**Introduction**

A new research field emerged in climate science in the early 2000s that wanted to explore the increasing prevalence of extreme weather events like floods, storms, cyclones, etc. The field is known as "extreme event attribution" and has gained momentum in recent years in media in addition to the scientific world. There is mounting evidence that human activity is to blame for the increased risk of these extreme weather-type events. Researchers have also given importance to analyzing the economic costs linked to the human contribution to weather events. A study in 2020 approximated that nearly $67bn of damages caused by Hurricane Harvey in 2017 could attribute to human influences on climate. There are numerous methods to carry out attribution analysis. One way is to record instances of an extreme weather event and see their frequencies change with changes in environmental factors. We aim to build a model that accurately predicts the estimated damage to property while considering various event-related factors, in addition to external factors that might be influencing the extent of the damage.

**Dataset**

For this project, we have used publicly available data from the National Oceanic and Atmospheric Administration (NOAA) that contains event details on disaster incidents occurring in the US ranging from 1950 to August 2021. Some of the variables that we use from this dataset are as follows:

* begin and end date-time of event
* state where the event occurred
* the type of event (Hail, Storm, Drought, etc.)
* number of injuries and deaths
* starting and ending latitudes and longitudes of the event

The complete data dictionary for reference is accessible through [this link](https://www.ncei.noaa.gov/pub/data/swdi/stormevents/csvfiles/Storm-Data-Bulk-csv-Format.pdf).

We have also pulled in environmental indicators from yearly data collected by the United States Environmental Protection Agency (EPA). We have joined this data as additional information against the event year. The datasets that we have considered from the EPA source are as follows:

* emissions of greenhouse gases from 1990 to 2019
* events of heavy precipitation by land area percentage
* yearly earth surface temperature
* CSIRO and NOAA data for yearly sea-level changes
* variations in average seasonal temperature for fall, winter, summer, and spring
* artic ice coverage in March (yearly high) and September (yearly low)
* Glacier mass balance and number of observed glaciers

Additional dataset information is available at [this source](https://www.epa.gov/climate-indicators).

**Algorithms Used:**

Linear Regression:

It is a method to model a relationship between one or more independent variables and a response variable by fitting a linear equation on the observed data. Regression tells us the value of the response variable for an arbitrary explanatory variable value. The regression equation is:

ŷ = *b*0 + *b*1 *x*1 + *b*2 *x*2 + … where

b0: intercept

b*i*: slope/rate of change

Bootstrap aggregation:

Bootstrapping is a sampling technique to create subsets of observations from the original data and is also known as bagging. In this technique, a generalized result combines the results of various predictive models. The subset size for bagging may be smaller than the original dataset.

Random Forest Regression:

It is a supervised machine learning algorithm that uses bagging to solve regression and classification problems. The algorithm works by training multiple decision tree estimators concurrently and outputting the mean or mode of all the individual predictions. It helps against individual trees overfitting the data and getting stuck in locally optimal solutions.

Diagram

Description automatically generated

Random Forest prediction working [3]

## Extreme Gradient Boosting Regression:

Gradient boosting is a class of ensemble machine learning algorithms constructed from decision tree models. It fits the model using any arbitrary differentiable loss function and gradient descent optimization algorithm. This technique is known as gradient boosting as we minimize the loss gradient while training the model.

Extreme Gradient Boosting, or XGBoost for short, is an efficient open-source implementation of the gradient boosting algorithm. XGBoost is a powerful approach for building supervised regression models.

# Experimental Setup:

## Extraction:

The NOAA data files are extracted from this link and have the following naming structure: StormEvents\_details-ftp\_v1.0\_d1950\_c20210803.csv.gz.

The files are then concatenated into one to form our source data frame. We save this as a pickle file.

## Preprocessing:

* The replace\_str2num() function cleans up the DAMAGE\_CROPS and DAMAGE\_PROPERTY variables to convert them into numeric values.
* The winds() and hail() functions split the MAGNITUDE variable based on the values of MAGNITUDE\_TYPE into WIND\_SPEED and HAIL\_SIZE.
* The missing\_swap() function imputes missing values for variables where the counterpart has valid values. For example, if the BEGIN\_LAT is present and the END\_LAT is not present, we fill it with the BEGIN\_LAT value.
* The calc\_duration() function calculates the time difference between the event start and end.
* The geo\_distance() function uses the Haversine formula to calculate the geographical distance covered by the event (Tornado, etc.)
* The dict\_mapping() function replaces junk values from CZ\_TIMEZONE, BEGIN\_AZIMUTH, and END\_AZIMUTH with appropriate values.
* We use the EVENT\_TYPE variable to derive three variables: COLD\_WEATHER\_EVENT, WINDY\_EVENT, and WATER\_EVENT, based on keywords like Snow, Storm, Hurricane, etc.
* Then, we fill the missing values in continuous variables with 0 and the categorical variables by N/A.
* We use Pandas to read the various EPA data CSV files and collate them into one. We interpolate the missing data ranging back to the year 1950 by using the impute\_EPA\_data() function. We use the interp1d method from SciPy to get the extrapolated variable values.
* Finally, we join the entire data into one data frame and remove the outliers from all the numerical variables.

Modeling:

We encode the categorical variables with many unique values using the mapping() function on the range of distinct values. We use the get\_dummies() method to split the other categorical variables. Our response variable TOTAL\_DAMAGE is the sum of the DAMAGE\_PROPERTY and DAMAGE\_CROPS.

We train and compare the efficiencies of four different models. Here is the information about these models, along with their training parameters.

1. Linear Regression with default parameters
2. Random Forest Regressor with parameters as follows:
   * 1. n\_estimators=100
     2. oob\_score='TRUE'
     3. n\_jobs=-1
     4. random\_state=50
     5. max\_features="auto"
     6. min\_samples\_leaf=50
3. Extreme Gradient Boosting Regressor with parameters as follows:
   * 1. learning\_rate=0.01
     2. subsample=0.7
     3. max\_depth=5
     4. n\_estimators=100
     5. colsample\_bytree=0.8
4. Ensemble model which is a combination of above three models using VotingRegressor

We compare the performance of the different models by making use of the following metrics:

1. Mean squared error
2. Train R-squared value
3. Test R-squared value

**References**

1. https://www.carbonbrief.org/mapped-how-climate-change-affects-extreme-weather-around-the-world
2. https://towardsdatascience.com/a-quick-and-dirty-guide-to-random-forest-regression-52ca0af157f8
3. https://www.oreilly.com/library/view/tensorflow-machine-learning/9781789132212/d3d388ea-3e0b-4095-b01e-a0fe8cb3e575.xhtml