Case Study: Hazardous Earth Objects

Eric Chu

2023-02-01

Business Task

Imagine a government own agency wants to figure out what to set their instruments or what devices they need to watch out for potentially hazardous astro objects closest to Earth. Explore data collected from NASA to find out what measurements and values that would deem an astro object dangerous.

Data Source

This is data collected in the year of 2021 of all the recorded celestial objects that are closest to the planet Earth in this link. This is a public data set under the license of CC0: Public Domain

Data Cleaning/Analysis

Setup the packages and data:

```
install.packages("tidyverse")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("skimr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("janitor")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("ggplot2")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2
## ──
```

```
## ✔ ggplot2 3.4.0      ✔ purrr   1.0.1
## ✔ tibble  3.1.8      ✔ dplyr   1.0.10
## ✔ tidyr   1.3.0      ✔ stringr 1.5.0
## ✔ readr   2.1.3      ✔ forcats 0.5.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(skimr)
library(janitor)
```

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

```
library(ggplot2)
```

Gather the Data:

```
data1 <- read_csv("neo_v2.csv")
```

```
## Rows: 90836 Columns: 10
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (2): name, orbiting_body
## dbl (6): id, est_diameter_min, est_diameter_max, relative_velocity, miss_dis...
## lgl (2): sentry_object, hazardous
##
## 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.
```

Check the gathered data:

```
head(data1)
```

id	name	est_diameter_min	est_diameter_max	relative_velocity	miss_distance ▸
<dbl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
2162635	162635 (2000 SS164)	1.19827080	2.67941497	13569.25	54839744
2277475	277475 (2005 WK4)	0.26580000	0.59434687	73588.73	61438127
2512244	512244 (2015 YE18)	0.72202956	1.61450717	114258.69	49798725
3596030	(2012 BV13)	0.09650615	0.21579430	24764.30	25434973
3667127	(2014 GE35)	0.25500869	0.57021676	42737.73	46275567
54138696	(2021 GY23)	0.03635423	0.08129053	34297.59	40585691

6 rows | 1-6 of 10 columns

```
colnames(data1)
```

```
##  [1] "id"                 "name"               "est_diameter_min"
##  [4] "est_diameter_max"   "relative_velocity"  "miss_distance"
##  [7] "orbiting_body"      "sentry_object"      "absolute_magnitude"
## [10] "hazardous"
```

```
str(data1)
```

```
## spc_tbl_ [90,836 × 10] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ id                : num [1:90836] 2162635 2277475 2512244 3596030 3667127 ...
##  $ name              : chr [1:90836] "162635 (2000 SS164)" "277475 (2005 WK4)" "512244 (2015 YE18)" "(2012 BV1
3)" ...
##  $ est_diameter_min  : num [1:90836] 1.1983 0.2658 0.722 0.0965 0.255 ...
##  $ est_diameter_max  : num [1:90836] 2.679 0.594 1.615 0.216 0.57 ...
##  $ relative_velocity : num [1:90836] 13569 73589 114259 24764 42738 ...
##  $ miss_distance     : num [1:90836] 54839744 61438127 49798725 25434973 46275567 ...
##  $ orbiting_body     : chr [1:90836] "Earth" "Earth" "Earth" "Earth" ...
##  $ sentry_object     : logi [1:90836] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ absolute_magnitude: num [1:90836] 16.7 20 17.8 22.2 20.1 ...
##  $ hazardous         : logi [1:90836] FALSE TRUE FALSE FALSE TRUE FALSE ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   id = col_double(),
##   ..   name = col_character(),
##   ..   est_diameter_min = col_double(),
##   ..   est_diameter_max = col_double(),
##   ..   relative_velocity = col_double(),
##   ..   miss_distance = col_double(),
##   ..   orbiting_body = col_character(),
##   ..   sentry_object = col_logical(),
##   ..   absolute_magnitude = col_double(),
##   ..   hazardous = col_logical()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Begin the cleaning process to prepare the data for the analysis:

```
#No unnecessary renaming of the columns
#Remove unnecessary columns: orbiting_body and sentry_object
data1 <- data1 %>%
  select(-c(id, orbiting_body, sentry_object))

# Reassign the true and false values into hazard and safe for better analysis
data1 <- data1 %>%
  mutate(hazardous = case_when(hazardous == TRUE ~ 'hazard', hazardous == FALSE ~ 'safe'))



#inspect the table once more
nrow(data1)
```

```
## [1] 90836
```

```
dim(data1)
```

```
## [1] 90836     7
```

```
summary(data1)
```

```
##      name           est_diameter_min   est_diameter_max   relative_velocity
##  Length:90836       Min.   : 0.00061   Min.   : 0.00136   Min.   :   203.3
##  Class :character   1st Qu.: 0.01926   1st Qu.: 0.04306   1st Qu.: 28619.0
##  Mode  :character   Median : 0.04837   Median : 0.10815   Median : 44190.1
##                     Mean   : 0.12743   Mean   : 0.28495   Mean   : 48066.9
##                     3rd Qu.: 0.14340   3rd Qu.: 0.32066   3rd Qu.: 62923.6
##                     Max.   :37.89265   Max.   :84.73054   Max.   :236990.1
##  miss_distance      absolute_magnitude  hazardous
##  Min.   :    6746   Min.   : 9.23      Length:90836
##  1st Qu.:17210820   1st Qu.:21.34      Class :character
##  Median :37846579   Median :23.70      Mode  :character
##  Mean   :37066546   Mean   :23.53
##  3rd Qu.:56548996   3rd Qu.:25.70
##  Max.   :74798651   Max.   :33.20
```

```
#drop any NULL values
data1 <- drop_na(data1)

#inspect the new table
nrow(data1)
```

```
## [1] 90836
```

```
dim(data1)
```

```
## [1] 90836     7
```

```
summary(data1)
```

```
##      name           est_diameter_min   est_diameter_max   relative_velocity
##  Length:90836       Min.   : 0.00061   Min.   : 0.00136   Min.   :   203.3
##  Class :character   1st Qu.: 0.01926   1st Qu.: 0.04306   1st Qu.: 28619.0
##  Mode  :character   Median : 0.04837   Median : 0.10815   Median : 44190.1
##                     Mean   : 0.12743   Mean   : 0.28495   Mean   : 48066.9
##                     3rd Qu.: 0.14340   3rd Qu.: 0.32066   3rd Qu.: 62923.6
##                     Max.   :37.89265   Max.   :84.73054   Max.   :236990.1
##  miss_distance      absolute_magnitude  hazardous
##  Min.   :    6746   Min.   : 9.23      Length:90836
##  1st Qu.:17210820   1st Qu.:21.34      Class :character
##  Median :37846579   Median :23.70      Mode  :character
##  Mean   :37066546   Mean   :23.53
##  3rd Qu.:56548996   3rd Qu.:25.70
##  Max.   :74798651   Max.   :33.20
```

Descriptive Data Analysis

Going over what each column means in this dat set:

*est_diameter_min is the minimum diameter estimated in kilometers

*est_diameter_min is the Maximum Estimated Diameter in Kilometers

*relative_velocity is the velocity that is relative to Earth

*miss_distance is the distance in Kilometers missed

*absolute_magnitude is the intrinsic luminosity which is the measure of brightness based on the distance of a star and the object.

Descriptive analysis on min and mean of all the values and save the values in a separate data frame for further reference:

```
#Descriptive analysis on est_diameter_min of all the values
mean(data1$est_diameter_min)
```

```
## [1] 0.1274321
```

```
median(data1$est_diameter_min)
```

```
## [1] 0.04836765
```

```
max(data1$est_diameter_min)
```

```
## [1] 37.89265
```

```
min(data1$est_diameter_min)
```

```
## [1] 0.0006089126
```

```
# You can condense the four lines above to one line
summary(data1$est_diameter_min)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00061  0.01926  0.04837  0.12743  0.14340 37.89265
```

```
#Descriptive analysis on est_diameter_max of all the values
mean(data1$est_diameter_max)
```

```
## [1] 0.2849469
```

```
median(data1$est_diameter_max)
```

```
## [1] 0.1081534
```

```
max(data1$est_diameter_max)
```

```
## [1] 84.73054
```

```
min(data1$est_diameter_max)
```

```
## [1] 0.00136157
```

```
#Condense the four lines above to one line
summary(data1$est_diameter_max)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00136  0.04306  0.10815  0.28495  0.32066 84.73054
```

```
#Descriptive analysis on relative_velocity of all the values
mean(data1$relative_velocity)
```

```
## [1] 48066.92
```

```
median(data1$relative_velocity)
```

```
## [1] 44190.12
```

```
max(data1$relative_velocity)
```

```
## [1] 236990.1
```

```
min(data1$relative_velocity)
```

```
## [1] 203.3464
```

```
#Condense the four lines above to one line
summary(data1$relative_velocity)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    203.3  28619.0  44190.1  48066.9  62923.6 236990.1
```

```
#Descriptive analysis on miss_distance of all the values
mean(data1$miss_distance)
```

```
## [1] 37066546
```

```
median(data1$miss_distance)
```

```
## [1] 37846579
```

```
max(data1$miss_distance)
```

```
## [1] 74798651
```

```
min(data1$miss_distance)
```

```
## [1] 6745.533
```

```
#Condense the four lines above to one line
summary(data1$miss_distance)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     6746 17210820 37846579 37066546 56548996 74798651
```

```
#Descriptive analysis on miss_distance of all the values
mean(data1$absolute_magnitude)
```

```
## [1] 23.5271
```

```
median(data1$absolute_magnitude)
```

```
## [1] 23.7
```

```
max(data1$absolute_magnitude)
```

```
## [1] 33.2
```

```
min(data1$absolute_magnitude)
```

```
## [1] 9.23
```

```
#Condense the four lines above to one line
summary(data1$absolute_magnitude)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    9.23   21.34   23.70   23.53   25.70   33.20
```

```
# Compare between the hazard and safe in ALL values
aggregate(data1$est_diameter_min ~ data1$hazardous, FUN = mean)
```

data1$hazardous	data1$est_diameter_min
<chr>	<dbl>
hazard	0.2941341
safe	0.1094599

2 rows

```
aggregate(data1$est_diameter_min ~ data1$hazardous, FUN = median)
```

data1$hazardous	data1$est_diameter_min
<chr>	<dbl>
hazard	0.20162992
safe	0.04023046

2 rows

```
aggregate(data1$est_diameter_min ~ data1$hazardous, FUN = max)
```

data1$hazardous	data1$est_diameter_min
<chr>	<dbl>
hazard	4.135757
safe	37.892650

2 rows

```
aggregate(data1$est_diameter_min ~ data1$hazardous, FUN = min)
```

data1$hazardous	data1$est_diameter_min
<chr>	<dbl>
hazard	0.0880146521
safe	0.0006089126

2 rows

```
aggregate(data1$est_diameter_max ~ data1$hazardous, FUN = mean)
```

data1$hazardous	data1$est_diameter_max
<chr>	<dbl>
hazard	0.6577038
safe	0.2447599

2 rows

```
aggregate(data1$est_diameter_max ~ data1$hazardous, FUN = median)
```

data1$hazardous	data1$est_diameter_max
<chr>	<dbl>
hazard	0.45085821
safe	0.08995804

2 rows

```
aggregate(data1$est_diameter_max ~ data1$hazardous, FUN = max)
```

data1$hazardous	data1$est_diameter_max
<chr>	<dbl>
hazard	9.247833
safe	84.730541

2 rows

```
aggregate(data1$est_diameter_max ~ data1$hazardous, FUN = min)
```

data1$hazardous	data1$est_diameter_max
<chr>	<dbl>
hazard	0.19680675
safe	0.00136157

2 rows

```
aggregate(data1$relative_velocity ~ data1$hazardous, FUN = mean)
```

data1$hazardous	data1$relative_velocity
<chr>	<dbl>
hazard	62794.34
safe	46479.15

2 rows

```
aggregate(data1$relative_velocity ~ data1$hazardous, FUN = median)
```

data1$hazardous	data1$relative_velocity
<chr>	<dbl>
hazard	58658.01
safe	42565.50

2 rows

```
aggregate(data1$relative_velocity ~ data1$hazardous, FUN = max)
```

data1$hazardous	data1$relative_velocity
<chr>	<dbl>
hazard	193387.0
safe	236990.1

2 rows

```
aggregate(data1$relative_velocity ~ data1$hazardous, FUN = min)
```

data1$hazardous	data1$relative_velocity
<chr>	<dbl>
hazard	5908.2918
safe	203.3464

2 rows

```
aggregate(data1$miss_distance ~ data1$hazardous, FUN = mean)
```

data1$hazardous	data1$miss_distance
<chr>	<dbl>
hazard	39946230
safe	36756087

2 rows

```
aggregate(data1$miss_distance ~ data1$hazardous, FUN = median)
```

data1$hazardous	data1$miss_distance
<chr>	<dbl>
hazard	40983721
safe	37487452

2 rows

```
aggregate(data1$miss_distance ~ data1$hazardous, FUN = max)
```

data1$hazardous	data1$miss_distance
<chr>	<dbl>
hazard	74790953
safe	74798651

2 rows

```
aggregate(data1$miss_distance ~ data1$hazardous, FUN = min)
```

data1$hazardous	data1$miss_distance
<chr>	<dbl>
hazard	143272.707
safe	6745.533

2 rows

```
aggregate(data1$absolute_magnitude ~ data1$hazardous, FUN = mean)
```

data1$hazardous	data1$absolute_magnitude
<chr>	<dbl>
hazard	20.3076
safe	23.8742

2 rows

```
aggregate(data1$absolute_magnitude ~ data1$hazardous, FUN = median)
```

data1$hazardous	data1$absolute_magnitude
<chr>	<dbl>
hazard	20.6
safe	24.1

2 rows

```
aggregate(data1$absolute_magnitude ~ data1$hazardous, FUN = max)
```

data1$hazardous	data1$absolute_magnitude
<chr>	<dbl>
hazard	22.4
safe	33.2

2 rows

```
aggregate(data1$absolute_magnitude ~ data1$hazardous, FUN = min)
```

data1$hazardous	data1$absolute_magnitude
<chr>	<dbl>
hazard	14.04
safe	9.23

2 rows

The information we got from the aggregated descriptive data analysis:

*Mean: From this, it is shown that the Hazard values are typically higher than the safe values except for the absolute magnitude.

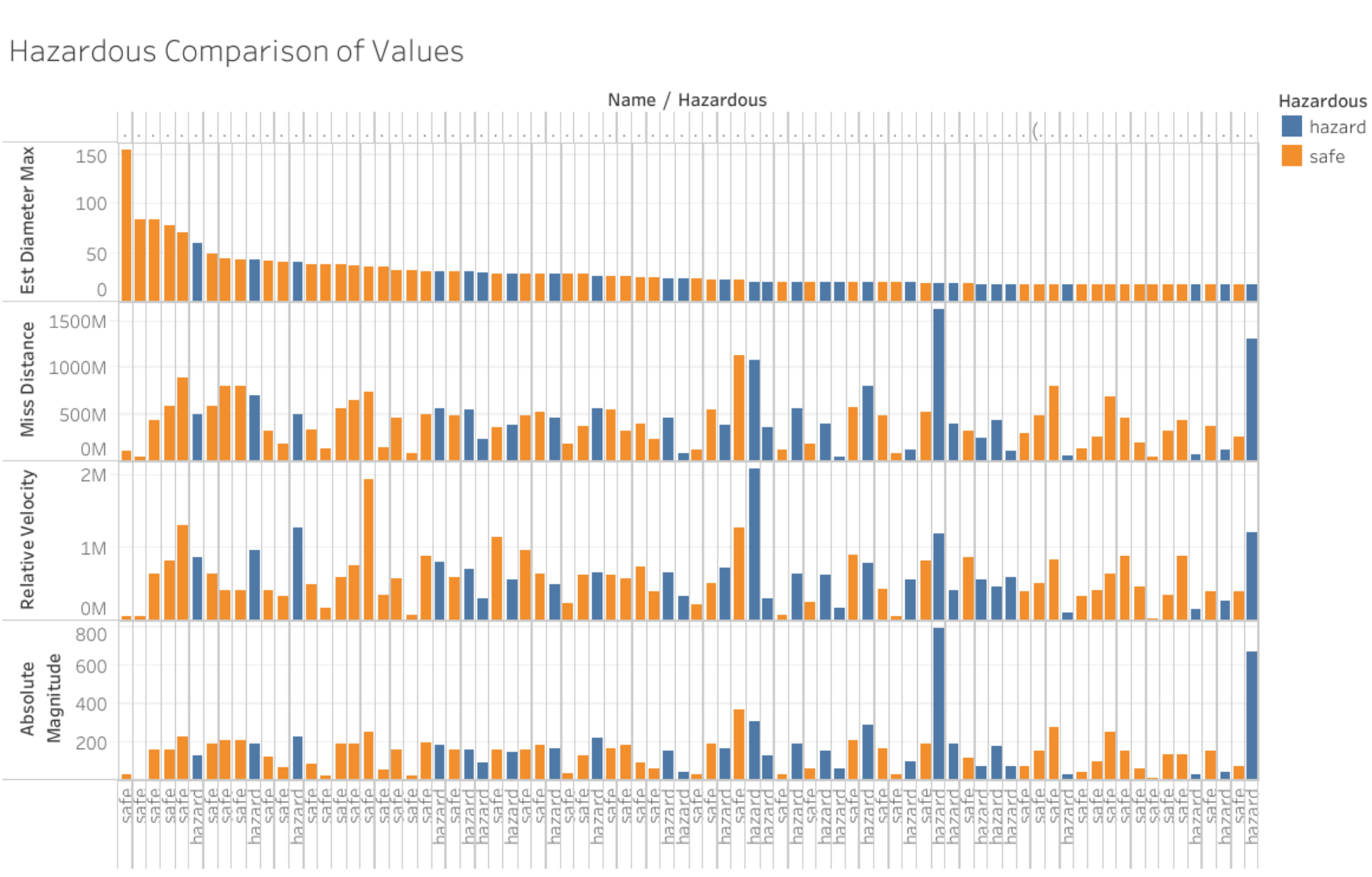
*Median: It is the same for the mean where the Hazard values are higher than the safe values except for the absolute magnitude.

*Max: This time it is shown that all the max values for the safe values are all higher than the hazard values.

*Min: From this, it is shown that the minimum values for the Hazard values are all higher than the safe values.

Visualiztion

Taking the aggregated data and the cleaned up data, tableau was used to visualize the data as a bar graph.



Hazard Data Comparison

This may seem overwhelming but that is the point of this visual. What is being shown here is how all over the place the data is when comparing them to each other. This is showing that there is no concrete combo of values that determines how an object is haphazardness or not. It also shows that there is no concrete correlation with each other.

Data Conclusions

*Putting the data through the visual mediums of a bar graph have shown that it is difficult to find combination of data that indicates whether they are hazardous or not. It can be seen that some hazard and safe objects have value that are close in numbers, making it not clear on the difference between the hazard and safe objects.

*The aggregations of each value from the descriptive data analysis portion does give a better idea of what makes an object hazardous. From the average and median, it can be seen that the hazard objects are typically larger, showing that distance, diameter, and velocity are important to what makes a object hazardous. The average and median also shares similarity where the absolute magnitude of the hazard objects are typically smaller than the safe objects.

*In the Maximum and Minimum portion of the data, it can be seen that the hazard values are mostly lower than the safe values. It is the reverse with he minimum values where the hazard value is higher than the safe value. Showing that the Hazard object values tend to have a smaller range of numbers than the safe objects.

Recommendations

With the data collected, we can conclude that the observers of celestial objects should keep watch of the diameter, velocity, distance, and the absolute magnitude. The higher the values are in the diameter, velocity, and distance, the more hazardous the objects can be. This combined with checking the absolute magnitude where the lower values is more common with a hazardous object.

Since the commonality of hazardous objects is the diameter, distance, and velocity, it is recommended to get devices that can not only take a clear image of the object but also be able to measure the distance of the object. Along with that, it should also take several pictures and record the time the pictures were taken since the velocity of the object can be more accurately calculated with these values.

Additional Data/Expansion

It would have been helpful fi there was a way to figure out the mass of the object. This would have given us another value that to look out for and see if that can have an effect in determining if an object can be hazardous. This could also give us more accurate values in figuring out the objects velocity and whether it is being affected by the other solar masses in our solar system. Though I can see how difficult that can be if there is no way to get a sample of the object.