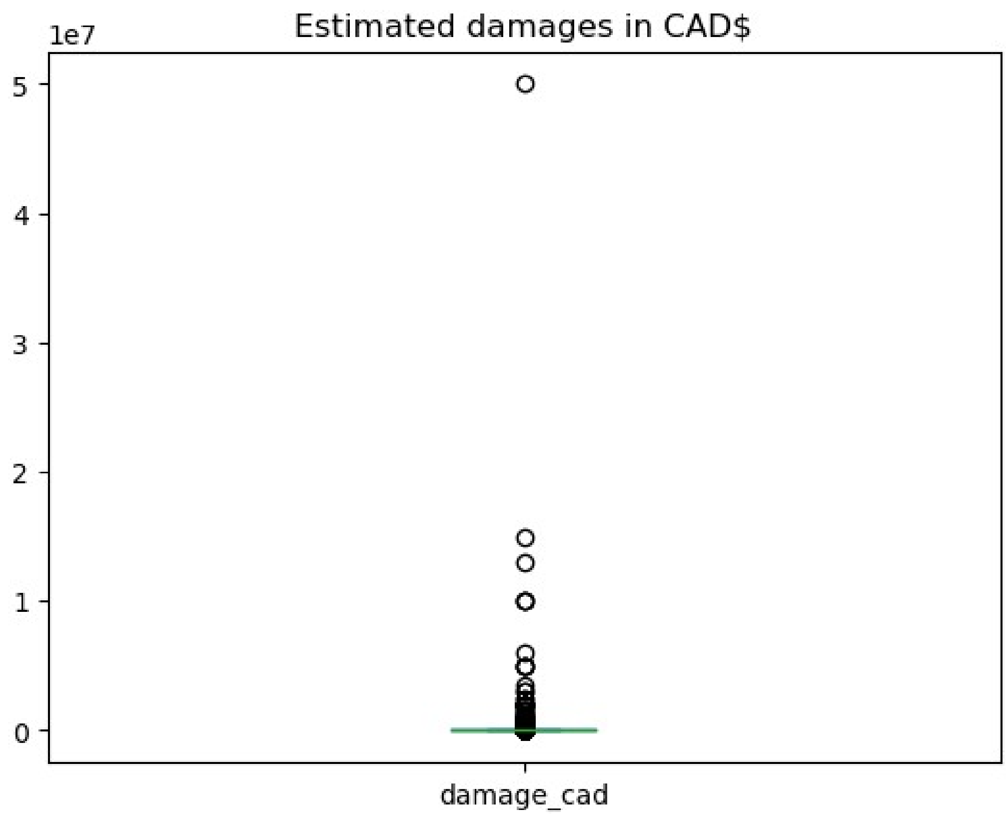
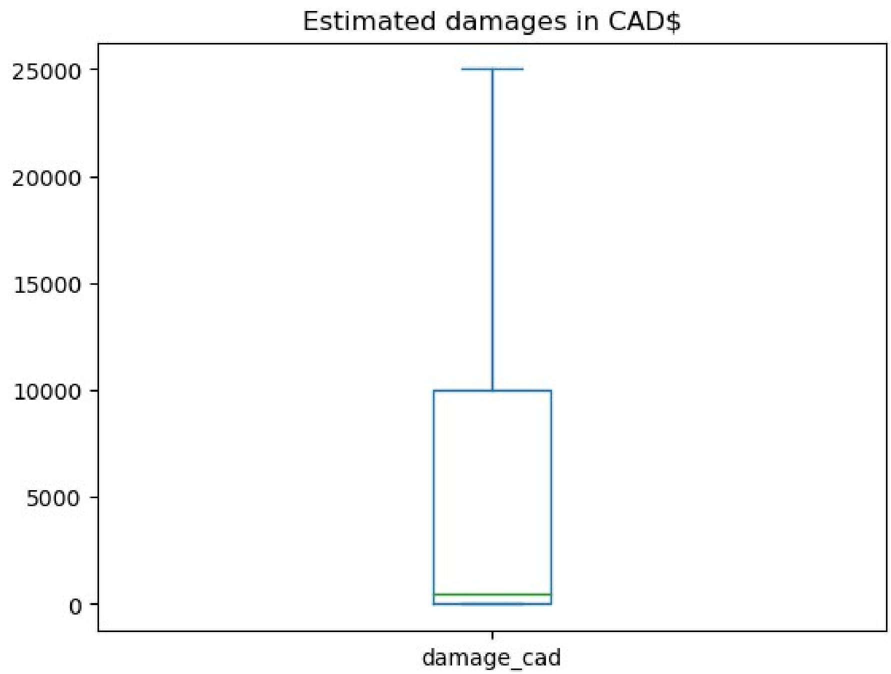
Data Summarization

We had an idea that we wanted to classify the outcomes of the fires as either good or bad. Depending on the number of casualties, the number of people displaced and the estimated damages in CAD$ so those were the attributes we evaluated.

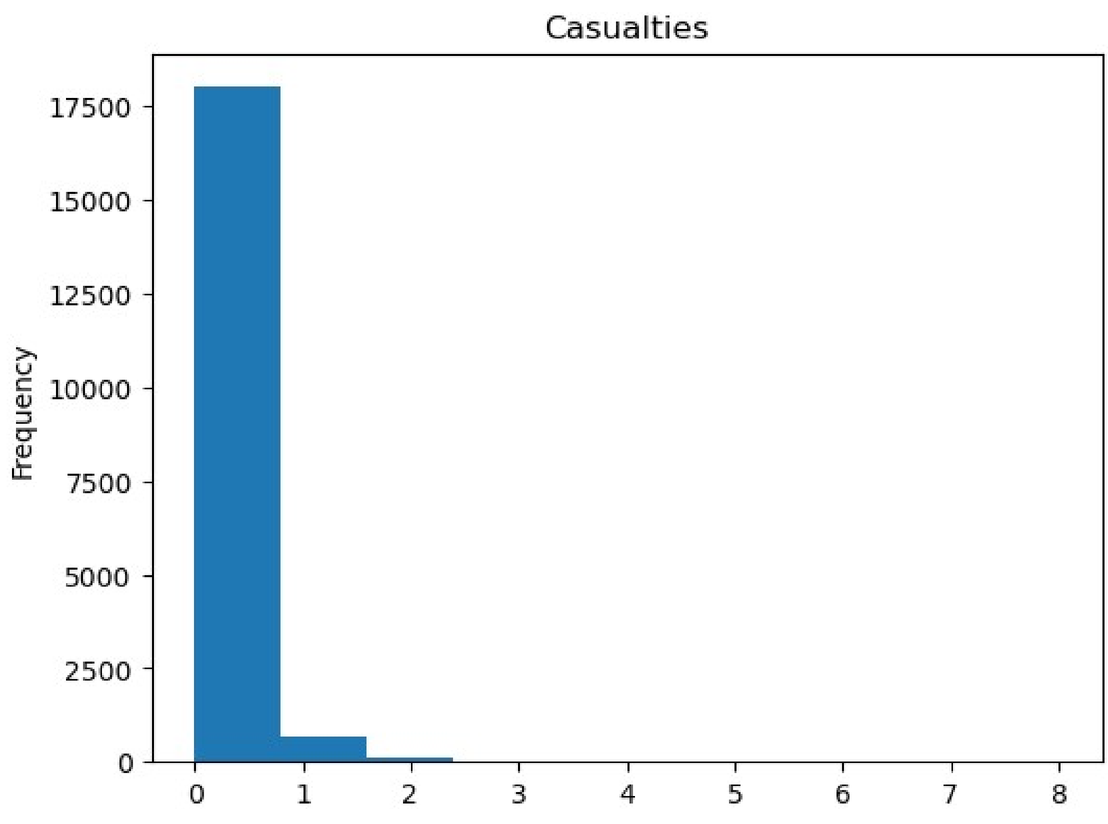
We began by looking at the estimated damages in CAD dollars, we chose to use a boxplot, as can be seen in the following graph



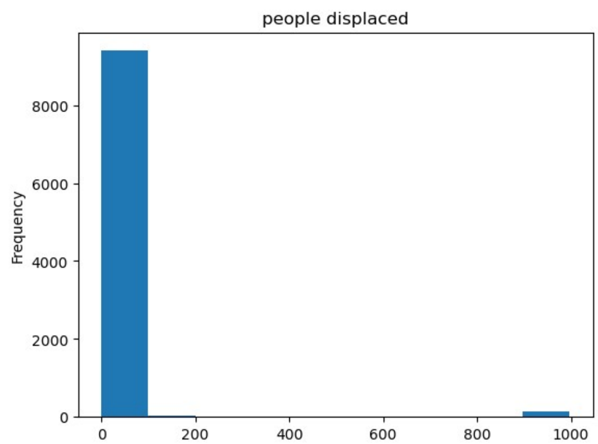
However the outliers prevent the values in the boxplot from being distinguishable so, a version without the dots representing the outliers was created to get a better idea of scale, as can be seen here, the first quantile consists of estimated damages being equal to 0$:

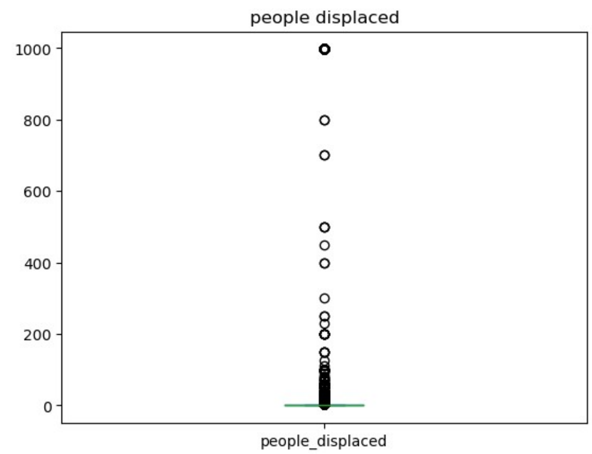


Next, we looked at the casualties and found that the vast majority of the data fell at 0 casualties, skewing us towards good outcomes. As is shown in the following graph:

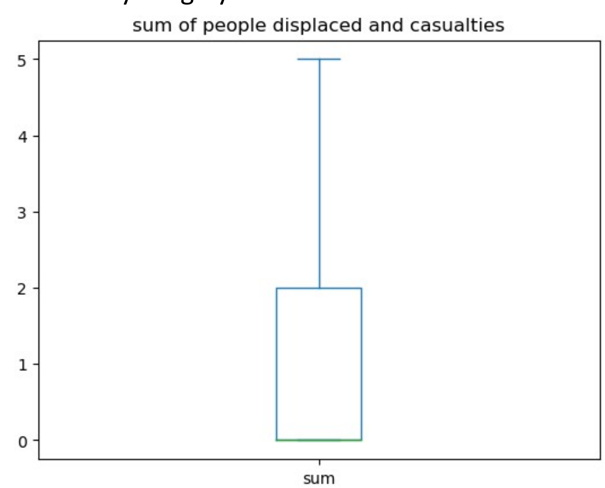


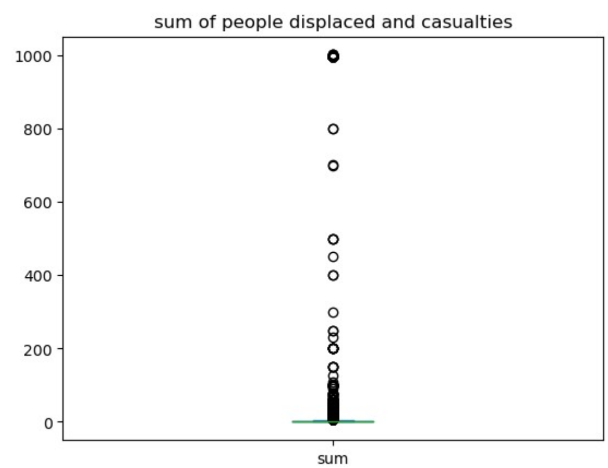
Similar findings were observed for the number of people displaced, where the majority of cases have 0 people displaced but here the outliers are much more drastic, as can be observed in the histogram and boxplot below:

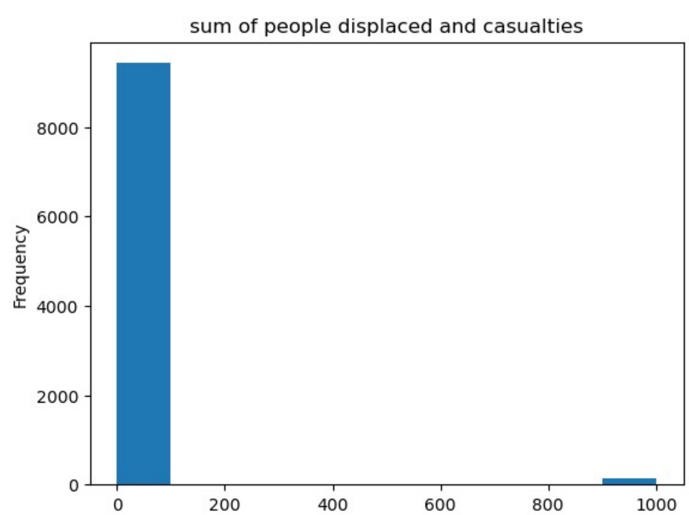




Upon finding that both of the above typically have low values, we wanted to see if they typically matched, that is to say that having 0 casualties usually meant having 0 people displaced, so we plotted the sum of the two measures to see if we would have more numbers above 0 or if the sum quantity of 0s would stay roughly consistent. The results can be found in the figures below:







<u>Feature Selection</u>

The features that we selected for our classification algorithms included:

Dimension	**Features**	**Notes**
Location dimension	Longitude Latitude	This dimension contained other data such as the postal code, nearest intersection, and dissemination area id. We decided to use only the latitude and longitude because they were the most precise numerical values to use in the AI and the other location data would have been redundant anyway.
Date dimension	Month Day Day of week Holiday Time of day	This dimension contained other data as well such as the exact timestamp of the previous fires. Since those exact timestamps will never occur again, they are likely to overfit the AI to the training data. Instead, we only used the date features that are repeatable.
Weather dimension	Temperature Relative humidity Precipitation Snow Wind direction Wind speed	We used almost all the features from this dimension because they could all potentially have a huge impact on how frequently fires can start and how damaging they may be.
Demographic dimension	Population Median age Total dwellings Average household size Median household income Mother tongue official percentage Mother tongue unofficial percentage	We used almost all the features from this dimension because, again, they could all potentially have an impact on the frequency and damage of fires.
Fire ward dimension	Stations in ward	This is the only feature in this dimension and it could have an affect on the response time to fires, thus limiting the damage.

We left out all the measures from the fact table itself because most of them were measures that could only be determined during or after a fire. Since the purpose of our AI is to predict if some conditions could result in a ‘bad’ or ‘good’ fire, we can’t use any variables that occur during or after the fire to predict this. Instead, we categorized all the fires as bad or good based on the number of casualties, the number of people displaced, the damage in Canadian dollars, and the response time. A fire was considered bad if the total number of casualties plus the number of people displaced was 5 or more, the damage was $10,000 or more, or the response time was more than 20 minutes. A fire was considered 'good' or 'acceptable' otherwise.

In addition, we performed under-sampling of the bad outcomes because there was about a 3:1 ratio between the bad and good outcomes. We reduced the number of bad outcome samples to equal those of the good outcomes.

<u>Imputation</u>

Out of the features that we chose, there was only one that had any missing values. The holiday feature was mostly missing because only a small percentage of the fires in the data occurred on holidays. Since over 95% of the data was missing for that feature, we created a new value for it called 'Not holiday' and filled those missing values. The figure below shows the number of missing values in each column before imputation.

```
longitude                               0
latitude                                0
month                                   0
day                                     0
day_of_week                             0
holiday                             18403
time_of_day                             0
temperature                             0
relative_humidity                       0
precipitaion                            0
snow                                    0
wind_direction                          0
wind_speed                              0
population                              0
median_age                              0
total_dwellings                         0
average_household_size                  0
median_household_income                 0
mother_tongue_official_percentage       0
mother_tongue_unofficial_percentage     0
stations_in_ward                        0
status                                  0
```

<u>Handling categorical attributes</u>

When viewing the features, most of them were numerical, but holiday and time of day where categorical. Holiday contained the name of the holiday that was occurring and the time of day could have been night (midnight to 6am), morning (6am to noon), afternoon (noon to 6pm), or evening (6pm to midnight). We used one-hot encoding to split both of those into multiple columns with values of 0 or 1. This next figure is the type of data that is held by each column before the one-hot encoding.

```
longitude                                 float64
latitude                                  float64
month                                       int64
day                                         int64
day_of_week                                 int64
holiday                                    object
time_of_day                                object
temperature                               float64
relative_humidity                         float64
precipitaion                              float64
snow                                      float64
wind_direction                            float64
wind_speed                                float64
population                                  int64
median_age                                float64
total_dwellings                           float64
average_household_size                    float64
median_household_income                   float64
mother_tongue_official_percentage         float64
mother_tongue_unofficial_percentage       float64
stations_in_ward                            int64
status                                      int64
```

The following figure is the type of data that is help by each column after the one-hot encoding.

```
longitude                                 float64
latitude                                  float64
month                                       int64
day                                         int64
day_of_week                                 int64
temperature                               float64
relative_humidity                         float64
precipitaion                              float64
snow                                      float64
wind_direction                            float64
wind_speed                                float64
population                                  int64
median_age                                float64
total_dwellings                           float64
average_household_size                    float64
median_household_income                   float64
mother_tongue_official_percentage         float64
mother_tongue_unofficial_percentage       float64
stations_in_ward                            int64
status                                      int64
holiday_Boxing day                          uint8
holiday_Canada day                          uint8
holiday_Christmas day                       uint8
holiday_Civic holiday                       uint8
holiday_Easter sunday                       uint8
holiday_Family day                          uint8
holiday_Good friday                         uint8
holiday_Halloween                           uint8
holiday_Labour day                          uint8
holiday_New year's day                      uint8
holiday_Not holiday                         uint8
holiday_St. Patricks day                    uint8
holiday_Thanksgiving                        uint8
holiday_Valentines day                      uint8
time_of_day_afternoon                       uint8
time_of_day_evening                         uint8
time_of_day_morning                         uint8
time_of_day_night                           uint8
```

Normalization

We normalized all the columns independently so they would be worth the same to the classification algorithms later. The following image is a sample of a few of the features after normalizing.

	longitude	latitude	month	day	day_of_week
count	11074.000000	11074.000000	11074.000000	11074.000000	11074.000000
mean	0.460070	0.440059	0.500304	0.489682	0.508774
std	0.201064	0.198740	0.290941	0.291790	0.335794
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.311009	0.273810	0.272727	0.233333	0.166667
50%	0.459480	0.414150	0.454545	0.500000	0.500000
75%	0.590710	0.605374	0.727273	0.733333	0.833333
max	1.000000	1.000000	1.000000	1.000000	1.000000

Classification algorithms

After running the algorithms multiple times with different data subsets, we noticed some patterns emerging. The decision tree was always the fastest to train, followed by the random forest. The gradient boosting was always the slowest. By contrast, the accuracy was always the opposite. The gradient boosting was always the most accurate while the decision tree was the least accurate. The following three images show a classification report for each of the three different AI types.

```
****************************************
*                                      *
* Decision tree - Classification report *
*                                      *
****************************************
              precision    recall  f1-score   support

         Bad       0.52      0.53      0.53      1389
        Good       0.52      0.51      0.52      1380

    accuracy                           0.52      2769
   macro avg       0.52      0.52      0.52      2769
weighted avg       0.52      0.52      0.52      2769


The decision tree took 0.14899659156799316 seconds to train
*******************************************
*                                         *
* Gradient boosting - Classification report *
*                                         *
*******************************************
              precision    recall  f1-score   support

         Bad       0.43      0.64      0.51       947
        Good       0.75      0.56      0.64      1822

    accuracy                           0.59      2769
   macro avg       0.59      0.60      0.58      2769
weighted avg       0.64      0.59      0.60      2769


The gradient boosting took 2.4593307971954346 seconds to train
```

```
****************************************
*                                      *
* Random forest - Classification report *
*                                      *
****************************************
              precision    recall  f1-score   support

         Bad       0.51      0.57      0.54      1255
        Good       0.61      0.55      0.57      1514

    accuracy                           0.56      2769
   macro avg       0.56      0.56      0.56      2769
weighted avg       0.56      0.56      0.56      2769


The random forest took 2.31719708442688 seconds to train
```

Data Nuggets

After training the AI on the testing dataset, test cases were created to see if changes in a single attribute could affect the severity of a fire. All other attribute values would take on the average of its values to represent the average day. We also set the holiday to none and the time of day to afternoon.

Here are the results:

Temperature	Decision tree	Gradient boosting	Random forest
-25	Bad	Good	Good
2.8	Bad	Good	Good
11.9	Good	Good	Bad
20.7	Good	Good	Good
36	Good	Good	Bad

Income	Decision tree	Gradient boosting	Random forest
12176	Good	Good	Bad
38896	Bad	Good	Bad
73500	Good	Good	Bad
95000	Good	Good	Good
476000	Good	Bad	Bad

Dwelling population	Decision tree	Gradient boosting	Random forest
55	Good	Good	Bad
185	Good	Good	Bad
270	Good	Good	Good
580	Good	Good	Good
5895	Good	Good	Good

Relative Humidity	Decision tree	Gradient boosting	Random forest
0	Good	Good	Good
50	Good	Good	Bad
63	Good	Good	Bad
76	Good	Good	Bad
100	Good	Good	Bad

We found that all three models agreed that the outcome of the fire would be good if the temperature was 20.7 degrees. This made sense because we defined the outcome of a fire to be good if the number of people displaced and injured was smaller than 5, proprietal damage was less than $10,000 and the response time was less than 20 minutes. Warm temperature during the afternoon is a good indication of good weather, and if the weather is good, then the roads are clear and firetrucks can drive at a high speed.

We also found that if the ideal income for the least fire damage was around $95,000 according to all three models. This made sense because a household with very high income is likely to spend more on their homes and even a small fire could cause a heavy monetary loss.
It also made sense that households with a lower income may be more susceptible to fire hazards because the lower salary could be an indication of retirees living there. Older people are generally much slower to react to fires than younger people.

Lastly, the three models predicted that a fire that occurred in a dwelling population of above 270 would not be devastating. This makes a lot of sense because high population numbers indicates larger buildings and they are more likely to be at the heart of the city, where firetrucks can spot them and reach them easily. Lower dwelling population may indicate people living in much smaller homes and further away from the city, making it much harder for firefighters to find the fire and reach the place on time.