Project 3

Parham & Banafshe

2022-07-15

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

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

```
data <- read_csv("C:\\Users\\Parham\\Desktop\\projectData.csv")
```

```
## Rows: 1624 Columns: 16
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (13): Incident.Number, Reported.Date, Nearest.Populated.Centre, Province...
## dbl  (3): Latitude, Longitude, Year
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
data <- data%>%
  rename(SubstanceRelease = `Substance release`)
glimpse(data)
```

```
## Rows: 1,624
## Columns: 16
## $ Incident.Number                  <chr> "INC2007-097", "INC2008-001", "INC200~
## $ Reported.Date                    <chr> "01/02/2008", "01/02/2008", "01/23/20~
## $ Nearest.Populated.Centre         <chr> "Grande Prairie", "Cromer", "Cromer",~
## $ Province                         <chr> "Alberta", "Manitoba", "Manitoba", "B~
## $ Company                          <chr> "Alliance Pipeline Ltd.", "Enbridge P~
## $ Status                           <chr> "Closed", "Closed", "Closed", "Closed~
## $ Latitude                         <dbl> 54.84000, 49.73135, 49.73135, 58.0120~
## $ Longitude                        <dbl> -118.65000, -101.23557, -101.23557, -~
## $ Approximate.Volume.Released..m3. <chr> "Not Provided", "8.0000", "100.0000",~
## $ Substance                         <chr> "Natural Gas - Sweet", "Crude Oil - S~
## $ Release.Type                      <chr> "Gas", "Liquid", "Liquid", "Gas", "Mi~
## $ Significant                       <chr> "No", "No", "No", "No", "Yes", "No", ~
## $ Year                              <dbl> 2008, 2008, 2008, 2008, 2008, 2008, 2~
## $ What.Happened                     <chr> "Corrosion and Cracking", "Corrosion ~
## $ Why.It.Happened                   <chr> "Maintenance", "Maintenance", "Mainte~
## $ SubstanceRelease                  <chr> "Yes", "Yes", "Yes", "Yes", "Yes", "Y~
```

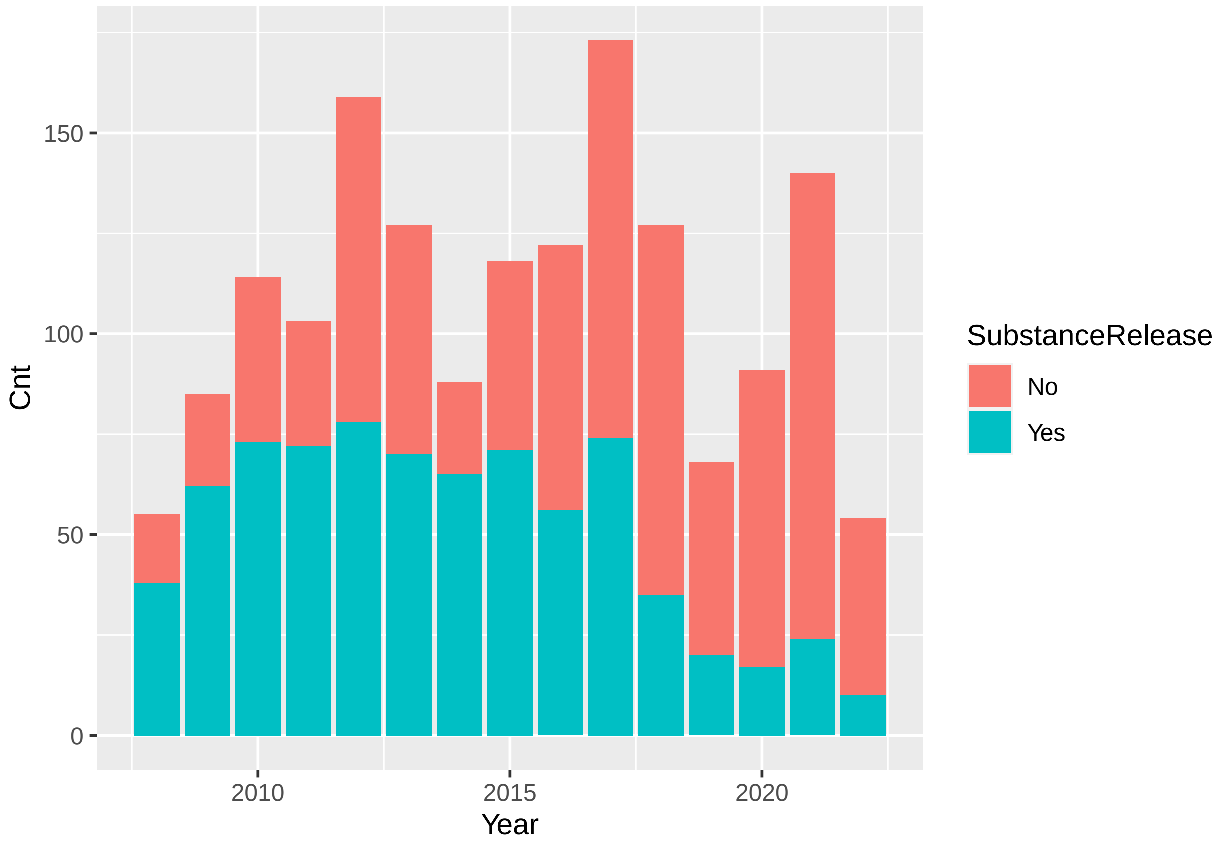
```
t<-data%>%
  group_by(Year,SubstanceRelease)%>%
  summarize(Cnt = n())
```

```
## ‘summarise()‘ has grouped output by ’Year’. You can override using the
## ‘.groups‘ argument.
```

```
t
```

```
## # A tibble: 30 x 3
## # Groups:   Year [15]
##     Year SubstanceRelease   Cnt
##    <dbl> <chr>            <int>
##  1  2008 No                  17
##  2  2008 Yes                 38
##  3  2009 No                  23
##  4  2009 Yes                 62
##  5  2010 No                  41
##  6  2010 Yes                 73
##  7  2011 No                  31
##  8  2011 Yes                 72
##  9  2012 No                  81
## 10  2012 Yes                 78
## # ... with 20 more rows
```

```
t%>%
  ggplot(aes(x=Year, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



```
t<-pivot_wider(
  t,
  names_from = SubstanceRelease,
  values_from = `Cnt`,
)
t
```

```
## # A tibble: 15 x 3
## # Groups:   Year [15]
##     Year    No   Yes
##    <dbl> <int> <int>
##  1  2008    17    38
##  2  2009    23    62
##  3  2010    41    73
##  4  2011    31    72
##  5  2012    81    78
##  6  2013    57    70
##  7  2014    23    65
##  8  2015    47    71
##  9  2016    66    56
## 10  2017    99    74
## 11  2018    92    35
## 12  2019    48    20
## 13  2020    74    17
## 14  2021   116    24
## 15  2022    44    10
```

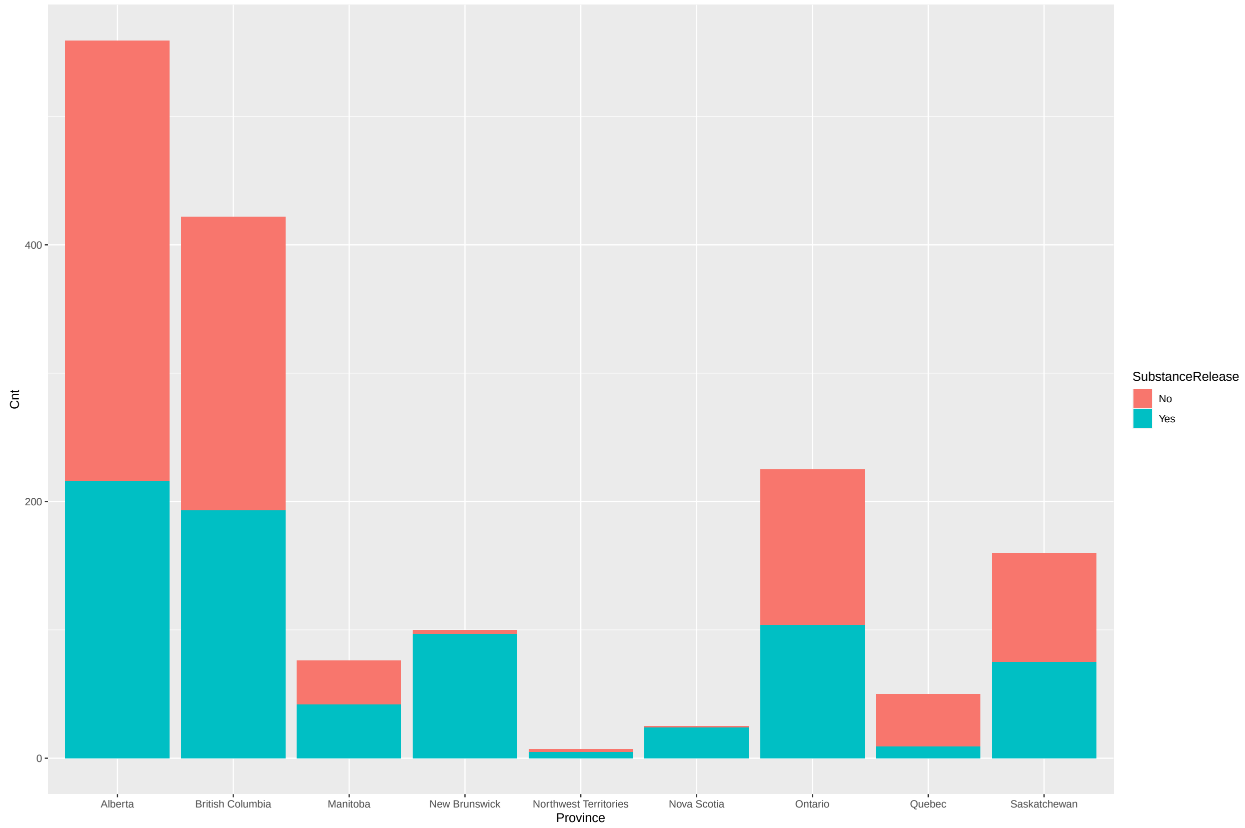
```
t2<-data%>%
  group_by(Province,SubstanceRelease)%>%
  summarize(Cnt = n())%>%
  arrange(desc(Cnt))
```

```
## ‘summarise()‘ has grouped output by ’Province’. You can override using the
## ‘.groups‘ argument.
```

```
t2
```

```
## # A tibble: 18 x 3
## # Groups:   Province [9]
##    Province              SubstanceRelease   Cnt
##    <chr>                 <chr>            <int>
##  1 Alberta               No                 343
##  2 British Columbia      No                 229
##  3 Alberta               Yes                216
##  4 British Columbia      Yes                193
##  5 Ontario               No                 121
##  6 Ontario               Yes                104
##  7 New Brunswick         Yes                 97
##  8 Saskatchewan          No                  85
##  9 Saskatchewan          Yes                 75
## 10 Manitoba              Yes                 42
## 11 Quebec                No                  41
## 12 Manitoba              No                  34
## 13 Nova Scotia           Yes                 24
## 14 Quebec                Yes                  9
## 15 Northwest Territories Yes                  5
## 16 New Brunswick         No                   3
## 17 Northwest Territories No                   2
## 18 Nova Scotia           No                   1
```

```
t2%>%
  ggplot(aes(x=Province, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



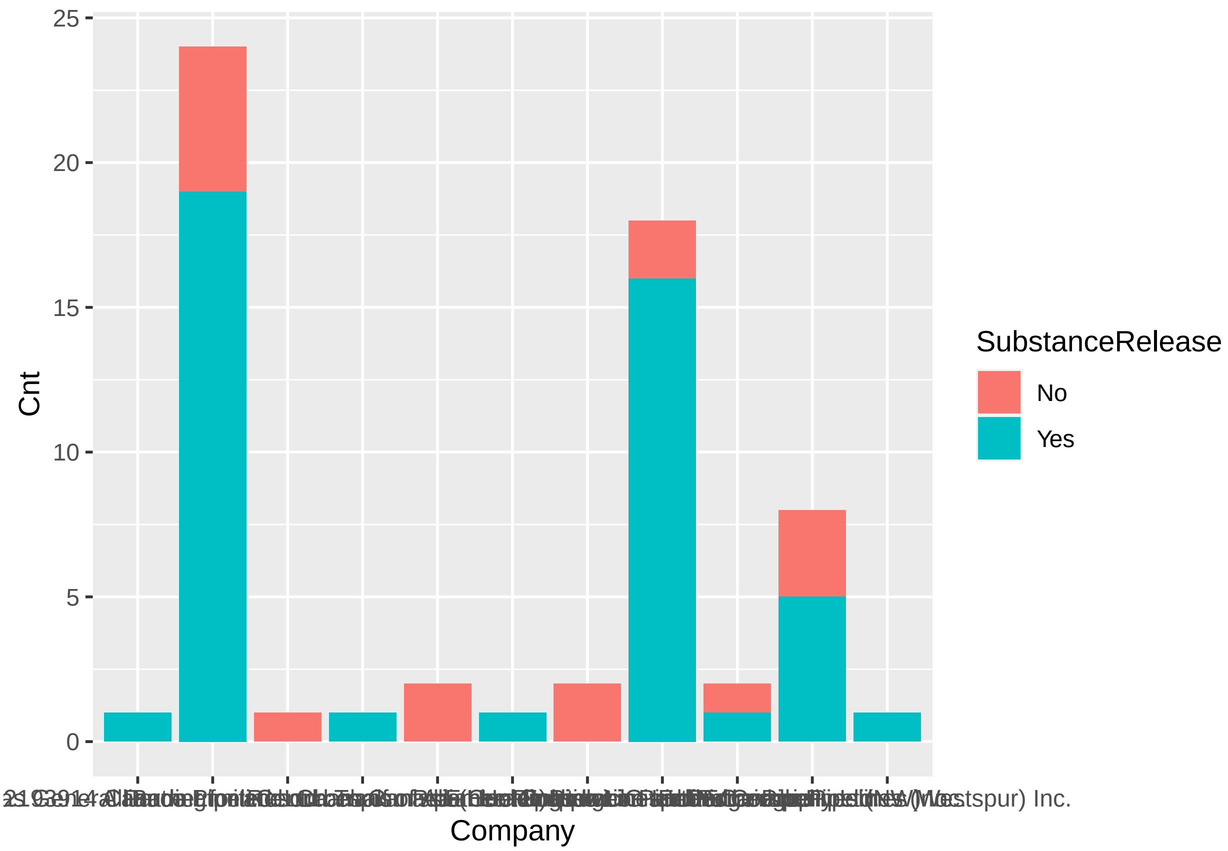
```
t3<-data%>%
  group_by(Company,SubstanceRelease)%>%
  summarize(Cnt = n())
```

```
## `summarise()` has grouped output by 'Company'. You can override using the
## `.groups` argument.
```

```
t3
```

```
## # A tibble: 74 x 3
## # Groups:   Company [50]
##    Company                                                SubstanceRelease   Cnt
##    <chr>                                                  <chr>            <int>
##  1 2193914 Canada Limited                                 Yes                  1
##  2 Alliance Pipeline Ltd.                                 No                   5
##  3 Alliance Pipeline Ltd.                                 Yes                 19
##  4 Alliance Pipeline Ltd., as General Partner for and on~ No                   1
##  5 Burlington Resources Canada (Hunter) Ltd.              Yes                  1
##  6 Centra Transmission Holdings Inc.                      No                   2
##  7 Champion Pipe Line Corporation Limited                 Yes                  1
##  8 Cochin Pipe Lines Ltd.                                 No                   2
##  9 Emera Brunswick Pipeline Company Ltd.                  No                   2
## 10 Emera Brunswick Pipeline Company Ltd.                  Yes                 16
## # ... with 64 more rows
```

```
t3[1:15,]%>%
  ggplot(aes(x=Company, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



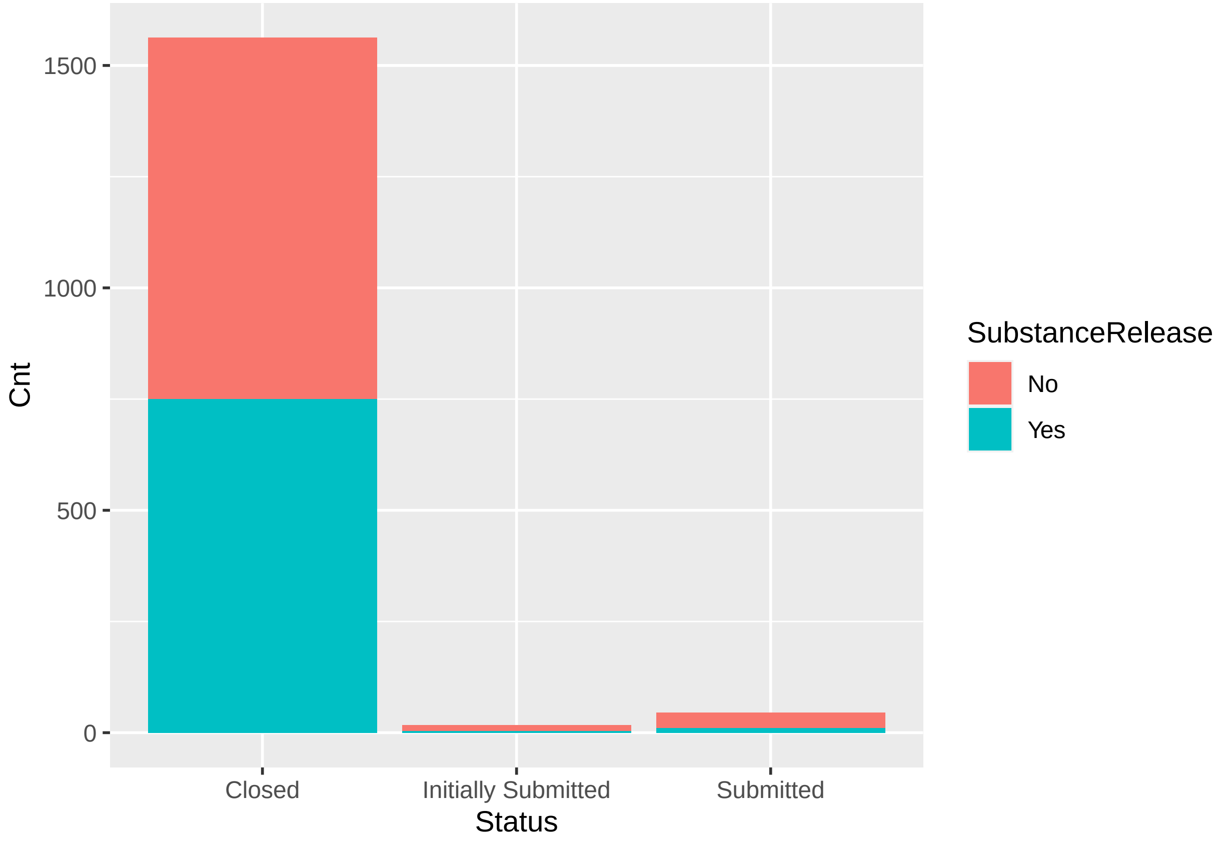
```
t4<-data%>%
  group_by(Status,SubstanceRelease)%>%
  summarize(Cnt = n())%>%
  arrange(desc(Cnt))
```

```
## ‘summarise()‘ has grouped output by ’Status’. You can override using the
## ‘.groups‘ argument.
```

```
t4
```

```
## # A tibble: 6 x 3
## # Groups:   Status [3]
##   Status              SubstanceRelease   Cnt
##   <chr>               <chr>            <int>
## 1 Closed              No                 812
## 2 Closed              Yes                750
## 3 Submitted           No                  34
## 4 Initially Submitted No                  13
## 5 Submitted           Yes                 11
## 6 Initially Submitted Yes                  4
```

```
t4%>%
  ggplot(aes(x=Status, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



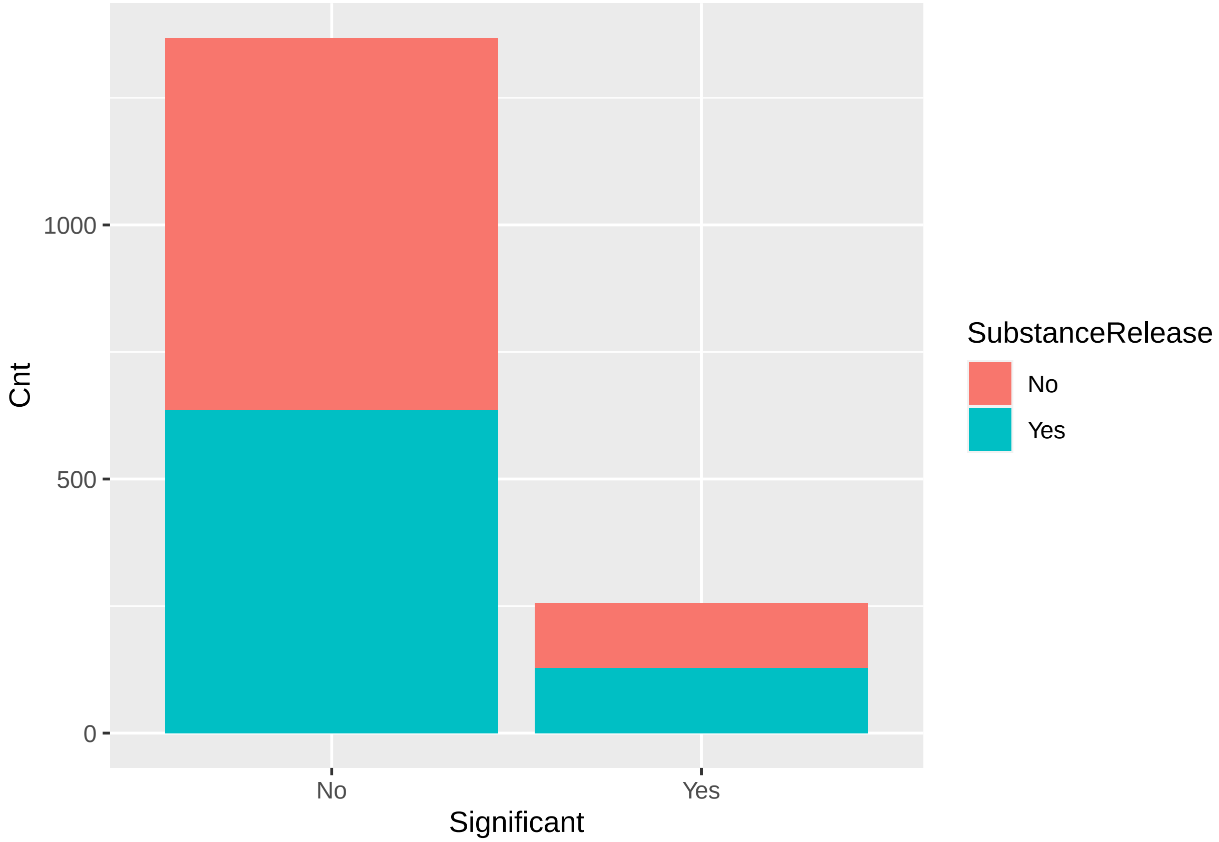
```
t5<-data%>%
  group_by(Significant,SubstanceRelease)%>%
  summarize(Cnt = n())%>%
  arrange(desc(Cnt))
```

```
## `summarise()` has grouped output by 'Significant'. You can override using the
## `.groups` argument.
```

```
t5
```

```
## # A tibble: 4 x 3
## # Groups:   Significant [2]
##   Significant SubstanceRelease   Cnt
##   <chr>       <chr>            <int>
## 1 No          No                 732
## 2 No          Yes                636
## 3 Yes         Yes                129
## 4 Yes         No                 127
```

```
t5%>%
  ggplot(aes(x=Significant, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



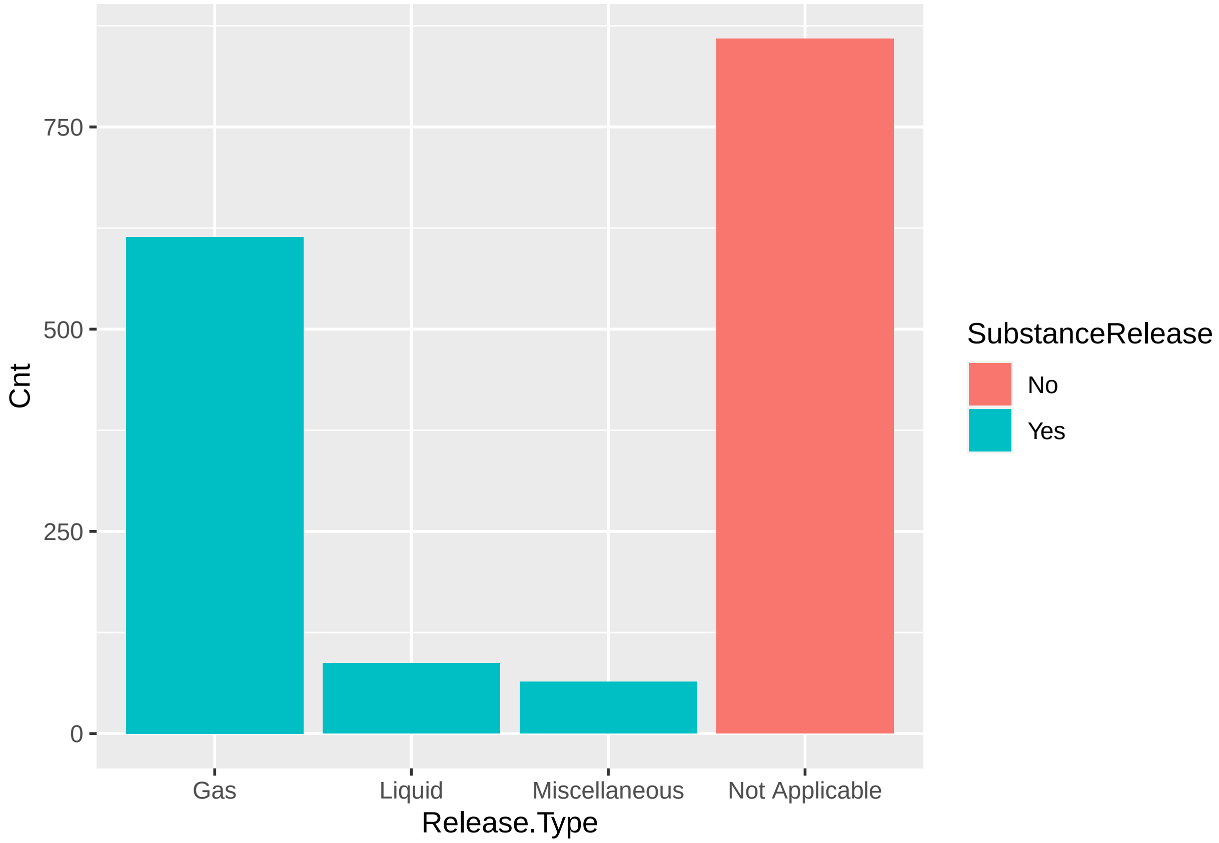
```
t6<-data%>%
  group_by(Release.Type,SubstanceRelease)%>%
  summarize(Cnt = n())%>%
  arrange(desc(Cnt))
```

```
## ‘summarise()‘ has grouped output by ’Release.Type’. You can override using the
## ‘.groups‘ argument.
```

```
t6
```

```
## # A tibble: 4 x 3
## # Groups:   Release.Type [4]
##   Release.Type   SubstanceRelease   Cnt
##   <chr>          <chr>            <int>
## 1 Not Applicable No                 859
## 2 Gas            Yes                614
## 3 Liquid         Yes                 87
## 4 Miscellaneous  Yes                 64
```

```
t6%>%
  ggplot(aes(x=Release.Type, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



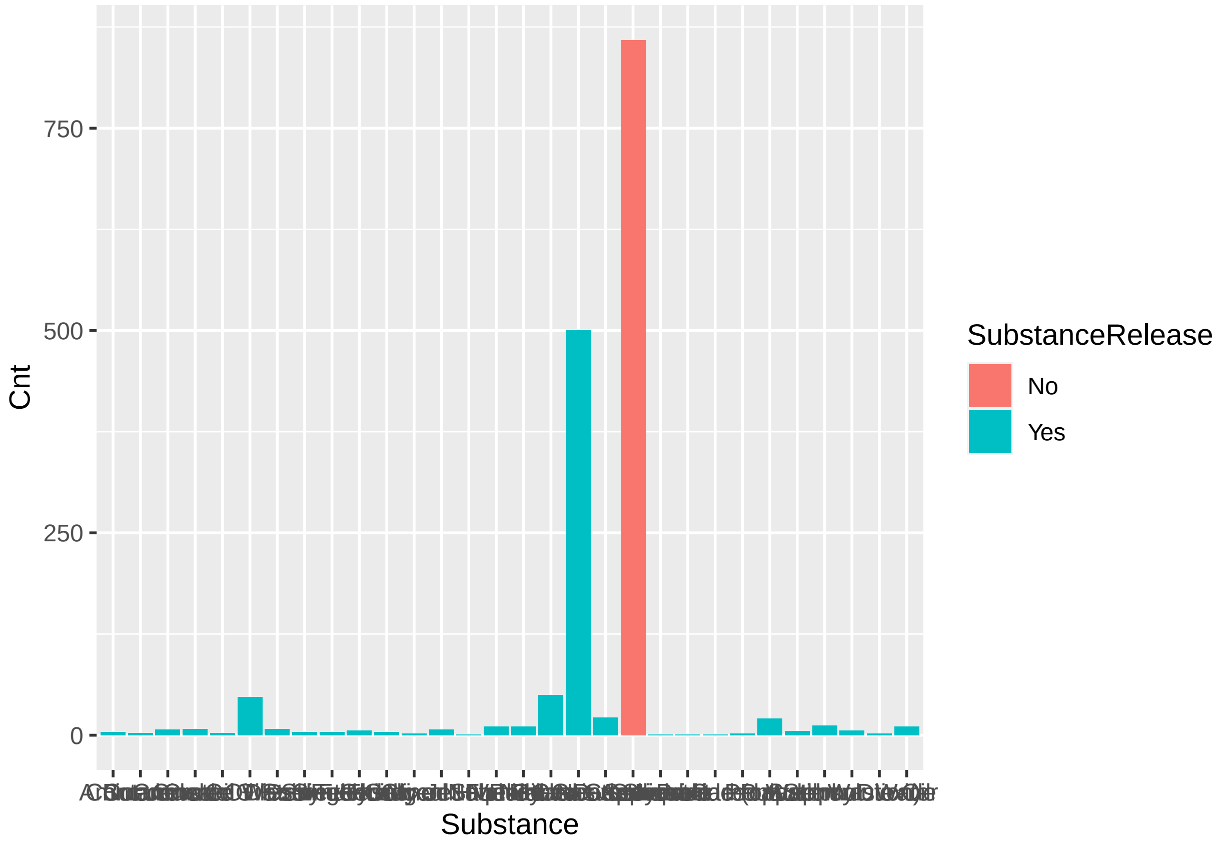
```
t7<-data%>%
  group_by(Substance,SubstanceRelease)%>%
  summarize(Cnt = n())%>%
  arrange(desc(Cnt))
```

```
## `summarise()` has grouped output by 'Substance'. You can override using the
## `.groups` argument.
```

```
t7
```

```
## # A tibble: 30 x 3
## # Groups:   Substance [30]
##    Substance              SubstanceRelease   Cnt
##    <chr>                  <chr>            <int>
##  1 Not Applicable         No                 859
##  2 Natural Gas - Sweet    Yes                501
##  3 Natural Gas - Sour     Yes                 50
##  4 Crude Oil - Sweet      Yes                 47
##  5 Natural Gas Liquids    Yes                 22
##  6 Propane                Yes                 21
##  7 Sulphur               Yes                 12
##  8 Lube Oil              Yes                 11
##  9 Mixed HVP Hydrocarbons Yes                11
## 10 Water                 Yes                 11
## # ... with 20 more rows
```

```
t7%>%
  ggplot(aes(x=Substance, y=Cnt,fill=SubstanceRelease)) +
  geom_bar(stat="identity")
```



```
table(data$SubstanceRelease,data$Release.Type)
```

```
##      
##       Gas Liquid Miscellaneous Not Applicable
##   No    0      0             0            859
##   Yes 614     87            64              0
```

```
table(data$SubstanceRelease,data$Significant)
```

```
##      
##        No Yes
##   No  732 127
##   Yes 636 129
```

```
chisq.test(table(data$SubstanceRelease,data$Significant))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(data$SubstanceRelease, data$Significant)
## X-squared = 1.1641, df = 1, p-value = 0.2806
```

#modeling with tain and test

```
data<-data%>%
  mutate(SubstanceRelease = ifelse(SubstanceRelease == "Yes",1,0),
         Significant = ifelse(Significant == "Yes",1,0))
n<-nrow(data)
n.train = trunc(0.7*n)
n.test = n - n.train
train = sample(1:n,n.train)
train.x = data[train,-16]
train.y = data[train,16]
test.x = data[-train,-16]
test.y = data[-train,16]

fit<-glm(SubstanceRelease ~ Significant ,family = binomial(link="logit"),data=cbind(train.x,train.y))
summary(fit)
```

```
##
## Call:
## glm(formula = SubstanceRelease ~ Significant, family = binomial(link = "logit"),
##     data = cbind(train.x, train.y))
##
## Deviance Residuals:
##    Min       1Q   Median       3Q      Max
## -1.177   -1.128   -1.128    1.228    1.228
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.11778    0.06493  -1.814   0.0697 .
## Significant  0.11778    0.16111   0.731   0.4647
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1572.1  on 1135  degrees of freedom
## Residual deviance: 1571.5  on 1134  degrees of freedom
## AIC: 1575.5
##
## Number of Fisher Scoring iterations: 3
```

```
yhat<-round(predict.glm(fit,newdata = test.x,type = "response"))
tb<-table(yhat,as.data.frame(test.y)[,1])
sum(diag(tb))/sum(tb)
```

```
## [1] 0.5389344
```

```
fit1<-glm(SubstanceRelease ~ Latitude + Longitude ,family = binomial(link="logit"),data=cbind(train.x,t
summary(fit1)
```

```
##
## Call:
## glm(formula = SubstanceRelease ~ Latitude + Longitude, family = binomial(link = "logit"),
##     data = cbind(train.x, train.y))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6972  -1.1022  -0.6614   1.2190   1.8533
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.938870   0.945354  -4.167 3.09e-05 ***
## Latitude     0.188423   0.028994   6.499 8.10e-11 ***
## Longitude    0.055836   0.006889   8.105 5.29e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1572.1  on 1135  degrees of freedom
## Residual deviance: 1496.3  on 1133  degrees of freedom
## AIC: 1502.3
##
## Number of Fisher Scoring iterations: 4
```

```
yhat<-round(predict.glm(fit1,newdata = test.x,type = "response"))
tb<-table(yhat,as.data.frame(test.y)[,1])
sum(diag(tb))/sum(tb)
```

```
## [1] 0.5942623
```

```
fit2<-glm(SubstanceRelease ~ Province,family = binomial(link="logit"),data=cbind(train.x,train.y))
summary(fit2)
```

```
##
## Call:
## glm(formula = SubstanceRelease ~ Province, family = binomial(link = "logit"),
##     data = cbind(train.x, train.y))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6444  -1.1247  -0.9776   1.2310   1.8123
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    -0.4900     0.1020  -4.804 1.56e-06 ***
## ProvinceBritish Columbia        0.3648     0.1560   2.338  0.01939 *
## ProvinceManitoba                0.7311     0.3026   2.416  0.01569 *
## ProvinceNew Brunswick           3.9557     0.7253   5.454 4.92e-08 ***
## ProvinceNorthwest Territories   1.5886     1.1592   1.370  0.17056
## ProvinceNova Scotia             3.5345     1.0286   3.436  0.00059 ***
## ProvinceOntario                 0.3886     0.1892   2.054  0.03995 *
## ProvinceQuebec                 -0.9372     0.4659  -2.011  0.04428 *
## ProvinceSaskatchewan            0.4349     0.2171   2.003  0.04514 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1572.1  on 1135  degrees of freedom
## Residual deviance: 1439.2  on 1127  degrees of freedom
## AIC: 1457.2
##
## Number of Fisher Scoring iterations: 6
```

```
yhat<-round(predict.glm(fit2,newdata = test.x,type = "response"))
tb<-table(yhat,as.data.frame(test.y)[,1])
sum(diag(tb))/sum(tb)
```

```
## [1] 0.6168033
```

```
fit3<-glm(SubstanceRelease ~ Release.Type,family = binomial(link="logit"),data=cbind(train.x,train.y))
```

```
## Warning: glm.fit: algorithm did not converge
```

```
summary(fit3)
```

```
##
## Call:
## glm(formula = SubstanceRelease ~ Release.Type, family = binomial(link = "logit"),
##     data = cbind(train.x, train.y))
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06   2.409e-06
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  2.657e+01  1.709e+04   0.002    0.999
## Release.TypeLiquid           4.829e-06  4.979e+04   0.000    1.000
## Release.TypeMiscellaneous   -3.594e-09  5.417e+04   0.000    1.000
## Release.TypeNot Applicable -5.313e+01  2.247e+04  -0.002    0.998
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.5721e+03  on 1135  degrees of freedom
## Residual deviance: 6.5906e-09  on 1132  degrees of freedom
## AIC: 8
##
## Number of Fisher Scoring iterations: 25
```

```
yhat<-round(predict.glm(fit3,newdata = test.x,type = "response"))
tb<-table(yhat,as.data.frame(test.y)[,1])
sum(diag(tb))/sum(tb)
```

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

```
fit4<-glm(SubstanceRelease ~ Status,family = binomial(link="logit"),data=cbind(train.x,train.y))
summary(fit4)
```

```
##
## Call:
## glm(formula = SubstanceRelease ~ Status, family = binomial(link = "logit"),
##     data = cbind(train.x, train.y))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1565  -1.1565  -0.5829   1.1984   2.0963
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.04933    0.06046  -0.816 0.414583
## StatusInitially Submitted -2.03012    1.06236  -1.911 0.056010 .
## StatusSubmitted            -1.63707    0.49060  -3.337 0.000847 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1572.1  on 1135  degrees of freedom
## Residual deviance: 1551.3  on 1133  degrees of freedom
## AIC: 1557.3
##
## Number of Fisher Scoring iterations: 4
```

```
yhat<-round(predict.glm(fit4,newdata = test.x,type = "response"))
tb<-table(yhat,as.data.frame(test.y)[,1])
sum(diag(tb))/sum(tb)
```

```
## [1] 0.5389344
```

```
#fit5<-glm(SubstanceRelease ~ Substance,family = #binomial(link="logit"),data=cbind(train.x,train.y))
#yhat<-round(predict.glm(fit5,newdata = test.x,type = "response"))
#tb<-table(yhat,as.data.frame(test.y)[,1])
#sum(diag(tb))/sum(tb)

#fit6<-glm(SubstanceRelease ~ Nearest.Populated.Centre,family = #binomial(link="logit"),data=cbind(trai
#yhat<-round(predict.glm(fit6,newdata = test.x,type = "response"))
#tb<-table(yhat,as.data.frame(test.y)[,1])
#sum(diag(tb))/sum(tb)

#fit6<-glm(SubstanceRelease ~ Company,family = #binomial(link="logit"),data=cbind(train.x,train.y))
```

```
#yhat<-round(predict.glm(fit6,newdata = test.x,type = "response"))
#tb<-table(yhat,as.data.frame(test.y)[,1])
#sum(diag(tb))/sum(tb)
```