ITC Pt 1

Fayre-Ella Ooi

2025-04-09

```r
# data set
CircuitOutage = read.csv(file = "/Users/fayreooi/Desktop/circuitWRegions.csv")
LookUp = read.csv(file = "/Users/fayreooi/Downloads/LookUpUpdate.csv")

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

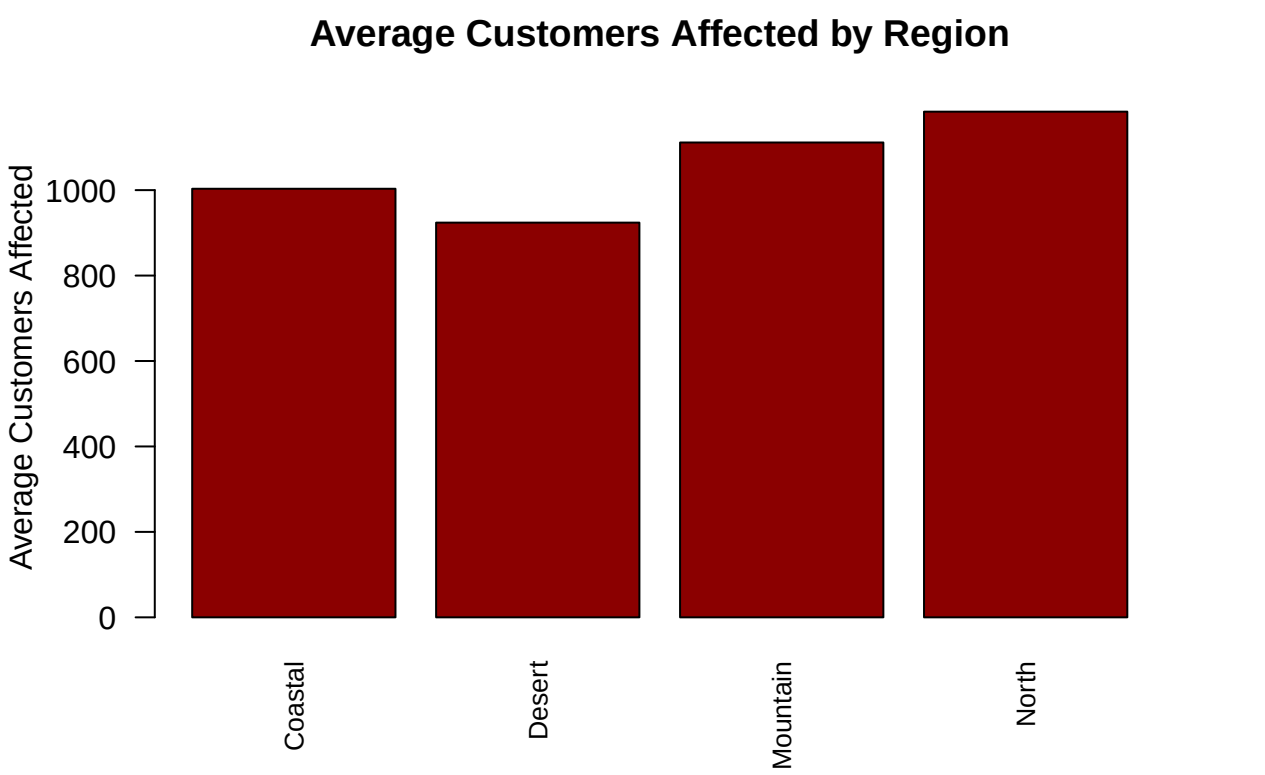
```r
df = left_join(CircuitOutage, LookUp, by = "Circuit.Name")

# clean

# make values numerical
df$Outage.Duration..min. = as.numeric(gsub(",", "", df$Outage.Duration..min.))
df$Customers.Affected = as.numeric(gsub(",", "", df$Customers.Affected))
df$Customer.Count = as.numeric(gsub(",", "", df$Customer.Count))

# drop unnecessary columns
df = df[, !(names(df) %in% c("Region.x", "X", "Circuit.Number.y"))]

# add new column
df$Percentage.Customers.Affected = df$Customers.Affected / df$Customer.Count
```

```r
# find the mean of customers affected per region
tapply(df$Customers.Affected, df$Region.y, mean)
```

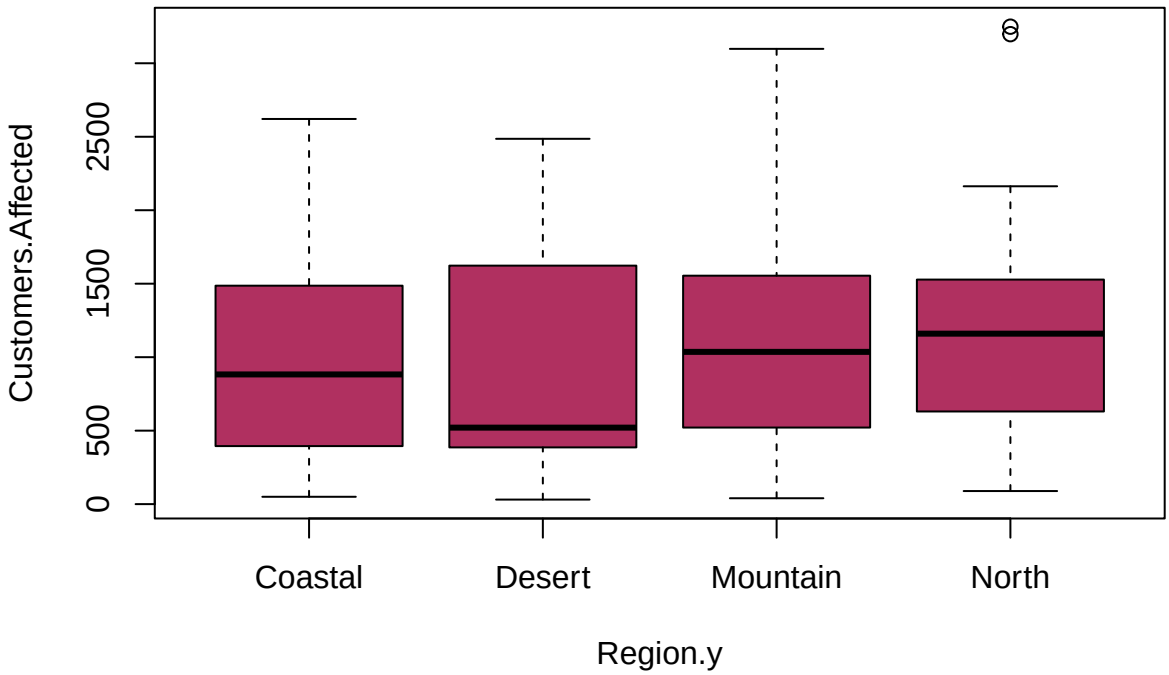
```
##   Coastal    Desert  Mountain     North
## 1003.1071  923.8333 1111.4348 1183.6129
```

```r
# put the means into a dataframe
df2 = matrix(c(1003.1071, 923.8333, 1111.4348, 1183.6129),
             ncol = 4, nrow = 1,
             dimnames = list( rownames = c("Average Customers Affected"),
                              colnames = c("Coastal", "Desert", "Mountain", "North")))

# barplot of average customers affected per region
```

```r
barplot(df2,
        main = "Average Customers Affected by Region",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```




```r
boxplot(Customers.Affected ~ Region.y, data = df,
        col = "maroon")
```




```
# perform chisquare test
X2.CA.region = chisq.test(df2)
X2.CA.region$expected
```

```
## [1] 1055.497 1055.497 1055.497 1055.497
```

```
X2.CA.region$residuals
```

```
## [1] -1.612573 -4.052636  1.721776  3.943433
```

```
X2.CA.region$p.value
```

```
## [1] 3.537852e-08
```

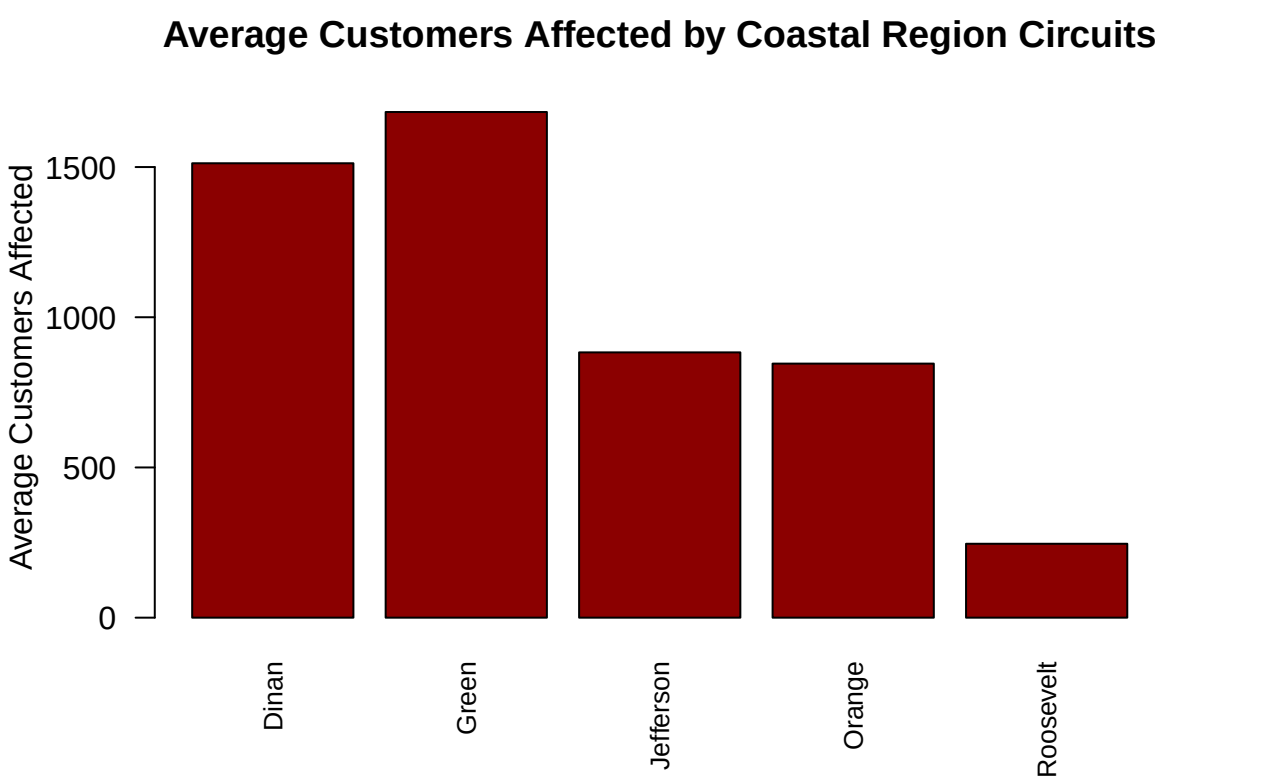
```
# p-value is less than alpha = 0.05
# H0: amount of customers affected is evenly distributed among the regions
# H1: not H0
# reject the null:
# so the region of outages matter
```

```
# focus: COASTAL
df3 = subset(df, Region.y %in% c("Coastal"))

# find the mean of amt of customers affected per circuit name
tapply(df3$Customers.Affected, df3$Circuit.Name, mean)
```

```
##     Dinan     Green Jefferson    Orange Roosevelt
## 1512.5714 1683.1667  883.0000  845.6667  246.0000
```

```
# put means into dataframe
df.coastal = matrix(c(1512.5714, 1683.1667, 883, 845.6667, 246),
                    ncol = 5, nrow = 1,
                    dimnames = list(rownames = c("Average Customers Affected"),
                                    colnames = c("Dinan", "Green", "Jefferson",
                                                 "Orange", "Roosevelt")))
```

```r
# barplot of amt of customers affected per coastal circuit names
barplot(df.coastal,
        main = "Average Customers Affected by Coastal Region Circuits",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```




```r
# run chisquare test
X2.coastal.CA = chisq.test(df.coastal)
X2.coastal.CA$expected
```

```
## [1] 1034.081 1034.081 1034.081 1034.081 1034.081
```

```r
X2.coastal.CA$residuals
```

```
## [1]  14.879762  20.184816  -4.698210  -5.859175 -24.507193
```

```r
X2.coastal.CA$p.value
```

```
## [1] 3.910938e-277
```

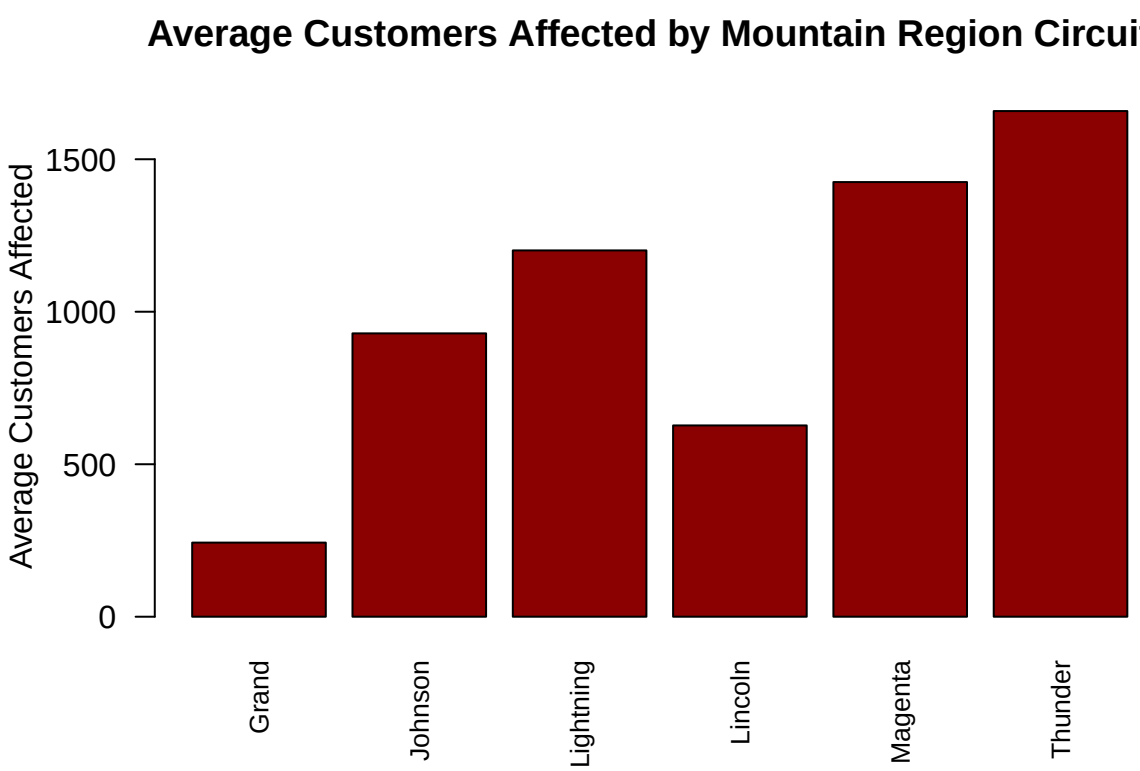
```r
# p-val < alpha
# reject the null
# so circuit name with region of coastal matters
```

```r
# focus: MOUNTAIN
df4 = subset(df, Region.y %in% c("Mountain"))

# find mean of amount customers affected
tapply(df4$Customers.Affected, df4$Circuit.Name, mean)
```

```
##     Grand   Johnson Lightning   Lincoln   Magenta   Thunder
```

```
##  243.0000  929.6667 1201.0000  627.1667 1425.1000 1658.0000
```

```
# put means into dataframe
df.mountain = matrix(c(243, 929, 1201, 627.1667, 1425.1, 1658),
                     nrow = 1, ncol = 6,
                     dimnames = list(rownames = c("Average Customers Affected"),
                                     colnames = c("Grand", "Johnson", "Lightning",
                                                  "Lincoln", "Magenta", "Thunder")))

#barplot of avg customers affected per mountain circuit names
barplot(df.mountain,
        main = "Average Customers Affected by Mountain Region Circuits",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```




```
# perform chi square test
X2.mountain.CA = chisq.test(df.mountain)
X2.mountain.CA$expected
```

```
## [1] 1013.878 1013.878 1013.878 1013.878 1013.878 1013.878
```

```
X2.mountain.CA$residuals
```

```
## [1] -24.209885  -2.665638   5.876687 -12.144896  12.914683  20.229049
```

```
X2.mountain.CA$p.value
```

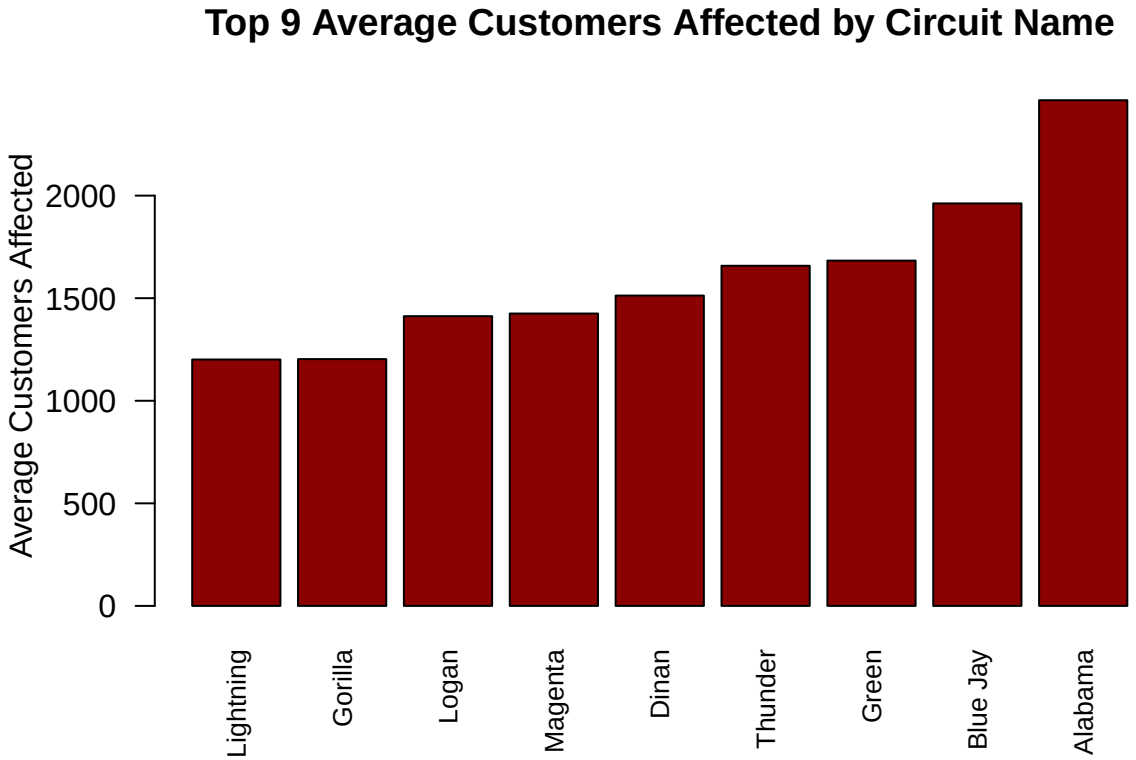
```
## [1] 5.002212e-290
```

```
# p-val < alpha
# circuit name within region of mountains matter
```

```r
# dataset with just significant circuit names based on customers affected
df7 = subset(df, Circuit.Name %in% c("Green", "Dinan", "Alabama", "Logan",
                                     "Magenta", "Thunder", "Lightning", "Gorilla",
                                     "Blue Jay"))

# average customers affected
sort(tapply(df7$Customers.Affected, df7$Circuit.Name, mean))
```

```
## Lightning   Gorilla     Logan   Magenta     Dinan   Thunder     Green  Blue Jay
##  1201.000  1203.111  1412.200  1425.100  1512.571  1658.000  1683.167  1962.000
##   Alabama
##  2465.000
```

```r
# dataset of just avg customers affects vs circuit name
df8 = matrix(c(1201, 1203.111, 1412.2, 1425.1, 1512.571, 1658, 1683.167, 1962,
               2465),
             nrow = 1, ncol = 9,
             dimnames = list(rownames = c("Average Customers Affected"),
                             colnames = c("Lightning", "Gorilla", "Logan", "Magenta",
                                          "Dinan", "Thunder", "Green", "Blue Jay",
                                          "Alabama")))

# plot data
barplot(df8,
        main = "Top 9 Average Customers Affected by Circuit Name",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```



```r
X2.CA.top9 = chisq.test(df8)
X2.CA.top9$expected
```

```
## [1] 1613.572 1613.572 1613.572 1613.572 1613.572 1613.572 1613.572 1613.572
## [9] 1613.572
```

```r
X2.CA.top9$residuals
```

```
## [1] -10.270833 -10.218281  -5.013086  -4.691945  -2.514386   1.106016   1.732540
## [8]   8.673986  21.195989
```

```r
X2.CA.top9$p.value
```

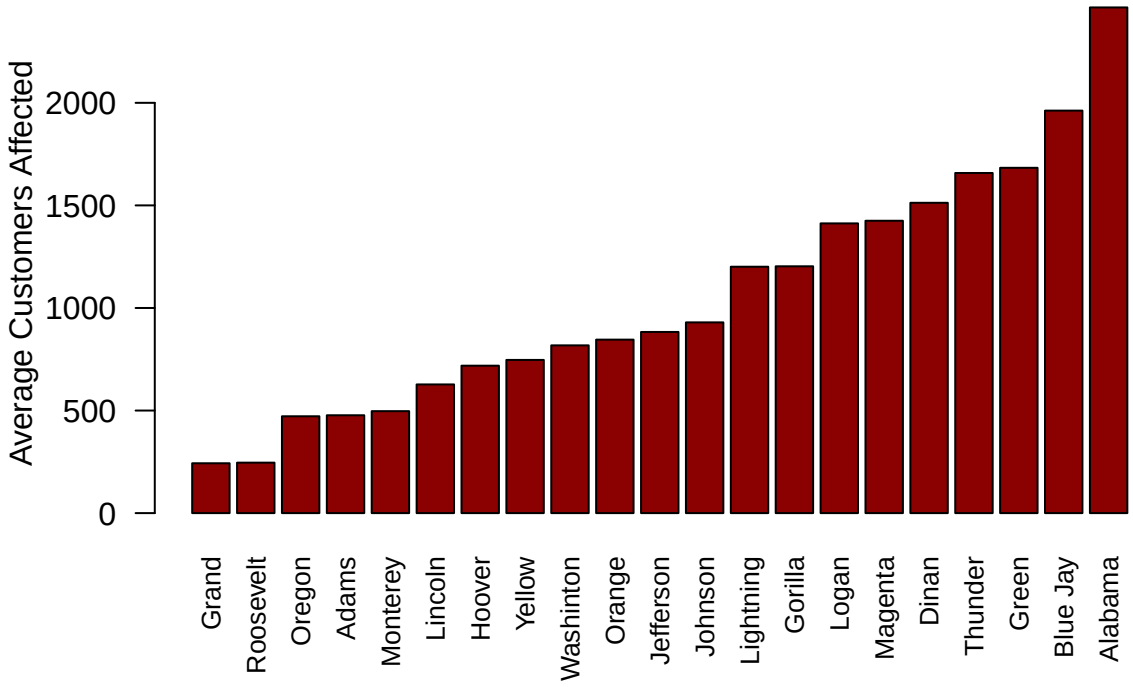
```
## [1] 1.03576e-165
```

```r
# ALABAMA AND BLUE JAY ARE THE MOST PROBLEMATIC CIRCUITS IN TERMS OF CUSTOMERS AFFECTED.
```

```r
# average customers affected from ALL circuit names
sort(tapply(df$Customers.Affected, df$Circuit.Name, mean))
```

```
##      Grand  Roosevelt     Oregon      Adams   Monterey    Lincoln     Hoover
##   243.0000   246.0000   472.7500   477.0000   497.0000   627.1667   718.5714
##     Yellow Washington     Orange  Jefferson    Johnson  Lightning    Gorilla
##   746.7143   817.6667   845.6667   883.0000   929.6667  1201.0000  1203.1111
##      Logan    Magenta      Dinan    Thunder      Green   Blue Jay    Alabama
##  1412.2000  1425.1000  1512.5714  1658.0000  1683.1667  1962.0000  2465.0000
```

```r
# put into dataset
df9 = matrix(c(243, 246, 472, 477, 497, 627.1667, 718.5714, 746.7143, 817.6667,
               845.6667, 883, 929.6667, 1201, 1203.1111, 1412.2, 1425.1, 1512.5714,
               1658, 1683.1667, 1962, 2465),
             ncol = 21, nrow = 1,
             dimnames = list(rownames = c("Average Customers Affected"),
                             colnames = c("Grand", "Roosevelt", "Oregon", "Adams",
                                          "Monterey", "Lincoln", "Hoover",
                                          "Yellow", "Washinton", "Orange",
                                          "Jefferson", "Johnson", "Lightning",
                                          "Gorilla", "Logan", "Magenta", "Dinan",
                                          "Thunder", "Green", "Blue Jay", "Alabama")))

# barplot
barplot(df9,
        main = "Average Customers Affected by Circuit Name",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```




```
df10 = subset(df, Region.y %in% c("North"))

tapply(df10$Customers.Affected, df10$Circuit.Name, mean)
```

```
##    Alabama      Hoover      Logan Washington     Yellow
##  2465.0000   718.5714  1412.2000   817.6667   746.7143
```

```
df.north = matrix(c(2465.0000, 718.5714, 1412.2000, 817.6667, 746.7143), nrow=1, ncol=5, byrow=T,
                  dimnames=list(c("Customers Affected"),c("Alabama","Hoover","Logan","Washington"," Yel
df.north
```

```
##                     Alabama   Hoover  Logan Washington   Yellow
## Customers Affected    2465 718.5714 1412.2   817.6667 746.7143
```

```
chi.north = chisq.test(df.north)
chi.north$exp
```

```
## [1] 1232.03 1232.03 1232.03 1232.03 1232.03
```

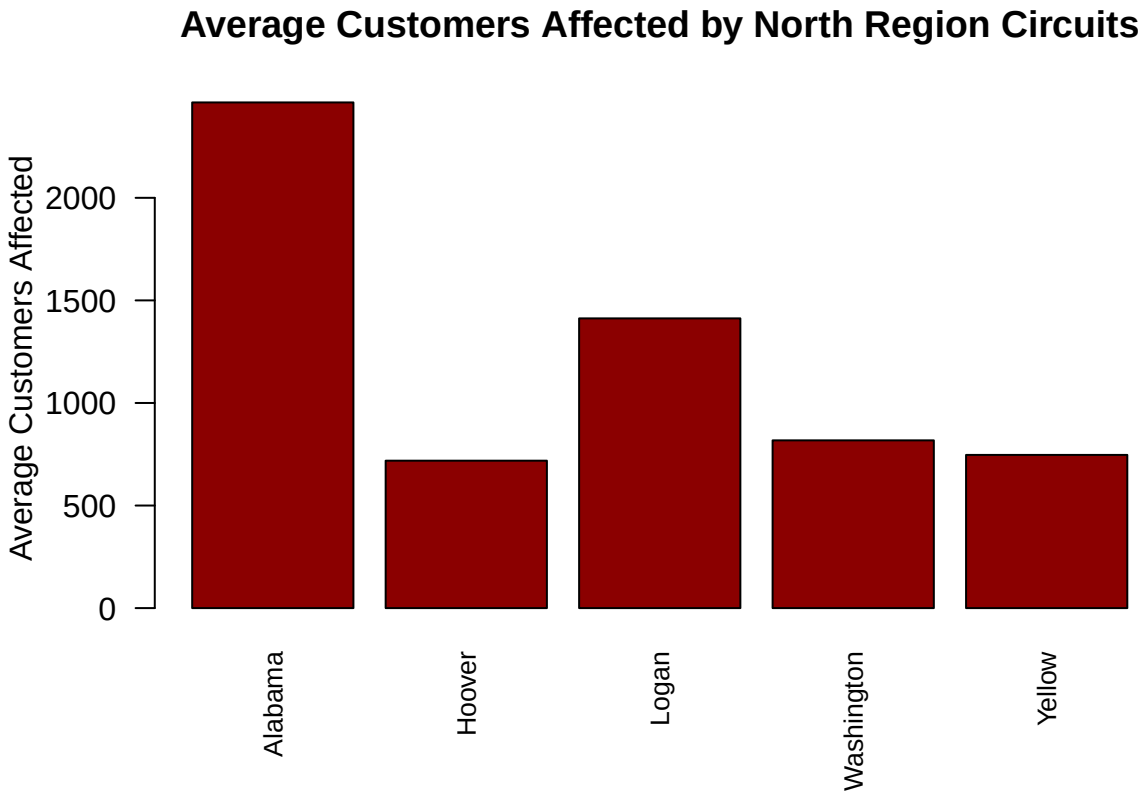
```
chi.north$residuals
```

```
## [1]  35.127045 -14.628342   5.132992 -11.805138 -13.826557
```

```
chi.north$p.value
```

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

```
barplot(df.north,
        main = "Average Customers Affected by North Region Circuits",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```




```
df11 = subset(df, Region.y %in% c("Desert"))

tapply(df11$Customers.Affected, df11$Circuit.Name, mean)
```

```
##    Adams Blue Jay  Gorilla Monterey   Oregon
##  477.000 1962.000 1203.111  497.000  472.750
```

```
df.desert = matrix(c(477.000, 1962.000, 1203.111 , 497.000, 472.750 ), nrow=1, ncol=5, byrow=T,
                   dimnames=list(c("Customers Affected"),c("Adams","Blue Jay","Gorilla","Monterey","Oreg
df.desert
```

```
##                    Adams Blue Jay  Gorilla Monterey  Oregon
## Customers Affected   477     1962 1203.111      497  472.75
```

```
chi.desert = chisq.test(df.desert)
chi.desert$exp
```

```
## [1] 922.3722 922.3722 922.3722 922.3722 922.3722
```

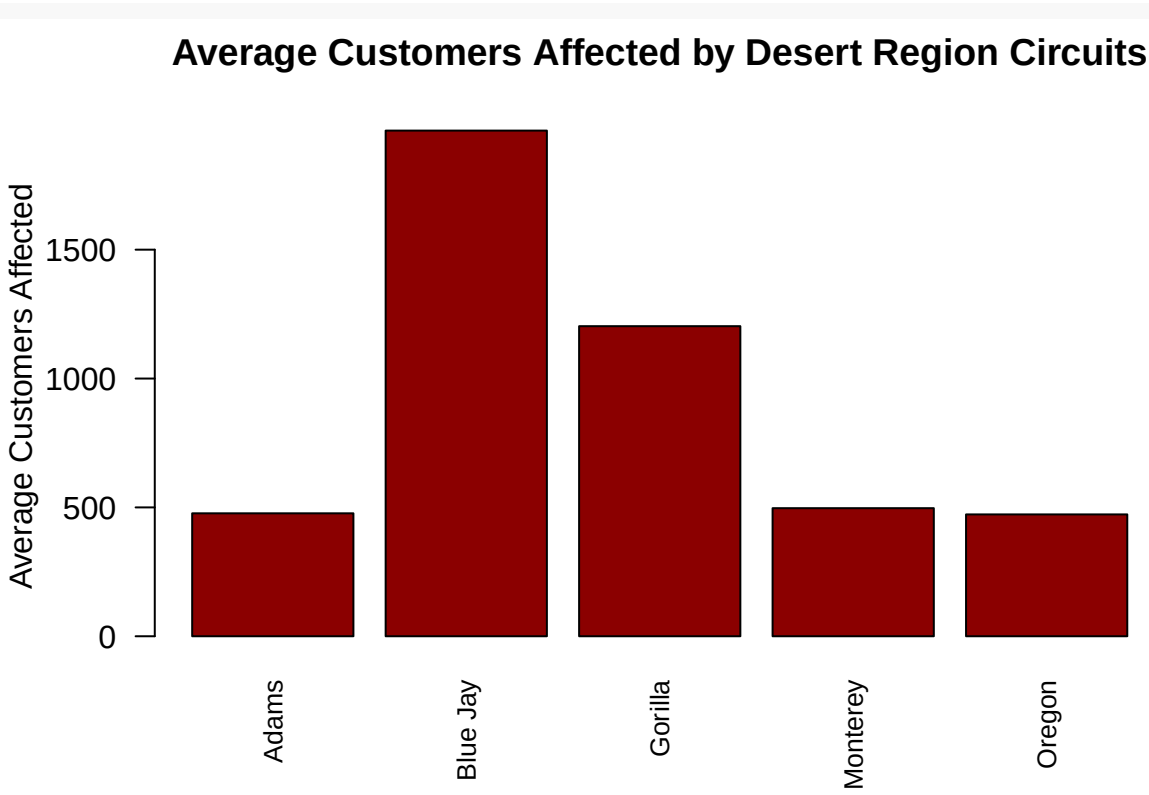
```
chi.desert$residuals
```

```
## [1] -14.664593  34.231409   9.243774 -14.006061 -14.804531
```

```
chi.desert$p.value
```

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

```
barplot(df.desert,
        main = "Average Customers Affected by Desert Region Circuits",
        ylab = "Average Customers Affected",
        las = 2,
        col = "darkred",
        cex.names = 0.8)
```




```
## ANALYSIS OF THE TOP 9 CIRCUITS
# look at the average outage duration of each

df.top9 = subset(df, Circuit.Name %in% c("Lightning","Gorilla","Logan","Magenta","Dinan","Thunder","Gre

tapply(df.top9$Outage.Duration..min., df.top9$Circuit.Name, mean)
```

```
##   Alabama  Blue Jay     Dinan   Gorilla     Green Lightning     Logan   Magenta
##  518.2500  411.0000  713.4286  590.6667  919.0000   72.0000  681.1000  759.7000
##   Thunder
## 1341.5000
```

```
df.top9.OD =(matrix(c(518.2500, 411.0000, 713.4286, 590.6667, 919.0000, 72.0000, 681.1000, 759.7000, 13
                dimnames=list(c("Outage Duration"),c("Alabama","Blue Jay","Dinan","Gorilla"," Green",
df.top9.OD
```

```
##                 Alabama Blue Jay    Dinan  Gorilla  Green Lightning Logan
## Outage Duration  518.25      411 713.4286 590.6667    919        72 681.1
##                 Magenta Thunder
## Outage Duration   759.7  1341.5
```

```
chi.topOD = chisq.test(df.top9.OD)
chi.topOD$exp
```

```
## [1] 667.405 667.405 667.405 667.405 667.405 667.405 667.405 667.405 667.405
```

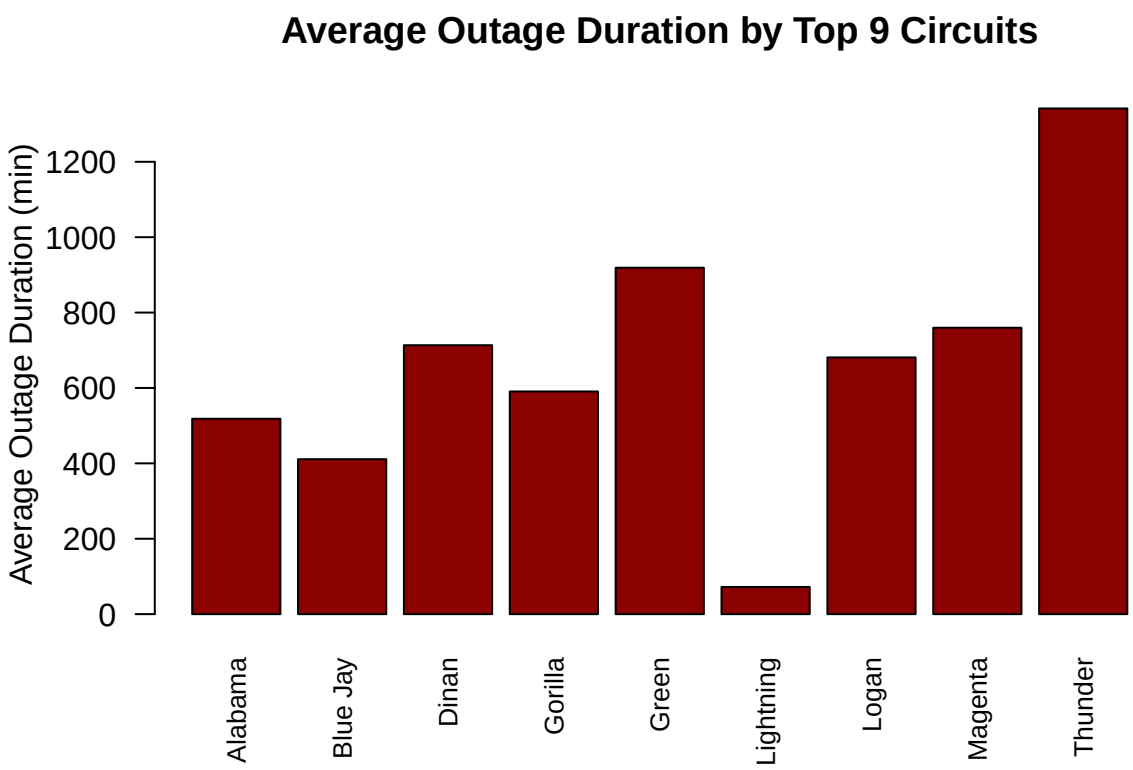
```
chi.topOD$residuals
```

```
## [1]  -5.7735532  -9.9250295   1.7814988  -2.9704184   9.7388395 -23.0471784
## [7]   0.5301103   3.5725908  26.0931401
```

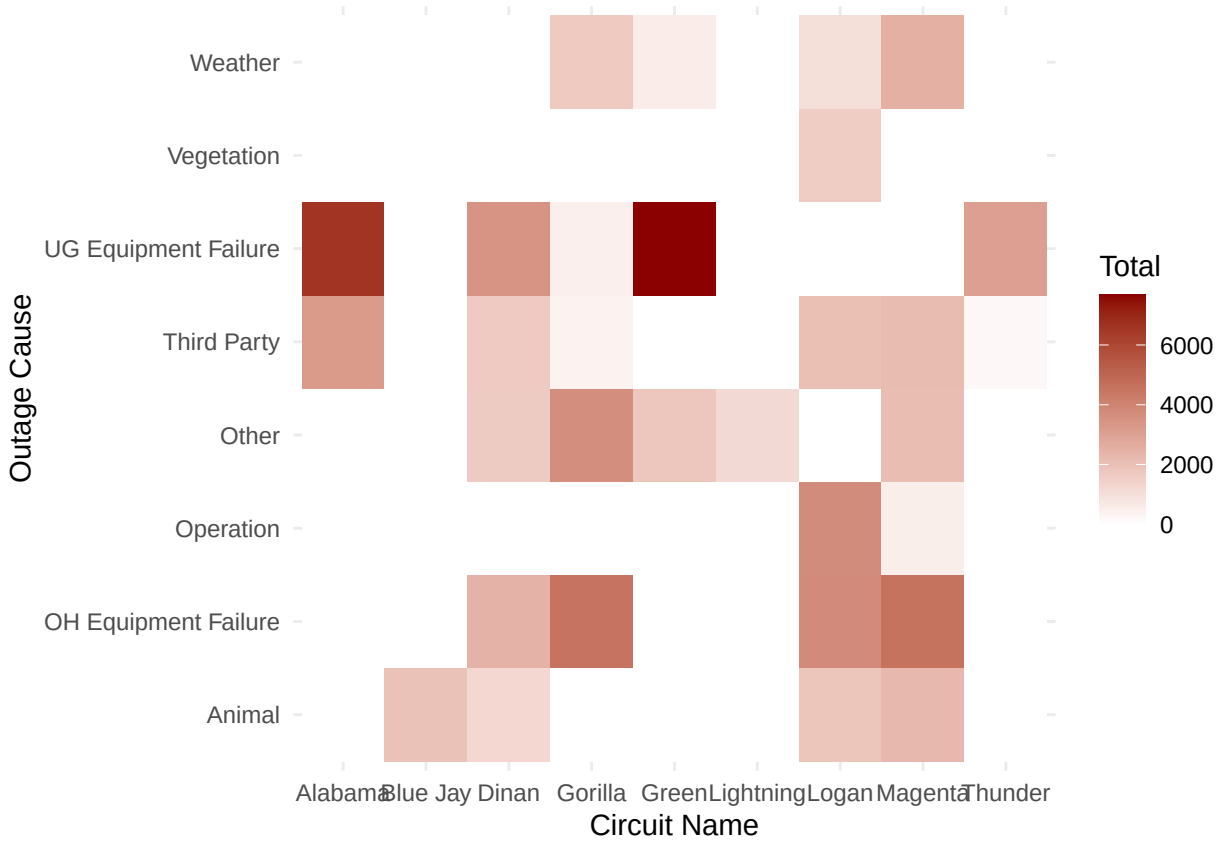
```
chi.topOD$p.value
```

```
## [1] 9.277654e-311
```

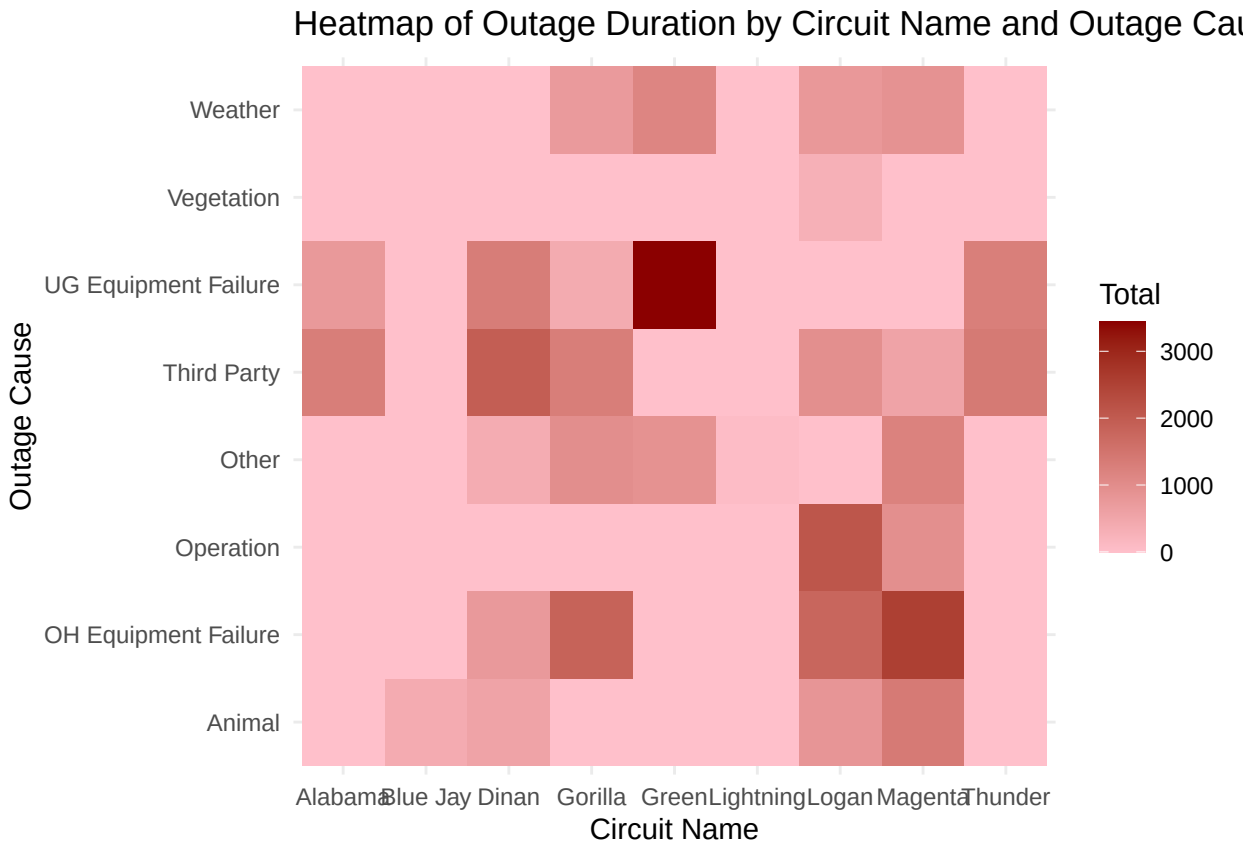
```
barplot(df.top9.OD,
        main = "Average Outage Duration by Top 9 Circuits",
        ylab = "Average Outage Duration (min)",
        col = "darkred",
        las = 2,
        cex.names =0.8)
```




```
table.top9.1 = xtabs(Customers.Affected ~ Outage.Cause + Circuit.Name, data = df.top9)

library(ggplot2)
library(reshape2)

long_table = melt(table.top9.1, id.vars = "Outage.Cause", variable.name = "Circuit.Name", value.name =

ggplot(long_table, aes(x = Circuit.Name, y = Outage.Cause, fill = Total)) +
  geom_tile() +
  scale_fill_gradient2(low = "white", high = "darkred") +
  theme_minimal() +
  labs(x = "Circuit Name", y = "Outage Cause")
```




```
# circuit causes by top 9 circuits

library(dplyr)

table.top9 = xtabs(Outage.Duration..min. ~ Outage.Cause + Circuit.Name, data = df.top9)

library(ggplot2)
library(reshape2)

long_table = melt(table.top9, id.vars = "Outage.Cause", variable.name = "Circuit.Name", value.name = "T

ggplot(long_table, aes(x = Circuit.Name, y = Outage.Cause, fill = Total)) +
  geom_tile() +
  scale_fill_gradient2(low = "white", mid = "pink", high = "darkred") +
  theme_minimal() +
  labs(title = "Heatmap of Outage Duration by Circuit Name and Outage Cause",
       x = "Circuit Name", y = "Outage Cause")
```




```
X2.top9.1 = chisq.test(table.top9)
```

```
## Warning in chisq.test(table.top9): Chi-squared approximation may be incorrect
```

```
X2.top9.1$expected
```

```
##                       Circuit.Name
## Outage.Cause             Alabama  Blue Jay      Dinan    Gorilla      Green
##   Animal               186.89786 37.055003  450.24984  479.28076  497.13208
##   OH Equipment Failure 406.64018 80.621860  979.62426 1042.78785 1081.62758
##   Operation            179.88481 35.664571  433.35491  461.29650  478.47797
##   Other                209.57340 41.550732  504.87677  537.42990  557.44704
##   Third Party          438.31581 86.901976 1055.93302 1124.01680 1165.88199
##   UG Equipment Failure 423.17930 83.900961 1019.46813 1085.20076 1125.62020
##   Vegetation            18.40926  3.649883   44.34919   47.20871   48.96704
##   Weather              210.09938 41.655014  506.14389  538.77872  558.84610
##                       Circuit.Name
## Outage.Cause            Lightning      Logan    Magenta   Thunder
##   Animal                6.4913873  614.06721  684.93152 241.89434
##   OH Equipment Failure 14.1235375 1336.04742 1490.22937 526.29793
##   Operation             6.2478081  591.02529  659.23053 232.81763
##   Other                 7.2789603  688.56942  768.03141 271.24237
##   Third Party          15.2237039 1440.12010 1606.31220 567.29441
##   UG Equipment Failure 14.6979786 1390.38795 1550.84088 547.70384
##   Vegetation            0.6393956   60.48504   67.46511  23.82637
##   Weather               7.2972287  690.29757  769.95898 271.92312
```

```
# test independency between Circuit Name and Outage Cause
```

```r
#chisquare test to stimulate p val
X2.top9.2 = chisq.test(table.top9, simulate.p.value = TRUE, B = 10000)

#G2 test
library(DescTools)
GTest(table.top9)
```

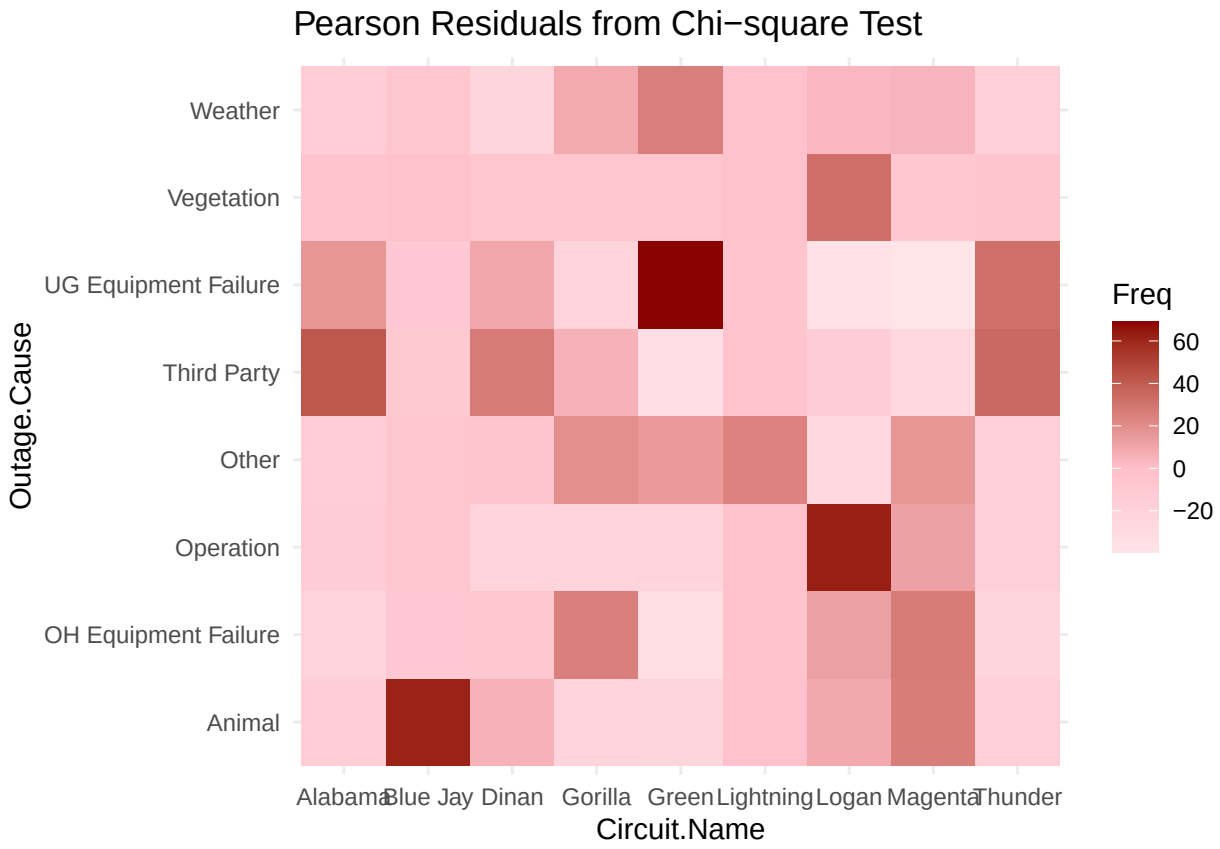
```
##
##  Log likelihood ratio (G-test) test of independence without correction
##
## data:  table.top9
## G = 40368, X-squared df = 56, p-value < 2.2e-16
```

```r
# H0: circuit name and outage cause are independent
# low pval < 2.2e^-16 so reject the null
# so there is a statistically significant relationship between circuit name and outage
# in terms of outage duration

# start partitioning based on residual
X2.top9.2$residuals
```

```
##                       Circuit.Name
## Outage.Cause               Alabama    Blue Jay       Dinan     Gorilla
##   Animal               -13.6710592  61.4305358   5.5492556 -21.8924819
##   OH Equipment Failure -20.1653213  -8.9789677  -7.0808848  25.1829077
##   Operation            -13.4121142  -5.9719822 -20.8171782 -21.4778140
##   Other                -14.4766501  -6.4459857  -5.3796022  19.4788927
##   Third Party           41.4923805  -9.3221229  27.4523345   5.6666813
##   UG Equipment Failure  16.6649812  -9.1597468   9.9135823 -20.3749895
##   Vegetation            -4.2906018  -1.9104667  -6.6595185  -6.8708592
##   Weather              -14.4948052  -6.4540696 -22.4976419   8.7982394
##                       Circuit.Name
## Outage.Cause                 Green   Lightning       Logan     Magenta
##   Animal               -22.2964589  -2.5478201   9.0770519  26.5585290
##   OH Equipment Failure -32.8881070  -3.7581295  12.6108777  27.3750030
##   Operation            -21.8741392  -2.4995616  62.3575330  12.1426908
##   Other                 15.0168660  23.9889263 -26.2406064  16.6334313
##   Third Party          -34.1450141  -3.9017565 -12.3355333 -25.9317034
##   UG Equipment Failure  68.8036261  -3.8337943 -37.2879062 -39.3807172
##   Vegetation            -6.9976456  -0.7996221  32.7257273  -8.2137149
##   Weather               25.7680122  -2.7013383   3.4522383   4.7945932
##                       Circuit.Name
## Outage.Cause                Thunder
##   Animal               -15.5529527
##   OH Equipment Failure -22.9411842
##   Operation            -15.2583625
##   Other                -16.4694374
##   Third Party           34.6253910
##   UG Equipment Failure  31.7606216
##   Vegetation            -4.8812258
##   Weather              -16.4900916
```

```r
resid_df = as.data.frame(as.table(X2.top9.2$residuals))

#heatmap for residuals
```

```r
ggplot(resid_df, aes(x = Circuit.Name, y = Outage.Cause, fill = Freq)) +
  geom_tile() +
  scale_fill_gradient2(low = "white", mid = "pink", high = "darkred", midpoint = 0) +
  theme_minimal() +
  labs(title = "Pearson Residuals from Chi-square Test")
```




```r
# most significance found in Green and UG Equipment Failure, Operation and Logan,
# Animal and Blue Jay
```