Final group project by Finance_5

Siri, Chantrika, Aparna, Avinash, Shashi

9/15/2021

EFFECT OF DISASTERS ON THE INSURANCE STOCK

Question of interest

1. How are financial markets reacting to a natural disaster.
2. Capture the movement of the stock of the insurance company on the day of the disaster
3. Finally calculate the cumulative movement of the stock on different disaster begin dates.

Short Description:-

In this project we are considering four datasets, the first dataset is "Disasters", in this dataset we are considering 5 disasters - Fire,Flood,Hurricane,Severe Stroms and Snow. The second, third and fourth datasets are Stocks data for top 3 listed insurance companies - AllState,Progressive and Travellers. So taking stocks data into consideration we are analyzying how disasters effected the stocks prices from year 2000 -2020.

Source:-

FEMA(Federal Emergency Management Agency) Disaster Declarations Summary is a summarized data set describing all federally declared disasters. This data set lists all official FEMA Disaster Declarations, beginning with the first disaster declaration in 1953 and features all three disaster declaration types: major disaster, emergency, and fire management assistance.

'https://www.fema.gov/api/open/v2/DisasterDeclarationsSummaries.csv'.

Yahoo - This is a media property that is part of Yahoo! network.It provides financial news, data and commentary including stock quotes, press releases, financial reports, and original content.

'https://query1.finance.yahoo.com/v7/finance/download/ALL?period1=944006400&period2=1631664000&interval=1d&events=history&includeAdjustedClose=true'

'https://query1.finance.yahoo.com/v7/finance/download/PGR?period1=944006400&period2=1631664000&interval=1d&events=history&includeAdjustedClose=true'

'https://query1.finance.yahoo.com/v7/finance/download/TRV?period1=944006400&period2=1631664000&interval=1d&events=history&includeAdjustedClose=true'

Libraries

We used three R libraries for this project

tidyverse and **janitor** is used for cleaning and analysing the data, while **ggrepel** is used on top of ggplot to produce beautiful visualizations.

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
```

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

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(ggrepel)
```

Loading Disasters dataset

We are loading the dataset directly from the FEMA api.

```
disaster_data <- read_csv('https://www.fema.gov/api/open/v2/DisasterDeclarationsSummaries.csv')
```

```
## Rows: 62591 Columns: 23
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (10): femaDeclarationString, state, declarationType, incidentType, decl...
## dbl   (8): disasterNumber, fyDeclared, ihProgramDeclared, iaProgramDeclared,...
## dttm  (5): declarationDate, incidentBeginDate, incidentEndDate, disasterClos...
```

```
##
## 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.
```

Loading stock datasets

We are using Yahoo finance links to directly load all the stock datasets.

Loading AllState Stock dataset

```
allstate_data <- read_csv('https://query1.finance.yahoo.com/v7/finance/download/ALL?period1=944006400&pe
```

```
## Rows: 5482 Columns: 7
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (6): Open, High, Low, Close, Adj Close, Volume
## date (1): Date
```

```
##
## 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.
```

```
allstate_data <- subset(allstate_data, select = c("Date","Open","High","Low","Close"))
```

```
colnames(allstate_data) <- c("Date", "allstate_open", "allstate_high","allstate_low","allstate_close")

head(allstate_data)
```

```
## # A tibble: 6 x 5
##   Date       allstate_open allstate_high allstate_low allstate_close
##   <date>             <dbl>         <dbl>        <dbl>          <dbl>
## 1 1999-12-01          26.3          26.7         26.1           26.1
## 2 1999-12-02          26.1          26.9         26.1           26.6
## 3 1999-12-03          26.8          28           26.8           27.2
## 4 1999-12-06          27            28           26.8           27
## 5 1999-12-07          26.9          27.9         26.9           27
## 6 1999-12-08          27            27.7         27             27.2
```

Selected the columns that are needed for the visualizations and formed a subset using those columns. Renamed the columns according to convenience

Loading Progressive Stock dataset

```
progressive_data <- read_csv('https://query1.finance.yahoo.com/v7/finance/download/PGR?period1=944006400
```

```
## Rows: 5482 Columns: 7
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (6): Open, High, Low, Close, Adj Close, Volume
## date (1): Date
```

```
##
## 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.
```

```
progressive_data <- subset(progressive_data, select = c("Date","Open","High","Low","Close"))

colnames(progressive_data) <- c("Date", "progressive_open", "progressive_high","progressive_low","progre

head(progressive_data)
```

```
## # A tibble: 6 x 5
##   Date       progressive_open progressive_high progressive_low progressive_close
##   <date>                <dbl>            <dbl>           <dbl>             <dbl>
## 1 1999-12-01             6.72             6.77            6.64              6.64
## 2 1999-12-02             6.67             6.71            6.55              6.61
## 3 1999-12-03             6.62             6.71            6.55              6.56
## 4 1999-12-06             6.49             6.52            6.38              6.41
## 5 1999-12-07             6.37             6.38            6.26              6.27
## 6 1999-12-08             6.23             6.29            5.99              6
```

Selected the columns that are needed for the visualizations and formed a subset using those columns. Renamed the columns according to our convenience

Loading Travellers Stock dataset

```
travellers_data <- read_csv('https://query1.finance.yahoo.com/v7/finance/download/TRV?period1=944006400
```

```
## Rows: 5482 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (6): Open, High, Low, Close, Adj Close, Volume
## date (1): Date
```

```
##
## 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.
```

```
travellers_data <- subset(travellers_data, select = c("Date","Open","High","Low","Close"))

colnames(travellers_data) <- c("Date", "travellers_open", "travellers_high","travellers_low","traveller

head(travellers_data)
```

```
## # A tibble: 6 x 5
##   Date       travellers_open travellers_high travellers_low travellers_close
##   <date>               <dbl>           <dbl>          <dbl>            <dbl>
## 1 1999-12-01            30.4            31.2           30.2             30.9
## 2 1999-12-02            31              31.5           31               31.5
## 3 1999-12-03            32              32.9           31.6             31.8
## 4 1999-12-06            31.9            31.9           30.8             31
## 5 1999-12-07            30.4            31.2           30.4             30.5
## 6 1999-12-08            30.5            31.2           30.4             30.8
```

Selected the columns that are needed for the visualizations and formed a subset using those columns. Renamed the columns according to our convenience

Cleaning Disaster dataset

```
disaster_data <- subset(disaster_data, select = c("incidentType", "state", "declarationType","fyDeclare

disaster_data <- disaster_data %>%  distinct()

disaster_data$incidentBeginDate = as.Date(disaster_data$incidentBeginDate)

disaster_data <- disaster_data %>% filter(incidentBeginDate>as.Date("2000-01-01"))
```

Selected the columns that are needed for the visualizations and formed a subset using those columns. To remove duplicate values distinct() function is used and filtered the values from the year 2000

Combining Disaster and Stock datasets

Joining all the four datasets to get into a single dataset, we are left joining all the datasets on the basis of dates the disasters occurred.

```
disaster_stock_data <- disaster_data %>%
  left_join(allstate_data, by=c("incidentBeginDate" = "Date")) %>%
  left_join(progressive_data, by=c("incidentBeginDate" = "Date"))%>%
  left_join(travellers_data, by=c("incidentBeginDate" = "Date"))%>%
  na.omit((disaster_data_all))

disaster_stock_data
```

```
## # A tibble: 1,189 x 20
##    incidentType    state declarationType fyDeclared declarationDate
##    <chr>           <chr> <chr>                <dbl> <dttm>
##  1 Tornado         KY    DR                      2000 2000-01-10 13:00:00
##  2 Severe Storm(s) NC    DR                      2000 2000-01-31 18:30:00
##  3 Snow            LA    DR                      2000 2000-02-15 17:30:00
##  4 Severe Storm(s) GA    DR                      2000 2000-02-15 15:00:00
##  5 Severe Storm(s) VA    DR                      2000 2000-02-28 14:20:00
##  6 Flood           WV    DR                      2000 2000-02-28 15:00:00
##  7 Severe Storm(s) KY    DR                      2000 2000-02-28 17:30:00
##  8 Severe Storm(s) OH    DR                      2000 2000-03-07 14:45:00
##  9 Severe Storm(s) AL    DR                      2000 2000-03-17 16:00:00
## 10 Tornado         TX    DR                      2000 2000-04-07 13:45:00
## # ... with 1,179 more rows, and 15 more variables: incidentBeginDate <date>,
## #   incidentEndDate <dttm>, disasterCloseoutDate <dttm>, allstate_open <dbl>,
## #   allstate_high <dbl>, allstate_low <dbl>, allstate_close <dbl>,
## #   progressive_open <dbl>, progressive_high <dbl>, progressive_low <dbl>,
## #   progressive_close <dbl>, travellers_open <dbl>, travellers_high <dbl>,
## #   travellers_low <dbl>, travellers_close <dbl>
```

We are using the tabyl function in janitor package to find out how many times each disaster occurred in the timeframe we considered i.e (2000-present) and what is its percentage of occurrence. Also we are only considering the disasters that occurred more than 50 times for our project.

```
# library(janitor)

no_of_disasters <- tabyl(disaster_data,incidentType) %>%
  filter(n >50) %>% arrange(desc(n))
no_of_disasters
```

```
##      incidentType    n    percent
##              Fire 1164 0.41364606
##   Severe Storm(s)  758 0.26936745
##         Hurricane  286 0.10163468
##             Flood  186 0.06609808
##        Biological  136 0.04832978
##              Snow   92 0.03269367
##  Severe Ice Storm   62 0.02203269
```

Till now we have loaded the datasets and filtered the data according to our suitability, now we find out the Number of Disasters that occurred in each year and try to visualize the trend.

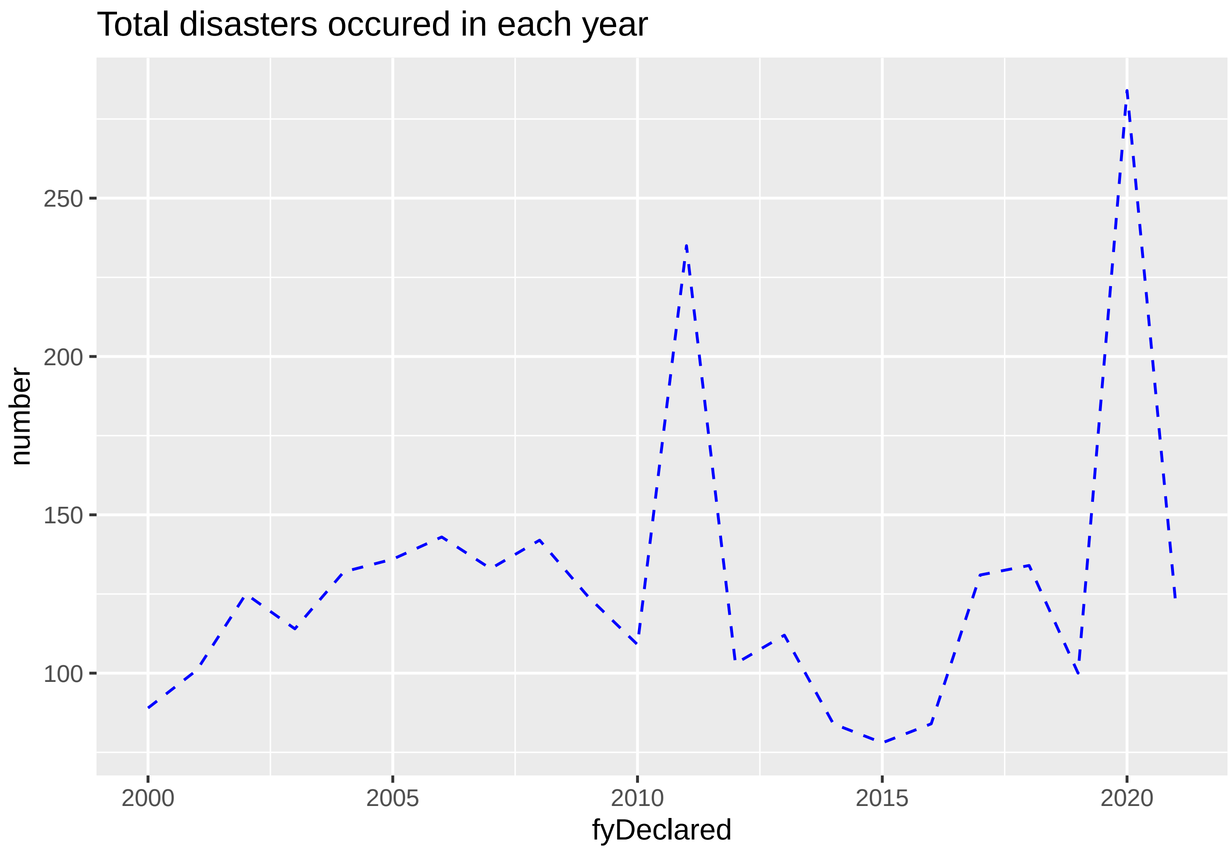
```
count_incidents <- disaster_data %>% group_by(fyDeclared)%>%  summarise(number = n())

count_incidents
```

```
## # A tibble: 22 x 2
##    fyDeclared number
##         <dbl>  <int>
##  1       2000     89
##  2       2001    101
##  3       2002    125
##  4       2003    114
##  5       2004    132
##  6       2005    136
##  7       2006    143
##  8       2007    133
##  9       2008    142
## 10       2009    124
## # ... with 12 more rows
```

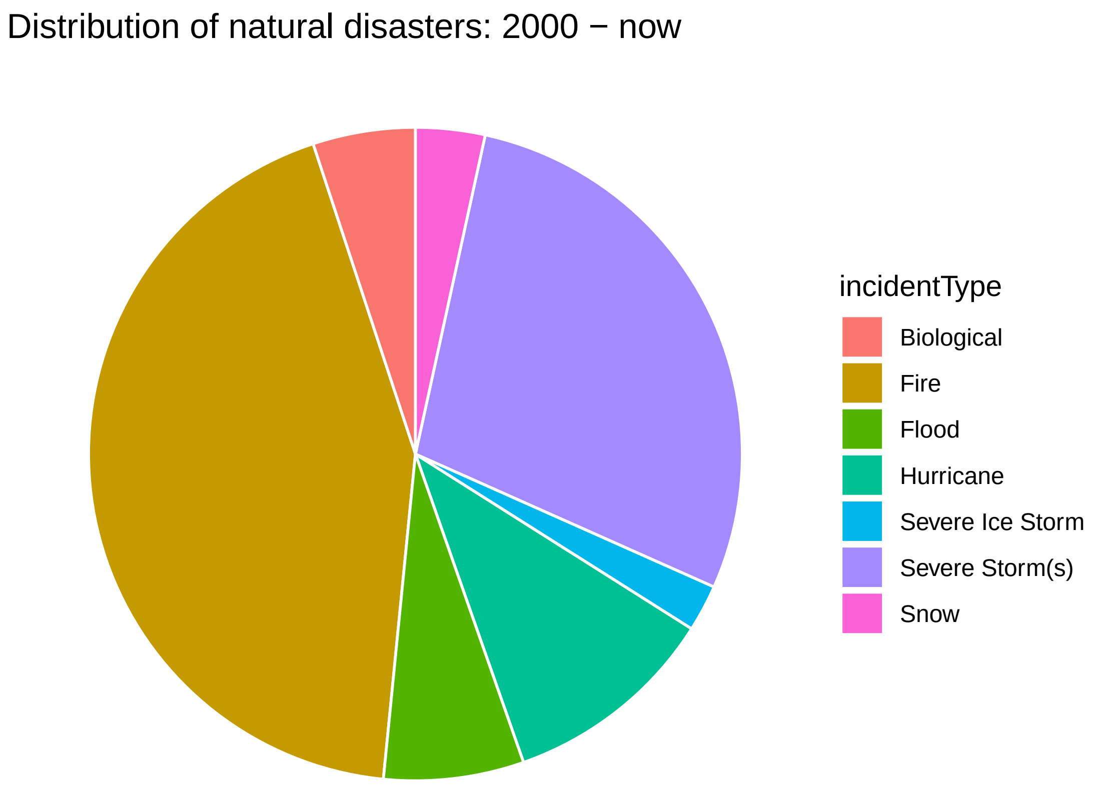
We're plotting a line chart to give us a better understanding of the occurrence of disasters in each year.

```
ggplot(data=count_incidents, aes(x=fyDeclared, y = number))+ ggtitle("Total disasters occured in each ye
    geom_line(color = "blue", linetype = 2)
```



Visualization to plot the distribution of the disasters in the US from 2000 - Present

```
# library(ggrepel)
ggplot(no_of_disasters %>% arrange(desc(n)), aes(x="", y=n, fill=incidentType ,inherit.aes = FALSE)) +
 geom_bar(stat="identity", width=1, color="white") +
 coord_polar("y", start=0) +
 theme_void() + ggtitle("Distribution of natural disasters: 2000 - now")
```



Impact of natural disasters on the stocks of the three insurance Companies we chose

Here we are calculating the movement of the stock on the day of occurrence of each disaster. In addition to that we are also calculating the minimum dip or rise of the three stocks.

```
disaster_stock_data<- disaster_stock_data %>% mutate(dip_allstate = allstate_close - allstate_open)

disaster_stock_data<- disaster_stock_data %>% mutate(dip_progressive = progressive_close- progressive_op

disaster_stock_data<-disaster_stock_data %>% mutate(dip_travellers = travellers_close - travellers_open

disaster_stock_data <- disaster_stock_data %>% mutate(dip = apply(disaster_stock_data[,c('dip_allstate'

glimpse(disaster_stock_data)
```

```
## Rows: 1,189
## Columns: 24
## $ incidentType        <chr> "Tornado", "Severe Storm(s)", "Snow", "Severe Sto~
## $ state               <chr> "KY", "NC", "LA", "GA", "VA", "WV", "KY", "OH", "~
## $ declarationType     <chr> "DR", "DR", "DR", "DR", "DR", "DR", "DR", "DR", "~
## $ fyDeclared          <dbl> 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2000, 2~
## $ declarationDate     <dttm> 2000-01-10 13:00:00, 2000-01-31 18:30:00, 2000-0~
## $ incidentBeginDate   <date> 2000-01-03, 2000-01-24, 2000-01-27, 2000-02-14, ~
## $ incidentEndDate     <dttm> 2000-01-04 00:00:00, 2000-02-01 00:00:00, 2000-0~
## $ disasterCloseoutDate <dttm> 2012-05-10 00:00:00, 2012-06-28 00:00:00, 2016-0~
## $ allstate_open        <dbl> 24.0000, 23.6250, 23.4375, 21.6875, 23.1250, 21.2~
## $ allstate_high        <dbl> 24.1875, 23.8750, 23.9375, 22.0625, 23.6250, 21.5~
## $ allstate_low         <dbl> 23.2500, 22.8125, 23.1875, 21.0625, 22.8125, 20.7~
## $ allstate_close       <dbl> 23.5000, 22.8125, 23.1875, 21.2500, 22.8750, 21.0~
## $ progressive_open     <dbl> 6.052083, 5.453125, 5.343750, 4.500000, 5.515625,~
## $ progressive_high     <dbl> 6.083333, 5.557292, 5.473958, 4.500000, 5.515625,~
## $ progressive_low      <dbl> 5.890625, 5.416667, 5.322917, 4.312500, 5.328125,~
## $ progressive_close    <dbl> 5.890625, 5.494792, 5.390625, 4.343750, 5.333333,~
## $ travellers_open      <dbl> 33.2500, 31.6250, 32.0000, 24.2500, 32.1250, 24.5~
## $ travellers_high      <dbl> 33.2500, 32.4375, 32.3750, 24.5625, 32.8125, 24.6~
## $ travellers_low       <dbl> 32.6250, 31.3125, 29.8125, 24.0625, 31.2500, 23.1~
## $ travellers_close     <dbl> 33.0000, 32.1250, 30.5625, 24.2500, 31.3750, 23.6~
## $ dip_allstate         <dbl> -0.5000, -0.8125, -0.2500, -0.4375, -0.2500, -0.2~
## $ dip_progressive      <dbl> -0.161458, 0.041667, 0.046875, -0.156250, -0.1822~
## $ dip_travellers       <dbl> -0.2500, 0.5000, -1.4375, 0.0000, -0.7500, -0.937~
## $ dip                  <dbl> -0.500000, -0.812500, -1.437500, -0.437500, -0.75~
```

Here we are creating the dataset mean_dip_rise which contains the mean dip or rise of a particular stock depending on the disaster, we are then joining that dataset with the no_of_disasters dataset we created which contains the total number of each disasters that occurred in the timeframe.

```
mean_dip_rise <- disaster_stock_data %>% filter(incidentType == c('Fire','Flood','Hurricane','Severe Sto
```

```
## Warning in incidentType == c("Fire", "Flood", "Hurricane", "Severe Storm(s)", :
## longer object length is not a multiple of shorter object length
```

```
no_of_disasters <- full_join(no_of_disasters, mean_dip_rise)
```

```
## Joining, by = "incidentType"
```

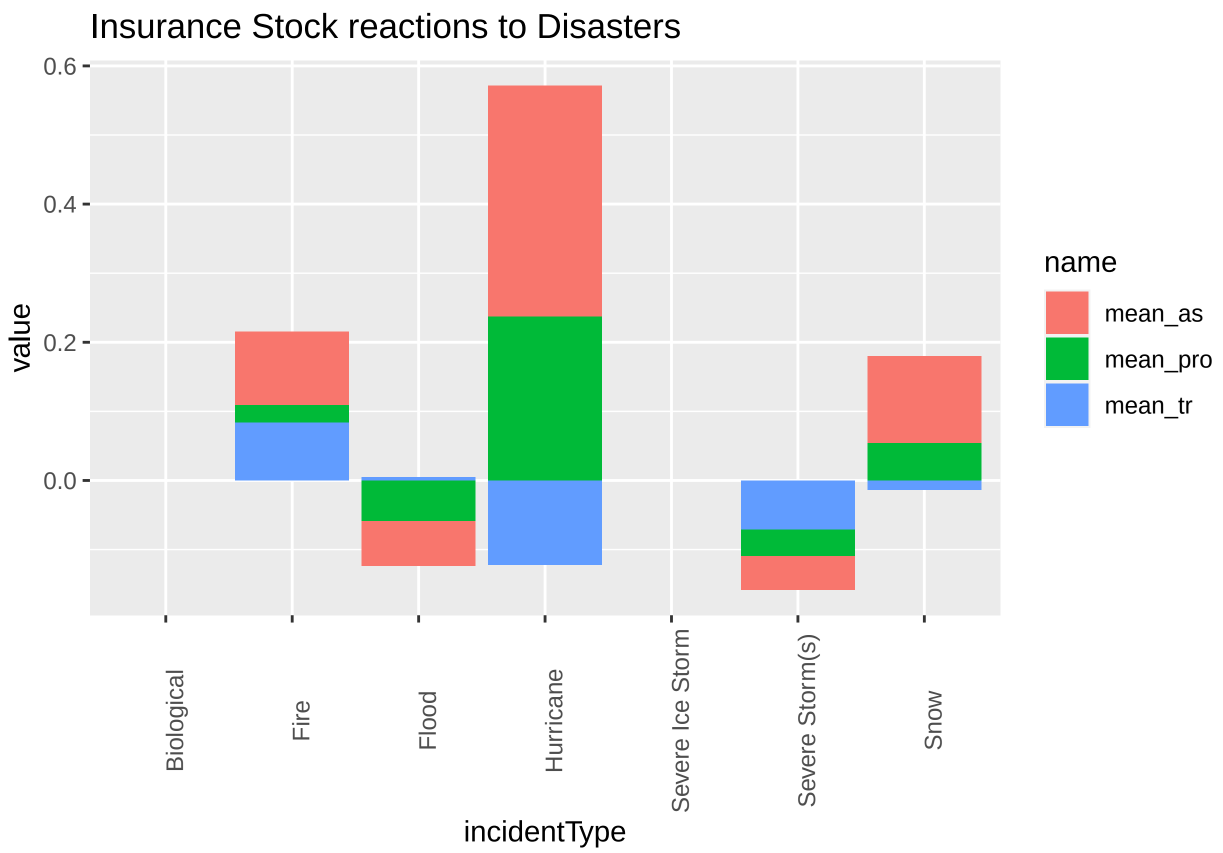
```
no_of_disasters
```

```
##      incidentType    n    percent     mean_as      mean_tr    mean_pro
##              Fire 1164 0.41364606  0.10705260  0.084039295  0.02465141
##   Severe Storm(s)  758 0.26936745 -0.04961045 -0.070779052 -0.03813579
##         Hurricane  286 0.10163468  0.33500014 -0.122142393  0.23669582
##             Flood  186 0.06609808 -0.06533247  0.004667067 -0.05855573
##        Biological  136 0.04832978          NA           NA          NA
##              Snow   92 0.03269367  0.12596077 -0.013269846  0.05413485
##  Severe Ice Storm   62 0.02203269          NA           NA          NA
```

```
summ_p <- pivot_longer(no_of_disasters,cols = c('mean_as','mean_tr','mean_pro'))

ggplot(summ_p, aes(fill=name, y=value, x=incidentType)) +
    geom_bar(position="stack", stat="identity") + ggtitle("Insurance Stock reactions to Disasters") + th
          axis.text.x = element_text(angle = 90))
```

```
## Warning: Removed 6 rows containing missing values (position_stack).
```



Basic Linear Model of the data

We ran a multiple linear regression model for our data using the mean dips or rises in the stock value to predict the number of disasters, but because of the limitations in our data we ended up with a high error. An advanced model can be used in the future to get more accurate results.

```
model <- lm(formula = n ~ mean_as+mean_tr+ mean_pro, data = no_of_disasters)

summary(model)
```

```
##
## Call:
## lm(formula = n ~ mean_as + mean_tr + mean_pro, data = no_of_disasters)
##
## Residuals:
##       1       2       3       4       6
##  354.85  421.63   30.38 -388.86 -418.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    600.5     1343.7   0.447    0.732
## mean_as      -1268.2    37430.5  -0.034    0.978
## mean_tr       3473.2    17363.2   0.200    0.874
## mean_pro      2130.0    56987.8   0.037    0.976
##
## Residual standard error: 794.1 on 1 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.2294, Adjusted R-squared:  -2.082
## F-statistic: 0.09925 on 3 and 1 DF,  p-value: 0.9497
```

Conclusions:-

1. During Fire and Hurricane insurance companies gain profit.
2. When floods, severe storms and snow occurred stocks of progresive increased.
3. Allstate was affected the most by natural disasters.
4. Insurance companies took better measures against fire and severe storms.

Bias:-

- We considered the disasters that occurred more than 50 times. What about the other natural disasters ?
- Weekends are not considered as the stock markets are closed on weekends.
- National holidays are also not considered.
- Data was lost during cleaning.

Extension

- Can we deduce why some natural disasters affected particular companies differently?
    1. Flood - progressive
    2. Severe storm(s) - all state
    3. Snow - travellers
- Can we find out the best insurance company based on geographic data?