Overview:

The human society has been battling poverty for centuries. Countless politicians and social activists have tried to end poverty, but it still remains as the number one problem today. In Google and the UN's " 2030 agenda", it has listed "no poverty" as one of their most important goals. In this project, we will use machine learning to explore the correlations between poverty and various societal and geographical attributes such as consumption and climate types. Ideally, from the data and results from the machine learning model, we will be able to find which of those attributes, either societal or geographical, are most strongly correlated to high poverty rates, and hopefully it will provide us a unique perspective to identify the critical problems to tackle in order to solve the world's poverty problem.

We will be using three main algorithms in this projects, which are decision tree, logistic regression and neural network. Since we are trying to explore the correlation between each attribute and poverty level, we will run the three algorithms iteratively, each time removing a different attribute. Then we compare the accuracies during each run, and identify which attribute is removed when the algorithm reaches highest accuracy. Then that attribute is permanently removed, and we repeat the same process for the remaining dataset.

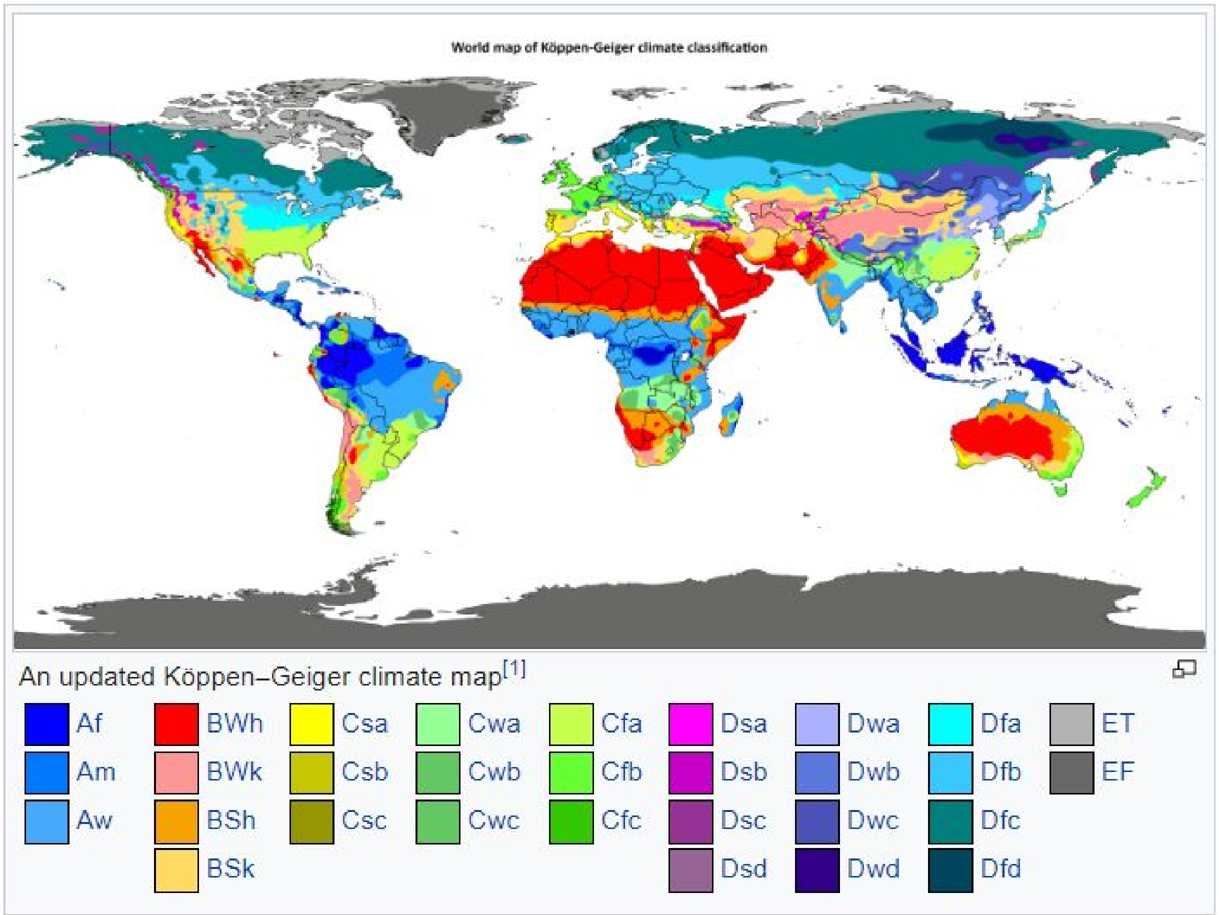
Dataset:

There are three sources that we obtain our data from. All the geographical data are downloaded from Kaggle and Nation Master. All the societal attributes are acquired from the Google Visualize 2030 Challenge. However, in order to select the most relevant data, we need to filter out a few irrelevant fields. For example, the "education level" category are further separated into different subcategories, such as the literacy rate of children under the age of 10, the literacy rate of children under the age of 18, and the literacy rate of all adults. Besides age group, the data also has subcategories in terms of gender, ethnicity, and etc. However, we only care about the overall literacy rate of the entire country, so we need to compile the data from different sub-categories, and compute the average literacy rate of a given country. Some preprocessing of data are performed within Excel, and Google Studio, a data visualization platform. Finally, we wrote a Python code to concatenate all attributes into one Excel spreadsheet. In order to do that, we first matched and converted all countries' names with their unique 3-letter country code based on an online library, and created a list of dictionaries where keys are the country code, and values are all the attribute values of that country. The code would read the data from all CSV files iteratively, add the values into corresponding countries, and output one single CSV file containing all information.

We further discretize the poverty levels into upper half and lower half categories based on the median poverty rate, so that they can be used to train decision tree and logistic and regression models.

For the neural network algorithm, we need to convert all entries into "tensor" objects in order to be able to use the Tensorflow library. Fortunately, tensorflow comes with a method called "TextFileReader" for reading CSV files. Then we need to call "tf.decode_CSV()", and provide default values for the inputs of each column. Finally, we use tf.stack to pack a list of tensors into one tensor.

Climate:



An updated Köppen–Geiger climate map[1]

https://github.com/thomasyang1207/PredictFuture
Methodology:

We first merged 3 separate datasets containing geographical, climate, and economic information about UN-recognized countries in the world. Since the datasets were from multiple sources, we needed to unify the country names by converting all the names into their ISO alpha3 country codes. Methodology can be found in merge.py in the “scripts” section of the repository.

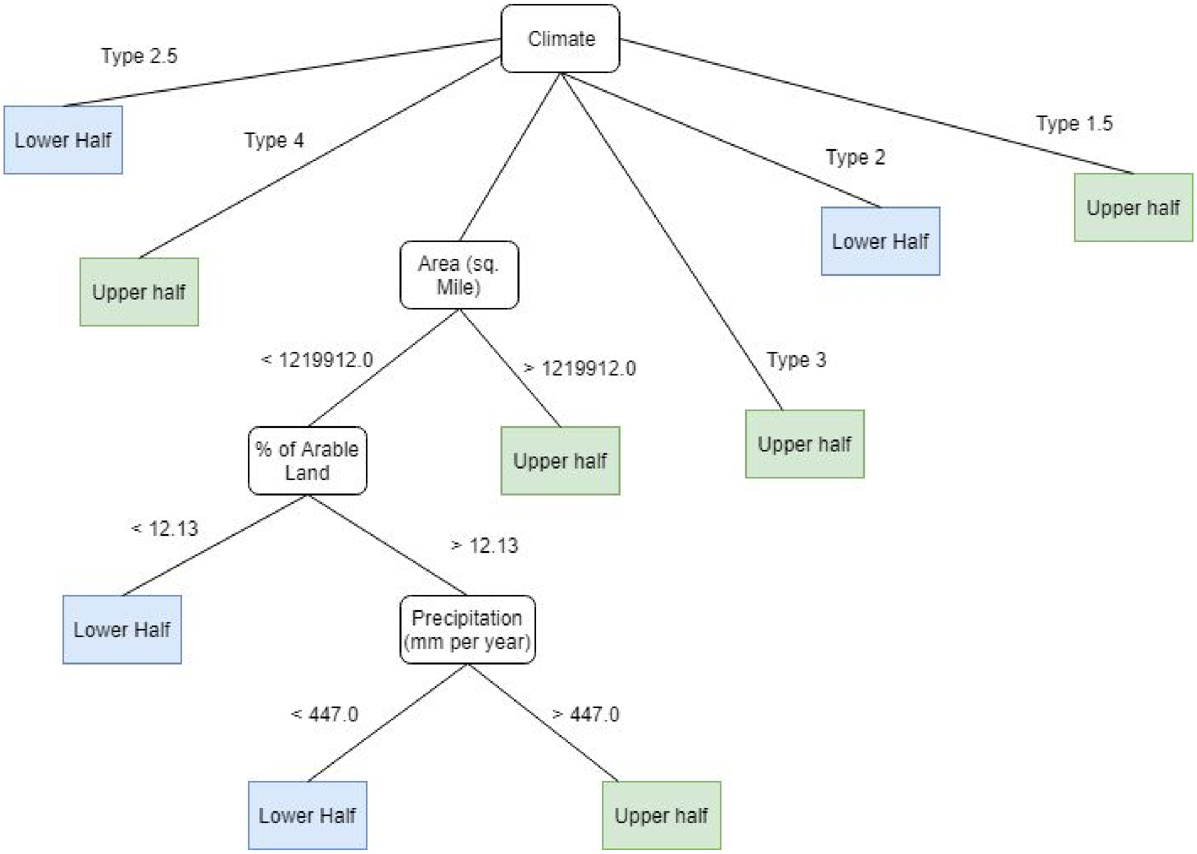
For this project, we decided to partition all countries based on their poverty rate into two groups, separated by the median poverty rate (14.4 in this case). Countries with poverty rates over 14.4 went into the “lower half” in terms of economic prosperity, and countries with poverty rates under 14.4 went into the “upper half”. While this method of division does not tell us the distribution of poverty rates, it serves as a good starting point for output classes.

We built an implementation of ID3 that accepted continuous attributes from scratch (see repository). If an attribute is discrete, we calculate the entropy based on the entropy gain classifying into each value. If an attribute is continuous, we calculate a “best split” value. Examples with attribute values less that the split were classified into the “lesser” group, and those with greater values were classified into the “greater” group. This “best split” value was determined iteratively, and the split value that resulted in the greatest post-classification entropy gain was chosen. Since most of our attributes were continuous, this method served as a good heuristic. Methods that produced more than two partitions would have contributed significantly to training time.

After training a tree, reduced error pruning was performed. The accuracy was then calculated through averaging the accuracies of 100 trials, where in each trial 75 of the dataset was used for training/pruning, and 25 were used for testing. To remove unneeded attributes, we eventually limited the depth of the tree so that noisy attributes did not contribute significantly. We also manually experimented with which attribute resulted in the greatest test accuracy over 100 trials.

Results

After repeated trials, and trying to predict multiple markers of economic prosperity (poverty rate, annual gdp growth, consumption), we found that the decision tree that resulted in the highest test accuracy was one that predicted a country’s poverty rate relative to the rest of the world based on 7 geographical and climate attributes: ['Precipitation', 'Climate', 'Arable ', 'Population', 'Area sq mi', 'Pop Density per sq mi', 'CoastlineLength']. The decision tree shown below was part of a set of **100 trials that achieved on average 70% accuracy when trained on 75% and tested on 25% of the dataset.**



Other decision trees that tried to predict the annual growth rate and consumption resulted in at best 60% accuracy. We will focus on the experiment that produced the above tree in our analysis section.

We also experimented the logistic regression model within Weka using different parameters, and the best accuracy we got is 65.9% when using 10-fold cross-validation.

Analysis of results

From our prediction of the poverty rate, the type of climate was always the root node. In many trees, after pruning was performed, most examples could be classified after looking at just the type of climate. Based on the climate data, most countries with climates similar to type 2 (wet tropical climates), tended to have relatively higher poverty rates compared to the rest of the world. Countries with climates above type 2 have relatively lower poverty rates. However, the trend is not linear. We see a drop in the poverty rates when observing type 1 and type 1.5 (hybrid of 1 and 2) climates. It is more surprising that the definition of the climate types in the original dataset states that type 1 climates tended to be less suited for agriculture.

It is also interesting to note that within countries with type 1 climates, larger countries tended to have lower poverty rates. This could be attributed to the fact that many of the largest countries in area in the world (Russia and Canada) have extremely low population density, leading to a lack of need for agriculture, and a greater focus on the control of natural resources. Below the area size, for the smaller countries, the arability of land became important. Countries that were small in size, with less arable land, tended to have higher poverty rates. This is reasonable, as countries that have neither natural resources nor arable land tend to run into food shortages and low economic growth.

These results, however, are not completely conclusive. First, there are a limited number of countries in the world. We may not have an adequately large dataset to produce a statistically significant result from machine learning methods for predicting nation-level poverty rates. However, the same decision tree classification could be applied to cities above a certain population, which could potentially lead to more accurate and interesting conclusions. Second, our augmented ID3 implementation does not consider partitions of continuous attributes greater than 2. It is possible that multiple partitions could be made at one node, multiple partitions of the same attribute could occur at multiple nodes. The method thus runs into issues with discretization. Finally, the usage of the “climate” attribute is ambiguous. The dataset where the “climate” classification originated from had already assessed each climate type defined in the Koppen scale with how it related to agriculture. While this led to a good accuracy in our model, a more interesting task would have involved using the original Koppen scale.

Future Plan:

Unfortunately, due to limited time and the complexity of Tensorflow API, we are not able to build a fully-functioning neural net. If time permits, we will run 50 epochs on the training and validation set, and visualize accuracy and loss over time on TensorBoard.