# State_and_indiv

Courtney

4/5/2021

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 4.0.4
```

```r
breaches <- read_csv('C:/Users/student/Documents/SYS 2202/cyber-security-final/courtney-final-variables/
                     col_types = cols(
                      State = col_factor(),
                      Individuals_Affected = col_integer()
                     )
                  )
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

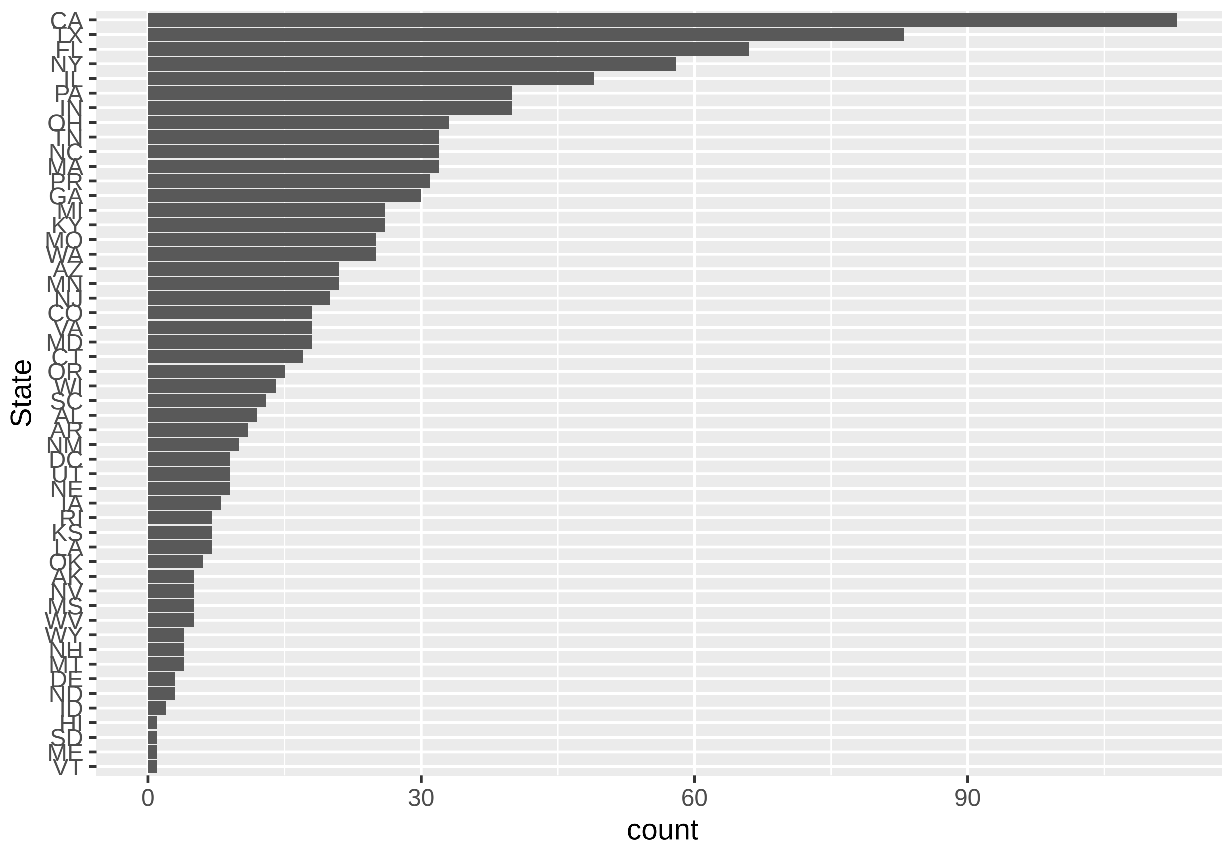
```r
head(breaches)
```

```
## # A tibble: 6 x 14
##      X1 Number Name_of_Covered_Entity  State Business_Associat~ Individuals_Aff~
##   <dbl>  <dbl> <chr>                   <fct> <chr>                         <int>
## 1     1      0 Brooke Army Medical Ce~ TX    <NA>                           1000
## 2     2      1 Mid America Kidney Sto~ MO    <NA>                           1000
## 3     3      2 Alaska Department of H~ AK    <NA>                            501
## 4     4      3 Health Services for Ch~ DC    <NA>                           3800
## 5     5      4 L. Douglas Carlson, M.~ CA    <NA>                           5257
## 6     6      5 David I. Cohen, MD      CA    <NA>                            857
## # ... with 8 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>
```

## State Variable

### Visualizing Distribution of State Variable

```
state_bar <- breaches %>%
  mutate(State = State %>% fct_infreq() %>% fct_rev()) %>%
  ggplot(aes(x=State)) +
  geom_bar()+
  coord_flip()

state_bar
```




```
count_state <- breaches %>%
  mutate(State = State %>% fct_infreq() %>% fct_rev()) %>%
  count(State)

count_state
```

```
## # A tibble: 52 x 2
##    State     n
##  * <fct> <int>
##  1 VT        1
##  2 ME        1
```

```
##   3 SD         1
##   4 HI         1
##   5 ID         2
##   6 ND         3
##   7 DE         3
##   8 MT         4
##   9 NH         4
##  10 WY         4
## # ... with 42 more rows
```

**- Which values are the most common? Why?**

Breaches in the State of California are the most common since they have the most breaches at 113. This is most likely due to the fact that California is highly populated with lots of buisnesses and tech industries, therefore can have more opportunities for breaches.

**- Which values are rare? Why? Does that match your expectations?** The most rare values are VT, ME, SD, and HI which all have only one breach. Since these are not very largely populated states this does make sense.

**- Can you see any unusual patterns? What might explain them?**

There does not appear to be any unusal patterns in the State breach count. Some states have more breaches than others but there is not any outliers of cycles of number of breaches.

**- Are there clusters in the data? If so,** No there are no clusters in the data, all of the data is relatively evenly distributed.

**- How are the observations within each cluster similar to or different from each other?**

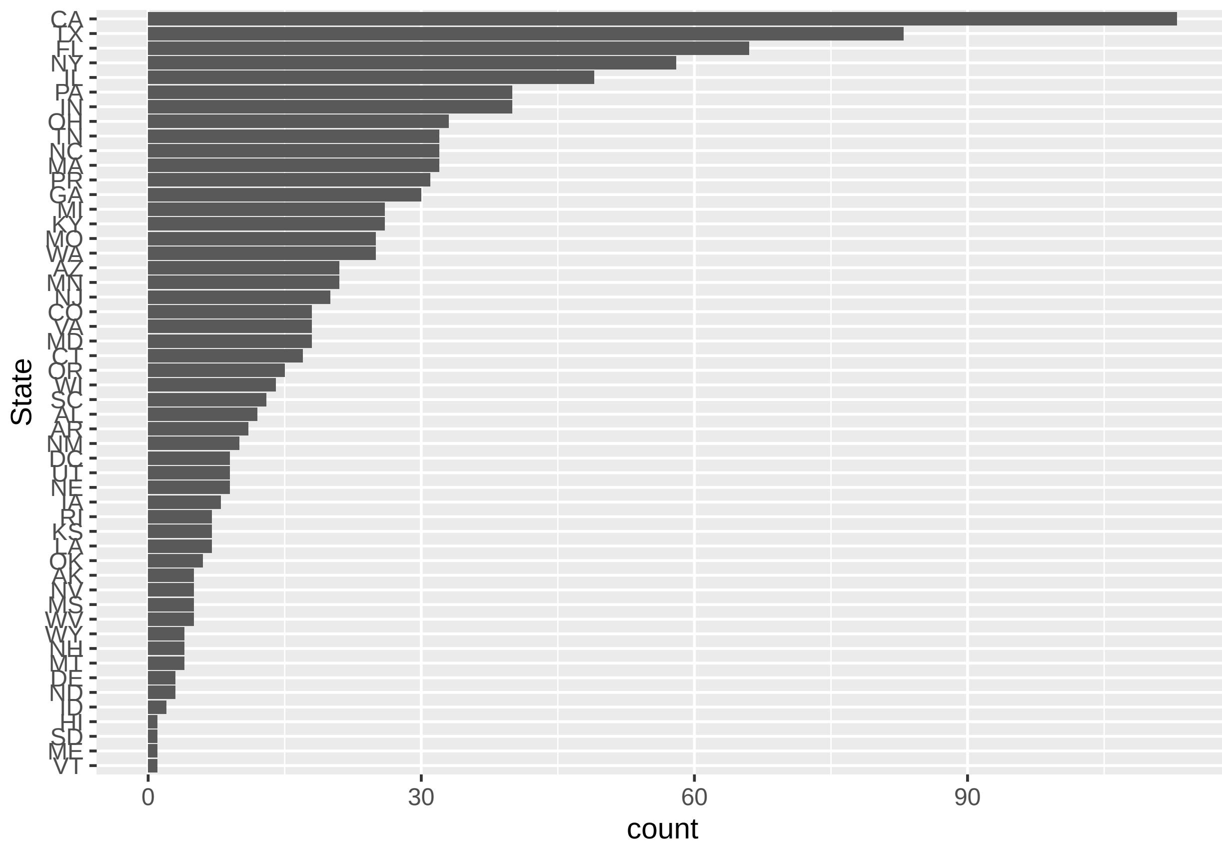
As mentioned above there are no clusters present.

**- How can you explain or describe the clusters?**

As mentioned above there are no clusters present.

**3.1.2.2 Unusual values (2 points) - Describe and demonstrate how you determine if there are unusual values in the data. E.g. too large, too small, negative, etc.**

```
breaches %>%
  mutate(State = State %>% fct_infreq() %>% fct_rev()) %>%
  ggplot(aes(x=State)) +
  geom_bar()+
  coord_flip()
```




There were no negative state breaches and no values that were unexpectadly high or low. This is seen in the bar graph. More exploration has to be done to determine if any values should be removed.

**- Describe and demonstrate how you determine if they are outliers.**

An outlier is 1.5 times the interquartile range away from either the lower or upper quartile. In order to determine if any of the state count values are outliers the interquartile range, first quartile, and third quartile need to be calculated. The State count data then has to be filtered for values that are less than the first quartile minus the IQR times 1.5 and values that are greater than the third quartile plus the IQR times 1.5. The outliers can be seen in the outlier list data frame, it includes, TX, CA, FL.

```
state_count <- breaches %>%
  group_by(State) %>%
  count()
state_count
```

```
## # A tibble: 52 x 2
## # Groups:   State [52]
##    State     n
##    <fct> <int>
##  1 TX       83
##  2 MO       25
##  3 AK        5
##  4 DC        9
##  5 CA      113
##  6 PA       40
##  7 TN       32
```

```
## 8 NY       58
## 9 NC       32
## 10 MI      26
## # ... with 42 more rows
```

```
stdev <-  sd(state_count$n, na.rm = TRUE)
stdev
```

```
## [1] 21.85544
```

```
innerQ <-  IQR(state_count$n, na.rm = TRUE)
innerQ
```

```
## [1] 22
```

```
firstQ <- quantile(state_count$n, 0.25, na.rm = TRUE)
firstQ <- firstQ[[1]]

thirdQ <- quantile(state_count$n, 0.75, na.rm = TRUE)
thirdQ <- thirdQ[[1]]

outlier_list <- state_count %>%
  filter(n < (firstQ - innerQ * 1.5) |
        n > (thirdQ + innerQ * 1.5))

outlier_list
```

```
## # A tibble: 3 x 2
## # Groups:   State [3]
##   State     n
##   <fct> <int>
## 1 TX       83
## 2 CA      113
## 3 FL       66
```

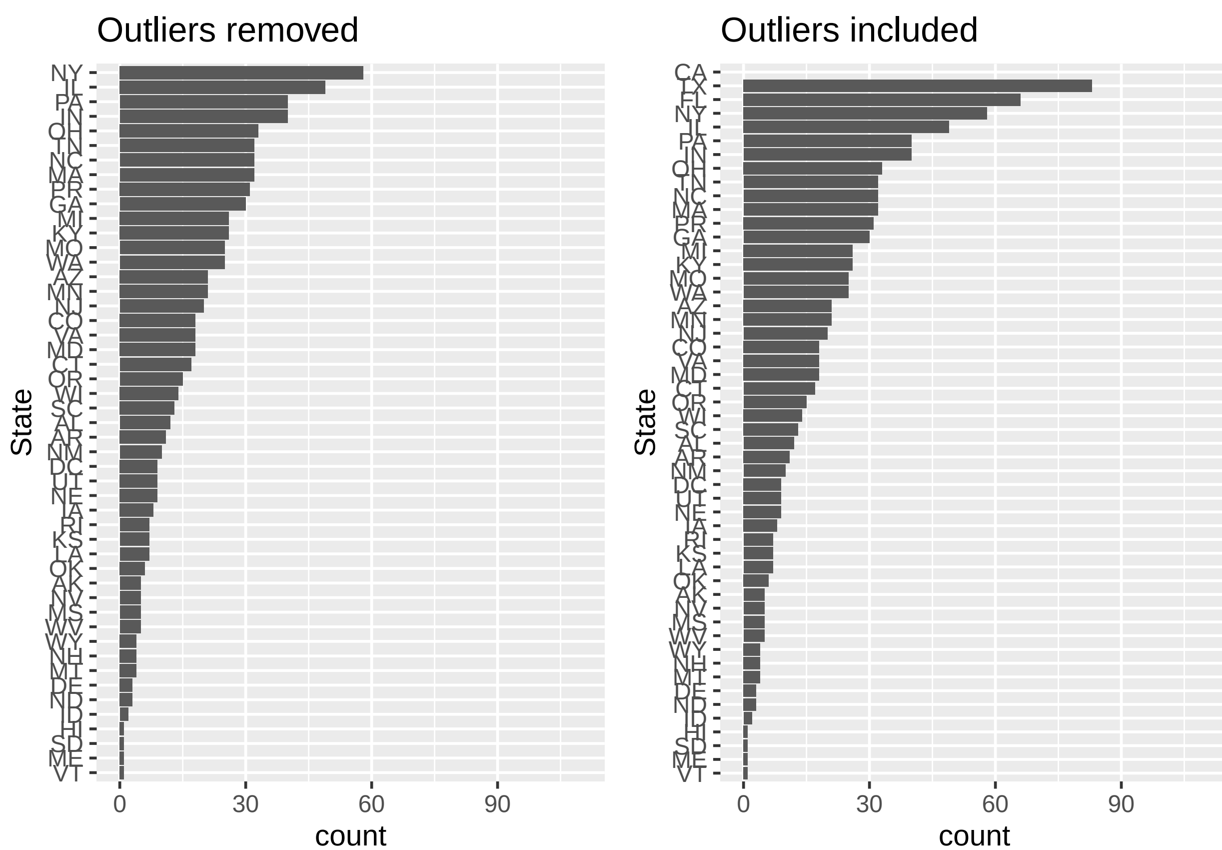
**- Show how do your distributions look like with and without the unusual values.**

With the outliers removed the distribution is made narrower with less variation. Since the largest state breach counts are removed overall the distribution becomes more similar throughout.

```
outlier_state = c("TX", "CA", "FL")

no_out_bar <- breaches %>%
  mutate(State = State %>% fct_infreq() %>% fct_rev()) %>%
  filter(!(State %in% outlier_state)) %>%
  ggplot(aes(x=State)) +
  geom_bar()+
  coord_flip()+
  ylim(0, 110) +
  labs(title = "Outliers removed")

out_in_bar <- breaches %>%
```

```
  mutate(State = State %>% fct_infreq() %>% fct_rev()) %>%
  ggplot(aes(x=State)) +
  geom_bar()+
  coord_flip()+
  ylim(0,110) +
  labs(title = "Outliers included")

ggarrange(no_out_bar, out_in_bar, ncol = 2)
```

```
## Warning: Removed 1 rows containing missing values (geom_bar).
```




**- Discuss whether or not you need to remove unusual values and why.**

Since the largest values will provide d the most insight into why breaches are happeing at such a large rate in certain states they should not be removed.

#### 3.1.2.3 Missing values (2 points) - Does this variable include missing values? Demonstrate how you determine that.

There are no missing values in the State variable. The method is.na with the column name can be used and then the vector returned can be turned into a data frame that represents the number of NA values (TRUE) and non NA values (FALSE). It can also be confirmed by calling summary() on the State, which also shows that there are no NA values in the State variable. There should also be information for all 50 states plus PR and DC, which is confirmed using unique() to show there are 52 unqique State values.

```r
missing <- is.na(breaches$State)

num_missing <- as.data.frame(table(missing))

num_missing
```

```
##   missing Freq
## 1   FALSE 1055
```

```r
summary(breaches$State)
```

```
##   TX  MO  AK  DC  CA  PA  TN  NY  NC  MI  MA  IL  UT  NV  AZ  RI  PR  FL  NM  CO
##   83  25   5   9 113  40  32  58  32  26  32  49   9   5  21   7  31  66  10  18
##   WY  WI  WA  CT  AL  NE  SC  KY  MN  VA  OH  KS  GA  MD  IN  ID  OR  NJ  DE  IA
##    4  14  25  17  12   9  13  26  21  18  33   7  30  18  40   2  15  20   3   8
##   OK  AR  MS  LA  NH  MT  WV  ND  HI  SD  ME  VT
##    6  11   5   7   4   4   5   3   1   1   1   1
```

```r
breaches$State %>%
  unique()
```

```
##  [1] TX MO AK DC CA PA TN NY NC MI MA IL UT NV AZ RI PR FL NM CO WY WI WA CT AL
## [26] NE SC KY MN VA OH KS GA MD IN ID OR NJ DE IA OK AR MS LA NH MT WV ND HI SD
## [51] ME VT
## 52 Levels: TX MO AK DC CA PA TN NY NC MI MA IL UT NV AZ RI PR FL NM CO ... VT
```

**- Demonstrate and discuss how you handle the missing values. E.g., removing, replacing with a constant value, or a value based on the distribution, etc.**

There are no missing values so they do no need to be handled.

**- Show how your data looks in each case after handling missing values.Describe and discuss the distribution.**

Since there is no missing values the distribution does not changed, see earlier bar graph for distribution.

**3.1.2.4 Does converting the type of this variable help exploring the distribution of its values or identifying outliers or missing values? (3)** Yes converting State to a logical may be helpful in exploring the distribution of its values or identifying outliers or missing values since logical are simpler to evaluate when larger continuous data is converted into two groups.

**- What type can the variable be converted to?**

State is of type factor, but it can converted to a logical. By making the value of State TRUE when the State is in the northeast and FALSE when the value of the State is not in the northeast, we can see if the northeast has a large number of breaches. Converting State to a logical is a simpler way to interpret State values. The converted State type is saved as a new variable northeast.

```r
northeast_list <- c("CT", "MA", "NH", "NJ", "NY", "PA", "RI", "VT", "DE", "MD", "ME")
#function to determine if the states are in the northeast
northeast_check <- function(x) {
  if(is.na(x)){
    return(NA)
```

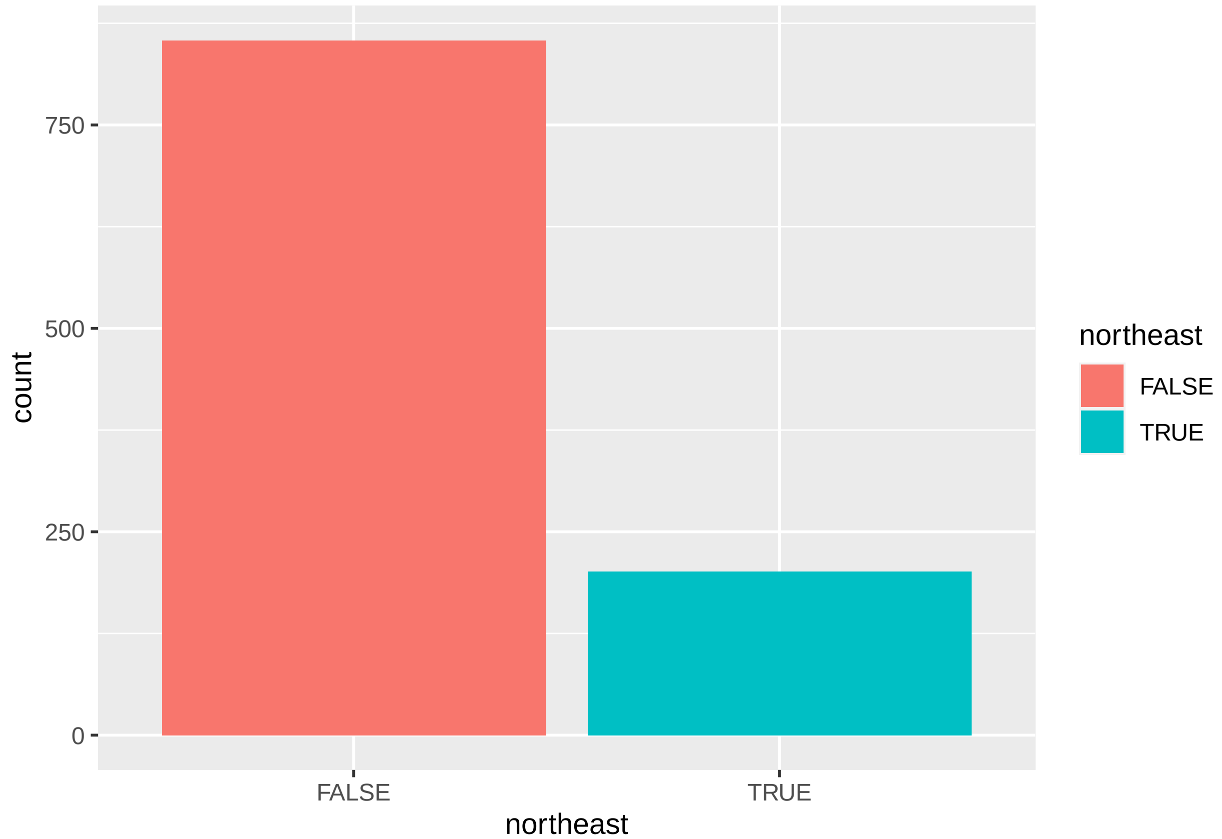
```r
  }
  else if(x %in% northeast_list){
    return(TRUE)
  }
  else{
    return(FALSE)
  }
}

breaches$northeast <- sapply(breaches$State, northeast_check)
head(breaches)
```

```
## # A tibble: 6 x 15
##      X1 Number Name_of_Covered_Entity  State Business_Associat~ Individuals_Aff~
##   <dbl>  <dbl> <chr>                   <fct> <chr>                         <int>
## 1     1      0 Brooke Army Medical Ce~ TX    <NA>                           1000
## 2     2      1 Mid America Kidney Sto~ MO    <NA>                           1000
## 3     3      2 Alaska Department of H~ AK    <NA>                            501
## 4     4      3 Health Services for Ch~ DC    <NA>                           3800
## 5     5      4 L. Douglas Carlson, M.~ CA    <NA>                           5257
## 6     6      5 David I. Cohen, MD      CA    <NA>                            857
## # ... with 9 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>
```

**- How will the distribution look? Please demonstrate with appropriate plots.**

From plotting the converted logical State as a bar graph, we can see that the majority of the breaches were not in the northeast. However the number of breaches is large for the northeast since there is only 9 states vs the other 43 States and territories. We can also see that there are no NA values, which confirms the analysis done earlier.

```r
breaches %>%
  ggplot(aes(x=northeast, fill = northeast)) +
  geom_bar()
```




**3.1.2.5 What new variables do you need to create? (3) - List the variables** northeast, westcoast, midwest, south.

All are logical variables that are true or false if the breach is in the region.

Region, which is a factor variable that sorts the US into northeast, westcoast, midwest, south and other.

```
westcoast_list <- c("WY", "CO", "UT", "NV", "ID", "CA", "OR", "WA", "AK", "AZ", "NM")
#function to determine if the states are on the West Coast
westcoast_check <- function(x) {
  if(is.na(x)){
    return(NA)
  }
  else if(x %in% westcoast_list){
    return(TRUE)
  }
  else{
    return(FALSE)
  }
}

breaches$westcoast <- sapply(breaches$State, westcoast_check)
head(breaches)
```

```
## # A tibble: 6 x 16
##       X1 Number Name_of_Covered_Entity  State Business_Associat~ Individuals_Aff~
```

```
##   <dbl>  <dbl> <chr>                    <fct> <chr>                          <int>
## 1     1      0 Brooke Army Medical Ce~ TX    <NA>                            1000
## 2     2      1 Mid America Kidney Sto~ MO    <NA>                            1000
## 3     3      2 Alaska Department of H~ AK    <NA>                             501
## 4     4      3 Health Services for Ch~ DC    <NA>                            3800
## 5     5      4 L. Douglas Carlson, M.~ CA    <NA>                            5257
## 6     6      5 David I. Cohen, MD      CA    <NA>                             857
## # ... with 10 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>, westcoast <lgl>
```

```
midwest_list <- c("ND", "SD", "NE", "KS", "MO", "IA", "MN", "WI", "MI", "IL", "IN", "OH", "MT")
#function to determine if the states are in the midwest
midwest_check <- function(x) {
  if(is.na(x)){
    return(NA)
  }
  else if(x %in% midwest_list){
    return(TRUE)
  }
  else{
    return(FALSE)
  }
}

breaches$midwest <- sapply(breaches$State, midwest_check)
head(breaches)
```

```
## # A tibble: 6 x 17
##      X1 Number Name_of_Covered_Entity  State Business_Associat~ Individuals_Aff~
##   <dbl>  <dbl> <chr>                    <fct> <chr>                          <int>
## 1     1      0 Brooke Army Medical Ce~ TX    <NA>                            1000
## 2     2      1 Mid America Kidney Sto~ MO    <NA>                            1000
## 3     3      2 Alaska Department of H~ AK    <NA>                             501
## 4     4      3 Health Services for Ch~ DC    <NA>                            3800
## 5     5      4 L. Douglas Carlson, M.~ CA    <NA>                            5257
## 6     6      5 David I. Cohen, MD      CA    <NA>                             857
## # ... with 11 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>, westcoast <lgl>, midwest <lgl>
```

```
southwest_list <- c("AZ", "NM", "OK", "TX")
other_list <- c("DC", "PR")
```

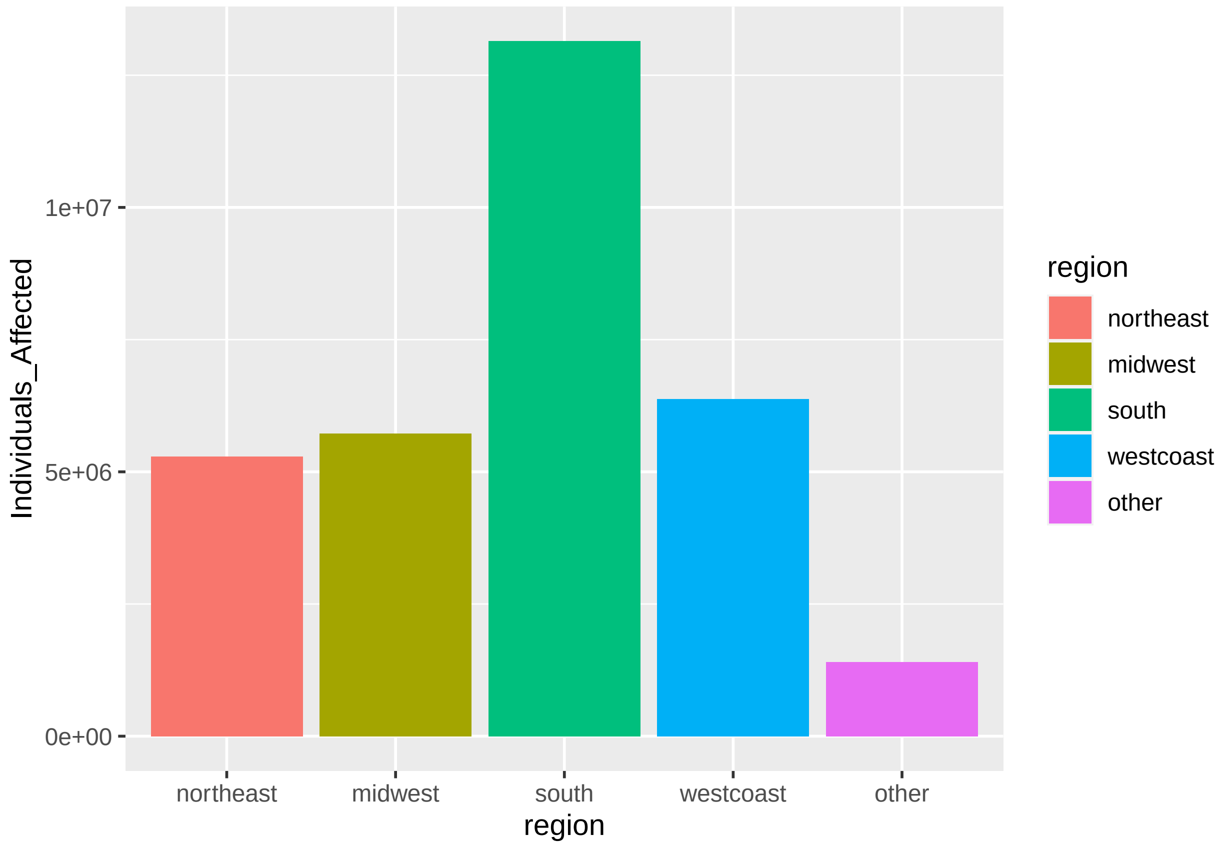
```
south_list <- c("MD", "DE", "VA", "WV", "KY", "TN", "NC", "SC", "FL", "GA", "AL", "MS", "LA", "AK", "OK
#function to determine if the states are in the south
south_check <- function(x) {
  if(is.na(x)){
    return(NA)
  }
  else if(x %in% south_list){
```

```r
    return(TRUE)
  }
  else{
    return(FALSE)
  }
}

breaches$south <- sapply(breaches$State, south_check)
head(breaches)
```

```
## # A tibble: 6 x 18
##      X1 Number Name_of_Covered_Entity  State Business_Associat~ Individuals_Aff~
##   <dbl>  <dbl> <chr>                   <fct> <chr>                         <int>
## 1     1      0 Brooke Army Medical Ce~ TX    <NA>                           1000
## 2     2      1 Mid America Kidney Sto~ MO    <NA>                           1000
## 3     3      2 Alaska Department of H~ AK    <NA>                            501
## 4     4      3 Health Services for Ch~ DC    <NA>                           3800
## 5     5      4 L. Douglas Carlson, M.~ CA    <NA>                           5257
## 6     6      5 David I. Cohen, MD      CA    <NA>                            857
## # ... with 12 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>, westcoast <lgl>, midwest <lgl>, south <lgl>
```

```r
region_check <- function(x) {
  if(is.na(x)){
    return(NA)
  }
  else if(x %in% westcoast_list){
    return("westcoast")
  }
  else if(x %in% northeast_list){
    return("northeast")
  }
  else if(x %in% midwest_list){
    return("midwest")
  }
  else if(x %in% south_list){
    return("south")
  }
  else{
    return("other")
  }
}


breaches$region <- sapply(breaches$State, region_check)

region_levels = c("northeast", "midwest", "south", "westcoast", "other")

breaches$region <- factor(breaches$region, levels= region_levels)
```

```r
breaches %>%
  ggplot(aes(x = region, y = Individuals_Affected, fill = region)) +
  geom_col()
```

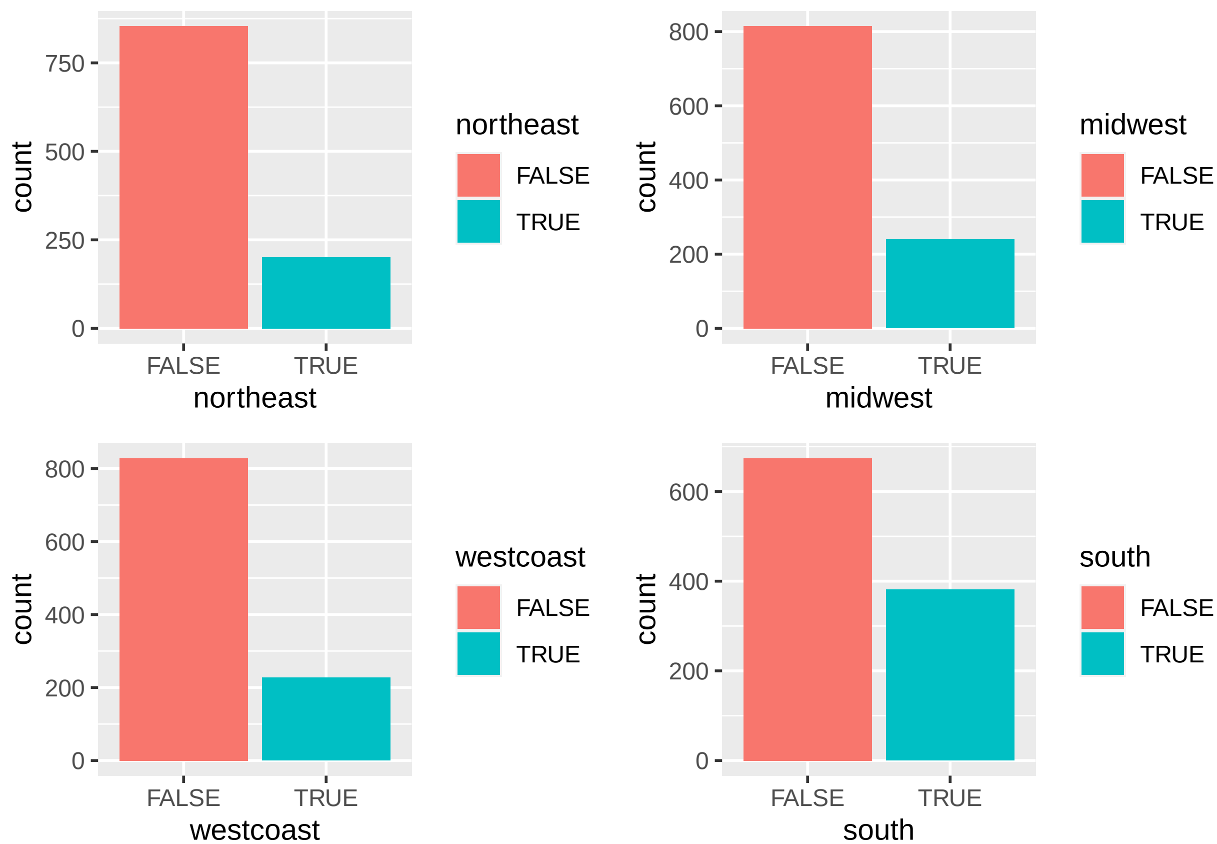



**- Describe and discuss why they are needed and how you plan to use them.** northeast, westcoast, midwest, and south, are all a logical variable. They are needed in exploring the distribution of breaches per state in different regions of the US. Logical variables are used since logical are simpler to evaluate when larger factor data is converted into two groups. I plan to use the variables to compare individuals affected by their location.

The region variable sorts the US into regions based on the state the breach occured in. I am planning on using the region variable to compare the categorical states to the individuals affected in boxplots and bar graphs.

```r
northeast_bar <-
breaches %>%
  ggplot(aes(x=northeast, fill = northeast)) +
  geom_bar()

midwest_bar <-
breaches %>%
  ggplot(aes(x=midwest, fill = midwest)) +
  geom_bar()

westcoast_bar <-
breaches %>%
```

```
  ggplot(aes(x=westcoast, fill = westcoast)) +
  geom_bar()

south_bar <-
breaches %>%
  ggplot(aes(x=south, fill = south)) +
  geom_bar()

ggarrange(northeast_bar, midwest_bar, westcoast_bar, south_bar, nrow = 2, ncol = 2)
```



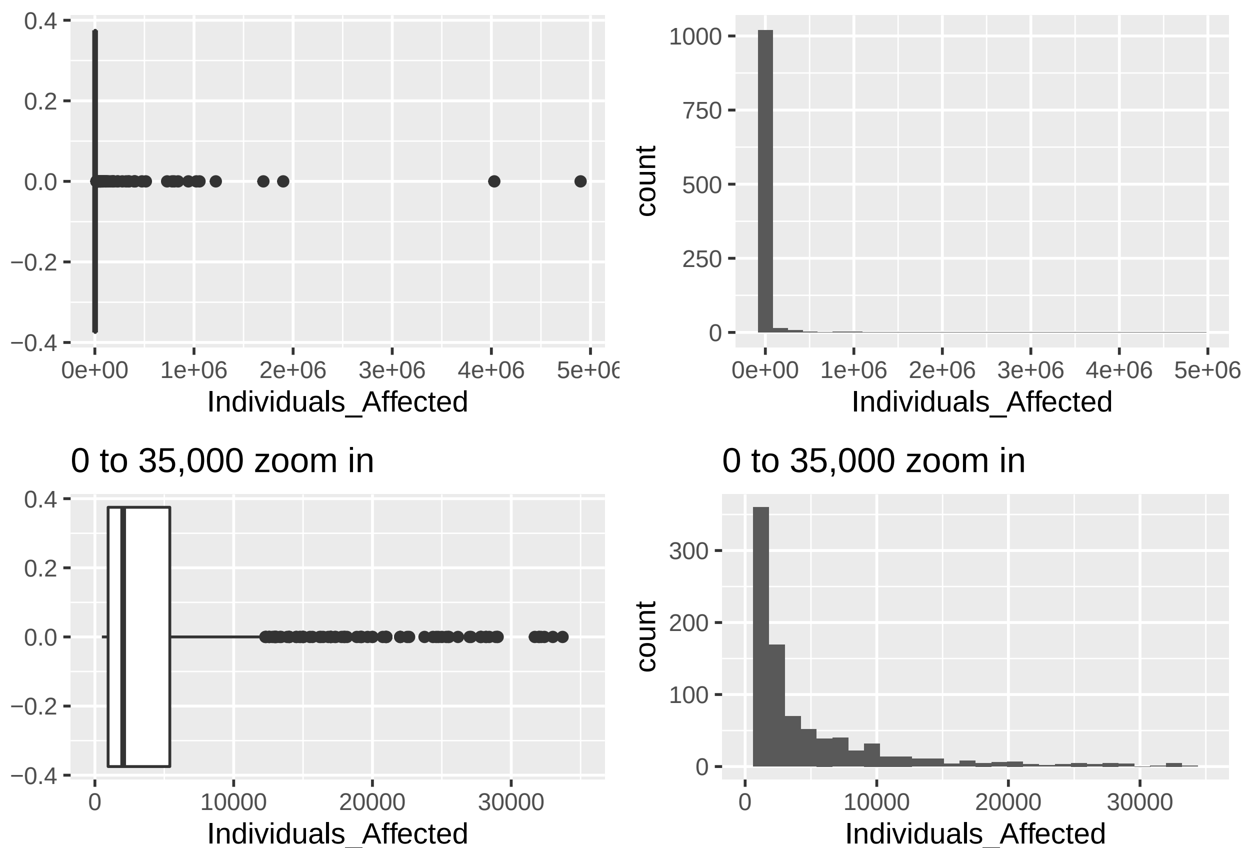

### 3.1.3 Individuals_affected Variable

```
indiv_box <- breaches %>%
  ggplot(aes(x=Individuals_Affected)) +
  geom_boxplot()

indiv_hist <- breaches %>%
  ggplot(aes(x=Individuals_Affected)) +
  geom_histogram()

indiv_box_zoom <- breaches %>%
  ggplot(aes(x=Individuals_Affected)) +
  geom_boxplot()+
```

```r
  xlim(0, 35000) +
  labs(title = "0 to 35,000 zoom in")

indiv_hist_zoom <- breaches %>%
  ggplot(aes(x=Individuals_Affected)) +
  geom_histogram() +
  xlim(0, 35000) +
  labs(title = "0 to 35,000 zoom in")

ggarrange(indiv_box, indiv_hist, indiv_box_zoom, indiv_hist_zoom,  nrow = 2, ncol = 2)
```

#### 3.1.2.1 Visualising distributions (Barcharts, Histograms) (5 points)

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 69 rows containing non-finite values (stat_boxplot).

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 69 rows containing non-finite values (stat_bin).

## Warning: Removed 2 rows containing missing values (geom_bar).
```



```r
summary(breaches)
```

```
##        X1             Number       Name_of_Covered_Entity     State    
##  Min.   :   1.0   Min.   :   0.0   Length:1055            CA     :113  
##  1st Qu.: 264.5   1st Qu.: 263.5   Class :character       TX     : 83  
##  Median : 528.0   Median : 527.0   Mode  :character       FL     : 66  
##  Mean   : 528.0   Mean   : 527.0                          NY     : 58  
##  3rd Qu.: 791.5   3rd Qu.: 790.5                          IL     : 49  
##  Max.   :1055.0   Max.   :1054.0                          PA     : 40  
##                                                           (Other):646  
##  Business_Associate_Involved Individuals_Affected Date_of_Breach    
##  Length:1055                 Min.   :    500      Length:1055       
##  Class :character            1st Qu.:   1000      Class :character  
##  Mode  :character            Median :   2300      Mode  :character  
##                              Mean   :  30262                        
##                              3rd Qu.:   6941                        
##                              Max.   :4900000                        
##                                                                     
##  Type_of_Breach     Location_of_Breached_Information Date_Posted_or_Updated
##  Length:1055        Length:1055                      Min.   :2014-01-23    
##  Class :character   Class :character                 1st Qu.:2014-01-23    
##  Mode  :character   Mode  :character                 Median :2014-01-23    
##                                                      Mean   :2014-02-23    
##                                                      3rd Qu.:2014-03-24    
##                                                      Max.   :2014-06-30    
##                                                                            
##    Summary           breach_start          breach_end              year     
##  Length:1055        Min.   :1997-01-01   Min.   :2007-06-14   Min.   :1997  
##  Class :character   1st Qu.:2010-11-08   1st Qu.:2012-04-22   1st Qu.:2010  
##  Mode  :character   Median :2012-01-11   Median :2012-10-29   Median :2012  
##                     Mean   :2011-12-09   Mean   :2012-10-28   Mean   :2011  
##                     3rd Qu.:2013-03-07   3rd Qu.:2013-05-29   3rd Qu.:2013  
##                     Max.   :2014-06-02   Max.   :2013-11-30   Max.   :2014  
##                                          NA's   :910                        
##  northeast       westcoast        midwest          south        
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical  
##  FALSE:854       FALSE:828       FALSE:815       FALSE:674      
##  TRUE :201       TRUE :227       TRUE :240       TRUE :381      
##                                                                 
##                                                                 
##                                                                 
##                                                                 
##        region   
##  northeast:201  
##  midwest  :240  
##  south    :355  
##  westcoast:227  
##  other    : 32  
##                 
## 
```

```r
IQR(breaches$Individuals_Affected, na.rm = TRUE)
```

```
## [1] 5941
```

**- Which values are the most common? Why?**

The values in the IQR are the most common which ranges from 500 to 6941 people. This can be seen in the histogram since the peak is centered around 2300 people, which is the median. The majority of the values fall in this range and therefore they are statistically the most common. This can be interpreted that in most data breaches the number of individuals affected is usually between 500 to around 7000 people.

**- Which values are rare? Why? Does that match your expectations?**

Values above 1 million are more rare. This does match my expectations since large breaches are less common, and therefore breaches with indiviuals affected being above 1 million are more rare.

**- Can you see any unusual patterns? What might explain them?**

There is no cycle pattern present in the individuals affected data. The only slightly unusual pattern is that there is a strong right skew. There are some very large values for individuals affected that drag the mean up, and therefore the data is very right skewed. Overall the median is a better reference to the middle of the data than the mean. This right skew is caused by a few data breaches that had very high numbers of indivuals affected.

**- Are there clusters in the data? If so,** There is a large grouping of data at about 5000 individuals affected and below. There is not another large grouping however that could be defined as a cluster.

**- How are the observations within each cluster similar to or different from each other?**

The observations in the low individuals affected cluster all come from different states, and there isn't an obvious connection between the points.

**- How can you explain or describe the clusters?**

The cluster can possibly be explained by the fact that most breaches are on the smaller side, and that breaches that affect a lot of people are harder to pull off and therefore more rare.

#### 3.1.2.2 Unusual values - Describe and demonstrate how you determine if there are unusual values in the data. E.g. too large, too small, negative, etc.

There are no negative values for individuals affected, and there are two very large values, above 3 million. I filtered for both situations to confirm this result of unusual values.

```r
neg_indiv <- breaches %>%
  filter(Individuals_Affected < 0)

neg_indiv
```

```
## # A tibble: 0 x 19
## # ... with 19 variables: X1 <dbl>, Number <dbl>, Name_of_Covered_Entity <chr>,
## #   State <fct>, Business_Associate_Involved <chr>, Individuals_Affected <int>,
## #   Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>, westcoast <lgl>, midwest <lgl>, south <lgl>, region <fct>
```

```r
large_indiv <- breaches %>%
  filter(Individuals_Affected > 3000000)

large_indiv
```

```
## # A tibble: 2 x 19
##      X1 Number Name_of_Covered_Entity State Business_Associate~ Individuals_Aff~
##   <dbl>  <dbl> <chr>                  <fct> <chr>                          <int>
## 1   410    409 TRICARE Management Ac~ VA    Science Applicatio~          4900000
## 2   800    799 Advocate Health and H~ IL    <NA>                         4029530
## # ... with 13 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>, westcoast <lgl>, midwest <lgl>, south <lgl>, region <fct>
```

**- Describe and demonstrate how you determine if they are outliers.**

An outlier is 1.5 times the interquartile range away from either the lower or upper quartile. In order to determine if any of the indivuals affected values are outliers the interquartile range, first quartile, and third quartile need to be calculated. The indivduals affected data then has to be filtered for values that are less than the first quartile minus the IQR times 1.5 and values that are greater than the third quartile plus the IQR times 1.5. The 129 outliers can be seen in the outlier list.

```r
stdev <-  sd(breaches$Individuals_Affected, na.rm = TRUE)
stdev
```

```
## [1] 227859.8
```

```r
innerQ <-  IQR(breaches$Individuals_Affected, na.rm = TRUE)
innerQ
```

```
## [1] 5941
```

```r
firstQ <- quantile(breaches$Individuals_Affected, 0.25, na.rm = TRUE)
firstQ <- firstQ[[1]]

thirdQ <- quantile(breaches$Individuals_Affected, 0.75, na.rm = TRUE)
thirdQ <- thirdQ[[1]]

outlier_list <- breaches %>%
  filter(Individuals_Affected < (firstQ - innerQ * 1.5) |
         Individuals_Affected > (thirdQ + innerQ * 1.5))

outlier_list
```

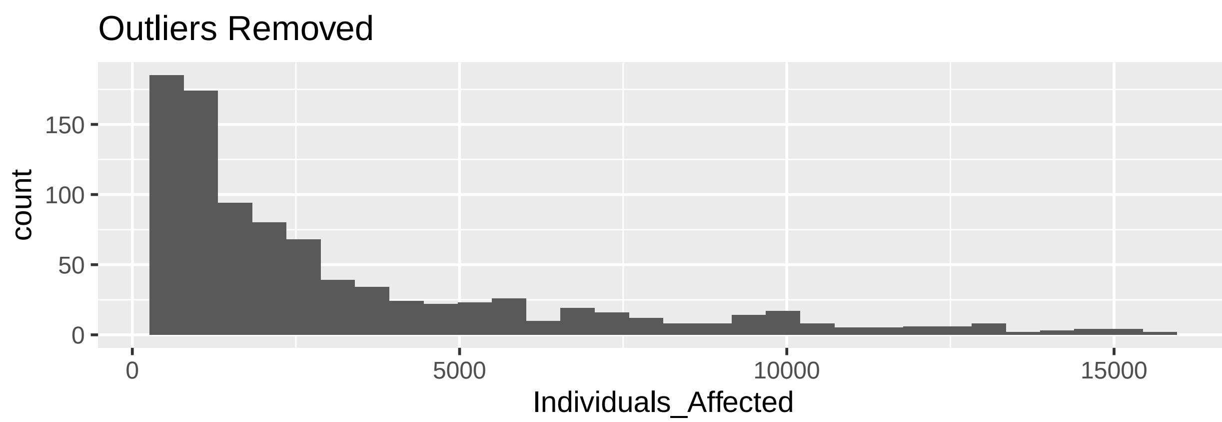
```
## # A tibble: 129 x 19
##       X1 Number Name_of_Covered_Enti~ State Business_Associate~ Individuals_Aff~
##    <dbl>  <dbl> <chr>                 <fct> <chr>                          <int>
##  1    13     12 "Universal American"  NY    Democracy Data & C~            83000
##  2    50     49 "Ernest T. Bice, Jr.~ TX    <NA>                           21000
##  3    59     58 "Providence Hospital" MI    <NA>                           83945
```

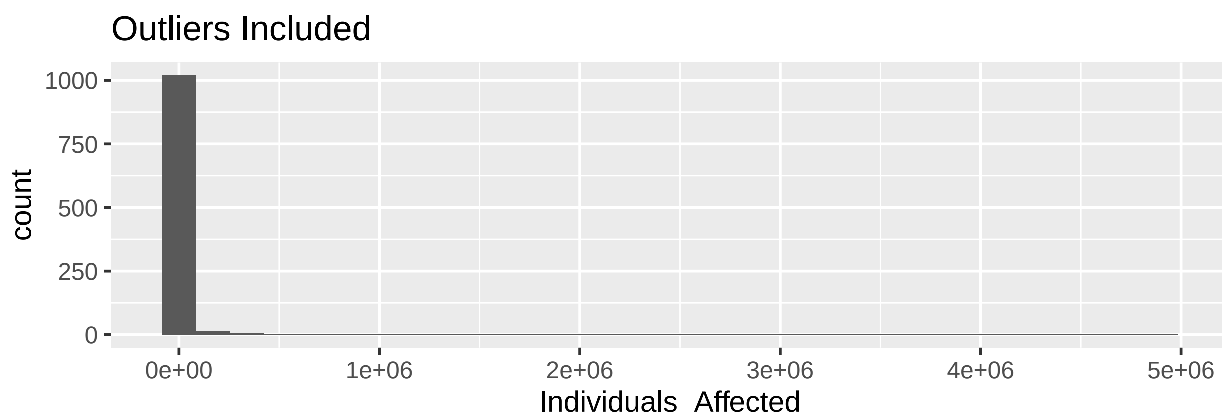
```
## 4     64      63 "Affinity Health Pla~ NY    <NA>                          344579
## 5     66      65 "Praxair Healthcare ~ CT    <NA>                           54165
## 6     70      69 "St. Joseph Heritage~ CA    <NA>                           22012
## 7     76      75 "Emergency Healthcar~ IL    Millennium Medical~          180111
## 8     81      80 "Silicon Valley Eyec~ CA    <NA>                           40000
## 9     91      90 "Cincinnati Children~ OH    <NA>                           60998
## 10    93      92 "AvMed, Inc."         FL    <NA>                         1220000
## # ... with 119 more rows, and 13 more variables: Date_of_Breach <chr>,
## #   Type_of_Breach <chr>, Location_of_Breached_Information <chr>,
## #   Date_Posted_or_Updated <date>, Summary <chr>, breach_start <date>,
## #   breach_end <date>, year <dbl>, northeast <lgl>, westcoast <lgl>,
## #   midwest <lgl>, south <lgl>, region <fct>
```

**- Show how do your distributions look like with and without the unusual values.**

```
outliers_removed <- breaches %>%
  filter(!Individuals_Affected %in% outlier_list$Individuals_Affected) %>%
  ggplot(aes(x=Individuals_Affected))+
  geom_histogram() +
  labs(title = "Outliers Removed")

outliers_included <- breaches %>%
  ggplot(aes(x=Individuals_Affected)) +
  geom_histogram()+
  labs(title = "Outliers Included")

ggarrange(outliers_removed, outliers_included, nrow = 2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




**- Discuss whether or not you need to remove unusual values and why.**

The unusual values should not be removed because since the high individuals affected values will most likely give the most insight into cyber security issues.

#### 3.1.2.3 Missing values (2 points) - Does this variable include missing values? Demonstrate how you determine that.

No there are no missing values. The method is.na with the column name can be used and then the vector returned can be turned into a data frame that represents the number of NA values (TRUE) and non NA values (FALSE). It can also be confirmed by calling summary() on the Individuals Affected variable, which also shows that there are no NA values in the Individuals affected variable.

```
missing <- is.na(breaches$Individuals_Affected)

num_missing <- as.data.frame(table(missing))

num_missing
```

```
##   missing Freq
## 1   FALSE 1055
```

```
summary(breaches$Individuals_Affected)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     500    1000    2300   30262    6941 4900000
```

**- Demonstrate and discuss how you handle the missing values. E.g., removing, replacing with a constant value, or a value based on the distribution, etc.**

There are no missing values

**- Show how your data looks in each case after handling missing values.Describe and discuss the distribution.**

There are no missing values. Refer to histogram and boxplots above for distribution.

**3.1.2.4 Does converting the type of this variable help exploring the distribution of its values or identifying outliers or missing values? (3)** Yes converting Individuals affected to a logical may be helpful in exploring the distribution of its values or identifying outliers or missing values since logical are simpler to evaluate when larger continuous data is converted into two groups.

**- What type can the variable be converted to?**

Individuals affected is of type integer, but it can converted to a logical. By making the value of Individuals affected TRUE when the value is greater than 20,000 and FALSE when the value is lower than than 20,000, we can see if the Inidviduals Affected level is considered high or not. Converting Individuals Affected to a logical is a simpler way to interpret Individuals values. The converted Individuals Affected type is saved as a new variable Large_Affected.

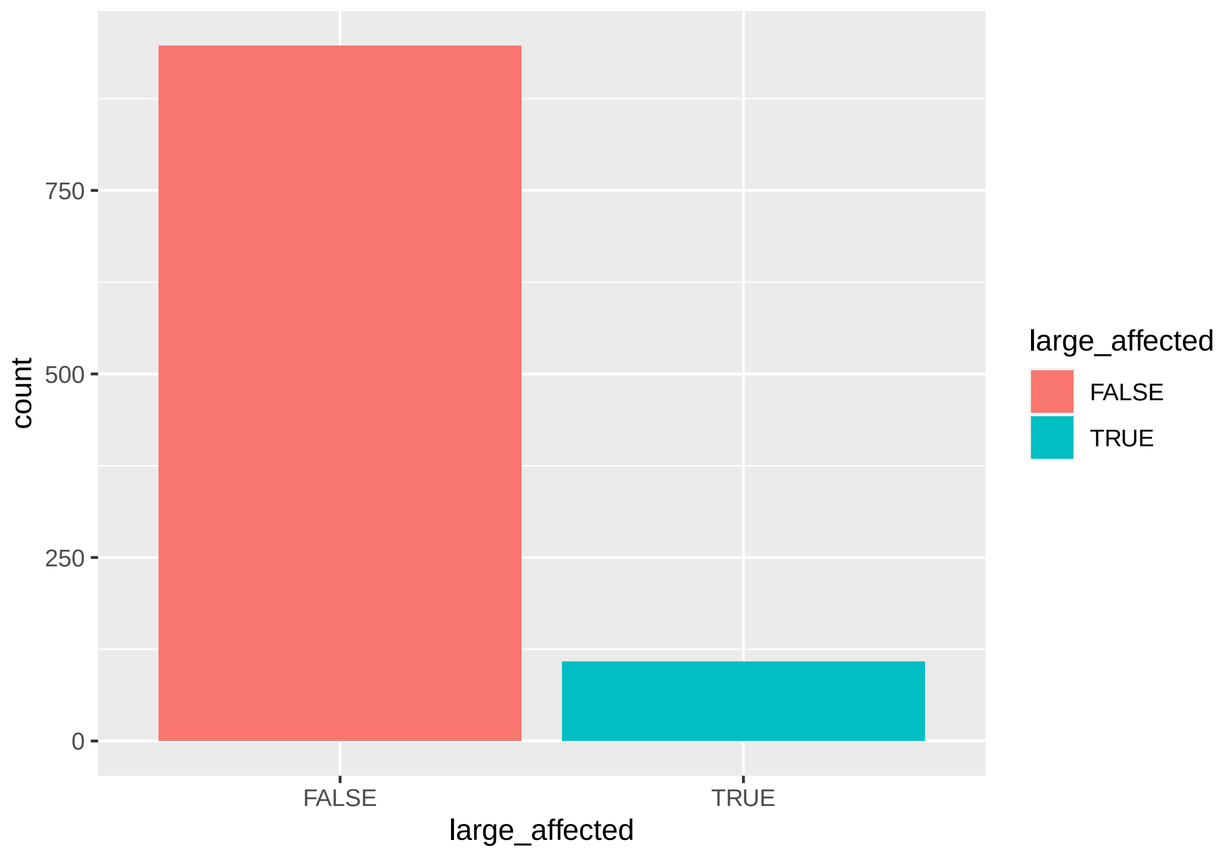
```
high_check <- function(x) {
  if(is.na(x)){
    return(NA)
  }
  else if(x >= 20000){
    return(TRUE)
  }
  else{
    return(FALSE)
  }
}

breaches$large_affected <- sapply(breaches$Individuals_Affected, high_check)
head(breaches)
```

```
## # A tibble: 6 x 20
##      X1 Number Name_of_Covered_Entity  State Business_Associat~ Individuals_Aff~
##   <dbl>  <dbl> <chr>                   <fct> <chr>                         <int>
## 1     1      0 Brooke Army Medical Ce~ TX    <NA>                           1000
## 2     2      1 Mid America Kidney Sto~ MO    <NA>                           1000
## 3     3      2 Alaska Department of H~ AK    <NA>                            501
## 4     4      3 Health Services for Ch~ DC    <NA>                           3800
## 5     5      4 L. Douglas Carlson, M.~ CA    <NA>                           5257
## 6     6      5 David I. Cohen, MD      CA    <NA>                            857
## # ... with 14 more variables: Date_of_Breach <chr>, Type_of_Breach <chr>,
## #   Location_of_Breached_Information <chr>, Date_Posted_or_Updated <date>,
## #   Summary <chr>, breach_start <date>, breach_end <date>, year <dbl>,
## #   northeast <lgl>, westcoast <lgl>, midwest <lgl>, south <lgl>, region <fct>,
## #   large_affected <lgl>
```

**- How will the distribution look? Please demonstrate with appropriate plots.**

From plotting the converted logical Individuals Affected variable as a bar graph, we can see that the majority of the breaches were above 20,000 people affected. We can also see that there are no NA values, which confirms the analysis done earlier.

```
breaches %>%
  ggplot(aes(x=large_affected, fill = large_affected)) +
  geom_bar()
```




**3.1.2.5 What new variables do you need to create? (3) - List the variables** The new variable large_affected was created above and also explained above.

**- Describe and discuss why they are needed and how you plan to use them.** Large_affected is needed to look at the outliers of the individuals affected and see if there is a trend with the large values and the states. I plan to use the logical and see if there is a correlation with the state the breach occured in.

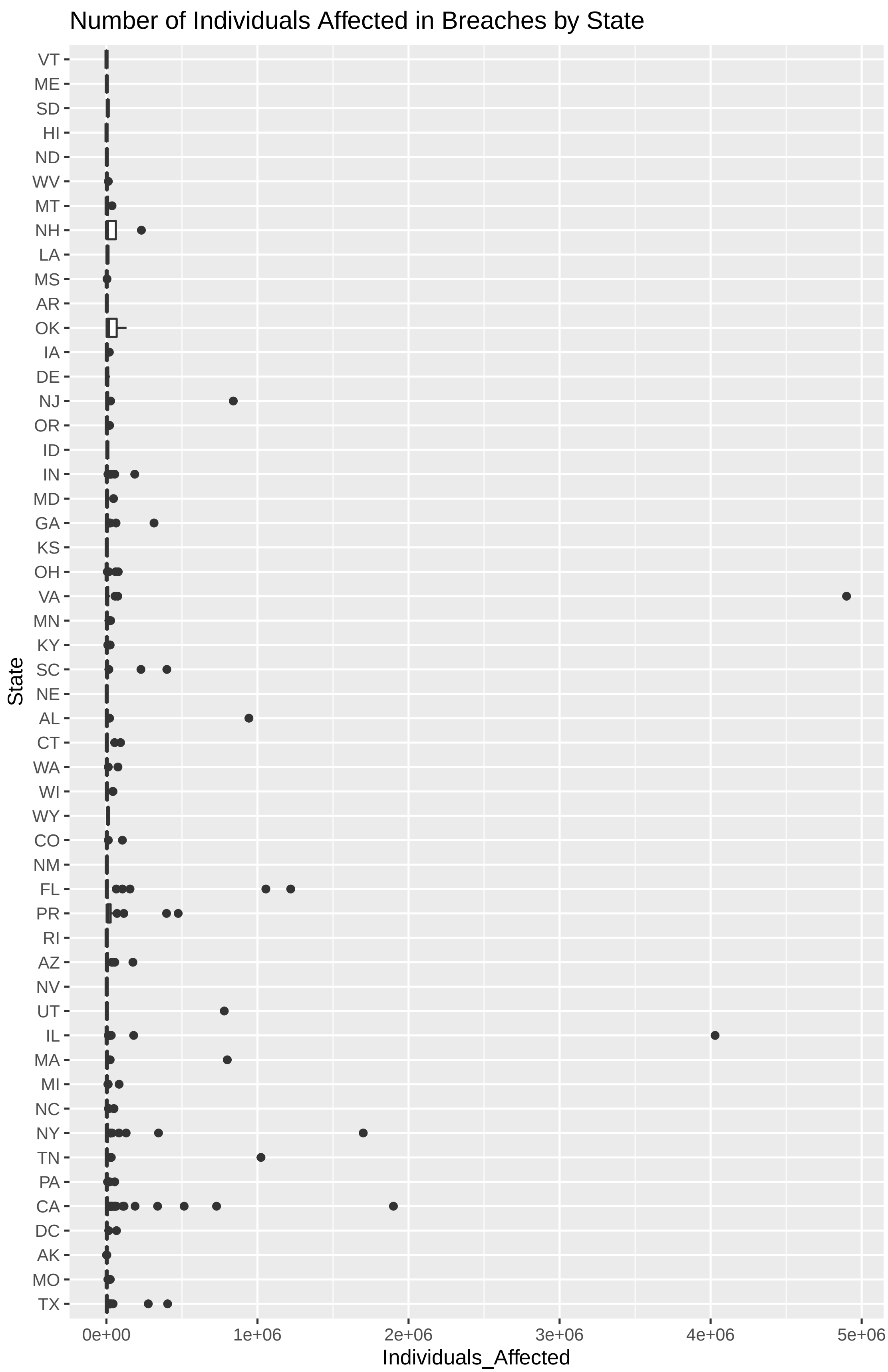
## 3.2. What type of covariation occurs between the variables? (30 points)

If you don't have variables of a certain type in the original dataset or among the created variables (features), you can further create them from the existing variables. See RDS chap. 5, 7.5 and 7.6.

### 3.2.1 Between a categorical and continuous variable (10 points)

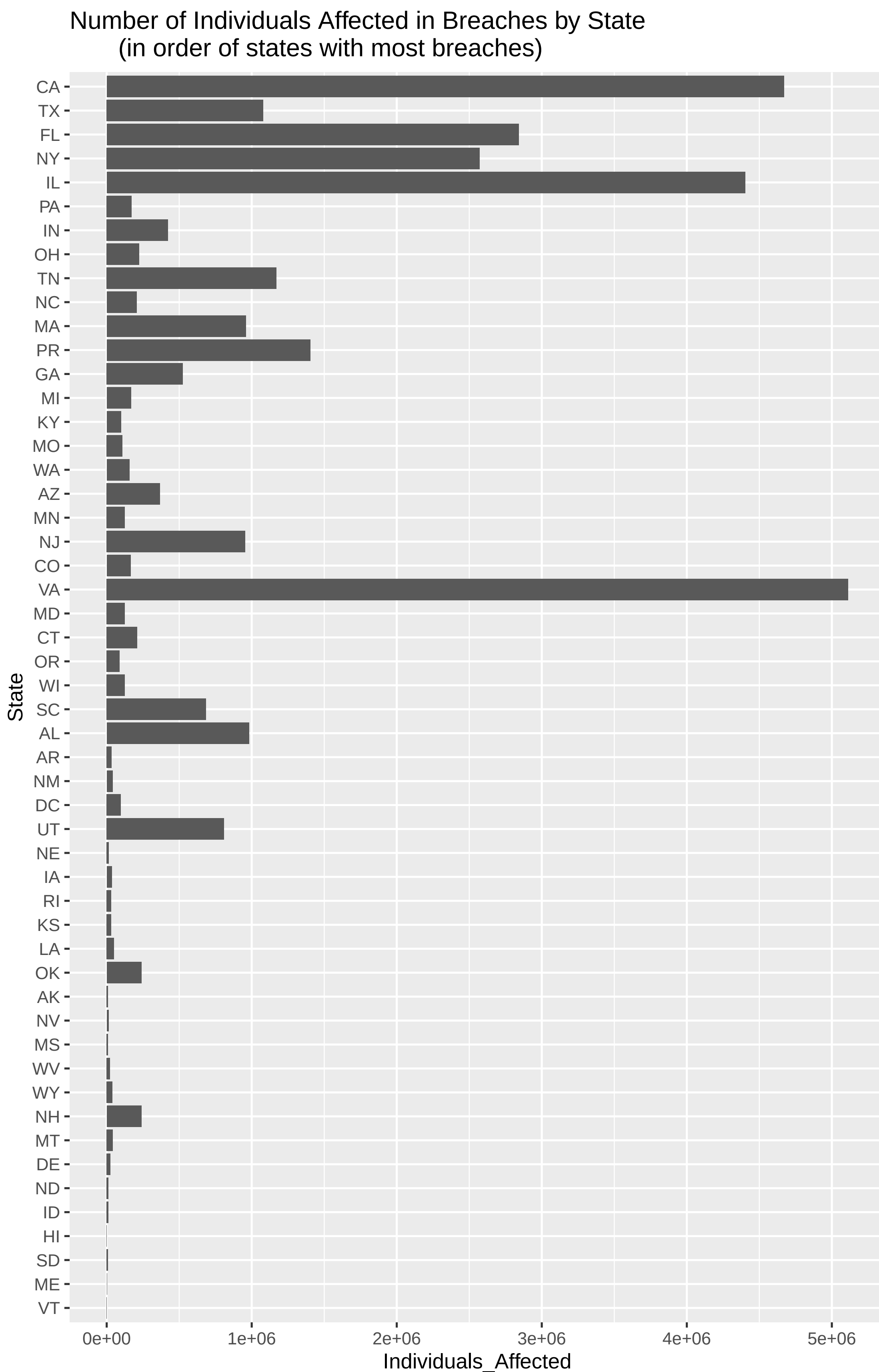
**- Describe what type of visualization you can use and why.** A boxplot of State as a categorical variable and Individuals Affected as a continuous variable can be used. Using the box plot makes it clear the spread of the data depending on each state of the US. Boxplots are also compact and easier to compare the different states and the individuals affected distributions. A bar graph can also be used to look at the total number of individuals affected rather than the distribution by state.

```
breaches %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Number of Individuals Affected in Breaches by State")
```

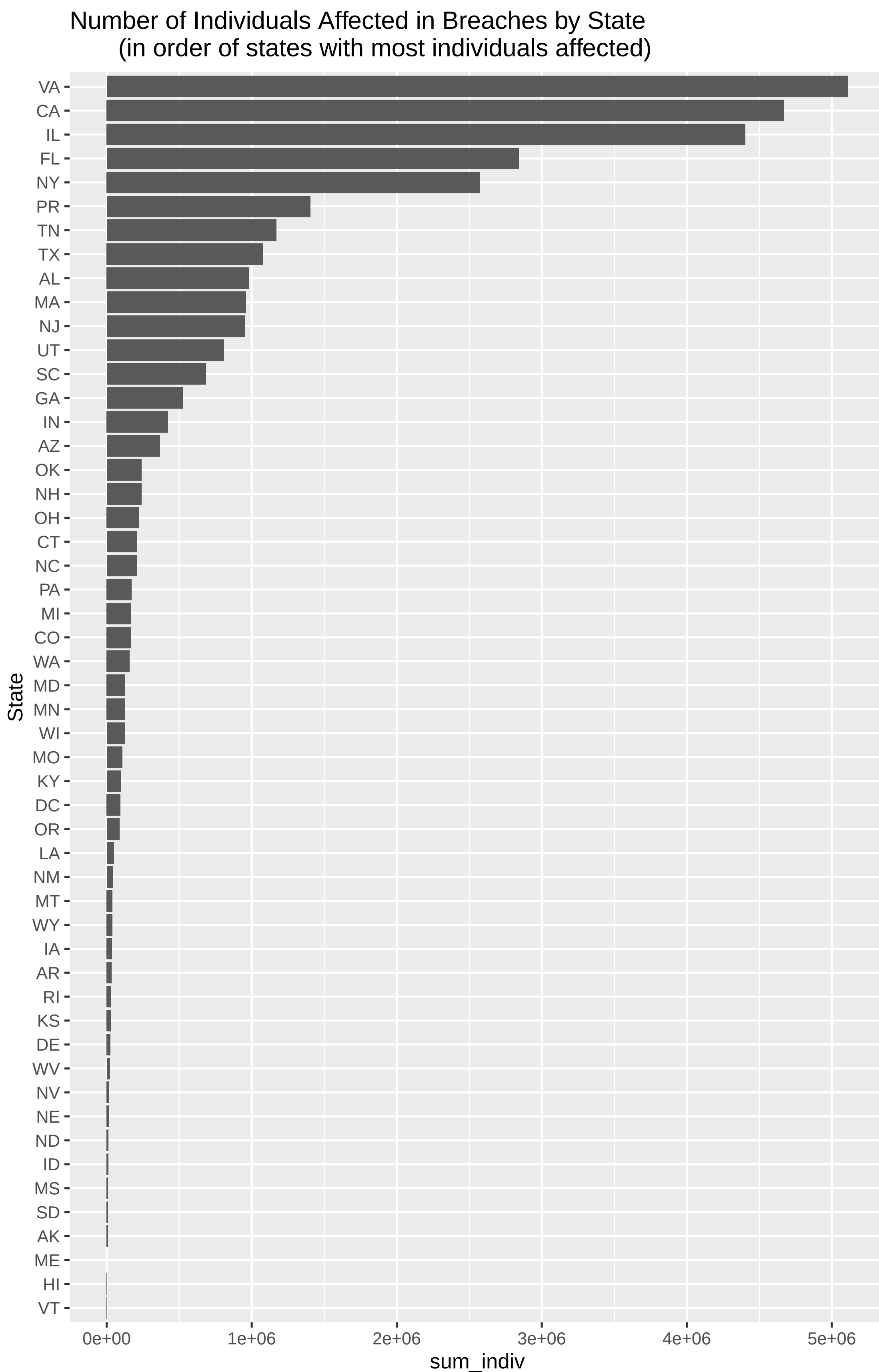


Number of Individuals Affected in Breaches by State
VT
ME
SD
HI
ND
WV
MT
NH
LA
MS
AR
OK
IA
DE
NJ
OR
ID
IN
MD
GA
KS
OH
VA
MN
KY
SC
NE
AL
CT
WA
WI
WY
CO
NM
FL
PR
RI
AZ
NV
UT
IL
MA
MI
NC
NY
TN
PA
CA
DC
AK
MO
TX
State
0e+00
1e+06
2e+06
3e+06
4e+06
5e+06
Individuals_Affected

```
breaches %>%
  mutate(State = as.factor(State) %>% fct_infreq() %>% fct_rev()) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_col() +
  coord_flip() +
  labs(title = "Number of Individuals Affected in Breaches by State
       (in order of states with most breaches)")
```


Number of Individuals Affected in Breaches by State
(in order of states with most breaches)
State
CA
TX
FL
NY
IL
PA
IN
OH
TN
NC
MA
PR
GA
MI
KY
MO
WA
AZ
MN
NJ
CO
VA
MD
CT
OR
WI
SC
AL
AR
NM
DC
UT
NE
IA
RI
KS
LA
OK
AK
NV
MS
WV
WY
NH
MT
DE
ND
ID
HI
SD
ME
VT
0e+00
1e+06
2e+06
3e+06
4e+06
5e+06
Individuals_Affected

```r
total_affected_state <- breaches %>%
  group_by(State) %>%
  summarize(sum_indiv = sum(Individuals_Affected))

total_affected_state$State = with(total_affected_state, reorder(State, sum_indiv))


total_affected_state$region <- sapply(total_affected_state$State, region_check)


total_affected_state$region <- factor(total_affected_state$region, levels= region_levels)

total_affected_state %>%
  ggplot(aes(State, sum_indiv)) +
  geom_col() +
  coord_flip()+
  labs(title = "Number of Individuals Affected in Breaches by State
       (in order of states with most individuals affected)")
```


Number of Individuals Affected in Breaches by State
(in order of states with most individuals affected)
State
VA
CA
IL
FL
NY
PR
TN
TX
AL
MA
NJ
UT
SC
GA
IN
AZ
OK
NH
OH
CT
NC
PA
MI
CO
WA
MD
MN
WI
MO
KY
DC
OR
LA
NM
MT
WY
IA
AR
RI
KS
DE
WV
NV
NE
ND
ID
MS
SD
AK
ME
HI
VT
0e+00
1e+06
2e+06
3e+06
4e+06
5e+06
sum_indiv

**- Describe the patterns and relationships you observe. Could the identified patterns be due to coincidence (i.e. random chance)?** The boxplot does not appear to have a clear pattern, but there are a lot of outliers, indicating that there were many breaches that affected a large number of individuals. Looking at the bar graphs, the states with big well known cities have breaches that affect the most number of people. This could be due to chance since just one large breach could influence a state's total individuals affected, however the trend seems to be consistent for most of the states with big cities. The outlier in this trend is Puerto Rico, which doesn't have a large city.

**- Describe the relationship implied by the pattern? (e.g., positive or negative correlation)**

There is a positive correlation between the states with big cities and the number of individuals affected. Overall however since States is not a measurable factor there is not a correlation. Sorting by feature may lead to a stronger correlation by region.

**- Calculate the strength of the relationship implied by the pattern (e.g., correlation)**

One approach at looking at correlation between categorical and continuous variables is from "https://medium.com/@outside2SDs/an-overview-of-correlation-measures-between-categorical-and-continuous-variables-4c7f85610365".

The approach is to group the continuous variable using the categorical variable, measure the variance in each group and comparing it to the overall variance of the continuous variable. If the variance after grouping falls down significantly, it means that the categorical variable can explain most of the variance of the continuous variable and so the two variables likely have a strong association. If the variables have no correlation, then the variance in the groups is expected to be similar to the original variance.

These calculations were done and can be seen in the data frames state_and_indiv and indiv_summary. Overall I would say there is minimal correlation because the variance for just the indivuals affected variable is 51920099070, and when breaches is grouped by region, northeast, westcoast, and other variances decrease, but midwest and south increase. Therefore grouping by region has a minimal correlation on the Individuals Affected data.

```r
state_and_indiv <- breaches %>%
  group_by(region) %>%
  summarize(sd = sd(Individuals_Affected))

state_and_indiv %>%
  mutate(var = sd^2)
```

```
## # A tibble: 5 x 3
##   region         sd          var
## * <fct>       <dbl>        <dbl>
## 1 northeast 146987. 21605211841.
## 2 midwest   260387. 67801639077.
## 3 south     284834. 81130225641.
## 4 westcoast 150096. 22528726555.
## 5 other     106298. 11299218049.
```

```r
indiv_summary <- breaches %>%
  summarize(sd = sd(Individuals_Affected))

indiv_summary %>%
  mutate(var = sd^2)
```

```
## # A tibble: 1 x 2
##        sd          var
```

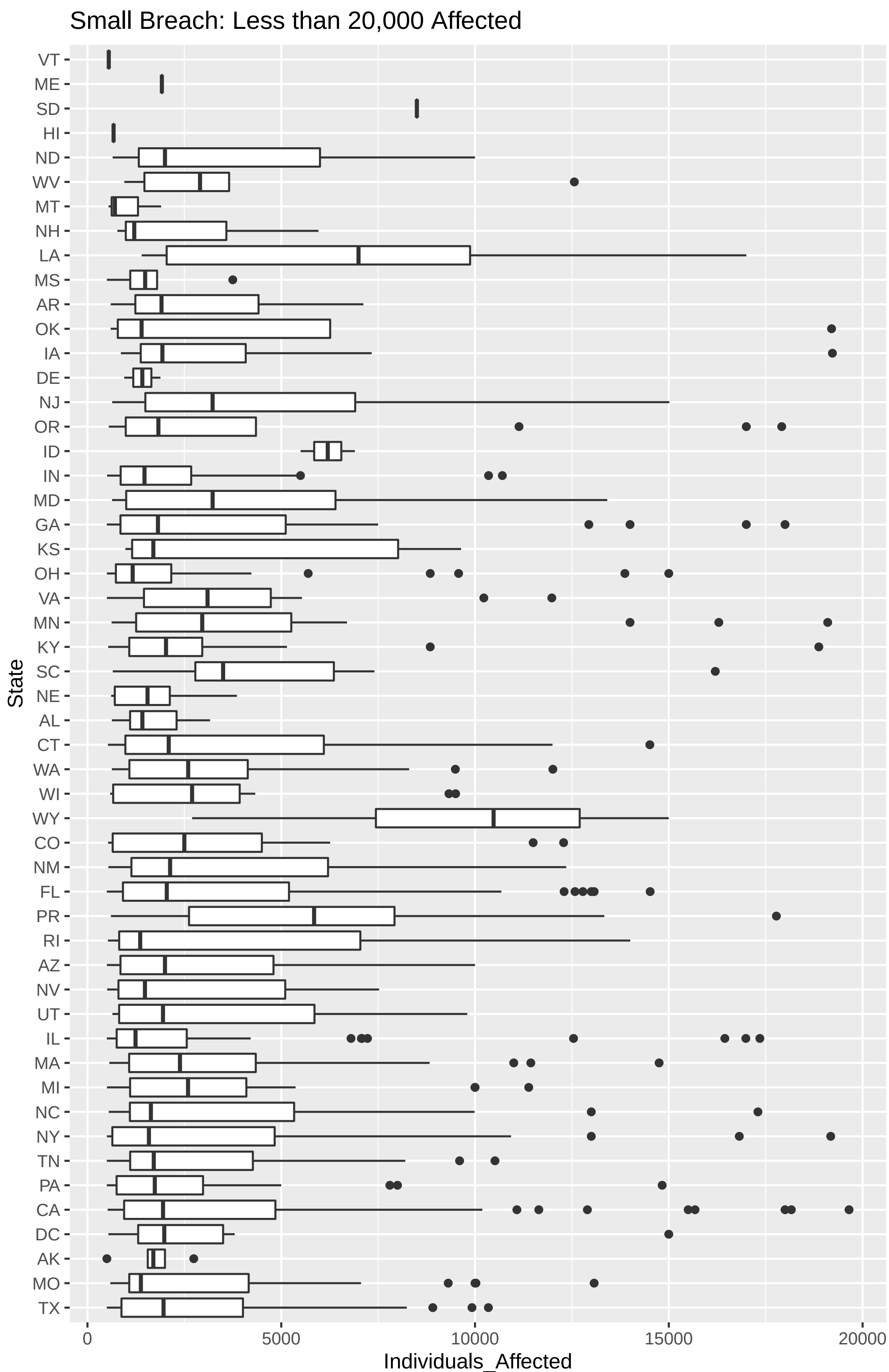
```
##     <dbl>        <dbl>
## 1 227860. 51920099070.
```

**- Discuss what other variables might affect the relationship** Some other variables that may affect the relationship between State and Individuals Affected are type, length of breach, year, and location of breached information, all of which are being explored by other members.

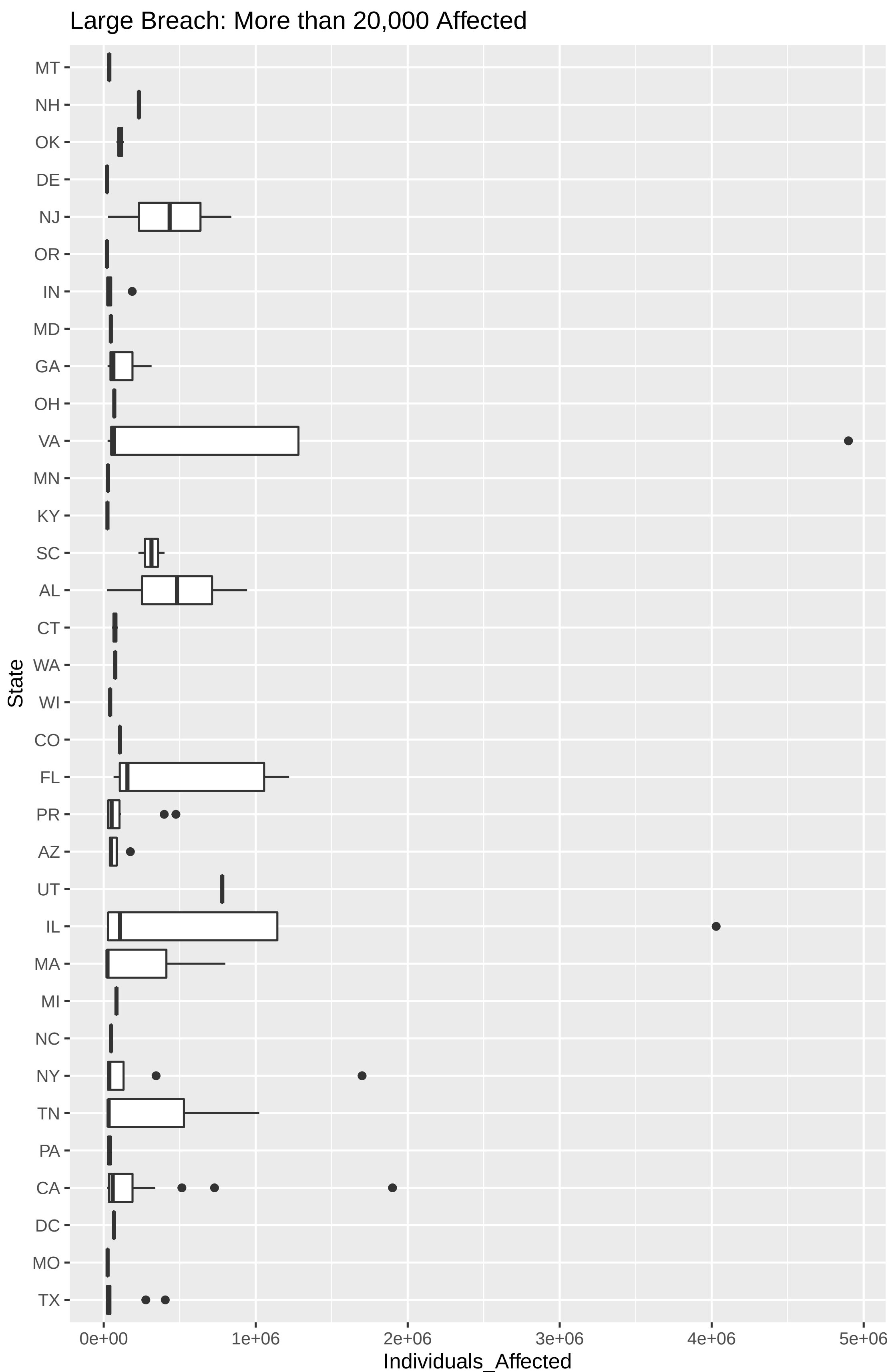
**- Does the relationship change if you look at individual subgroups of the data? Please discuss and demonstrate.**

Looking at whether the breach was below 20,000 people affected or above, gives new insight into the relationship between State and Individuals affected. In the small breaches, WY and LA and PR all stand out with a box plot that has a median higher than the other states. In just the large breaches VA, GA, FL, IL, MA, CA and TN are all positively skewed in the large breach, meaning they have breaches with a larger variation above the median.

```
small_breach <- breaches %>%
  filter(large_affected == FALSE) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breach: Less than 20,000 Affected")

large_breach <- breaches %>%
  filter(large_affected == TRUE) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot()+
  coord_flip() +
  labs(title = "Large Breach: More than 20,000 Affected")

small_breach
```


Small Breach: Less than 20,000 Affected
VT
ME
SD
HI
ND
WV
MT
NH
LA
MS
AR
OK
IA
DE
NJ
OR
ID
IN
MD
GA
KS
OH
VA
MN
KY
SC
NE
AL
CT
WA
WI
WY
CO
NM
FL
PR
RI
AZ
NV
UT
IL
MA
MI
NC
NY
TN
PA
CA
DC
AK
MO
TX
State
0
5000
10000
15000
20000
Individuals_Affected

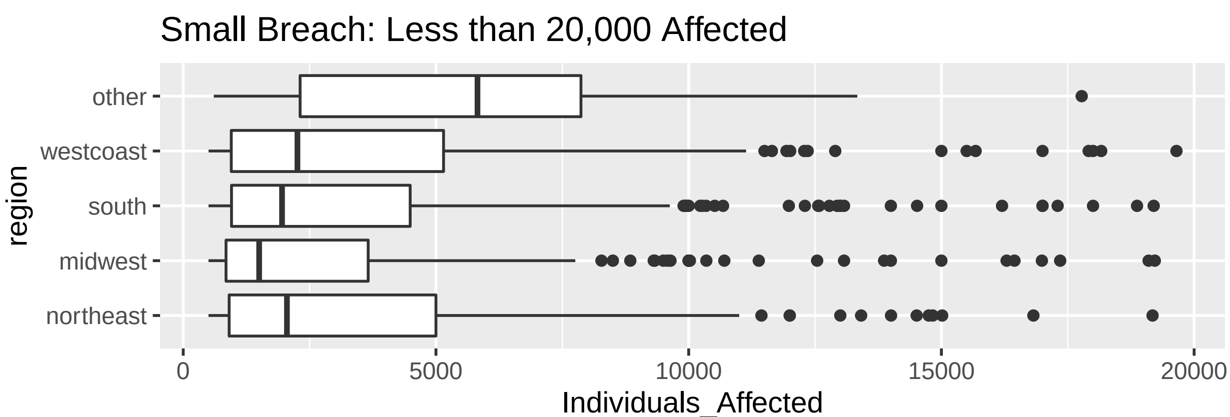
```
large_breach
```

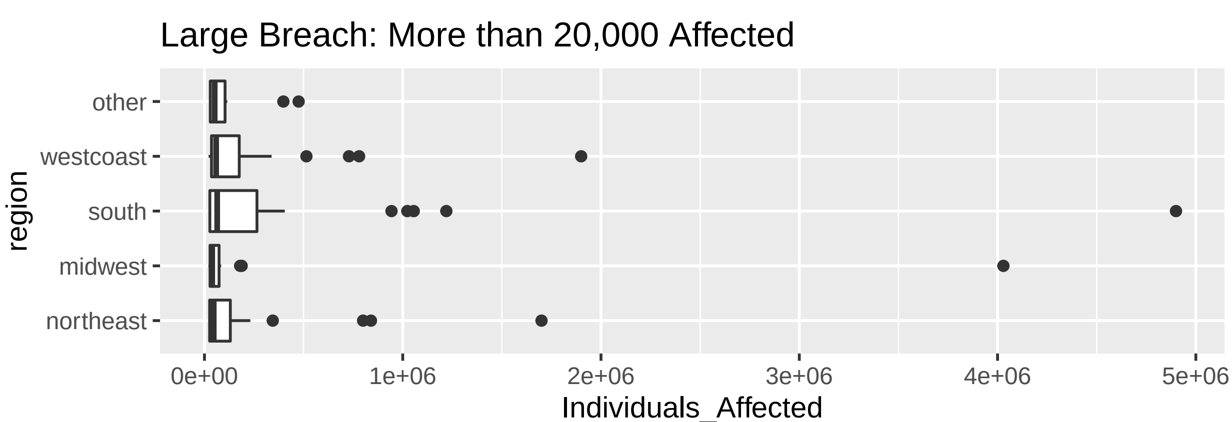


Large Breach: More than 20,000 Affected
State
MT
NH
OK
DE
NJ
OR
IN
MD
GA
OH
VA
MN
KY
SC
AL
CT
WA
WI
CO
FL
PR
AZ
UT
IL
MA
MI
NC
NY
TN
PA
CA
DC
MO
TX
0e+00
1e+06
2e+06
3e+06
4e+06
5e+06
Individuals_Affected

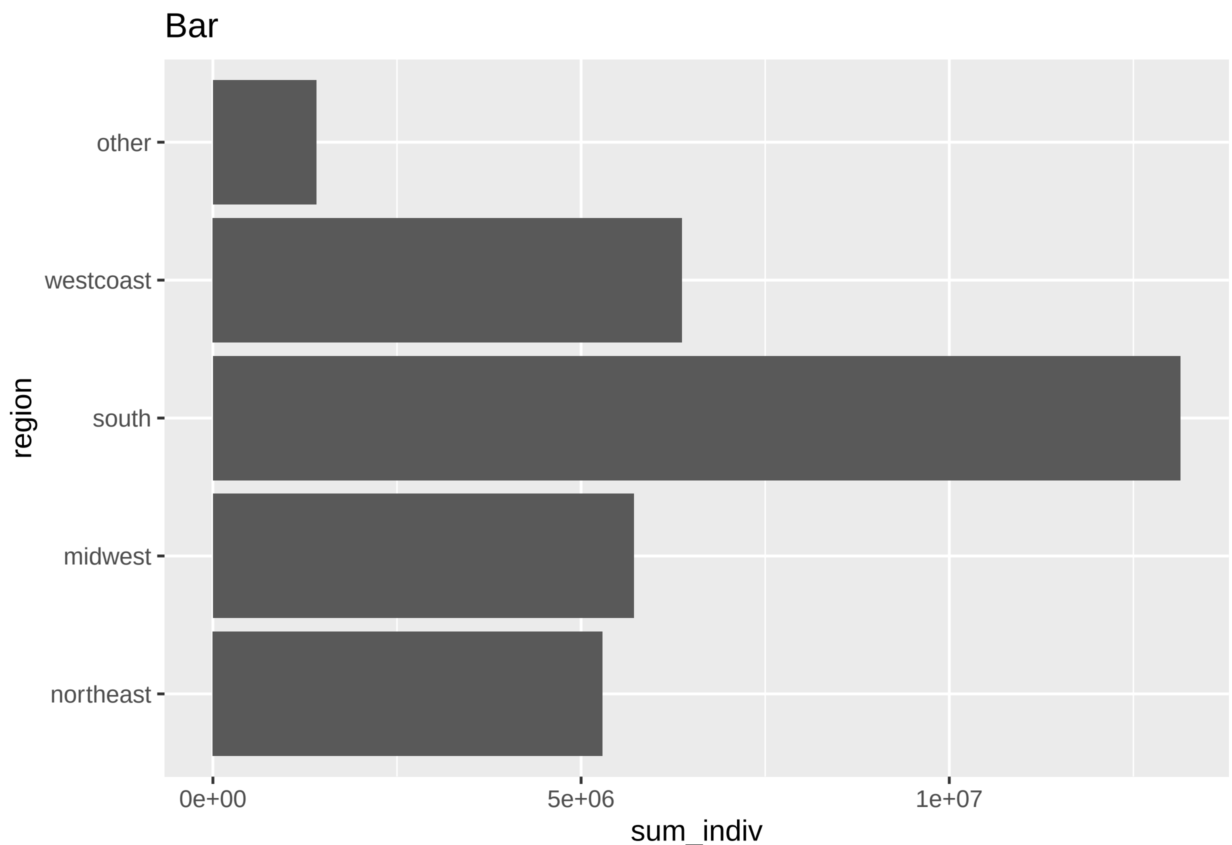
**- Demonstrate if converting the type of these variables help exploring the relationship.** Converting the State to a factor, called region we are able to explore how the distribution and total number of individuals affected changes by region.

```
small_breach_region <- breaches %>%
  filter(large_affected == FALSE) %>%
  ggplot(aes(x=region, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breach: Less than 20,000 Affected")

large_breach_region <- breaches %>%
  filter(large_affected == TRUE) %>%
  ggplot(aes(x=region, y=Individuals_Affected)) +
  geom_boxplot()+
  coord_flip() +
  labs(title = "Large Breach: More than 20,000 Affected")


region_bar <- total_affected_state %>%
  ggplot(aes(x=region, y=sum_indiv)) +
  geom_col()+
  coord_flip() +
  labs(title = "Bar")

ggarrange(small_breach_region, large_breach_region, nrow = 2)
```



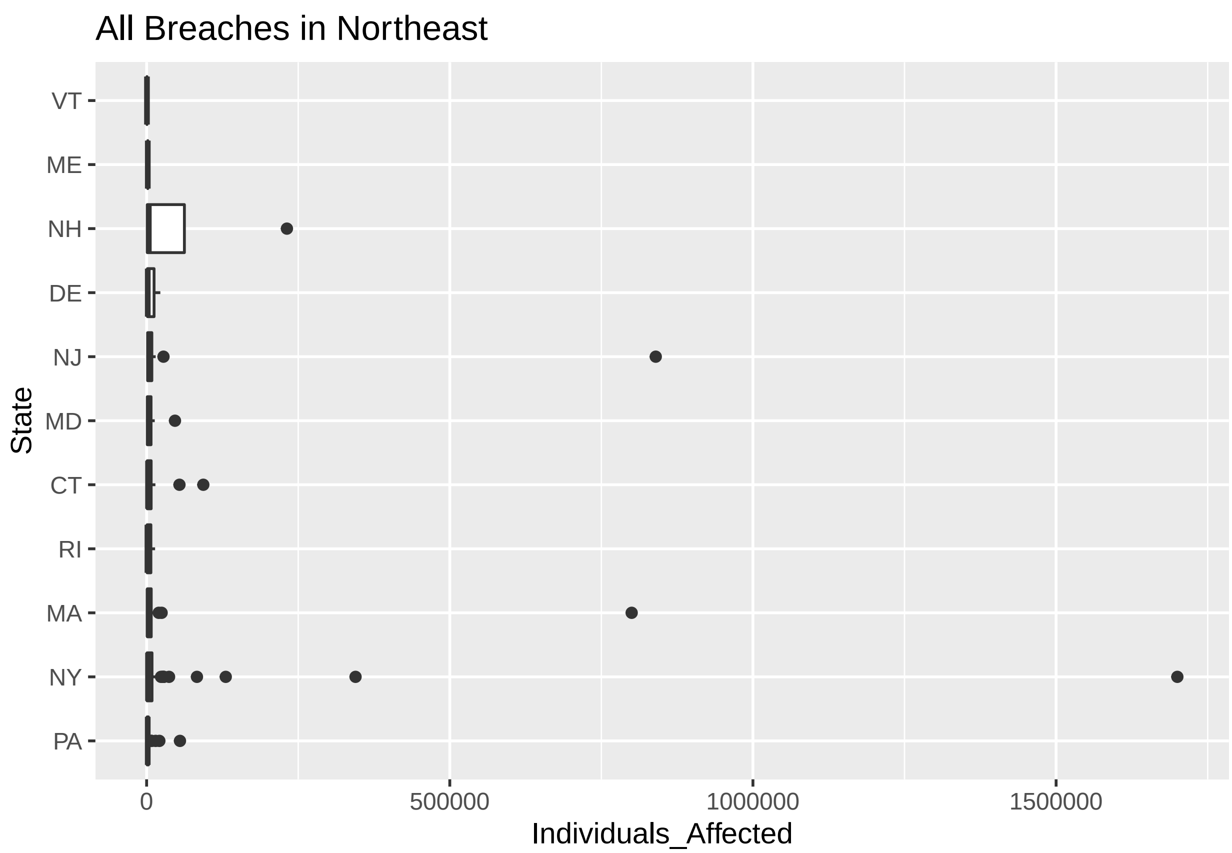
```
region_bar
```



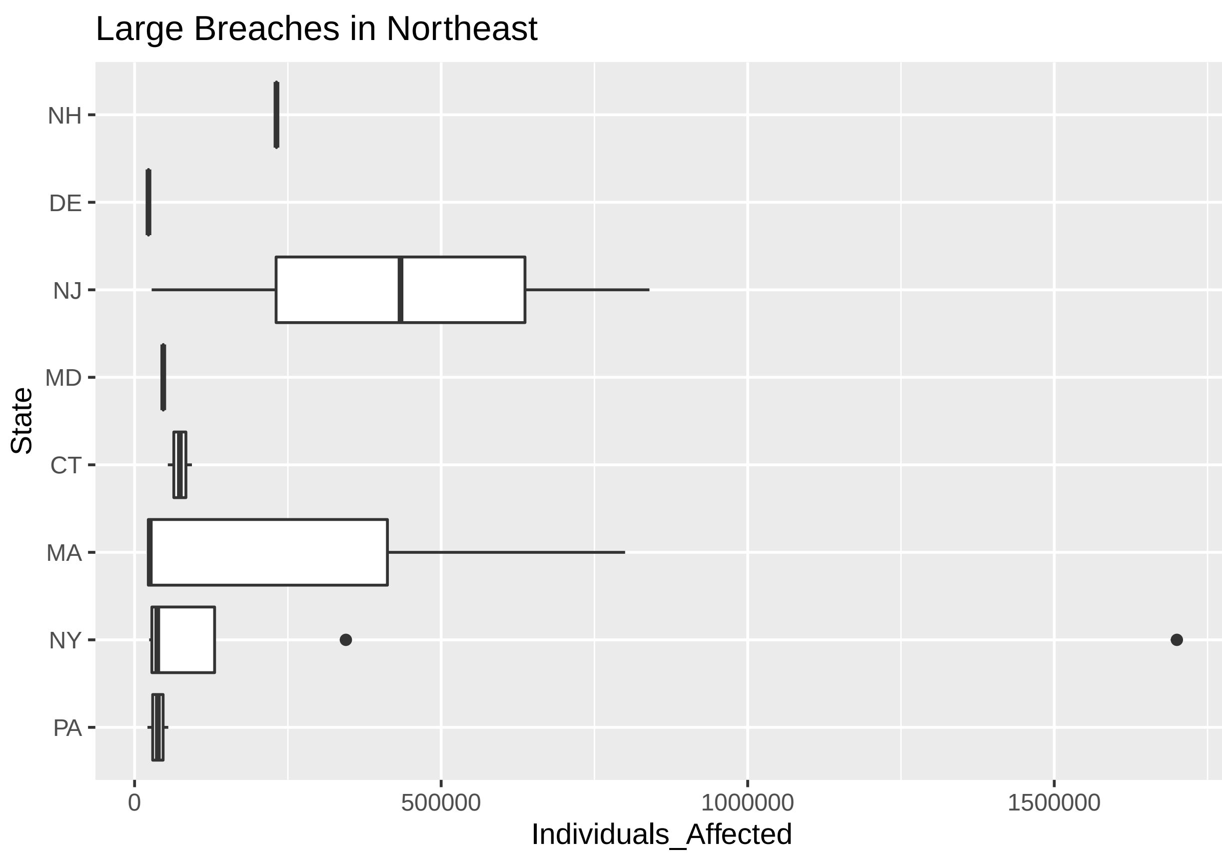

In small breaches, the other states have a higher median value than the other regions, however the total number of individuals affected is the lowest. The south's distribution has the most variation in the large breaches and also has the highest total number of individuals affected.

```
northeast_states <- breaches %>%
  filter(northeast == TRUE) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "All Breaches in Northeast")

large_northeast_states <- breaches %>%
  filter(northeast == TRUE, large_affected == TRUE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Large Breaches in Northeast")

small_northeast_states <- breaches %>%
  filter(northeast == TRUE, large_affected == FALSE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breaches in Northeast")
```

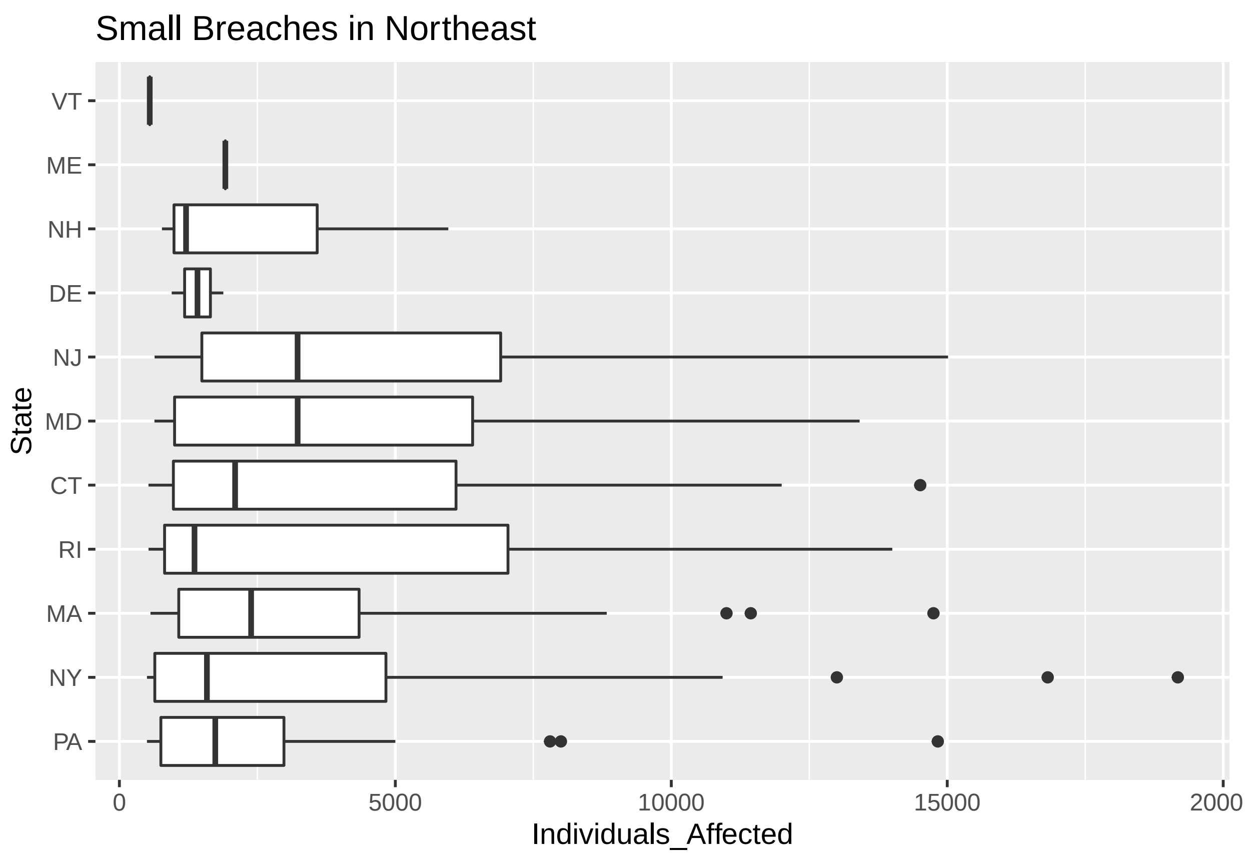
```
northeast_bar <- total_affected_state %>%
  filter(region == "northeast") %>%
  ggplot(aes(x=State, y=sum_indiv)) +
  geom_col()+
  coord_flip() +
  labs(title = "Bar")

northeast_states
```



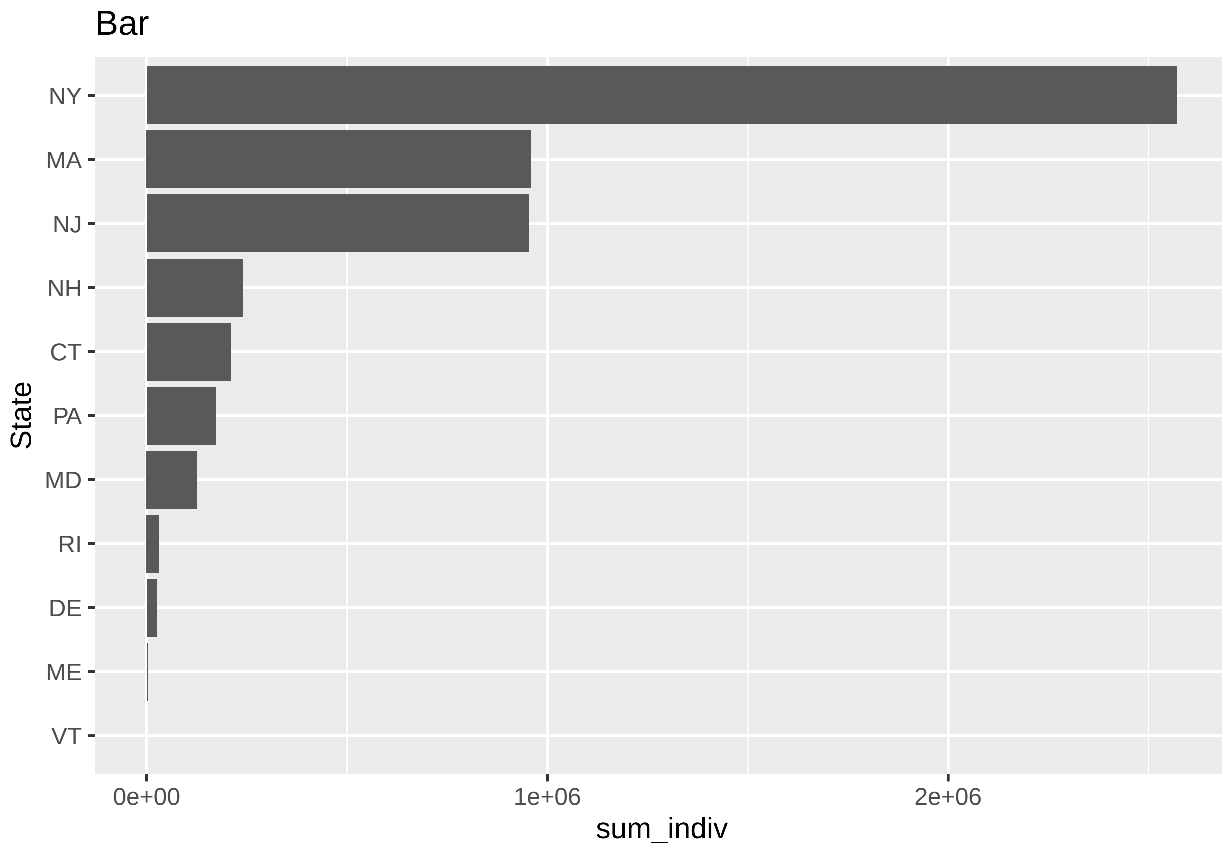

```
large_northeast_states
```




```
small_northeast_states
```



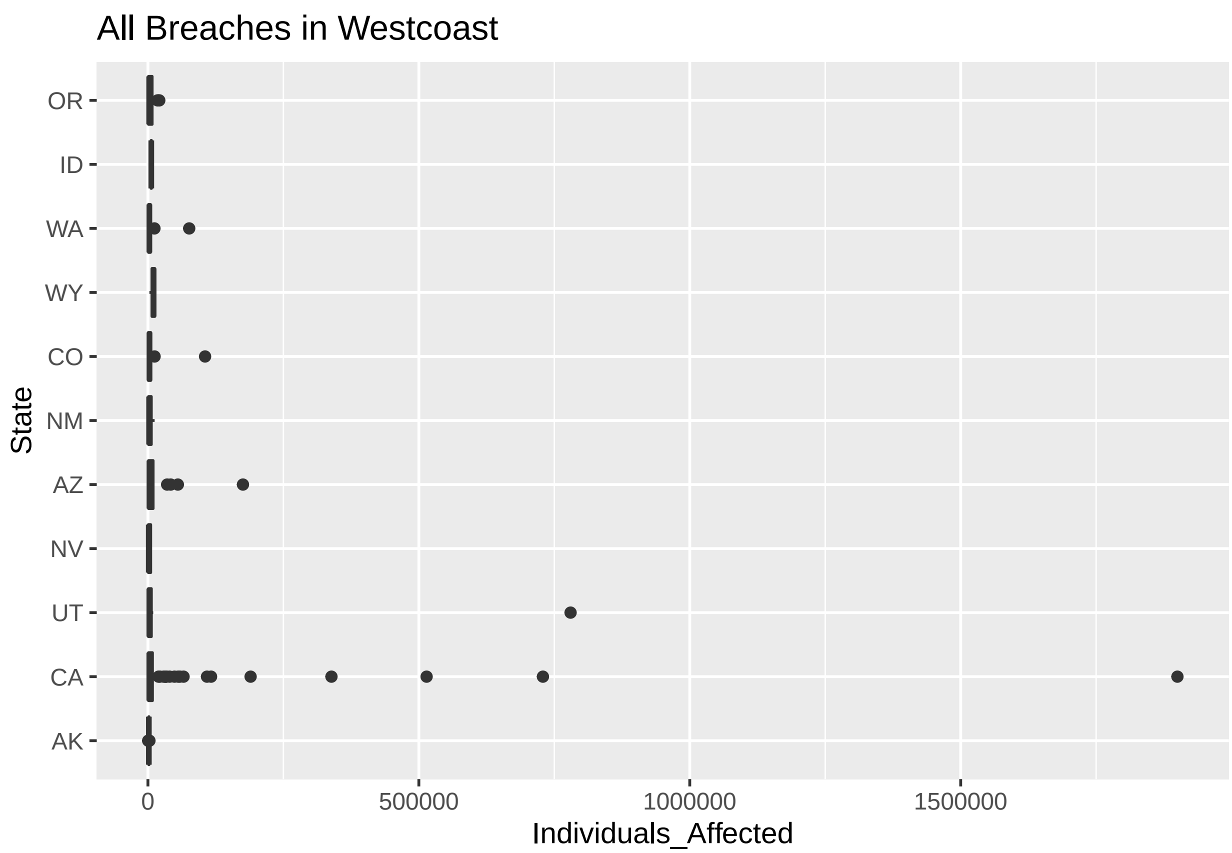

```
northeast_bar
```



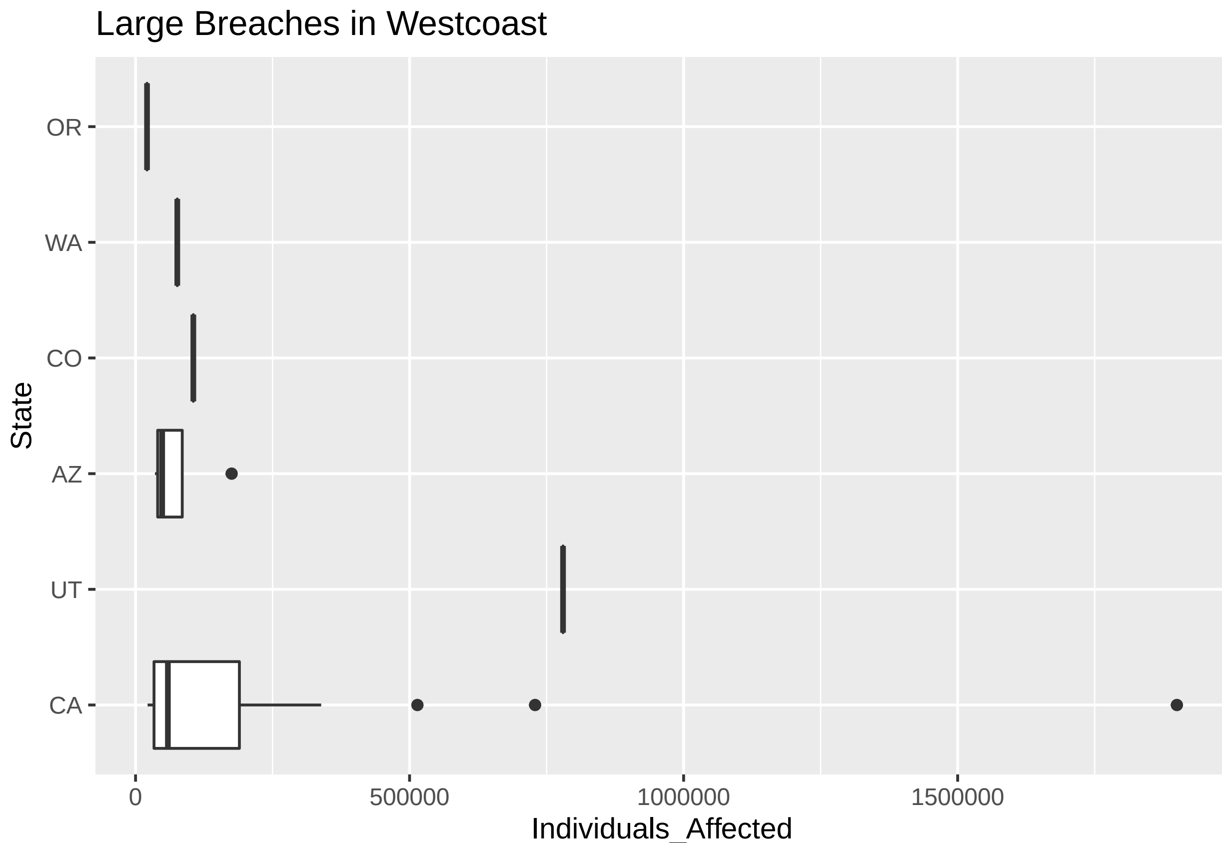

New Jersey has the highest median of individuals affected for large breaches, but does not stand out in small breaches. New York does not stand out in either small or large breaches, however it does have an extremely large outlier value, which makes it the largest total number of individuals affected.

```
westcoast_states <- breaches %>%
  filter(westcoast == TRUE) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "All Breaches in Westcoast")

large_westcoast_states <- breaches %>%
  filter(westcoast == TRUE, large_affected == TRUE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Large Breaches in Westcoast")

small_westcoast_states <- breaches %>%
  filter(westcoast == TRUE, large_affected == FALSE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breaches in Westcoast")

westcoast_bar <- total_affected_state %>%
  filter(region == "westcoast") %>%
```

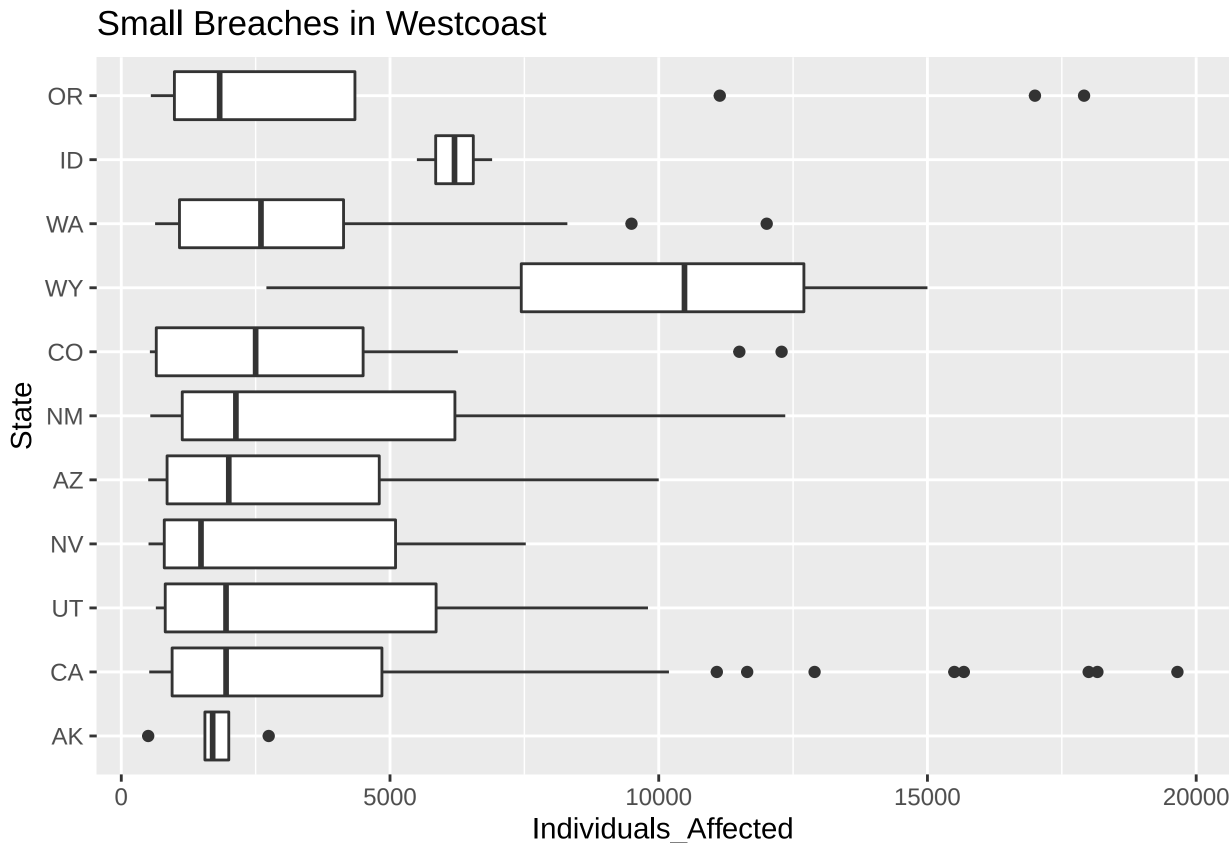
```
  ggplot(aes(x=State, y=sum_indiv)) +
  geom_col()+
  coord_flip() +
  labs(title = "Total Indivduals Affected by State in Westcoast")

westcoast_states
```



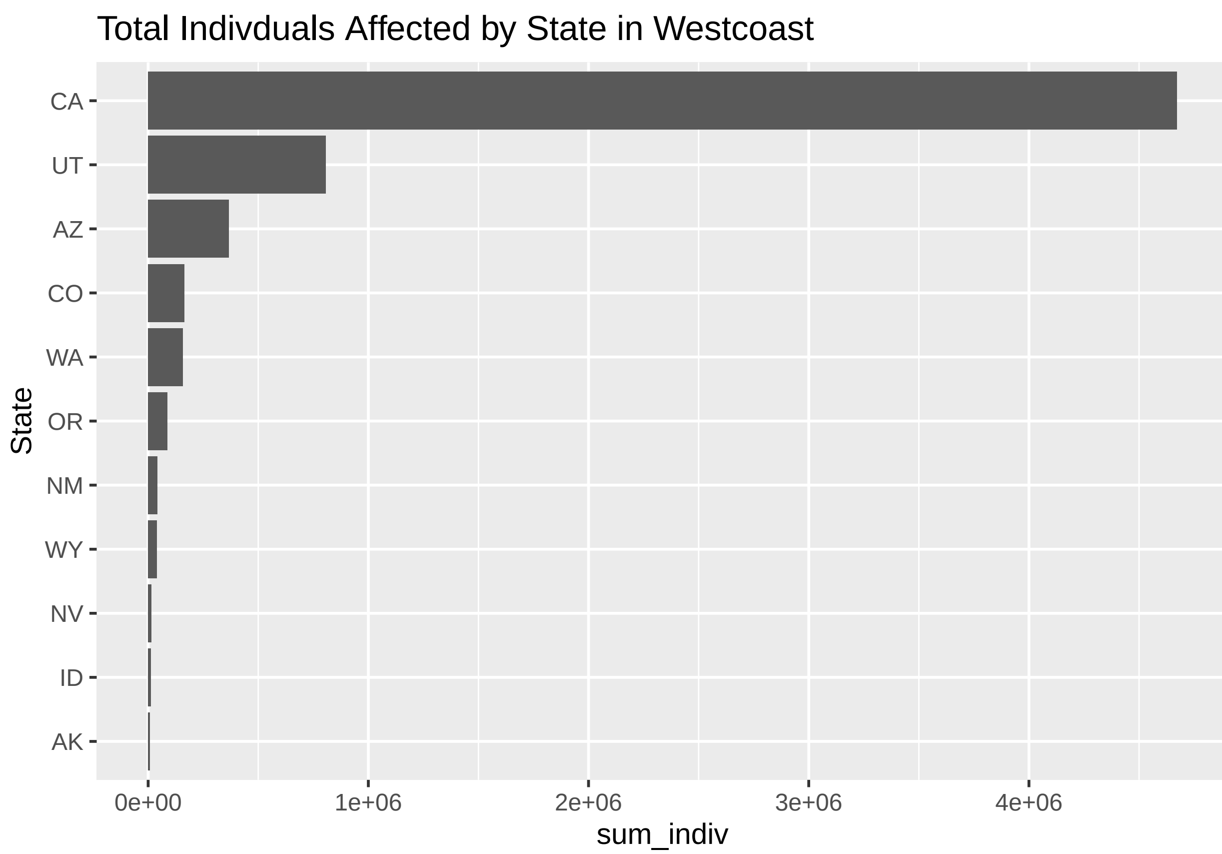

```
large_westcoast_states
```




```
small_westcoast_states
```



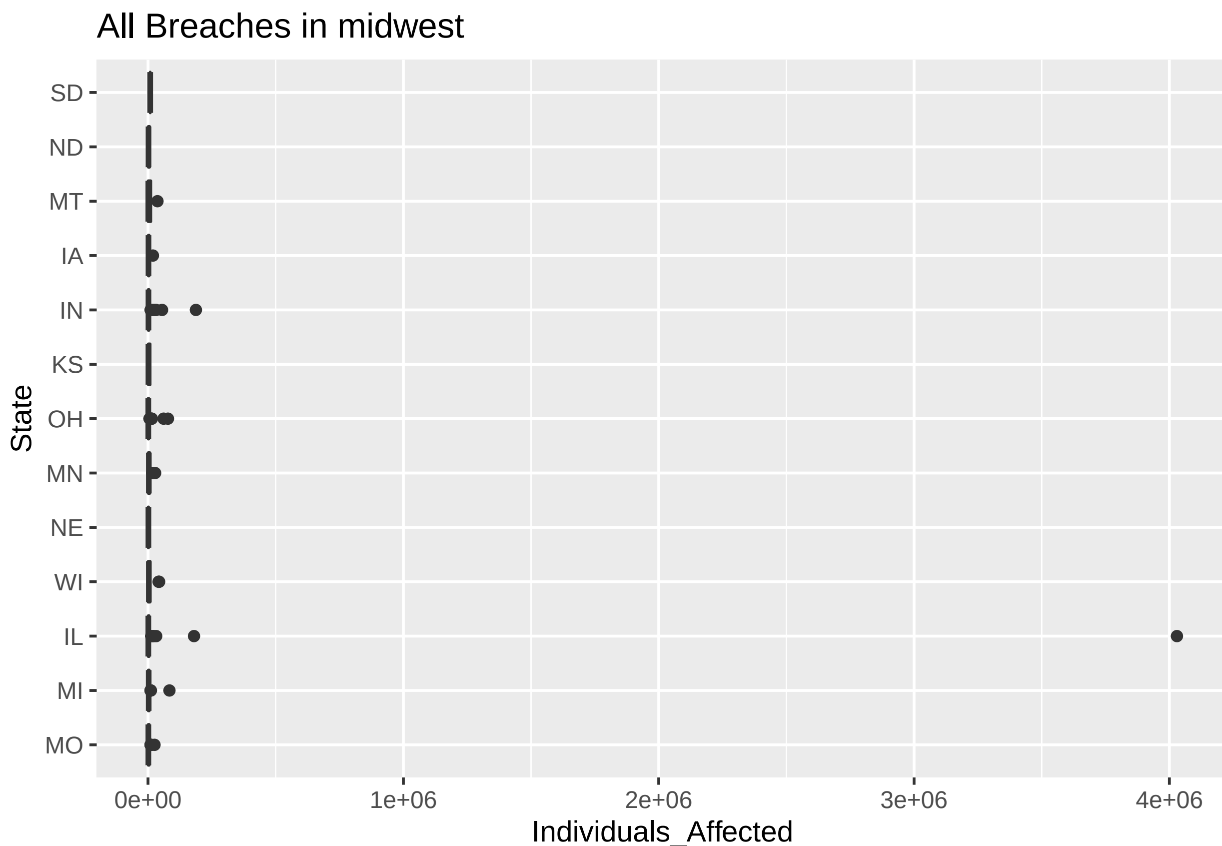

```
westcoast_bar
```



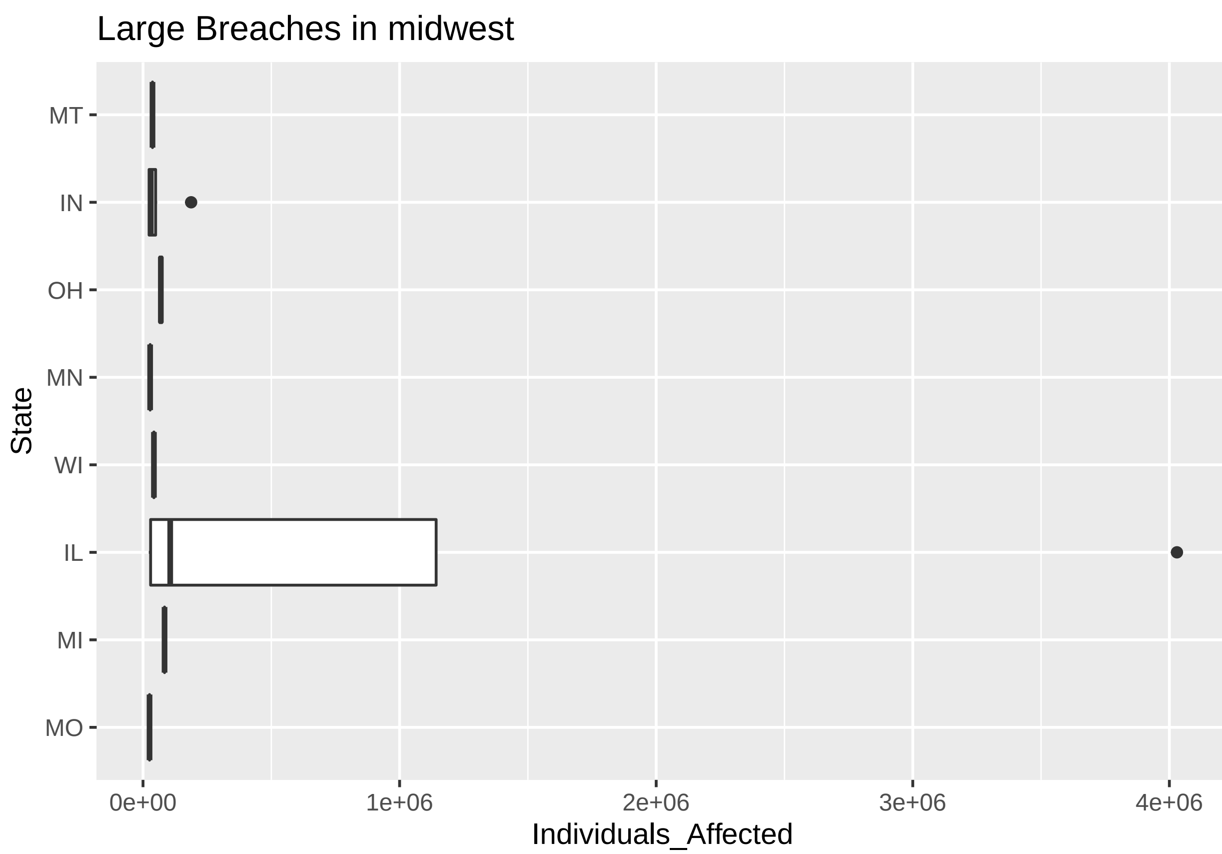

There are not many large breaches on the westcoast, with CA as the exception, having 3 outlier breaches that lead to CA having the highest total number of individuals affected on the Westcoast. Overall the small breaches have a median of slightly below 2500 indivuals affected, with ID and WY standing out and having a higher median. However both ID and WY are two of the lowest total number of individuals affected.

```
midwest_states <- breaches %>%
  filter(midwest == TRUE) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "All Breaches in midwest")

large_midwest_states <- breaches %>%
  filter(midwest == TRUE, large_affected == TRUE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Large Breaches in midwest")

small_midwest_states <- breaches %>%
  filter(midwest == TRUE, large_affected == FALSE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breaches in midwest")

midwest_bar <- total_affected_state %>%
```

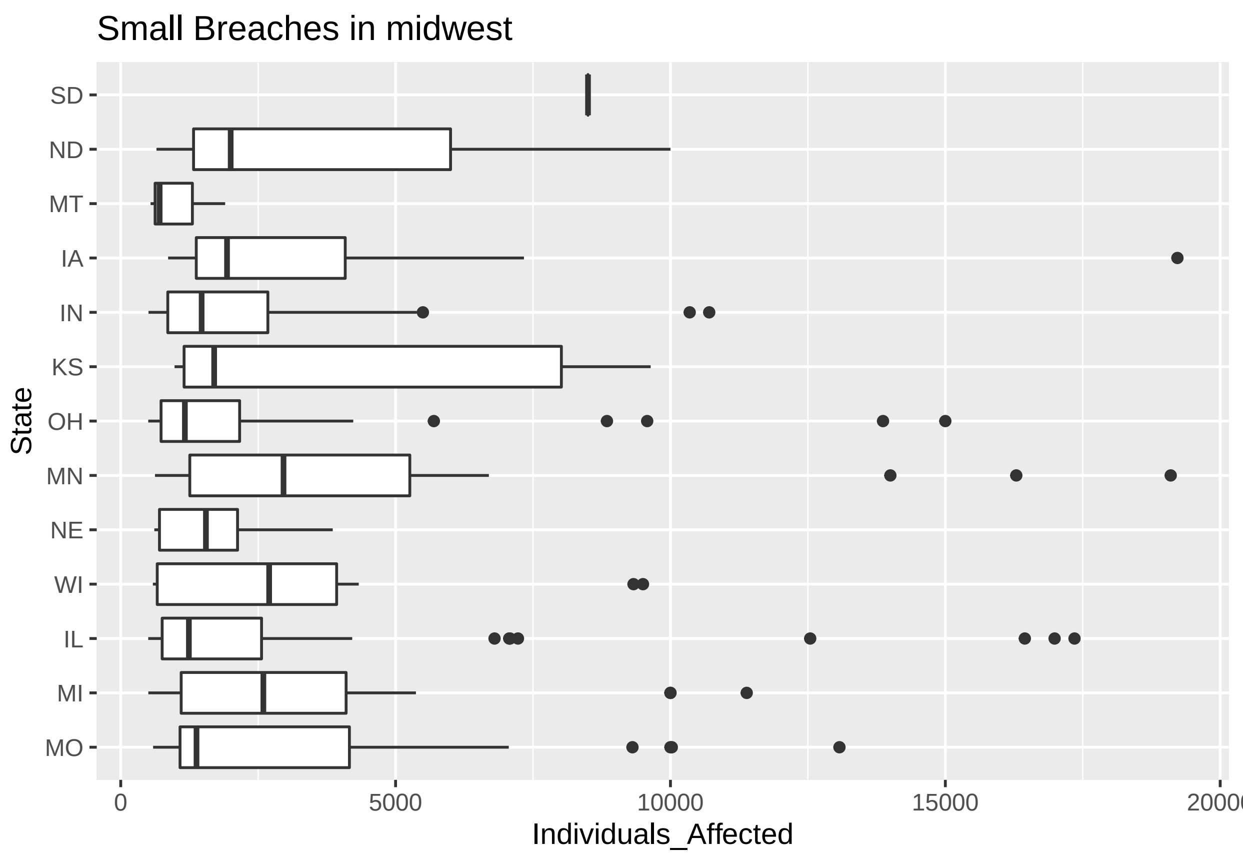
```r
  filter(region == "midwest") %>%
  ggplot(aes(x=State, y=sum_indiv)) +
  geom_col()+
  coord_flip() +
  labs(title = "Total Indivduals Affected by State in midwest")

midwest_states
```



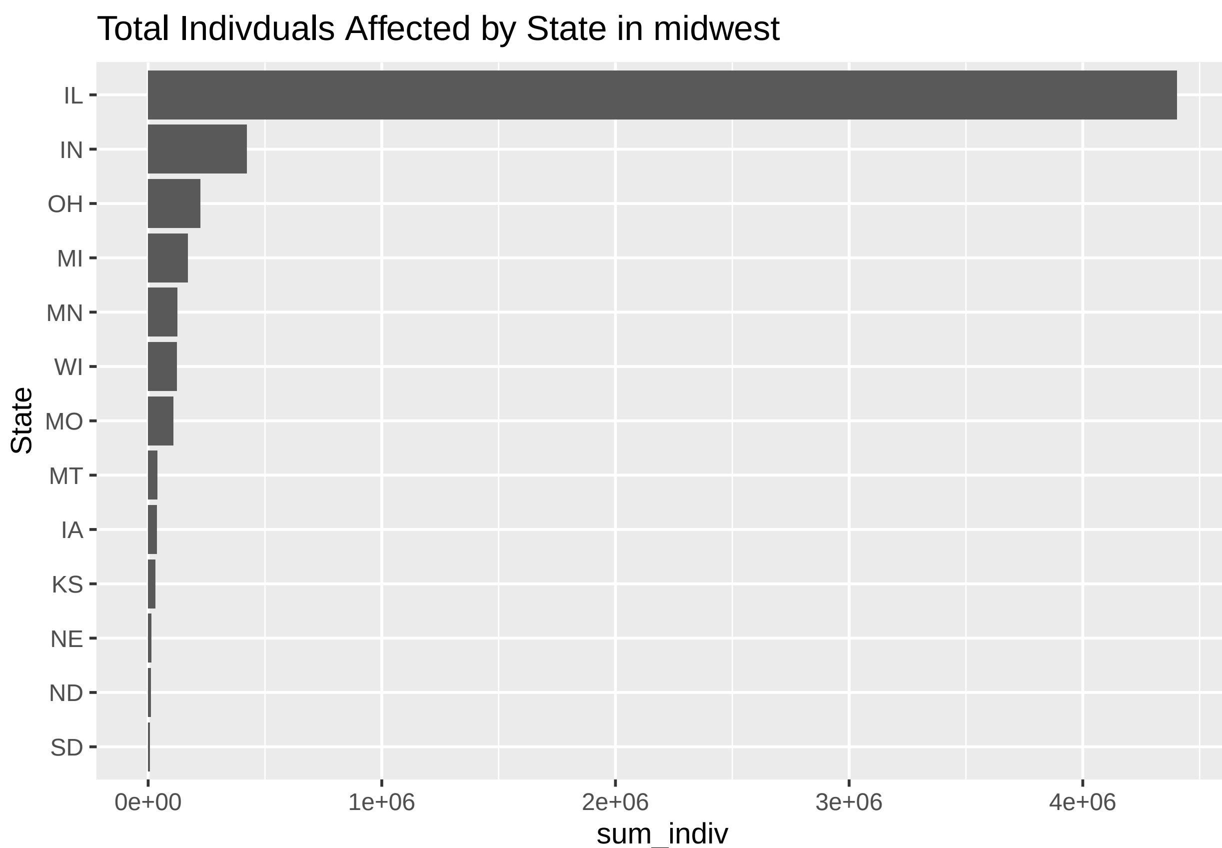

```r
large_midwest_states
```




```
small_midwest_states
```


Small Breaches in midwest
State
SD
ND
MT
IA
IN
KS
OH
MN
NE
WI
IL
MI
MO
0
5000
10000
15000
2000
Individuals_Affected

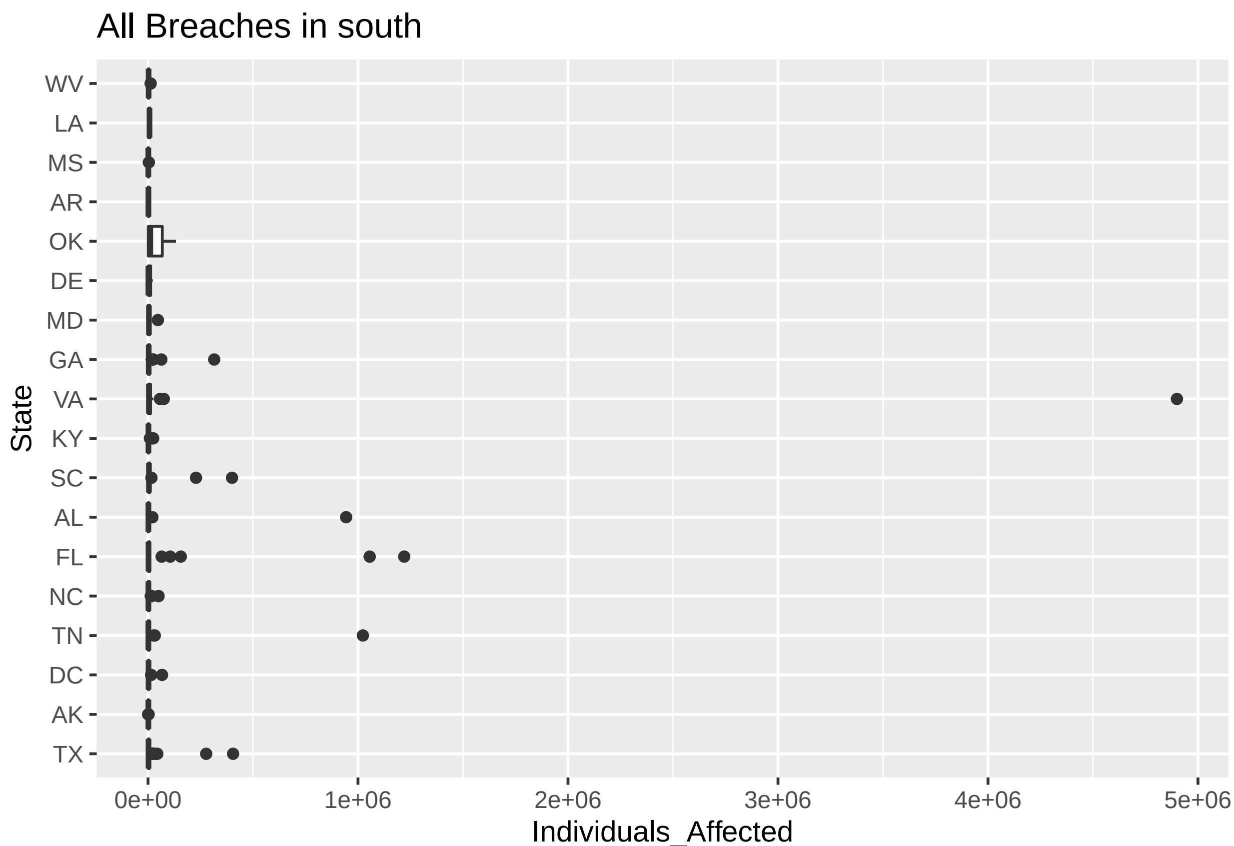
```
midwest_bar
```



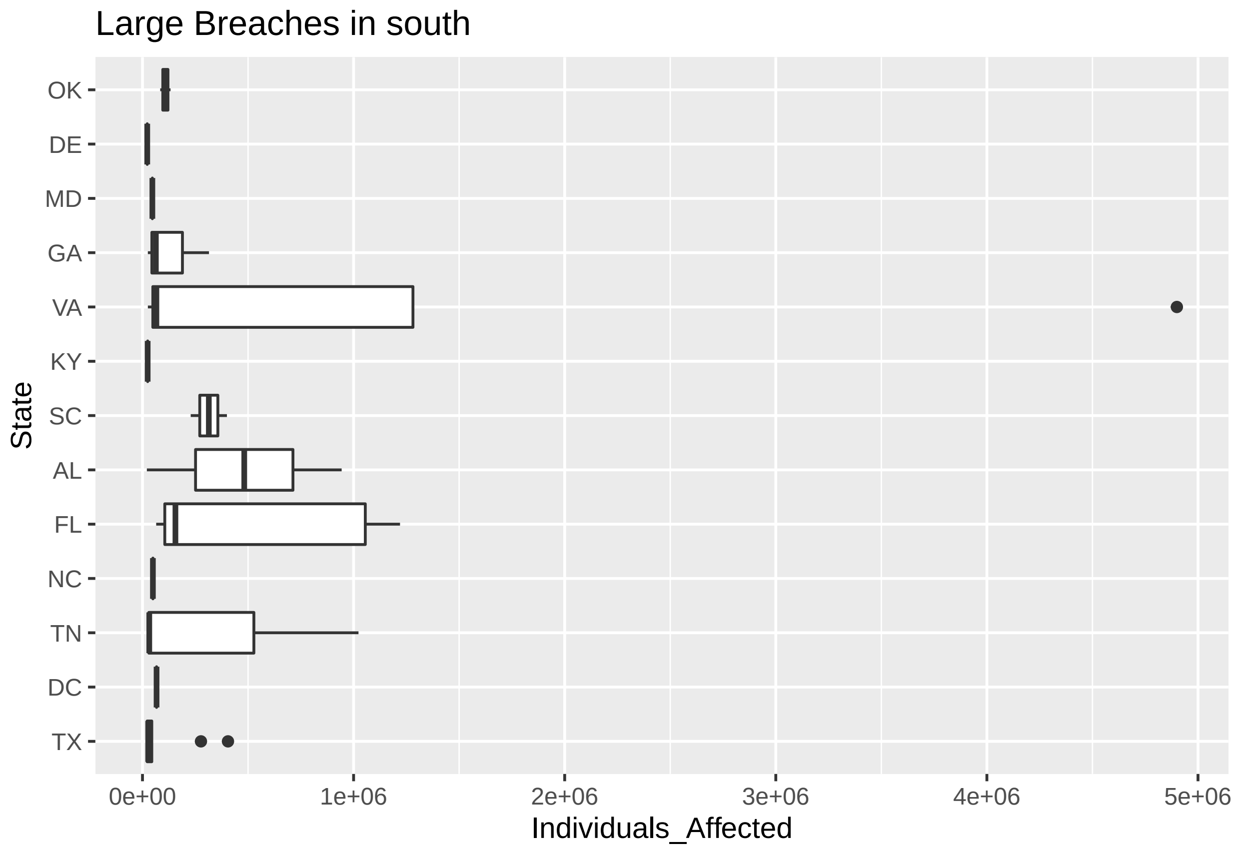

In the small breaches, there are many outlier values, but no states median is significantly higher than any of the others. SD is the exception, but there is only one small breach and that value is therefore the median. IL stands out in the large breaches, having the widest distribution and the largest outlier value. IL is significantly higher in the total number of individuals affected.

```
south_states <- breaches %>%
  filter(south == TRUE) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "All Breaches in south")

large_south_states <- breaches %>%
  filter(south == TRUE, large_affected == TRUE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Large Breaches in south")

small_south_states <- breaches %>%
  filter(south == TRUE, large_affected == FALSE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breaches in south")

south_bar <- total_affected_state %>%
```

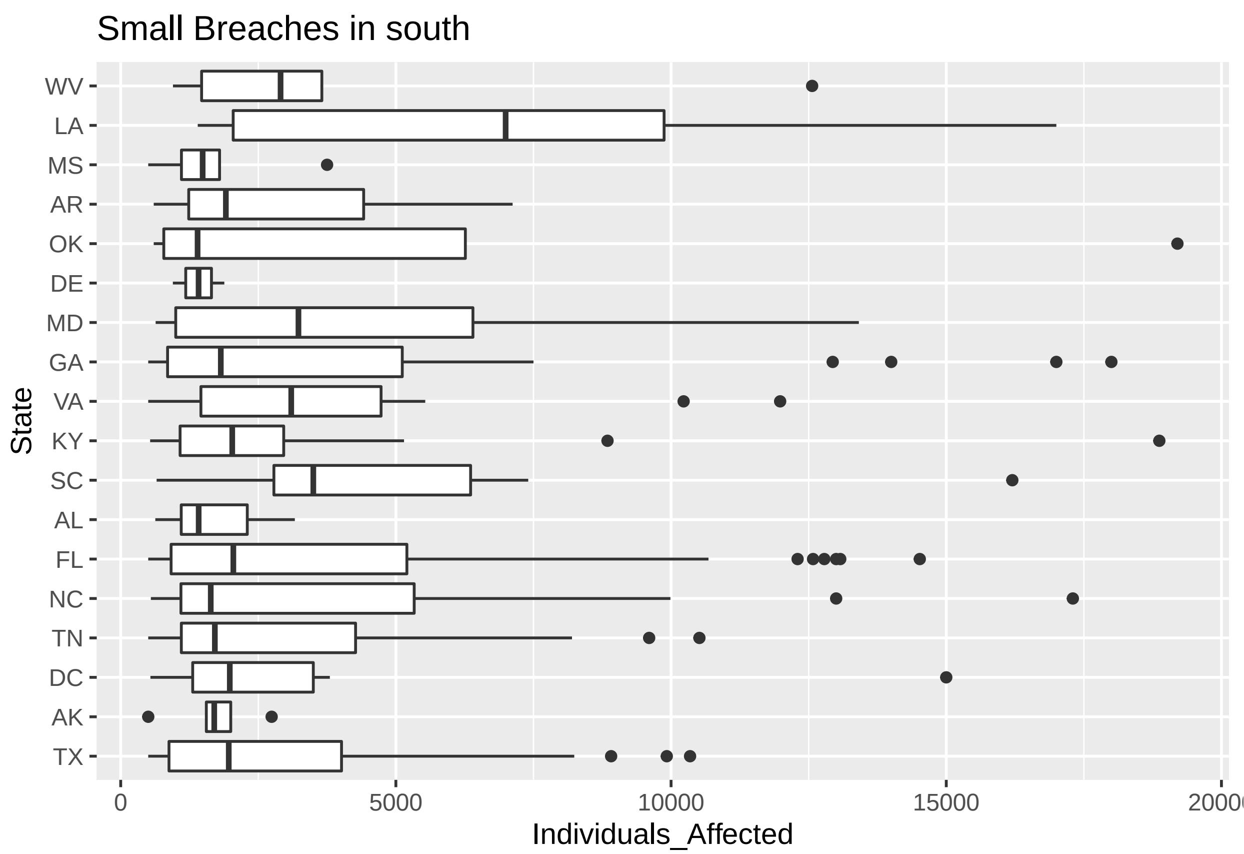
```
  filter(region == "south") %>%
  ggplot(aes(x=State, y=sum_indiv)) +
  geom_col()+
  coord_flip() +
  labs(title = "Total Indivduals Affected by State in south")

south_states
```



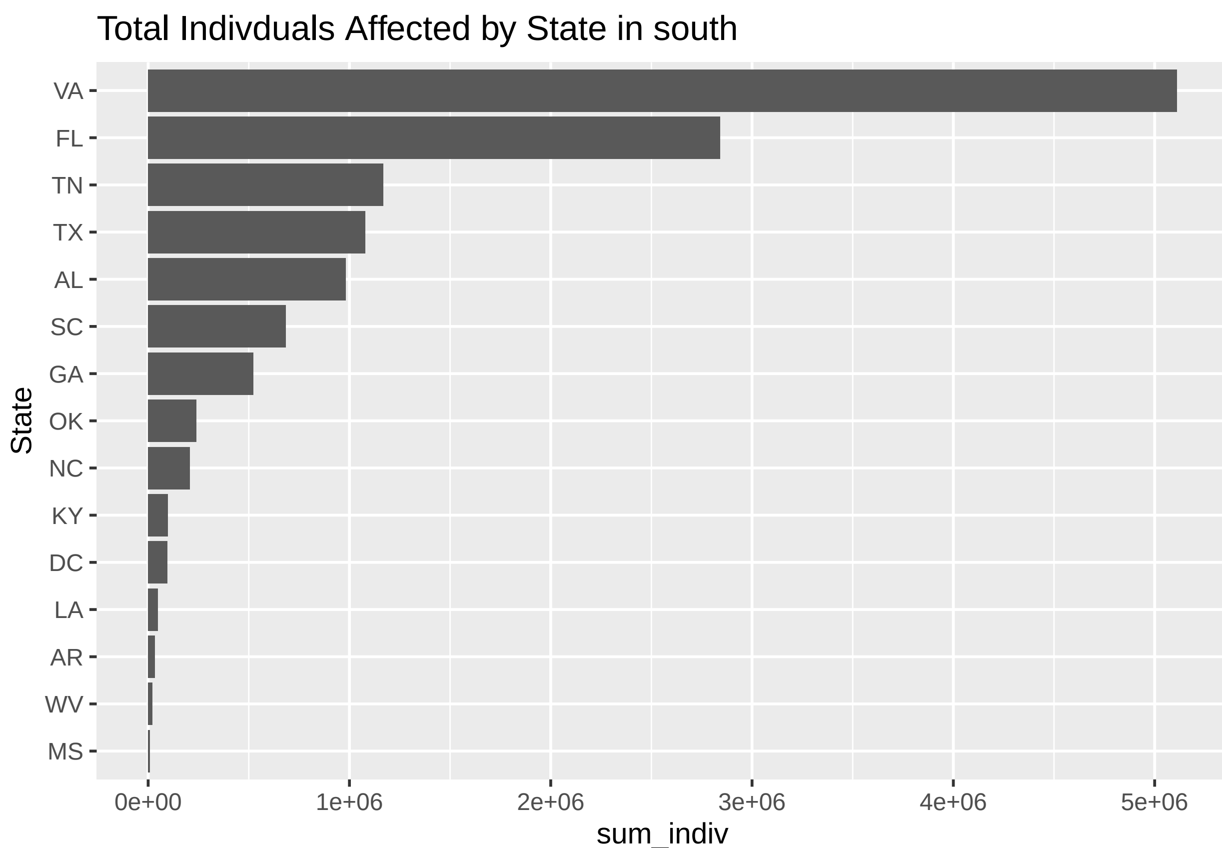

```
large_south_states
```




```
small_south_states
```


Small Breaches in south
State
WV
LA
MS
AR
OK
DE
MD
GA
VA
KY
SC
AL
FL
NC
TN
DC
AK
TX
0
5000
10000
15000
20000
Individuals_Affected

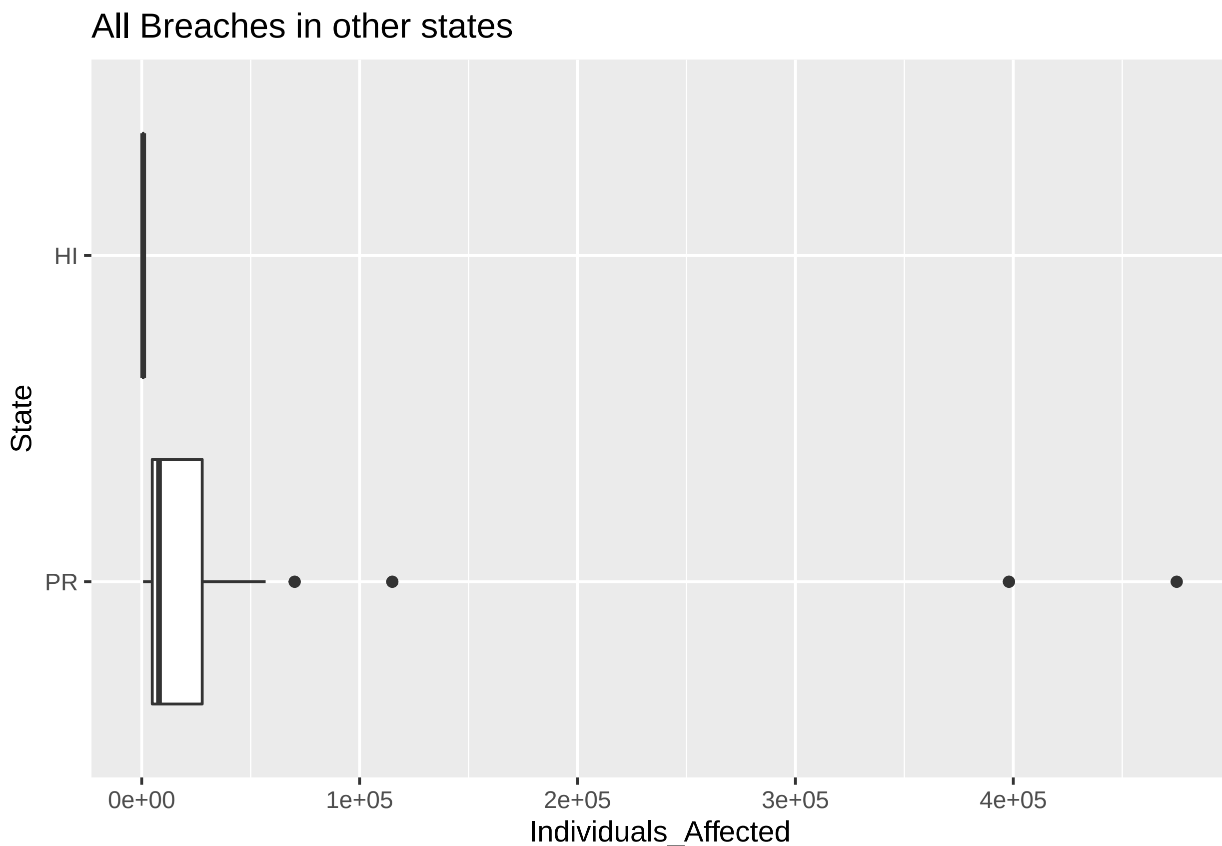
```
south_bar
```



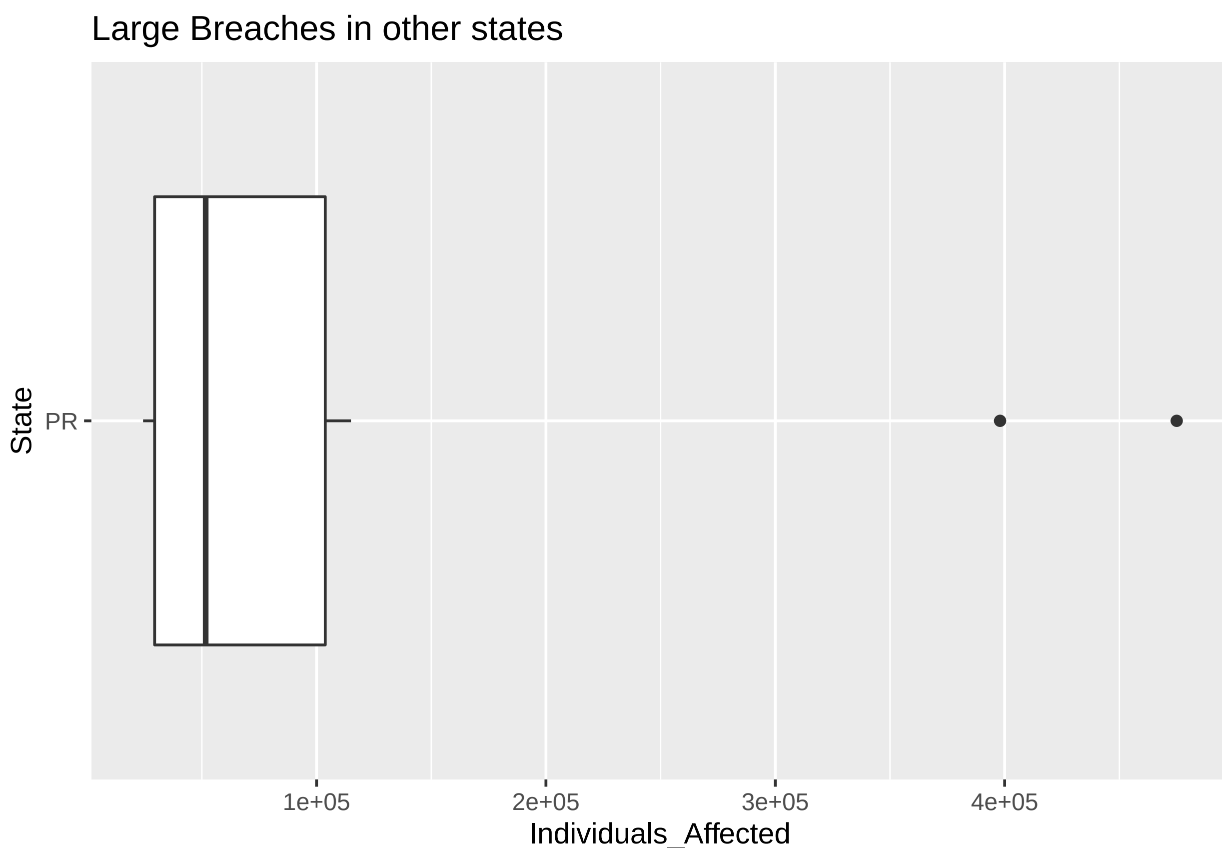

In the south the top 5 states by total individuals affected are VA, FL, TN, TX, and AL, all of which have distribution that are more spread out in large breaches, other than TX. Texas does have 2 outlier values in the large breaches that bring the total individuals affected up. In smaller breaches, LA has a higher median, but the total number of individuals affected is one of the lowest in the south.

```
other_states <- breaches %>%
  filter(region == "other") %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "All Breaches in other states")

large_other_states <- breaches %>%
  filter(region == "other", large_affected == TRUE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Large Breaches in other states")

small_other_states <- breaches %>%
  filter(region == "other", large_affected == FALSE ) %>%
  ggplot(aes(x=State, y=Individuals_Affected)) +
  geom_boxplot() +
  coord_flip() +
  labs(title = "Small Breaches in other states")

other_bar <- total_affected_state %>%
```

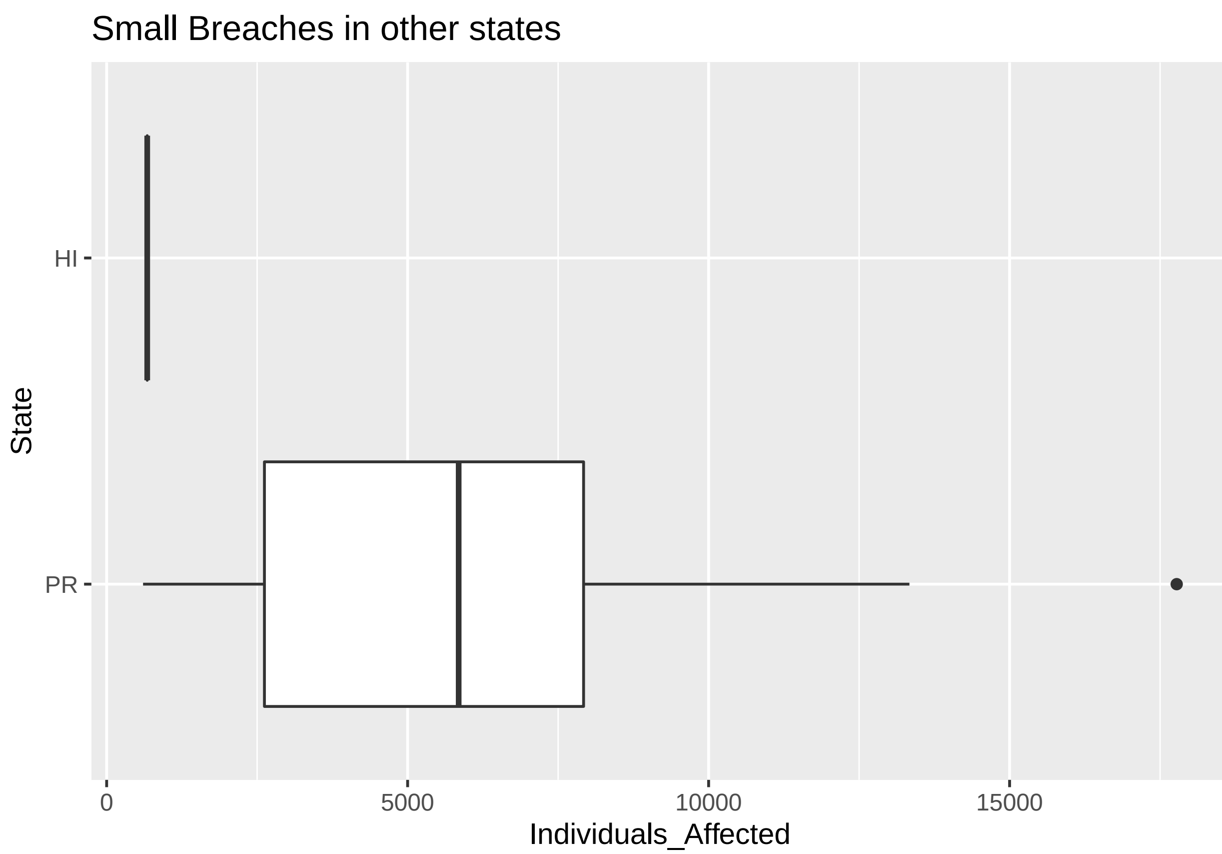
```
  filter(region == "other") %>%
  ggplot(aes(x=State, y=sum_indiv)) +
  geom_col()+
  coord_flip() +
  labs(title = "Total Indivduals Affected by State in other states")

other_states
```



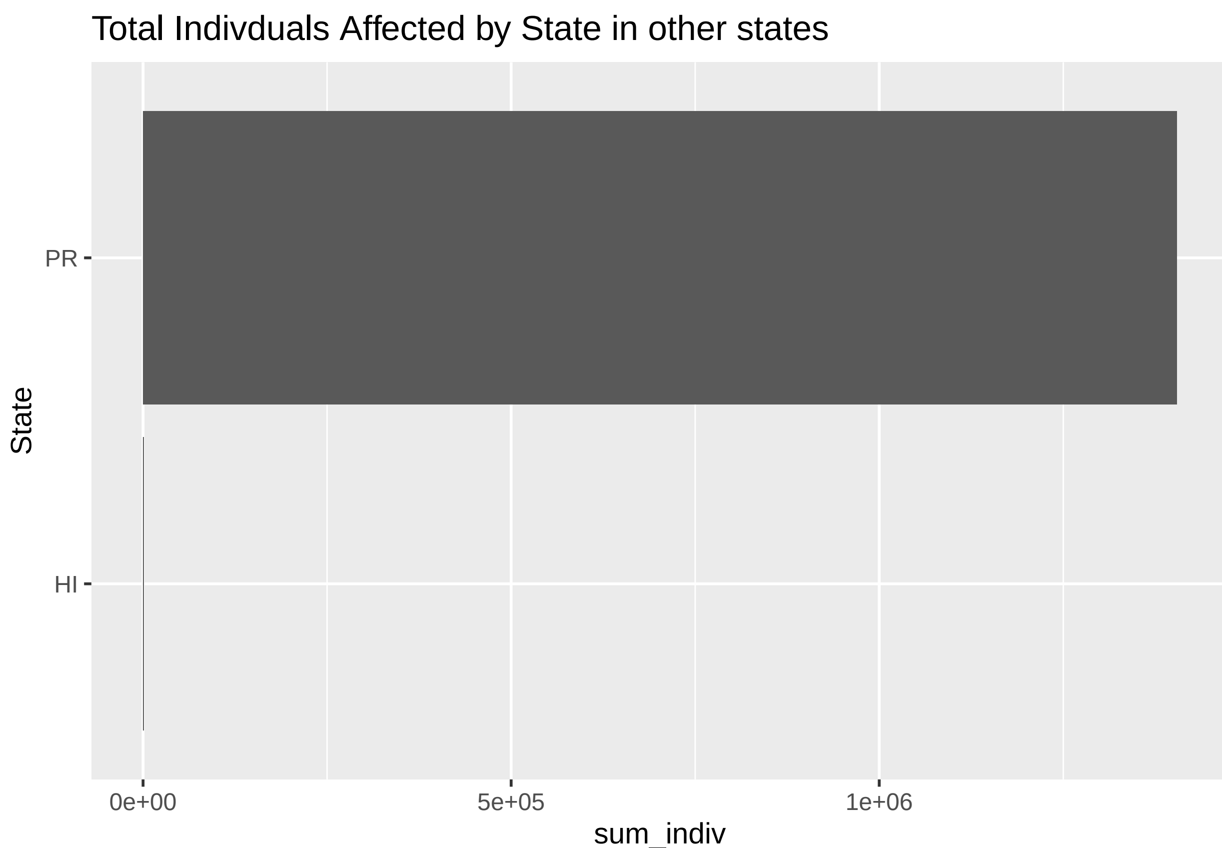

```
large_other_states
```




```
small_other_states
```


Small Breaches in other states
HI
PR
State
0
5000
10000
15000
Individuals_Affected

```
other_bar
```




The only state that truly plays a role in breaches in other is PR, which has some outlier individuals affected in the large breach, that brings the total number of individuals up. HI only had 1 breach and it was a small breach.

**Normalized individuals** Since each region has different total populations, looking at the normalized Individuals Affected gives a better picture of how impactful the breaches were. Also the other region only includes PR and HI so any breach

```
state_populations <- read_csv("state-population.csv")
```

```
## Warning: Missing column names filled in: 'X3' [3], 'X4' [4]
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   State = col_character(),
##   Population = col_number(),
##   X3 = col_logical(),
##   X4 = col_logical()
## )
```

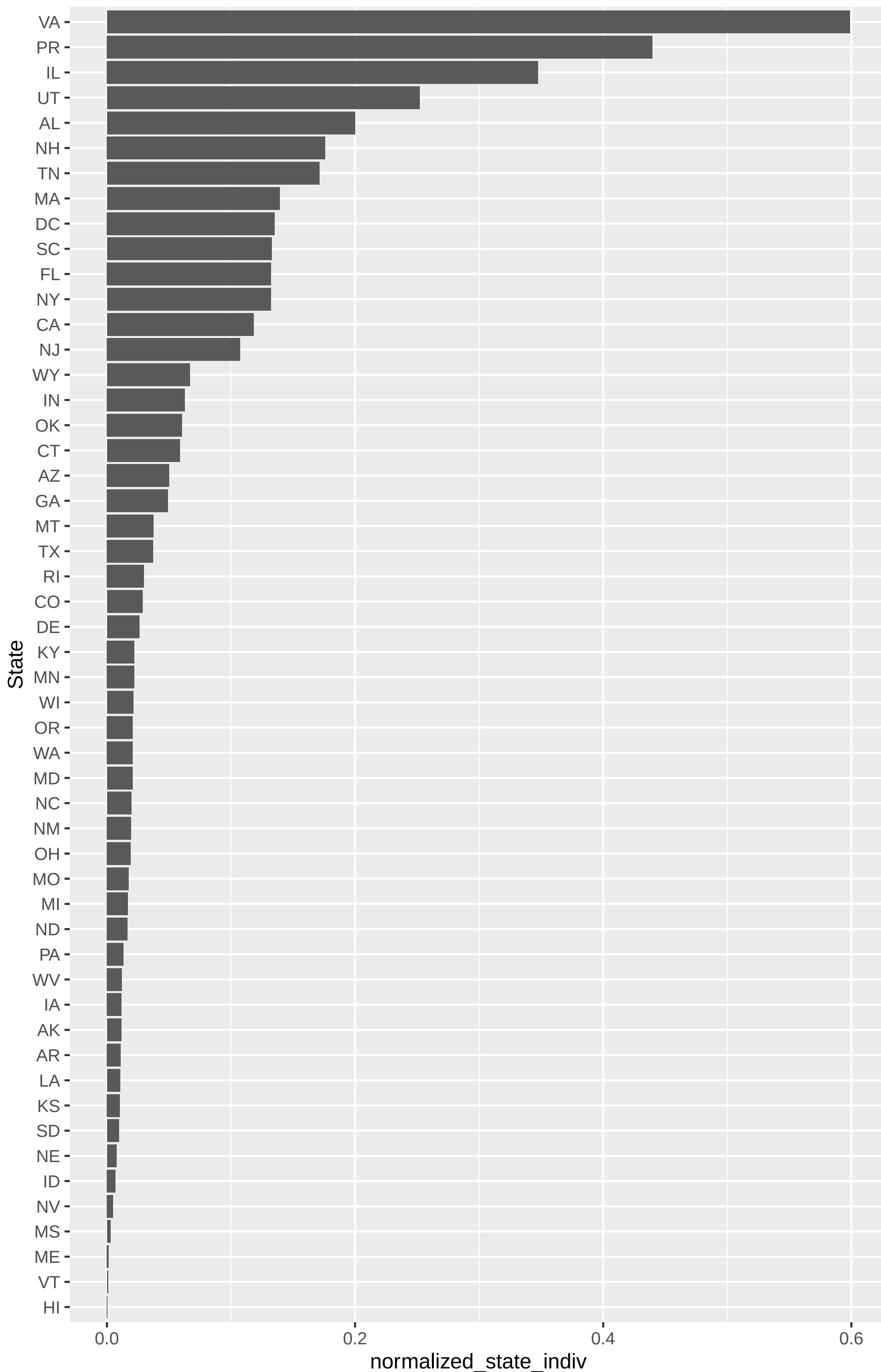
```
state_populations <- state_populations %>% select(State, Population)
```

```
total_affected_state <- total_affected_state %>%
  left_join(state_populations)
```

```
## Joining, by = "State"
```

```
total_affected_state <- total_affected_state %>%
  mutate(normalized_state_indiv = sum_indiv / Population)

total_affected_state_sorted <- total_affected_state[order(total_affected_state$normalized_state_indiv),]
```

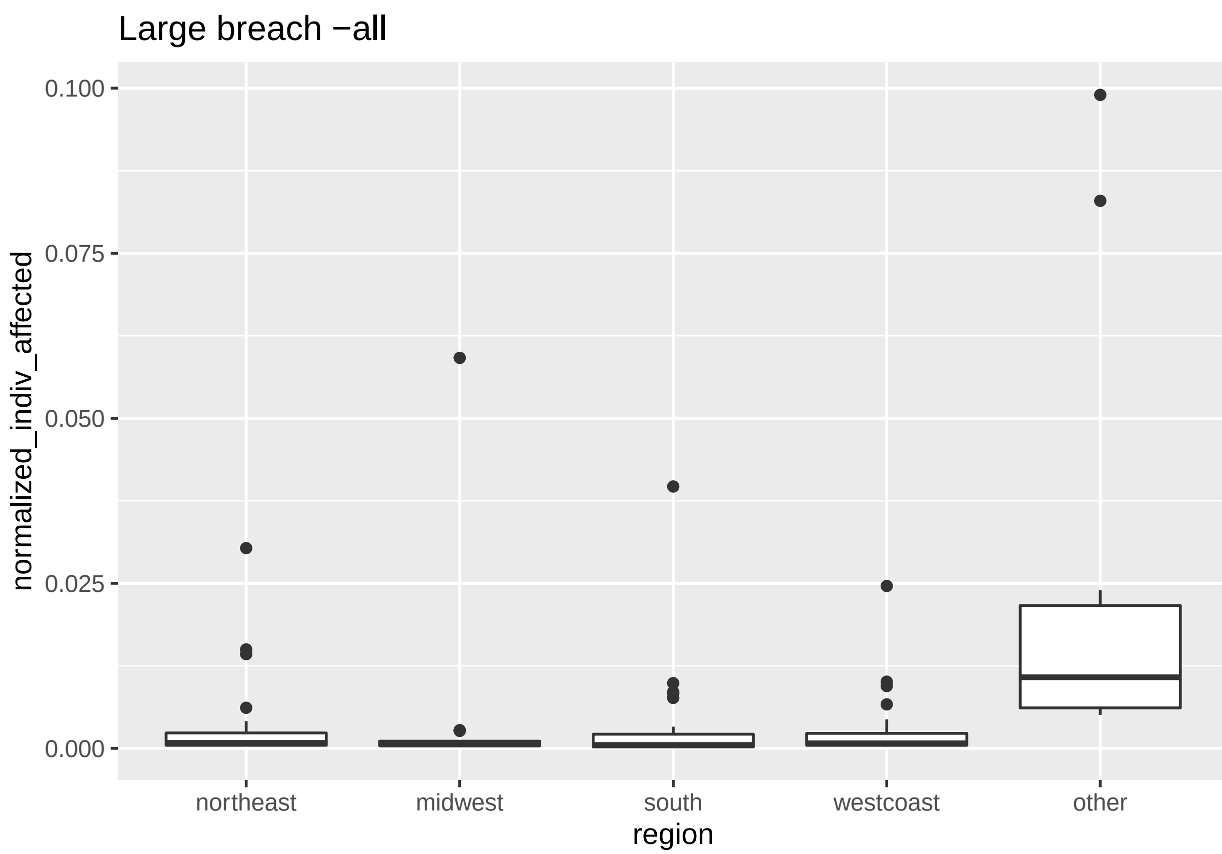
```
total_affected_state_sorted$State <- factor(total_affected_state_sorted$State, levels = total_affected_s
total_affected_state_sorted %>%
  ggplot(aes(State, normalized_state_indiv)) +
  geom_col() +
  coord_flip()
```


VA
PR
IL
UT
AL
NH
TN
MA
DC
SC
FL
NY
CA
NJ
WY
IN
OK
CT
AZ
GA
MT
TX
RI
CO
DE
KY
MN
WI
OR
WA
MD
NC
NM
OH
MO
MI
ND
PA
WV
IA
AK
AR
LA
KS
SD
NE
ID
NV
MS
ME
VT
HI
State
0.0
0.2
0.4
0.6
normalized_state_indiv

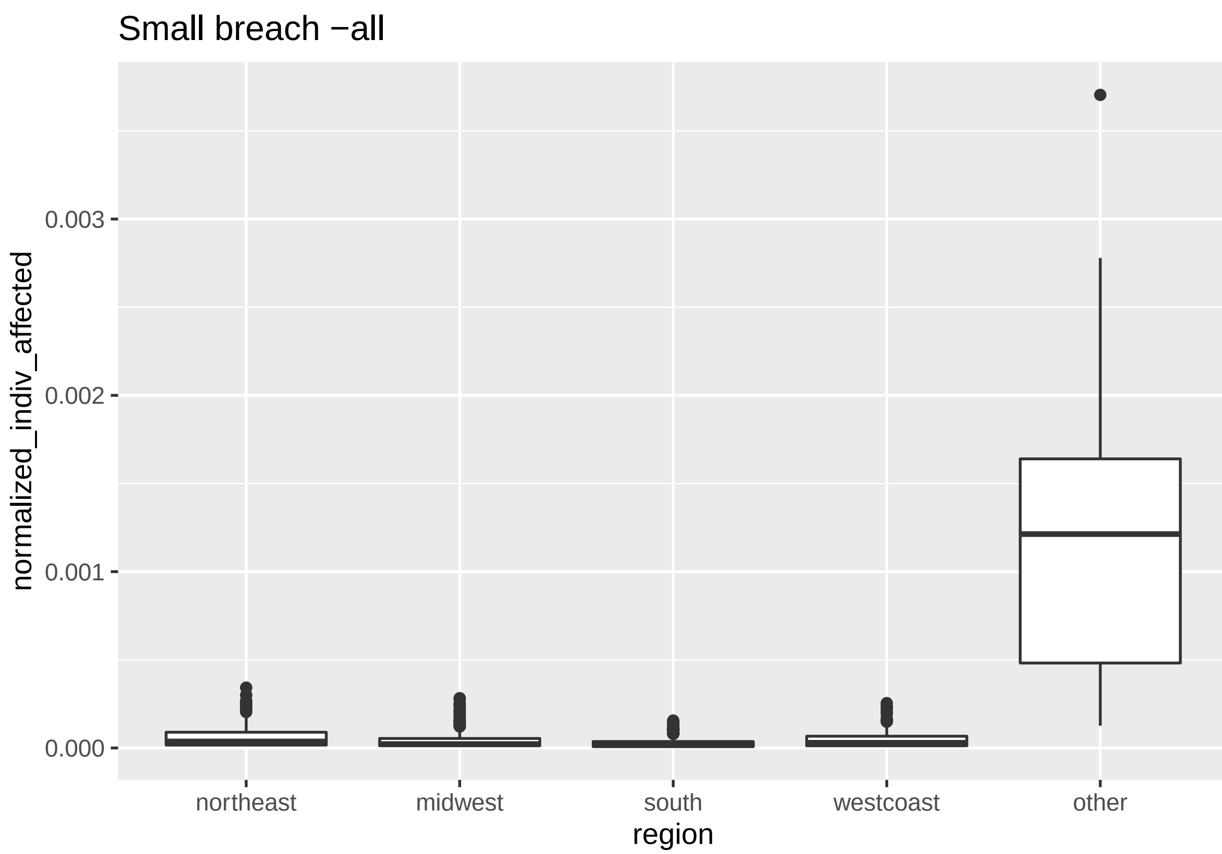
By normalizing the individuals affected by the state population a better comparison of the breaches affects in a state. Virginia, Puerto Rico, and Illinois are the states with the top percentage of individuals affected in their state. Therefore in these states getting affected by a breach is more likely.

```
northeast_total_population <- 56059240
midwest_total_population <- 68126781
west_total_population <- 77257329
south_total_population <- 123542189
other_total_population <- 3375000 + 1424000

normal_function <- function(x) {
  if(is.na(x)){
    return(NA)
  }
  else if(x == "northeast"){
    return(northeast_total_population)
  }
  else if(x == "midwest"){
    return(midwest_total_population)
  }
  else if(x == "westcoast"){
    return(west_total_population)
  }
  else if(x == "south"){
    return(south_total_population)
  }
  else{
    return(other_total_population)
  }
}


breaches$normalized_indiv_affected <- sapply(breaches$region, normal_function)


breaches <- breaches %>%
  mutate(normalized_indiv_affected = Individuals_Affected / normalized_indiv_affected)
```

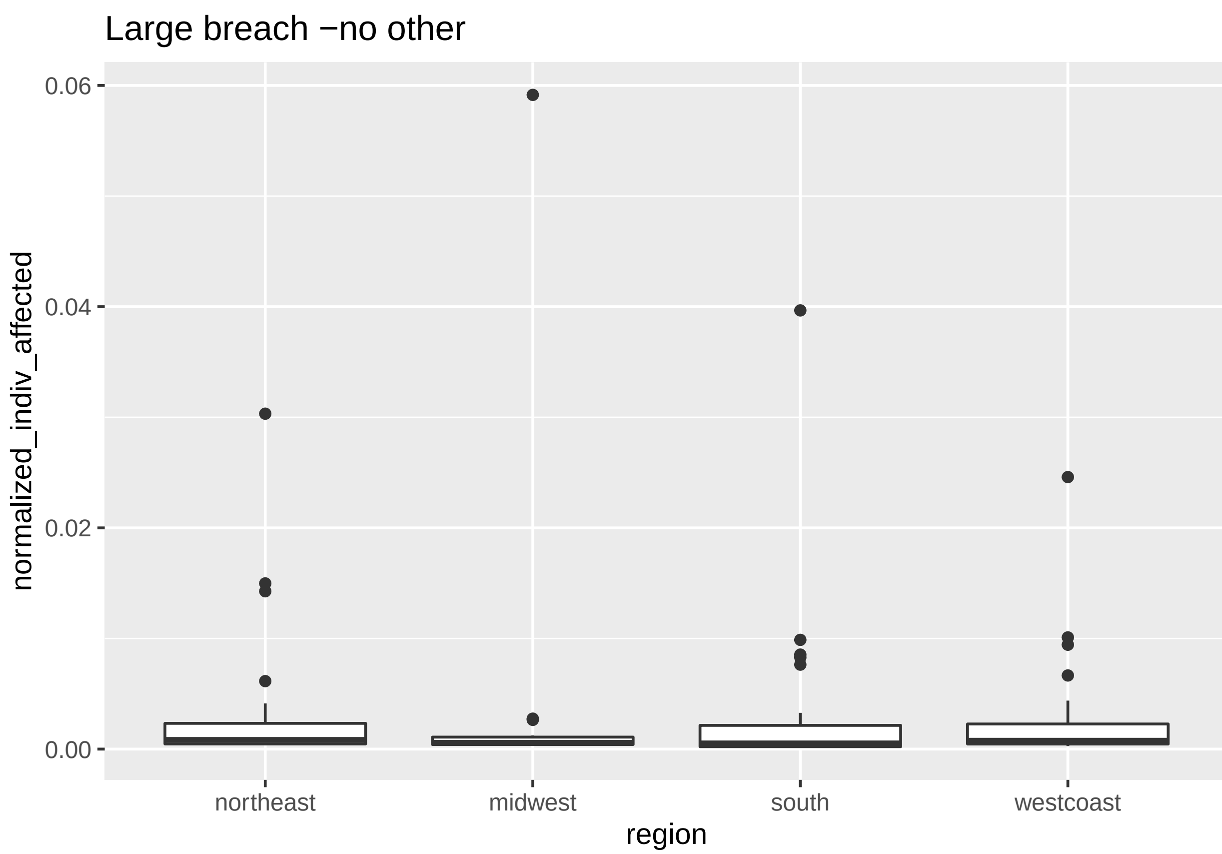
```
breaches %>%
  filter(large_affected == TRUE) %>%
  ggplot(aes(region, normalized_indiv_affected)) +
  geom_boxplot() +
  labs(title = "Large breach -all")
```



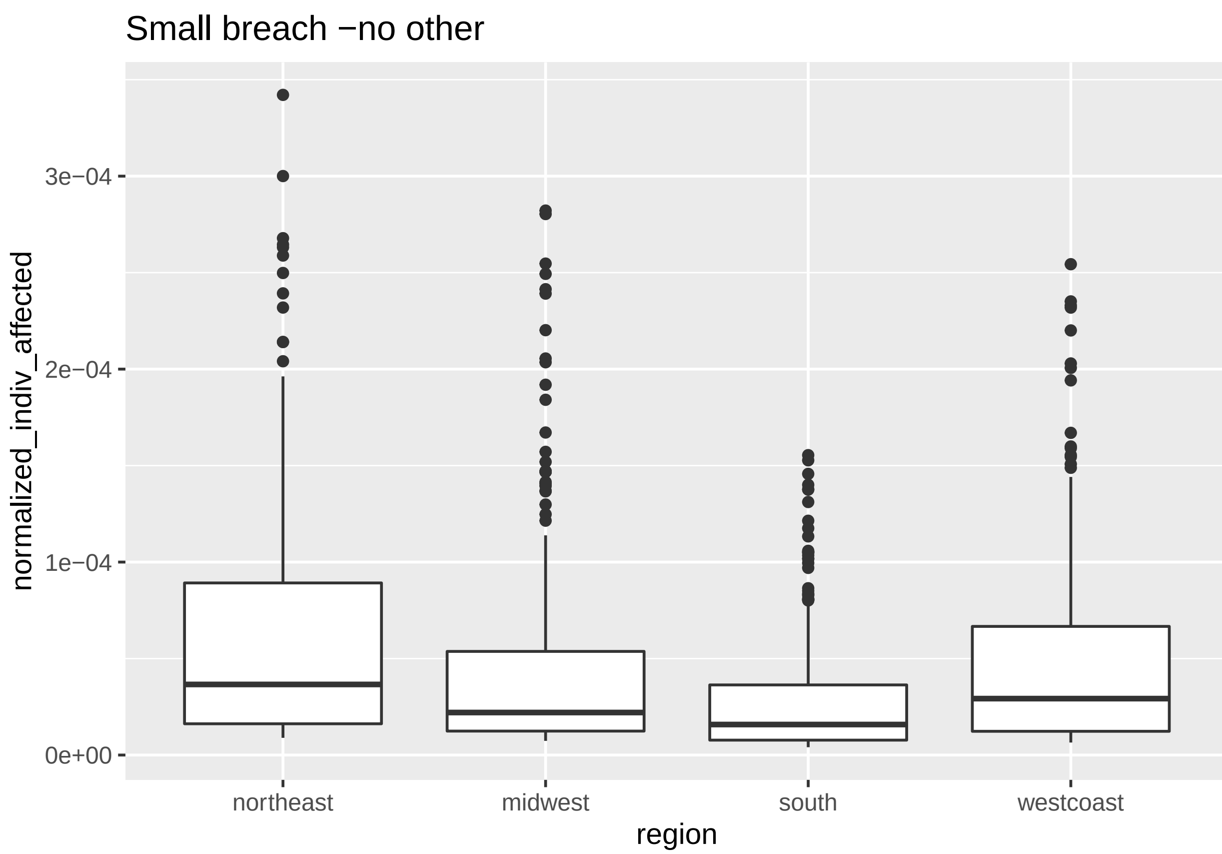

```
breaches %>%
  filter(large_affected == FALSE) %>%
  ggplot(aes(region, normalized_indiv_affected)) +
  geom_boxplot()+
  labs(title = "Small breach -all")
```




```
breaches %>%
  filter(large_affected == TRUE, region != "other") %>%
  ggplot(aes(region, normalized_indiv_affected)) +
  geom_boxplot()+
  labs(title = "Large breach -no other")
```




```
breaches %>%
  filter(large_affected == FALSE, region != "other") %>%
  ggplot(aes(region, normalized_indiv_affected)) +
  geom_boxplot()+
  labs(title = "Small breach -no other")
```




By normalizing the individuals by state population and then grouping into regions we can compare the affect of each breach by region. The other region, which includes PR and HI, has a much higher percentage of their population being affected by breaches, for both large and small breaches. To observe the distribution of the other regions of the US better I removed the other region, but overall there were not any strong trends in either the small breach or the large breach. In small breaches, the northeast has a slightly higher distribution of percent of individuals affected by breaches, but nothing significant engough to make a claim about.

**- Discuss how the observed patterns support/reject your hypotheses or answer your questions.** The state of the breach does affect the number of individuals affected by the breach. The states with the most individuals affected have a large city associated with them, Virginia(Virginia Beach, Arlington), CA (Los Angles, San Francisco), IL (Chicago), FL(Miami and Tampa), NY(New York City), TN(Nashville), TX(Houston, San Antonio, Dallas, Austin), AL(Birmingham), MA(Boston), NJ(Newark), UT(Salt Lake City). PR which is a territory breaks this trend. Since most of the breaches were in the medical field, states with large cities have larger populations and therefore have more opportunity to affect more individuals. Also looking at the distribution of normalized values for individuals affected, the states VA, PR, and IL stand out, as well as the other region, which includes PR and HI, for higher percentages of their population being affected by breaches. From this analysis we can answer the question that if you are in a certain state if you are more likely to be affected by a breach, by saying if you are in VA, IL, HI, and especially PR you are more likely to be effected by a breach.

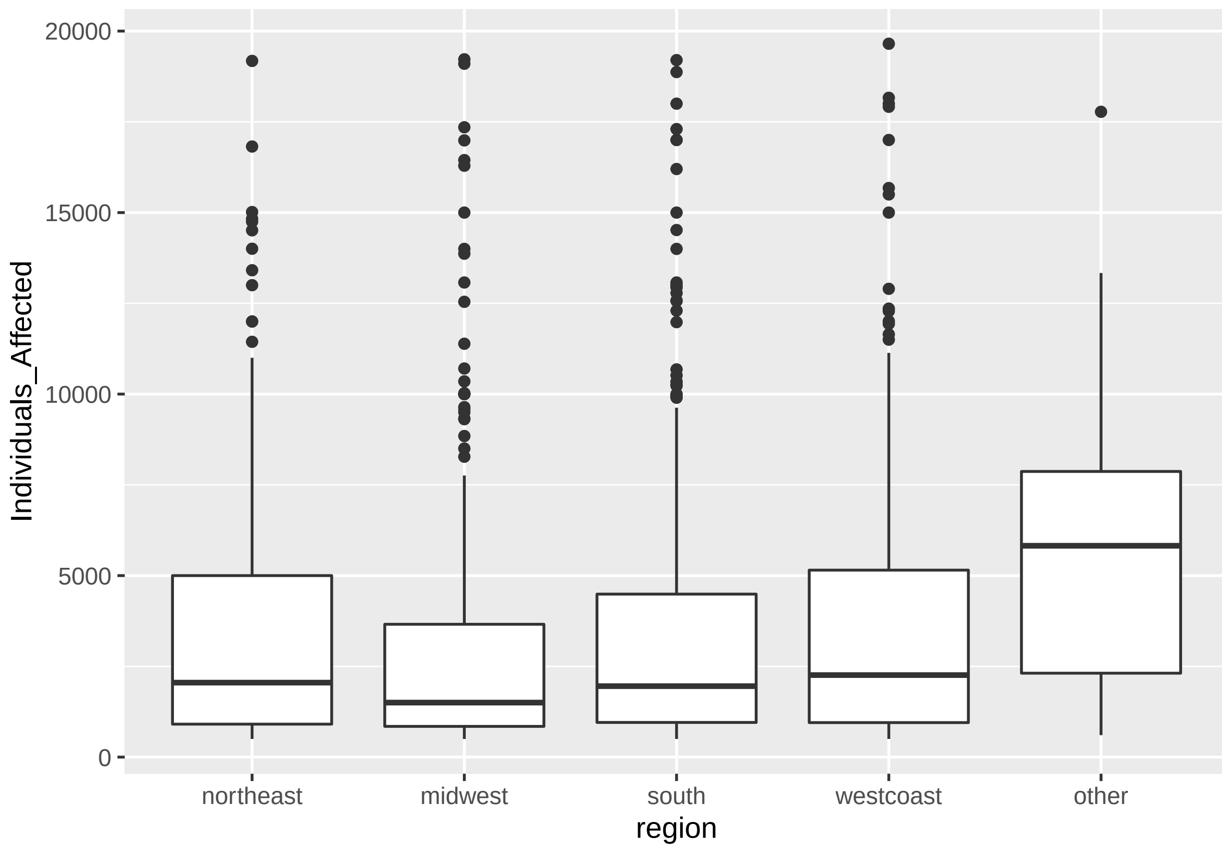
## Model Building

Load modelr

```
library(modelr)
options(na.action = na.warn)
```

Look at distribution of small breaches by region.

```
breaches %>%
  filter(large_affected == FALSE) %>%
  ggplot(aes(region, Individuals_Affected)) +
  geom_boxplot()
```



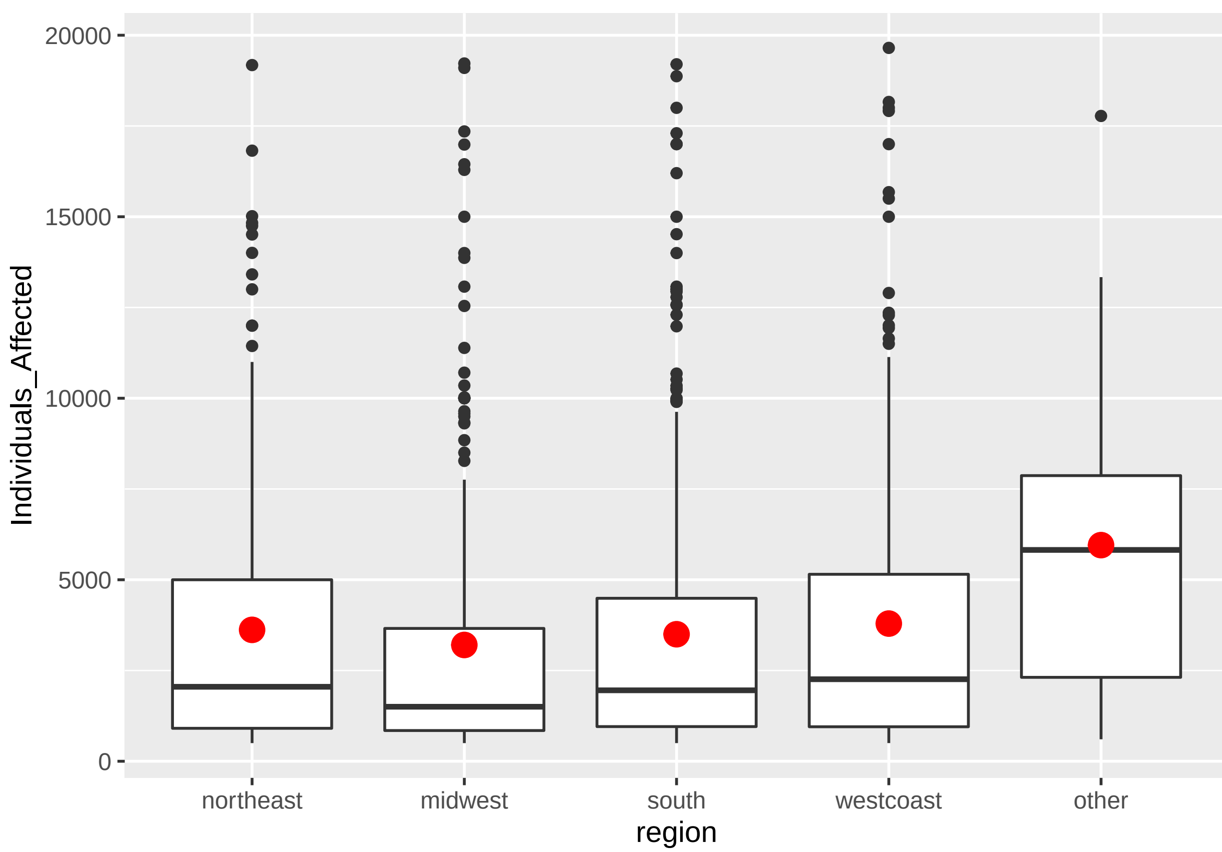

Fit the model and display its predictions overlaid on the orginial data

```
small_breaches <- breaches %>%
  filter(large_affected == FALSE)

mod <- lm(Individuals_Affected ~ region, data = small_breaches)

summary(mod)
```

```
##
## Call:
## lm(formula = Individuals_Affected ~ region, data = small_breaches)
##
## Residuals:
##    Min      1Q Median      3Q     Max
```

```
##  -5349  -2588  -1544   1004  16018
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3619.9      292.2  12.389  < 2e-16 ***
## regionmidwest     -416.0      393.2  -1.058  0.29036
## regionsouth       -120.9      365.0  -0.331  0.74062
## regionwestcoast    173.1      401.8   0.431  0.66665
## regionother       2333.8      885.4   2.636  0.00853 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3920 on 942 degrees of freedom
## Multiple R-squared:  0.01139,    Adjusted R-squared:  0.007197
## F-statistic: 2.714 on 4 and 942 DF,  p-value: 0.02882
```

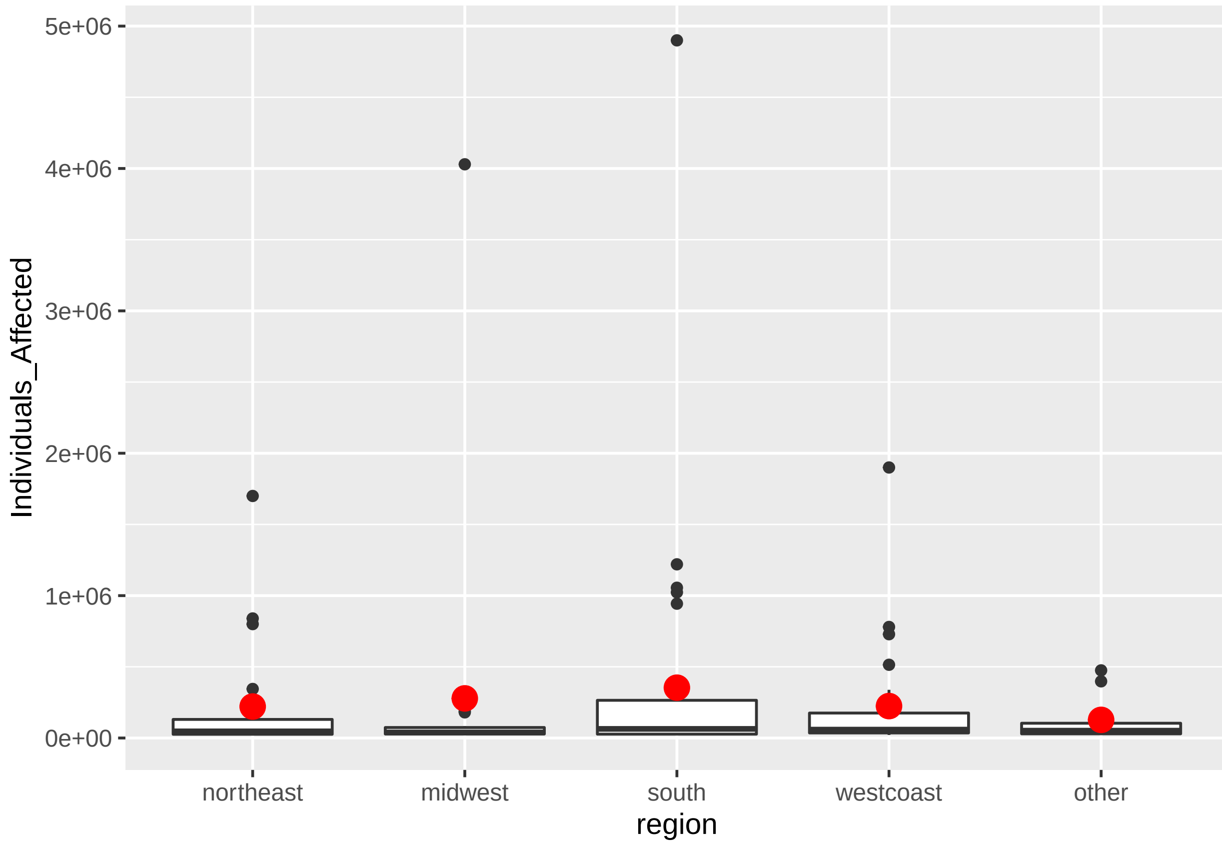
```r
grid <- small_breaches %>%
  data_grid(region) %>%
  add_predictions(mod, "Individuals_Affected")

ggplot(small_breaches, aes(region, Individuals_Affected)) +
  geom_boxplot() +
  geom_point(data = grid, colour = "red", size = 4)+
  labs("Model for small data breaches")
```



```r
large_breaches <- breaches %>%
  filter(large_affected == TRUE)

mod <- lm(Individuals_Affected ~ region, data = large_breaches)

summary(mod)
```

```
##
## Call:
## lm(formula = Individuals_Affected ~ region, data = large_breaches)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -332691 -248168 -189683  -79784 4546565
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        220862     148313   1.489    0.139
## regionmidwest       57336     218311   0.263    0.793
## regionsouth        132573     188634   0.703    0.484
## regionwestcoast      3416     201181   0.017    0.986
## regionother        -93484     261132  -0.358    0.721
##
## Residual standard error: 679700 on 103 degrees of freedom
## Multiple R-squared:  0.01124,    Adjusted R-squared:  -0.02716
## F-statistic: 0.2927 on 4 and 103 DF,  p-value: 0.8822
```

```r
grid <- large_breaches %>%
  data_grid(region) %>%
  add_predictions(mod, "Individuals_Affected")

ggplot(large_breaches, aes(region, Individuals_Affected)) +
  geom_boxplot() +
  geom_point(data = grid, colour = "red", size = 4) +
  labs("Model for large data breaches")
```

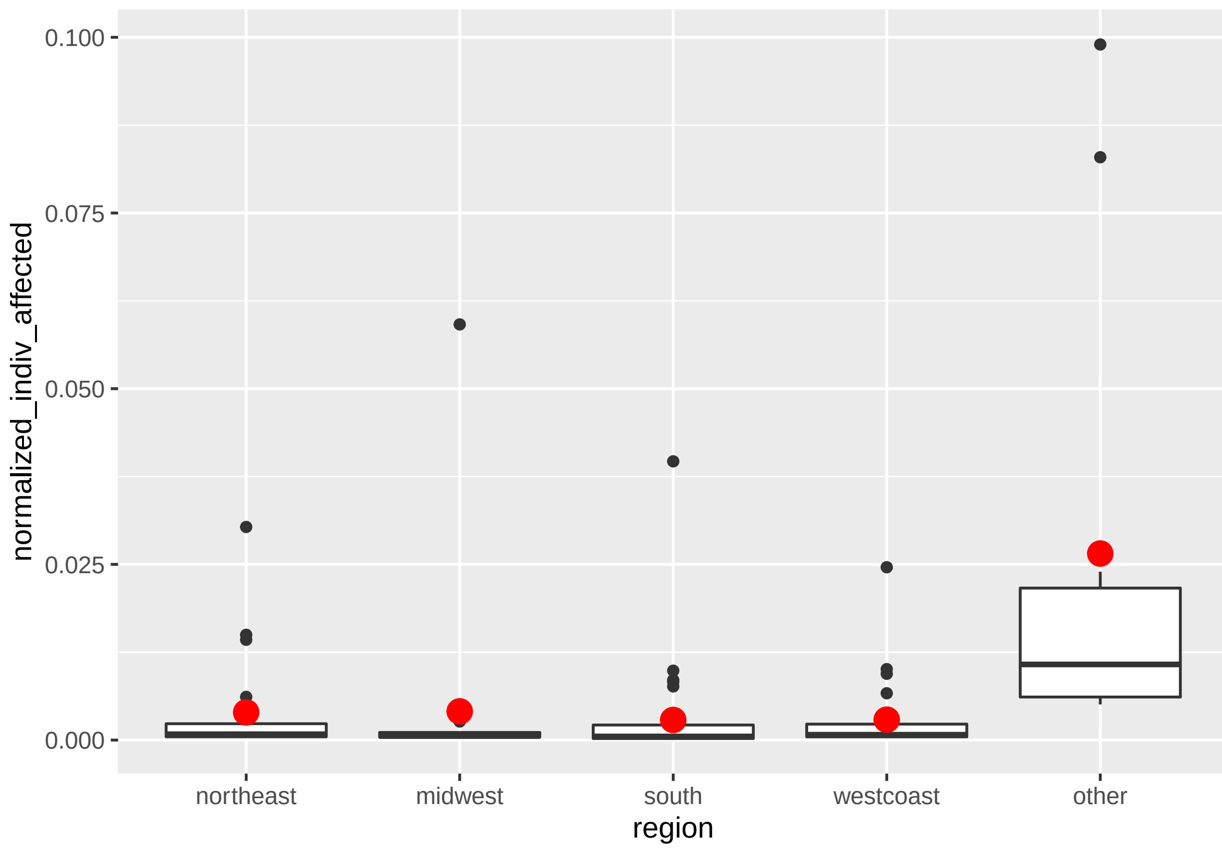



```r
large_breaches <- breaches %>%
  filter(large_affected == TRUE)

mod <- lm(normalized_indiv_affected ~ region, data = large_breaches)

summary(mod)
```

```
##
## Call:
## lm(formula = normalized_indiv_affected ~ region, data = large_breaches)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.021466 -0.003319 -0.002471 -0.001429  0.072436
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.0039398  0.0028369   1.389    0.168
## regionmidwest    0.0001437  0.0041758   0.034    0.973
## regionsouth     -0.0010790  0.0036081  -0.299    0.766
## regionwestcoast -0.0010368  0.0038481  -0.269    0.788
## regionother      0.0226028  0.0049948   4.525 1.62e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.013 on 103 degrees of freedom
## Multiple R-squared:  0.2204, Adjusted R-squared:  0.1901
## F-statistic: 7.279 on 4 and 103 DF,  p-value: 3.34e-05
```

```
grid <- large_breaches %>%
  data_grid(region) %>%
  add_predictions(mod, "normalized_indiv_affected")

ggplot(large_breaches, aes(region, normalized_indiv_affected)) +
  geom_boxplot() +
  geom_point(data = grid, colour = "red", size = 4)+
  labs("Model for large data breaches with normalized indiv affected")
```

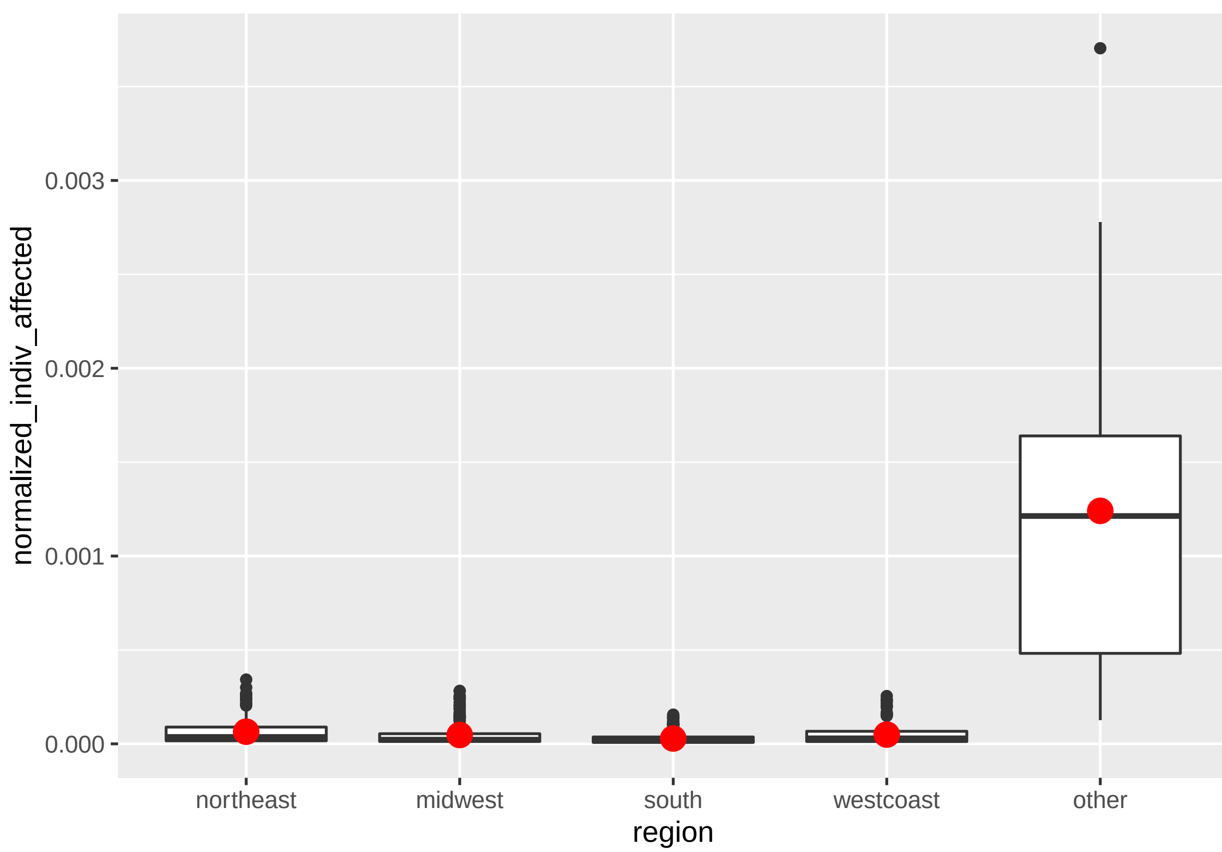



```
small_breaches <- breaches %>%
  filter(large_affected == FALSE)

mod <- lm(normalized_indiv_affected ~ region, data = small_breaches)

summary(mod)
```

```
##
## Call:
## lm(formula = normalized_indiv_affected ~ region, data = small_breaches)
##
## Residuals:
```

```
##          Min          1Q      Median          3Q         Max
## -1.115e-03 -3.232e-05 -1.769e-05  1.168e-05  2.463e-03
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      6.457e-05  1.089e-05   5.930 4.24e-09 ***
## regionmidwest   -1.754e-05  1.465e-05  -1.197  0.23147
## regionsouth     -3.625e-05  1.360e-05  -2.665  0.00783 **
## regionwestcoast -1.548e-05  1.497e-05  -1.034  0.30159
## regionother      1.176e-03  3.299e-05  35.644  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0001461 on 942 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6043
## F-statistic: 362.1 on 4 and 942 DF,  p-value: < 2.2e-16
```

```r
grid <- small_breaches %>%
  data_grid(region) %>%
  add_predictions(mod, "normalized_indiv_affected")

ggplot(small_breaches, aes(region, normalized_indiv_affected)) +
  geom_boxplot() +
  geom_point(data = grid, colour = "red", size = 4)+
  labs("Model for small data breaches with normalized indiv affected")
```



Overall the models do not do a good job of predicting the number of individuals affected by a breach, or the percentage of individuals affected out of a state population. This is seen by all of the models only having a significant p value (represented by the 2 or 3 stars) for the other region. The model that has the strongest significance is the normalized individuals affected and small data breaches by region, which has a 2 p values that are significant, for the southern region and other region. Since most of the p values are not statistically significant the models can not be accurately used to predict the individuals affected by region.