CDC 500 Cities: Healthcare Access, Behaviors, and Health Outcomes

Stat 198 Final Project

Maya Ghanem and Isabelle Xiong

11/1/2021

Description of Data

(Include description of how you edited the data)

Research Questions

1) Do cities with a greater lack of healthcare access have poorer mental health and/or physical health outcomes?

2)Does healthcare access, mental health, and/or physical health outcomes vary by state?

Variables of Interest

Explanatory Variables:

1) Healthcare Access for Adults (18+): Percent of City Population that Lacks Insurance, Percent of City Population with visits to doctor for routine checkup within the past year, Percent of City Population who have high blood pressure and are taking medicine for high blood pressure control.
2) Geographic Distribution by State

Response Variables:

1) Behavior for Adults (18+): Percent of city population currently smoking, percent of city population currently reporting binge drinking habits, percent of city population reporting No leisure-time physical activity
2) Health Outcomes for Adults (18+): Percent of city population with coronary heart disease, percent of population diagnosed with diabetes, percent of city population with kidney disease

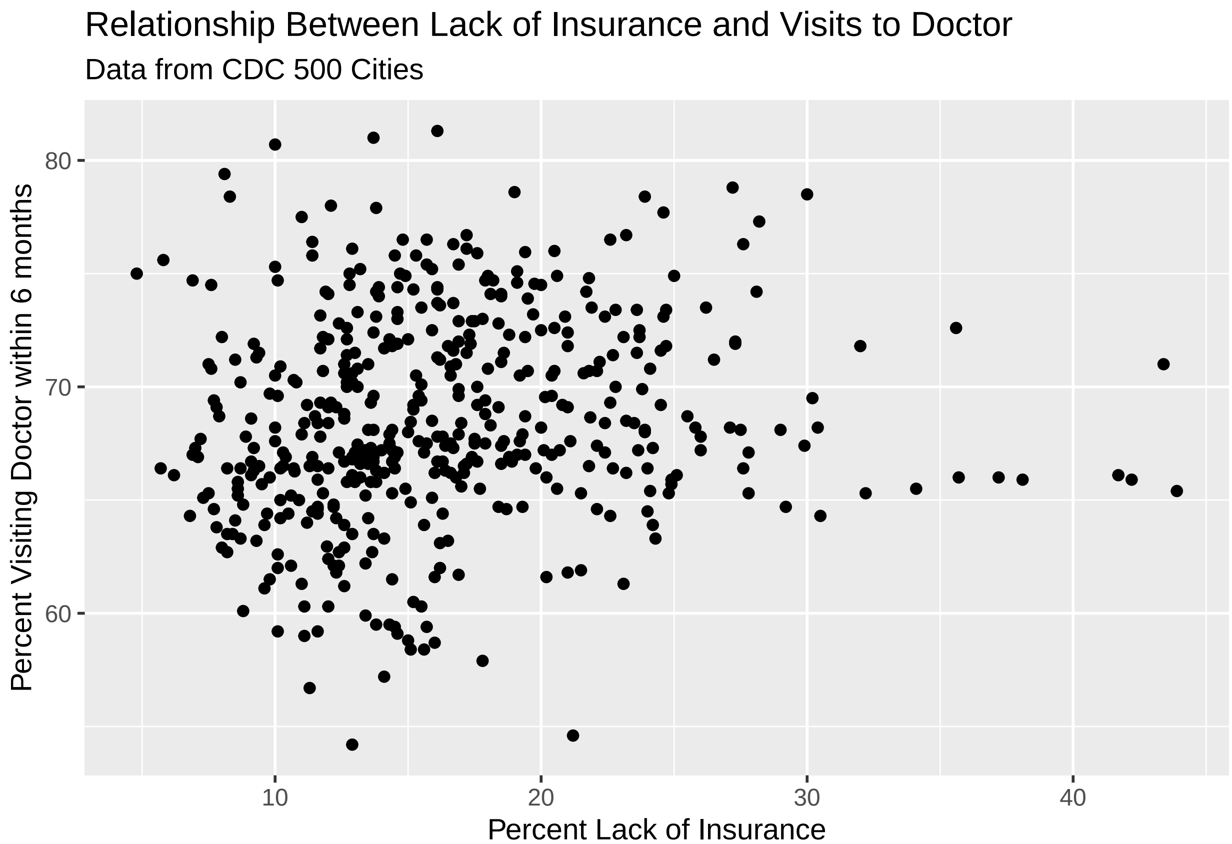
Linear Regressions

New Method:

a) Run correlations between the explanatory variables
b) Run linear regressions and adjusted r squared values
c) Assess which regression is better
d) Run the residual plot and the graph

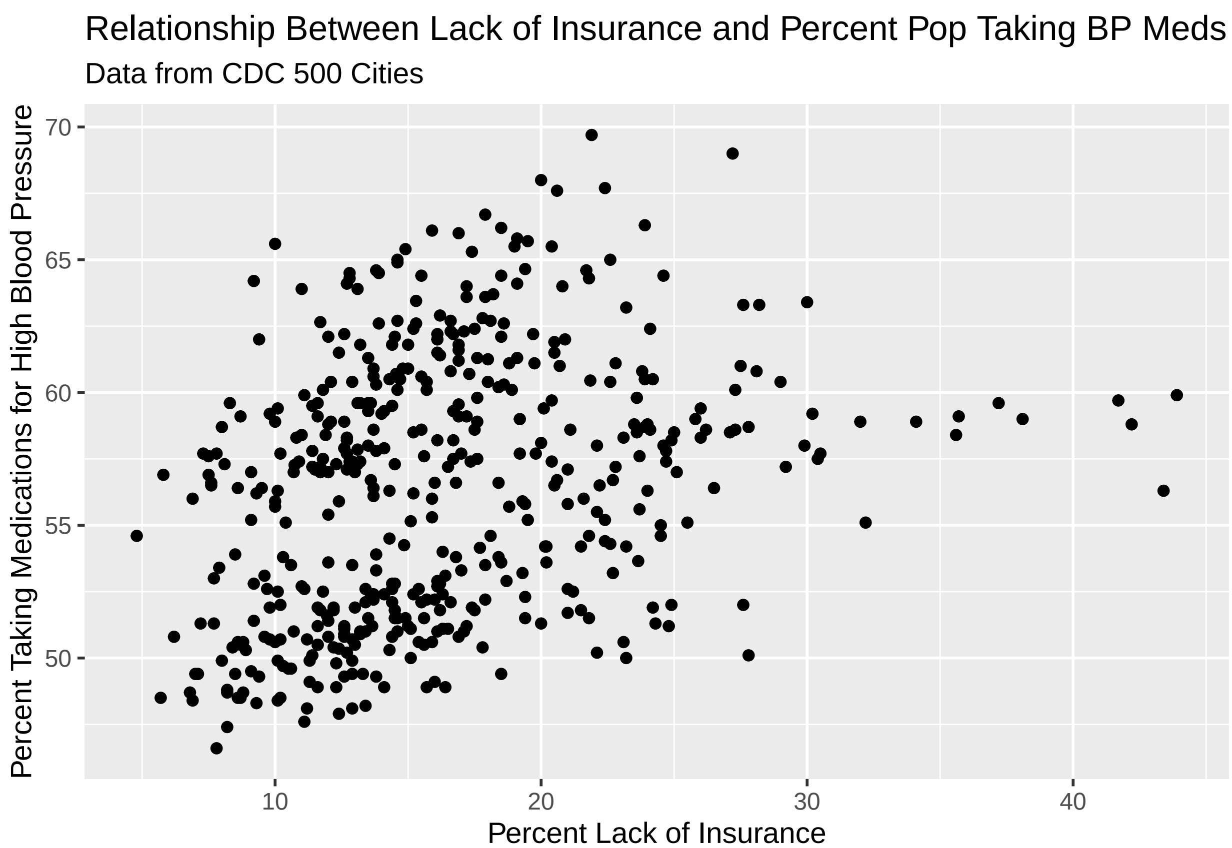
Correlations between Explanatory Variables

```
data_500_cities %>%
  ggplot(mapping = aes(x = insurance, y = visits_to_doctor)) +
  geom_point() +
  labs(
    title = "Relationship Between Lack of Insurance and Visits to Doctor",
    subtitle = "Data from CDC 500 Cities",
    x = "Percent Lack of Insurance",
    y = "Percent Visiting Doctor within 6 months"
  )
```



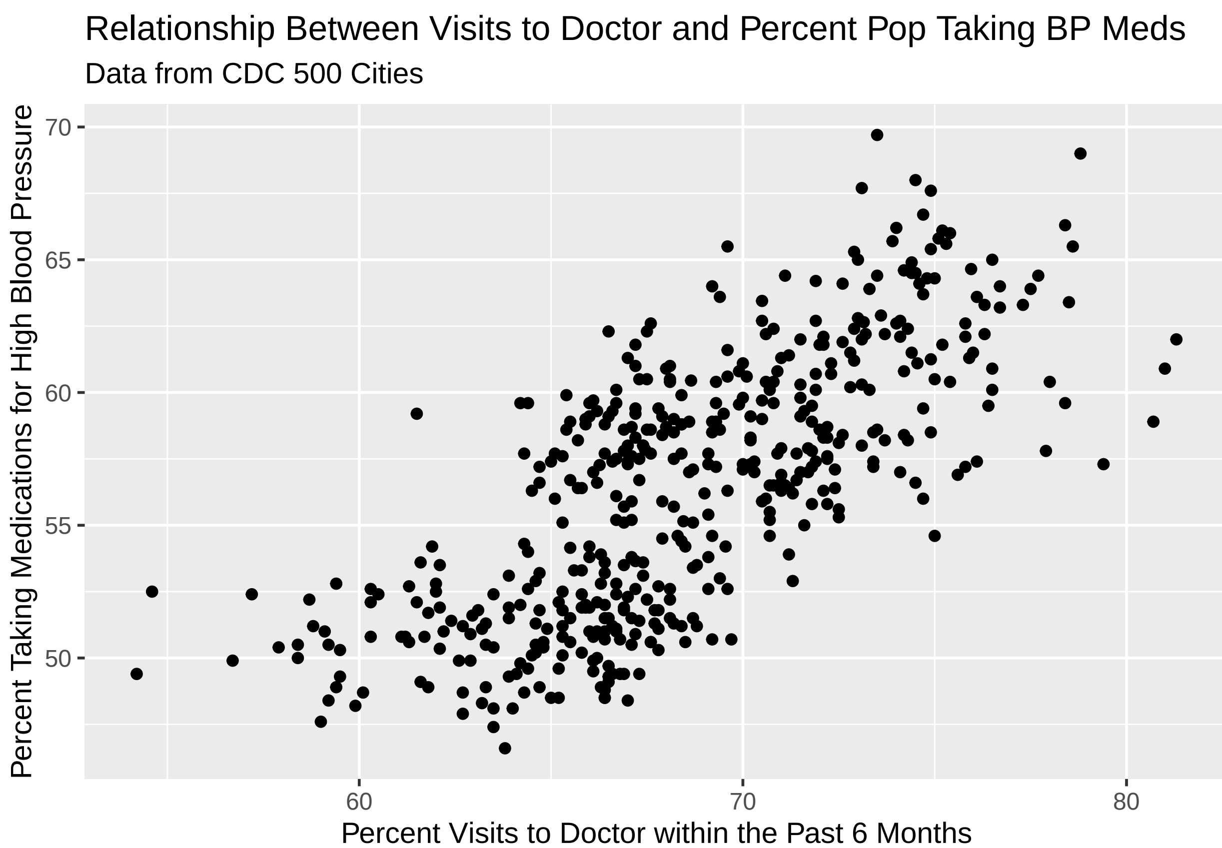
There does not seem to be any significant correlation.

```
data_500_cities %>%
  ggplot(mapping = aes(x = insurance, y = medicine_high_bp)) +
  geom_point() +
  labs(
    title = "Relationship Between Lack of Insurance and Percent Pop Taking BP Meds",
    subtitle = "Data from CDC 500 Cities",
    x = "Percent Lack of Insurance",
    y = "Percent Taking Medications for High Blood Pressure"
  )
```



There does not seem to be any significant correlation.

```
data_500_cities %>%
  ggplot(mapping = aes(x = visits_to_doctor, y = medicine_high_bp)) +
  geom_point() +
  labs(
    title = "Relationship Between Visits to Doctor and Percent Pop Taking BP Meds",
    subtitle = "Data from CDC 500 Cities",
    x = "Percent Visits to Doctor within the Past 6 Months",
    y = "Percent Taking Medications for High Blood Pressure"
  )
```



There seems to be a significant correlation between Visits to Doctor and Taking Medications.

As a result, I will test three models: one with no interaction variables, one with only one interaction variable (Visits_to_Doctor * medicine_high_bp), and one with all three interaction variables.

Access Variables vs. Smoking

Running Linear Regressions

Linear Regression with All Interaction Variables:

```
access_smoking_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(smoking ~ insurance + visits_to_doctor + medicine_high_bp, data = data_500_cities)
access_smoking_fit_aug <- augment(access_smoking_fit$fit)
tidy(access_smoking_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)      -15.0       2.08       -7.23 1.99e-12
## 2 insurance          0.0523    0.0237      2.21 2.79e- 2
## 3 visits_to_doctor  -0.0966    0.0446     -2.17 3.08e- 2
## 4 medicine_high_bp   0.674     0.0438     15.4  1.59e-43
```

Linear Regression with one interaction variable:

```
one_access_smoking_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(smoking ~ insurance + visits_to_doctor + medicine_high_bp + (visits_to_doctor * medicine_high_bp)
one_access_smoking_fit_aug <- augment(one_access_smoking_fit$fit)
tidy(one_access_smoking_fit) %>%
  print()
```

```
## # A tibble: 5 x 5
##   term                             estimate std.error statistic   p.value
##   <chr>                               <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)                       84.8     24.7          3.43 0.000657
## 2 insurance                          0.0653   0.0235       2.77 0.00576
## 3 visits_to_doctor                  -1.54     0.360       -4.29 0.0000217
## 4 medicine_high_bp                  -1.12     0.444       -2.52 0.0121
## 5 visits_to_doctor:medicine_high_bp  0.0258   0.00637      4.05 0.0000594
```

Linear Regression with All Interaction Variables

```
int_access_smoking_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(smoking ~ insurance + visits_to_doctor + medicine_high_bp + (insurance * visits_to_doctor) + (insu
int_access_smoking_fit_aug <- augment(int_access_smoking_fit$fit)
tidy(int_access_smoking_fit) %>%
  print()
```

```
## # A tibble: 7 x 5
##   term                             estimate std.error statistic  p.value
##   <chr>                               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                       88.9     24.0          3.70 2.41e- 4
## 2 insurance                          0.872    0.417        2.09 3.71e- 2
## 3 visits_to_doctor                  -2.13     0.362       -5.90 6.95e- 9
## 4 medicine_high_bp                  -0.756    0.463       -1.63 1.03e- 1
## 5 insurance:visits_to_doctor         0.0227   0.00634      3.59 3.69e- 4
## 6 insurance:medicine_high_bp        -0.0414   0.00628     -6.58 1.25e-10
## 7 visits_to_doctor:medicine_high_bp  0.0299   0.00667      4.48 9.60e- 6
```

Comparing Adj R-Squared Values

Adj R-Squared Value with No Interactions:

```
glance(access_smoking_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.5150724
```

Adj R-Squared Value with One Interactions:

```
glance(one_access_smoking_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.5305757
```

Adj R-Squared Value with All Interactions:

```
glance(int_access_smoking_fit)$adj.r.squared %>%
  print()
```

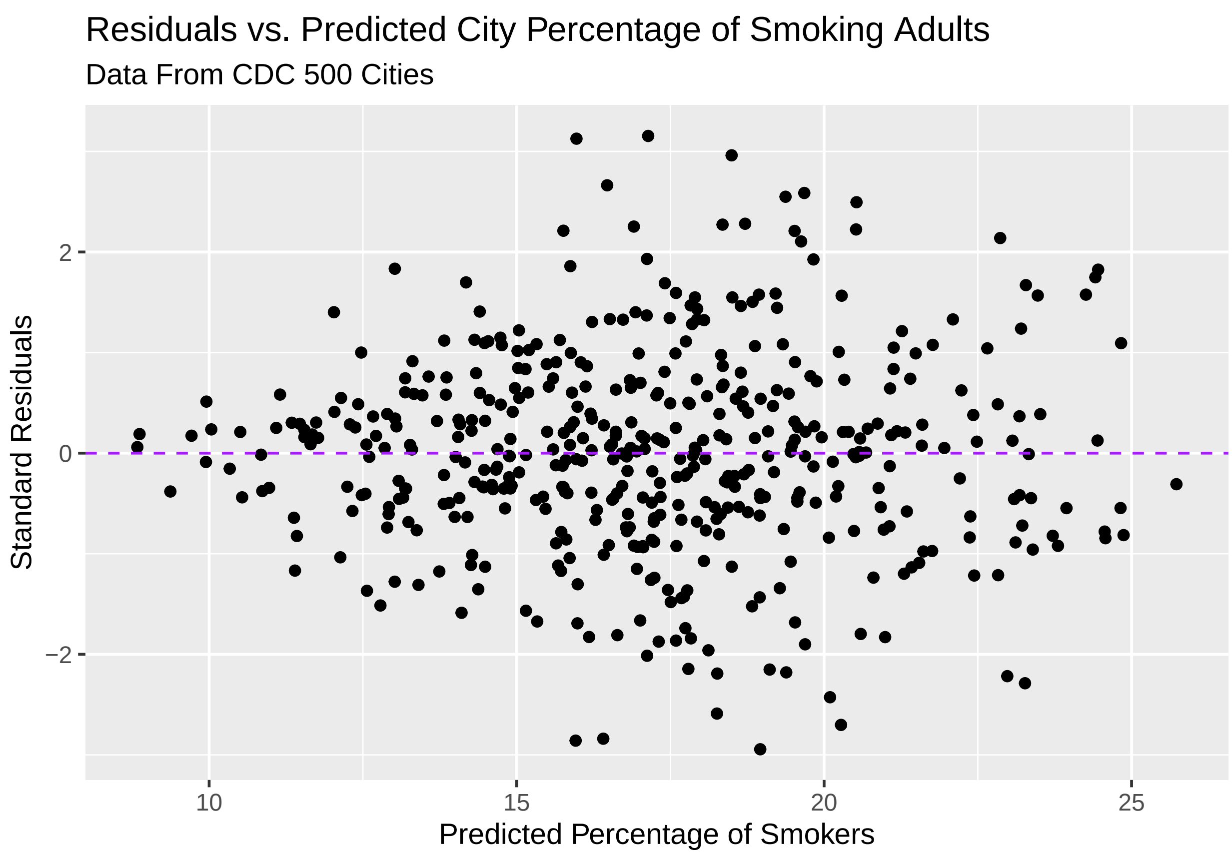
```
## [1] 0.5691301
```

The linear regression with all second order interactions that account for relationships between all explanatory variables is most appropriate because it has the highest adj R-squared value. We will use this regression in displaying our graphs.

Displaying Graphs

Residual Graph

```
int_access_smoking_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted City Percentage of Smoking Adults",
    subtitle = "Data From CDC 500 Cities",
    x = "Predicted Percentage of Smokers",
    y = "Standard Residuals"
  )
```

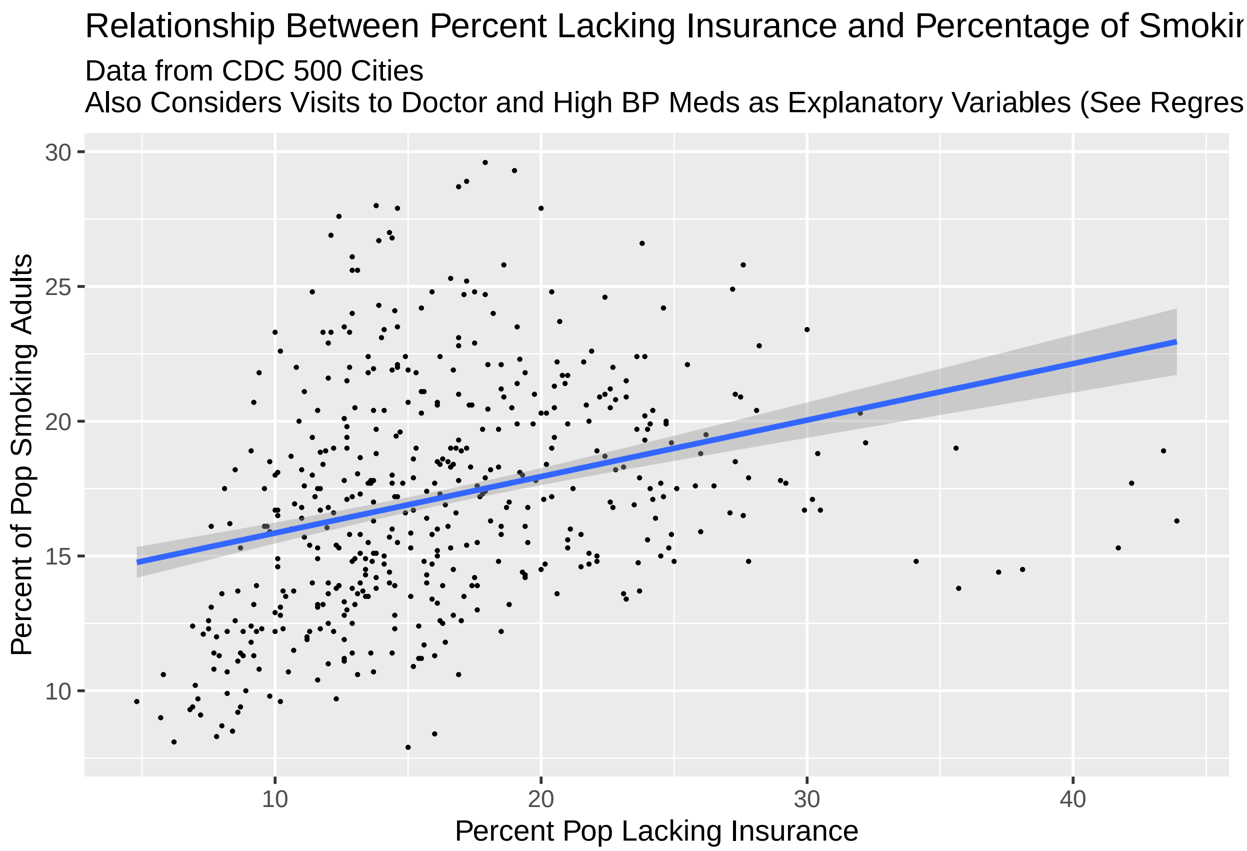


There does not seem to be any patterns in this residual graph, so a linear model would be appropriate.

Graph Between Explanatory and Response Variables

```
data_500_cities %>%
ggplot( mapping = aes(x = insurance, y = smoking)) +
geom_point(size = 0.25) +
geom_smooth(method = "lm", data = int_access_smoking_fit_aug, mapping = aes(x = insurance, y = .fitted)
  labs(
```

```
    title = "Relationship Between Percent Lacking Insurance and Percentage of Smoking Adults",
    subtitle = "Data from CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
    x = "Percent Pop Lacking Insurance",
    y = "Percent of Pop Smoking Adults"
  )
```



Percent of smoking adults in a city seems to increase with percent of adults in city lacking insurance.

Access Variables vs. Binge Drinking

Running Linear Regressions

Linear Regression for no interactions:

```
access_binge_drinking_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(binge_drinking ~ insurance + visits_to_doctor + medicine_high_bp, data = data_500_cities)
access_binge_drinking_fit_aug <- augment(access_binge_drinking_fit$fit)
tidy(access_binge_drinking_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       24.2       1.58       15.3  2.65e-43
## 2 insurance         -0.162     0.0179     -9.02 4.74e-18
## 3 visits_to_doctor   0.0565    0.0337      1.68 9.45e- 2
```

```
## 4 medicine_high_bp  -0.137     0.0331     -4.13 4.39e- 5
```

Linear regression with one interaction:

```
one_access_binge_drinking_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(binge_drinking ~ insurance + visits_to_doctor + medicine_high_bp + (visits_to_doctor * medicine_hi
one_access_binge_drinking_fit_aug <- augment(one_access_binge_drinking_fit$fit)
tidy(one_access_binge_drinking_fit) %>%
  print()
```

```
## # A tibble: 5 x 5
##   term                              estimate std.error statistic  p.value
##   <chr>                                <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                       -133.      17.6        -7.57 1.97e-13
## 2 insurance                           -0.183    0.0167     -10.9  8.43e-25
## 3 visits_to_doctor                     2.34     0.256        9.13 2.03e-18
## 4 medicine_high_bp                     2.69     0.316        8.50 2.50e-16
## 5 visits_to_doctor:medicine_high_bp   -0.0407   0.00453     -8.98 6.76e-18
```

Linear regression with all interactions:

```
int_access_binge_drinking_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(binge_drinking ~ insurance + visits_to_doctor + medicine_high_bp + (insurance * visits_to_doctor)
int_access_binge_drinking_fit_aug <- augment(int_access_binge_drinking_fit$fit)
tidy(int_access_binge_drinking_fit) %>%
  print()
```

```
## # A tibble: 7 x 5
##   term                               estimate std.error statistic  p.value
##   <chr>                                 <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                       -132.       17.8        -7.40  6.26e-13
## 2 insurance                           -0.125     0.309      -0.406 6.85e- 1
## 3 visits_to_doctor                     2.41      0.268       8.98  6.70e-18
## 4 medicine_high_bp                     2.54      0.344       7.38  7.12e-13
## 5 insurance:visits_to_doctor          -0.00655   0.00470    -1.39  1.64e- 1
## 6 insurance:medicine_high_bp           0.00686   0.00466     1.47  1.42e- 1
## 7 visits_to_doctor:medicine_high_bp   -0.0401    0.00495    -8.10  4.93e-15
```

Comparing Adj R-Squared Values

Adj R-squared value for regression with no interactions:

```
glance(access_binge_drinking_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.2367489
```

Adj R-squared value for regression with one interaction:

```
glance(one_access_binge_drinking_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.347712
```

Adj R-squared value for regression with all interactions:

```
glance(int_access_binge_drinking_fit)$adj.r.squared %>%
  print()
```

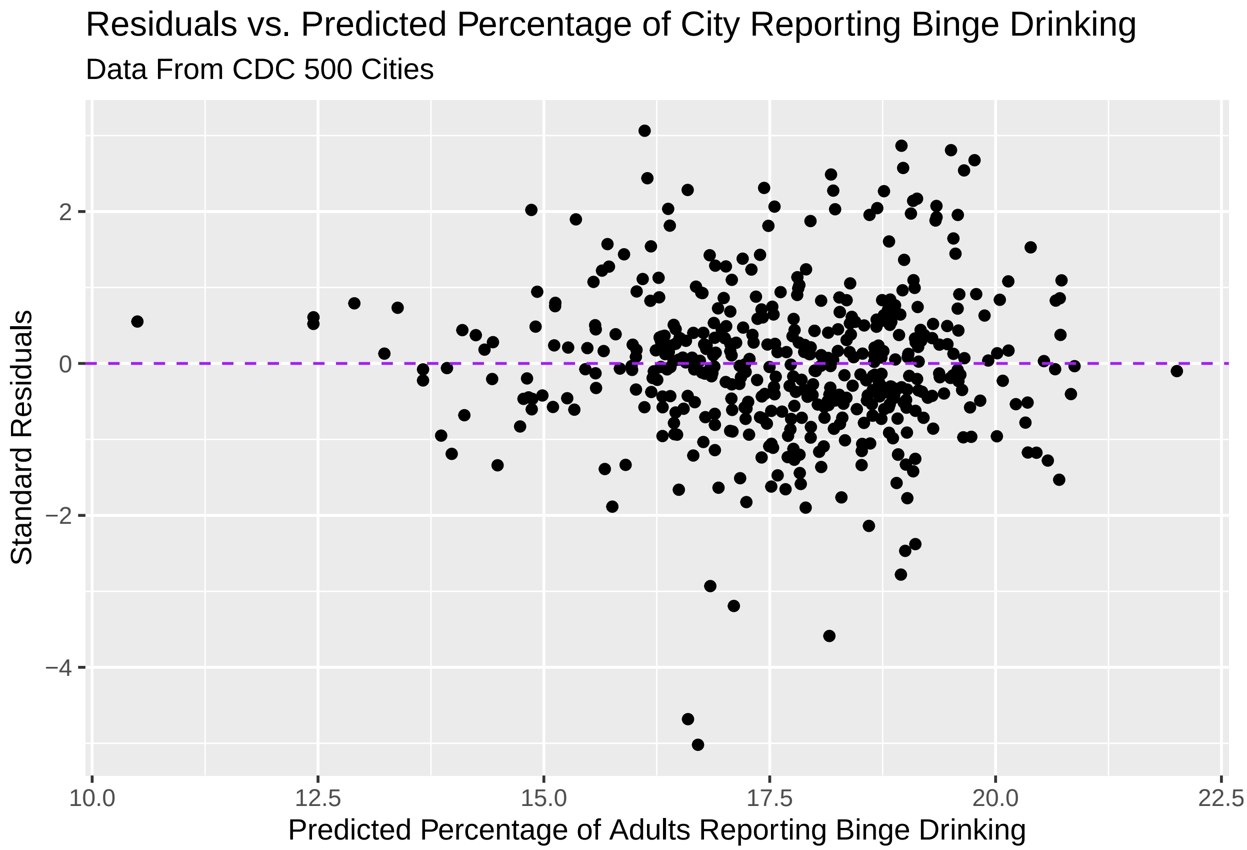
```
## [1] 0.3488416
```

The linear regression with all second order interactions that account for relationships between explanatory variables is most appropriate because it has the highest adj R-squared value. We will use this regression in displaying our graphs.

Displaying Graphs

Residual Graph

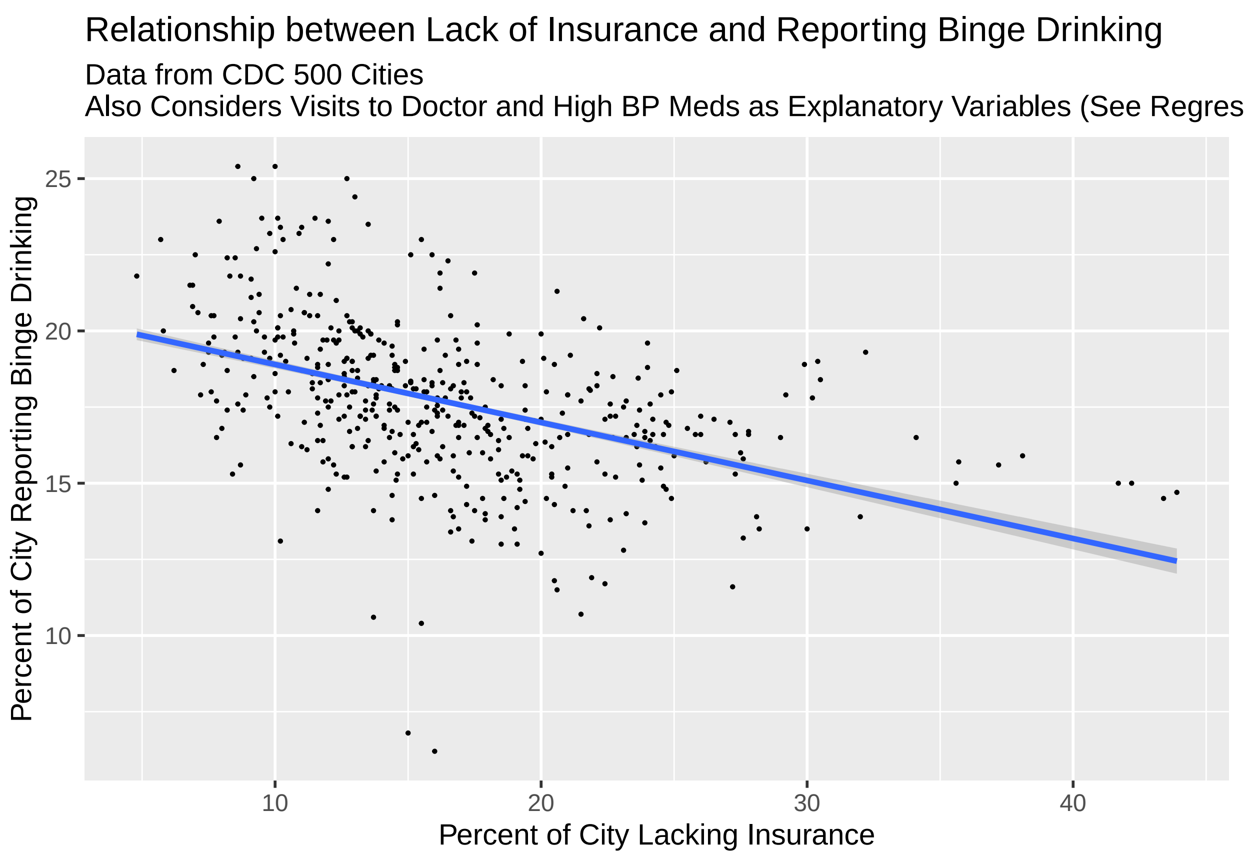
```
int_access_binge_drinking_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted Percentage of City Reporting Binge Drinking",
    subtitle = "Data From CDC 500 Cities",
    x = "Predicted Percentage of Adults Reporting Binge Drinking",
    y = "Standard Residuals"
  )
```



There doesn't seem to be any major patterns in this residual graph, except for some clumping around the mean residual. A linear regression still seems appropriate.

Graph Comparing Explanatory and Response Variables

```
data_500_cities %>%
ggplot(mapping = aes(x = insurance, y = binge_drinking)) +
geom_point(size = 0.25) +
geom_smooth(method = "lm", data = int_access_binge_drinking_fit_aug, mapping = aes(x = insurance, y = .f
  labs(
    title = "Relationship between Lack of Insurance and Reporting Binge Drinking",
    subtitle = "Data from CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
    x = "Percent of City Lacking Insurance",
    y = "Percent of City Reporting Binge Drinking"
  )
```



As the percentage of city population lacking health insurance increases, the percentage of city reporting binge drinking decreases.

Access Variables vs. Physical Activity

Running Linear Regressions

Linear regression with no interactions

```
access_physical_activity_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(physical_activity ~ insurance + visits_to_doctor + medicine_high_bp, data = data_500_cities)
access_physical_activity_fit_aug <- augment(access_physical_activity_fit$fit)
tidy(access_physical_activity_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)      -28.1       1.77      -15.9  5.76e-46
## 2 insurance          0.533     0.0201     26.5  3.31e-95
## 3 visits_to_doctor   0.0625    0.0378      1.65 9.95e- 2
## 4 medicine_high_bp   0.738     0.0371     19.9  3.54e-64
```

Linear regression with one interaction

```
one_access_physical_activity_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(physical_activity ~ insurance + visits_to_doctor + medicine_high_bp + (visits_to_doctor * medicine
one_access_physical_activity_fit_aug <- augment(one_access_physical_activity_fit$fit)
tidy(one_access_physical_activity_fit) %>%
  print()
```

```
## # A tibble: 5 x 5
##   term                              estimate std.error statistic  p.value
##   <chr>                                <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                        43.5     21.1          2.06 3.98e- 2
## 2 insurance                           0.543    0.0201      27.0  1.71e-97
## 3 visits_to_doctor                   -0.976    0.307       -3.18 1.57e- 3
## 4 medicine_high_bp                   -0.548    0.379       -1.44 1.49e- 1
## 5 visits_to_doctor:medicine_high_bp   0.0185   0.00543      3.41 7.11e- 4
```

Linear regression with all interactions

```
int_access_physical_activity_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(physical_activity ~ insurance + visits_to_doctor + medicine_high_bp + (insurance * visits_to_docto
int_access_physical_activity_fit_aug <- augment(int_access_physical_activity_fit$fit)
tidy(int_access_physical_activity_fit) %>%
  print()
```

```
## # A tibble: 7 x 5
##   term                              estimate std.error statistic      p.value
##   <chr>                                <dbl>     <dbl>     <dbl>        <dbl>
## 1 (Intercept)                      55.1       20.8         2.64  0.00845
## 2 insurance                         1.96       0.361       5.42  0.0000000972
## 3 visits_to_doctor                 -1.47       0.313      -4.69  0.00000361
## 4 medicine_high_bp                 -0.744      0.402      -1.85  0.0646
## 5 insurance:visits_to_doctor        0.000790   0.00549     0.144 0.886
## 6 insurance:medicine_high_bp       -0.0257     0.00545    -4.72  0.00000317
## 7 visits_to_doctor:medicine_high_bp  0.0271     0.00578     4.68  0.00000373
```

Comparing Adj R-Squared Values

Adj R-squared value for regression with no interactions

```
glance(access_physical_activity_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.8369087
```

Adj R-squared value for regression with one interaction

```
glance(one_access_physical_activity_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.8405259
```

Adj R-squared value for regression with all interactions

```
glance(int_access_physical_activity_fit)$adj.r.squared %>%
  print()
```

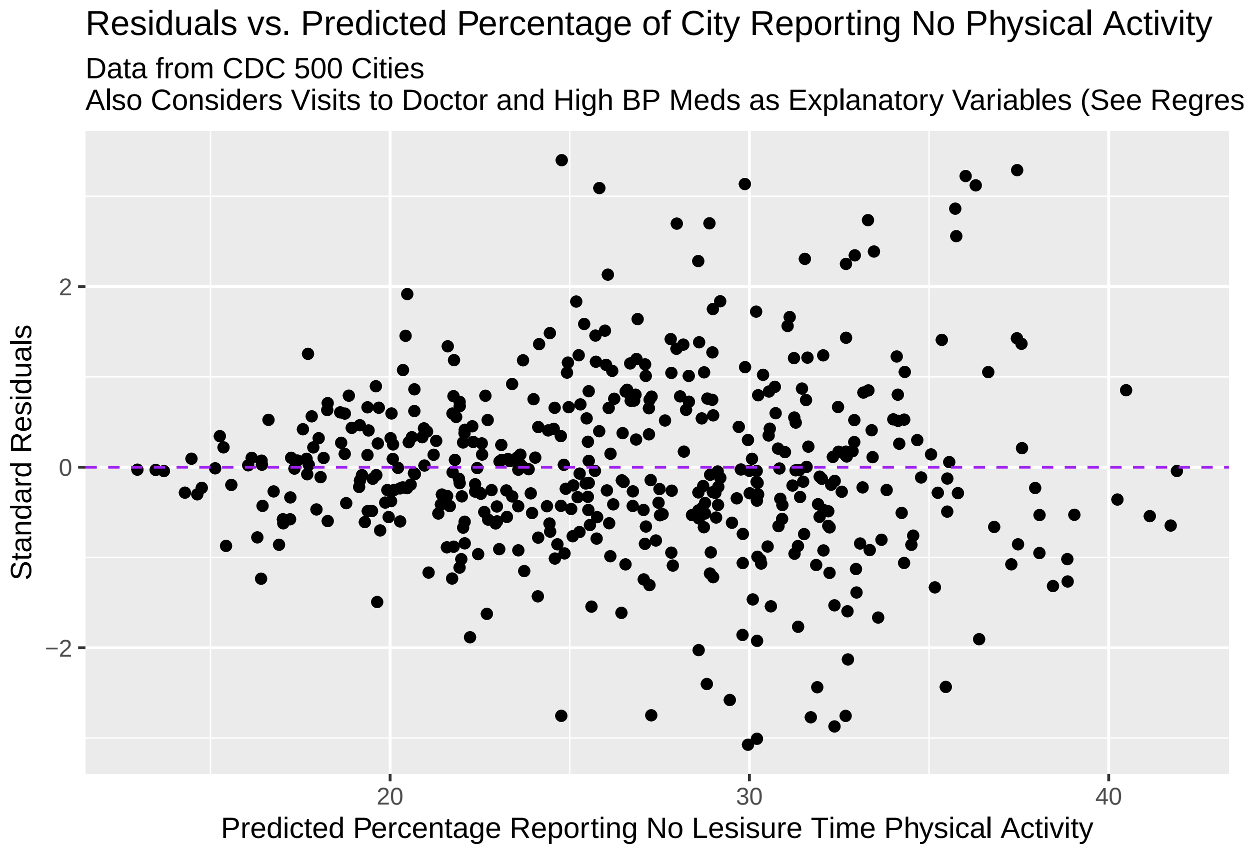
```
## [1] 0.8488063
```

The linear regression that includes all possible second order interactions for the three explanatory variables is most appropriate because it has the highest adjusted R-squared value. It will therefore be visualized in the residual plot and displayed in a graph.

Displaying Graphs

Residual Graph

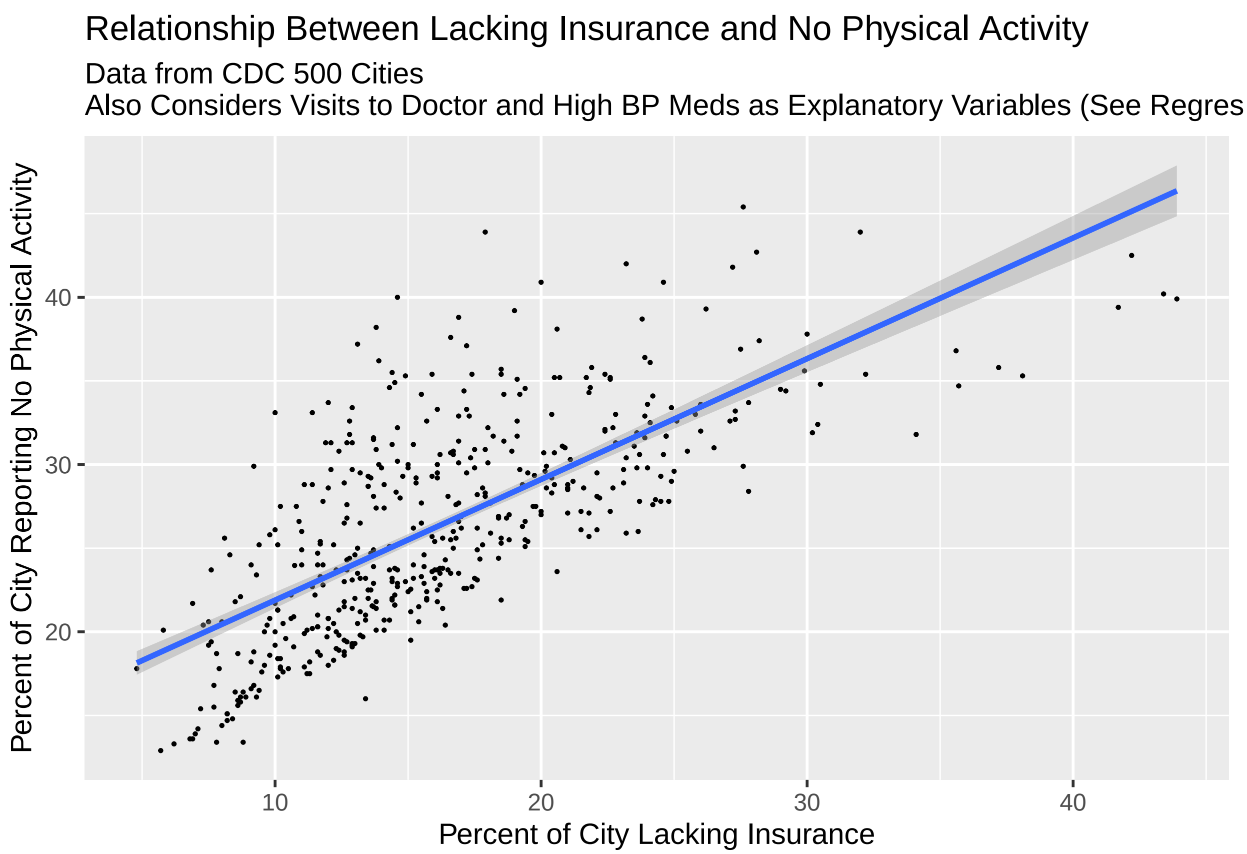
```
int_access_physical_activity_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted Percentage of City Reporting No Physical Activity",
    subtitle = "Data from CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
    x = "Predicted Percentage Reporting No Lesisure Time Physical Activity",
    y = "Standard Residuals"
  )
```



Because there does not seem to be any patterns in the residual plot, a lienar model is likely appropriate.

Graph Comparing Explanatory and Response Variables

```
data_500_cities %>%
ggplot( mapping = aes(x = insurance, y = physical_activity)) +
geom_point(size = 0.25) +
geom_smooth(method = "lm", data = int_access_physical_activity_fit_aug, mapping = aes(x = insurance, y =
  labs(
    title = "Relationship Between Lacking Insurance and No Physical Activity",
    subtitle = "Data from CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
x = "Percent of City Lacking Insurance",
y = "Percent of City Reporting No Physical Activity"
  )
```



There seems to be a very strong positive correlation between percent of city lacking health insurance and percent of city reporting no physical activity.

Access Variables vs. Coronary Heart Disease

Running Linear Regressions

Linear regression with no interactions:

```
access_heart_disease_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(heart_disease ~ insurance + visits_to_doctor + medicine_high_bp, data = data_500_cities)
access_heart_disease_fit_aug <- augment(access_heart_disease_fit$fit)
tidy(access_heart_disease_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       -1.54     0.427       -3.60 3.56e- 4
## 2 insurance          0.0669   0.00487     13.7  2.32e-36
## 3 visits_to_doctor  -0.0113   0.00916     -1.23 2.20e- 1
## 4 medicine_high_bp   0.122    0.00898     13.6  1.16e-35
```

Linear regression with one interaction

```
one_access_heart_disease_fit <- linear_reg() %>%
  set_engine("lm") %>%
```

```r
  fit(heart_disease ~ insurance + visits_to_doctor + medicine_high_bp + (visits_to_doctor * medicine_hig
access_heart_disease_fit_aug <- augment(access_heart_disease_fit$fit)
tidy(access_heart_disease_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       -1.54     0.427       -3.60 3.56e- 4
## 2 insurance          0.0669   0.00487     13.7  2.32e-36
## 3 visits_to_doctor  -0.0113   0.00916     -1.23 2.20e- 1
## 4 medicine_high_bp   0.122    0.00898     13.6  1.16e-35
```

Linear regression with all interactions

```r
int_access_heart_disease_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(heart_disease ~ insurance + visits_to_doctor + medicine_high_bp + (insurance * visits_to_doctor) +
int_access_heart_disease_fit_aug <- augment(int_access_heart_disease_fit$fit)
tidy(int_access_heart_disease_fit) %>%
  print()
```

```
## # A tibble: 7 x 5
##   term                              estimate std.error statistic  p.value
##   <chr>                                <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                       23.9       4.94         4.84 1.74e- 6
## 2 insurance                          0.352     0.0857       4.10 4.79e- 5
## 3 visits_to_doctor                  -0.480     0.0743      -6.46 2.70e-10
## 4 medicine_high_bp                  -0.289     0.0952      -3.04 2.52e- 3
## 5 insurance:visits_to_doctor         0.00239   0.00130      1.84 6.67e- 2
## 6 insurance:medicine_high_bp        -0.00780   0.00129     -6.04 3.19e- 9
## 7 visits_to_doctor:medicine_high_bp  0.00767   0.00137      5.59 3.80e- 8
```

Comparing Adj R Squared Values

Adj R-squared values for regression with no interactions

```r
glance(access_heart_disease_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.6254959
```

Adj R-squared values for regression with one interaction

```r
glance(one_access_heart_disease_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.6413167
```

Adj R-squared values for regression with all interactions

```r
glance(int_access_heart_disease_fit)$adj.r.squared %>%
  print()
```

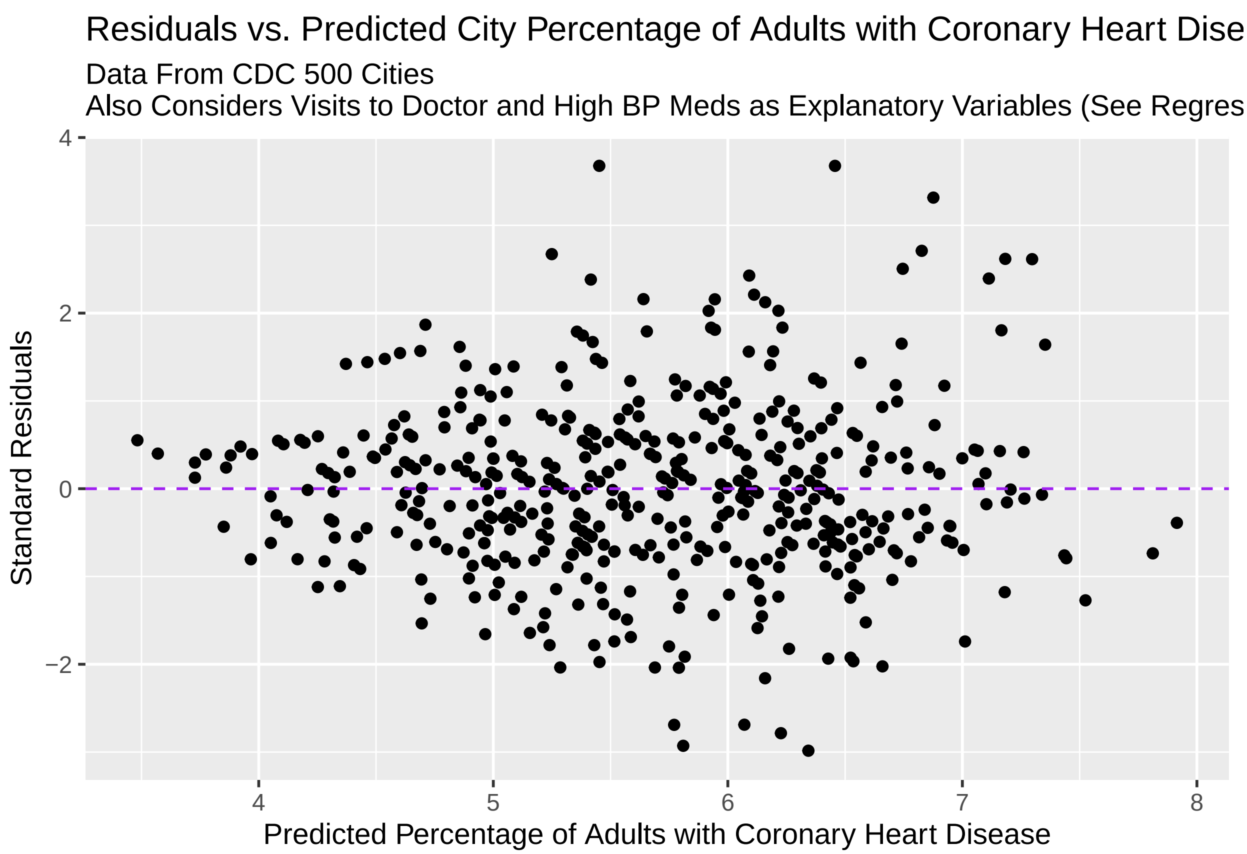
```
## [1] 0.6667498
```

The linear regression that includes all possible interactions between the three explanatory variables is most appropriate because it has the greatest adj R-squared value. This will then be used when displaying graphs.

Displaying Graphs

Residual Graphs

```
int_access_heart_disease_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted City Percentage of Adults with Coronary Heart Disease",
    subtitle = "Data From CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
    x = "Predicted Percentage of Adults with Coronary Heart Disease",
    y = "Standard Residuals"
  )
```



There does seem to be a significant pattern in the residual model, so a linear model does not seem appropriate. Try a logistical model here?

Access Variables vs. Diabetes

Running linear regressions

Linear regression with one interaction

```
access_diabetes_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(diabetes ~ insurance + visits_to_doctor + medicine_high_bp, data = data_500_cities)
access_diabetes_fit_aug <- augment(access_diabetes_fit$fit)
```

```r
tidy(access_diabetes_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       -7.57      0.982      -7.71 7.45e-14
## 2 insurance          0.239     0.0112     21.4  2.12e-71
## 3 visits_to_doctor   0.0650    0.0210      3.09 2.13e- 3
## 4 medicine_high_bp   0.171     0.0206      8.29 1.18e-15
```

Linear regression with one interaction

```r
one_access_diabetes_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(diabetes ~ insurance + visits_to_doctor + medicine_high_bp + (visits_to_doctor * medicine_high_bp)
access_diabetes_fit_aug <- augment(access_diabetes_fit$fit)
tidy(access_diabetes_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       -7.57      0.982      -7.71 7.45e-14
## 2 insurance          0.239     0.0112     21.4  2.12e-71
## 3 visits_to_doctor   0.0650    0.0210      3.09 2.13e- 3
## 4 medicine_high_bp   0.171     0.0206      8.29 1.18e-15
```

Linear regression with all interactions

```r
int_access_diabetes_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(diabetes ~ insurance + visits_to_doctor + medicine_high_bp + (insurance * visits_to_doctor) + (ins
int_access_diabetes_fit_aug <- augment(int_access_diabetes_fit$fit)
tidy(int_access_diabetes_fit) %>%
  print()
```

```
## # A tibble: 7 x 5
##   term                              estimate std.error statistic  p.value
##   <chr>                                <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                       69.9      11.4          6.12  1.97e- 9
## 2 insurance                          0.975     0.198        4.92  1.22e- 6
## 3 visits_to_doctor                  -1.07      0.172       -6.25  9.40e-10
## 4 medicine_high_bp                  -1.40      0.220       -6.36  4.72e-10
## 5 insurance:visits_to_doctor        -0.00935   0.00301     -3.10  2.03e- 3
## 6 insurance:medicine_high_bp        -0.00147   0.00299     -0.493 6.22e- 1
## 7 visits_to_doctor:medicine_high_bp  0.0230    0.00317      7.24  1.87e-12
```

Comparing Adj R-Squared Values

Adj R-squared value for regression with no interactions

```r
glance(access_diabetes_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.6797326
```

Adj R-squared value for regression with one interaction

```r
glance(one_access_diabetes_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.703361
```

Adj R-squared value for regression with all interactions

```r
glance(int_access_diabetes_fit)$adj.r.squared %>%
  print()
```

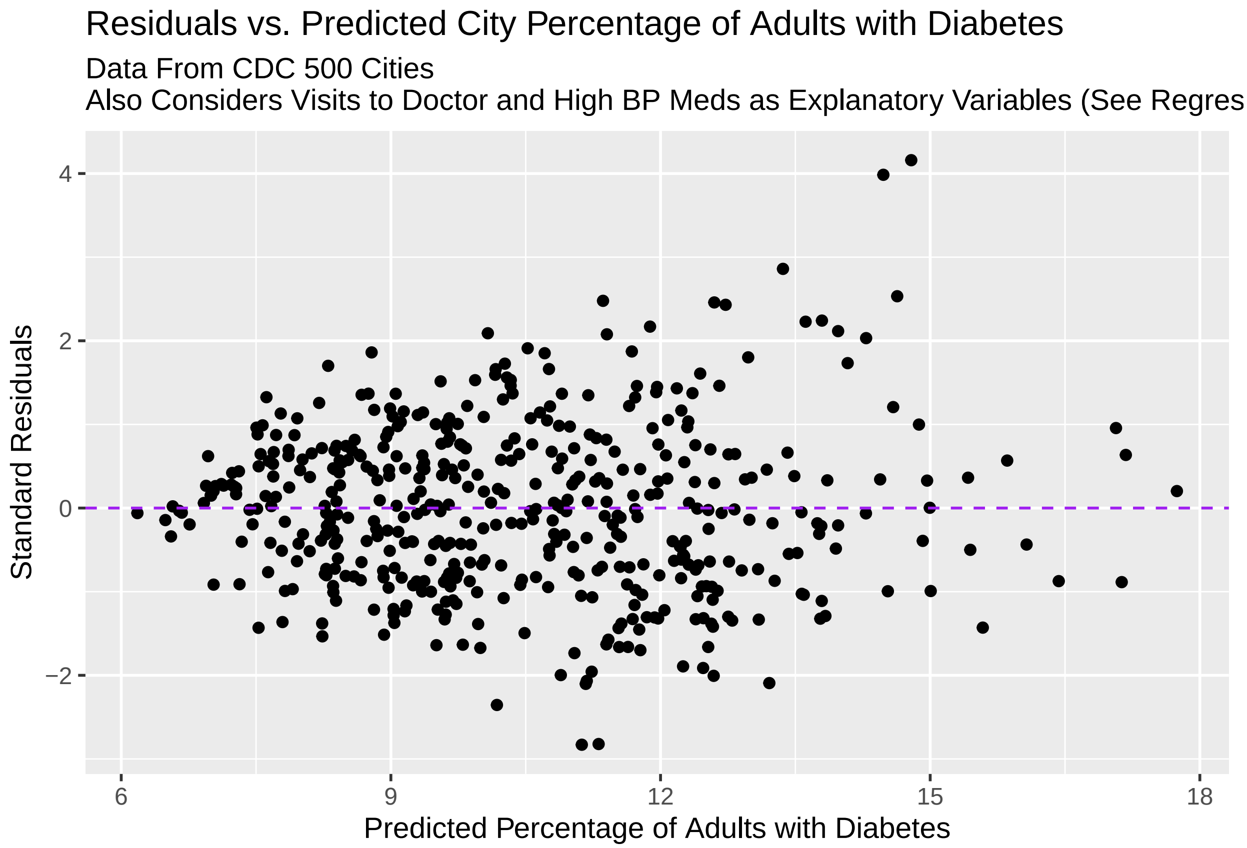
```
## [1] 0.7110294
```

The linear regression including all possible second order interactions between the explanatory variables is most appropriate because it has the highest adj R-squared value. Graphs displayed will therefore use this model.

Displaying Graphs

Residual Graph (Note any patterns)

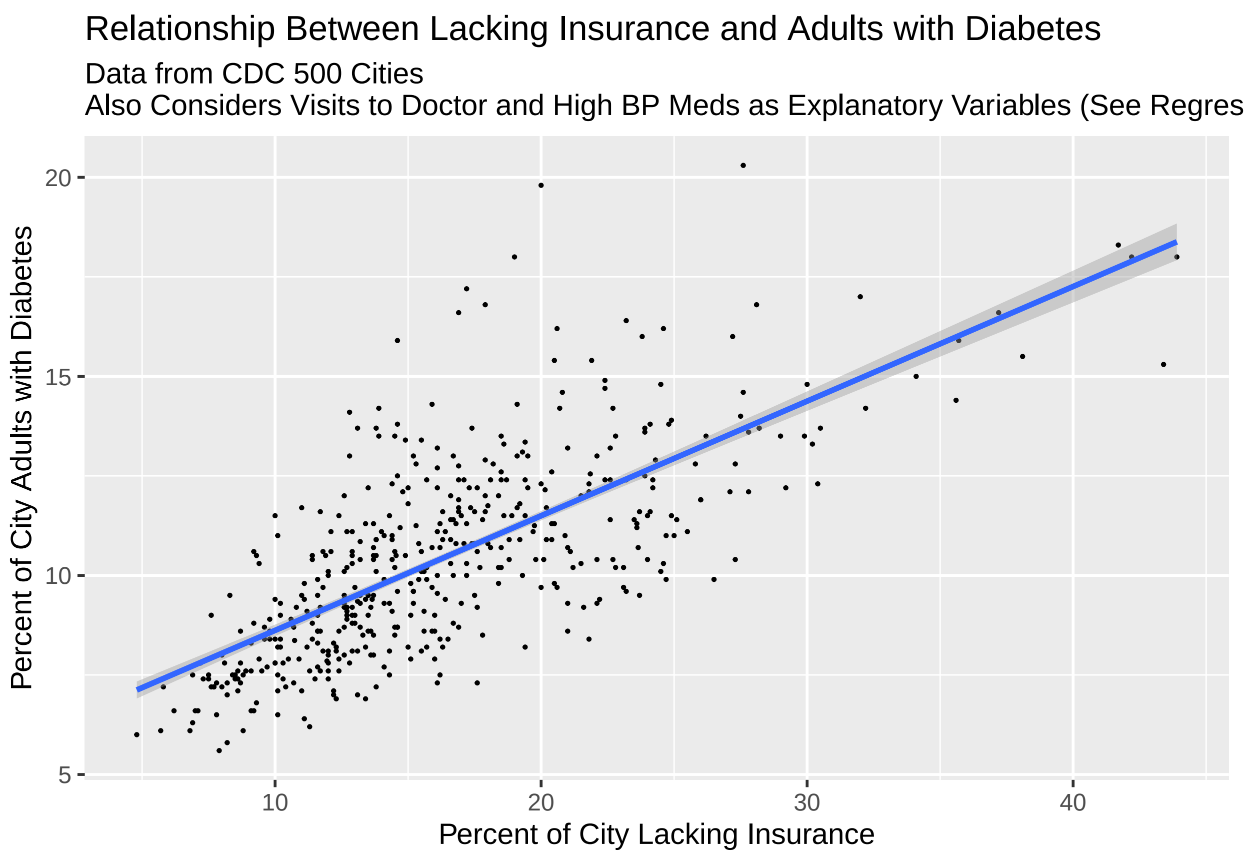
```r
int_access_diabetes_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted City Percentage of Adults with Diabetes",
    subtitle = "Data From CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
    x = "Predicted Percentage of Adults with Diabetes",
    y = "Standard Residuals"
  )
```



There does not seem to be a significant pattern in the residual plot. Therefore, a linear model is appropriate.

Graph comparing explanatory and response variables

```
data_500_cities %>%
ggplot( mapping = aes(x = insurance, y = diabetes)) +
geom_point(size = 0.25) +
geom_smooth(method = "lm", data = int_access_diabetes_fit_aug, mapping = aes(x = insurance, y = .fitted)
  labs(
    title = "Relationship Between Lacking Insurance and Adults with Diabetes",
    subtitle = "Data from CDC 500 Cities
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
x = "Percent of City Lacking Insurance",
y = "Percent of City Adults with Diabetes"
  )
```



There seems to be a strong positive correlation between percent of city lacking health insurance and percent of city adults diagnosed with diabetes.

Access Variables vs. Kidney Disease

Running Linear Regression Models

Linear Regression Model with no interactions

```
access_kidney_disease_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(kidney_disease ~ insurance + visits_to_doctor + medicine_high_bp, data = data_500_cities)
access_kidney_disease_fit_aug <- augment(access_kidney_disease_fit$fit)
tidy(access_kidney_disease_fit) %>%
  print()
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic  p.value
##   <chr>               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       0.290     0.225        1.29 1.97e- 1
## 2 insurance         0.0424    0.00256     16.6  7.48e-49
## 3 visits_to_doctor  0.00522   0.00482      1.08 2.79e- 1
## 4 medicine_high_bp  0.0305    0.00472      6.47 2.54e-10
```

Linear regression model with one interaction

```
one_access_kidney_disease_fit <- linear_reg() %>%
  set_engine("lm") %>%
```

```
  fit(kidney_disease ~ insurance + visits_to_doctor + medicine_high_bp + (visits_to_doctor * medicine_hi
one_access_kidney_disease_fit_aug <- augment(one_access_kidney_disease_fit$fit)
tidy(one_access_kidney_disease_fit) %>%
  print()
```

```
## # A tibble: 5 x 5
##   term                             estimate std.error statistic  p.value
##   <chr>                               <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                      21.7      2.53          8.59 1.34e-16
## 2 insurance                         0.0452   0.00241      18.8  4.81e-59
## 3 visits_to_doctor                 -0.305    0.0368       -8.30 1.16e-15
## 4 medicine_high_bp                 -0.354    0.0454       -7.79 4.40e-14
## 5 visits_to_doctor:medicine_high_bp  0.00554  0.000651      8.50 2.54e-16
```

Linear regression model with all interactions

```
int_access_kidney_disease_fit <- linear_reg() %>%
  set_engine("lm") %>%
  fit(kidney_disease ~ insurance + visits_to_doctor + medicine_high_bp + (insurance * visits_to_doctor)
int_access_kidney_disease_fit_aug <- augment(int_access_kidney_disease_fit$fit)
tidy(int_access_kidney_disease_fit) %>%
  print()
```

```
## # A tibble: 7 x 5
##   term                               estimate std.error statistic  p.value
##   <chr>                                 <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)                      22.9       2.50          9.16  1.63e-18
## 2 insurance                         0.198     0.0435        4.56  6.44e- 6
## 3 visits_to_doctor                 -0.361     0.0377       -9.57  6.10e-20
## 4 medicine_high_bp                 -0.372     0.0483       -7.70  8.53e-14
## 5 insurance:visits_to_doctor        0.000243  0.000661      0.368 7.13e- 1
## 6 insurance:medicine_high_bp       -0.00297   0.000655     -4.53  7.40e- 6
## 7 visits_to_doctor:medicine_high_bp  0.00646   0.000696      9.28  6.23e-19
```

Comparing Adj R-Squared Values

Adj R-squared value for regression with no interactions

```
glance(access_kidney_disease_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.5403031
```

Adj R-squared value for regression with one interaction

```
glance(one_access_kidney_disease_fit)$adj.r.squared %>%
  print()
```

```
## [1] 0.6010605
```

Adj R-squared value for regression with all interactions

```
glance(int_access_kidney_disease_fit)$adj.r.squared %>%
  print()
```

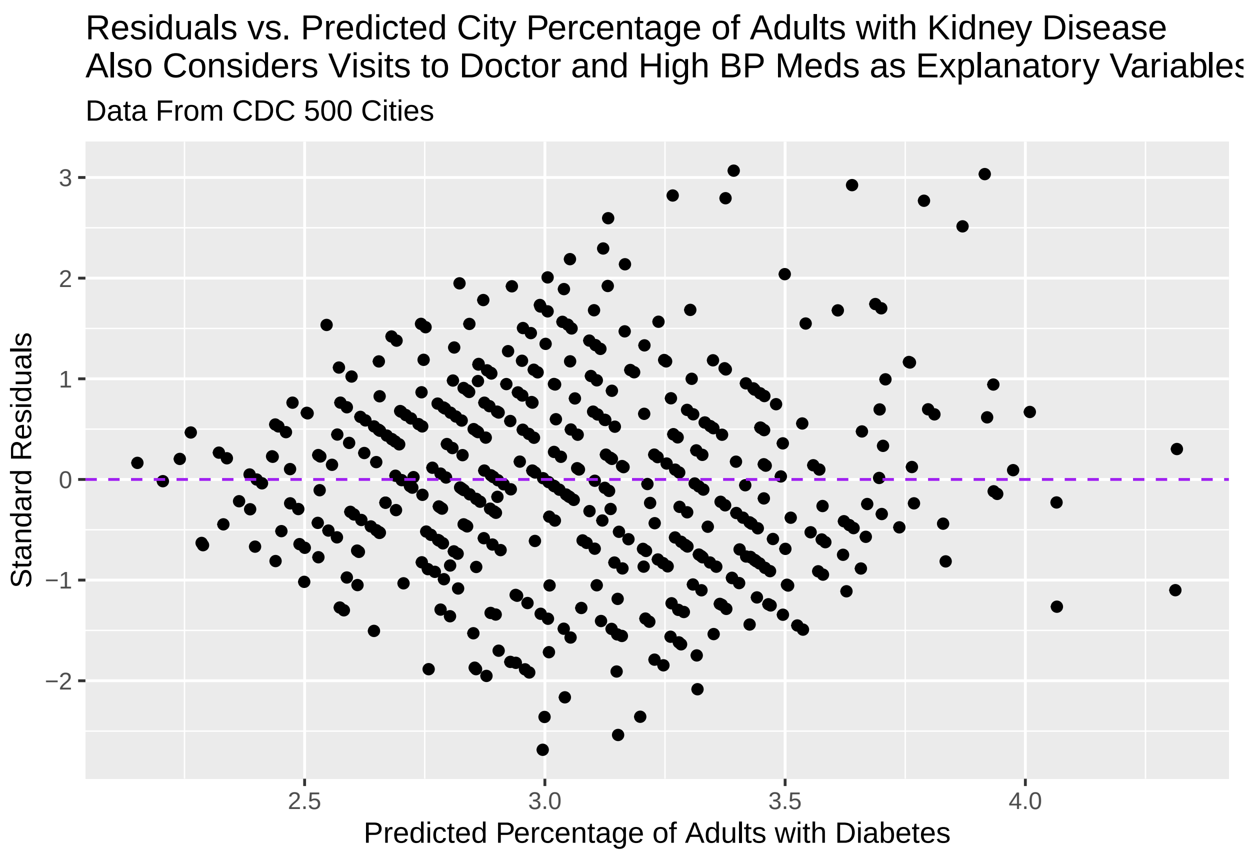
```
## [1] 0.6193093
```

The linear model with all possible second order interactions between the three explanatory variables is most appropriate because it has the highest R-squared value.

Displaying Graphs:

Residual Graph

```
int_access_kidney_disease_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted City Percentage of Adults with Kidney Disease
Also Considers Visits to Doctor and High BP Meds as Explanatory Variables (See Regression)",
    subtitle = "Data From CDC 500 Cities",
    x = "Predicted Percentage of Adults with Diabetes",
    y = "Standard Residuals"
  )
```



There seems to be a significant pattern in the residual plot, so a linear model would not be appropriate. Try a logistic model?

ANOVA Testing

Map Visualization

```
theme_set(theme_bw())
world <- ne_countries(scale = "medium", returnclass = "sf")
names(world)
```

```
##  [1] "scalerank"  "featurecla" "labelrank"  "sovereignt" "sov_a3"
```

```
##  [6] "adm0_dif"   "level"      "type"       "admin"      "adm0_a3"
## [11] "geou_dif"   "geounit"    "gu_a3"      "su_dif"     "subunit"
## [16] "su_a3"      "brk_diff"   "name"       "name_long"  "brk_a3"
## [21] "brk_name"   "brk_group"  "abbrev"     "postal"     "formal_en"
## [26] "formal_fr"  "note_adm0"  "note_brk"   "name_sort"  "name_alt"
## [31] "mapcolor7"  "mapcolor8"  "mapcolor9"  "mapcolor13" "pop_est"
## [36] "gdp_md_est" "pop_year"   "lastcensus" "gdp_year"   "economy"
## [41] "income_grp" "wikipedia"  "fips_10"    "iso_a2"     "iso_a3"
## [46] "iso_n3"     "un_a3"      "wb_a2"      "wb_a3"      "woe_id"
## [51] "adm0_a3_is" "adm0_a3_us" "adm0_a3_un" "adm0_a3_wb" "continent"
## [56] "region_un"  "subregion"  "region_wb"  "name_len"   "long_len"
## [61] "abbrev_len" "tiny"       "homepart"   "geometry"
```

```
state.name
```

```
##  [1] "Alabama"        "Alaska"         "Arizona"        "Arkansas"
##  [5] "California"     "Colorado"       "Connecticut"    "Delaware"
##  [9] "Florida"        "Georgia"        "Hawaii"         "Idaho"
## [13] "Illinois"       "Indiana"        "Iowa"           "Kansas"
## [17] "Kentucky"       "Louisiana"      "Maine"          "Maryland"
## [21] "Massachusetts"  "Michigan"       "Minnesota"      "Mississippi"
## [25] "Missouri"       "Montana"        "Nebraska"       "Nevada"
## [29] "New Hampshire"  "New Jersey"     "New Mexico"     "New York"
## [33] "North Carolina" "North Dakota"   "Ohio"           "Oklahoma"
## [37] "Oregon"         "Pennsylvania"   "Rhode Island"   "South Carolina"
## [41] "South Dakota"   "Tennessee"      "Texas"          "Utah"
## [45] "Vermont"        "Virginia"       "Washington"     "West Virginia"
## [49] "Wisconsin"      "Wyoming"
```

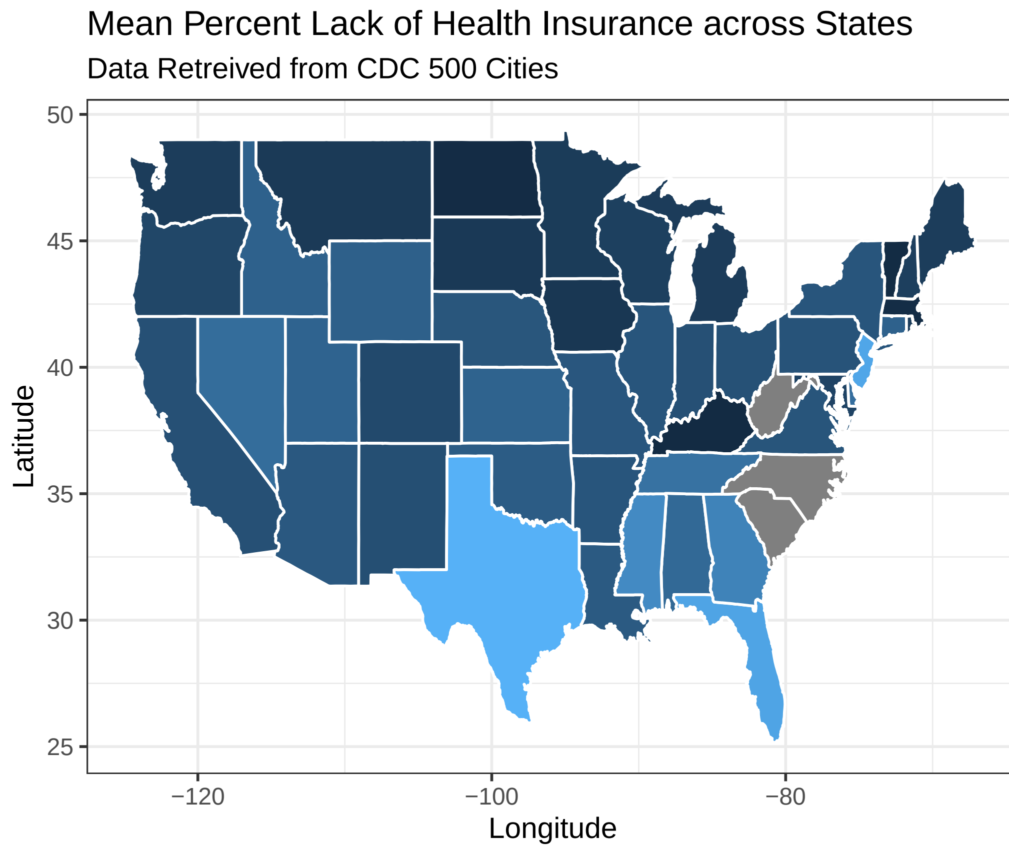
```
head(world)
```

```
## Simple feature collection with 6 features and 63 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -70.06611 ymin: -18.01973 xmax: 74.89131 ymax: 60.40581
## CRS:           +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
##   scalerank      featurecla labelrank     sovereignt sov_a3 adm0_dif level
## 0         3 Admin-0 country         5    Netherlands    NL1        1     2
## 1         1 Admin-0 country         3    Afghanistan    AFG        0     2
## 2         1 Admin-0 country         3         Angola    AGO        0     2
## 3         1 Admin-0 country         6 United Kingdom    GB1        1     2
## 4         1 Admin-0 country         6        Albania    ALB        0     2
## 5         3 Admin-0 country         6        Finland    FI1        1     2
##                type       admin adm0_a3 geou_dif     geounit gu_a3 su_dif
## 0           Country       Aruba     ABW        0       Aruba   ABW      0
## 1 Sovereign country Afghanistan     AFG        0 Afghanistan   AFG      0
## 2 Sovereign country      Angola     AGO        0      Angola   AGO      0
## 3        Dependency    Anguilla     AIA        0    Anguilla   AIA      0
## 4 Sovereign country     Albania     ALB        0     Albania   ALB      0
## 5           Country       Aland     ALD        0       Aland   ALD      0
##       subunit su_a3 brk_diff        name     name_long brk_a3    brk_name
## 0       Aruba   ABW        0       Aruba         Aruba    ABW       Aruba
## 1 Afghanistan   AFG        0 Afghanistan   Afghanistan    AFG Afghanistan
## 2      Angola   AGO        0      Angola        Angola    AGO      Angola
## 3    Anguilla   AIA        0    Anguilla      Anguilla    AIA    Anguilla
## 4     Albania   ALB        0     Albania       Albania    ALB     Albania
```

```
## 5       Aland   ALD        0       Aland Aland Islands    ALD       Aland
##   brk_group abbrev postal                    formal_en formal_fr note_adm0
## 0      <NA>  Aruba     AW                        Aruba      <NA>     Neth.
## 1      <NA>   Afg.     AF Islamic State of Afghanistan      <NA>      <NA>
## 2      <NA>   Ang.     AO  People's Republic of Angola      <NA>      <NA>
## 3      <NA>   Ang.     AI                         <NA>      <NA>      U.K.
## 4      <NA>   Alb.     AL          Republic of Albania      <NA>      <NA>
## 5      <NA>  Aland     AI                Åland Islands      <NA>      Fin.
##   note_brk   name_sort name_alt mapcolor7 mapcolor8 mapcolor9 mapcolor13
## 0     <NA>       Aruba     <NA>         4         2         2          9
## 1     <NA> Afghanistan     <NA>         5         6         8          7
## 2     <NA>      Angola     <NA>         3         2         6          1
## 3     <NA>    Anguilla     <NA>         6         6         6          3
## 4     <NA>     Albania     <NA>         1         4         1          6
## 5     <NA>       Aland     <NA>         4         1         4          6
##    pop_est gdp_md_est pop_year lastcensus gdp_year                   economy
## 0   103065     2258.0       NA       2010       NA      6. Developing region
## 1 28400000    22270.0       NA       1979       NA 7. Least developed region
## 2 12799293   110300.0       NA       1970       NA 7. Least developed region
## 3    14436      108.9       NA         NA       NA      6. Developing region
## 4  3639453    21810.0       NA       2001       NA      6. Developing region
## 5    27153     1563.0       NA         NA       NA 2. Developed region: nonG7
##                income_grp wikipedia fips_10 iso_a2 iso_a3 iso_n3 un_a3 wb_a2
## 0 2. High income: nonOECD        NA    <NA>     AW    ABW    533   533    AW
## 1           5. Low income        NA    <NA>     AF    AFG    004   004    AF
## 2  3. Upper middle income        NA    <NA>     AO    AGO    024   024    AO
## 3  3. Upper middle income        NA    <NA>     AI    AIA    660   660  <NA>
## 4  4. Lower middle income        NA    <NA>     AL    ALB    008   008    AL
## 5    1. High income: OECD        NA    <NA>     AX    ALA    248   248  <NA>
##   wb_a3 woe_id adm0_a3_is adm0_a3_us adm0_a3_un adm0_a3_wb     continent
## 0   ABW     NA        ABW        ABW         NA         NA North America
## 1   AFG     NA        AFG        AFG         NA         NA          Asia
## 2   AGO     NA        AGO        AGO         NA         NA        Africa
## 3  <NA>     NA        AIA        AIA         NA         NA North America
## 4   ALB     NA        ALB        ALB         NA         NA        Europe
## 5  <NA>     NA        ALA        ALD         NA         NA        Europe
##   region_un       subregion                 region_wb name_len long_len
## 0  Americas       Caribbean Latin America & Caribbean        5        5
## 1      Asia   Southern Asia                South Asia       11       11
## 2    Africa   Middle Africa        Sub-Saharan Africa        6        6
## 3  Americas       Caribbean Latin America & Caribbean        8        8
## 4    Europe Southern Europe     Europe & Central Asia        7        7
## 5    Europe Northern Europe     Europe & Central Asia        5       13
##   abbrev_len tiny homepart                       geometry
## 0          5    4       NA MULTIPOLYGON (((-69.89912 1...
## 1          4   NA        1 MULTIPOLYGON (((74.89131 37...
## 2          4   NA        1 MULTIPOLYGON (((14.19082 -5...
## 3          4   NA       NA MULTIPOLYGON (((-63.00122 1...
## 4          4   NA        1 MULTIPOLYGON (((20.06396 42...
## 5          5    5       NA MULTIPOLYGON (((20.61133 60...
```

```
states <- map_data("state")
states %>%
  mutate(StateDesc = str_to_title(region)) -> states
```

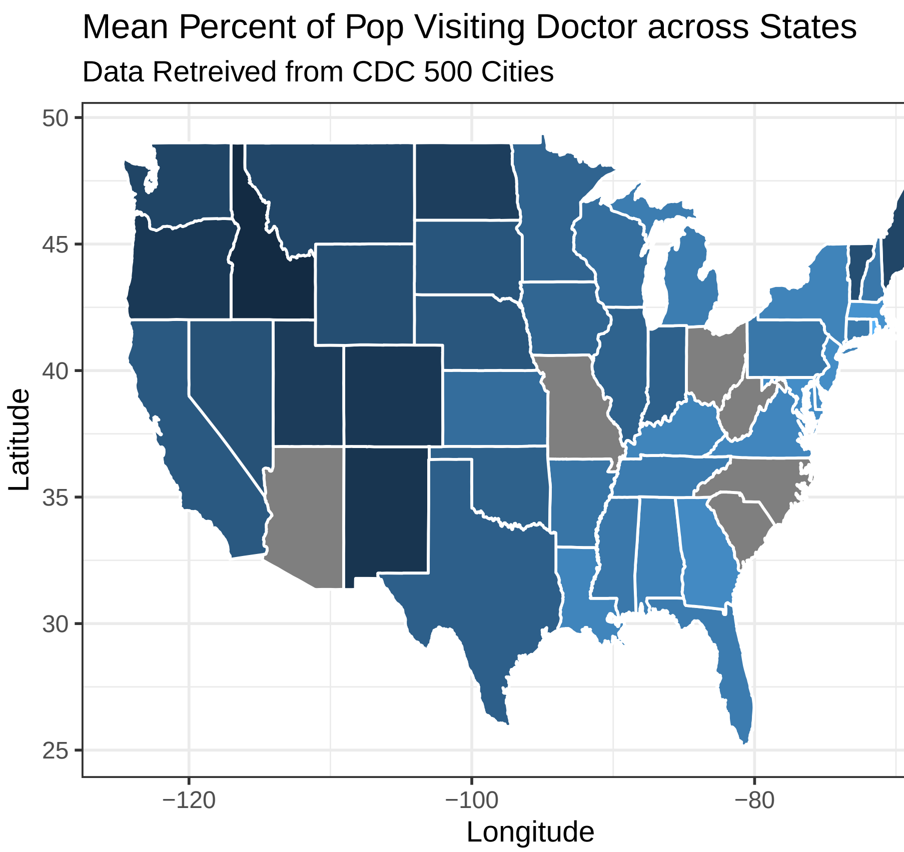
```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(insurance)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent No
Insurance",
       title = "Mean Percent Lack of Health Insurance across States",
subtitle = "Data Retreived from CDC 500 Cities")
```



State Map of Health Insurance

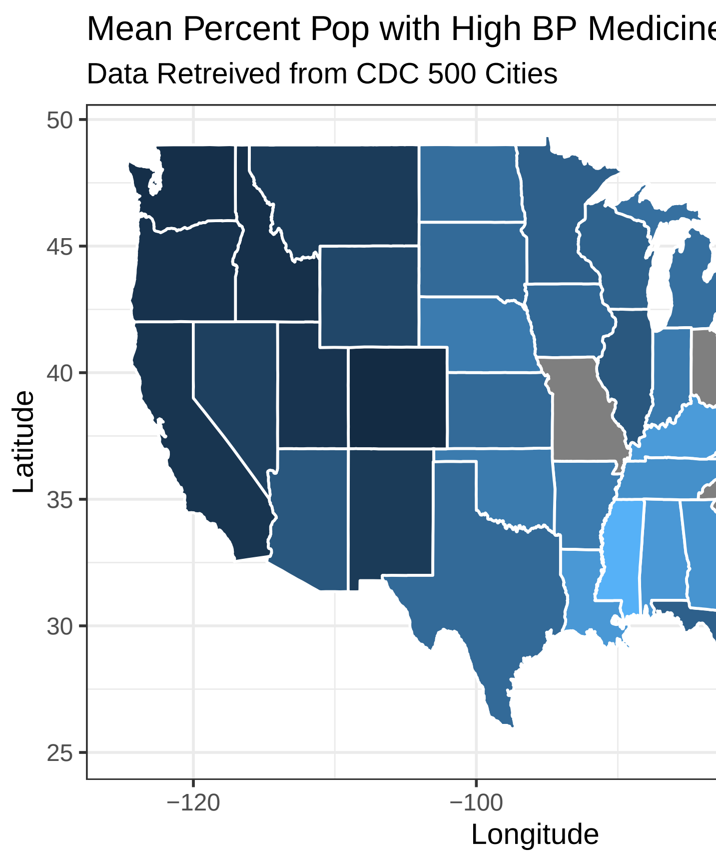
```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(visits_to_doctor)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
```

```r
  geom_polygon(color = "white") +
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent
Visiting Doctor",
       title = "Mean Percent of Pop Visiting Doctor across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



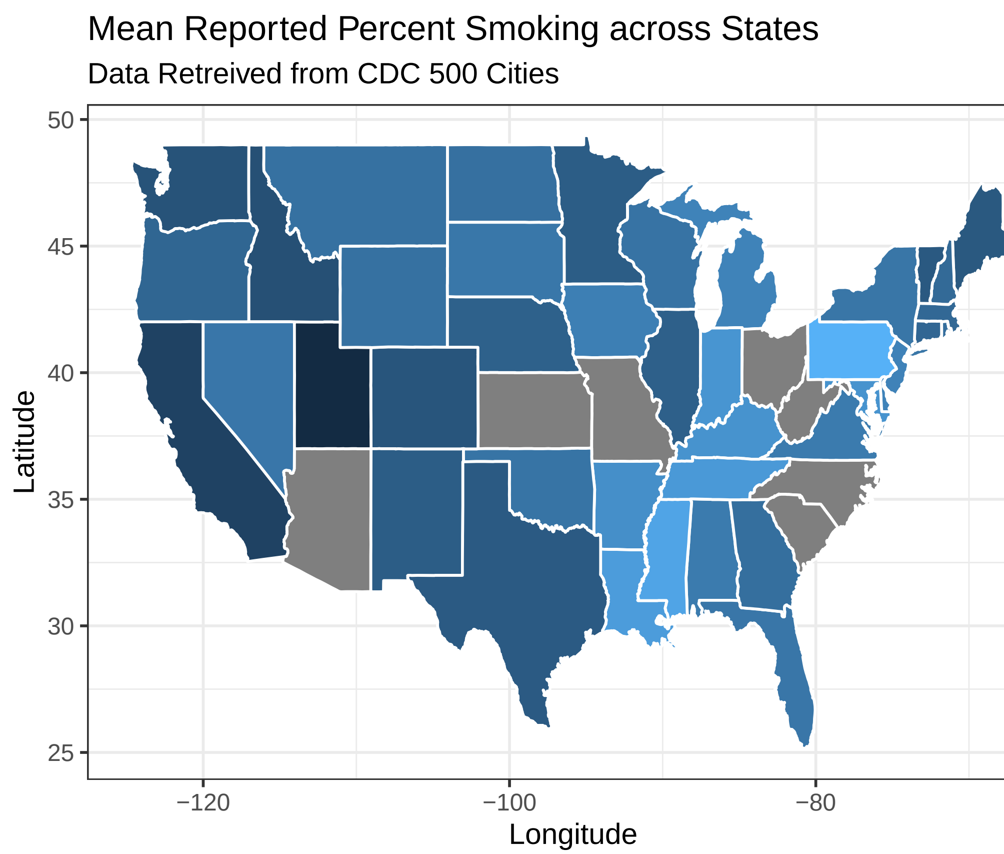
State Map of Visits to Doctor Variable

```r
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(medicine_high_bp)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent with
High BP Meds",
       title = "Mean Percent Pop with High BP Medicine across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



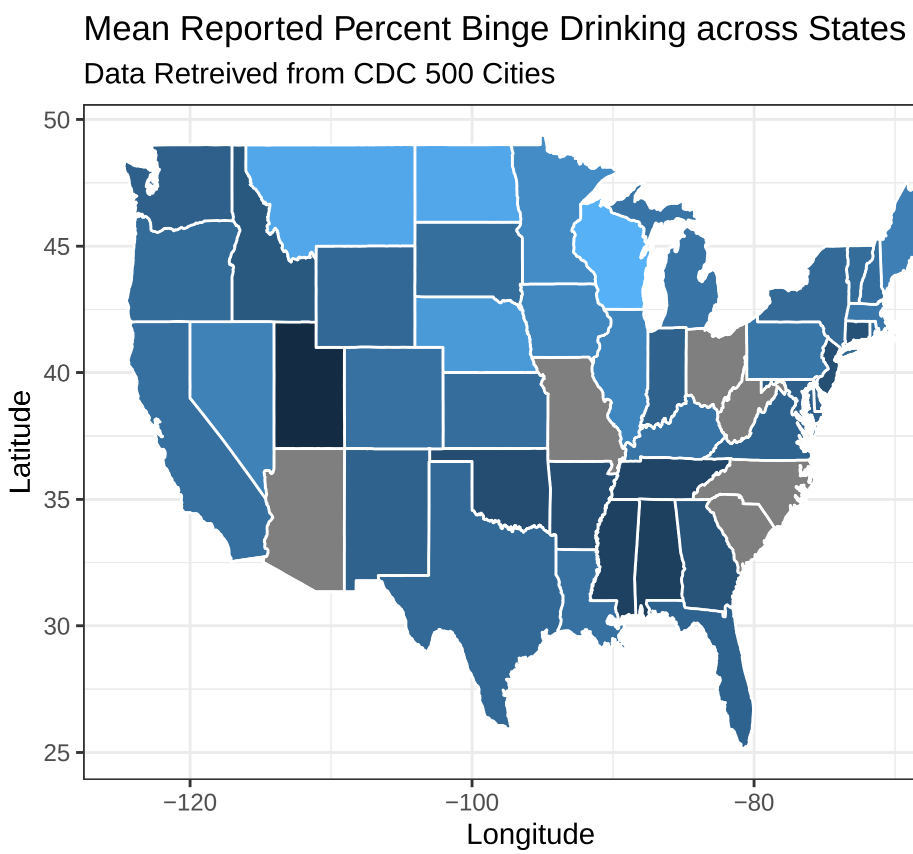
State Map of High Blood Pressure Medicine Variable

```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(smoking)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent
Smoking",
       title = "Mean Reported Percent Smoking across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



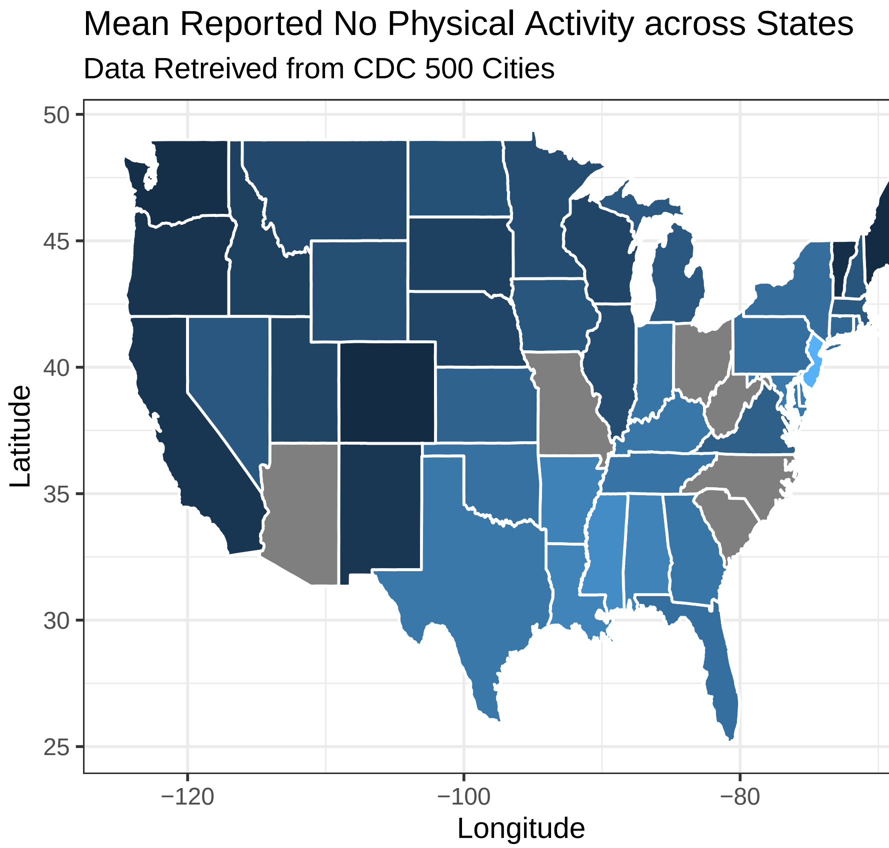
State Map of Smoking Variable

```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(binge_drinking)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent
Binge Drinking",
       title = "Mean Reported Percent Binge Drinking across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



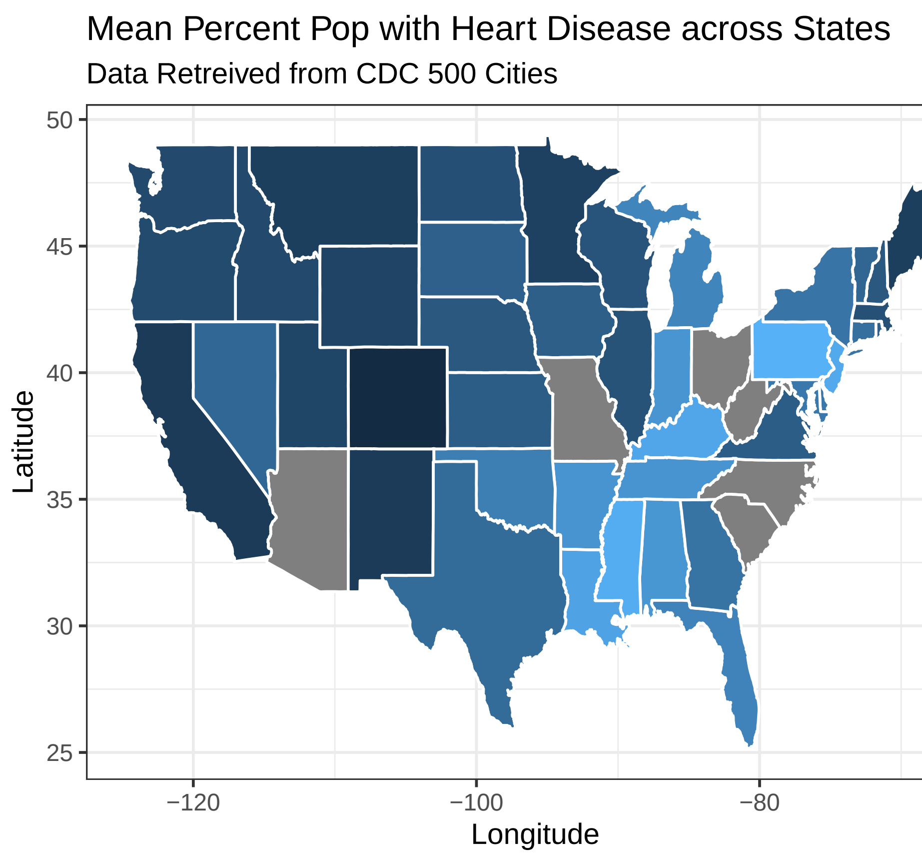
State Map of Binge Drinking Variable

```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(physical_activity)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent No
Physical Activity",
       title = "Mean Reported No Physical Activity across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



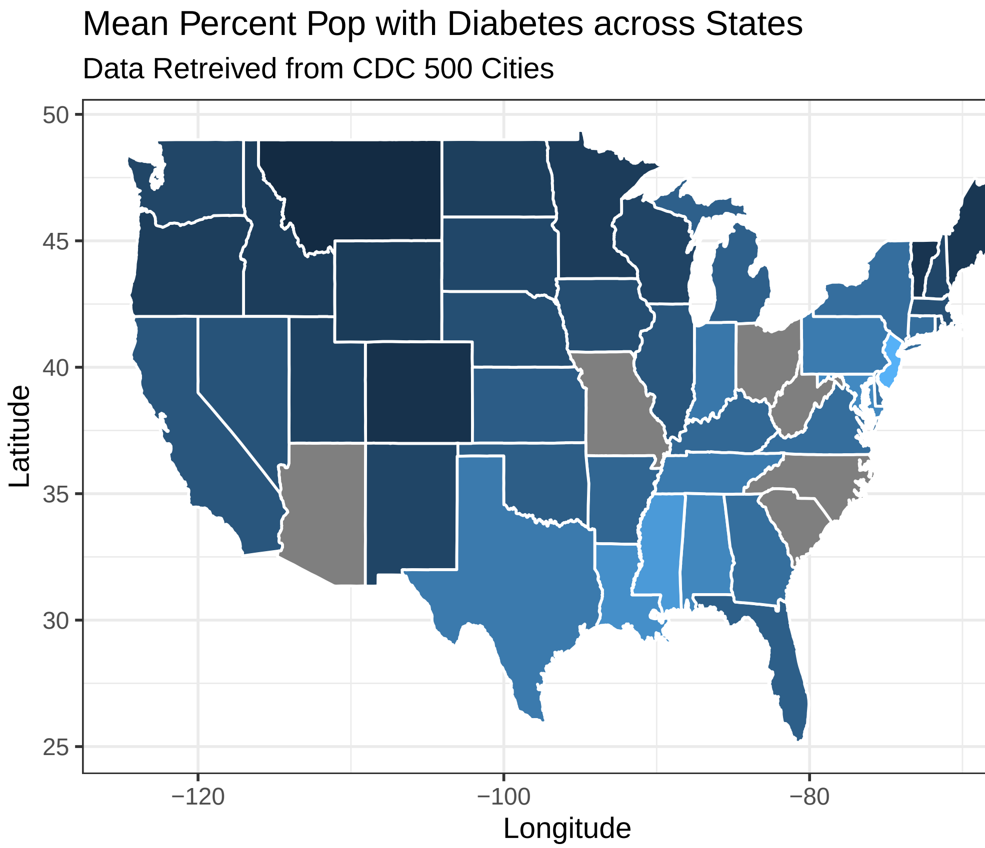
State Map of Physical Activity Variable

```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(heart_disease)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white") +
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent
Heart Disease",
       title = "Mean Percent Pop with Heart Disease across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



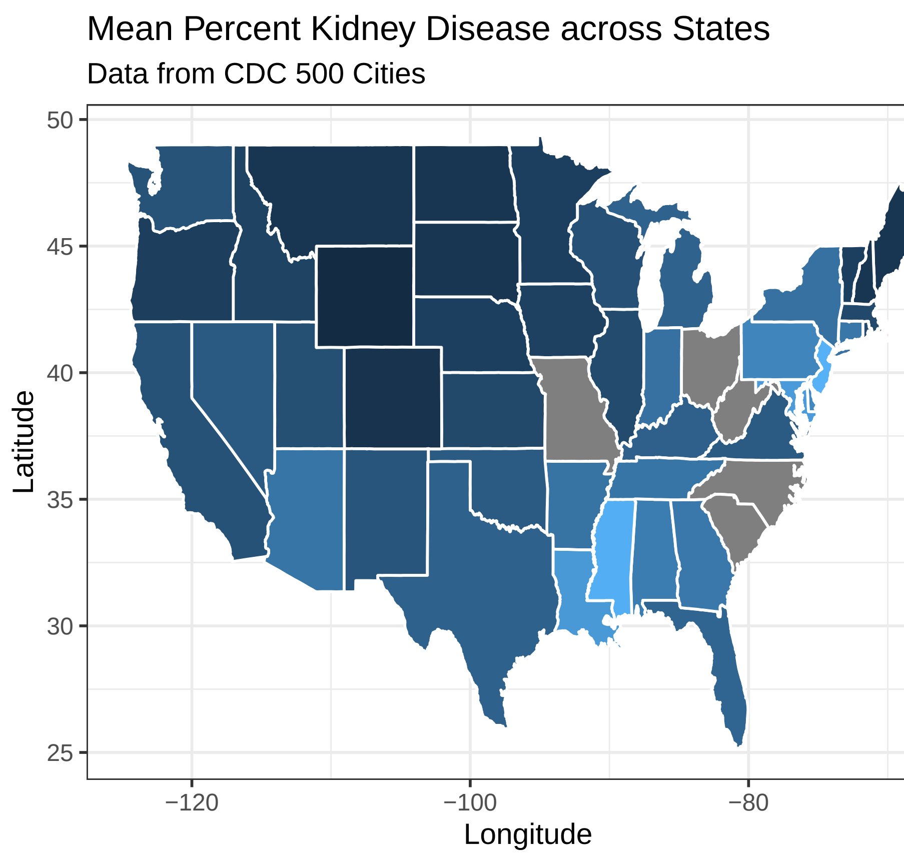
States Map of Heart Disease Variable

```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(diabetes)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent
Diabetes",
       title = "Mean Percent Pop with Diabetes across States",
       subtitle = "Data Retreived from CDC 500 Cities")
```



States Map of Diabetes Variable

```
states %>%
  left_join(data_500_cities, by = "StateDesc") -> test
data_500_cities %>%
  group_by(StateDesc) %>%
  summarise(test = mean(kidney_disease)) %>%
  right_join(states, by = "StateDesc") -> test2

ggplot(test2, aes(x = long, y = lat, group = group, fill = test)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       fill = "Percent
Kidney Disease",
       title = "Mean Percent Kidney Disease across States",
       subtitle = "Data from CDC 500 Cities")
```



States Map of Kidney Disease Variable

ANOVA Assumptions Visualizations

First, I will filter the data so that states with at least 10 city observations are present.

```
data_500_cities %>%
  group_by(StateDesc) %>%
  summarize(n = n()) %>%
  print(n = 51)
```

```
## # A tibble: 51 x 2
##    StateDesc          n
##    <chr>          <int>
##  1 Alabama            6
##  2 Alaska             1
##  3 Arizona           12
##  4 Arkansas           5
##  5 California       120
##  6 Colorado          12
##  7 Connecticut        7
##  8 Delaware           1
##  9 District of C      1
## 10 Florida           33
## 11 Georgia           10
## 12 Hawaii             1
## 13 Idaho              3
## 14 Illinois          15
## 15 Indiana           10
```

```
## 16 Iowa              6
## 17 Kansas            6
## 18 Kentucky          2
## 19 Louisiana         5
## 20 Maine             1
## 21 Maryland          1
## 22 Massachusetts    11
## 23 Michigan         16
## 24 Minnesota         6
## 25 Mississippi       2
## 26 Missouri          7
## 27 Montana           2
## 28 Nebraska          2
## 29 Nevada            5
## 30 New Hampshire     2
## 31 New Jersey        8
## 32 New Mexico        4
## 33 New York          7
## 34 North Carolin    10
## 35 North Dakota      1
## 36 Ohio              9
## 37 Oklahoma          6
## 38 Oregon            7
## 39 Pennsylvania      7
## 40 Rhode Island      4
## 41 South Carolin     4
## 42 South Dakota      2
## 43 Tennessee         6
## 44 Texas            46
## 45 United States     1
## 46 Utah              9
## 47 Vermont           1
## 48 Virginia         10
## 49 Washington       14
## 50 Wisconsin         7
## 51 Wyoming           1
```

Based on the table above, Arizona, California, Colorado, Florida, Georgia, Illinois, Indiana, Massachusetts, Michigan, North Carolina, Texas, Virginia, and Washington are the states with at least 10 observations. These states will be the only ones considered in the ANOVA test. Note that California has significantly more observations than all other states.

I will now create a new dataset that filters for the states and mutates new log versions of each variable.

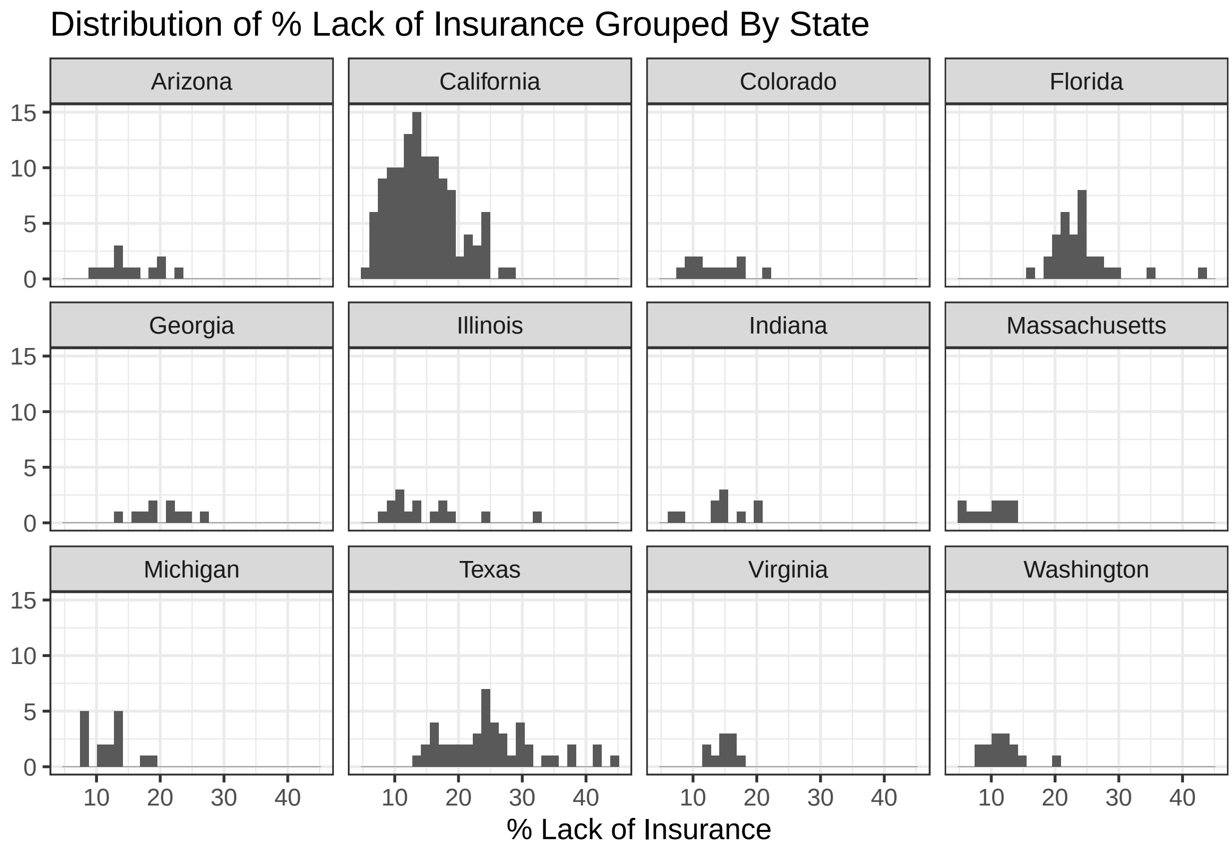
```
ANOVA_data_500_cities <- data_500_cities %>%
  filter(StateDesc %in% c("Arizona", "California", "Colorado", "Florida", "Georgia", "Illinois", "Indian
  mutate(linsruance = log(insurance)) %>%
  mutate(lvisits_to_doctor = log(visits_to_doctor)) %>%
  mutate(lmedicine_high_bp = log(medicine_high_bp)) %>%
  mutate(lheart_disease = log(heart_disease)) %>%
  mutate(ldiabetes = log(diabetes)) %>%
  mutate(lkidney_disease = log(kidney_disease))
```

Assumptions of the ANOVA Test are as Follows:

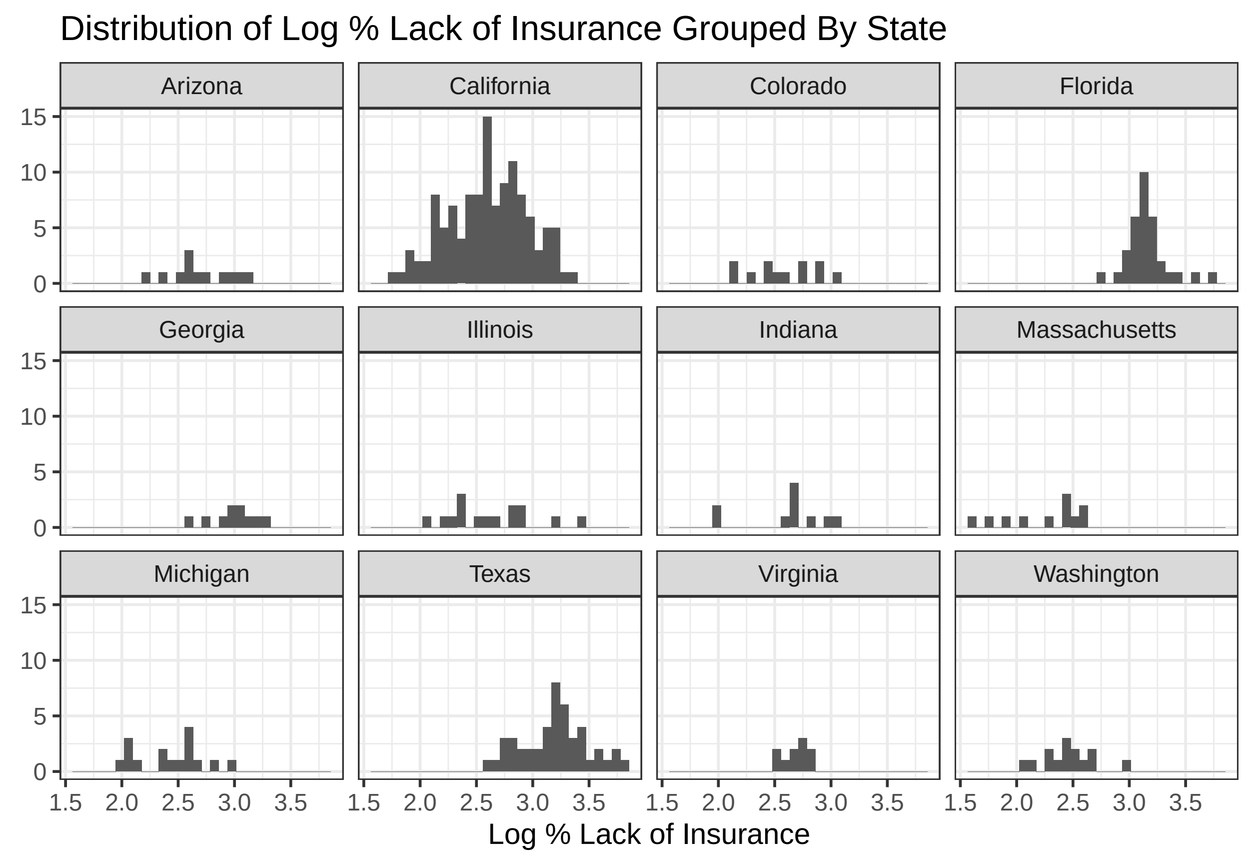
1) Outcomes within groups are normally distributed

2) Homoscedastic variance (same variance of individual observations in each group)
3) Samples are independent. This is likely not the case for this data, so we can compensate for this with a Bonferroni Correction or a Random Effects Model.

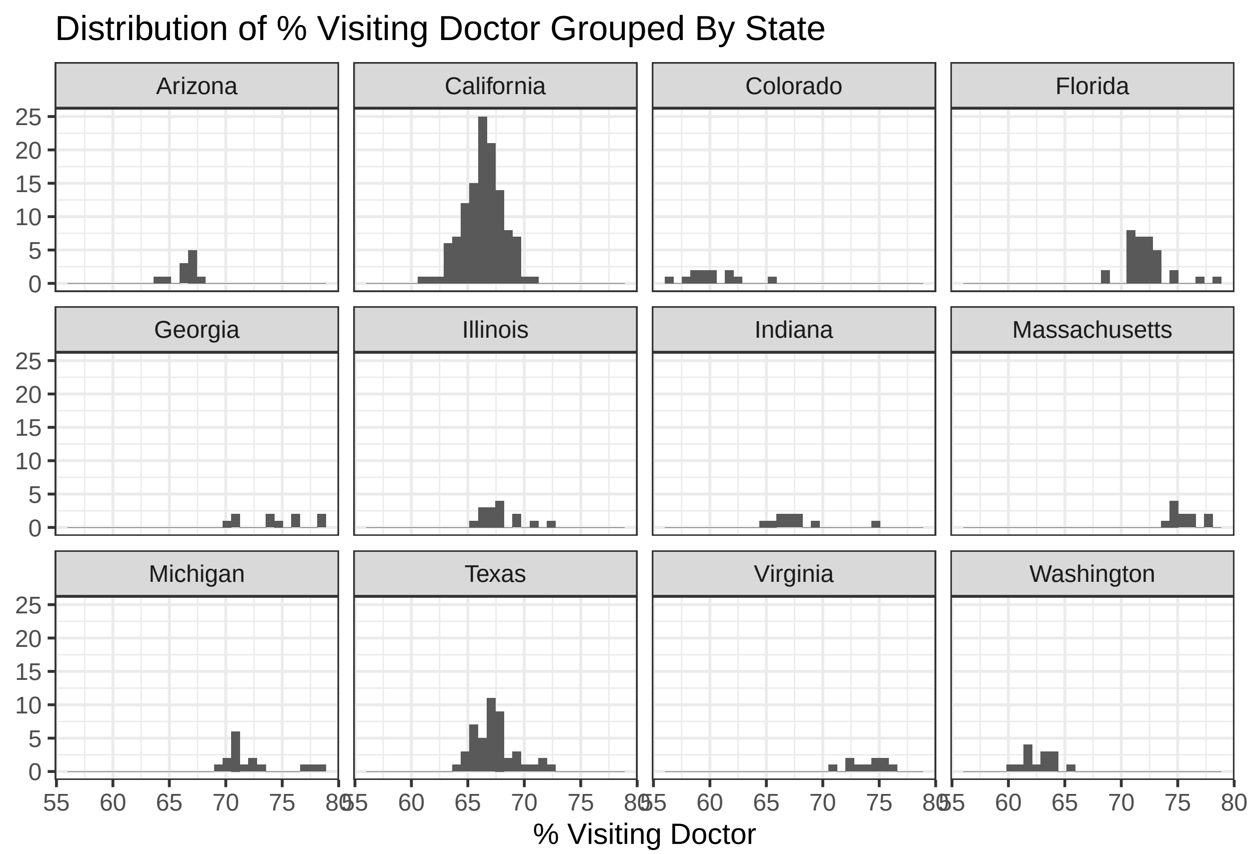
```
ANOVA_data_500_cities %>%
  ggplot(aes(x = insurance)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% Lack of Insurance", y = NULL, title = "Distribution of % Lack of Insurance Grouped By S
```



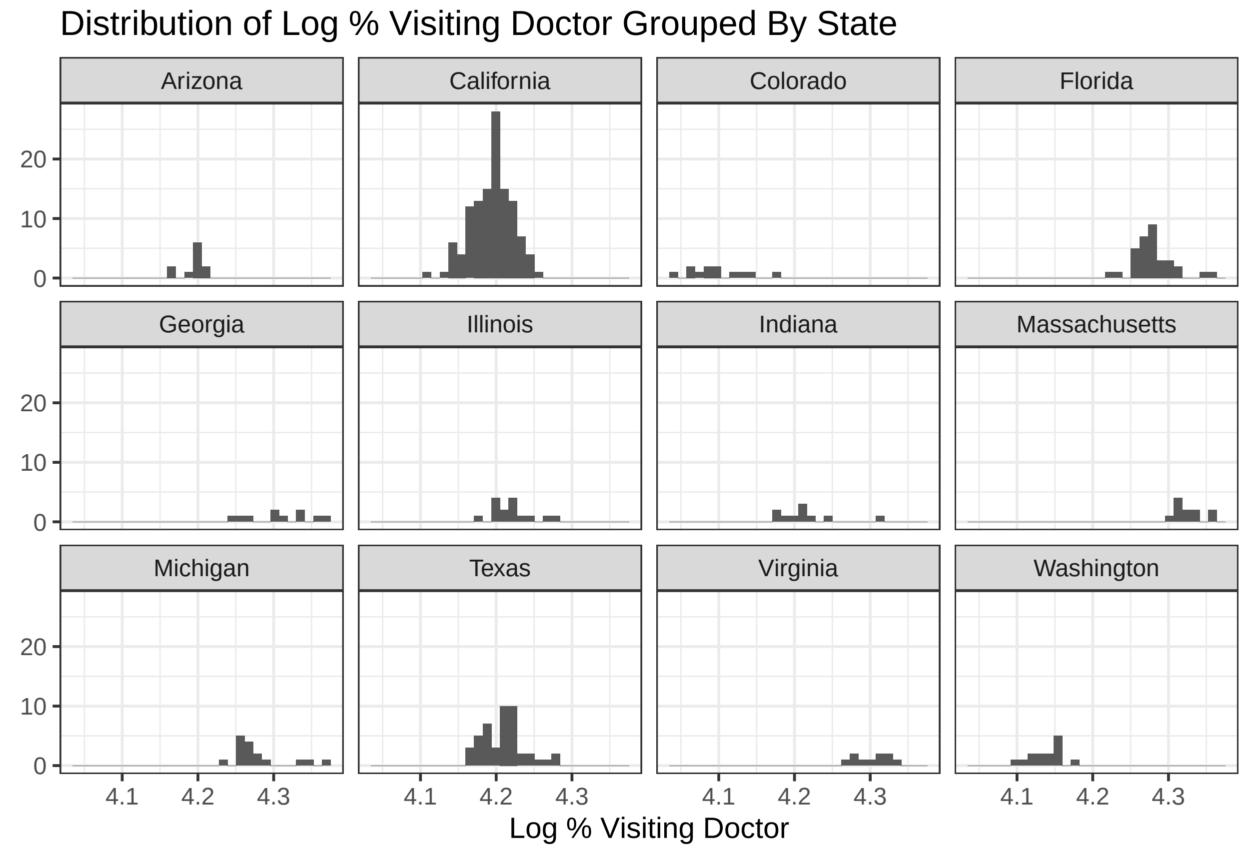

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = linsruance)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "Log % Lack of Insurance", y = NULL, title = "Distribution of Log % Lack of Insurance Grou
```



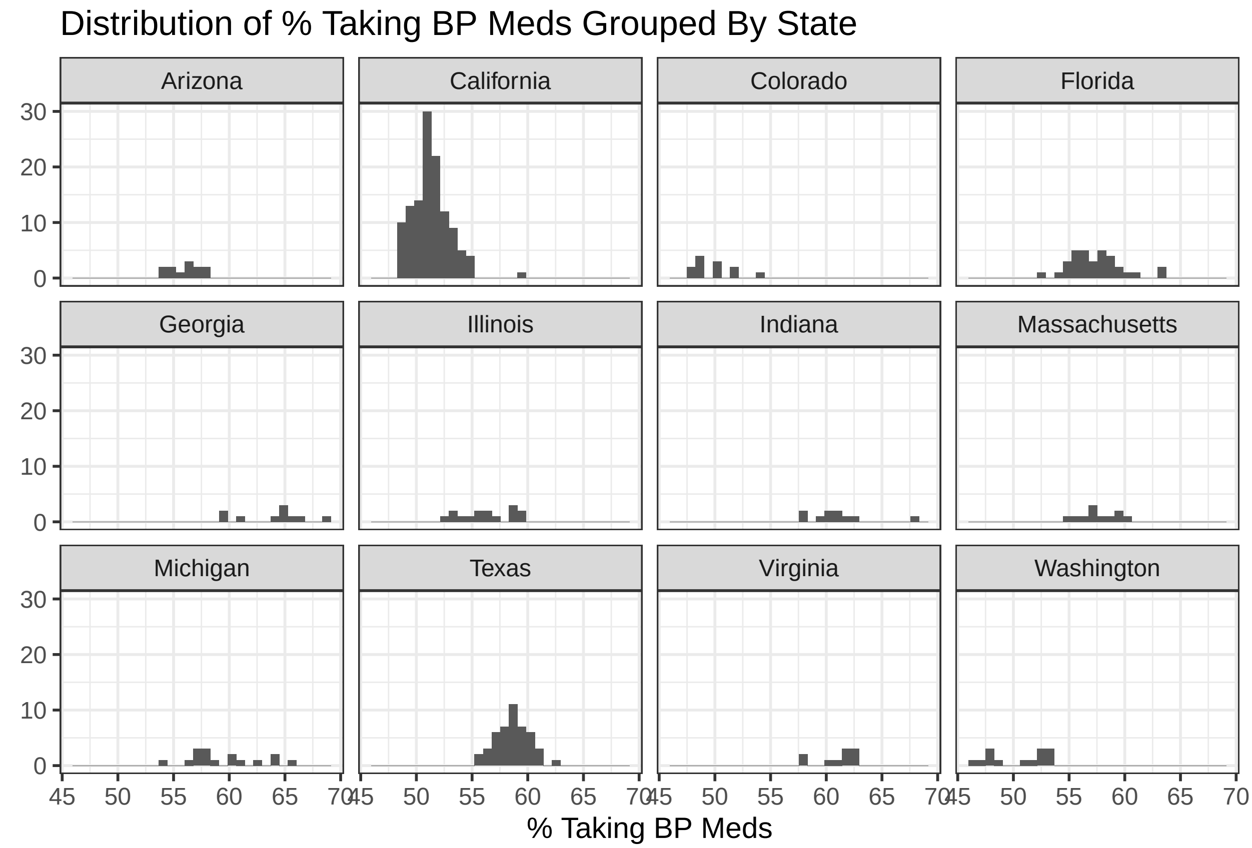

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = visits_to_doctor)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% Visiting Doctor", y = NULL, title = "Distribution of % Visiting Doctor Grouped By State
```



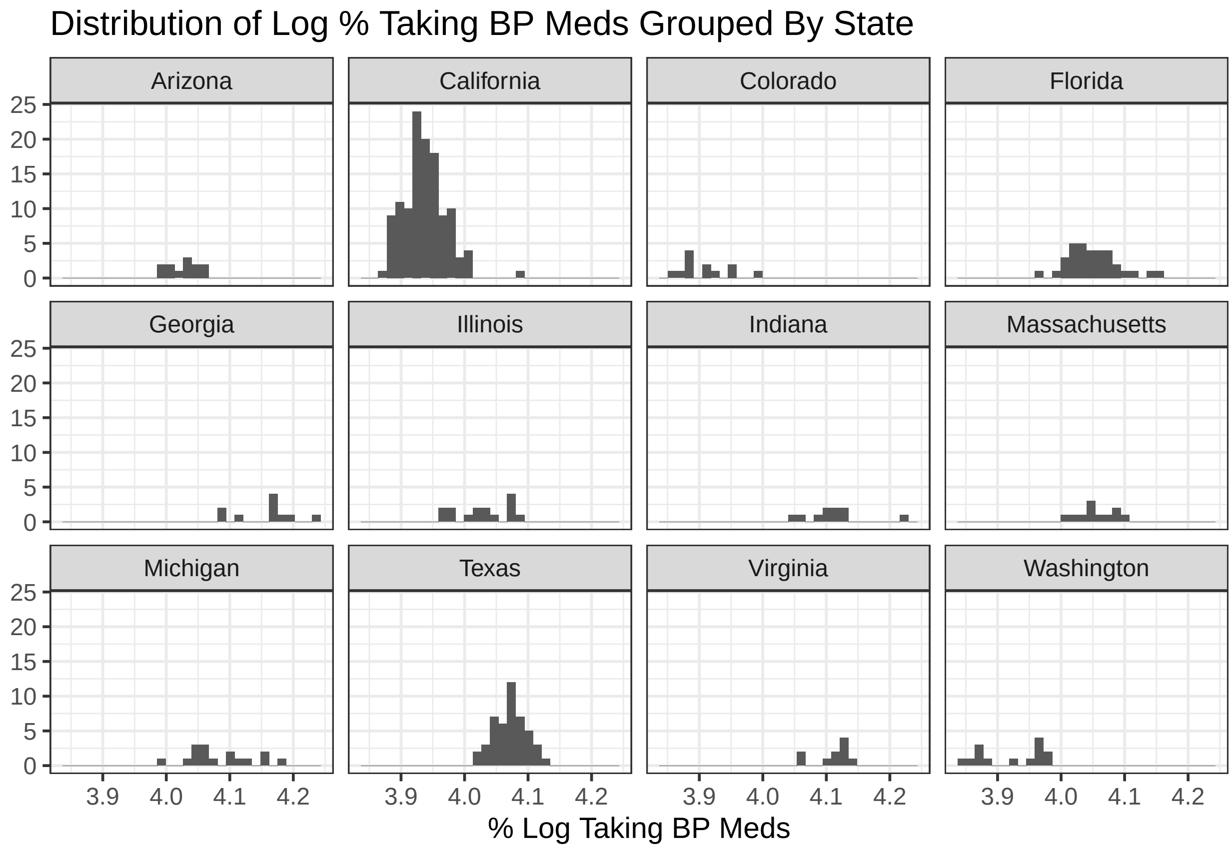

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = lvisits_to_doctor)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "Log % Visiting Doctor", y = NULL, title = "Distribution of Log % Visiting Doctor Grouped I
```



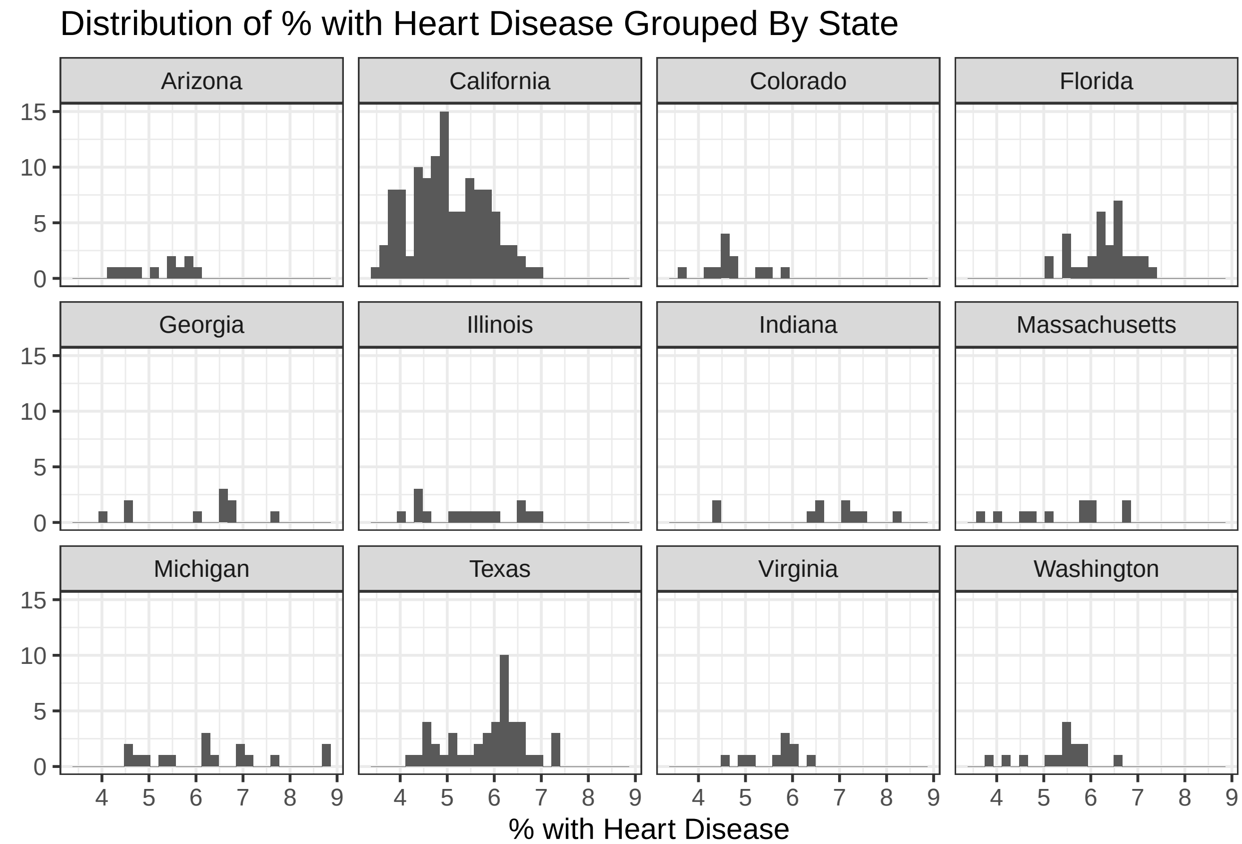

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = medicine_high_bp)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% Taking BP Meds", y = NULL, title = "Distribution of % Taking BP Meds Grouped By State")
```



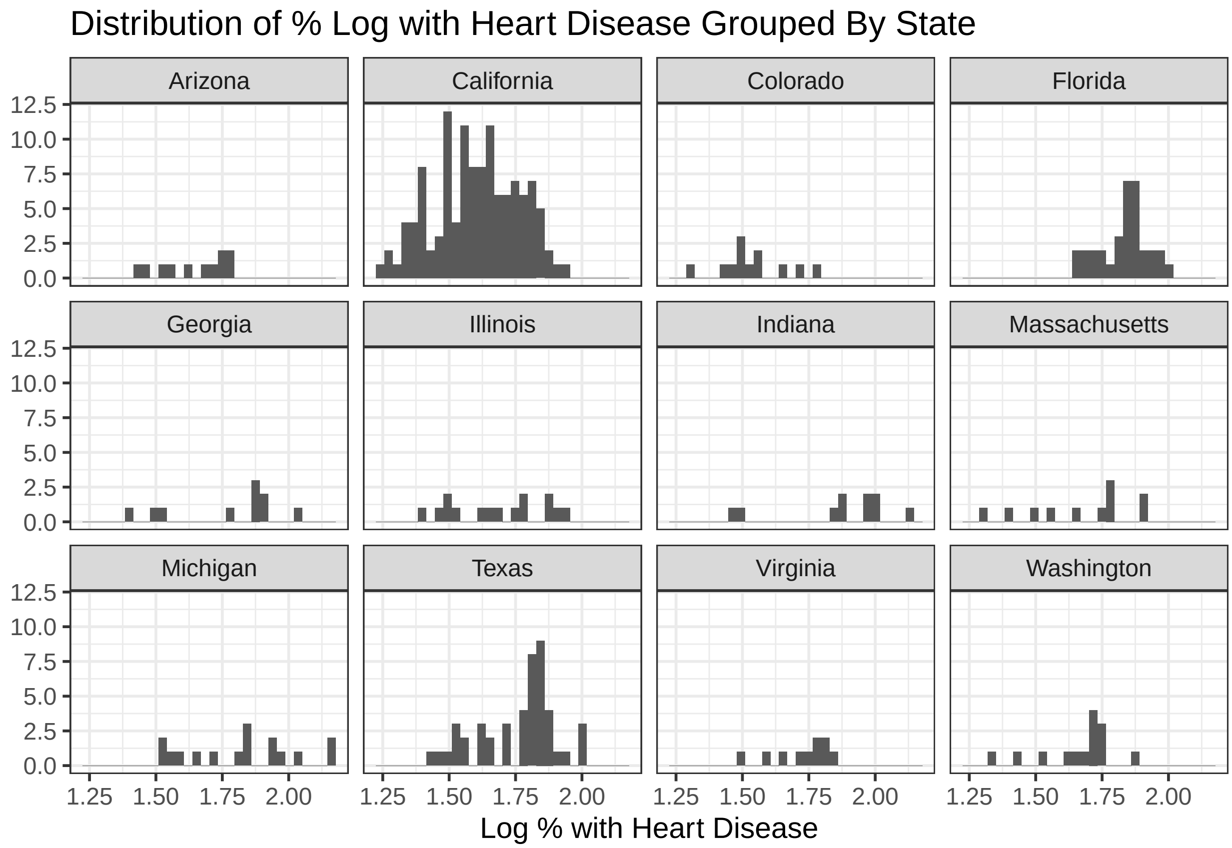

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = lmedicine_high_bp)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% Log Taking BP Meds", y = NULL, title = "Distribution of Log % Taking BP Meds Grouped By
```



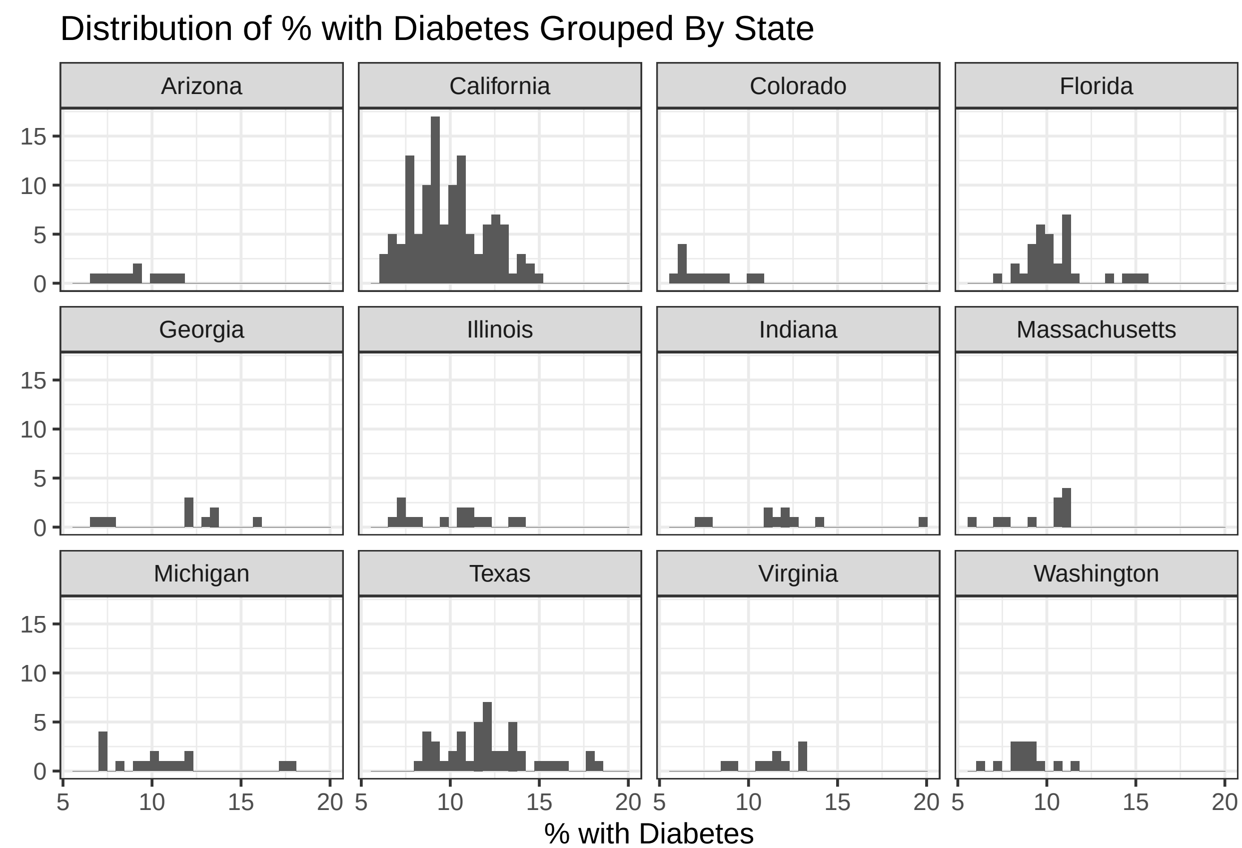

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = heart_disease)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% with Heart Disease", y = NULL, title = "Distribution of % with Heart Disease Grouped By
```



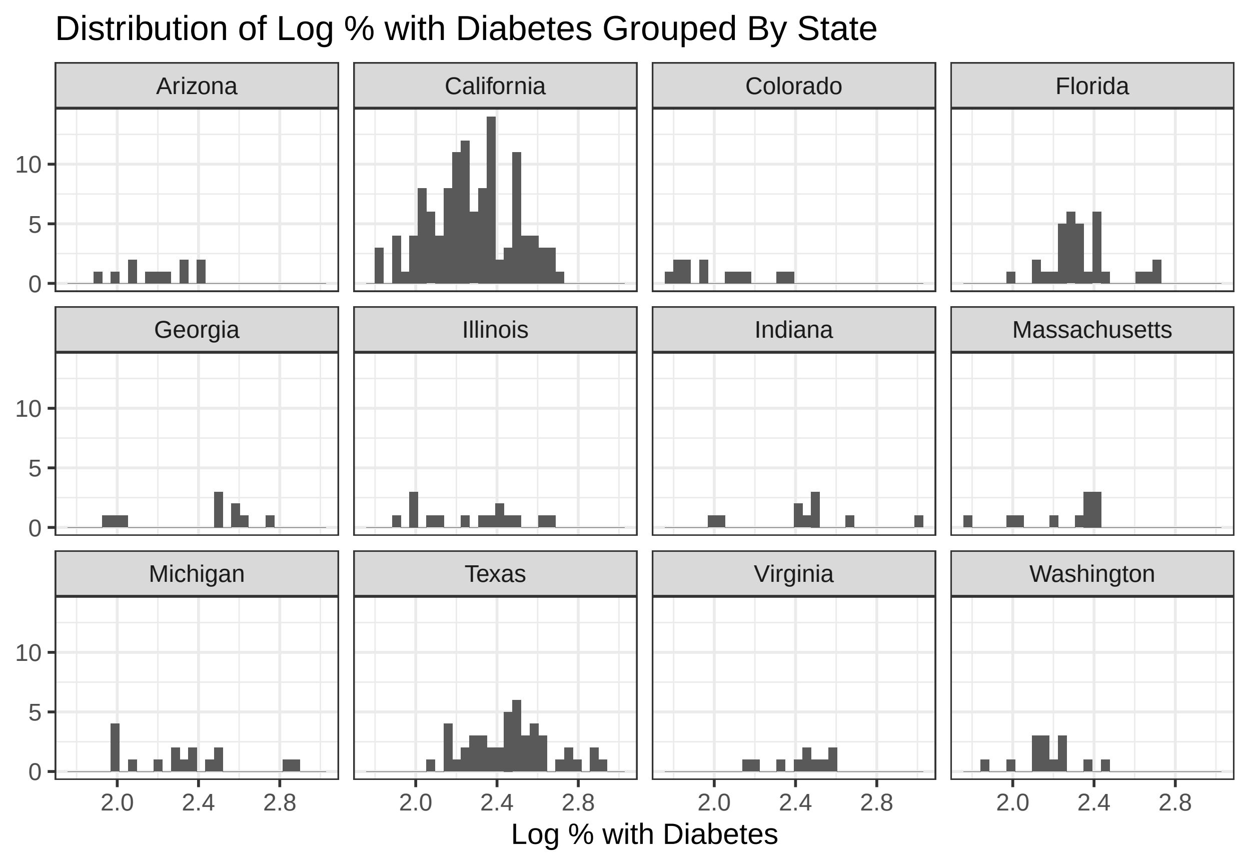

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = lheart_disease)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "Log % with Heart Disease", y = NULL, title = "Distribution of % Log with Heart Disease Gro
```



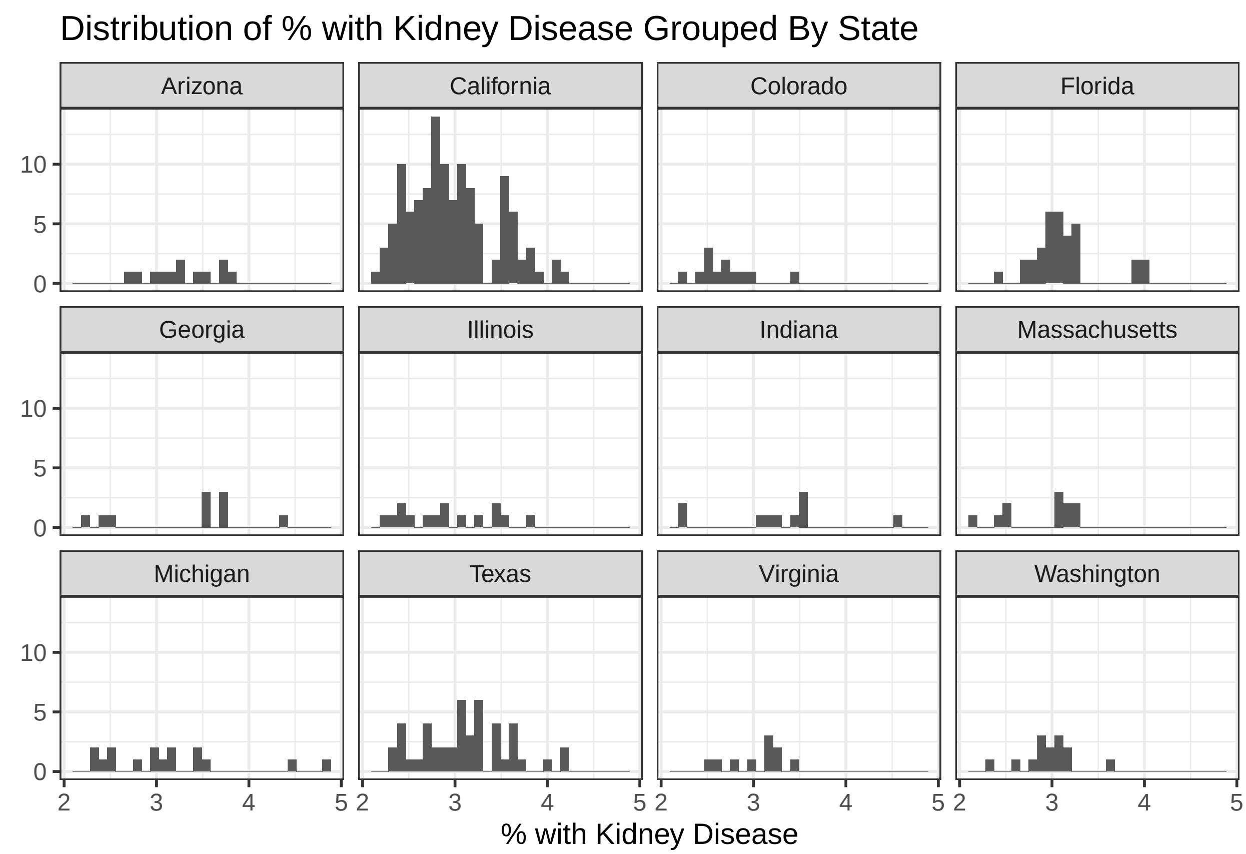

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = diabetes)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% with Diabetes", y = NULL, title = "Distribution of % with Diabetes Grouped By State")
```



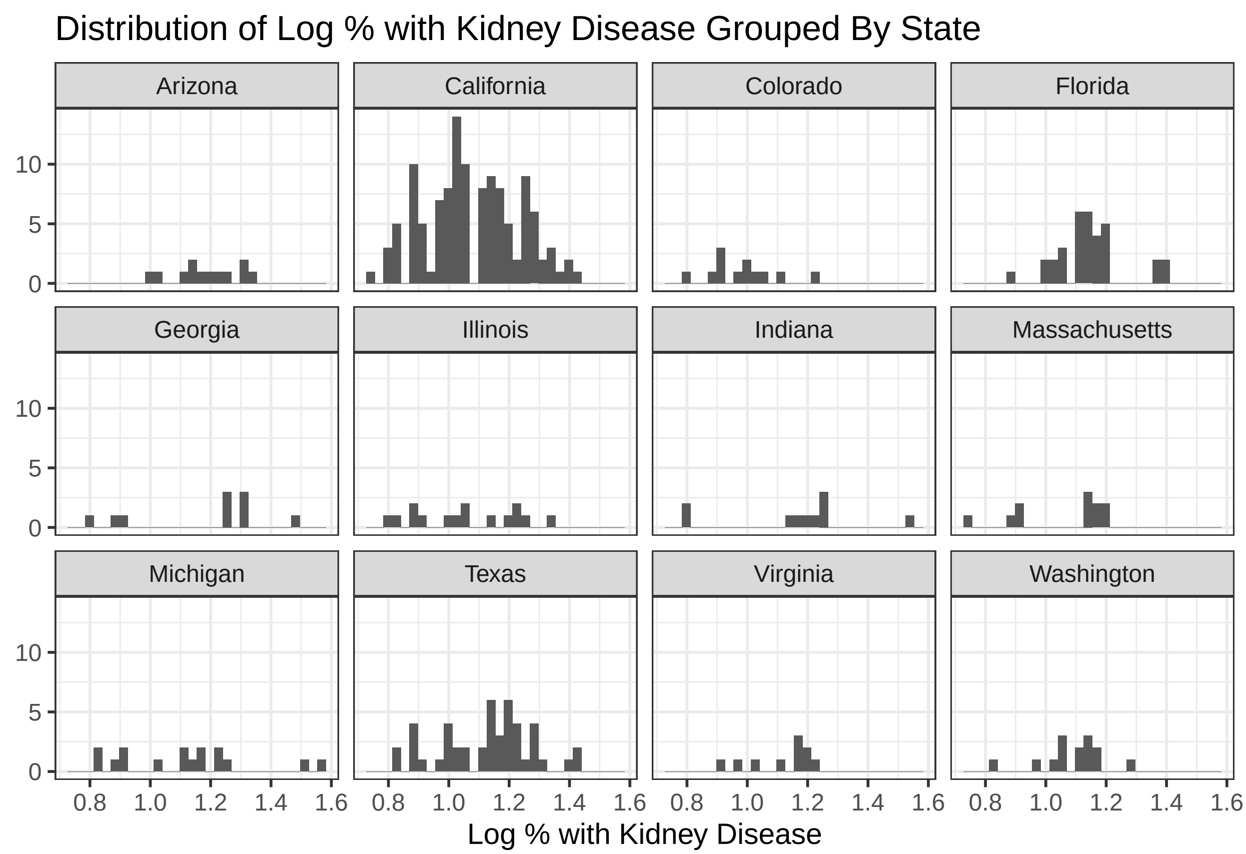

```
ANOVA_data_500_cities %>%
  ggplot(aes(x = ldiabetes)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "Log % with Diabetes", y = NULL, title = "Distribution of Log % with Diabetes Grouped By St
```




```
ANOVA_data_500_cities %>%
  ggplot(aes(x = kidney_disease)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "% with Kidney Disease", y = NULL, title = "Distribution of % with Kidney Disease Grouped I
```




```
ANOVA_data_500_cities %>%
  ggplot(aes(x = lkidney_disease)) +
    geom_histogram() +
    facet_wrap(~StateDesc) +
    labs(x = "Log % with Kidney Disease", y = NULL, title = "Distribution of Log % with Kidney Disease G
```



Based on the visuals above, the variables with the most normal distribution and variance within state groups are insurance, visits_to_doctor, lmedicine_high_bp, and lheartdisease. These will be the variables examined in the ANOVA testing. Since it is unlikely that there is independence in these tests, we will perform a Bonferroni Correction for the step down tests, where there are 78 tests being conducted at the same time (13 choose 2). The new significance level is 0.05/78 = 0.000641025641

Overall Tests

```
summary(aov(insurance~StateDesc,data=ANOVA_data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    11   7449   677.2    25.2 <2e-16 ***
## Residuals   297   7982    26.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(visits_to_doctor~StateDesc,data=ANOVA_data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    11   3780   343.6   90.03 <2e-16 ***
## Residuals   296   1130     3.8
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 1 observation deleted due to missingness
```

```r
summary(aov(lmedicine_high_bp~StateDesc,data=ANOVA_data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    11  1.627 0.14794   113.3 <2e-16 ***
## Residuals   297  0.388 0.00131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(aov(lheart_disease~StateDesc,data=ANOVA_data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## StateDesc    11  2.834 0.25760   10.33 5.44e-16 ***
## Residuals   296  7.384 0.02495
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 1 observation deleted due to missingness
```

All of these tests seem to demonstrate that there are significant differences across states.

Step Down Tests

```r
insurancepair <- pairwise.t.test(ANOVA_data_500_cities$insurance, ANOVA_data_500_cities$StateDesc, p.ad
siginsurancepairs <- broom::tidy(insurancepair) %>%
  filter(p.value<0.000641025641) %>%
  arrange(group1,group2)
nrow(siginsurancepairs)
```

```
## [1] 19
```

```r
print(siginsurancepairs, n= 19)
```

```
## # A tibble: 19 x 3
##    group1        group2          p.value
##    <chr>         <chr>             <dbl>
##  1 Florida       Arizona        6.58e- 5
##  2 Florida       California     1.06e-16
##  3 Florida       Colorado       3.73e- 7
##  4 Illinois      Florida        3.18e- 6
##  5 Indiana       Florida        3.05e- 5
##  6 Massachusetts Florida        6.47e-12
##  7 Massachusetts Georgia        3.01e- 4
##  8 Michigan      Florida        1.29e-11
##  9 Texas         Arizona        3.47e- 7
## 10 Texas         California     2.63e-26
## 11 Texas         Colorado       7.23e-10
## 12 Texas         Illinois       4.61e- 9
## 13 Texas         Indiana        2.23e- 7
## 14 Texas         Massachusetts  3.33e-15
## 15 Texas         Michigan       1.54e-15
## 16 Virginia      Florida        1.10e- 4
## 17 Virginia      Texas          1.01e- 6
## 18 Washington    Florida        1.82e-10
## 19 Washington    Texas          5.69e-14
```

```r
visits_to_doctor_pair <- pairwise.t.test(ANOVA_data_500_cities$visits_to_doctor, ANOVA_data_500_cities$S
sig_visits_to_doctor_pair <- broom::tidy(visits_to_doctor_pair) %>%
  filter(p.value<0.000641025641) %>%
  arrange(group1,group2)
nrow(sig_visits_to_doctor_pair)
```

```
## [1] 47
```

```r
print(sig_visits_to_doctor_pair, n= 47)
```

```
## # A tibble: 47 x 3
##    group1        group2         p.value
##    <chr>         <chr>            <dbl>
##  1 Colorado      Arizona       1.94e-11
##  2 Colorado      California    6.21e-20
##  3 Florida       Arizona       1.82e-14
##  4 Florida       California    1.61e-37
##  5 Florida       Colorado      2.15e-48
##  6 Georgia       Arizona       3.88e-17
##  7 Georgia       California    9.06e-28
##  8 Georgia       Colorado      6.20e-44
##  9 Illinois      Colorado      1.42e-19
## 10 Illinois      Florida       5.29e-10
## 11 Illinois      Georgia       4.05e-13
## 12 Indiana       Colorado      1.62e-15
## 13 Indiana       Florida       1.71e- 8
## 14 Indiana       Georgia       5.94e-12
## 15 Massachusetts Arizona       6.65e-23
## 16 Massachusetts California    2.55e-37
## 17 Massachusetts Colorado      2.35e-51
## 18 Massachusetts Florida       1.03e- 5
## 19 Massachusetts Illinois      7.98e-19
## 20 Massachusetts Indiana       7.27e-17
## 21 Michigan      Arizona       5.24e-12
## 22 Michigan      California    1.18e-23
## 23 Michigan      Colorado      7.93e-41
## 24 Michigan      Illinois      5.73e- 8
## 25 Michigan      Indiana       3.51e- 7
## 26 Michigan      Massachusetts 2.18e- 4
## 27 Texas         Colorado      2.41e-23
## 28 Texas         Florida       2.39e-21
## 29 Texas         Georgia       6.81e-20
## 30 Texas         Massachusetts 6.29e-28
## 31 Texas         Michigan      1.75e-14
## 32 Virginia      Arizona       5.79e-15
## 33 Virginia      California    1.91e-24
## 34 Virginia      Colorado      3.08e-41
## 35 Virginia      Illinois      5.01e-11
## 36 Virginia      Indiana       4.21e-10
## 37 Virginia      Texas         4.85e-17
## 38 Washington    Arizona       1.45e- 4
## 39 Washington    California    1.00e- 8
## 40 Washington    Florida       4.53e-37
## 41 Washington    Georgia       1.95e-34
```

```
## 42 Washington    Illinois      1.89e-10
## 43 Washington    Indiana       7.34e- 8
## 44 Washington    Massachusetts 7.31e-42
## 45 Washington    Michigan      2.51e-30
## 46 Washington    Texas         3.75e-12
## 47 Washington    Virginia      1.08e-31
```

```
lmedicine_high_bp_pair <- pairwise.t.test(ANOVA_data_500_cities$lmedicine_high_bp, ANOVA_data_500_cities
sig_lmedicine_high_bp_pair <- broom::tidy(lmedicine_high_bp_pair) %>%
  filter(p.value<0.000641025641) %>%
  arrange(group1,group2)
nrow(sig_lmedicine_high_bp_pair)
```

```
## [1] 39
```

```
print(sig_lmedicine_high_bp_pair, n= 39)
```

```
## # A tibble: 39 x 3
##    group1        group2         p.value
##    <chr>         <chr>            <dbl>
##  1 California    Arizona       1.34e-13
##  2 Colorado      Arizona       5.08e-13
##  3 Florida       California    1.45e-39
##  4 Florida       Colorado      3.58e-24
##  5 Georgia       Arizona       8.59e-14
##  6 Georgia       California    8.76e-50
##  7 Georgia       Colorado      1.21e-40
##  8 Georgia       Florida       1.45e-13
##  9 Illinois      California    1.52e-16
## 10 Illinois      Colorado      1.71e-14
## 11 Illinois      Georgia       8.23e-15
## 12 Indiana       Arizona       1.07e- 5
## 13 Indiana       California    2.19e-34
## 14 Indiana       Colorado      5.57e-29
## 15 Indiana       Florida       2.28e- 4
## 16 Indiana       Illinois      4.42e- 6
## 17 Massachusetts California    1.52e-19
## 18 Massachusetts Colorado      1.42e-17
## 19 Massachusetts Georgia       6.22e- 9
## 20 Michigan      California    1.22e-37
## 21 Michigan      Colorado      2.85e-28
## 22 Michigan      Georgia       2.38e- 5
## 23 Texas         California    1.48e-60
## 24 Texas         Colorado      1.51e-32
## 25 Texas         Georgia       1.63e- 9
## 26 Virginia      Arizona       7.61e- 6
## 27 Virginia      California    9.68e-35
## 28 Virginia      Colorado      3.04e-29
## 29 Virginia      Florida       1.59e- 4
## 30 Virginia      Illinois      3.05e- 6
## 31 Washington    Arizona       2.41e-11
## 32 Washington    Florida       1.28e-22
## 33 Washington    Georgia       1.63e-39
## 34 Washington    Illinois      8.54e-13
## 35 Washington    Indiana       1.62e-27
```

```
## 36 Washington    Massachusetts 6.65e-16
## 37 Washington    Michigan      8.14e-27
## 38 Washington    Texas         1.82e-31
## 39 Washington    Virginia      8.69e-28
```

```
lheart_disease_pair <- pairwise.t.test(ANOVA_data_500_cities$lheart_disease, ANOVA_data_500_cities$State
sig_lheart_disease_pair <- broom::tidy(lheart_disease_pair) %>%
  filter(p.value<0.000641025641) %>%
  arrange(group1,group2)
nrow(sig_lheart_disease_pair)
```

```
## [1] 7
```

```
print(sig_lheart_disease_pair, n= 7)
```

```
## # A tibble: 7 x 3
##   group1   group2      p.value
##   <chr>    <chr>         <dbl>
## 1 Florida  California 4.14e-11
## 2 Florida  Colorado   6.45e- 6
## 3 Indiana  California 5.22e- 5
## 4 Indiana  Colorado   2.31e- 4
## 5 Michigan California 1.71e- 5
## 6 Michigan Colorado   3.53e- 4
## 7 Texas    California 7.75e- 7
```