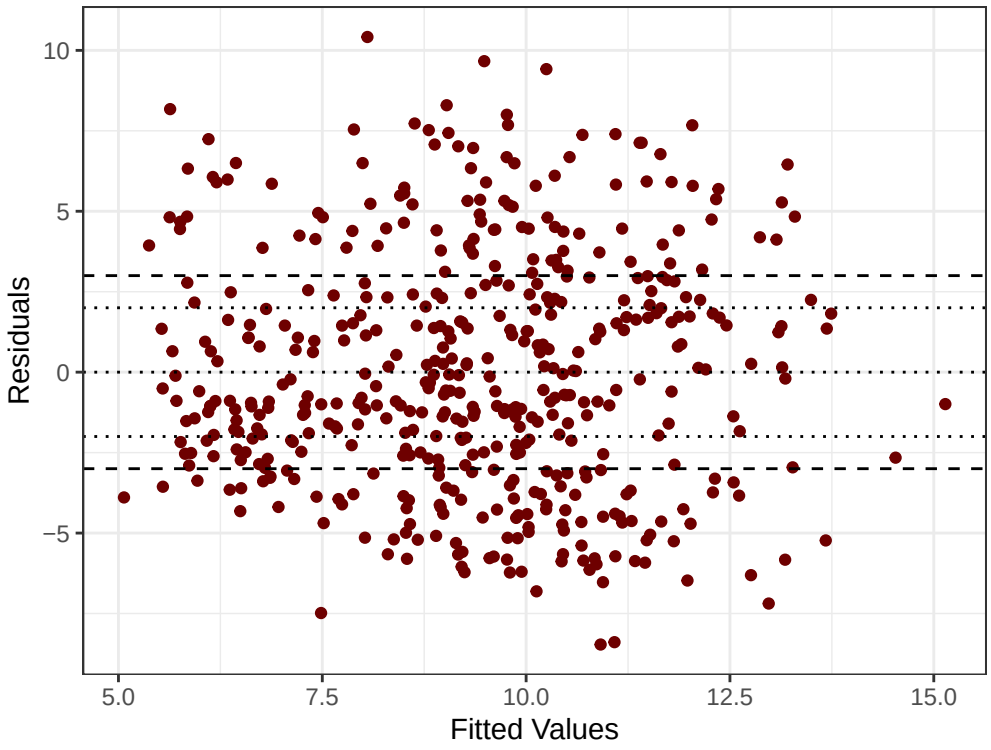
Histogram of Residuals The following graph is showing the looking at the distribution of the residuals, with the green line representing the normal distribution. It is clear that the residuals are very close to normally distributed, therefore there are no concerns.

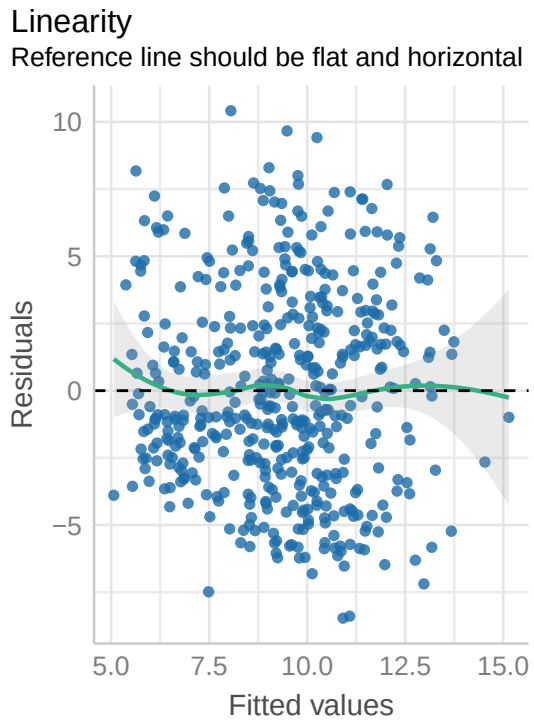


Normal QQ Plot The QQ Plot is also looking at the distribution of the residuals, but this visual can help understand where the data is not following the normal distribution. The QQ plot shows that while the distribution is fairly normal, the violations are worst where the model is over-predicting on low distances and under-predicting on high distances.



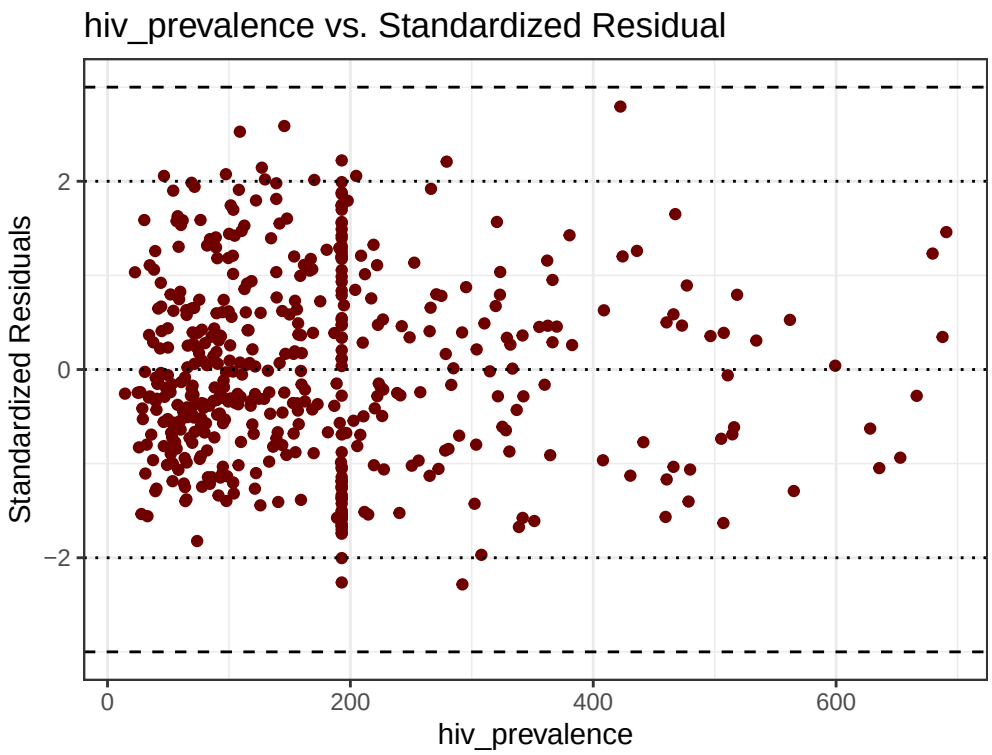
Assumption 3: Homoscedasticity of errors (or, equal variance around the 0 across fitted y values). Here we have a plot of the residuals vs the fitted values. The goal is to have there be equal variance around residual = 0 across all fitted values. This means that regardless of the fitted value, the model is over/under estimating equally. In this plot, we can see that even though we have some rather large residuals, we do have pretty close to homoscedasticity with a few large residuals for fitted values that are overestimated the distance to the nearest ssp.

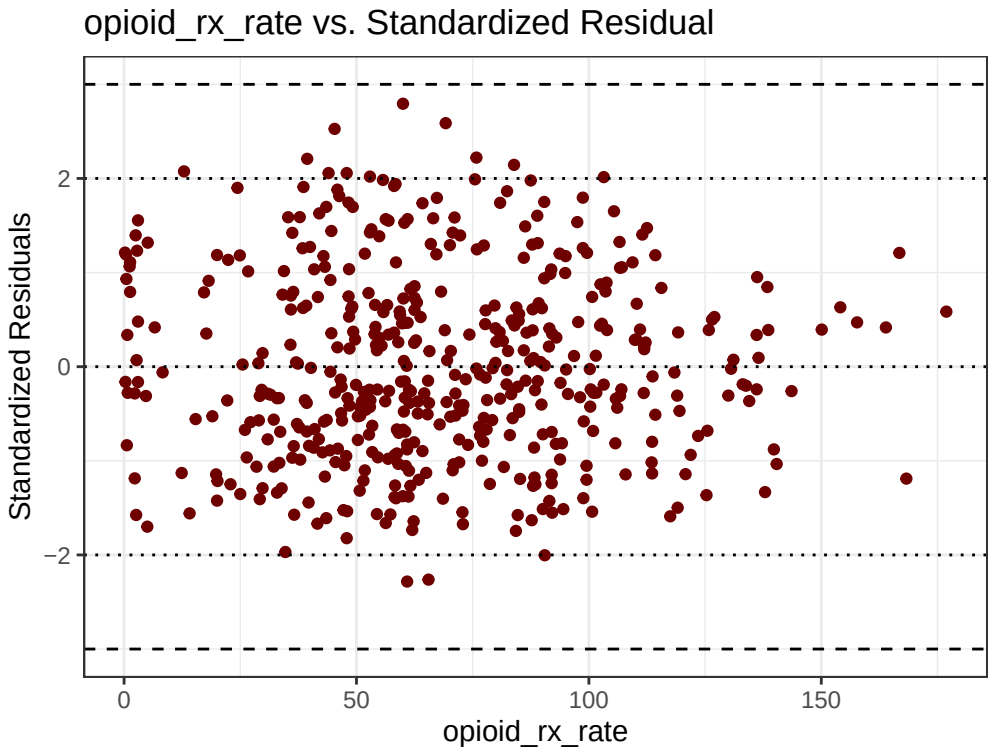


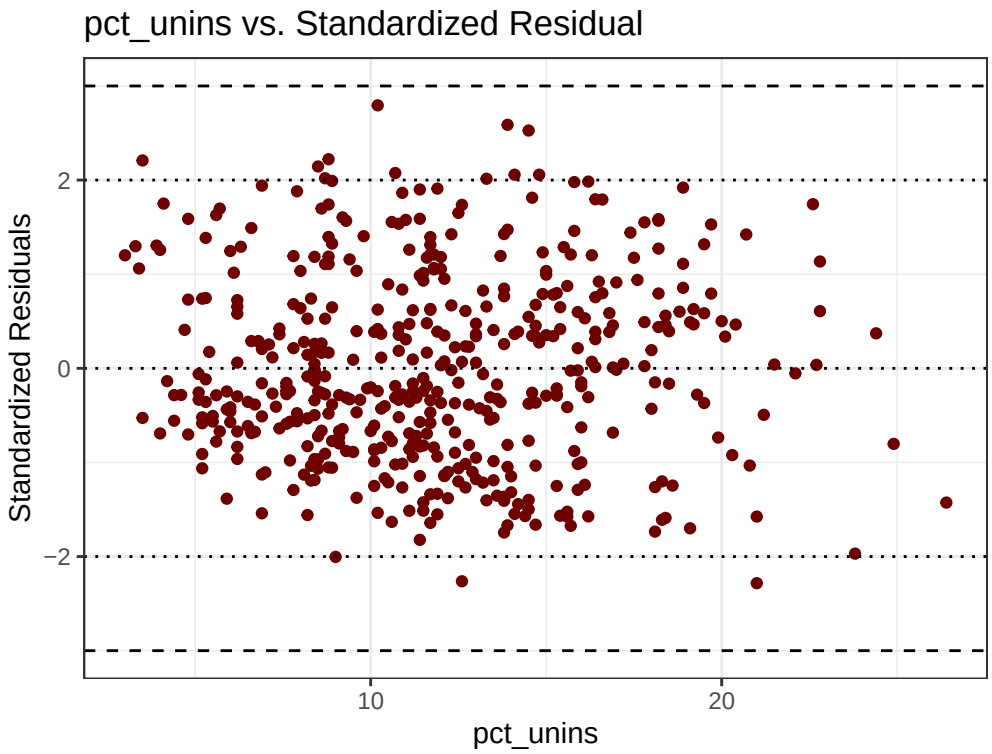


Residuals vs. Continuous Predictors The residuals can also be looked at against each predictor to see if there are certain variables that have any patterns. Here we can see that each variable equally contributes to

there being slightly more over predicting, despite a few observations they have pretty equal variance around 0.







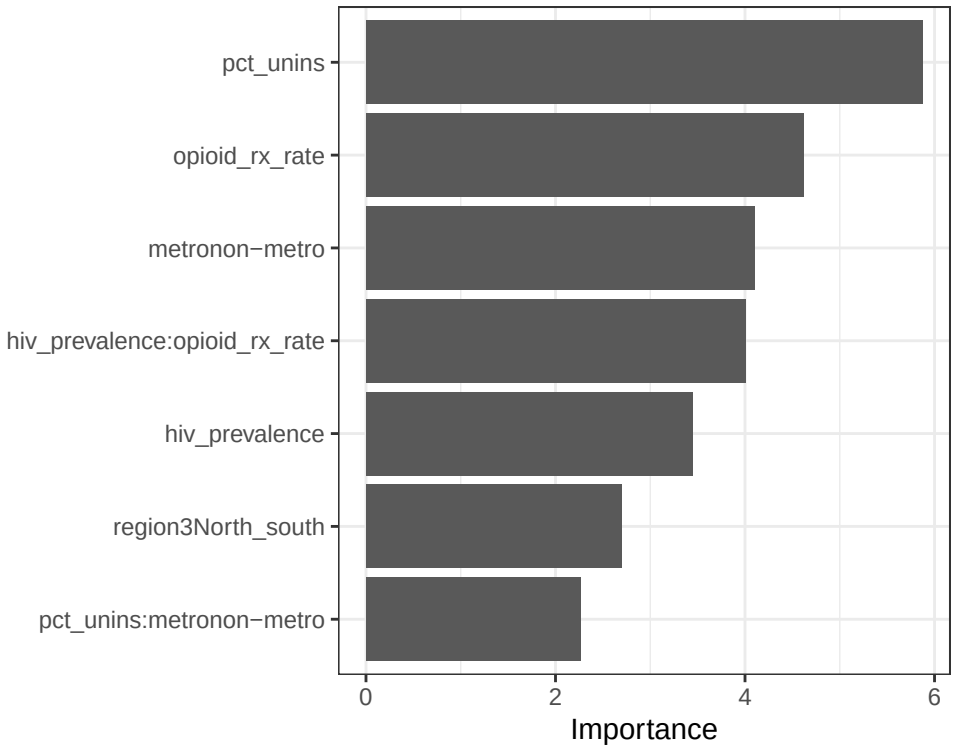
Assumption 4: Independence of the observations Here we are using the Durbin Watsin test to check if the residuals have any autocorrelation/are related based on the order of the data. We want our observations to be independent, or not have any autocorrelation.

Ideally, the Durbin Watson test statistic would be between 1.5-2.5, but for this model the test statistic is 1.07 meaning there is a positive autocorrelation and the observations are not independent.

```
    Durbin-Watson test

data:  lm_final$fit
DW = 1.0719, p-value < 2.2e-16
alternative hypothesis: true autocorrelation is greater than 0
```

Most Important Predictors Now that the regression assumptions have all been tested, we also can look at which predictors are considered the most important within our model, just for a deeper understanding.



Conclusions

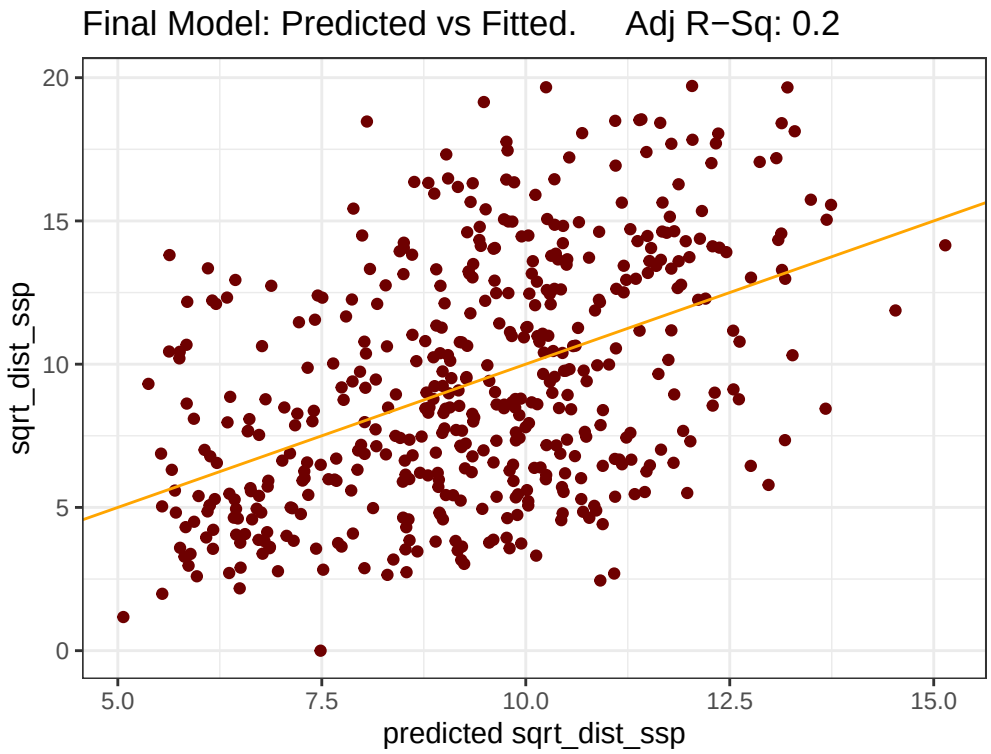
Based on our diagnostics, the regression model predicting sqrt_dist_ssp using both interaction variables is the strongest model. This model still has room for improvement as seen by the adjusted R^2 of 20.38%. The most important predictors in the model are pct_unins, opioid_rx_rate and metronon-metro. The final version of the regression equation can be seen here:

dist_ssp = (6.0097584 + -0.0102525(hiv_prevalence) + -0.0406647(opioid_rx_rate) + 0.3691861(pct_unins) + 4.134465(metronon-metro) + 1.0699818(region3North_south) + 1.5697619×10^{-4}(hiv_prevalence*opioid_rx_rate) + -0.1820985(pct_unins*metronon-metro))^2

and the 95% confidence intervals for the coefficients are give here:

	2.5 %	97.5 %
(Intercept)	4.2656	7.7539
hiv_prevalence	-0.0161	-0.0044
opioid_rx_rate	-0.0580	-0.0234
pct_unins	0.2457	0.4927
metronon-metro	2.1552	6.1137
region3North_south	0.2894	1.8506
hiv_prevalence:opioid_rx_rate	0.0001	0.0002
pct_unins:metronon-metro	-0.3405	-0.0237

We can see from the plot comparing the actual distance to the predicted distance, that our values have a wide spread from line which is not surprising given our adjusted R^2 of 20.38% and the spread of our residual plots when checking the regression assumptions.



Overall, this is not a very strong model that definitely still has room for improvement, despite the many variations tried. Additional data cleaning may be necessary, along with comparing the model to a version created without the observations that had high leverage. A complete table combining the tables of models attempted can be found in Appendix B, while the code for this analysis can be found in Appendix B.

APPENDIX A: All Models

Predictors	Target	AdjRSquared	MAE	Insig_Vars
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region	dist_ssp	0.172	62.362	hiv_prev, regionNortheast, regionSouth
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region2	dist_ssp	0.172	62.340	hiv_prev, region2North_South
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3	dist_ssp	0.162	63.023	hiv_prev
hiv_prevalence+opioid_rx_rate+ pct_unins+metro	dist_ssp	0.154	63.245	hiv_prev
log_hp+opioid_rx_rate+pct_unins+ metro	dist_ssp	0.158	62.973	log_hp
opioid_rx_rate+pct_unins+metro	dist_ssp	0.153	63.567	
opioid_rx_rate+pct_unins+metro+ region3	dist_ssp	0.163	63.145	
opioid_rx_rate+pct_unins+metro+ region2	dist_ssp	0.174	62.486	region2North_South
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region	sqrt_dist_ssp	0.187	3.104	hiv_prev, regionNortheast, regionSouth, opioid_rx_rate
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region2	sqrt_dist_ssp	0.187	3.104	hiv_prev, region2North_South

Predictors	Target	AdjRSquare	MAE	Insig_Vars
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3	sqrt_dist_ssp	0.172	3.152	hiv_prev
hiv_prevalence+opioid_rx_rate+ pct_unins+metro	sqrt_dist_ssp	0.164	3.168	hiv_prev, opioid_rx_rate
log_hp+opioid_rx_rate+pct_unins+metro	sqrt_dist_ssp	0.166	3.156	log_hp, opioid_rx_rate
log_hp+opioid_rx_rate+pct_unins+ metro+region2	sqrt_dist_ssp	0.188	3.095	log_hp,region2North_South
log_hp+opioid_rx_rate+pct_unins+ metro+region3	sqrt_dist_ssp	0.173	3.142	log_hp
opioid_rx_rate+pct_unins+metro	sqrt_dist_ssp	0.164	3.182	opioid_rx_rate
opioid_rx_rate+pct_unins+metro+region3	sqrt_dist_ssp	0.173	3.156	
log_hp+opioid_rx_rate+pct_unins+ metro+region3+log_hp*pct_unins	sqrt_dist_ssp	0.171	3.142	log_hp, pct_unins, log_hp*pct_unins
log_hp+opioid_rx_rate+pct_unins+ metro+region3+metro*pct_unins	sqrt_dist_ssp	0.179	3.120	log_hp
log_hp+opioid_rx_rate+pct_unins+ metro+region3+region3*pct_unins	sqrt_dist_ssp	0.180	3.131	log_hp, pct_unins,region3North_south
opioid_rx_rate+pct_unins+metro+ region3+region3*pct_unins	sqrt_dist_ssp	0.180	3.147	region3North_south
opioid_rx_rate+pct_unins+metro+ region3+metro*pct_unins	sqrt_dist_ssp	0.180	3.132	
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3+ opioid_rx_rate*hiv_prevalence	sqrt_dist_ssp	0.197	3.109	
log_hp+opioid_rx_rate+pct_unins+ metro+region3+ opioid_rx_rate*log_hp	sqrt_dist_ssp	0.190	3.105	
hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3+ opioid_rx_rate*hiv_prevalence+metro*pct_unins	sqrt_dist_ssp	0.204	3.083	

APPENDIX B: The Code

```
knitr::opts_chunk$set(fig.align="center",
                      fig.height=4,
                      fig.width=5.25,
                      warning = FALSE,
                      message = FALSE,
                      comment = NA,
                      echo=FALSE)
# Load Packages
library(tidymodels)
  #library(parsnip) #v1.0.0 linear_reg(), set_engine(), set_mode(), fit(), predict()
  #library(yardstick) #v1.0.0 metrics()
  #library(dplyr) #v1.0.9 %>%, select(), select_if(), filter(), mutate(), group_by(),
      #summarize(), tibble()
  #library(ggplot2) #v3.3.6 ggplot()
  #library(broom) #v0.8.0 for tidy(), augment(), glance()
library(knitr) #v1.39 allows you to create Appendix with all_labels()
library(readr) #v2.1.2 read_csv()
```

```
library(GGally) #v2.1.2 for ggpairs
library(performance) #v0.9.1 check_model
library(see) #v0.7.1  for check_model plots from performance
library(patchwork) #v1.1.1for check_model plots from performance
library(skimr) #v2.1.4 for for skim()
library(lmtest) #v0.9-40 bptest() (constant variance), dwtest() (indep)
library(interactions) #v1.1.5 interaction_plot
library(ggcorrplot) #v0.1.3 ggcorrplot()
theme_set(theme_bw()) #sets default ggplot output style
library(janitor) #clean_names()
library(gridExtra) #grid.arrange()
library(vip) #vip

# code needed earlier to pull in values for executive summary. This code is repeated again below in ord
#Load Data
df_ops <- read_csv("../../data/dist_ssp_amfar_ch9.csv")
df_Region <- read_csv("../../data/StateRegion.csv")


#Add region to the other data
df_ops <- df_ops %>%
  rename(State=STATEABBREVIATION)

df_ops <- merge(x=df_ops,y=df_Region,by="State")

#Rename variables
df_ops <- clean_names(df_ops)

df_ops <- df_ops %>%
  rename(hiv_prevalence=hi_vprevalence,
         pct_unins=pctunins) %>%
  mutate(region = replace(region, region == 'W','West')) %>%
  mutate(region = replace(region, region == 'S','South')) %>%
  mutate(region = replace(region, region == 'N','Northeast')) %>%
  mutate(region = replace(region, region == 'M','Midwest')) %>%
  select(-county,-state)

# replace hiv_prev=-1 with mean
df_ops$hiv_prevalence[df_ops$hiv_prevalence==-1] <- round(mean(df_ops$hiv_prevalence[df_ops$hiv_prevale

# removing outliers
df_ops <- df_ops %>%
  filter(dist_ssp <= 390.3) %>%
  filter(hiv_prevalence <= 786.3) %>%
  filter(opioid_rx_rate <= 178.7) %>%
  filter(pct_unins <= 27.2)

# transform
df_ops <- df_ops %>%
  mutate(sqrt_dist_ssp = sqrt(dist_ssp))

# need to create region 3 to create executive summary
df_ops <- df_ops %>%
```

```
  mutate(region3 = case_when(region == 'Northeast' ~ "North_south",
                             region == 'South' ~ "North_south",
                             region == 'West' ~ "Midwest_West",
                             region == 'Midwest' ~ "Midwest_West"))

# lm_model
lm_model <- linear_reg() %>%
            set_engine("lm") %>%
            set_mode("regression")

# create model for exec summary

lm_final <- lm_model %>%
            fit(sqrt_dist_ssp ~ hiv_prevalence+opioid_rx_rate+pct_unins+metro+region3+opioid_rx_rate*hi

glance_final <- glance(summary(lm_final$fit))


# model output
knitr::kable(glance_final, digits=3)

knitr::kable(tidy(lm_final$fit), digits=5)
#Load Data
df_ops <- read_csv("../../data/dist_ssp_amfar_ch9.csv")
df_Region <- read_csv("../../data/StateRegion.csv")
#Add region to the other data
df_ops <- df_ops %>%
  rename(State=STATEABBREVIATION)

df_ops <- merge(x=df_ops,y=df_Region,by="State")

#Rename variables
df_ops <- clean_names(df_ops)

df_ops <- df_ops %>%
  rename(hiv_prevalence=hi_vprevalence,
         pct_unins=pctunins) %>%
  mutate(region = replace(region, region == 'W','West')) %>%
  mutate(region = replace(region, region == 'S','South')) %>%
  mutate(region = replace(region, region == 'N','Northeast')) %>%
  mutate(region = replace(region, region == 'M','Midwest')) %>%
  select(-county,-state)

#View summary statistics
skim_no_p25p75 <-skim_with(numeric = sfl(p25 = NULL,
                                         p75 =NULL))
skim_no_p25p75(df_ops)

# table of hiv_prev values less than or equal to 1 for interpreting when value = -1
hiv_prev0 <- df_ops %>%
  filter(hiv_prevalence<=0) %>%
  group_by(hiv_prevalence) %>%
  summarize(count = n())
```

```
knitr::kable(hiv_prev0,digits=3)

# replace hiv_prev=-1 with mean
df_ops$hiv_prevalence[df_ops$hiv_prevalence==-1] <- round(mean(df_ops$hiv_prevalence[df_ops$hiv_prevale

summary(df_ops$hiv_prevalence)

#View qualitative predictors
  # make list of qual predictors
ops_qual <- df_ops %>%
select_if(is.character) %>%
  names()

#function to make bar graph of predictors
for(n in ops_qual){
  p <- ggplot(df_ops, aes(x=get(n))) +
  geom_bar() +
    geom_text(stat='count', aes(label=..count..), vjust=-.5, cex=2)+
    xlab(n)+
    ggtitle(paste('Count of counties per', n))
  print(p)
}

# View quantitative predictors distributions
 # make list of quant predictors
ops_quan <- df_ops %>%
select_if(is.numeric) %>%
  names()

#function to make hist/boxplot of predictors
for(n in ops_quan){
  currmean <- round(as.numeric(df_ops %>%
                summarize(mean=mean(get(n)))),1)
  currmedian <- round(as.numeric(df_ops %>%
                summarize(median=median(get(n)))),1)
  p <- df_ops %>%
          ggplot(aes(x=get(n))) +
            geom_histogram(bins=20) +
            geom_vline(aes(xintercept=currmean)) +
            xlab(n) +
            ggtitle(paste("Distribution of", n," - Mean:", currmean))
  p2 <- df_ops %>%
          ggplot(aes(x=get(n))) +
            geom_boxplot() +
            xlab(n) +
            ggtitle(paste("Distribution of", n," - Median:", currmedian))
  grid.arrange(p,p2,ncol=1, nrow=2)
}

# create empty list to store values in
Lower <- c()
Count_lower <- c()
Upper <- c()
```

```
Count_upper <- c()

# function to find mean and standard deviation, along with how many values fall outside these lower/upp
for(n in ops_quan){
  mean = round(as.numeric(df_ops %>%
                 summarize(mean=mean(get(n)))),1)
  sd = round(as.numeric(df_ops %>%
                 summarize(sd=sd(get(n)))),1)
  lower = mean-(3*sd)
  upper = mean+(3*sd)
  count_lower = as.numeric(df_ops %>%
    filter(get(n)<lower) %>%
    summarize(count=n()))
  count_upper = as.numeric(df_ops %>%
    filter(get(n)>upper) %>%
    summarize(count=n()))
  Lower <- c(Lower,lower)
  Count_lower <- c(Count_lower,count_lower)
  Upper <- c(Upper,upper)
  Count_upper <- c(Count_upper,count_upper)
}

# create table of values
outliers <- tibble(Variable=ops_quan,
                   Lower_bound=Lower,
                   Count_lower=Count_lower,
                   Upper_bound=Upper,
                   Count_upper=Count_upper)

knitr::kable(outliers)

# removing outliers
df_ops <- df_ops %>%
  filter(dist_ssp <= 390.3) %>%
  filter(hiv_prevalence <= 786.3) %>%
  filter(opioid_rx_rate <= 178.7) %>%
  filter(pct_unins <= 27.2)

#View correlation matrix plot for numeric variables

ggpairs(data = df_ops[,c("dist_ssp","hiv_prevalence", "opioid_rx_rate","pct_unins")])

# scatter plot of quan predictor variables vs dist_spp

# quantitative predictors
ops_quan_pred <- c("hiv_prevalence", "opioid_rx_rate","pct_unins")

# function to make scatterplots of predictor variables vs dist_ssp
for(n in ops_quan_pred){
  p <- ggplot(df_ops, aes(x = get(n), y=dist_ssp))
  p <- p + geom_point() +
    geom_smooth(method = "lm") +
    xlab(n) +
```

```
    ggtitle(paste(n,"vs. dist_ssp"))
print(p)
}

# box plot of qual predictor variables for dist_spp

for(n in ops_qual){
  p <- ggplot(df_ops, aes(x = get(n), y=dist_ssp)) +
        geom_boxplot() +
        xlab(n) +
        ggtitle(paste('dist_ssp distribution by', n))
print(p)
}

#View metro density plot
metro <- df_ops %>% filter(metro=='metro') %>% summarize(mean(dist_ssp)) %>% pull()
nonmetro <- df_ops %>% filter(metro=='non-metro') %>% summarize(mean(dist_ssp)) %>% pull()

ggplot(df_ops,aes(x=dist_ssp,col=metro)) +
  geom_density() +
  scale_color_manual(values= c('Blue','Orange','Dark Green','Red')) +
  geom_vline(xintercept=metro, linetype = 4, col = 'Blue') +
  geom_vline(xintercept=nonmetro, linetype = 4, col = 'Orange') +
  ggtitle("dist_ssp vs metro")


#View region density plot
west_region <- df_ops %>% filter(region=='West') %>% summarize(mean(dist_ssp)) %>% pull()
south_region <- df_ops %>% filter(region=='South') %>% summarize(mean(dist_ssp)) %>% pull()
northeast_region <- df_ops %>% filter(region=='Northeast') %>% summarize(mean(dist_ssp)) %>% pull()
midwest_region <- df_ops %>% filter(region=='Midwest') %>% summarize(mean(dist_ssp)) %>% pull()

ggplot(df_ops,aes(x=dist_ssp,col=region)) +
  geom_density() +
  scale_color_manual(values= c('Blue','Orange','Dark Green','Red')) +
  geom_vline(xintercept=west_region, linetype = 4, col = 'Blue') +
  geom_vline(xintercept=south_region, linetype = 4, col = 'Orange') +
  geom_vline(xintercept=midwest_region, linetype = 4, col = 'Dark Green') +
  geom_vline(xintercept=northeast_region, linetype = 4, col = 'Red') +
  ggtitle("dist_ssp vs region")


# transform dist_ssp
p1 <- ggplot(df_ops,aes(x=dist_ssp)) + geom_histogram(bins=20)
p2 <-ggplot(df_ops,aes(x=log(dist_ssp+1))) + geom_histogram(bins=20)
p3 <- ggplot(df_ops,aes(x=log10(dist_ssp+1))) + geom_histogram(bins=20)
p4 <- ggplot(df_ops,aes(x=sqrt(dist_ssp))) + geom_histogram(bins=20)

grid.arrange(p1,p2, p3, p4, ncol=2, nrow=2)
# transform hiv_prevalence
p1 <- ggplot(df_ops,aes(x=hiv_prevalence)) + geom_histogram(bins=20)
p2 <- ggplot(df_ops,aes(x=log(hiv_prevalence))) + geom_histogram(bins=20)
p3 <-ggplot(df_ops,aes(x=log10(hiv_prevalence))) + geom_histogram(bins=20)
```

```
p4 <- ggplot(df_ops,aes(x=sqrt(hiv_prevalence))) + geom_histogram(bins=20)
grid.arrange(p1,p2, p3, p4, ncol=2, nrow=2)
# create transformed variables
df_ops <- df_ops %>%
  mutate(log_hp = log(hiv_prevalence)) %>%
  mutate(sqrt_dist_ssp = sqrt(dist_ssp))

# look at correlation of new variables
ggpairs(data = df_ops[,c("dist_ssp","sqrt_dist_ssp", "hiv_prevalence", "log_hp", "opioid_rx_rate","pct_

# lm_model
lm_model <- linear_reg() %>%
            set_engine("lm") %>%
            set_mode("regression")


# Model 1: Predict Dist_ssp, all original predictors

lm_full <- lm_model %>%
            fit(dist_ssp ~ hiv_prevalence+opioid_rx_rate+pct_unins+metro+region, data = df_ops)

glance_full <- glance(summary(lm_full$fit))

knitr::kable(glance_full, digits=3)

knitr::kable(tidy(lm_full$fit), digits=3)

lm_full %>%
      check_model(check=c('vif','linearity','normality','outliers'))

# create region variables based on model 1

# region2
df_ops <- df_ops %>%
  mutate(region2 = case_when(region == 'Northeast' ~ "North_south",
                             region == 'South' ~ "North_south",
                             region == 'Midwest' ~ "Midwest",
                             region == 'West' ~ "West")
         )

# region3
df_ops <- df_ops %>%
  mutate(region3 = case_when(region == 'Northeast' ~ "North_south",
                             region == 'South' ~ "North_south",
                             region == 'West' ~ "Midwest_West",
                             region == 'Midwest' ~ "Midwest_West"))

# dist_ssp models
predictors = c('hiv_prevalence+opioid_rx_rate+pct_unins+metro+ region',
               'hiv_prevalence+opioid_rx_rate+pct_unins+metro+ region2',
               'hiv_prevalence+opioid_rx_rate+pct_unins+metro+ region3',
               'hiv_prevalence+opioid_rx_rate+pct_unins+metro',
               'log_hp+opioid_rx_rate+pct_unins+metro',
```

```
               'opioid_rx_rate+pct_unins+metro',
               'opioid_rx_rate+pct_unins+metro+region3',
               'opioid_rx_rate+pct_unins+metro+region2'
               )
insig_vars = c('hiv_prev, regionNortheast, regionSouth',
               'hiv_prev, region2North_South',
               'hiv_prev',
               'hiv_prev',
               'log_hp',
               '',
               '',
               'region2North_South')

out_metrics <- tibble(Predictors = predictors)
AR2s <- c()
MAEs <- c()
for (p in predictors){
  reg_curr_fit <- lm_model %>%
    fit(formula(paste('dist_ssp ~', p)), data = df_ops)
  out <- select(tidy(reg_curr_fit$fit), term, p.value)
  #print(knitr::kable(out, digits=3)) #this row can be included if you want to see the variable p-value
  AR2s <- c(AR2s, round(pull(select(glance(reg_curr_fit$fit),adj.r.squared)),3))
  curr_MAE <- pull((reg_curr_fit$fit %>%
              augment(df_ops) %>%
              metrics(truth = dist_ssp, estimate = .fitted))[3,3])
  MAEs <- c(MAEs, round(curr_MAE,3))
}

out_metrics <- out_metrics %>%
          mutate(AdjRSquare = AR2s,
                 MAE = MAEs,
                 Insig_Vars = insig_vars)

knitr::kable(out_metrics,digits=3)

# sqrt_dist_ssp models

predictors2 = c('hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region',
                'hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region2',
                'hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region3',
                'hiv_prevalence+opioid_rx_rate+pct_unins+ metro',
                'log_hp+opioid_rx_rate+pct_unins+metro',
                'log_hp+opioid_rx_rate+pct_unins+metro+ region2',
                'log_hp+opioid_rx_rate+pct_unins+metro+ region3',
                'opioid_rx_rate+pct_unins+metro',
                'opioid_rx_rate+pct_unins+metro+region3')
insig_vars2 = c('hiv_prev, regionNortheast, regionSouth, opioid_rx_rate',
               'hiv_prev, region2North_South',
               'hiv_prev',
               'hiv_prev, opioid_rx_rate',
               'log_hp, opioid_rx_rate',
               'log_hp,region2North_South',
               'log_hp',
```

```
              'opioid_rx_rate',
              '')

out_metrics2 <- tibble(Predictors = predictors2)
AR2s2 <- c()
MAEs2 <- c()
for (p in predictors2){
  reg_curr_fit <- lm_model %>%
    fit(formula(paste('sqrt_dist_ssp ~', p)), data = df_ops)
  out <- select(tidy(reg_curr_fit$fit), term, p.value)
  # print(knitr::kable(out, digits=3))
  AR2s2 <- c(AR2s2, round(pull(select(glance(reg_curr_fit$fit),adj.r.squared)),3))
  curr_MAE <- pull((reg_curr_fit$fit %>%
              augment(df_ops) %>%
              metrics(truth = sqrt_dist_ssp, estimate = .fitted))[3,3])
  MAEs2 <- c(MAEs2, round(curr_MAE,3))
}

out_metrics2 <- out_metrics2 %>%
          mutate(AdjRSquare = AR2s2,
                 MAE = MAEs2,
                 Insig_Vars = insig_vars2)

knitr::kable(out_metrics2,digits=3)

#Model 13: Predict sqrt_dist_ssp, use region3, no hiv_prev
lm_13 <- lm_model %>%
            fit(sqrt_dist_ssp ~ opioid_rx_rate+pct_unins+metro+region3, data = df_ops)

glance_13 <- glance(summary(lm_13$fit))

knitr::kable(glance_13, digits=3)

knitr::kable(tidy(lm_13$fit), digits=3)

lm_13 %>%
      check_model(check=c('vif','linearity','normality','outliers'))
#interaction plots with pct_unins
#hiv_prevalence
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ pct_unins * hiv_prevalence, data=df_ops)
p1 <- interact_plot(reg_prun_int$fit,pred=pct_unins,modx=hiv_prevalence)

#log_hp
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ pct_unins * log_hp, data=df_ops)
p2 <- interact_plot(reg_prun_int$fit,pred=pct_unins,modx=log_hp)

#opioid_rx_rate
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ pct_unins * opioid_rx_rate, data=df_ops)
p3 <- interact_plot(reg_prun_int$fit,pred=pct_unins,modx=opioid_rx_rate)
```

```
#metro
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ pct_unins * metro, data=df_ops)
p4 <- interact_plot(reg_prun_int$fit,pred=pct_unins,modx=metro)

#region3
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ pct_unins * region3, data=df_ops)
p5 <- interact_plot(reg_prun_int$fit,pred=pct_unins,modx=region3)

grid.arrange(p1,p2,p3,p4,p5,ncol=2, nrow=3)
#interactions with opioid_rx_rate
#hiv_prevalence
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ opioid_rx_rate * hiv_prevalence, data=df_ops)
p1 <- interact_plot(reg_prun_int$fit,pred=opioid_rx_rate,modx=hiv_prevalence)

#log_hp
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ opioid_rx_rate * log_hp, data=df_ops)
p2 <- interact_plot(reg_prun_int$fit,pred=opioid_rx_rate,modx=log_hp)

#metro
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ opioid_rx_rate * metro, data=df_ops)
p3 <- interact_plot(reg_prun_int$fit,pred=opioid_rx_rate,modx=metro)

#region3
reg_prun_int <- lm_model %>%
  fit(sqrt_dist_ssp ~ opioid_rx_rate * region3, data=df_ops)
p4 <- interact_plot(reg_prun_int$fit,pred=opioid_rx_rate,modx=region3)

grid.arrange(p1,p2,p3,p4,ncol=2, nrow=2)
# sqrt_dist_ssp models

predictors2 = c('log_hp+opioid_rx_rate+pct_unins+metro+ region3+log_hp*pct_unins',
                'log_hp+opioid_rx_rate+pct_unins+metro+ region3+metro*pct_unins',
                'log_hp+opioid_rx_rate+pct_unins+metro+ region3+region3*pct_unins',
                'opioid_rx_rate+pct_unins+metro+ region3+region3*pct_unins',
                'opioid_rx_rate+pct_unins+metro+region3+ metro*pct_unins',
                'hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region3+opioid_rx_rate*hiv_prevalence',
                'log_hp+opioid_rx_rate+pct_unins+metro+ region3+opioid_rx_rate*log_hp',
                'hiv_prevalence+opioid_rx_rate+pct_unins+ metro+region3+opioid_rx_rate*hiv_prevalence+m

insig_vars2 = c('log_hp, pct_unins, log_hp*pct_unins',
               'log_hp',
               'log_hp, pct_unins,region3North_south',
               'region3North_south',
               '',
               '',
               '',
               '')
```

```
out_metrics2 <- tibble(Predictors = predictors2)
AR2s2 <- c()
MAEs2 <- c()
for (p in predictors2){
  reg_curr_fit <- lm_model %>%
    fit(formula(paste('sqrt_dist_ssp ~', p)), data = df_ops)
  out <- select(tidy(reg_curr_fit$fit), term, p.value)
  #print(knitr::kable(out, digits=3))
  AR2s2 <- c(AR2s2, round(pull(select(glance(reg_curr_fit$fit),adj.r.squared)),3))
  curr_MAE <- pull((reg_curr_fit$fit %>%
              augment(df_ops) %>%
              metrics(truth = sqrt_dist_ssp, estimate = .fitted))[3,3])
  MAEs2 <- c(MAEs2, round(curr_MAE,3))
}

out_metrics2 <- out_metrics2 %>%
          mutate(AdjRSquare = AR2s2,
                 MAE = MAEs2,
                 Insig_Vars = insig_vars2)

knitr::kable(out_metrics2,digits=3)

#Model 19: Predict sqrt_dist_ssp, create interaction between metro and log_hp

lm_final <- lm_model %>%
            fit(sqrt_dist_ssp ~ hiv_prevalence+opioid_rx_rate+pct_unins+metro+region3+opioid_rx_rate*hiv

glance_final <- glance(summary(lm_final$fit))

knitr::kable(glance_final, digits=3)

knitr::kable(tidy(lm_final$fit), digits=5)

# lm_final %>%
#      check_model(check=c('vif','linearity','normality','outliers'))


#make dataframe only using variables in final model
df_ops2 <- df_ops %>%
  select(sqrt_dist_ssp, hiv_prevalence, opioid_rx_rate, pct_unins, metro, region3)


#Outliers and studentized residuals
knitr::kable(lm_final$fit %>%
               augment(data=df_ops2) %>%
               mutate(outliers = if_else(abs(.std.resid) > 2.5, TRUE, FALSE)) %>%
               filter(outliers==TRUE) %>%
               select(-.hat, -.sigma, -.cooksd),
             digits=3)

#Outliers plot
lm_final$fit %>%
  augment(data=df_ops2) %>%
```

```
  mutate(outliers = if_else(abs(.std.resid) > 2.5, TRUE, FALSE)) %>%
  ggplot(aes(x=.fitted, y=.std.resid, col=outliers)) +
    geom_point() +
    scale_x_continuous("Fitted Values") +
    scale_y_continuous("Studentized Residuals") +
    geom_hline(yintercept=0, linetype = 3) +
    geom_hline(yintercept=3, linetype = 2) +
    geom_hline(yintercept=2, linetype = 3) +
    geom_hline(yintercept=-2, linetype = 3) +
    geom_hline(yintercept=-3, linetype = 2)

#Influential observations and leverage
knitr::kable(lm_final$fit %>%
                augment(data=df_ops2) %>%
                mutate(high_lev = if_else(.hat > 3*mean(.hat), TRUE, FALSE)) %>%
                filter(high_lev==TRUE) %>%
                select(-.sigma, -.cooksd) %>%
                arrange(metro,desc(sqrt_dist_ssp)),
             digits=3)


## Influential observations - fitted vs. residuals plot
ggplot(data = mutate(augment(lm_final$fit,data=df_ops2),
                          high_lev = if_else(.hat > 3 * mean(.hat), TRUE, FALSE)),
            aes(x = .fitted, y = .std.resid, col = high_lev)) +
    geom_point() +
    geom_hline(yintercept=0, linetype = 3) +
    geom_hline(yintercept=3, linetype = 2) +
    geom_hline(yintercept=2, linetype = 3) +
    geom_hline(yintercept=-2, linetype = 3) +
    geom_hline(yintercept=-3, linetype = 2)


## Influential observations - leverage value vs. residuals plot
ggplot(data = mutate(augment(lm_final$fit,data=df_ops2),
                          high_lev = if_else(.hat > 3 * mean(.hat), TRUE, FALSE)),
            aes(x = .hat, y = .std.resid, col = high_lev)) +
    geom_point() +
    geom_hline(yintercept=0, linetype = 3) +
    geom_hline(yintercept=3, linetype = 2) +
    geom_hline(yintercept=2, linetype = 3) +
    geom_hline(yintercept=-2, linetype = 3) +
    geom_hline(yintercept=-3, linetype = 2)

# VIF Calculations
lm_final %>%
  check_model(check='vif')
#Histogram of Residuals
#lm_final$fit %>%
#  augment(data=df_ops2) %>%
#  ggplot(aes(x=.resid)) +
#  geom_histogram(bins = 25) +
```

```r
# residual density plot with normal curve
lm_final %>%
  check_model(check='normality')


#Normal QQ Plot
plot(lm_final$fit,2)
#Scatterplot of residual vs predicted for homoscedasticity check
lm_final$fit %>%
  augment(data=df_ops2) %>%
  ggplot(aes(y=.resid, x=.fitted)) +
    geom_point(col="#6E0000") +
    geom_hline(yintercept=0, linetype = 3) +
    geom_hline(yintercept=3, linetype = 2) +
    geom_hline(yintercept=2, linetype = 3) +
    geom_hline(yintercept=-2, linetype = 3) +
    geom_hline(yintercept=-3, linetype = 2) +
    scale_x_continuous("Fitted Values") +
    scale_y_continuous("Residuals")

lm_final %>%
  check_model(check='linearity')

# residuals vs continous predictors
for (pred in c('hiv_prevalence', 'opioid_rx_rate', 'pct_unins')){
  p <- lm_final$fit %>%
    augment(data=df_ops2) %>%
    ggplot(aes(y=.std.resid, x=get(pred))) +
    geom_point(col="#6E0000") +
    geom_hline(yintercept=0, linetype = 3) +
    geom_hline(yintercept=3, linetype = 2) +
    geom_hline(yintercept=2, linetype = 3) +
    geom_hline(yintercept=-2, linetype = 3) +
    geom_hline(yintercept=-3, linetype = 2) +
    ggtitle(paste(pred,"vs. Standardized Residual"))+
    scale_x_continuous(pred) +
    scale_y_continuous("Standardized Residuals")
  print(p)
}


#Durbin Watson test of independence
set.seed(123)
dwtest(lm_final$fit, iterations=15)
# most important predictors
vip(lm_final)
#View Confidence intervals for the coefficients
knitr::kable(confint(lm_final$fit), digits=4)
#View plot actual vs predicted sales
lm_final$fit %>%
  ggplot(aes(x = .fitted, y = sqrt_dist_ssp)) +
    geom_point(col = "#6e0000") +
    xlab("predicted sqrt_dist_ssp") +
```

```
    geom_abline(col="orange") +
    ggtitle(paste("Final Model: Predicted vs Fitted.     Adj R-Sq:",round(summary(lm_final$fit)$adj.r.s

# dist_ssp models
predictors = c('hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region',
               'hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region2',
               'hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3',
               'hiv_prevalence+opioid_rx_rate+ pct_unins+metro',
               'log_hp+opioid_rx_rate+pct_unins+ metro',
               'opioid_rx_rate+pct_unins+metro',
               'opioid_rx_rate+pct_unins+metro+ region3',
               'opioid_rx_rate+pct_unins+metro+ region2'
               )
insig_vars = c('hiv_prev, regionNortheast, regionSouth',
               'hiv_prev, region2North_South',
               'hiv_prev',
               'hiv_prev',
               'log_hp',
               '',
               '',
               'region2North_South')

out_metrics <- tibble(Predictors = predictors,
                      Target = "dist_ssp")
AR2s <- c()
MAEs <- c()
for (p in predictors){
  reg_curr_fit <- lm_model %>%
    fit(formula(paste('dist_ssp ~', p)), data = df_ops)
  out <- select(tidy(reg_curr_fit$fit), term, p.value)
  #print(knitr::kable(out, digits=3)) #this row can be included if you want to see the variable p-value
  AR2s <- c(AR2s, round(pull(select(glance(reg_curr_fit$fit),adj.r.squared)),3))
  curr_MAE <- pull((reg_curr_fit$fit %>%
              augment(df_ops) %>%
              metrics(truth = dist_ssp, estimate = .fitted))[3,3])
  MAEs <- c(MAEs, round(curr_MAE,3))
}

out_metrics <- out_metrics %>%
          mutate(AdjRSquare = AR2s,
                 MAE = MAEs,
                 Insig_Vars = insig_vars)


# sqrt_dist_ssp models

predictors2 = c('hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region',
                'hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region2',
                'hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3',
                'hiv_prevalence+opioid_rx_rate+ pct_unins+metro',
                'log_hp+opioid_rx_rate+pct_unins+metro',
                'log_hp+opioid_rx_rate+pct_unins+ metro+region2',
                'log_hp+opioid_rx_rate+pct_unins+ metro+region3',
```

```
                'opioid_rx_rate+pct_unins+metro',
                'opioid_rx_rate+pct_unins+metro+region3',
                'log_hp+opioid_rx_rate+pct_unins+ metro+region3+log_hp*pct_unins',
                'log_hp+opioid_rx_rate+pct_unins+ metro+region3+metro*pct_unins',
                'log_hp+opioid_rx_rate+pct_unins+ metro+region3+region3*pct_unins',
                'opioid_rx_rate+pct_unins+metro+ region3+region3*pct_unins',
                'opioid_rx_rate+pct_unins+metro+ region3+metro*pct_unins',
                'hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3+ opioid_rx_rate*hiv_prevalence'
                'log_hp+opioid_rx_rate+pct_unins+ metro+region3+ opioid_rx_rate*log_hp',
                'hiv_prevalence+opioid_rx_rate+ pct_unins+metro+region3+ opioid_rx_rate*hiv_prevalence+
                )

insig_vars2 = c('hiv_prev, regionNortheast, regionSouth, opioid_rx_rate',
                'hiv_prev, region2North_South',
                'hiv_prev',
                'hiv_prev, opioid_rx_rate',
                'log_hp, opioid_rx_rate',
                'log_hp,region2North_South',
                'log_hp',
                'opioid_rx_rate',
                '',
                'log_hp, pct_unins, log_hp*pct_unins',
                'log_hp',
                'log_hp, pct_unins,region3North_south',
                'region3North_south',
                '',
                '',
                '',
                ''
                )

out_metrics2 <- tibble(Predictors = predictors2,
                       Target = 'sqrt_dist_ssp')
AR2s2 <- c()
MAEs2 <- c()
for (p in predictors2){
  reg_curr_fit <- lm_model %>%
    fit(formula(paste('sqrt_dist_ssp ~', p)), data = df_ops)
  out <- select(tidy(reg_curr_fit$fit), term, p.value)
  #print(knitr::kable(out, digits=3))
  AR2s2 <- c(AR2s2, round(pull(select(glance(reg_curr_fit$fit),adj.r.squared)),3))
  curr_MAE <- pull((reg_curr_fit$fit %>%
              augment(df_ops) %>%
              metrics(truth = sqrt_dist_ssp, estimate = .fitted))[3,3])
  MAEs2 <- c(MAEs2, round(curr_MAE,3))
}

out_metrics2 <- out_metrics2 %>%
          mutate(AdjRSquare = AR2s2,
                 MAE = MAEs2,
                 Insig_Vars = insig_vars2)

total_metrics <- rbind(out_metrics,out_metrics2)
```

```
knitr::kable(total_metrics,digits=3)
```