Milestone 1 - Report (amosl - 301315811 / Timothy 301307082 / Bianca B 301304375)

1.1 Exploratory Data Analysis

In 1.1, we performed an exploratory data analysis to get a better understanding of the training (cases_train.csv) as well as the location dataset (location.csv). We created various visualizations to model attribute distribution based on the data's value and type. For example, we plotted geographic data(the combination of longitude and latitude) on a world map to visualize the world-wide distribution of Covid cases contained in our dataset. We have many more data plots, and brief explanations that are shown in the eda.ipynb file. Missing values of each attribute are described at the end of the notebook.

1.2 Data cleaning and Imputing missing values

Initially, all string variables have been converted to lower case and 887 rows that have age, sex, and province missing concurrently have been removed as their quality is not as ideal. There are no missing entries for long and lat. Additional information has been tokenized, stop words have been removed, and the domain in the source has been parsed out. For missing values of additional information and source, an empty string has been filled in. Only rows with “Taiwan” as a province have their country missing, thus “China” has been filled in. The province ‘Caba’ has been replaced with ‘Buenos Aires’ since it refers to the same province, and ‘prefecture’ has been replaced with an empty string to keep consistency with location.csv. Since their coordinates are not missing, they have been used to reverse geocode the missing provinces. For countries that are categorized by district or region, “Not applicable” has been filled if the province is missing. It seems unreasonable to assign sex as it would result in a highly skewed distribution, so the category “not available” has been introduced. Age range has been replaced by the mean of the range, and ages containing “months” have been parsed and divided by 12. Ages close to 0 have been rounded to 0, and ages containing “+” or ”-” have been replaced by the mean of that value with either 0, 18, or 100, whichever is closest. For example, “80+” has been replaced to (80+100)/2 = 90. For rows missing values for age, non-null values for ‘long, 'lat', 'sex', 'date confirmation' have been grouped by age and the mode of the group that best matches replaces the missing value. If there are still missing values, then it would take the mode of the overall age column to fill in the rest of the missing values. The intuition is to create a donor pool that has similar feature values as the missing row and extract the most probable values. Two more columns have been derived from age, one to measure the effect of averaging age ranges and filling missing values, another to determine if any further generalization of age benefits performance. For date confirmation, date ranges have been replaced with the first date in the range because it may represent the time when a patient is hospitalized until the patient leaves the hospital. The impact of such replacement will be small because there are only a few records with non-null date ranges. For missing values in the date confirmation column, the non-null values are grouped by long and lat, and the mode of the group that best matches the missing value’s row is being filled in. The process repeats with province and country if no suitable match is found. Otherwise, we replace it with the mode of the entire column. The idea is that people in the same area are likely to be exposed to an outbreak at the same time. In addition, we turned date confirmation into a timestamp variable from a string variable. This will impose an ordering on the dates and the categorized data. This same assigning process is applied to the test dataset, except for the removal of low quality entries in the initial step.

1.3 Dealing with outliers

For this section, we removed outliers for various attributes. For the age attribute, we removed the data from 141 rows that contained ranges spanning more than 40 years. We felt this appropriate as recomputing this value in the imputation step would provide much better quality to this age variable than a mean on such a massive range. Initially, we wanted to assume a normal distribution based on the histogram of age (See outlier_age_init.png) and purge the top 1% of data, but the values of age range from 0 (baby) to 106 after imputation, all of which fall into a normal human lifespan. For the sex variable, after imputation, we have 3 categorical values of, male, female, and not available. And, the ratio between male/female is a reasonable split. Further, since the source and additional information attributes have no specific structure, we conclude that there are no outliers in those columns. There are also no outliers in the date confirmation column because the value of the day is in a valid range and so are month and year.

That being said, there are some erroneous rows in the province column in which Finland is categorized by hospital districts. However, I did not consider them as outliers because removing them would remove all representation of Finland. We also detected a case for Ontario,China. This is a location that does not exist. So, we threw that out as well. There are two rows that concatenate 3 different areas by ‘or’ while other provinces under the same country refer only to a specific area, thus, we consider these outliers, and the rows have been removed. There are no outliers within the country columns after manual verification in the unique values. At first glance, the range of longitude and latitude are inside the appropriate range of -180 to 180 and -90 to 90. However, looking at the heat map, we realize that there are some cases happening in the ocean. We decided to apply isolation forest onto the combination of long/lat to remove the outliers, but it was capturing too many valid pairs. We ended up using another package called global_land_mask and it was able to detect long/lat pairs not located on land (See outlier_init_lonlat.png). Due to technological limitations, we still had to manually verify a few pairs of outliers, but, a total of 3 rows that were in the ocean were deleted. (See outlier_final_lonlat.png)

1.4 Transformations

As tasked, we took the US county data in locations.dat and aggregated it such that every county was combined into a single row for each individual state or territory. Latitude and longitude of the state was calculated by taking the mean of the latitude and longitude data given by the counties, as that should approximate the state’s latitude and longitude fairly well. For columns regarding the case, death, activity and recovery counts, we simply summed the appropriate column, since the sum of cases for the data in all counties should be the sum of data for the state. The combined key did need to be recalculated for the state, by setting it to form ‘STATE, United States’. And, given the incidence rate is the prevalence of new cases per 100,000 population, we were able to simply sum over those columns in the county data as well. Finally, we did need to recalculate the case-fatality rate, since we now had a new number of cases and deaths, but that was a straight calculation of (State Deaths)/(All Cases) * 100 for each respective state. In transforming the data, one additional data cleaning step was taken, as it was noted that some rows had a negative number of active cases, which has no logical grounding.So, these rows were filtered out and removed from location_transformed.csv.

1.5 Joining the cases and locations dataset

We chose to take an adaptive technique to join our dataset. First, we did an exact match on longitude and latitude between the cases and location data. This helped match approximately 11 rows that could not be joined otherwise due to the format of the province imputed being different from the format of the province in the location_transformed.csv datafile. Then, we joined on 'province, country', because approximately matching longitudes and latitudes would have been much more challenging, given that cases may be registered within one province, but have coordinates more similar to another. Additionally, because we imputed province and country data in an earlier step, this was an especially good matching technique.

However, some countries in our location_transformed.csv file contained no province information, so, after matching on 'province,country', we also matched on 'country' alone. Once that was completed, the remaining rows were entries that could not be matched by province and country as their data was not available in location.csv previous steps. Therefore, we aggregated the location.csv at the national level and took the mean for each column. We then matched it back to the missing values by the country column as well. At this time, only 3 entries from Puerto Rico couldn’t be matched back, therefore we simply imputed the values to the overall mean from the match result as that would represent the average value for a country.

1.6 Outcome labels

The outcome labels are: hospitalized, recovered, nonhospitalized, and deceased. The definitions of which are fairly apparent. For example, the "Hospitalized" label identifies Covid cases that required the patient to be hospitalized, perhaps currently, or perhaps ever. The "Nonhospitalized" label identifies Covid cases that didn't require hospitalization, probably referencing cases that only required self-isolation. The "Recovered" label represents the cases that ultimately recovered from the illness, and the "Deceased" label means that the patient in the case ultimately ended up dead, most probably from Covid, or Covid-related complications. We believe that the prediction of the outcome labels falls under classification. Classification in data mining is the prediction of outcome labels (containing different categories) and can be done through numerous techniques. In the next step, we will test several classification models and evaluate their performance to correctly predict outcome labels, most notably the "deceased" outcomes.