Math 530/630 Final Replication Project

Josh Burkhart, Steve Chamberlin, Kristen Stevens

December 2, 2015

Association of Arsenic Exposure with Lung Cancer Incidence Rates in the United States

Citation: Putila JJ, Guo NL (2001) Association of Arsenic Exposure with Lung Cancer Incidence Rates in the United States. PLOS ONE 6(10): e25886.

```
full <- read.csv("~/SoftwareProjects/Probability/final_proj/File_S4.csv", row.names = 1)  # change dire
```

The authors' data file (FileS4.csv) includes 757 observations of 20 variables. We have named this data.frame "full". The unit of analysis is County. The variables of interest are mean aresenic level in parts per million weighted by county population (Ascounty), median income of county (MedIncome), population (Population), county smoking prevalence calculated as a percent of respondents age 18 or older who reported having smoked more than 100 cigarettes in their lifetime (smkrate), and age-adjusted lung cancer incidence rates (AdjRate).

We first generated log transformed and centered variables from the variables described above. The authors chose not to log transform smoking rate, but still moved the raw smoking rate into a variable that appears log transformed.

```
full$lnAs <- log(full$Ascounty) - mean(na.omit(log(full$Ascounty)))
full$lnInc <- log(full$MedIncome) - mean(na.omit(log(full$MedIncome)))
full$Population <- as.numeric(as.character(full$Population))
full$lnsmk <- full$smkrate
full$lnar <- log(full$AdjRate)
```

Regression Analysis

"The first analysis sought to determine the influence of exposure levels of arsenic on lung cancer incidence in the U.S., and persistance of these effects controlling for possible confounders. The association betweeen each contaminant and lung cancer incidence was assessed using Poisson regression in order to reflect the annual incidence rate as a counting measure."

For comparison, we performed a linear regression, with a log transformed lung cancer incidence rate, in addition to a Poisson regression for each analysis. The following models predict lung cancer incidence weighted by county population using the untransformed data.

```
# Poisson regression
glm1 <- glm(full$AdjRate ~ full$Ascounty, family = poisson, weights = as.numeric(full$Population))
summary(glm1)
```

```
Call:
glm(formula = full$AdjRate ~ full$Ascounty, family = poisson,
    weights = as.numeric(full$Population))
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-6257.0   -103.2    161.1    444.0   2459.2

Coefficients:
                 Estimate  Std. Error z value            Pr(>|z|)
(Intercept)   4.167391730 0.000023700  175836 <0.0000000000000002 ***
full$Ascounty 0.004479178 0.000001937    2312 <0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 317986702  on 741  degrees of freedom
Residual deviance: 312986732  on 740  degrees of freedom
  (15 observations deleted due to missingness)
AIC: 835273995

Number of Fisher Scoring iterations: 4
```

```
# odds ratio with CI
glm1_odds <- exp(cbind(OR = coef(glm1), confint(glm1)))
```

```
Waiting for profiling to be done...
```

```
# Linear regression
lm1 <- lm(full$lnar ~ full$Ascounty, weights = as.numeric(full$Population))
summary(lm1)
```

```
Call:
lm(formula = full$lnar ~ full$Ascounty, weights = as.numeric(full$Population))

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-738.43   -8.18   23.65   55.32  293.99

Coefficients:
              Estimate Std. Error t value            Pr(>|t|)
(Intercept)    4.14686    0.01727 240.168 <0.0000000000000002 ***
full$Ascounty  0.00373    0.00147   2.537              0.0114 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 83.81 on 740 degrees of freedom
  (15 observations deleted due to missingness)
Multiple R-squared:  0.008621,  Adjusted R-squared:  0.007282
F-statistic: 6.435 on 1 and 740 DF,  p-value: 0.01139
```

```
# odds ratio with CI
lm1_odds <- exp(cbind(OR = coef(lm1), confint(lm1)))
```

```r
# Poisson regression
SESassmk <- glm(full$AdjRate ~ full$smkrate + full$Ascounty + full$MedIncome,
    family = poisson, weights = as.numeric(full$Population))
summary(SESassmk)
```

```
Call:
glm(formula = full$AdjRate ~ full$smkrate + full$Ascounty + full$MedIncome,
    family = poisson, weights = as.numeric(full$Population))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3559.7   -240.9     46.1    347.7   1949.6

Coefficients:
                      Estimate      Std. Error z value            Pr(>|z|)
(Intercept)     3.520651941992  0.000121330940   29017 <0.0000000000000002
full$smkrate    1.801926275227  0.000191317650    9419 <0.0000000000000002
full$Ascounty   0.003931137793  0.000001929776    2037 <0.0000000000000002
full$MedIncome -0.000003538024  0.000000001308   -2706 <0.0000000000000002

(Intercept)    ***
full$smkrate   ***
full$Ascounty  ***
full$MedIncome ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 305018305  on 584  degrees of freedom
Residual deviance: 180096592  on 581  degrees of freedom
  (172 observations deleted due to missingness)
AIC: 687702879

Number of Fisher Scoring iterations: 4
```

```r
SESassmk_odds <- exp(cbind(OR = coef(SESassmk), confint(SESassmk)))
```

```
Waiting for profiling to be done...
```

```r
# Linear regressions
SESassmklm <- lm(full$lnar ~ full$smkrate + full$Ascounty + full$MedIncome,
    weights = as.numeric(full$Population))
summary(SESassmklm)
```

```
Call:
lm(formula = full$lnar ~ full$smkrate + full$Ascounty + full$MedIncome,
    weights = as.numeric(full$Population))

Weighted Residuals:
```

```
    Min      1Q  Median      3Q     Max
-442.09  -28.41    7.35   42.54  234.40

Coefficients:
                    Estimate    Std. Error t value             Pr(>|t|)
(Intercept)     3.2915838146  0.0692881380  47.506 < 0.0000000000000002
full$smkrate    2.1732197826  0.1129146604  19.247 < 0.0000000000000002
full$Ascounty   0.0034245343  0.0012310863   2.782              0.005583
full$MedIncome -0.0000023964  0.0000007225  -3.317              0.000968

(Intercept)    ***
full$smkrate   ***
full$Ascounty  **
full$MedIncome ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 69.48 on 581 degrees of freedom
  (172 observations deleted due to missingness)
Multiple R-squared:  0.4433,    Adjusted R-squared:  0.4404
F-statistic: 154.2 on 3 and 581 DF,  p-value: < 0.00000000000000022
```

```
SESassmklm_odds <- exp(cbind(OR = coef(SESassmklm), confint(SESassmklm)))
```

One point to make about the comparison of Poisson regressions with linear regressions is the R squared values from the linear regression in both the unadjusted and adjusted model. It should be noted that the R square in the unadjusted model is only .008, indicating that arsenic only accounts for .8% of the variation seen in lung cancer incidence rate. When smoking and income were added to the model the R square increased to 44%. Although both models were significant, this seems to indicate that smoking and income are really explaining most of the variation in lung cancer incidence.

Table 1: Unadjusted Model

Summary of Poisson regressions of the effect of arsenic concentration (ppm) on county-level lung cancer incidence rates in the U.S. in an unadjusted model. (We broke up Table 1 into two parts: the Unadjusted Model and the Adjusted Model.)

```
Model_and_Variable = c("Arsenic")
Coefficient = c(round(summary(glm1)$coefficients[2], digits = 4))
Std.Error = c("1.9 x 10^-6")  # summary(glm1)$coefficients[4]
Odds_Ratio_CI = c(paste(c(round(glm1_odds[2, 1], digits = 3), "(", round(glm1_odds[2,
    2], digits = 3), "-", round(glm1_odds[2, 3], digits = 3), ")"), collapse = " "))
P_value = c("P<0.0001")  # summary(glm1)$coefficients[8]
N = c(742)  # N = total observations (757) - missing observations (15)
df = data.frame(Model_and_Variable, Coefficient, Std.Error, Odds_Ratio_CI, P_value,
    N)
kable(df)
```

Model_and_Variable	Coefficient	Std.Error	Odds_Ratio_CI	P_value	N
Arsenic	0.0045	1.9 x 10^-6	1.004 (1.004 - 1.004)	P<0.0001	742

Table 1: Adjusted Model

Summary of Poisson regressions of the effect of arsenic concentration (ppm) on county-level lung cancer incidence rates in the U.S. in a model adjusted for both smoking and median county income.

```
Model_and_Variable = c("Arsenic","Smoking","Median Income")
Coefficient = c(round(summary(SESassmk)$coefficients[3], digits = 4),
                round(summary(SESassmk)$coefficients[2], digits = 2),
                "-3.54 x 10^-6") # summary(SESassmk)$coefficients[4]
Std.Error = c("1.9 x 10^-6", # summary(SESassmk)$coefficients[7]
              round(summary(SESassmk)$coefficients[6], digits = 4),
              "1.31 x 10^-9") # summary(SESassmk)$coefficients[8]
Odds_Ratio_CI = c(paste(c(round(SESassmk_odds[3], digits = 3), "(",
                          round(SESassmk_odds[7], digits = 3), "-",
                          round(SESassmk_odds[11], digits = 3), ")")
                          ,collapse = " "),
                  paste(c(round(SESassmk_odds[2], digits = 3), "(",
                          round(SESassmk_odds[6], digits = 3), "-",
                          round(SESassmk_odds[10], digits = 3), ")")
                          ,collapse = " "),
                  paste(c("0.999", "(", # SESassmk_odds[4]
                          "0.999", "-", # SESassmk_odds[8]
                          "0.999", ")") # SESassmk_odds[12]
                          ,collapse = " "))
P_value = c("P<0.0001") #summary(SESassmk)Coefficients[14,15,16]
N = c(585) # N = total observations (757) - missing observations (172)
df = data.frame(Model_and_Variable,Coefficient,Std.Error,Odds_Ratio_CI,P_value,N)
kable(df)
```

Model_and_Variable	Coefficient	Std.Error	Odds_Ratio_CI	P_value	N
Arsenic	0.0039	1.9 x 10^-6	1.004 (1.004 - 1.004)	P<0.0001	585
Smoking	1.8	0.0002	6.061 (6.059 - 6.064)	P<0.0001	585
Median Income	-3.54 x 10^-6	1.31 x 10^-9	0.999 (0.999 - 0.999)	P<0.0001	585

Table 2

Difference in lung cancer incidence attributable to arsenic exposure alone for high and low-exposure areas in the U.S. based on the results of the adjusted Poisson models and the USGS survey quantiles in Figure 1. The value of 5.3% was determined by the difference between the highest and lowest arsenic quantiles multiplied by the percent change in lung cancer incidence rate per 1 ppm of arsenic.

```
Compound = c("Arsenic")
Low_ppm = c(1.477)  # from Figure 1
High_ppm = c(14.525)  # from Figure 1
B_Estimate = c(round(summary(SESassmk)$coefficients[3], digits = 4))
Lung_Cancer_Rate_Increase_Pct = c("5.3%")
df = data.frame(Compound, Low_ppm, High_ppm, B_Estimate, Lung_Cancer_Rate_Increase_Pct)
kable(df)
```

Compound	Low_ppm	High_ppm	B_Estimate	Lung_Cancer_Rate_Increase_Pct
Arsenic	1.477	14.525	0.0039	5.3%

This next section sets up variables that are categorical based on quartiles of the independent variables, or, in the case of income, based on low income cutoffs. This setup is for the series of ANOVAs that are to follow, and the associated graphs in Figure 2.

```
## Estimate the 25, 50, amd 75% quartile points for each variable for the
## quartiles interaction models
AsCut <- NA
AsCut[1] <- as.numeric(summary(full$lnAs)[2])
AsCut[2] <- as.numeric(summary(full$lnAs)[3])
AsCut[3] <- as.numeric(summary(full$lnAs)[5])

SmkCut <- NA
SmkCut[1] <- as.numeric(summary(full$lnsmk)[2])
SmkCut[2] <- as.numeric(summary(full$lnsmk)[3])
SmkCut[3] <- as.numeric(summary(full$lnsmk)[5])

SESCut <- NA
SESCut[1] <- as.numeric(summary(full$lnInc)[2])
SESCut[2] <- as.numeric(summary(full$lnInc)[3])
SESCut[3] <- as.numeric(summary(full$lnInc)[5])

## Calculate Strat Groups for the ANOVA##

## Smoking Quartiles
smkgrp <- ifelse(is.na(full$lnsmk), NA, ifelse(full$lnsmk < SmkCut[1], 1, ifelse(full$lnsmk >=
    SmkCut[1] & full$lnsmk < SmkCut[2], 2, ifelse(full$lnsmk >= SmkCut[2] &
    full$lnsmk < SmkCut[3], 3, 4))))

## SES Low-Income Cutoffs
SESgrp <- ifelse(is.na(full$MedIncome), NA, ifelse(full$MedIncome < 24000 &
    !is.na(full$MedIncome), 1, ifelse(full$MedIncome >= 24000 & full$MedIncome <
    28700, 2, ifelse(full$MedIncome >= 28700 & full$MedIncome < 38300, 3, 4))))

# Arsenic quartiles
AsQ <- ifelse(is.na(full$lnAs), NA, ifelse(full$lnAs < AsCut[1], 1, ifelse(full$lnAs >=
    AsCut[1] & full$lnAs < AsCut[2], 2, ifelse(full$lnAs >= AsCut[2] & full$lnAs <
    AsCut[3], 3, 4))))

## Smoking quartiles added to the main DF full
full$smkgrp <- ifelse(is.na(full$lnsmk), NA, ifelse(full$lnsmk < SmkCut[1],
    1, ifelse(full$lnsmk >= SmkCut[1] & full$lnsmk < SmkCut[2], 2, ifelse(full$lnsmk >=
        SmkCut[2] & full$lnsmk < SmkCut[3], 3, 4))))

## SES Low-Income Cutoffs added to the main DF full
full$SESgrp <- ifelse(is.na(full$MedIncome), NA, ifelse(full$MedIncome < 24000 &
    !is.na(full$MedIncome), 1, ifelse(full$MedIncome >= 24000 & full$MedIncome <
    28700, 2, ifelse(full$MedIncome >= 28700 & full$MedIncome < 38300, 3, 4))))

## Arsenic Quartiles added to the main DF full
full$AsQ <- ifelse(is.na(full$lnAs), NA, ifelse(full$lnAs < AsCut[1], 1, ifelse(full$lnAs >=
```

```
    AsCut[1] & full$lnAs < AsCut[2], 2, ifelse(full$lnAs >= AsCut[2] & full$lnAs <
    AsCut[3], 3, 4))))

## Quartile-Based Interaction Models Convert quartiles to factors
AsQf <- as.factor(AsQ)
smkgrpf <- as.factor(smkgrp)
smkgrpfbak <- smkgrpf
SESgrpf <- as.factor(SESgrp)

full$AsQf <- as.factor(full$AsQ)
full$smkgrpf <- as.factor(full$smkgrp)
full$smkgrpfbak <- full$smkgrpf
full$SESgrpf <- as.factor(full$SESgrp)
```

This next section runs all of the ANOVAs with the main goal to check for interactions between arsenic and smoking. They actually ran three ANOVAs, one is the arsenic/smoking interaction itself, another is arsenic/smoking interaction while controlling for SES and the last is arsenic/SES interaction controlling for smoking.

```
############## ARSENIC ## figure 2

# This section is collapsing the four level smoking factor in to two groups
# at the median so the data can be divided into two groups for two separate
# regressions
smkgrpf <- smkgrpfbak
smkgrpf <- ifelse(is.na(smkgrpf), NA, ifelse(smkgrpf == 1 | smkgrpf == 2, 1,
    2))

## Arsenic and Smoking This creates the data for the first line of table 3
## and the p value displayed on the right graph on figure 2
intAsSmk <- aov(full$AdjRate ~ SESgrpf + AsQf * smkgrpf, weights = as.numeric(full$Population))
summary(intAsSmk)
```

```
              Df      Sum Sq    Mean Sq F value               Pr(>F)
SESgrpf        3  2695113615  898371205  37.943 < 0.0000000000000002 ***
AsQf           3  1566705980  522235327  22.057    0.000000000000157 ***
smkgrpf        1  2613664023 2613664023 110.390 < 0.0000000000000002 ***
AsQf:smkgrpf   3   188901394   62967131   2.659               0.0475 *
Residuals    574 13590400837   23676657
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
172 observations deleted due to missingness
```

```
## Without SES This creates the p value displayed on the left graph for
## figure 2
intAsSmk2 <- aov(full$AdjRate ~ AsQf * smkgrpf, weights = as.numeric(full$Population))
summary(intAsSmk2)
```

```
              Df      Sum Sq    Mean Sq F value                Pr(>F)
AsQf           3  1890647672  630215891  24.392  0.00000000000000715 ***
smkgrpf        1  3665090147 3665090147 141.857 < 0.0000000000000002 ***
AsQf:smkgrpf   3   191344179   63781393   2.469                0.0611 .
```

```
Residuals    577 14907703852   25836575
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
172 observations deleted due to missingness
```

```
## Arsenic and SES this creates the statistics for table3 second model
intAsSES <- aov(full$AdjRate ~ smkgrpf + AsQf * SESgrpf, weights = as.numeric(full$Population))
summary(intAsSES)
```

```
              Df      Sum Sq    Mean Sq F value               Pr(>F)
smkgrpf        1  4252222627 4252222627  180.22 < 0.0000000000000002 ***
AsQf           3  1303515192  434505064   18.42      0.0000000000205 ***
SESgrpf        3  1319745800  439915267   18.64      0.0000000000151 ***
AsQf:SESgrpf   9   377928730   41992081    1.78               0.0691 .
Residuals    568 13401373501   23593967
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
172 observations deleted due to missingness
```

This next section creates the graph for Figure 2 for the left graph. This graph is attempting to visualize the unadjusted interaction between smoking and arsenic by running two separate regressions for low smoking and high smoking groups split by the median. The fitted points are then plotted creating two lines for each group. This section also creates a graph using raw data to illustrate this interaction for both unadjusted and adjusted models.

```
## Plot the Interaction between Arsenic and Smoking NOT adjusted for SES
smkgrpf <- smkgrpfbak
smkgrpf <- ifelse(is.na(smkgrpf), NA, ifelse(smkgrpf==1 | smkgrpf==2,1,2))
# Two models are run to get the two sets of fitted values for low and high smoking,     # # unadjusted

r1 <- glm(full[smkgrpf==1,]$AdjRate ~ full[smkgrpf==1,]$lnAs, family=poisson,
          weights=as.numeric(full[smkgrpf==1,]$Population))
summary(r1)
```

```
Call:
glm(formula = full[smkgrpf == 1, ]$AdjRate ~ full[smkgrpf ==
    1, ]$lnAs, family = poisson, weights = as.numeric(full[smkgrpf ==
    1, ]$Population))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4555.0   -105.5    172.0    525.9   2252.1

Coefficients:
                            Estimate Std. Error  z value
(Intercept)               4.14377600 0.00001604 258419.4
full[smkgrpf == 1, ]$lnAs 0.01931181 0.00003293    586.4
                                     Pr(>|z|)
(Intercept)               <0.0000000000000002 ***
full[smkgrpf == 1, ]$lnAs <0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

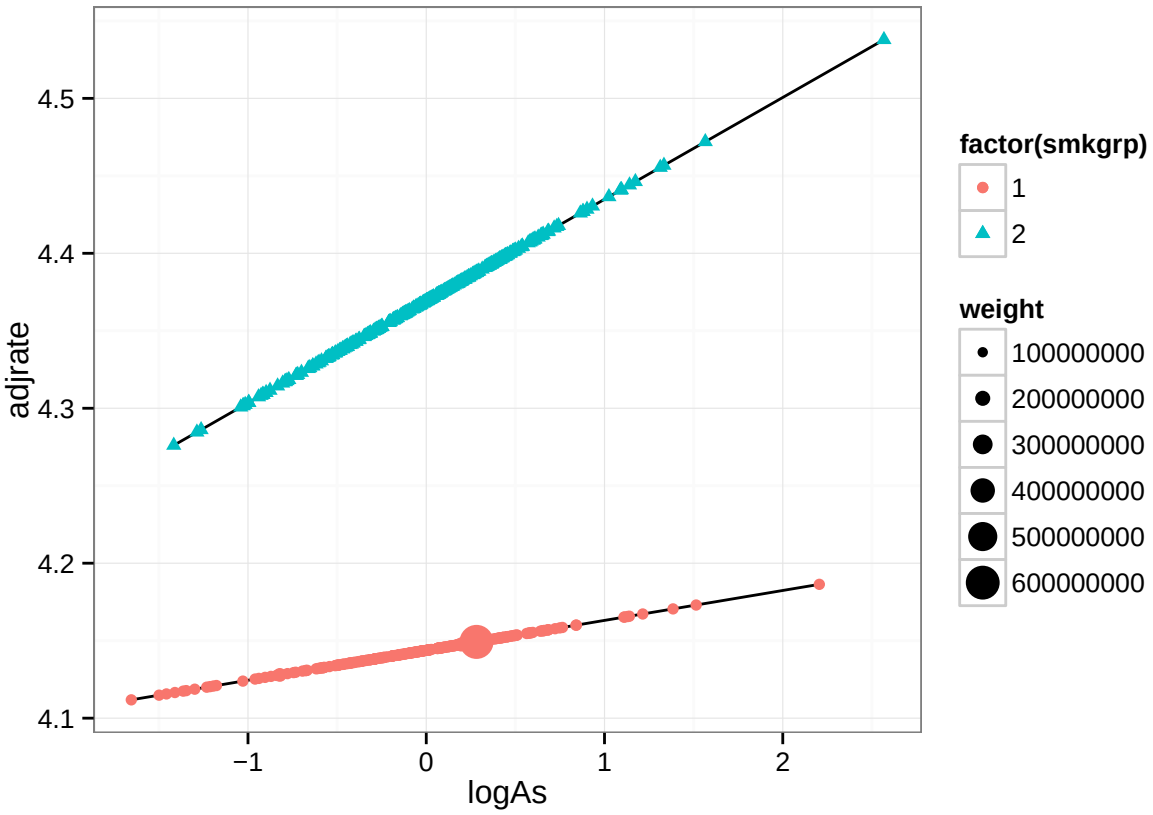
```
(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 175916938  on 289  degrees of freedom
Residual deviance: 175572576  on 288  degrees of freedom
  (167 observations deleted due to missingness)
AIC: 543281859

Number of Fisher Scoring iterations: 4
```

```
r2 <- glm(full[smkgrpf==2,]$AdjRate ~ full[smkgrpf==2,]$lnAs, family=poisson,
          weights=as.numeric(full[smkgrpf==2,]$Population))
summary(r2)
```

```
Call:
glm(formula = full[smkgrpf == 2, ]$AdjRate ~ full[smkgrpf ==
    2, ]$lnAs, family = poisson, weights = as.numeric(full[smkgrpf ==
    2, ]$Population))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1773.74   -228.76     53.34    326.82   1574.57

Coefficients:
                            Estimate Std. Error z value
(Intercept)                4.36914074 0.00002369  184430
full[smkgrpf == 2, ]$lnAs 0.06571306 0.00004283    1534
                                    Pr(>|z|)
(Intercept)                <0.0000000000000002 ***
full[smkgrpf == 2, ]$lnAs <0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 68053265  on 294  degrees of freedom
Residual deviance: 65713881  on 293  degrees of freedom
  (162 observations deleted due to missingness)
AIC: 205610885

Number of Fisher Scoring iterations: 4
```

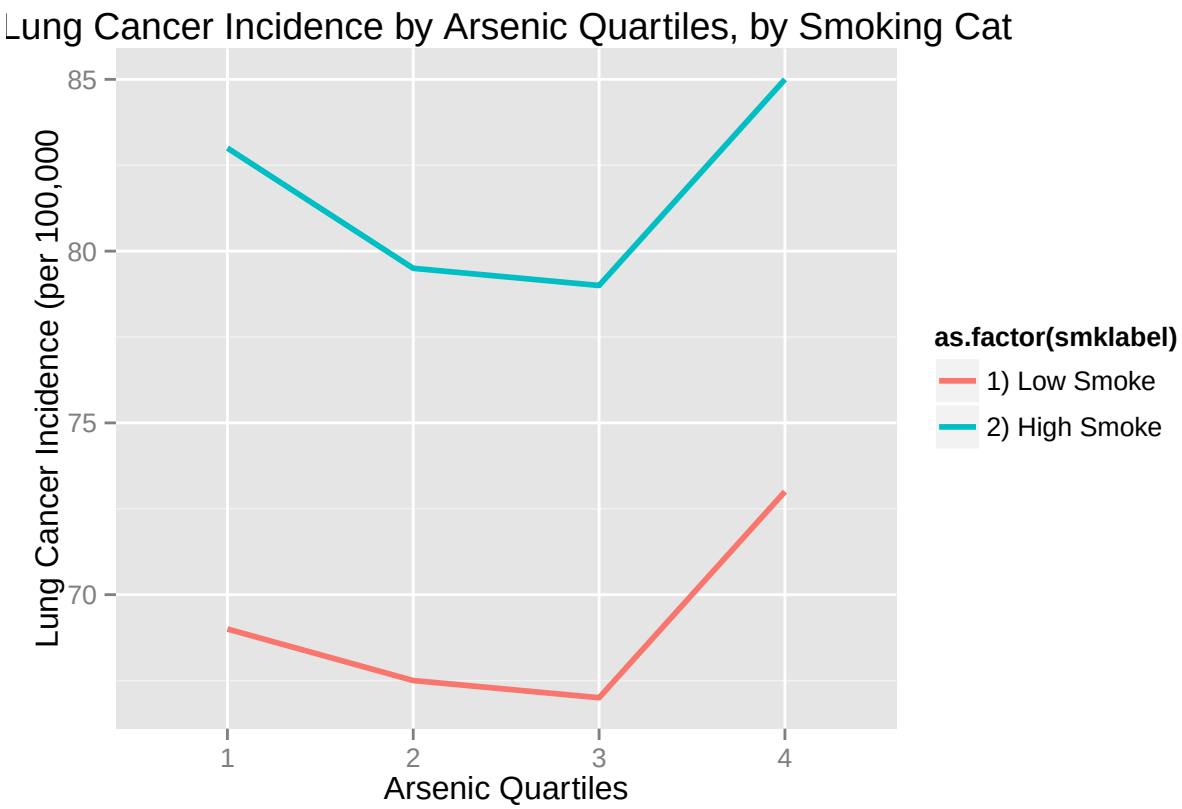
```
#   this section builds the dataset from the model output for graphing figure 2 left

data1 <- cbind( c(t(r1$model[2]),t(r2$model[2])),               # ln arsenic values
                c(log(r1$fitted.values),log(r2$fitted.values)), # fitted dependent vars
                c(r1$weights, r2$weights),                      # county populations
                c(rep("1",dim(r1$model[2])[1]), rep("2",dim(r2$model[2])[1])))
                # smoking group

# Variables are added to the model object so it can be used for figure 2
data1         <- as.data.frame(data1, stringsAsFactors=FALSE)
```

```
names(data1)  <- c("logAs","logRate", "weight", "smkgrp")
data1$logAs   <- as.numeric(as.character(data1$logAs))
data1$logRate <- as.numeric(as.character(data1$logRate))
data1$weight  <- as.numeric(as.character(data1$weight))
data1$smkgrp  <- as.numeric(as.character(data1$smkgrp))
data1$adjinc  <- c(as.numeric(coef(r1)[2])*r1$model[,2]*0,
                   as.numeric(coef(r2)[2])*r2$model[,2]*0)
data1$adjrate <- data1$adjinc+data1$logRate

#This creates the graph used for figure 2 on the left, this represents the ars/smoking
#interaction not adjusted for SES

assmkp <- ggplot(data1, aes(x=logAs, y=adjrate, shape=factor(smkgrp),
                            color=factor(smkgrp)))
assmkp + stat_smooth(method = "glm", level=0.95, alpha=1, fill="grey80", color="black") +
             geom_point(aes(size=weight)) +
             geom_point() +
             theme(legend.position = "right") +
             theme_bw()
```

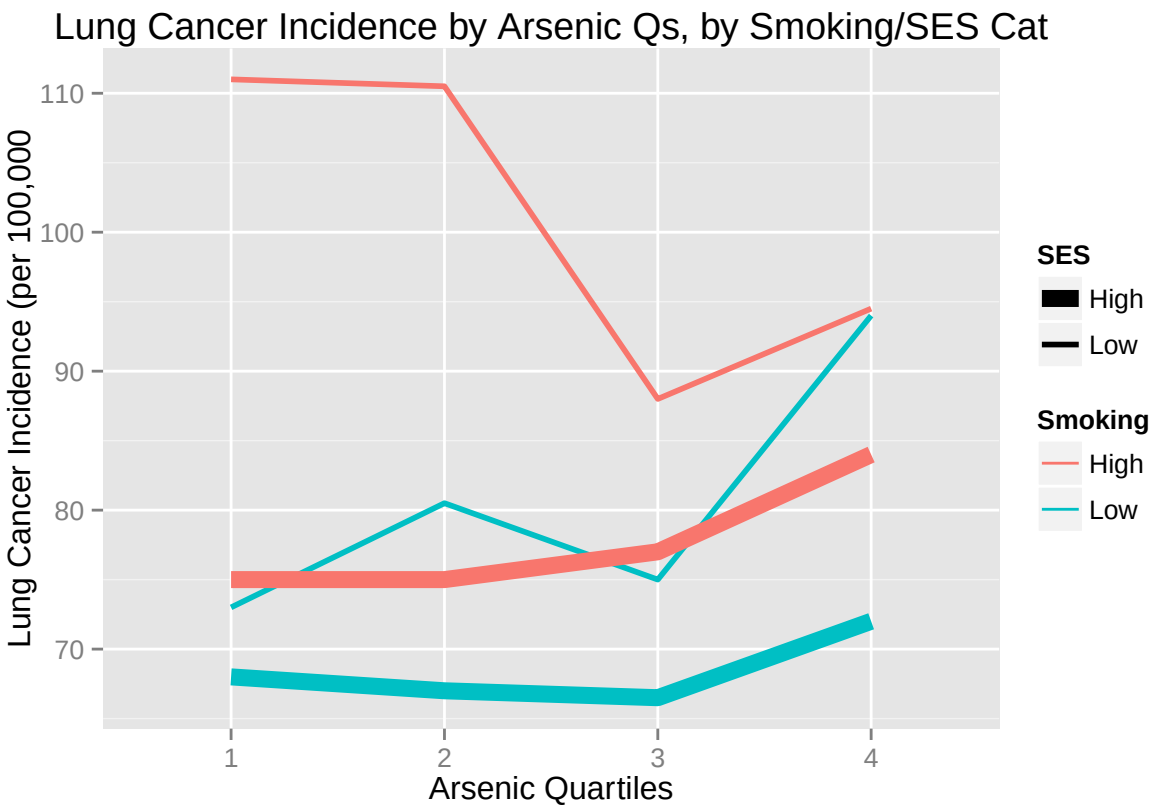



```
# These variables are created for the raw data graphs illustrating interactions
full$smkgrpf <- full$smkgrpfbak
full$smkgrpf <- ifelse(is.na(full$smkgrpf), NA,
                       ifelse(full$smkgrpf==1 | full$smkgrpf==2, 1, 2))
full$SESgrp2f <- full$SESgrpf
full$SESgrp2f <- ifelse(is.na(full$SESgrp2f), NA, ifelse(full$SESgrp2f==1 | full$SESgrp2f==2, 1, 2))
```

```
# This creates the combined SES and smoking group into one variable, categories 1,2
# are the low smoking categories, important to note that the SES categories are not
# quartiles

full$sessmk = ifelse(full$SESgrp2f==1 & full$smkgrpf==1,"1) Low SES, Low Smoke",
              ifelse(full$SESgrp2f==2 & full$smkgrpf==1,"2) High SES, Low Smoke",
              ifelse(full$SESgrp2f==1 & full$smkgrpf==2,"3) Low SES, High Smoke",
              ifelse(full$SESgrp2f==2 & full$smkgrpf==2,"4) High SES, High Smoke",NA))))

# This is the graph with only two lines for the two smoking groups with actual data

full$smklabel <- ifelse(is.na(full$smkgrpf), NA,
                        ifelse(full$smkgrpf==1,'1) Low Smoke', '2) High Smoke'))

graphfile <- full %>%
             group_by(AsQf,smklabel) %>%
             summarise(meanrate = median(AdjRate) )

finalgraph <- graphfile[!is.na(graphfile$AsQf) & !is.na(graphfile$smklabel),]

ggplot(finalgraph, aes(x=AsQf, y=meanrate, color=as.factor(smklabel))) +
         geom_line(aes(group=smklabel, title='Smoking'), size=1) +
         ggtitle("Lung Cancer Incidence by Arsenic Quartiles, by Smoking Cat") +
         xlab("Arsenic Quartiles") +
         ylab("Lung Cancer Incidence (per 100,000")
```



```r
# This graph creates four lines for the combination of SES and smoking

graphfile <- full %>%
             group_by(AsQf,sessmk) %>%
             summarise(meanrate = median(AdjRate) )

finalgraph <- graphfile[!is.na(graphfile$AsQf) & !is.na(graphfile$sessmk),]

ggplot(finalgraph,
       aes(x=AsQf, y=meanrate,
      color= sessmk == "1) Low SES, Low Smoke" | sessmk == "2) High SES, Low Smoke",
      size= sessmk == "1) Low SES, Low Smoke" | sessmk == "3) Low SES, High Smoke"
      ) )          +
         geom_line(aes(group=factor(sessmk))) +
         scale_colour_manual(
           name='Smoking',
           values = setNames(c('#00BFC4','#F8766D'),c(T,F)),
           labels=(c('High','Low'))) +
         scale_size_manual(
           name='SES',
           values = setNames(c(1,3),c(T,F)),
           labels=(c('High','Low'))) +
         ggtitle("Lung Cancer Incidence by Arsenic Qs, by Smoking/SES Cat") +
         xlab("Arsenic Quartiles") +
         ylab("Lung Cancer Incidence (per 100,000")
```



```
#ggplot(np_graph) + geom_point(aes(x = C1, y = C2, colour = C1 >0)) +
#  scale_colour_manual(name = 'PC1 > 0', values = setNames(c('red','green'),c(T, F))) +
#  xlab('PC1') + ylab('PC2')
```

This section creates Figure 2 right side. This graph is again two regressions that create fitted points for two smoking groups to create two lines, but this time adjusted for SES. The points are now generated from a model with two independent variables, but only plotted against two dimensions of the model, so the points do not line up.

```
## GLMS for smoking levels WITH SES
smkgrpf <- smkgrpfbak
## Bottom 50% vs Top 50%
smkgrpf <- ifelse(is.na(smkgrpf), NA, ifelse(smkgrpf==1 | smkgrpf==2, 1, 2))

r1 <- glm(full[smkgrpf==1,]$AdjRate ~ full[smkgrpf==1,]$lnAs +
            full[smkgrpf==1,]$MedIncome, family=poisson,
            weights=as.numeric(full[smkgrpf==1,]$Population))
summary(r1)
```

```
Call:
glm(formula = full[smkgrpf == 1, ]$AdjRate ~ full[smkgrpf ==
    1, ]$lnAs + full[smkgrpf == 1, ]$MedIncome, family = poisson,
    weights = as.numeric(full[smkgrpf == 1, ]$Population))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4941.6   -151.1    150.1    425.7   2249.4

Coefficients:
                                    Estimate      Std. Error z value
(Intercept)                     4.325647218667  0.000071461711 60531.0
full[smkgrpf == 1, ]$lnAs       0.007596316777  0.000033284145   228.2
full[smkgrpf == 1, ]$MedIncome -0.000003861662  0.000000001486 -2599.0
                                           Pr(>|z|)
(Intercept)                    <0.0000000000000002 ***
full[smkgrpf == 1, ]$lnAs      <0.0000000000000002 ***
full[smkgrpf == 1, ]$MedIncome <0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 175916938  on 289  degrees of freedom
Residual deviance: 168750465  on 287  degrees of freedom
  (167 observations deleted due to missingness)
AIC: 536459750

Number of Fisher Scoring iterations: 4
```

```
r2 <- glm(full[smkgrpf==2,]$AdjRate ~ full[smkgrpf==2,]$lnAs +
            full[smkgrpf==2,]$MedIncome, family=poisson,
```

```
            weights=as.numeric(full[smkgrpf==2,]$Population))
summary(r2)
```

```
Call:
glm(formula = full[smkgrpf == 2, ]$AdjRate ~ full[smkgrpf ==
    2, ]$lnAs + full[smkgrpf == 2, ]$MedIncome, family = poisson,
    weights = as.numeric(full[smkgrpf == 2, ]$Population))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1769.12   -244.11     -9.96    270.72   1591.53

Coefficients:
                                    Estimate      Std. Error z value
(Intercept)                     4.688953956476  0.000113495701   41314
full[smkgrpf == 2, ]$lnAs       0.059353912842  0.000043115820    1377
full[smkgrpf == 2, ]$MedIncome -0.000007984707  0.000000002792   -2860
                                          Pr(>|z|)
(Intercept)                    <0.0000000000000002 ***
full[smkgrpf == 2, ]$lnAs      <0.0000000000000002 ***
full[smkgrpf == 2, ]$MedIncome <0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 68053265  on 294  degrees of freedom
Residual deviance: 57437520  on 292  degrees of freedom
  (162 observations deleted due to missingness)
AIC: 197334525

Number of Fisher Scoring iterations: 4
```

```
## Plot the Interaction between Arsenic and Smoking with SES
data1 <- cbind( c(t(r1$model[2]),t(r2$model[2])),               # ln arsenic values
                        c(log(r1$fitted.values),log(r2$fitted.values)), # fitted dependent vars
                        c(r1$weights, r2$weights),                      # county populations
                        c(rep("1",dim(r1$model[2])[1]), rep("2",dim(r2$model[2])[1])))
                #smoking groups

data1         <- as.data.frame(data1, stringsAsFactors=FALSE)
names(data1)  <- c("logAs","logRate", "weight", "smkgrp")
data1$logAs   <- as.numeric(as.character(data1$logAs))
data1$logRate <- as.numeric(as.character(data1$logRate))
data1$weight  <- as.numeric(as.character(data1$weight))
data1$smkgrp  <- as.numeric(as.character(data1$smkgrp))
data1$adjinc  <- c(as.numeric(coef(r1)[2])*r1$model[,2],
                   as.numeric(coef(r2)[2])*r2$model[,2])
data1$adjrate <- data1$adjinc+data1$logRate

assmkp <- ggplot(data1, aes(x=logAs, y=adjrate, shape=factor(smkgrp), color=factor(smkgrp)))
assmkp + stat_smooth(method = "glm", level=0.95, alpha=1, fill="grey80", color="black") +
```

```
        geom_point(aes(size=weight)) +
    geom_point() +
        theme(legend.position = "right") +
        theme_bw()
```

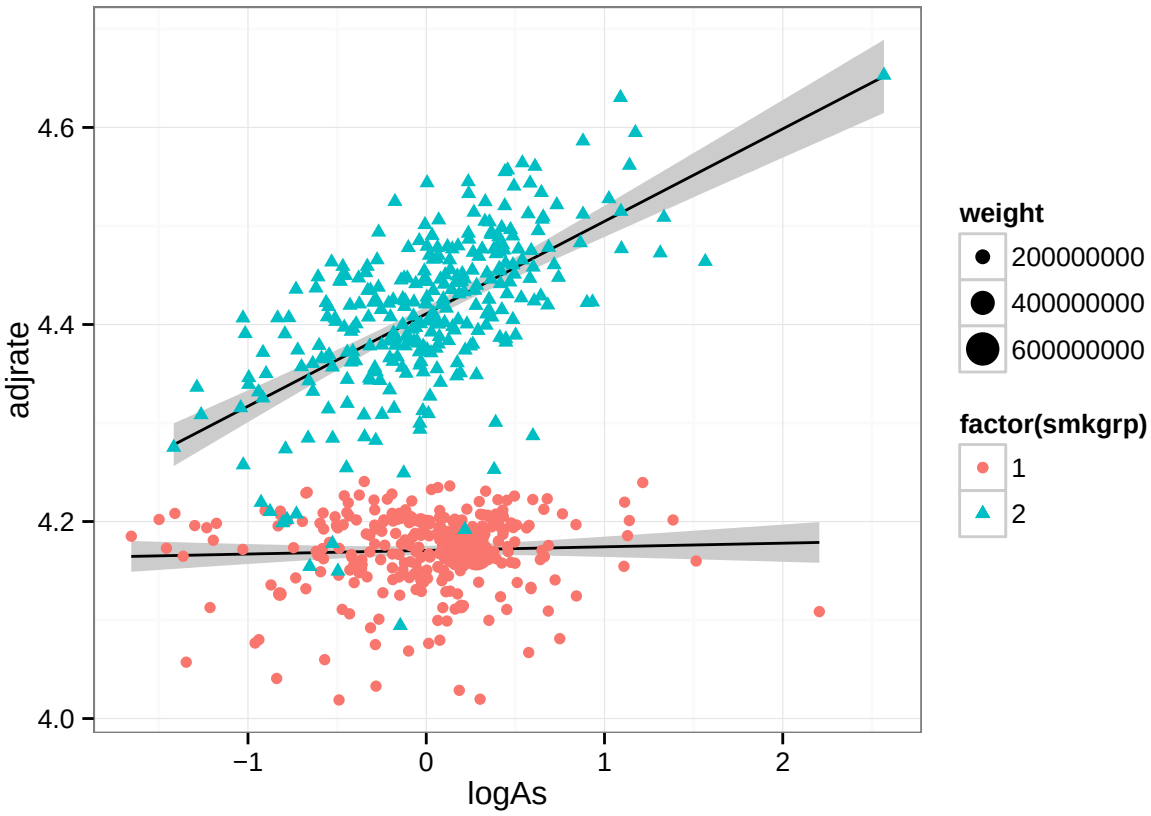


Table 3

Summary of ANOVA tests performed between Arsenic and covariates used in the regression analysis.

```
Interaction_Pair = c("Arsenic:Smoking", "Arsenic:MCI")
DF = c(3, 9)
F_value = c(2.6595, 1.7798)
P_value = c(0.04747, 0.06914)
df = data.frame(Interaction_Pair, DF, F_value, P_value)
kable(df)
```

Interaction_Pair	DF	F_value	P_value
Arsenic:Smoking	3	2.6595	0.04747
Arsenic:MCI	9	1.7798	0.06914

```
# change values in table to variables
```

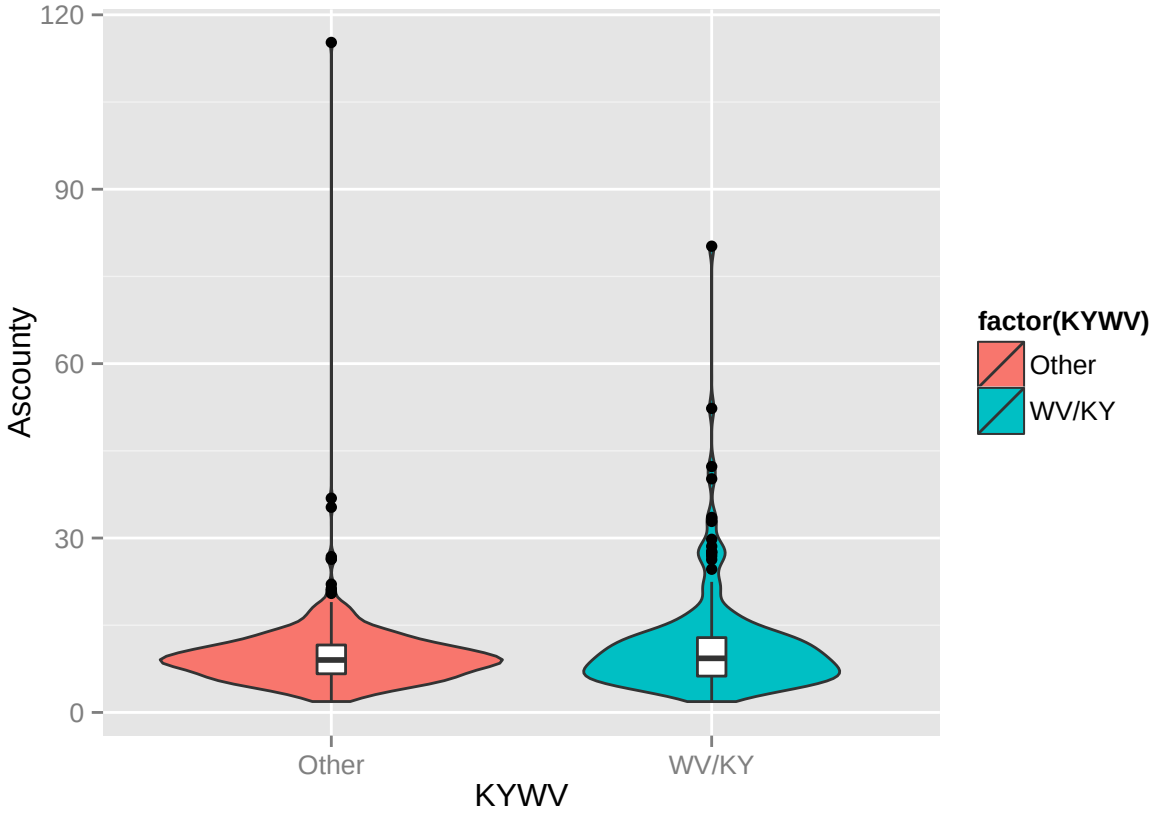
Figure 3

Combination violin and boxplots showing the average level of exposure and outcomes for counties in West Virginia or Kentucky comared with the remaining 10 states in the original sample.

Instead of the bar plot, we represented the data in Figure 3 with a series of combination violin and boxplots. The authors of the original paper attempted to represent each of the variables of interest for this data on the same y-axis scale, including arsenic, income, lung cancer incidence rates, and smoking. Although the authors likely chose this visualization to save space, we decided that it was somewhat confusing and that additional information about the data distributions could be added if we instead used a combination of violin and boxplots.

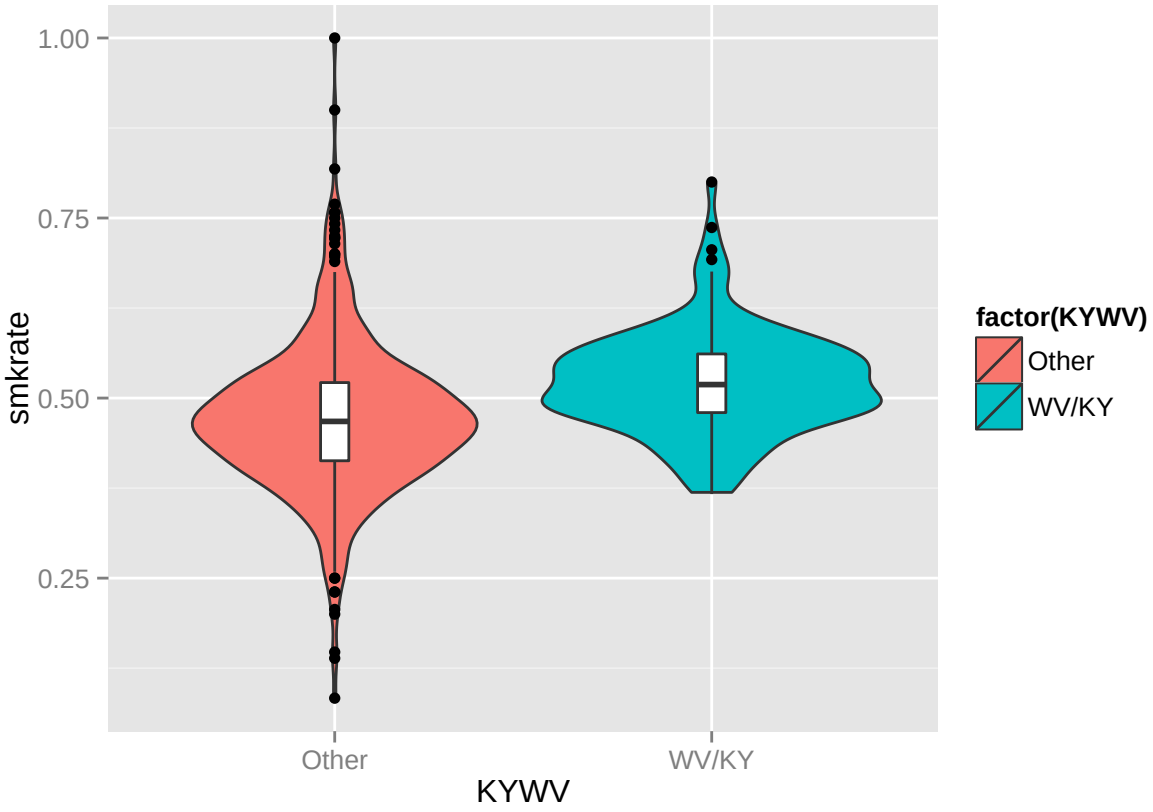
Explanation of variables here? Add x label (location) and y label (arsenic, income, etc.)

```
# arsenic
KYWVas <- na.omit(full[full$SFIPS == 21 | full$SFIPS == 54, ]$Ascounty)
notKYWVas <- na.omit(full[full$SFIPS != 21 & full$SFIPS != 54, ]$Ascounty)
KYWVdf <- KYWVas %>% as.data.frame() %>% mutate(KYWV = "WV/KY") %>% select(Ascounty = 1,
    KYWV = 2)
notKYWVdf <- notKYWVas %>% as.data.frame() %>% mutate(KYWV = "Other") %>% select(Ascounty = 1,
    KYWV = 2)
KYWVcombined = merge(KYWVdf, notKYWVdf, all = TRUE)
KYWVcombined %>% ggplot(aes(x = KYWV, y = Ascounty)) + geom_violin(aes(fill = factor(KYWV))) +
    geom_boxplot(width = 0.1)
```

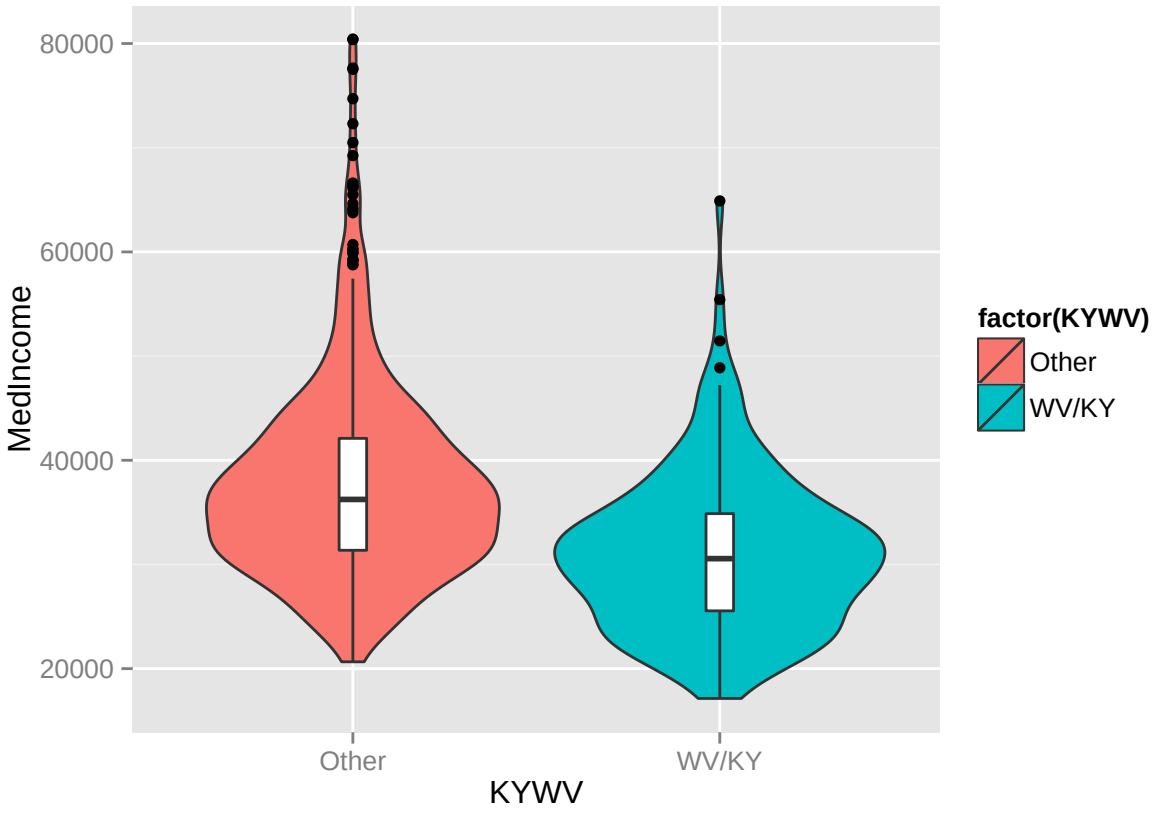



```
# smoking
KYWVsmk <- na.omit(full[full$SFIPS == 21 | full$SFIPS == 54, ]$smkrate)
notKYWVsmk <- na.omit(full[full$SFIPS != 21 & full$SFIPS != 54, ]$smkrate)
```

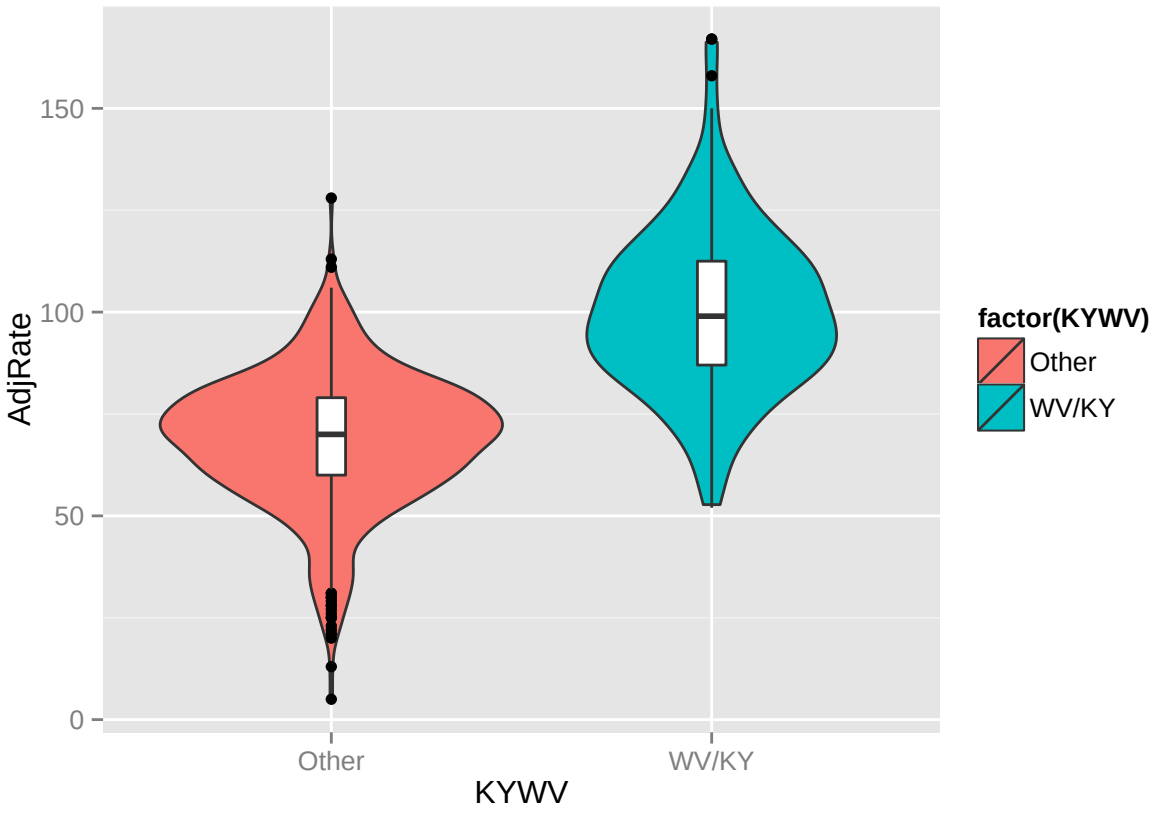
```
KYWVdf <- KYWVsmk %>% as.data.frame() %>% mutate(KYWV = "WV/KY") %>% select(smkrate = 1,
    KYWV = 2)
notKYWVdf <- notKYWVsmk %>% as.data.frame() %>% mutate(KYWV = "Other") %>% select(smkrate = 1,
    KYWV = 2)
KYWVcombined = merge(KYWVdf, notKYWVdf, all = TRUE)
KYWVcombined %>% ggplot(aes(x = KYWV, y = smkrate)) + geom_violin(aes(fill = factor(KYWV))) +
    geom_boxplot(width = 0.1)
```




```
# income
KYWVmed <- na.omit(full[full$SFIPS == 21 | full$SFIPS == 54, ]$MedIncome)
notKYWVmed <- na.omit(full[full$SFIPS != 21 & full$SFIPS != 54, ]$MedIncome)
KYWVdf <- KYWVmed %>% as.data.frame() %>% mutate(KYWV = "WV/KY") %>% select(MedIncome = 1,
    KYWV = 2)
notKYWVdf <- notKYWVmed %>% as.data.frame() %>% mutate(KYWV = "Other") %>% select(MedIncome = 1,
    KYWV = 2)
KYWVcombined = merge(KYWVdf, notKYWVdf, all = TRUE)
KYWVcombined %>% ggplot(aes(x = KYWV, y = MedIncome)) + geom_violin(aes(fill = factor(KYWV))) +
    geom_boxplot(width = 0.1)
```




```
# lung cancer incidence rate
KYWVrate <- na.omit(full[full$SFIPS == 21 | full$SFIPS == 54, ]$AdjRate)
notKYWVrate <- na.omit(full[full$SFIPS != 21 & full$SFIPS != 54, ]$AdjRate)
KYWVdf <- KYWVrate %>% as.data.frame() %>% mutate(KYWV = "WV/KY") %>% select(AdjRate = 1,
    KYWV = 2)
notKYWVdf <- notKYWVrate %>% as.data.frame() %>% mutate(KYWV = "Other") %>%
    select(AdjRate = 1, KYWV = 2)
KYWVcombined = merge(KYWVdf, notKYWVdf, all = TRUE)
KYWVcombined %>% ggplot(aes(x = KYWV, y = AdjRate)) + geom_violin(aes(fill = factor(KYWV))) +
    geom_boxplot(width = 0.1)
```

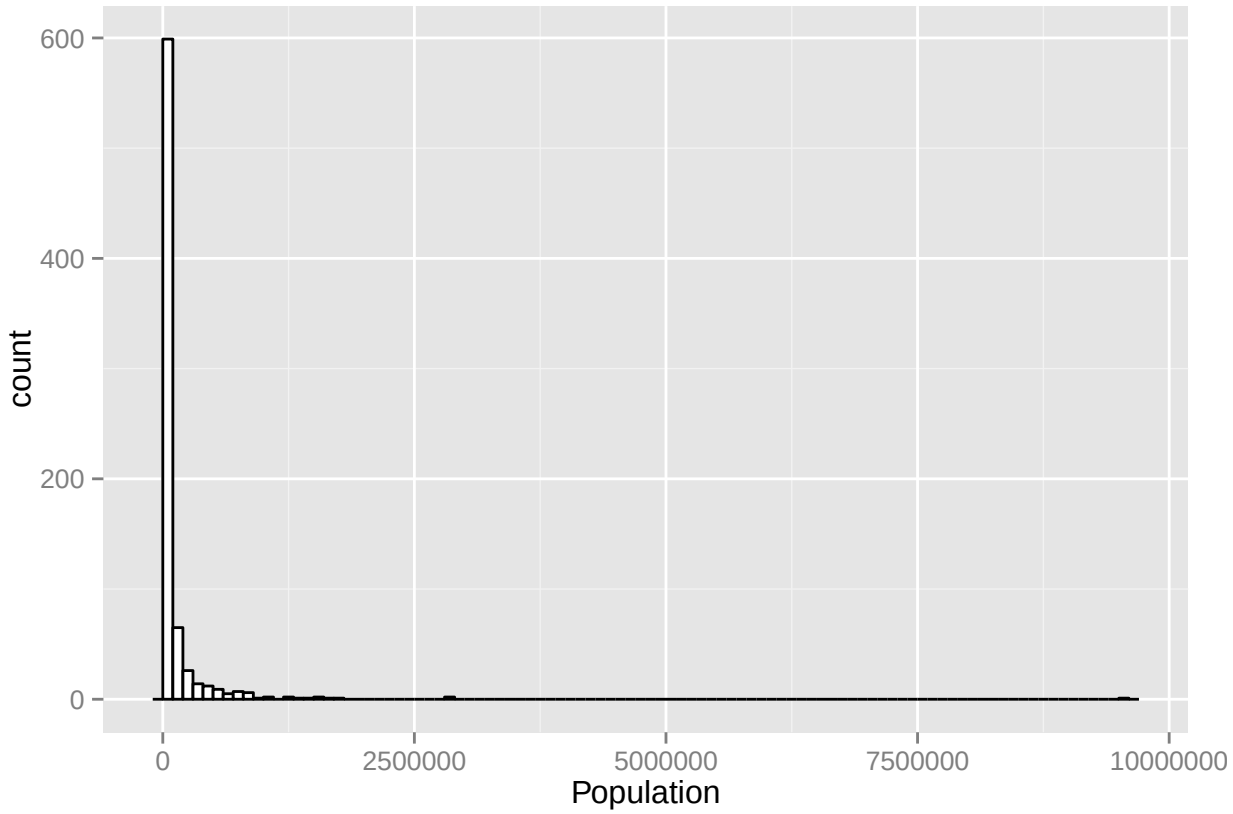


Extension

```
fips <- read.table("~/SoftwareProjects/Probability/final_proj/FIPS.csv", header = TRUE,
    sep = ",")

full <- left_join(full, fips, by = "SFIPS")

## Generate centered and transformed variables
full$lnAs <- log(full$Ascounty) - mean(na.omit(log(full$Ascounty)))
full$lnInc <- log(full$MedIncome) - mean(na.omit(log(full$MedIncome)))
full$Population <- as.numeric(as.character(full$Population))
full$lnsmk <- full$smkrate
full$lnar <- log(full$AdjRate)

summary(full$Population)
```

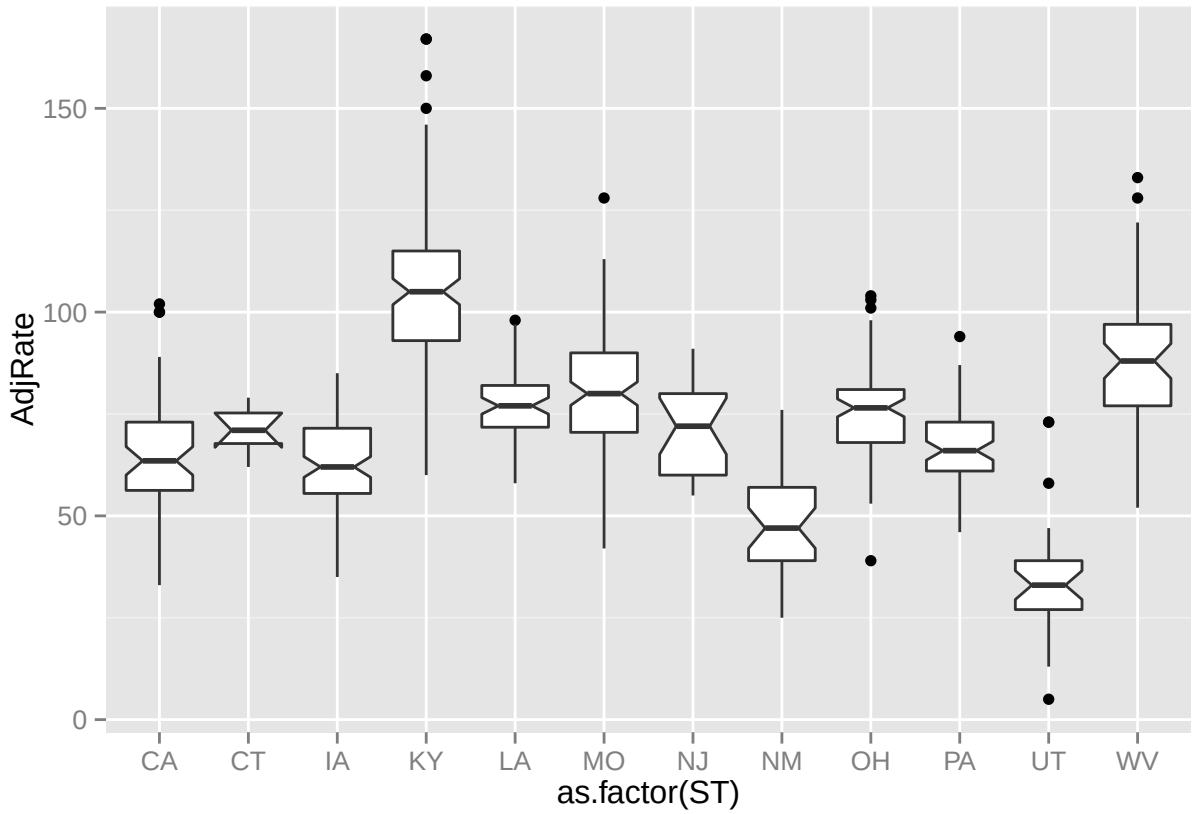
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    810   14390   29980  121800   81870 9519000
```

```
ggplot(full, aes(x = Population)) + geom_histogram(binwidth = 100000, fill = "white",
    colour = "black")
```




```
ggplot(full, aes(y = AdjRate, x = as.factor(ST))) + geom_boxplot(notch = T)
```

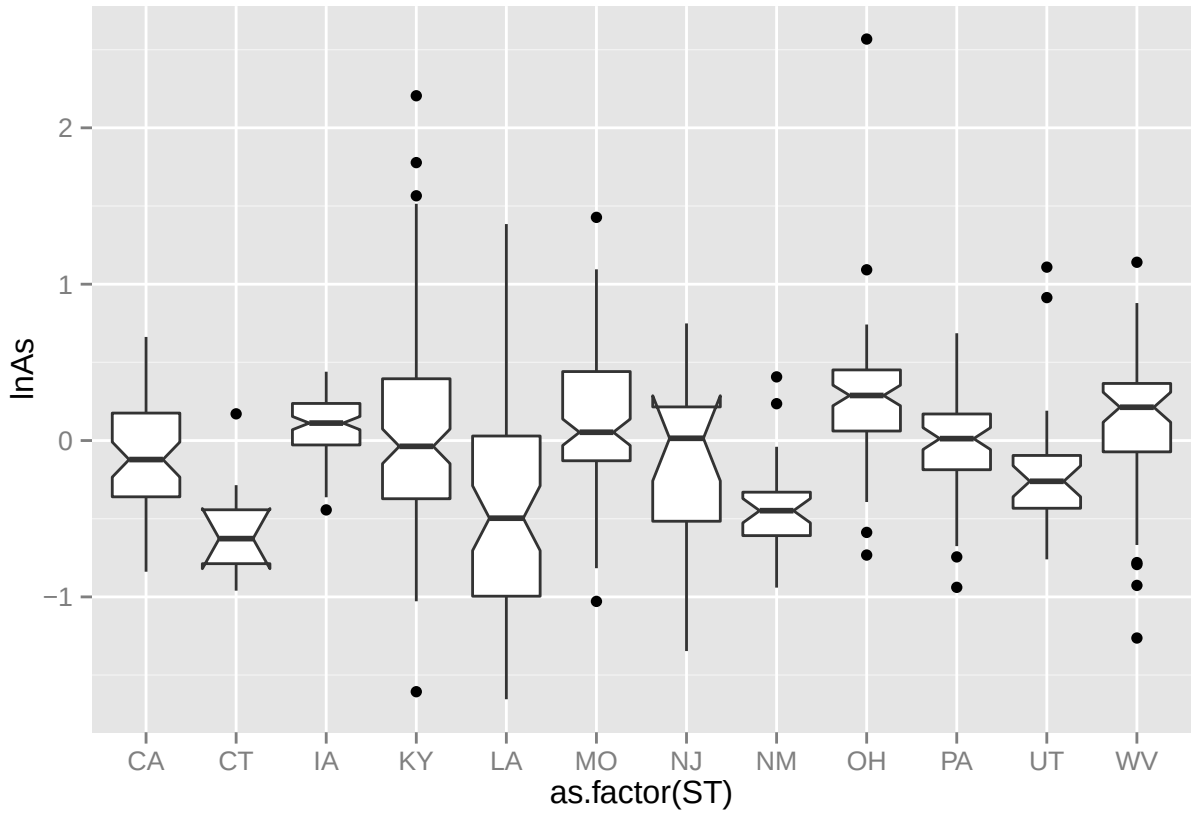
```
notch went outside hinges. Try setting notch=FALSE.
```




```
ggplot(full, aes(y = lnAs, x = as.factor(ST))) + geom_boxplot(notch = T)
```

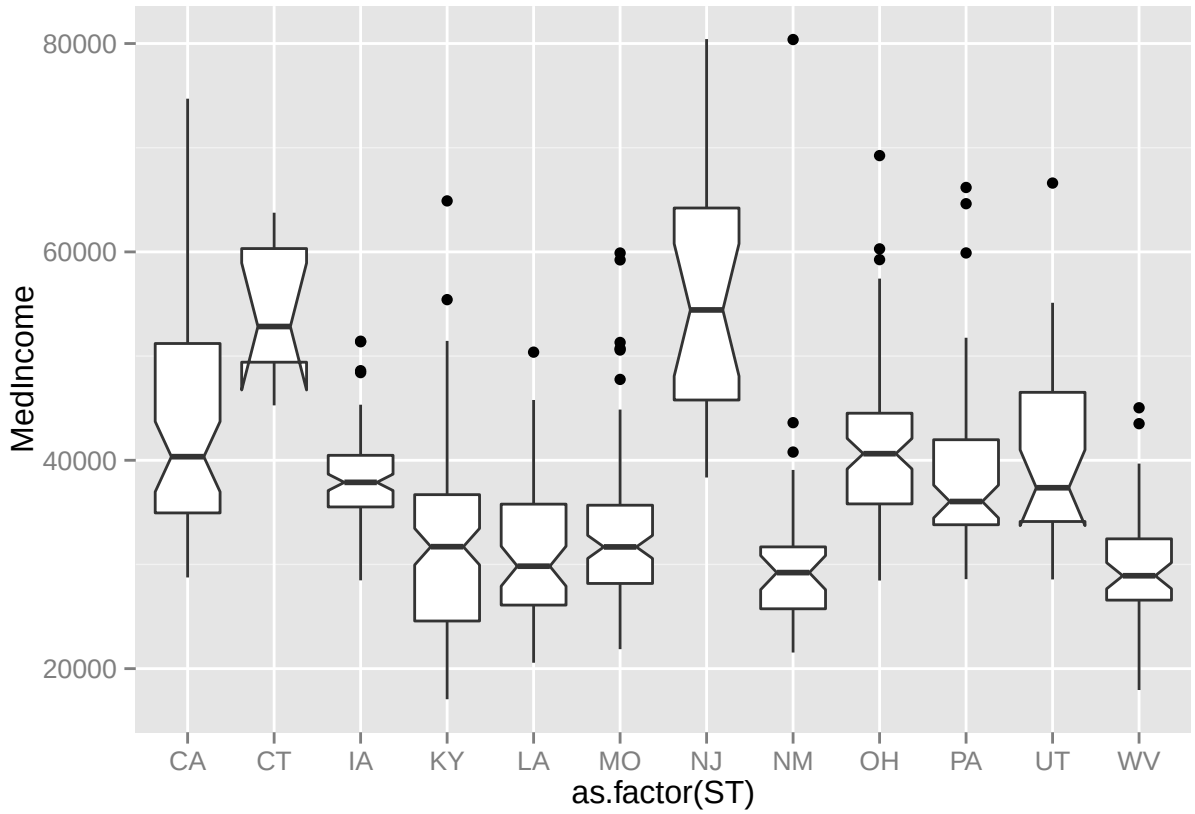
```
Warning: Removed 15 rows containing non-finite values (stat_boxplot).
```

```
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
```




```
ggplot(full, aes(y = MedIncome, x = as.factor(ST))) + geom_boxplot(notch = T)
```

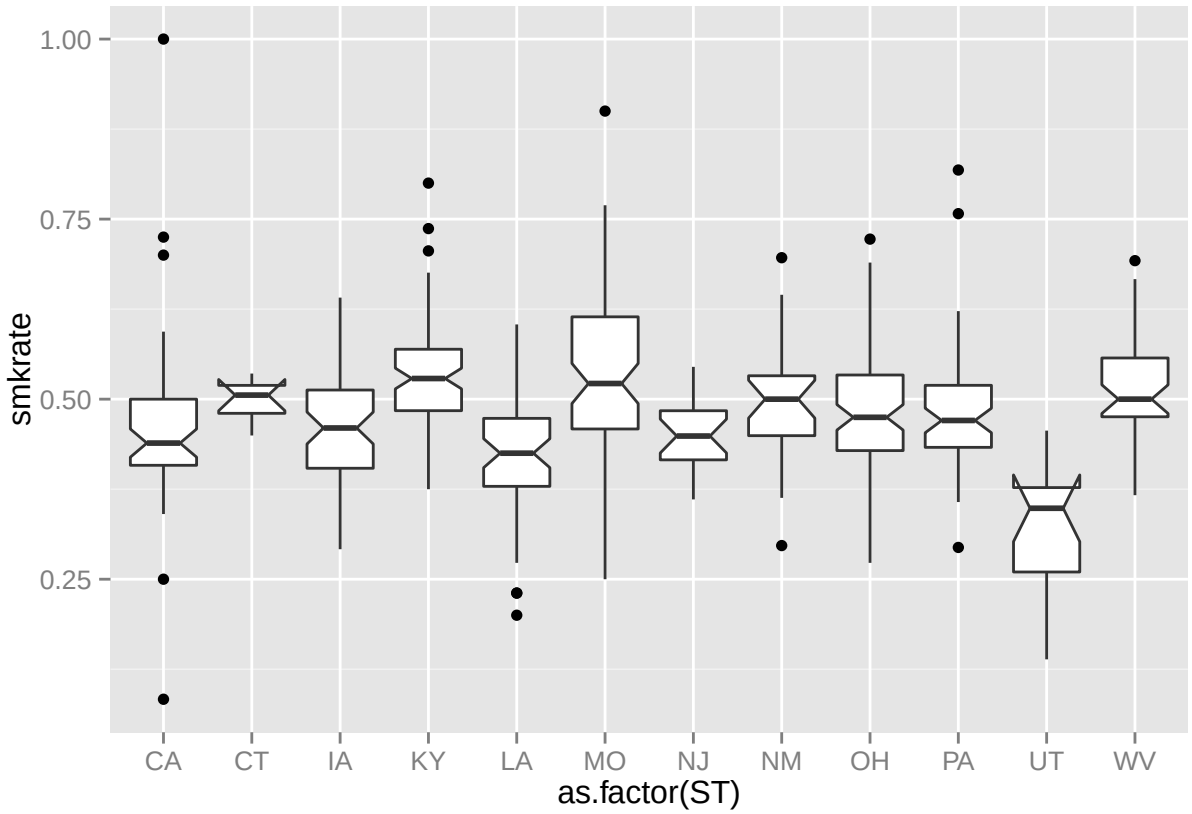
```
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
```




```
ggplot(full, aes(y = smkrate, x = as.factor(ST))) + geom_boxplot(notch = T)
```

```
Warning: Removed 157 rows containing non-finite values (stat_boxplot).
```

```
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
```




```
full$urban <- ifelse(full$Population < 1000000, "Rural", "Urban")
table(full$urban)
```

```
Rural Urban
  744    13
```

```
t.test(full$AdjRate[full$urban == "Rural"], full$AdjRate[full$urban == "Urban"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$AdjRate[full$urban == "Rural"] and full$AdjRate[full$urban == "Urban"]
t = 2.7257, df = 13.561, p-value = 0.01679
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  1.896362 16.102398
sample estimates:
mean of x mean of y
 76.15323  67.15385
```

```
t.test(full$Ascounty[full$urban == "Rural"], full$Ascounty[full$urban == "Urban"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$Ascounty[full$urban == "Rural"] and full$Ascounty[full$urban == "Urban"]
t = -0.2301, df = 12.324, p-value = 0.8218
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.655347  2.146725
sample estimates:
mean of x mean of y
 10.09131  10.34562
```

```
t.test(full$MedIncome[full$urban == "Rural"], full$MedIncome[full$urban == "Urban"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$MedIncome[full$urban == "Rural"] and full$MedIncome[full$urban == "Urban"]
t = -2.8831, df = 12.274, p-value = 0.01348
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -16547.359  -2322.385
sample estimates:
mean of x mean of y
 36047.05  45481.92
```

```
t.test(full$smkrate[full$urban == "Rural"], full$smkrate[full$urban == "Urban"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$smkrate[full$urban == "Rural"] and full$smkrate[full$urban == "Urban"]
t = 3.53, df = 13.078, p-value = 0.003664
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01972434 0.08184947
sample estimates:
mean of x mean of y
0.4834057 0.4326188
```

```
full$geo <- ifelse(full$SFIPS == 6 | full$SFIPS == 35 | full$SFIPS == 49, "West",
    "TheRest")
table(full$geo)
```

```
TheRest    West
    637     120
```

```
t.test(full$AdjRate[full$geo == "West"], full$AdjRate[full$geo == "TheRest"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$AdjRate[full$geo == "West"] and full$AdjRate[full$geo == "TheRest"]
t = -13.994, df = 169.92, p-value < 0.00000000000000022
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -30.45205 -22.92301
sample estimates:
mean of x mean of y
 53.54167  80.22920
```

```
t.test(full$Ascounty[full$geo == "West"], full$Ascounty[full$geo == "TheRest"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$Ascounty[full$geo == "West"] and full$Ascounty[full$geo == "TheRest"]
t = -6.2212, df = 338.17, p-value = 0.000000001458
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.623611 -1.882647
sample estimates:
mean of x mean of y
 7.780126 10.533255
```

```
t.test(full$MedIncome[full$geo == "West"], full$MedIncome[full$geo == "TheRest"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$MedIncome[full$geo == "West"] and full$MedIncome[full$geo == "TheRest"]
t = 3.0104, df = 147.47, p-value = 0.00307
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1157.472 5580.735
sample estimates:
mean of x mean of y
 39044.11  35675.00
```

```
t.test(full$smkrate[full$geo == "West"], full$smkrate[full$geo == "TheRest"],
    paired = F, weights = as.numeric(full$Population))
```

```
    Welch Two Sample t-test

data:  full$smkrate[full$geo == "West"] and full$smkrate[full$geo == "TheRest"]
t = -3.5014, df = 117.38, p-value = 0.0006554
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
```

```
 -0.07176368 -0.01991206
sample estimates:
mean of x mean of y
0.4438862 0.4897241
```

Rural vs. Urban

```
Variable = c("Lung_Cancer_Rate", "Arsenic", "Median Income", "Smoking Rate")
t = c(2.7257, -0.2301, -2.8831, 3.53)
P_value = c(0.01679, 0.8218, 0.01348, 0.003664)
Rural = c(76.15323, 10.09131, 36047.05, 0.4834057)
Urban = c(67.15385, 10.34562, 45481.92, 0.4326188)
df = data.frame(Variable, t, P_value, Rural, Urban)
kable(df)
```

Variable	t	P_value	Rural	Urban
Lung_Cancer_Rate	2.7257	0.016790	76.1532300	67.1538500
Arsenic	-0.2301	0.821800	10.0913100	10.3456200
Median Income	-2.8831	0.013480	36047.0500000	45481.9200000
Smoking Rate	3.5300	0.003664	0.4834057	0.4326188

West vs. Rest

```
Variable = c("Lung_Cancer_Rate", "Arsenic", "Median Income", "Smoking Rate")
t = c(-13.994, -6.2212, 3.0104, -3.5014)
P_value = c("< 0.00001", "< 0.00001", 0.00307, 0.0006554)
West = c(53.54167, 7.780126, 39044.11, 0.4438862)
Rest = c(80.2292, 10.533255, 35675, 0.4897241)
df = data.frame(Variable, t, P_value, West, Rest)
kable(df)
```

Variable	t	P_value	West	Rest
Lung_Cancer_Rate	-13.9940	< 0.00001	53.5416700	80.2292000
Arsenic	-6.2212	< 0.00001	7.7801260	10.5332550
Median Income	3.0104	0.00307	39044.1100000	35675.0000000
Smoking Rate	-3.5014	0.0006554	0.4438862	0.4897241