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

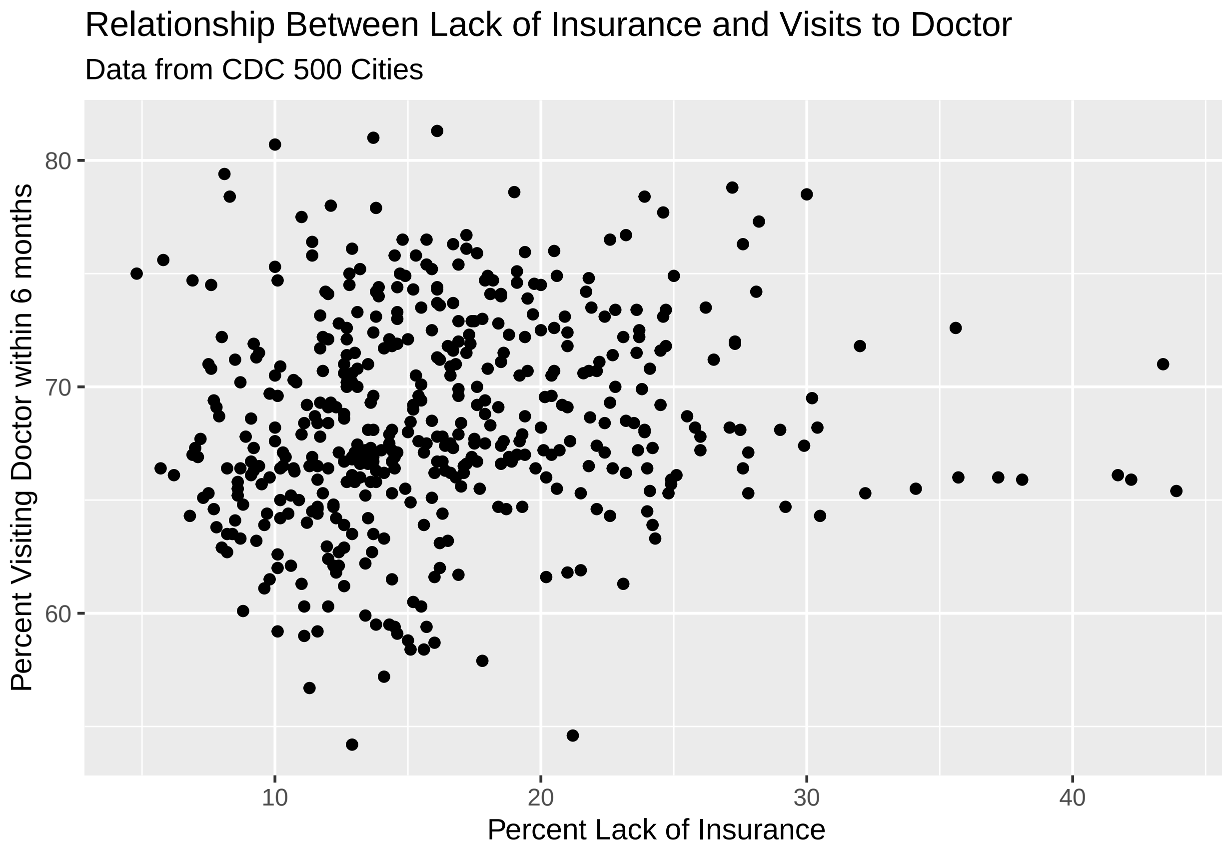
Linear Regressions

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

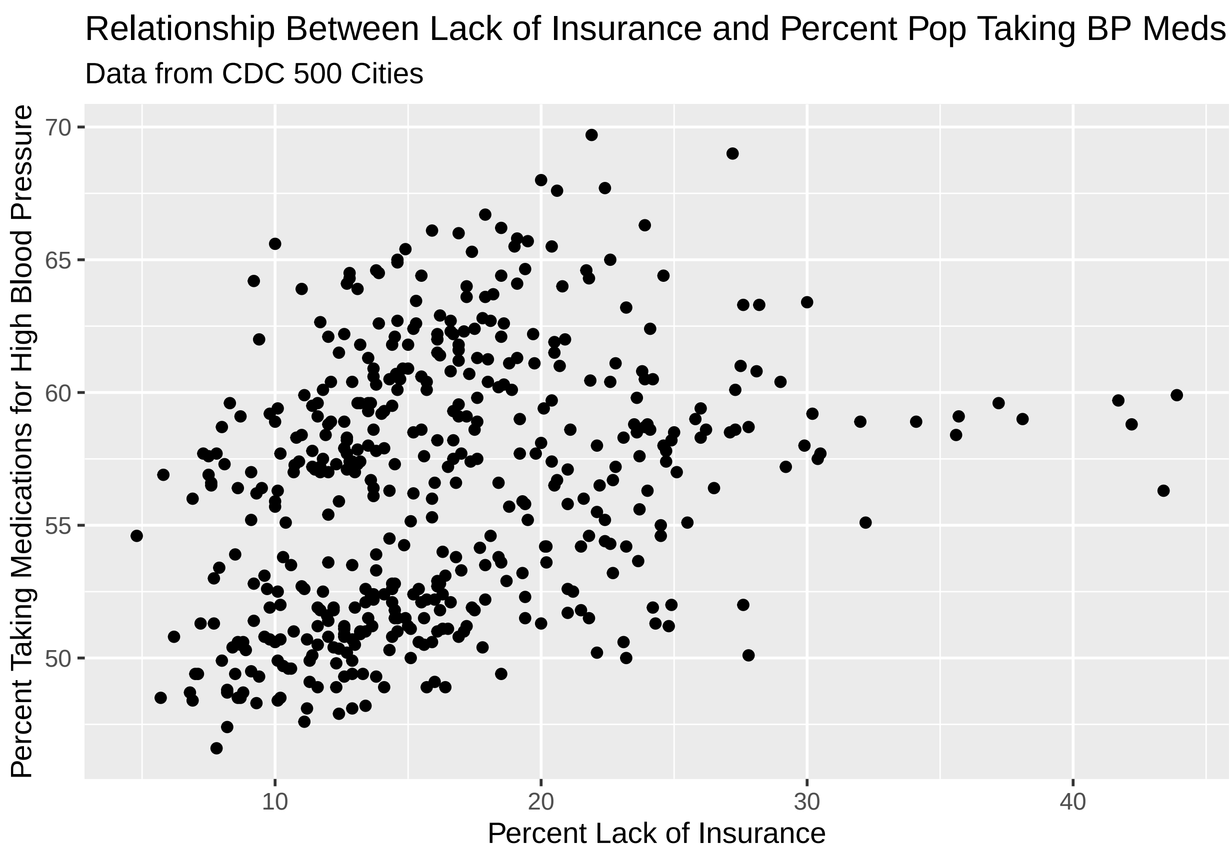
Correlations between Explanatory Variables

```
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"
  )
```



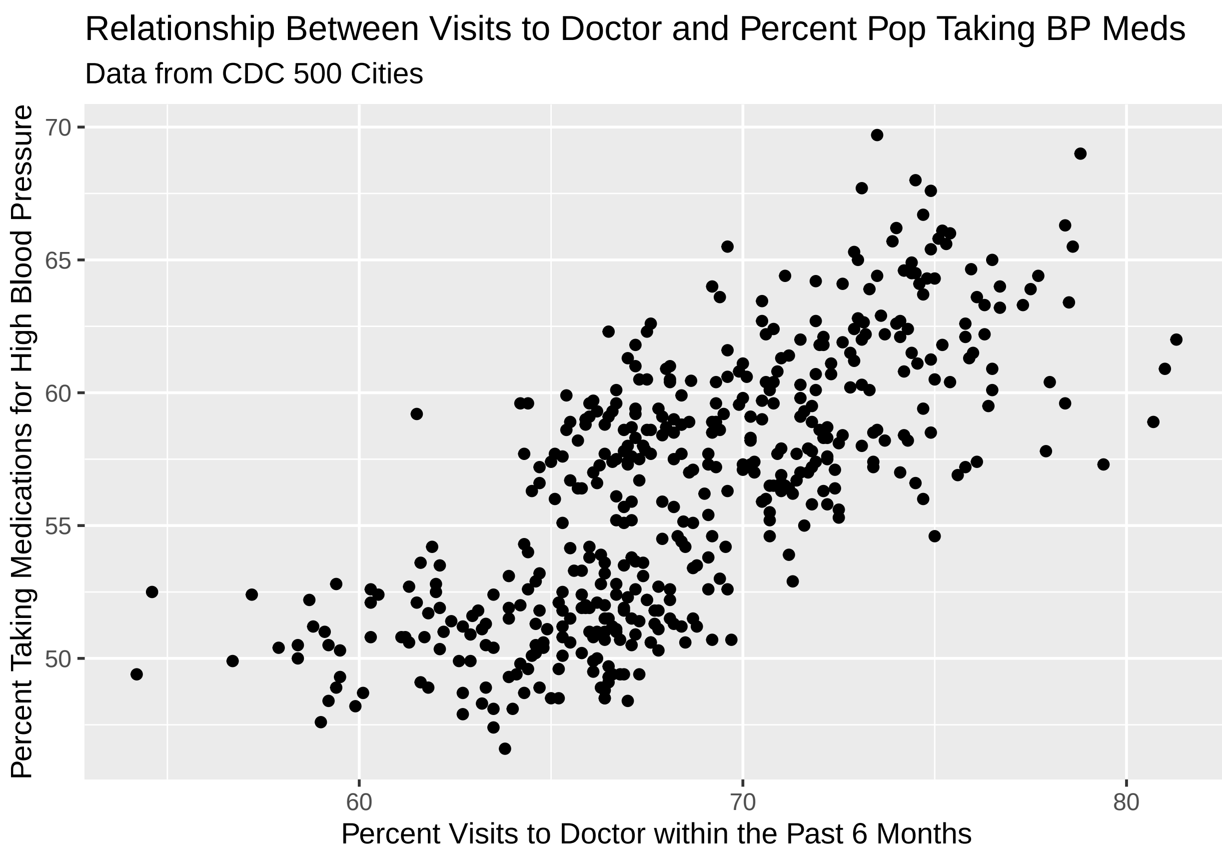
There does not seem to be any significant correlation.

```
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:

```r
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

```r
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:

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

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

Adj R-Squared Value with One Interactions:

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

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

Adj R-Squared Value with All Interactions:

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

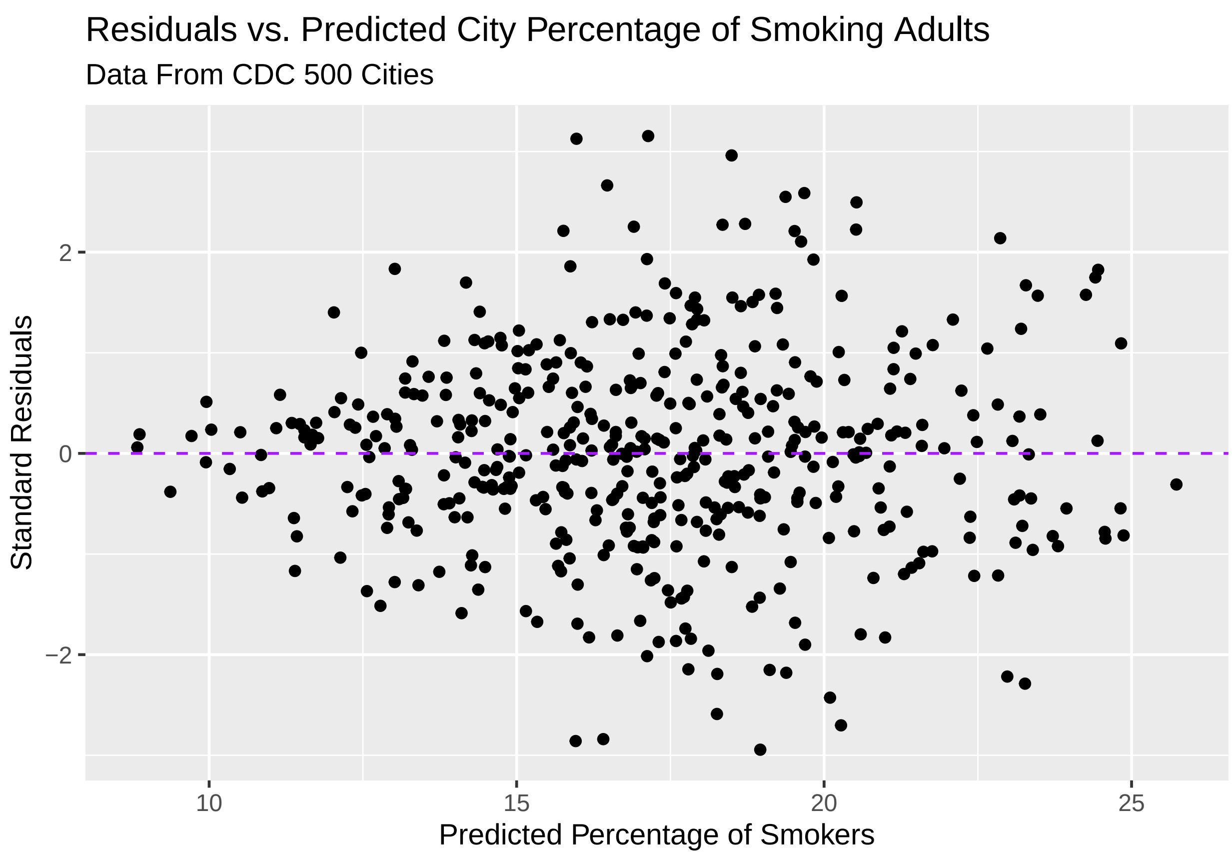
```
## [1] 0.5691301
```

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"
  )
```

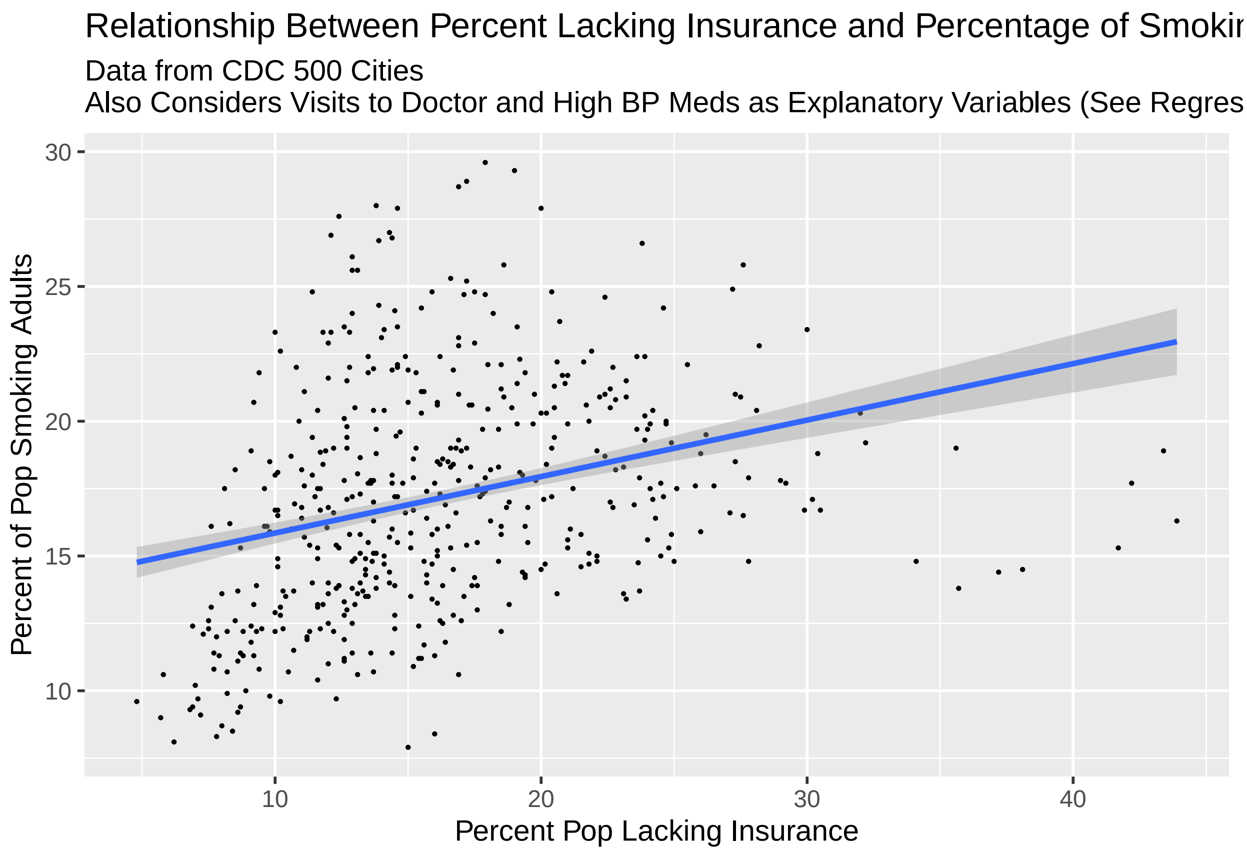


There does not seem to be any patterns in this residual graph, so a linear model would be appropriate.

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()
```

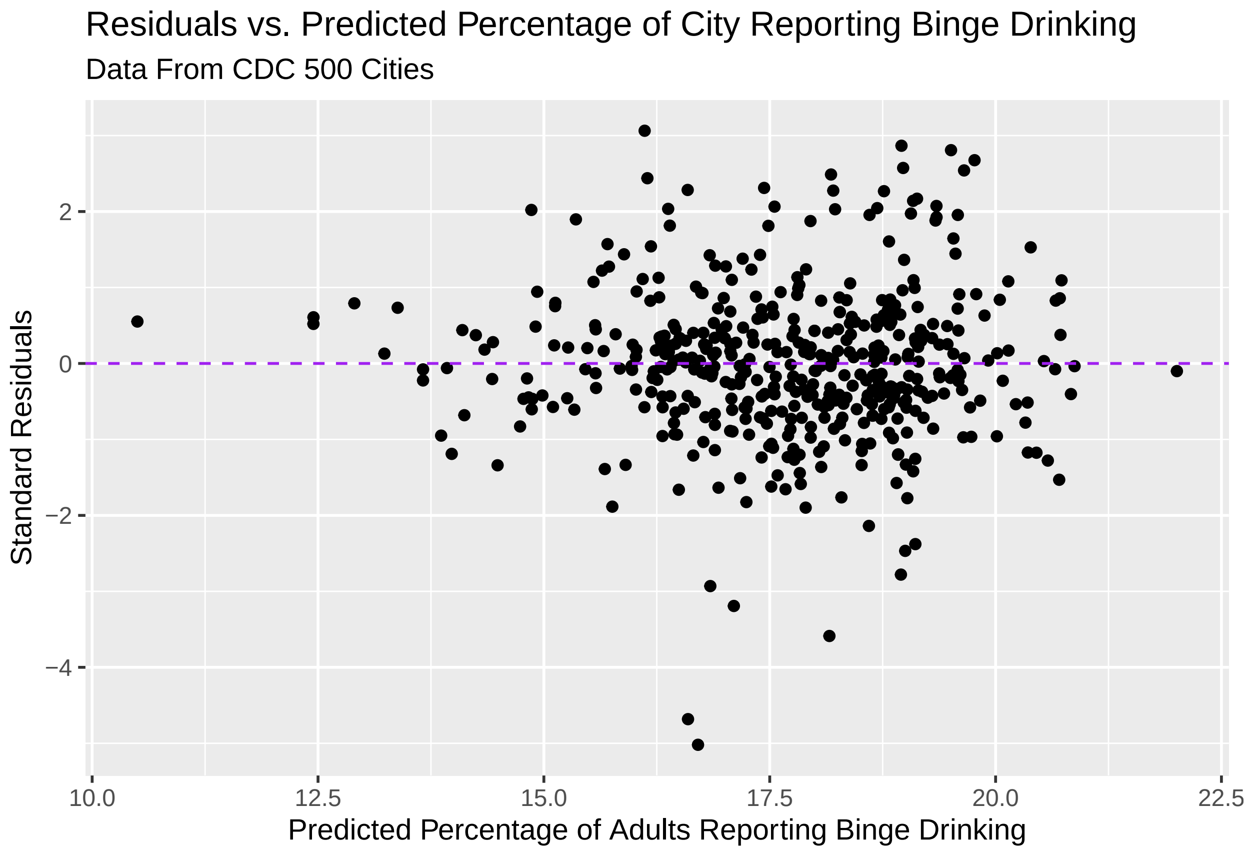
```
## [1] 0.3488416
```

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

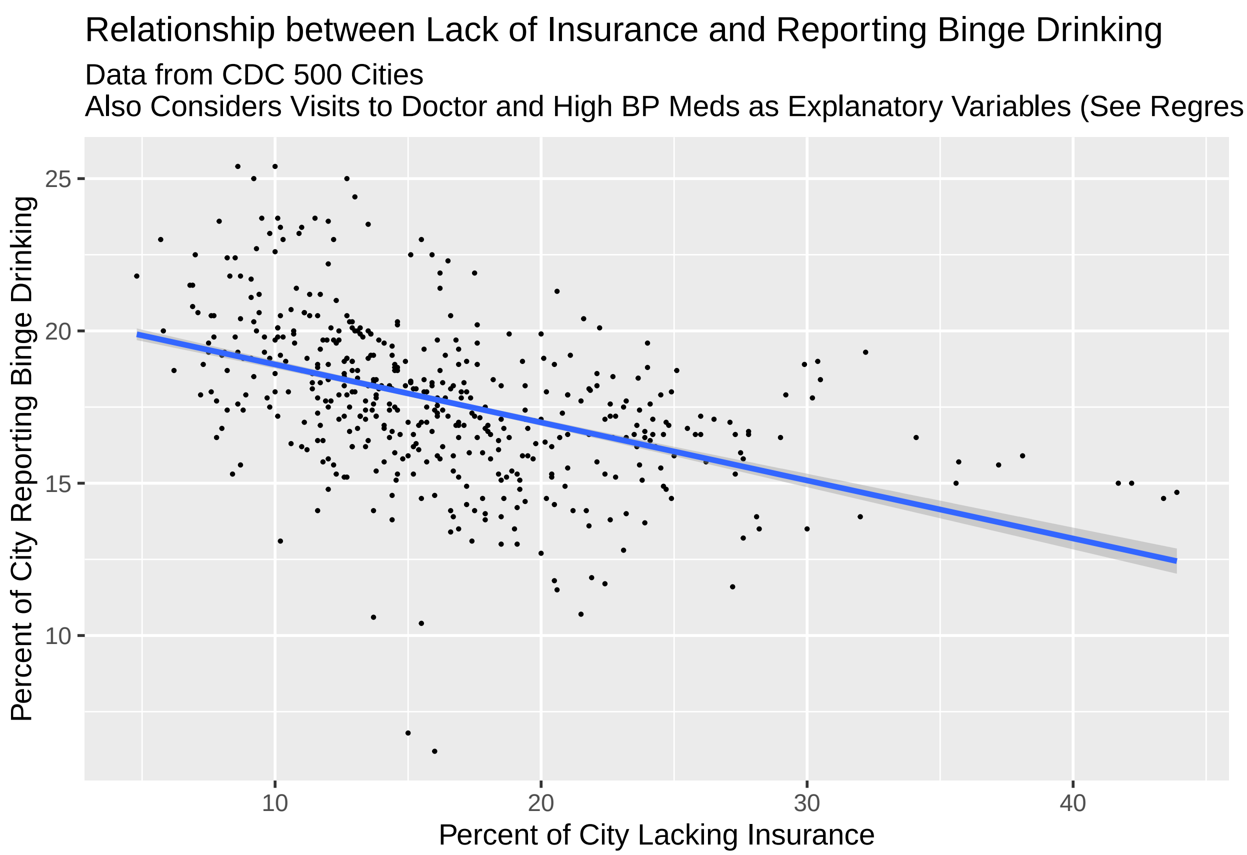
```
int_access_binge_drinking_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted Percentage of City Reporting Binge Drinking",
    subtitle = "Data From CDC 500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

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

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

Adj R-squared value for regression with all interactions

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

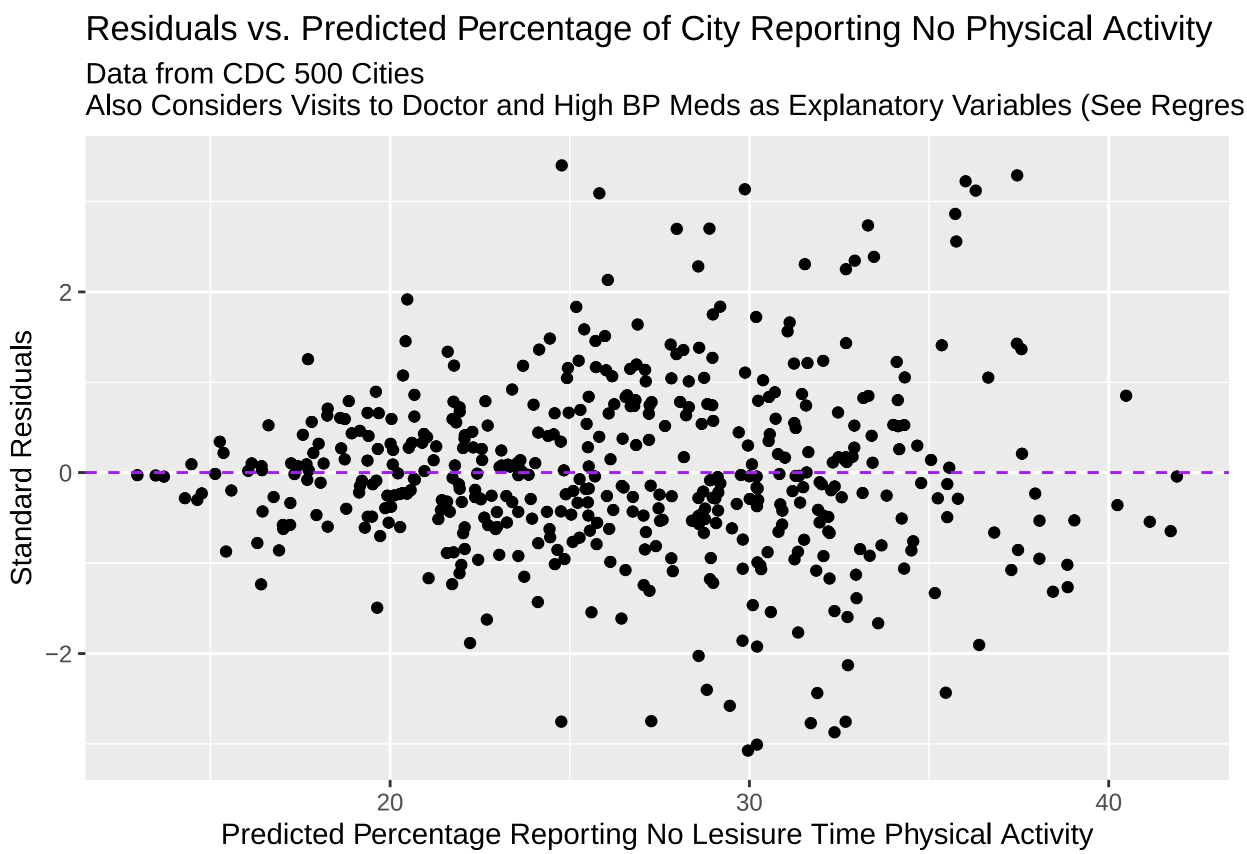
```
## [1] 0.8488063
```

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

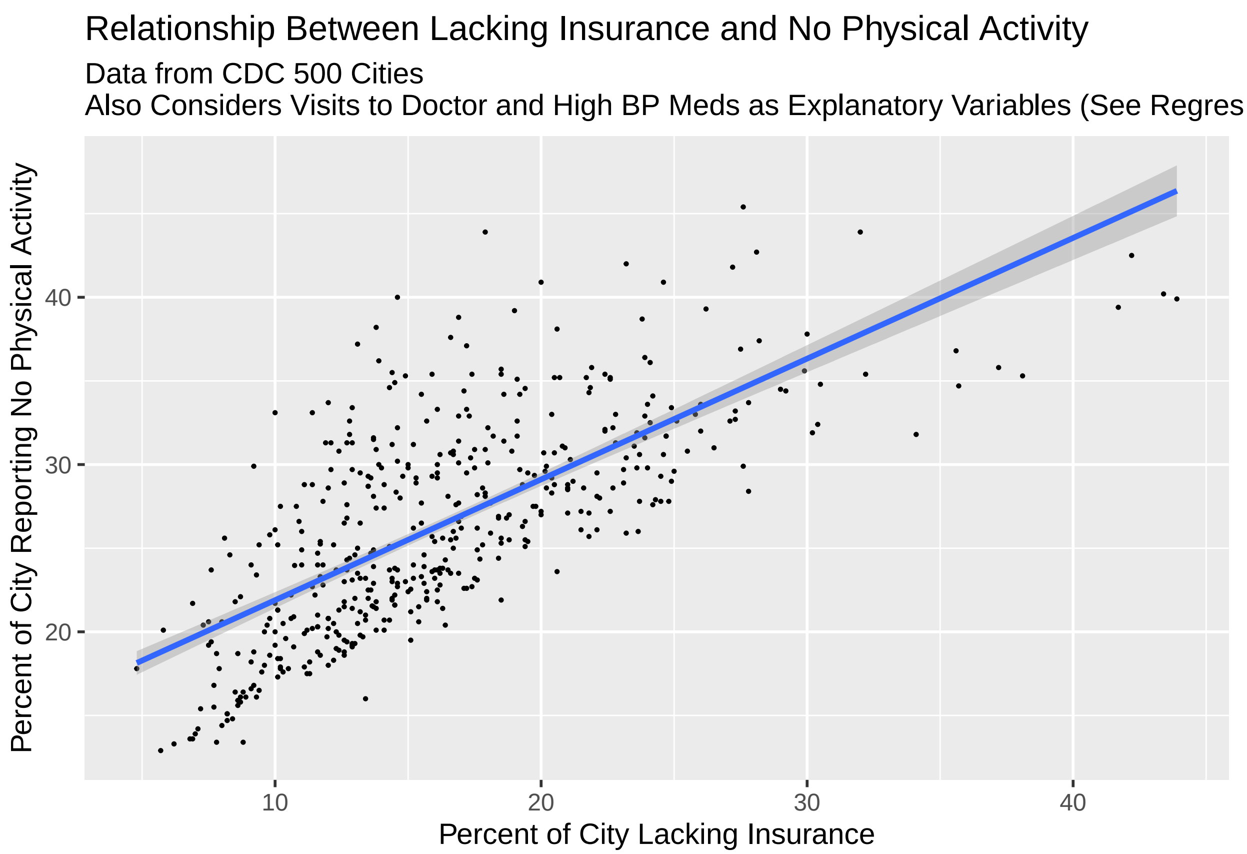
```r
int_access_physical_activity_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted Percentage of City Reporting No Physical Activity",
    subtitle = "Data 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") %>%
```

```
  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

```
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

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

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

Adj R-squared values for regression with one interaction

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

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

Adj R-squared values for regression with all interactions

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

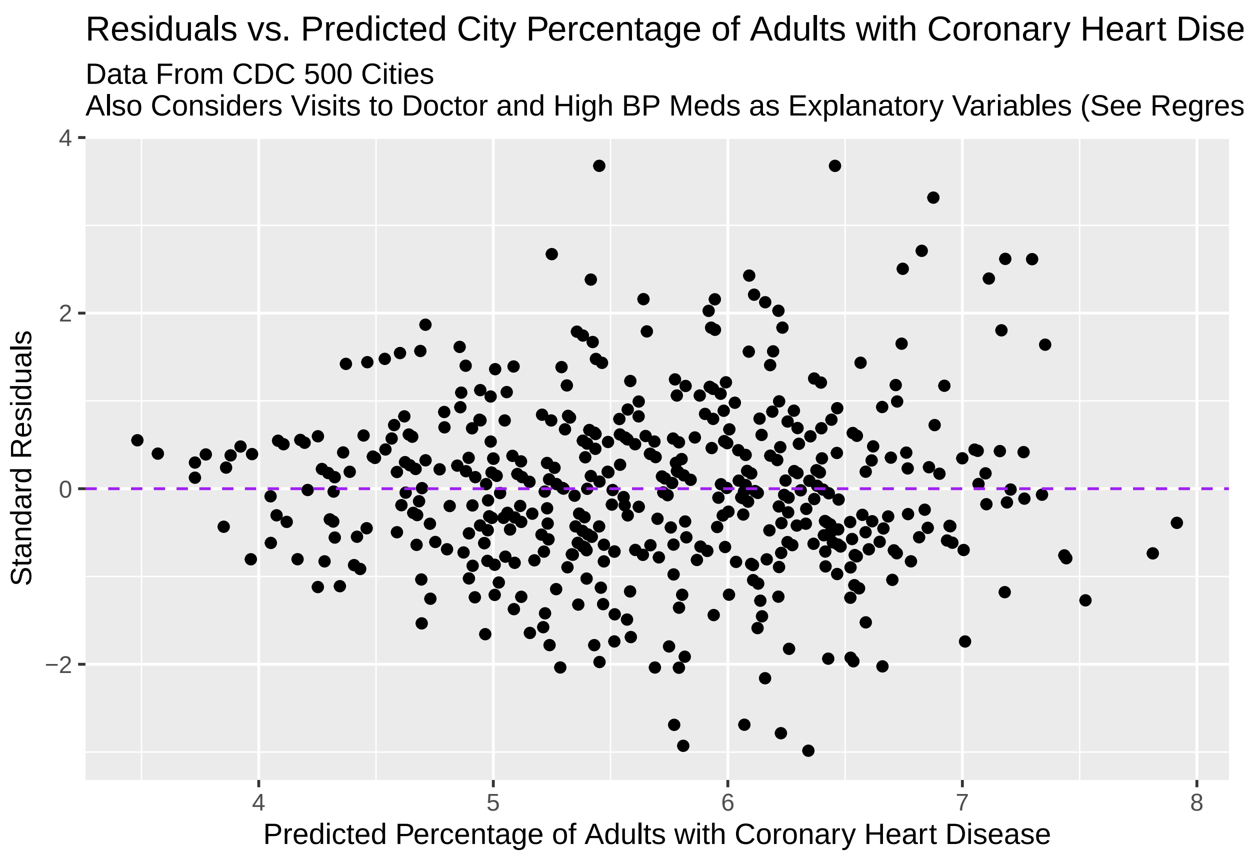
```
## [1] 0.6667498
```

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

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

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

Adj R-squared value for regression with all interactions

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

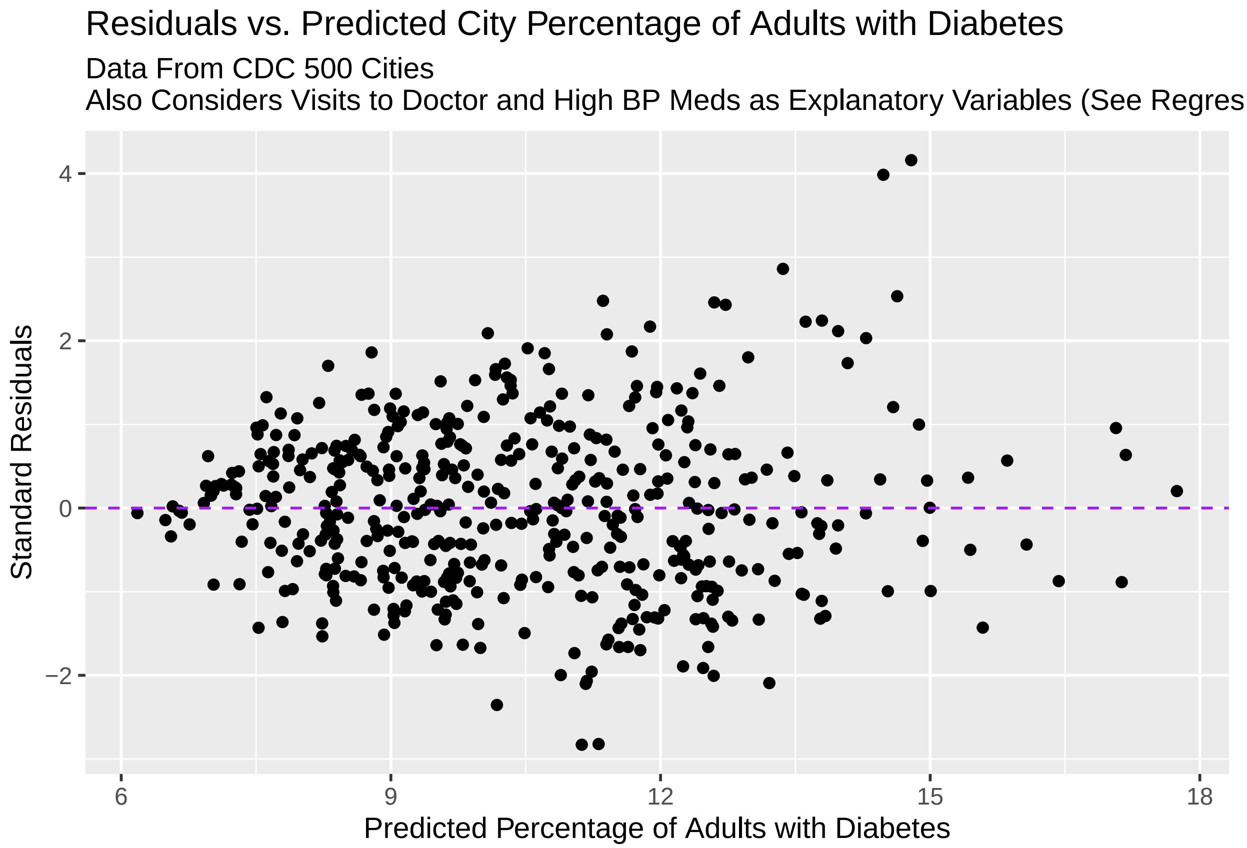
```
## [1] 0.7110294
```

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)

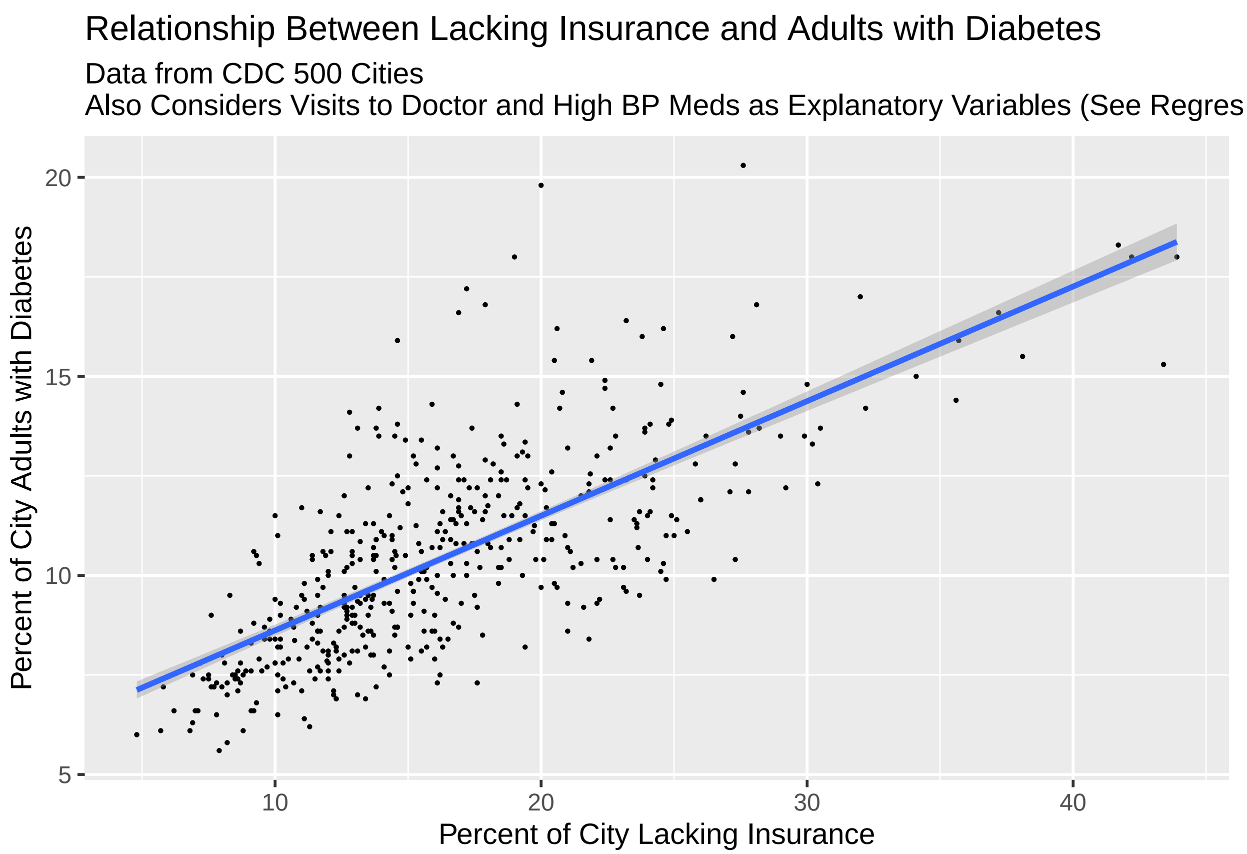
```
int_access_diabetes_fit_aug %>%
  ggplot(mapping = aes(x = .fitted, y = .std.resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "purple", lty = "dashed") +
  labs(
    title = "Residuals vs. Predicted City Percentage of Adults with Diabetes",
    subtitle = "Data From CDC 500 Cities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

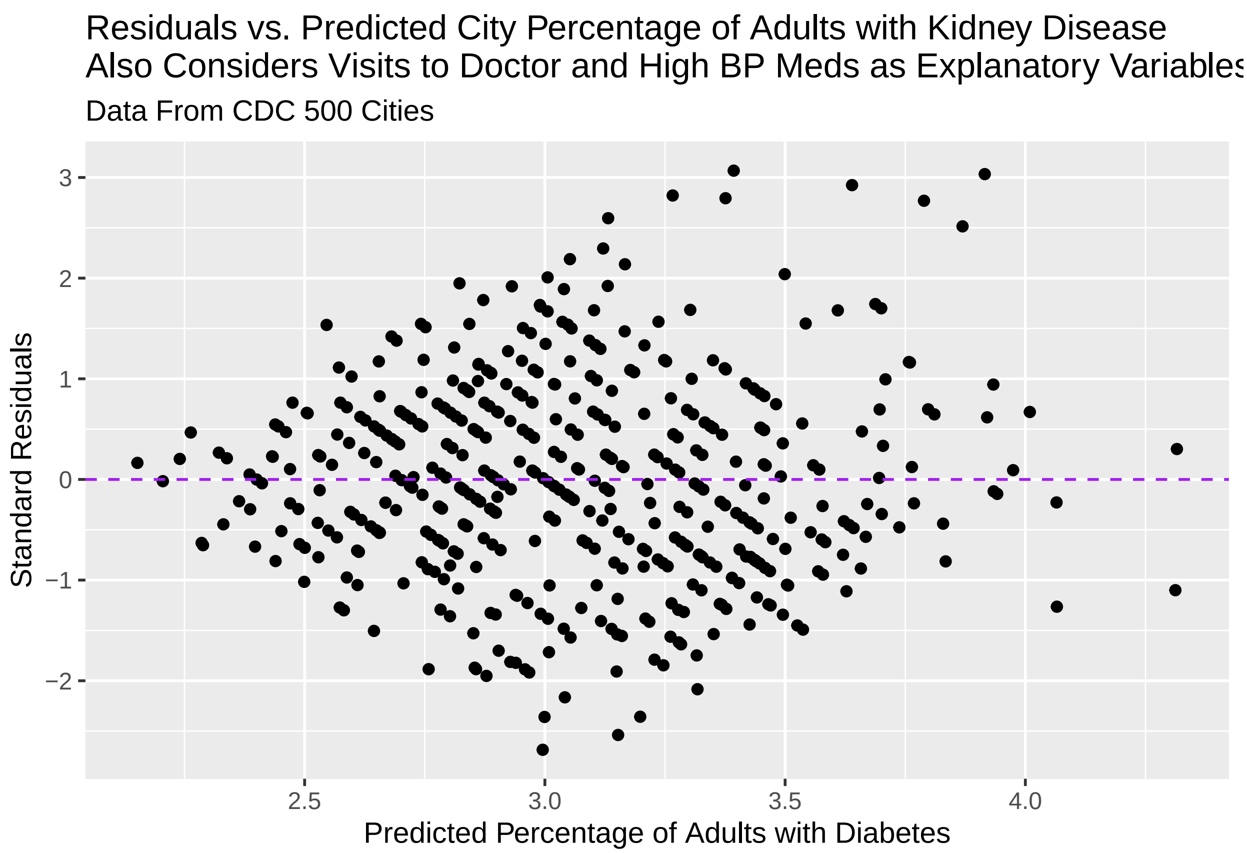
```
## [1] 0.6193093
```

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 lienar model would not be appropriate. Try a logistic model?

ANOVA Testing

Initial Visualizations

NOTE: Use initial visualizations to check if assumptions of ANOVA are met!

Overall Tests

```
summary(aov(insurance~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50   9260  185.20   8.487 <2e-16 ***
## Residuals   424   9252   21.82
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(visits_to_doctor~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50   8395  167.90   44.01 <2e-16 ***
## Residuals   421   1606    3.81
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 3 observations deleted due to missingness
```

```
summary(aov(medicine_high_bp~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50   9541  190.82   44.25 <2e-16 ***
## Residuals   422   1820    4.31
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 2 observations deleted due to missingness
```

```
summary(aov(smoking~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50   4752   95.03   9.747 <2e-16 ***
## Residuals   420   4095    9.75
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 4 observations deleted due to missingness
```

```
summary(aov(binge_drinking~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50   1719   34.37   9.579 <2e-16 ***
## Residuals   421   1511    3.59
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 3 observations deleted due to missingness
```

```
summary(aov(physical_activity~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50  10657  213.13   10.76 <2e-16 ***
## Residuals   421   8343   19.82
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 3 observations deleted due to missingness
```

```
summary(aov(heart_disease~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50  201.1   4.021   5.974 <2e-16 ***
## Residuals   421  283.4   0.673
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 3 observations deleted due to missingness
```

```
summary(aov(diabetes~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## StateDesc    50   1061   21.21   4.632 <2e-16 ***
## Residuals   421   1928    4.58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 3 observations deleted due to missingness
```

```
summary(aov(kidney_disease~StateDesc,data=data_500_cities)) %>%
  print()
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## StateDesc    50  21.77  0.4354   2.102 4.48e-05 ***
## Residuals   422  87.41  0.2071
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 2 observations deleted due to missingness
```

It seems like pretty much all the overall tests indicate significant variance across the groups.

Step Down Tests

```
insurance_state_pair <- pairwise.t.test(data_500_cities$insurance,    data_500_cities$StateDesc, p.adj
sig_ins_state_pairs <- broom::tidy(insurance_state_pair) %>%
  filter(p.value<0.05) %>%
  arrange(group1,group2)
nrow(sig_ins_state_pairs)
```

```
## [1] 0
```

```
print(sig_ins_state_pairs)
```

```
## # A tibble: 0 x 3
## # ... with 3 variables: group1 <chr>, group2 <chr>, p.value <dbl>
```

The overall ANOVA test says there is a significance but the Step Down stests show no significant pairs?

```
doctor_state_pair <- pairwise.t.test(data_500_cities$visits_to_doctor,    data_500_cities$StateDesc, p.
sig_doctor_state_pairs <- broom::tidy(doctor_state_pair) %>%
  filter(p.value<0.05) %>%
  arrange(group1,group2)
nrow(sig_doctor_state_pairs)
```

```
## [1] 0
```

```
print(sig_doctor_state_pairs)
```

```
## # A tibble: 0 x 3
## # ... with 3 variables: group1 <chr>, group2 <chr>, p.value <dbl>
```

Ok so there is clearly an issue. I think I am interpreting the F-statistic incorrectly?

```
smoking_state_pair <- pairwise.t.test(data_500_cities$smoking,    data_500_cities$StateDesc, p.adj = "ho
sig_smoking_state_pairs <- broom::tidy(smoking_state_pair) %>%
  filter(p.value<0.05) %>%
  arrange(group1,group2)
nrow(sig_smoking_state_pairs)
```

```
## [1] 0
```

```
print(sig_smoking_state_pairs)
```

```
## # A tibble: 0 x 3
## # ... with 3 variables: group1 <chr>, group2 <chr>, p.value <dbl>
```

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"
```

```r
head(world)
```

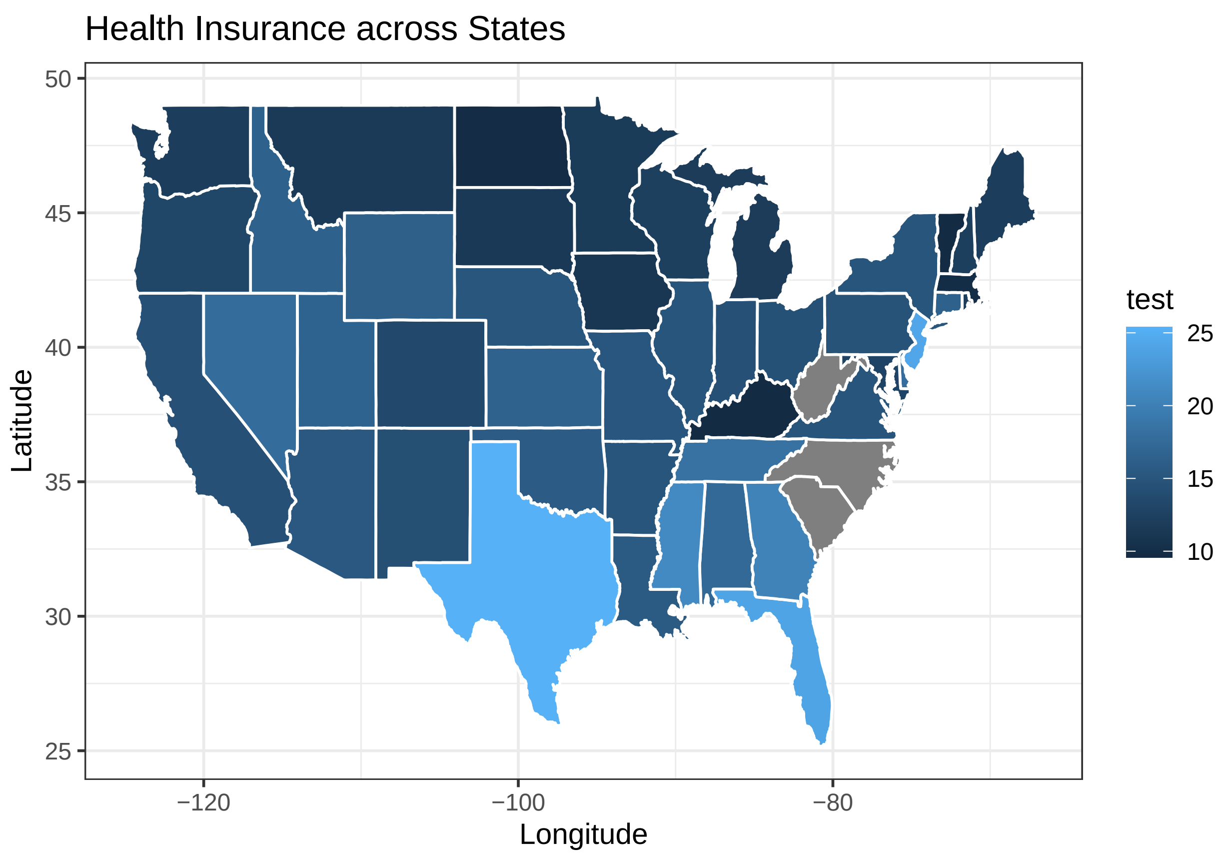
```
## Simple feature collection with 6 features and 63 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -70.06611 ymin: -18.01973 xmax: 74.89131 ymax: 60.40581
## CRS:           +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
##   scalerank      featurecla 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 abrev 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
```

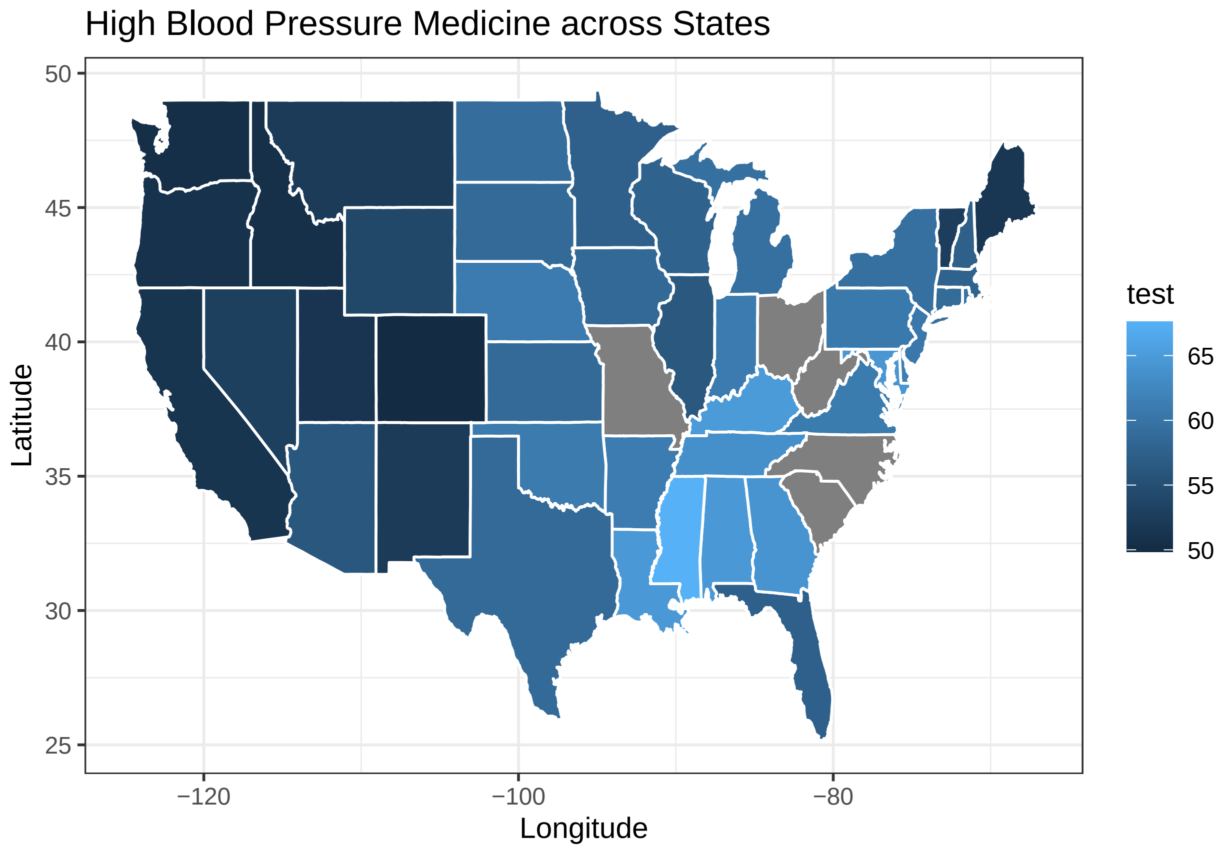
Access Variable: Health Insurance

```
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",
       title = "Health Insurance across States")
```



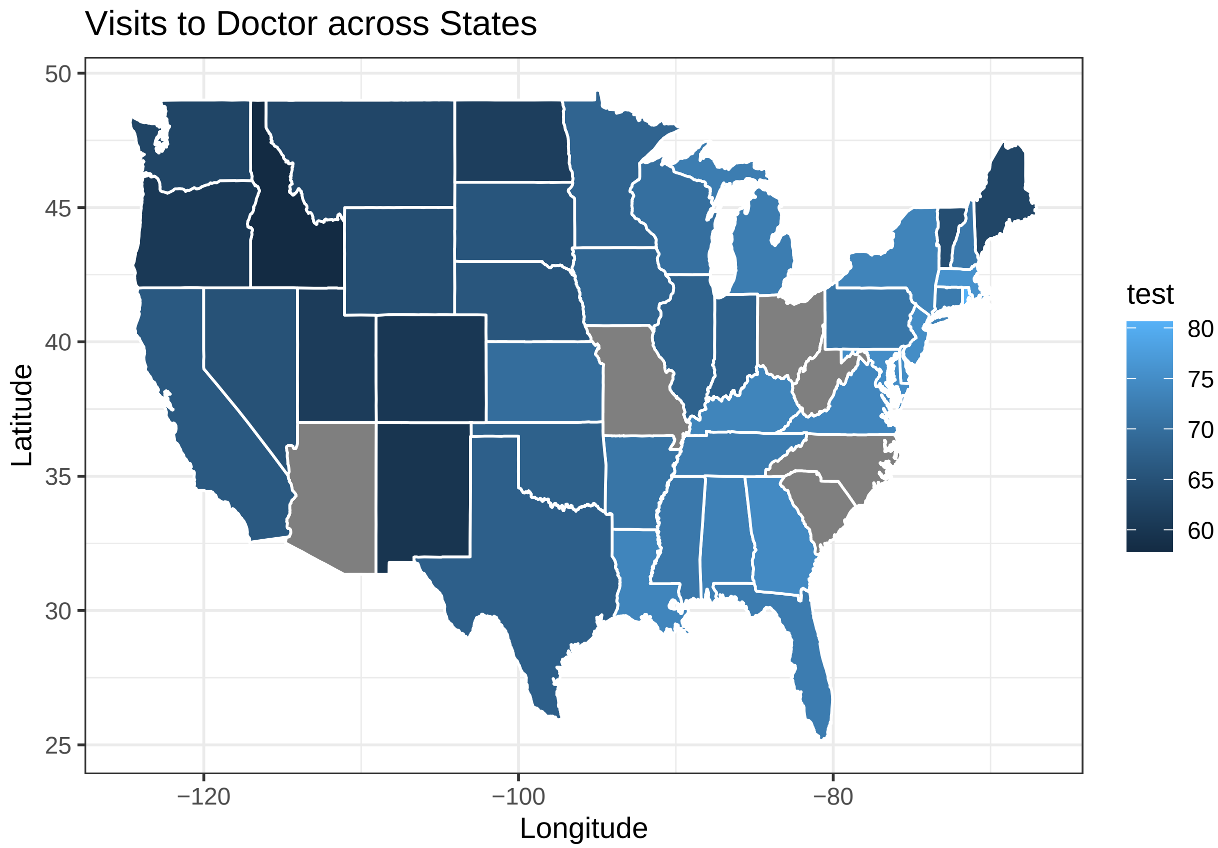
Access Variables: High Blood Pressure Medicine

```
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",
      title = "High Blood Pressure Medicine across States")
```



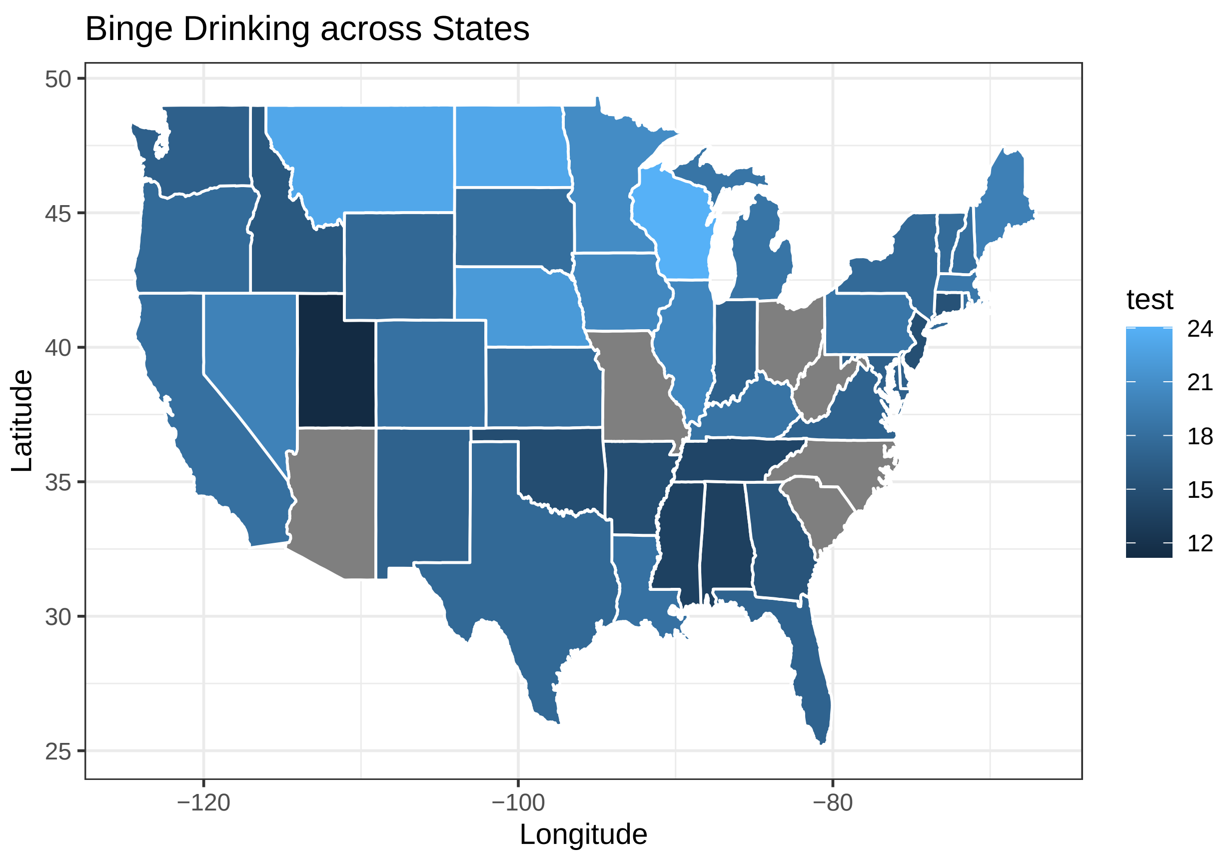
Access Variable: Visits to Doctor

```
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)) +
  geom_polygon(color = "white")+
    labs(x = "Longitude",
       y = "Latitude",
       title = "Visits to Doctor across States")
```



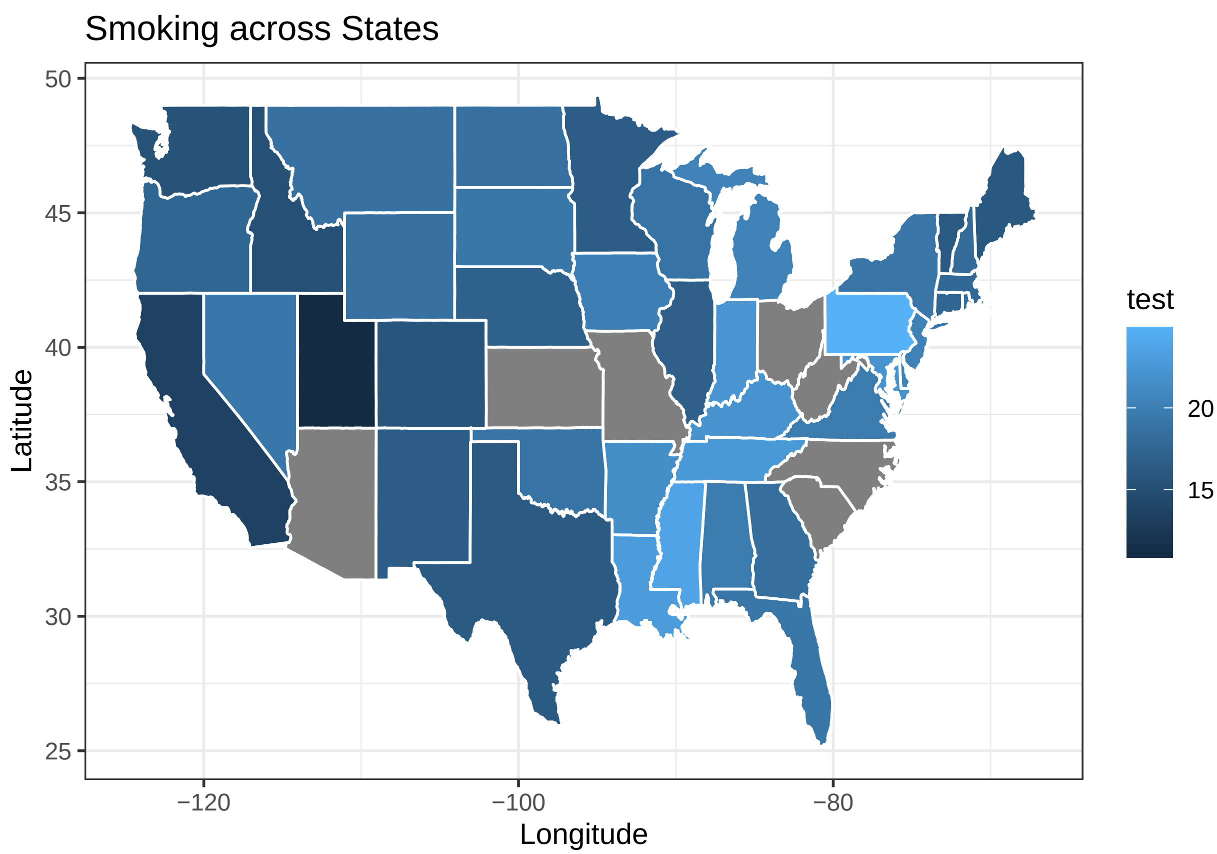
Behaviour: Binge Drinking

```
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",
       title = "Binge Drinking across States")
```



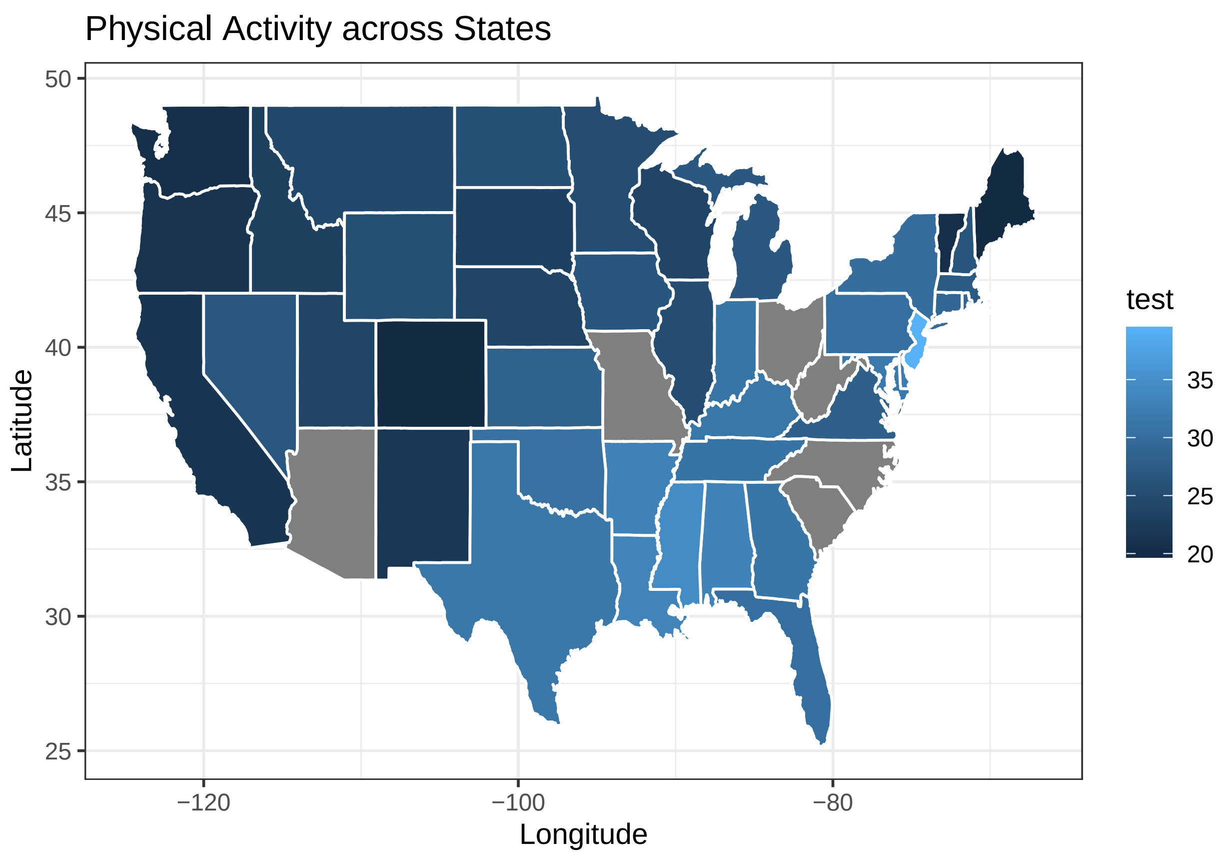
Behaviour: Smoking

```
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",
       title = "Smoking across States")
```



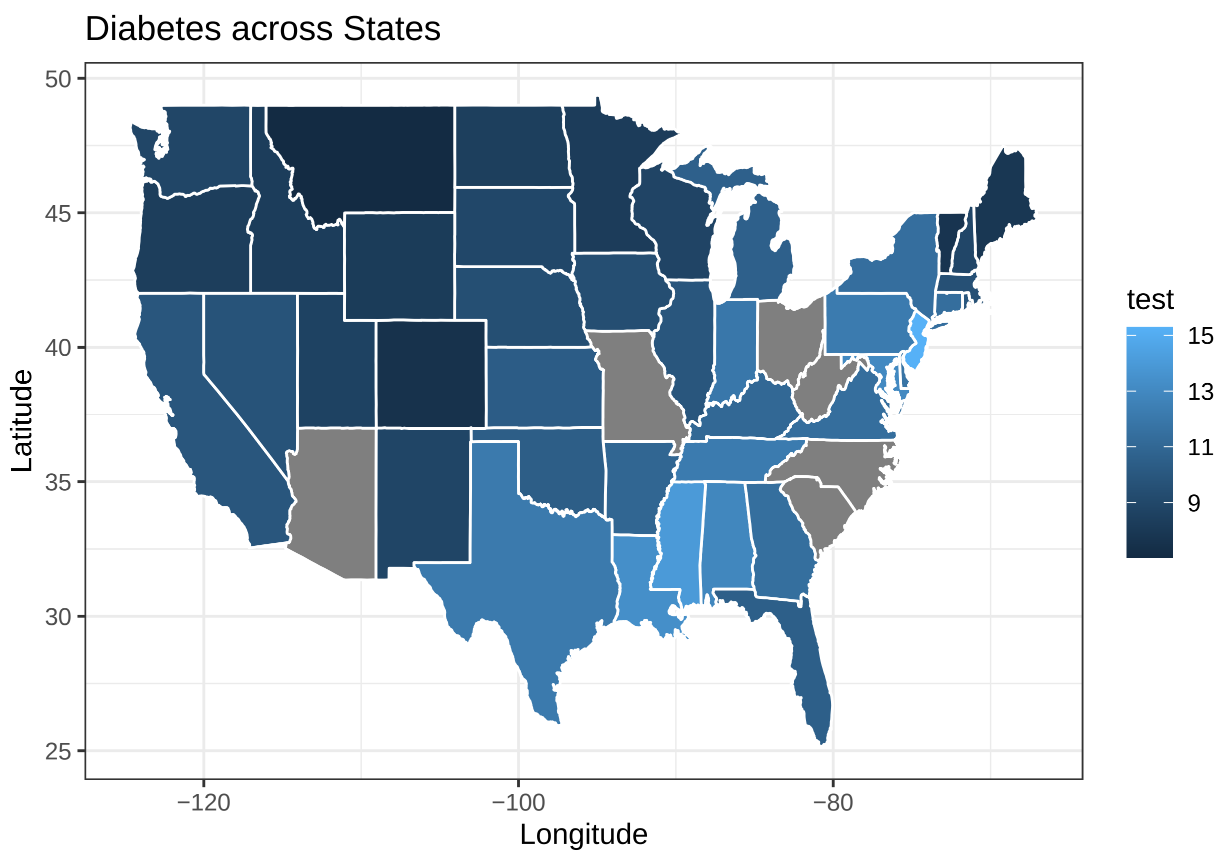
Behaviour: Physical Activity

```
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",
       title = "Physical Activity across States")
```



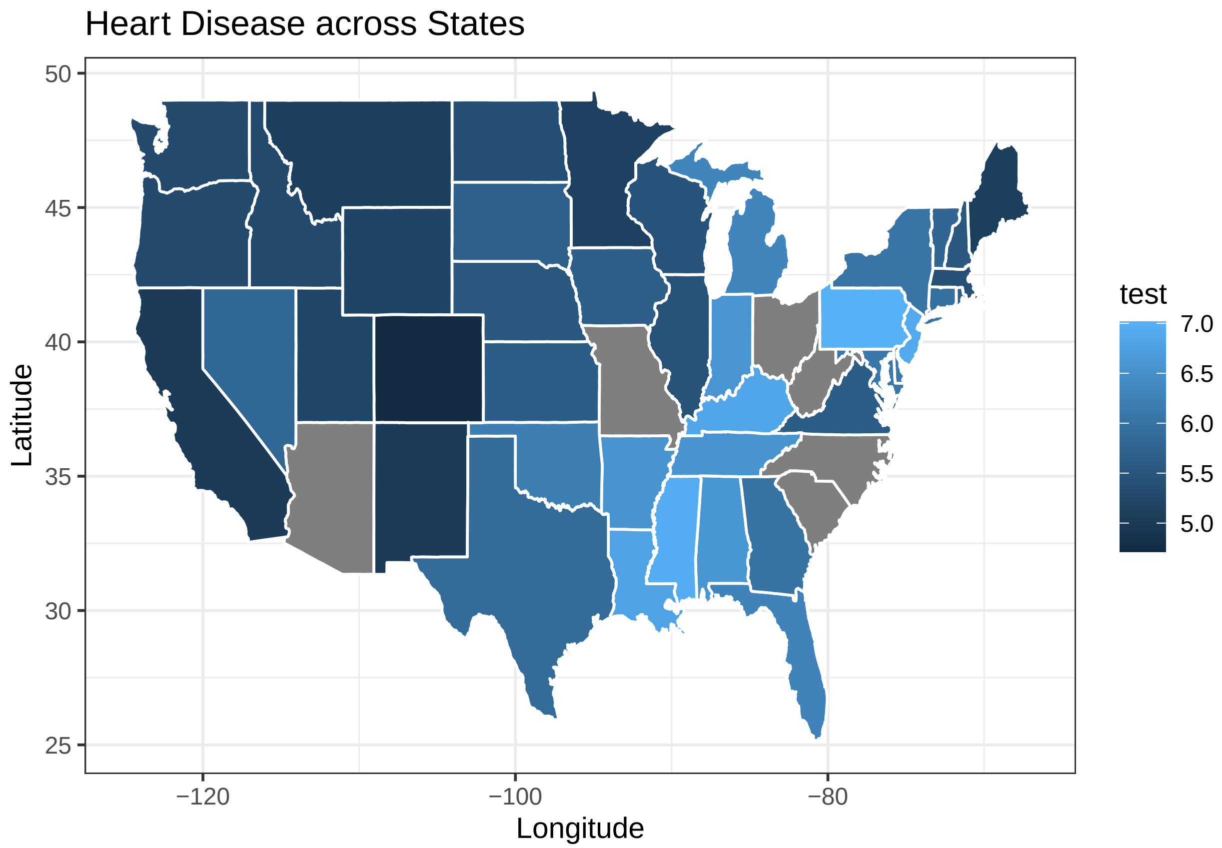
Health Outcome: Diabetes

```
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",
       title = "Diabetes across States")
```



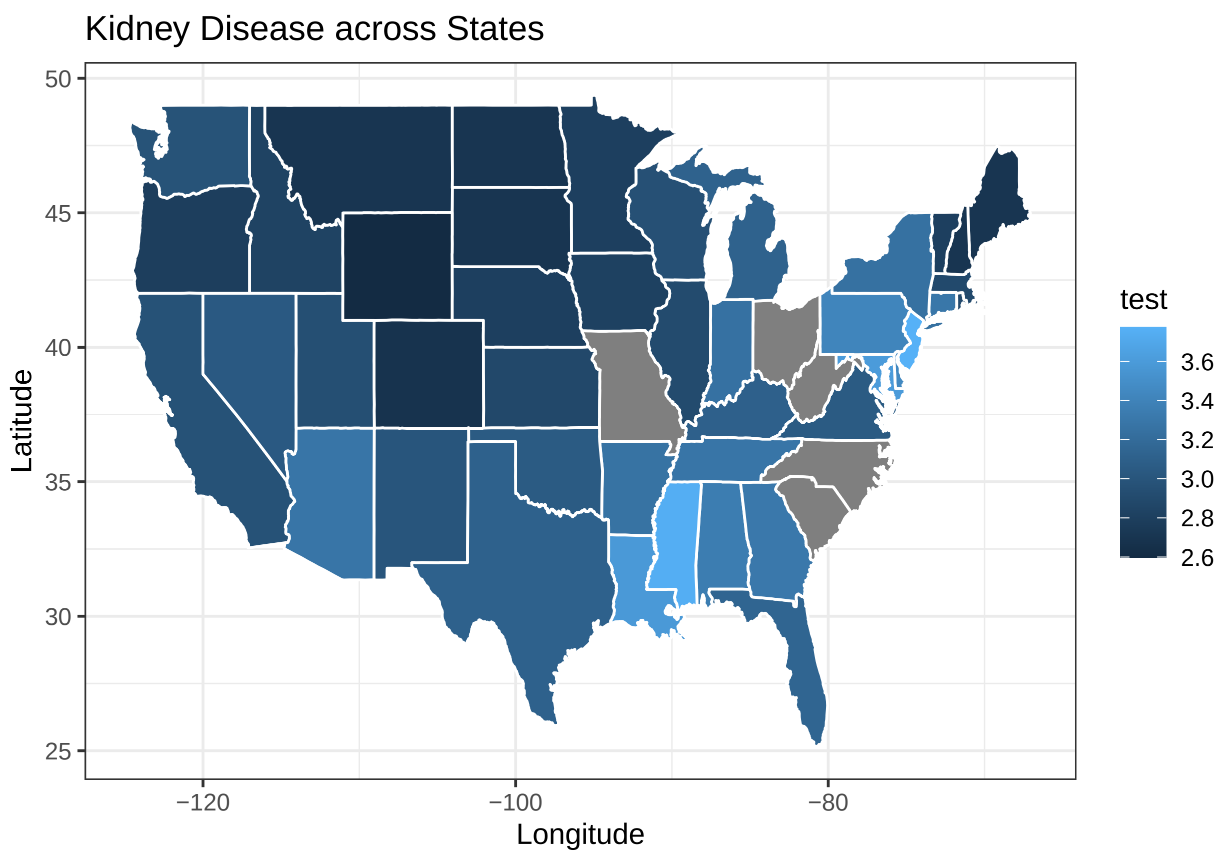
Health Outcome: Heart Disease

```
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",
       title = "Heart Disease across States")
```



Health Outcome: Kidney Disease

```
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",
      title = "Kidney Disease across States")
```


Kidney Disease across States
50
45
40
35
30
25
Latitude
−120
−100
−80
Longitude
test
3.6
3.4
3.2
3.0
2.8
2.6