COVID-19 Research Project

The objective of this project is to uncover insights from COVID-19 case data and build models that will provide a clearer picture of what is happening in the United States right now. The data collected spans from January 21st 2020 to April 30th 2020. Four models were built, using supervised learning methods, each answering one of the questions posed below:

1. If a state imposes a stay-at-home or shelter-in-place intervention, can I predict whether a county in that state will see a less than 30% increase in week-on-week COVID-19 case counts within 3-weeks of intervention?
2. If a state imposes a stay-at-home or shelter-in-place intervention, can I predict whether a county in that state will see a 100% decrease in weekly case counts from week 2 to week 3, post-intervention?
3. Can I predict whether a state will implement stay-at-home or shelter-in-place intervention?
4. Can I predict whether there is a greater than 5% chance of dying in a US county if diagnosed COVID-19 positive?

County-level demographic, weather, and COVID-19 case data for the contiguous United States, Alaska, and Hawaii was used to build each model. Overall, all models showed some predictive capability and would likely improve if provided with more county-level demographic data. The best performing model was the one for predicting whether a state would impose a stay-at-home or shelter-in-place intervention, which identified the political party of a state governor as a key indicator.

Two custom datasets were created since there did not exist a dataset that had all the features I thought would provide predictive power to the models I sought to build. The dataset, dataset_b.arff is, in a way, a subset of the dataset, dataset_a.arff, containing only observations of counties that implemented a stay-at-home or shelter-in-place intervention and for which there was data available within the span of 21-days (3-weeks) of the states intervention date.

All demographic data was collected using two datasets provided through the United States Census Bureau API. The datasets utilized were the American Community Survey 5-Year Data (2009-2018) and County Population Totals: 2010-2019. The National Conference of State Legislatures, “2020 State & Legislative Partisan Composition”, publication was used to extract data on a state governors political party and the political party with state control. The NYTimes, “See Which States Are Reopening and Which Are Still Shut Down”, publication was used to determine whether a state had implemented a form of stay-at-home or shelter-in-place intervention and retrieve the date the intervention had gone into effect.

All weather data was collected from the National Centers for Environmental Information (NOAA). For the purpose of this research, I was focused on gathering county-level monthly average temperatures for the month of January, February, March, and April 2020. Data was available at this level for all states except Hawaii. Data for Hawaii wasn’t available at the county-level in an aggregated form, but was instead available per station. I had to create a mapping between a station's unique identification code and the county the station fell within to be able to work with the data at the level I needed.

Using this station-to-county mapping I was able to map the temperature reported by each station to the county it corresponded to and then calculate the average temperature for each county. I was able to do this for January, February, and March. For April, the monthly average temperatures weren't available, but the max and min temperatures were. To calculate the average temperature for each county in April, I had to approximate. I calculated an average max and an average min temperature for all station readings and then calculated the average monthly temperature for April, per county. After calculating the monthly average temperatures for Hawaii, I merged the data with that collected for all other counties.

All COVID-19 case data was extracted from the NYTimes "Coronavirus (Covid-19) Data in the United States" dataset, at the county-level. All observations with "Unknown" or "District of Columbia as the county value were removed from the dataset. All observations for Kansas City, Missouri were also removed, because all temperature and demographic data was available only for the four counties (Cass, Clay, Jackson, and Platte) that overlap the municipality of Kansas City, Missouri and not exclusively Kansas City, which is what observations for "Kansas City, Missouri" in the case data represented.

The COVID-19 case data was the last dataset to be merged with the others because it required some additional preprocessing and feature extraction. The columns included in the COVID-19 case dataset include date, county, state, fips, cases, deaths. Prior to merging the COVID-19 case dataset with the main dataset, the data for counties within New York City (New York, Kings, Queens, Bronx and Richmond) in the main dataset was aggregated and a new row created with "New York City" as the county name and "36999" set as the state-county joint FIPS code. This fake FIPS code was also set as the "fips" value for "New York City" in the COVID-19 case dataset.

For dataset_a, the features cases_since_01212020, deaths_since_01212020, and high_chance_of_death_5pct (at least 5% chance of death) were derived using the COVID-19 case dataset. All cases and deaths were summed from Jan 21st 2020 (the first date available in the dataset) to April 30th 2020 (an arbitrary cutoff I set) and grouped by FIPS code (state, county) to derive cases_since_01212020 and deaths_since_01212020. To calculate high_chance_of_death_5pct, I divided deaths_since_01212020 by cases_since_01212020 and multiplied by 100.

For dataset_b, the features at_least_100pct_w2w3_decrease (at least 100% decrease between week 2 and week 3) and less_than_30pct_increase_wow (less than 30% increase week-on-week) were derived using the COVID-19 case dataset as well. To develop these features, all observations with a date prior to the date a state intervention was put in place and all observations with a date 22-days, or more, past the date an intervention was put in place, were removed. Then all cases were averaged on a weekly basis to calculate the week-on-week percentage change in case numbers by county. If a county experienced at least a 100% decrease in case numbers from week 2 to week 3, they received a classification (value) of “y” (“yes”) under the column at_least_100pct_w2w3_decrease. If a state experienced a less than 30% increase in case counts from week 1 to week 2 or week 2 to week 3 they received a classification (value) of “y” (“yes”) under the column less_than_30pct_increase_wow.

Once all separate datasets had been collected, cleaned, and any additional features created, it was time to merge them together. To merge all separate datasets together, I used the Federal Information Processing System (FIPS) Codes for States and Counties (source: https://transition.fcc.gov/oet/info/maps/census/fips/fips.txt) and created a CSV version containing all state-level FIPS codes, county-level FIPS codes, state-county joint FIPS codes, state names, and place (county) names. In this CSV I also added a column for county names which were a replica of the column containing place names but with “county”, “borough”, “census area”, and “parish” removed from the place name. This was done to better match the convention used by the NYTimes case dataset.

Using this FIPS mapping dataset I was able to merge all datasets except for the temperature dataset. The NOAA temperature dataset uses a different encoding scheme for its states. To correctly map the NOAA dataset I had to create a separate file that mapped the NOAA state encoding to the state FIPS codes. Once I had this, I updated the NOAA dataset to include columns for each observation’s state FIPS code and state-county joint FIPS code. I then used the state-county joint FIPS code to merge the NOAA dataset with the others.

The final cleaning of dataset_a.csv and dataset_b.csv included removing any observations with a “NULL” value in a column, ensuring the District of Columbia and Puerto Rico were not in the datasets, and ensuring only states that had implemented an intervention where included in dataset_a. In the process of collecting and preprocessing the data I needed, and then building my final datasets I made use of Python for extracting the Census API data and exporting it to CSV, Excel for the creation of the mapping datasets I required and quick preprocessing; SQL for preprocessing, cleaning, and merging the datasets I worked with and outputting the final datasets in CSV form; and Weka for converting the CSV files into .arff files for processing in Weka and model creation.

The table below includes the following details for the datasets created: feature name, type of value, a description of the feature, a link to the source dataset, a list of the models for which the feature was included, and the dataset the features can be found in. For reference, dataset_b was used to build the models answering questions 1 and 2, and dataset_a was used to build the models answering questions 3 and 4. The abbreviation “pct” stands for “percentage”.

Feature	Type	Description	Source	Models Used In	Dataset Included In
state	nominal	One of the 48 states in the contiguous United States, Alaska, or Hawaii	https://transition.fcc.gov/oet/info/maps/census/fips/fips.txt	Omitted from all	dataset_a, dataset_b
county	nominal	The counties within each of the states in that dataset	https://transition.fcc.gov/oet/info/maps/census/fips/fips.txt	Omitted from all	dataset_a, dataset_b
fips	numeric	A 4-5 digit code that represents the state FIPS code and county FIPS code combined (1001 - 1 (Alabama) + 001 (Autauga County)	https://transition.fcc.gov/oet/info/maps/census/fips/fips.txt	Omitted from all	dataset_a, dataset_b
median_household_income	numeric	[Estimate] Income and Benefits (in 2018 inflation-adjusted dollars). Total households, median household income (dollars).	The American Community Survey (ACS) 5-year, Data Profiles: https://www.census.gov/data/developers/data-sets/acs-5year.html	1, 2, 3, 4	dataset_a, dataset_b
pct_households_below_poverty	numeric	[Percent Estimate] Percentage of families and people whose income in the past 12 months is below the poverty level. Includes all families. Data collected at the county-level.	The American Community Survey (ACS) 5-year, Data Profiles: https://www.census.gov/data/developers/data-sets/acs-5year.html	1, 2, 3, 4	dataset_a, dataset_b
pct_pop_over_65	numeric	[Percent Estimate] Total population 65 years and over. Includes all sex and age groups. Data collected at the county-level.	The American Community Survey (ACS) 5-year, Data Profiles: https://www.census.gov/data/developers/data-sets/acs-5year.html	1, 2, 3, 4	dataset_a, dataset_b
pct_pop_uninsured	numeric	[Percent Estimate] Civilian noninstitutionalized population with no health insurance coverage. Data collected at the county-level.	The American Community Survey (ACS) 5-year, Data Profiles: https://www.census.gov/	1, 2, 3, 4	dataset_a, dataset_b

			data/developers/data-sets/acs-5year.html		
pct_working_pop_using_public_transit	numeric	[Percent Estimate] Workers 16 years and over using public transportation (excluding taxicab) for commuting to work. Data collected at the county-level.	The American Community Survey (ACS) 5-year, Data Profiles: https://www.census.gov/data/developers/data-sets/acs-5year.html	1, 2, 3, 4	dataset_a, dataset_b
pop_density	numeric	[Estimate] A measurement of population per unit area. Data collected at the county-level.	County Population Totals: 2010-2019: https://www.census.gov/data/datasets/time-series/demo/popest/2010s-counties-total.html	1, 2, 3, 4	dataset_a, dataset_b
governor_political_party	nominal	Counties were assigned a value of "republican" or "democrat" according to the state the county fell under.	National Conference of State Legislatures: https://www.ncsl.org/Portals/1/Documents/Elections/Legis_Control_2020_April%201.pdf	1, 2, 3, 4	dataset_a, dataset_b
state_control_political_party	nominal	Counties were assigned a value of "republican", "democrat", "divided", "non-partisan" according to the state the county fell under.	National Conference of State Legislatures: https://www.ncsl.org/Portals/1/Documents/Elections/Legis_Control_2020_April%201.pdf	1, 2, 3, 4	dataset_a, dataset_b
imposed_intervention	nominal	States that imposed an intervention up to, and including, April 30th 2020 were assigned a value of "y" and those who imposed no intervention were assigned "n". Data was gathered at the state level and assigned to each county based on the state it fell under.	NYTimes: https://www.nytimes.com/interactive/2020/us/states-reopen-map-coronavirus.html	3, 4	dataset_a

tavg_jan	numeric	The average temperature for the month, at the county-level	National Centers for Environmental Information: ftp://ftp.ncdc.noaa.gov/pub/data/cirs/climdiv/, for Hawaii Global Summary of the Month (GSOM) was used to derive monthly averages: https://www.ncei.noaa.gov/access/search/data-search/global-summary-of-the-month	1, 2, 3, 4	dataset_a, dataset_b
tavg_feb	numeric	The average temperature for the month, at the county-level	National Centers for Environmental Information: ftp://ftp.ncdc.noaa.gov/pub/data/cirs/climdiv/, for Hawaii Global Summary of the Month (GSOM) was used to derive monthly averages: https://www.ncei.noaa.gov/access/search/data-search/global-summary-of-the-month	1, 2, 3, 4	dataset_a, dataset_b
tavg_mar	numeric	The average temperature for the month, at the county-level	National Centers for Environmental Information: ftp://ftp.ncdc.noaa.gov/pub/data/cirs/climdiv/, for Hawaii Global Summary of the Month (GSOM) was used to derive monthly averages: https://www.ncei.noaa.gov/access/search/data-search/global-summary-of-the-month	3, 4	dataset_a

tavg_apr	numeric	The average temperature for the month, at the county-level	National Centers for Environmental Information: ftp://ftp.ncdc.noaa.gov/pub/data/cirs/climdiv/, for Hawaii Global Surface Summary of the Day was used to derive monthly averages: https://www.ncei.noaa.gov/access/search/data-search/global-summary-of-the-day	3, 4	dataset_a
cases_since_01212020	numeric	The number of total confirmed case counts from 1/21/2020 to 4/30/2020, at the county-level	Derived from NYTimes Coronavirus (COVID-19) Data in the United States: https://github.com/nytimes/covid-19-data	3	dataset_a
deaths_since_01212020	numeric	The number of total deaths from 1/21/2020 to 4/30/2020, at the county-level	Derived from NYTimes Coronavirus (COVID-19) Data in the United States: https://github.com/nytimes/covid-19-data	3	dataset_a
high_chance_of_death_5pct	nominal	The number of deaths divided by the number of cases, multiplied by 100, at the county-level. Those that experienced a death rate of 5% or higher were assigned a value of "y", those who didn't were assigned a value of "n".	Derived from NYTimes Coronavirus (COVID-19) Data in the United States: https://github.com/nytimes/covid-19-data	3, 4	dataset_a

less_than_30pct_increase_wow	nominal	Determined by calculating week-over-week percentage change in case numbers for counties in states that implemented a stay-at-home (or similar) intervention and for which there was data up to 3-weeks (21 days) of intervention date. Those who experienced a percentage change no greater than 30% during any week post-intervention, received a classification of "y" and those who didn't, received a classification of "n". Data was processed at the county-level.	Derived from NYTimes Coronavirus (COVID-19) Data in the United States: https://github.com/nytimes/covid-19-data	1	dataset_b
at_least_100pct_w2w3_decrease	nominal	Determined by calculating week-over-week percentage change in case numbers for counties in states that implemented a stay-at-home (or similar) intervention and for which there was data up to 3-weeks (21 days) of intervention date. Those who experienced at least a 100% decrease in percentage change from week 2 to week 3, received a classification of "y" and those who didn't, received a classification of "n". Data was processed at the county-level.	Derived from NYTimes Coronavirus (COVID-19) Data in the United States: https://github.com/nytimes/covid-19-data	2	dataset_b

When it came time to build the models, I tested several algorithms. Some algorithms, including K-Nearest Neighbors (kNN) and Artificial Neural Networks required that the data be normalized (rescaled). Normalization is when any feature with numerical data is transformed so that all values are between 0 and 1; it's used as a way to avoid any one feature (attribute) from dominating. When performing normalization on a dataset, I used the "Normalize" filter provided by Weka to update all numeric features at once. Other algorithms, like Support Vector Machine, required that any nominal (non-numeric) data be converted into numeric data, excluding the output variable. For this case, I simply retained the numeric representation of those features and removed their nominal counterpart from the dataset once imported into Weka. Since none of my numeric data had a Gaussian (Normal or bell-shaped) distribution, I didn't need to perform standardization, which is when all numeric data is transformed so that it has a mean of 0 and a standard deviation of 1.

Of the algorithms I used, I found that Support Vector Machine (SVM) and Artificial Neural Networks (ANN) performed worst across the board. SVM is a supervised learning algorithm that finds the line (or hyperplane) with the widest margin to best separate groups of data points with different classifications, and then uses this margin to classify unseen observations. The implementation of SVM used in Weka, LibSVM, wasn't able to find a hyperplane that cleanly divided the data, and it overfit, nearly (if not) always classifying unseen data as the majority class.

A somewhat similar situation occurred when building a model using Weka's implementation of ANN, the MultilayerPerceptron algorithm. ANN is a supervised learning algorithm that assigns a weight (a number) to each feature based on its perceived importance, performs a weighted sum, and then uses another function (e.g. Sigmoid function) to determine the output. I believe these implementations of SVM and ANN performed poorly, in general, because of a lack of sufficient data and a higher percentage of observations with one classification over the other. While LibSVM performed poorly across all questions, the MultilayerPerceptron algorithm performed at least as well as other models for questions 3, which included slightly more features and several more observations in the input dataset.

Other algorithms tested in the building of my four models include K-Nearest Neighbors (IBk in Weka), Decision Tree's (J48 in Weka), Random Forests, Bagging, and Bayes Net. Of the models that were built, those that were selected as "good enough" given the question at hand, were built using K-Nearest Neighbors and Decision Trees.

K-Nearest Neighbors (kNN) is a supervised learning algorithm where you assign a value for *k*, which determines the number of neighbors the algorithm evaluates when calculating the classification of an unseen data point. The algorithm will assign the unseen data point the classification of the majority. A Decision Tree (also known as CART) is a supervised learning algorithm that selects features that will produce subsets of the original dataset (or data in the parent node) containing less noise or entropy (the number of observations that have different classifications). This process is done repeatedly until the subsets of data can't be separated any further.

To evaluate the models that were built, and ultimately select one model for each question posed, I used two evaluation methods. I analyzed the confusion matrix of each model, which is a table that describes the performance of a classification model, showing you a count of the observations that were predicted under each classification and a count of the actual observations that fell under that classification. Since all my questions were binary classifications, the confusion matrix for each problem was 2-columns by 2-rows. I also analyzed the Receiver Operating Characteristic (ROC) curve, which is a graph that summarizes the performance of a model over all possible thresholds. The higher the area under the ROC curve, the more resilient the model is likely to be. For example, an ROC area of 0.5 is equivalent to tossing a coin when deciding the classification of an unseen data point.

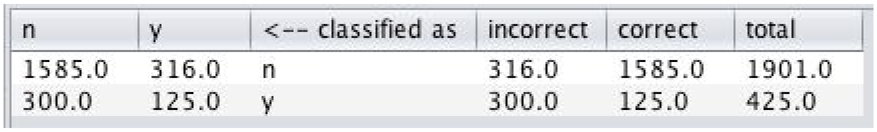
As a measure for testing the model, I selected k-fold cross-validation as my test option in Weka. K-fold cross-validation splits your data k-times (where k is a value you set; I used 10) into equal partitions, then a portion of the data is used as the training dataset and the rest is used as the testing dataset. At each iteration, the data is shifted so that a different combination makes up the training and testing datasets.

All the measures discussed below were used in the building, analysis, and selection of each model. It should also be noted that while kNN can see a performance boost from having input variables normalized, when building my models I didn't see a difference in performance when the data was normalized, so I kept the datasets I used as they were (unnormalized).

The model I ultimately selected to answer question 1, "If a state imposes a stay-at-home or shelter-in-place intervention, can I predict whether a county in that state will see a less than 30% increase in week-on-week COVID-19 case counts within 3-weeks of intervention?", was built using kNN. I used all the default values provided by Weka, including k=1, and adjusted only the random seed for xval/% split parameter to 5. I chose this model because if a state wanted to know whether one of its counties would experience a less than 30% increase in week-on-week COVID-19 case counts within 3-weeks of imposing an intervention, and my model output "y" ("yes"), the model would be accurate 30% of the time.

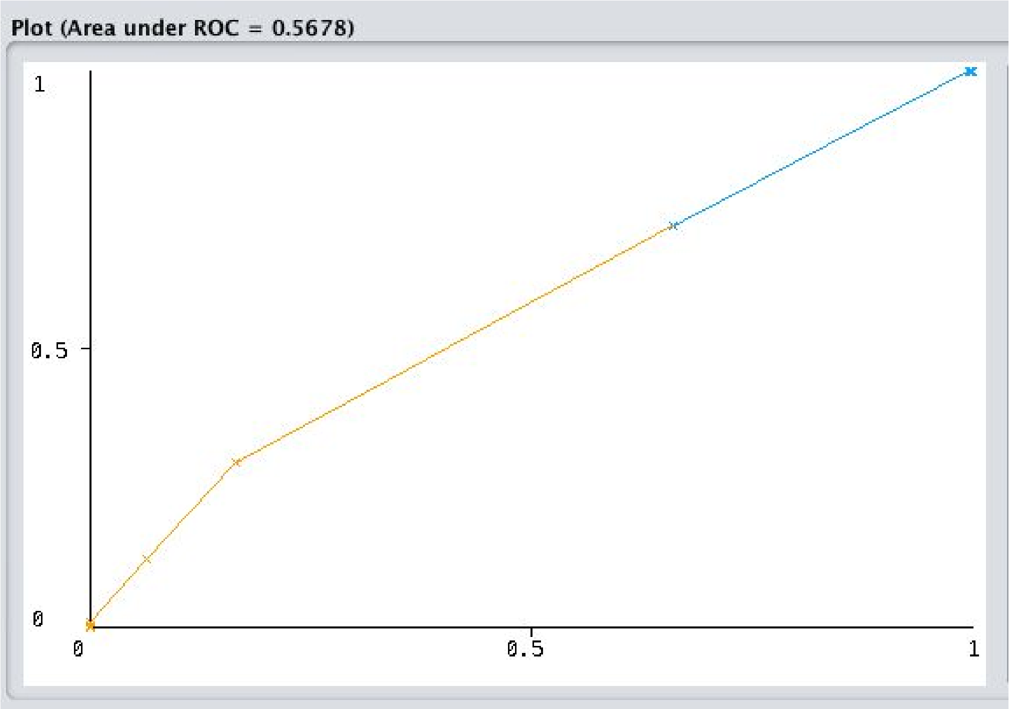
Even though the models ROC area, of 0.568, was lower than I'd ideally prefer, it was the only model I built that gave me that level of accuracy when classifying as "y", which is what I deemed was most important. The features used to build the model, followed by the confusion matrix and ROC curve for the model are available below.

Features retained from dataset_b.arff to build q1_knn.model	
median_household_income	governor_political_party
pct_households_below_poverty	state_control_political_party
pct_pop_over_65	tavg_jan
pct_pop_uninsured	tavg_feb
pct_working_pop_using_public_transit	less_than_30pct_increase_wow
pop_density	



n	y	<-- classified as	incorrect	correct	total
1585.0	316.0	n	316.0	1585.0	1901.0
300.0	125.0	y	300.0	125.0	425.0

Confusion Matrix for q1_knn.model

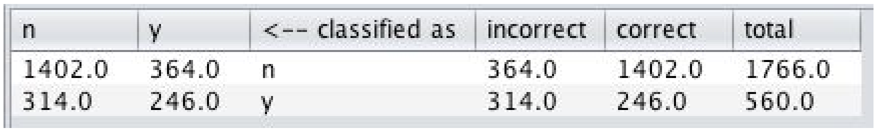


ROC curve for q1_knn.model

The model I selected to answer question 2, “If a state imposes a stay-at-home or shelter-in-place intervention, can I predict whether a county in that state will see a 100% decrease in weekly case counts from week 2 to week 3, post-intervention?”, was also built using kNN. I used most of the default values provided by Weka, but set k=2, used the weight by 1-distance distance weighting measure, and adjusted the random seed for xval/% split parameter to 6. As with question 1, I chose this model because if a state wanted to know whether one of its counties would experience at least a 100% decrease in weekly COVID-19 case counts from week 2 to week 3 of imposing an intervention, and my model output “y” (“yes”), the model would be accurate 44% of the time.

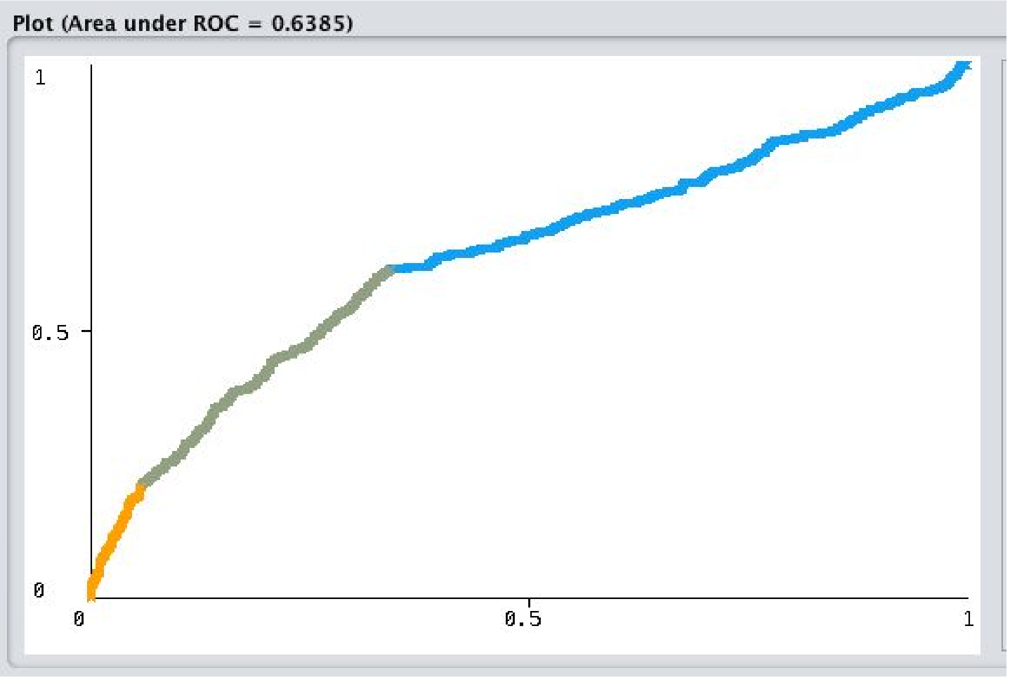
Similarly, even though the models ROC area, of 0.639, wasn’t as high as with other models I built, it was the only model I built that gave me that level of accuracy when classifying as “y”, which is what I deemed most important. The features used to build the model, followed by the confusion matrix and ROC curve for the model are available below.

Features retained from dataset_b.arff to build q2_knn.model	
median_household_income	governor_political_party
pct_households_below_poverty	state_control_political_party
pct_pop_over_65	tavg_jan
pct_pop_uninsured	tavg_feb
pct_working_pop_using_public_transit	at_least_100pct_w2w3_decrease
pop_density	



n	y	<-- classified as	incorrect	correct	total
1402.0	364.0	n	364.0	1402.0	1766.0
314.0	246.0	y	314.0	246.0	560.0

Confusion Matrix for q2_knn.model

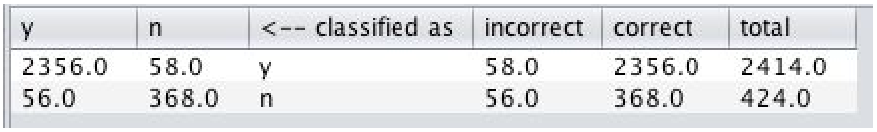


ROC curve for q2_knn.model

The model I selected to answer question 3, “Can I predict whether a state will implement stay-at-home or shelter-in-place intervention?”, was built using a Decision Tree (J48 in Weka). I used most of the default values provided by Weka but adjusted the random seed for xval/% split parameter to 9. This was not the best performing model, but still performed very well with an ROC area of 0.946, minimized the observations that were incorrectly classified, and provided a visual representation that is interpretable. The best performing model was one built using Bagging (with J48 as the classifier), and while I initially selected this as my model, I realized that a model which is simpler and easily interpreted is a better fit for what I think is most valuable.

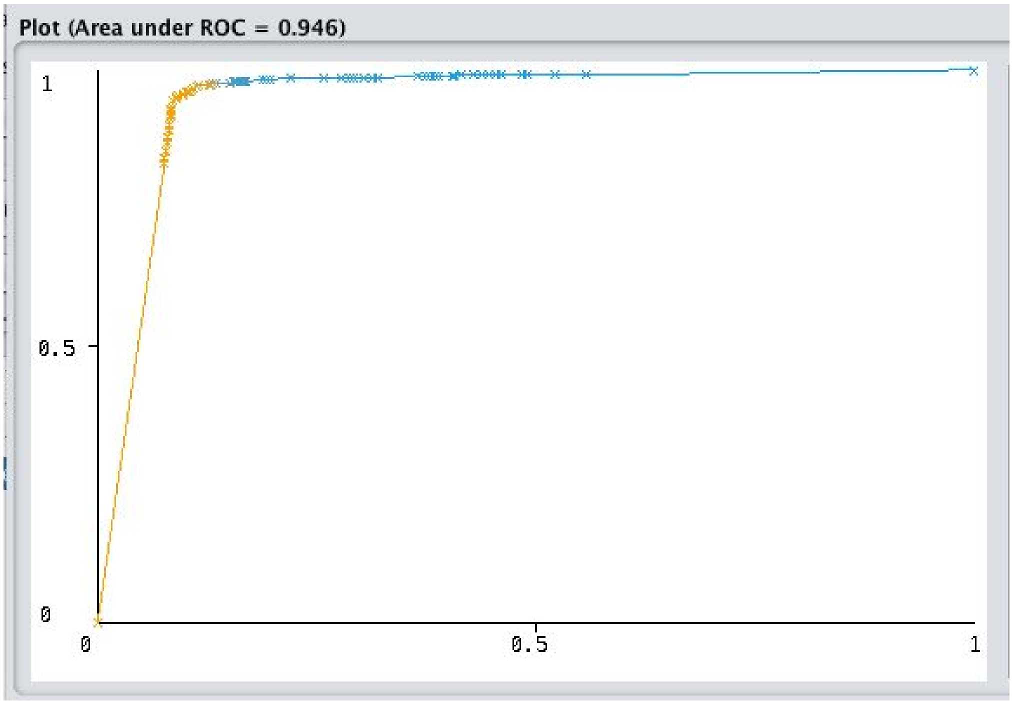
The features used to build the model, followed by the confusion matrix and ROC curve for the model are available below. The Decision Tree built by the model is provided in the appendix. I will note that when interpreting the Decision Tree, it appears the three key features are the political party of the state’s governor, the average temperature (which could be interpreted as a division between the Northern and Southern United States), and lastly, the political party that has state control.

Features retained from dataset_a.arff to build q3_j48.model	
median_household_income	state_control_political_party
pct_households_below_poverty	tavg_jan
pct_pop_over_65	tavg_feb
pct_pop_uninsured	tavg_mar
pct_working_pop_using_public_transit	tavg_apr
pop_density	cases_since_01212020
imposed_intervention	deaths_since_01212020
governor_political_party	high_chance_of_death_5pct



y	n	<-- classified as	incorrect	correct	total
2356.0	58.0	y	58.0	2356.0	2414.0
56.0	368.0	n	56.0	368.0	424.0

Confusion Matrix for q3_j48.model

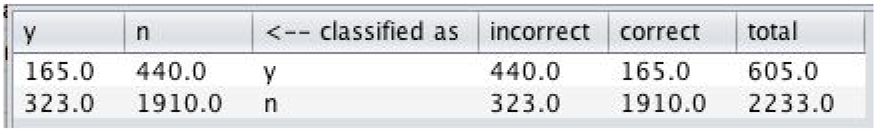


ROC curve for q3_j48.model

The model I selected to answer question 4, “Can I predict whether there is a greater than 5% chance of dying in a US county if diagnosed COVID-19 positive?, was built using kNN. I used most of the default values provided by Weka, but set k=6, and adjusted the random seed for xval/% split parameter to 2. As with question 1 and 2, I chose this model because if a state wanted to know whether someone newly diagnosed with COVID-19 was likely to die, and my model output “y” (“yes”), the model would be accurate 27% of the time.

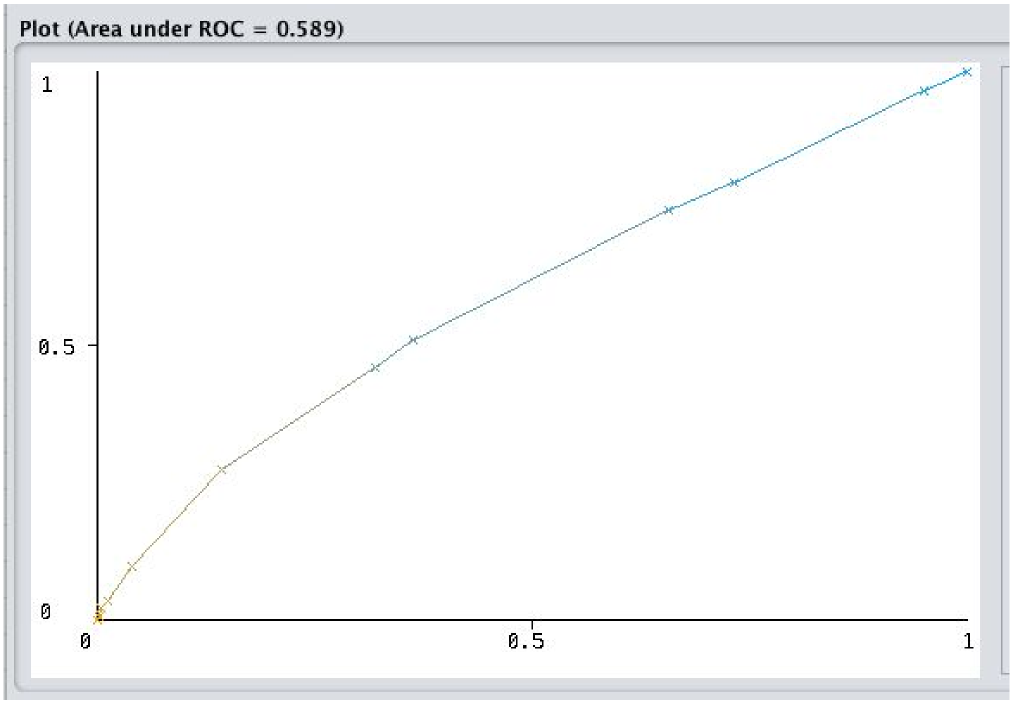
Similarly, even though the models ROC area, of 0.589, wasn’t as high as with other models I built, it was the only model I built that gave me that level of accuracy when classifying as “y”, which is what I deemed most valuable. The features used to build the model, followed by the confusion matrix and ROC curve for the model are available below.

Features retained from dataset_a.arff to build q4_knn.model	
median_household_income	governor_political_party
pct_households_below_poverty	state_control_political_party
pct_pop_over_65	tavg_jan
pct_pop_uninsured	tavg_feb
pct_working_pop_using_public_transit	tavg_mar
pop_density	tavg_apr
imposed_intervention	high_chance_of_death_5pct



y	n	<-- classified as	incorrect	correct	total
165.0	440.0	y	440.0	165.0	605.0
323.0	1910.0	n	323.0	1910.0	2233.0

Confusion Matrix for q4_knn.model

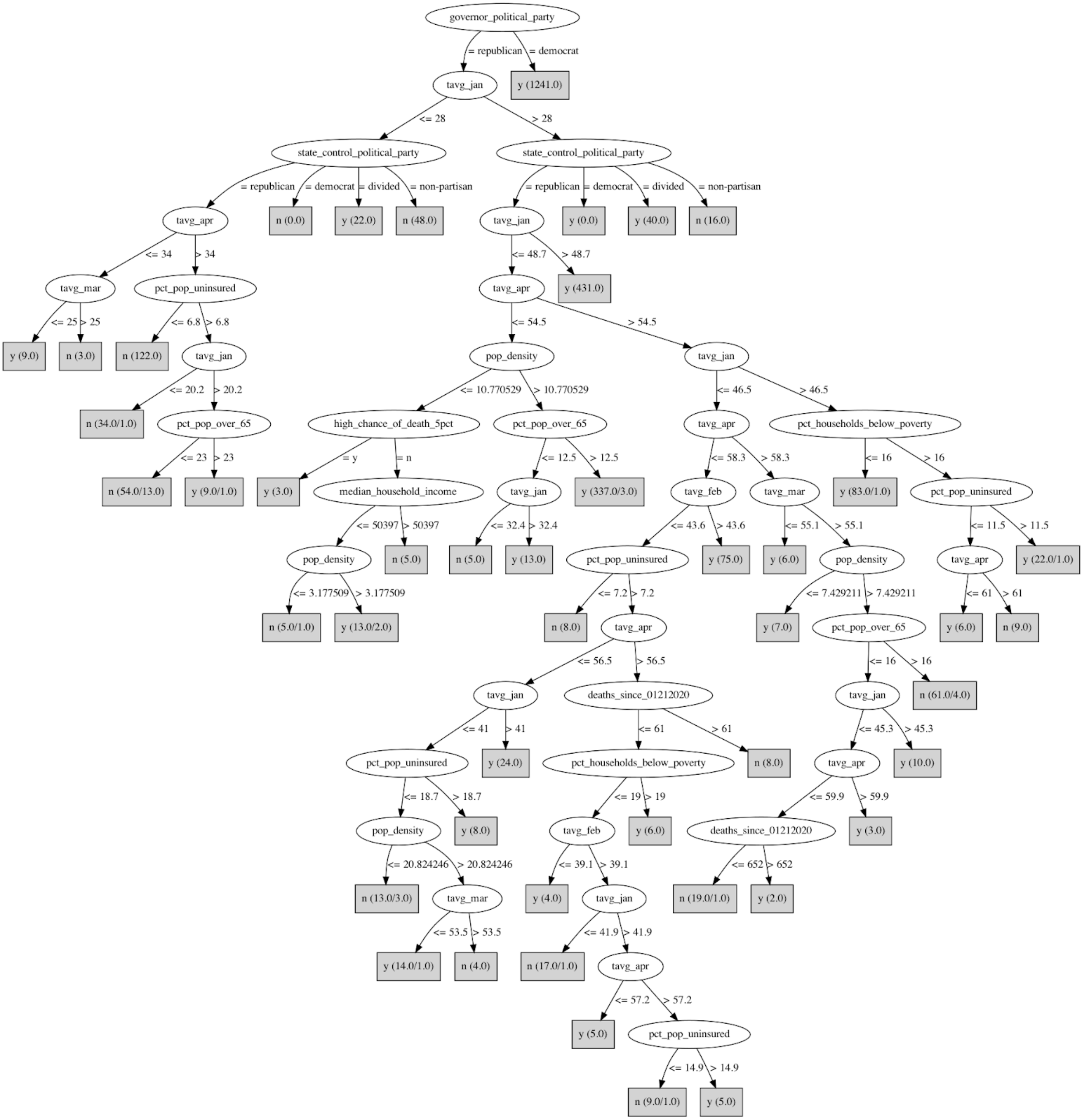


Plot (Area under ROC = 0.589)

ROC curve for q4_knn.model

In conclusion, the models built in this project help uncover information around COVID-19 survival rates, the impact of stay-at-home or shelter-in-place intervention, and the impact a states political alignment (amongst other features) plays in whether or not stay-at-home or shelter-in-place measures are put into action. This project has helped me better understand what is happening at this moment in history, and made it clear that it is important to make data easily and openly accessible so that individuals, like myself, can build upon it. To tackle this pandemic and make informed decisions going forward, more data should be made available and more individuals should come together to build machine learning models that can help guide decision making. If you'd like to look through the data that went into building these models or the models themselves, all this and more is available on GitHub at https://github.com/KISS/covid-19_research.

Appendix



Decision Tree (J48) output from q3_j48.model