There are two parts to this assignment: evaluating annotator agreement for each data file, and assembling the files together to come up with datasets where each text has a single label that can be used for modeling.

Each student is assigned a PRIMARY dataset and a SECONDARY dataset.  You will need to assemble the files for the primary and secondary datasets separately, so at the end of the assignment you will have two separate files to work with.

Assignments of primary and secondary datasets are in this document: <https://blackboard.iit.edu/bbcswebdav/pid-1080664-dt-content-rid-21815160_1/xid-21815160_1>

Download datasets and utilities for this assignment here: <https://gauss.cs.iit.edu/~cs585/datasets.html>

Your submission for this assignment will be the following files:

* a Jupyter notebook with your code for parts 1 and 2
* the output files for part 2 corresponding to the primary and secondary datasets

If your data files correspond to the STANCE task, annotators used three mutually-exclusive labels.  You will do parts 1 and 2 only once, and submit two data files (corresponding to the primary and secondary datasets).  If your data files correspond to the TOPIC task, annotators used three *independent* labels.  You will use the script provided to create a new datafile corresponding to each label (True/False for lockdowns, True/False for masking & distancing, True/False for vaccination).  So you will do parts 1 and 2 three times, and submit a total of six data files (three corresponding to the primary dataset and three corresponding to the secondary dataset).

**Part 1: Evaluate annotator agreement**

1. Look in the attached spreadsheet to find your primary and secondary datasets.
2. Go to the course website, and download all of the data files corresponding to your primary and secondary datasets.  (There will likely be multiple CSV files for each.  Each file contains annotation results for 200-300 texts.)
3. Each CSV file contains annotation results from 3-5 annotators, with each annotator's labels indicated in a separate column using a unique ID.  For each pair of annotators, calculate the Cohen's kappa agreement score.  You may use a library implementation of this function such as [the one in sklearn](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.cohen_kappa_score.html). (For example, there will be a kappa score for (annotator\_1, annotator\_2), and another kappa score for (annotatot\_1, annotator\_3).  If there are undefined or empty values in the data, treat them as a separate label (e.g., "missing")
4. Calculate a score for each individual annotator, which is the average kappa score for that annotator with all others.  (E.g., if there are 4 annotators, the score for annotator 1 would be   
   

**Part 2: Assemble datasets**

1. Assign a final label to each text, according to the following logic:
   1. First, eliminate any labels for annotators whose average kappa score is less than 0.2 (unreliable annotators)
   2. Second, assign the final label to each text as the most frequent label among the remainiing annotators
   3. If there are ties (the same number of annotators for different labels), use the label with higher-reliability annotators (higher kappa scores on average)
2. Combine all of the text/label pairs for the PRIMARY dataset into a single CSV file, with the columns "text" and "label"
3. Combine all of the text/label pairs for the SECONDARY dataset into a single CSV file, with the columns "text" and "label"

Submit the Jupyter notebook with code for parts 1 & 2, as well as the final datasets produced in part 2, as your answer to this assignment.