2022 - 2023 H5N1 Bird Flu Modeling and Prediction in the United States

Weilin Cheng* Hengyuan Liu† Kathy Mo‡ Sida Tian§ Li Yuan¶

March 12th, 2023

Abstract

This report presents an analysis of the likelihood of H5N1 outbreaks in different counties of the United States in January 2023 using logistic regression, ridge regression, and lasso regression models. The models were trained using historical data from 2022, and the accuracy of the models in predicting H5N1 outbreaks in January 2023 is about 98.4%. The lasso regression model performed the best among the three models, with an AUC of 0.8015. The map generated based on the lasso regression model indicated that counties in the north and west were at a higher risk of having H5N1 outbreaks in January 2023, which matched the actual result. The report concludes that there are limitations to the models, including the consideration of only a limited set of factors affecting the spread of the virus and the use of historical data. Future work could incorporate additional data sources and use more sophisticated machine learning techniques to improve the accuracy of the models. The report also proposes some possible remedies to help control the spread of H5N1.

Useful Links

Project Website: https://h5n1.gd.edu.kg/

Project Dashboard: https://iasc2023.gd.edu.kg/dashboard/

GitHub Repository: https://github.com/GitData-GA/iasc2023

Poster: https://iasc2023.gd.edu.kg/poster/iasc2023-poster.pdf

*University of California at Davis, wncheng@ucdavis.edu
†University of California at Davis, hyliu@ucdavis.edu
‡University of California at Davis, kamo@ucdavis.edu
§University of Michigan at Ann Arbor, startian@umich.edu
¶University of Michigan at Ann Arbor, leeyuan@umich.edu

Contents

I. Introduction

Poultry, such as chicken, turkey, goose, duck, and others, is a staple food on our dinner tables. According to the United States Department of Agriculture (USDA) in 2022, each person had access to 68.1 pounds of chicken for consumption in 2021. This indicates that chicken is the most popular meat in the United States, and per capita egg consumption has increased by 15% in the past 20 years (UEP, 2021). However, like humans, poultry can also be infected with viruses, and in the context of the COVID-19 pandemic, we are reminded of how a small virus can have a significant impact on our lives. Bird flu caused by the H5N1 virus is one such example, and highly pathogenic avian influenza (HPAI) A(H5) viruses have been detected since January 2022 in U.S. wild aquatic birds, commercial poultry, and backyard or hobbyist flocks (CDC, 2023).

The H5N1 virus can have severe effects, and its outbreak has already caused economic, ecological, and environmental consequences with long-term effects. For example, the price of a dozen large Grade A eggs has more than doubled in 2022 in the United States (Iacurci, 2023). Moreover, sometimes grocery stores run out of eggs due to the virus, making it difficult for millions of people in the U.S. to maintain their usual levels of egg and poultry consumption. The virus has affected over 58 million poultry in 47 states and about 6,218 wild birds in 50 states and 959 counties (CDC, 2023), posing incalculable risks to our ecological environment and the poultry industry.

The avian influenza is not a new occurrence, and its effects on humans have been long-lasting and severe since its discovery in the 1880s. The H1N1 virus of avian influenza, for instance, caused 50 million deaths in 1918, and the H5N1 virus has infected 868 people and caused 457 deaths since 2003, according to the World Health Organization (WHO) in 2018. Therefore, this virus not only affects people's food consumption but also their health.

Given the economic, ecological, environmental, and health effects of avian influenza, we aim to perform analyses on its cases to provide predictions and suggestions for reducing its negative impacts. We will use Mathematical and Statistical methods to determine which counties are more likely to be infected by the H5N1 virus and should thus implement more countermeasures. We will also conduct visualizations and analyses based on the datasets provided by various authoritative organizations and institutions such as the CDC, USDA, U.S. Census Bureau, and the Bureau of Labor Statistics.

The objective of this report is to develop and evaluate machine learning models to predict the outbreak of the H5N1 virus in the United States in the future. Our report will focus on analyzing data from past outbreaks to build models that can accurately predict the likelihood of future outbreaks in different regions of the country. By identifying high-risk areas and providing actionable insights, we hope to contribute to efforts to mitigate the impact of the H5N1 virus and protect public health.

II. Data Description

To develop a predictive model for identifying counties that might be at risk of H5N1 infection in the future, we need to understand the structure and content of our data. Our approach involves merging four datasets to create a single, curated dataset that contains information on reported H5N1 cases in each county, during a specific month, from January 2022 to January 2023.

The cleaned dataset will be used to train our classification model to predict which counties might be likely to experience H5N1 infection in the upcoming months. By analyzing patterns in the data, we can identify key variables that are correlated with increased risk of infection, such as location, temperature, and flock type. This information will help us develop targeted interventions and public health strategies to mitigate the spread of H5N1 in high-risk areas.

1. United States Counties Database

This public dataset is provided by Pareto Software, LLC., who builds from the ground up using authoritative sources such as the U.S. Census Bureau and the Bureau of Labor Statistics. It contains all 3,143 county names, their FIPS codes, longitude, and latitude with respect to 51 states in the United States in 2023.

We make some changes to this dataset for future convenience. Specifically, we change the state full names to their abbreviations. This dataset makes it possible to generate an detailed geographical report of H5N1 cases in each county in the United States by matching observations in the latter datasets provided by the CDC.

There are 3,143 observations (counties) and 5 variables after the modification, shown as table 1 below.

Table 1: First 5 Observations of US Counties Database

FIPS Code	State	County	Latitude	Longitude
6037	CA	Los Angeles County	34.3209	-118.2247
17031	IL	Cook County	41.8401	-87.8168
48201	TX	Harris County	29.8578	-95.3936
4013	AZ	Maricopa County	33.3490	-112.4915
6073	CA	San Diego County	33.0343	-116.7350
6059	CA	Orange County	33.7031	-117.7609

You can access this public dataset from https://simplemaps.com/data/us-counties.

2. H5N1 Bird Flu Detections across the United States (Backyard and Commercial)

The second public dataset is about H5N1 bird flu outbreaks involving commercial poultry facilities, backyard poultry and hobbyist bird flocks by county in the United States.

We still make some modification to this dataset for future convenience. Since the original dataset has all records with detection dates, which is too specific, so in each observation, we split the date, year and month for future data cleaning and analyses. We also ignore the cases happened after January 31st 2023 because the later data is not complete enough.

Moreover, we generalize flock types to `Poultry` and `Non-Poultry`. Originally there are 15 commercial flock types besides `Poultry` and `Non-Poultry`, because the World Organization for Animal Health (WOAH) defines poultry as "all birds reared or kept in captivity for the production of any commercial animal products or for breeding for this purpose, fighting cocks used for any purpose, and all birds used for restocking supplies of game or for breeding for this purpose, until they are released from captivity" in March 8th 2022, we categorize all these 15 commerical flock types as `poultry`.

There are 746 observations and 6 variables after modification. This dataset is one of the datasets that will play a magnificent role in our future analyses.

Table 2: First and Last 5 Outbreaks in the United States till Jan. 31st 2023 (Backyard and Commercial)

State	County	Year Month	Day	Type	Cases
Indiana	Dubois County	2022_02	08	Poultry	29000
Kentucky	Fulton County	2022_02	12	Poultry	231400

State	County	Year Month	Day	Type	Cases
Virginia	Fauquier County	2022_02	12	Non-Poultry	90
Kentucky	Webster County	2022_02	15	Poultry	53300
Indiana	Dubois County	2022_02	16	Poultry	26600
...	...	...	...	...	...
Oregon	Polk County	2023_01	25	Non-Poultry	20
New York	Suffolk County	2023_01	25	Non-Poultry	10
Iowa	Buena Vista County	2023_01	25	Poultry	27700
Virginia	Rockingham County	2023_01	25	Poultry	10700
Maine	Hancock County	2023_01	27	Non-Poultry	40

Table 2 shows the first and last 5 H5N1 backyard and commercial outbreaks in the United States till January 31st 2023. We can see that the first outbreak happened on February 8th, 2022 in Dubois, Indiana with 29,000 cases and its outbreak type was `Poultry`. The last outbreak happened on January 27th, 2023 in Hancock, Maine with 40 cases and its outbreak type was `Non-Poultry`.

You can access this public dataset from https://www.cdc.gov/flu/avianflu/data-map-commercial.html.

3. H5N1 Bird Flu Detections across the United States (Wild Birds)

This public dataset contains information about detections of highly pathogenic avian influenza (HPAI) A(H5) viruses in wild birds by county in the United States.

We also make some modification to this dataset for future convenience. Same as source data 2, we change the format of dates and ignore the cases happened after January 31st 2023. We also change the column names to the same as the previous dataset for future data cleaning and analyses.

There are 2,517 observations and 6 variables after modification. This dataset is also one of the datasets that will play a magnificent role in our future analyses.

Note that there are two outbreak types, which are `Wild bird`, "means an animal that has a phenotype unaffected by human selection and lives independently without requiring human supervision or control (WOAH, 2022)," and `Captive wild bird` "means an animal that has a phenotype not significantly affected by human selection but that is captive or otherwise lives under or requires human supervision or control(WOAH, 2022)."

Table 3: First and Last 5 Outbreaks in the United States till Jan. 31st 2023 (Wild Birds)

State	County	Year Month	Day	Type	Cases
North Carolina	Hyde County	2022_01	12	Wild bird	2
South Carolina	Colleton County	2022_01	13	Wild bird	2
North Carolina	Hyde County	2022_01	16	Wild bird	2
North Carolina	Hyde County	2022_01	20	Wild bird	3
North Carolina	Pamlico County	2022_01	20	Wild bird	34
...	...	...	...	...	...
South Carolina	Berkeley County	2023_01	31	Wild bird	1
South Carolina	Colleton County	2023_01	31	Wild bird	1
South Dakota	Hughes County	2023_01	31	Wild bird	1
Virginia	Henrico County	2023_01	31	Wild bird	1
Washington	Skagit County	2023_01	31	Wild bird	1

Table 3 shows the first and last 5 H5N1 wild bird outbreaks in the United States till January 31st 2023. We can see that the first outbreak happened on January 12th, 2022 in Hyde, North Carolina with 2 cases and its outbreak type was `Wild bird`. The last outbreak happened on January 31st, 2023 in Skagit, Washington with 1 case and its outbreak type was also `Wild bird`.

You can access this public dataset from https://www.cdc.gov/flu/avianflu/data-map-wild-birds.html.

4. Monthly Average Temperature of each County across the United States

This dataset is combined from the public datasets provided by National Centers for Environmental Information (NCEI), which provides the average temperature in Fahrenheit degree (°F) of all counties, except those in the state of Hawaii, from January 2022 to January 2023, and Cedar Lake Ventures, Inc., which provides the average temperature in Fahrenheit degree (°F) of all five counties in the state of Hawaii from January 2022 to January 2023.

We change the formats and column names of state, county, and month for future convenience. In addition, we fix some mismatched county names in this dataset based on source data 1. Moreover, because the average temperature data of Hawaii is not available in any offline format, we fill those values manually.

There are 40,872 observations and 4 variables after modification. This dataset provides an important factor in our analysis and prediction model.

Table 4: First 5 Observations of Average Temperature in ∘F across the United States

State	County	Month Index	Average Temperature
AL	autauga county	1	45.1
AL	baldwin county	1	50.1
AL	barbour county	1	45.4
AL	bibb county	1	43.2
AL	blount county	1	41.6
AL	bullock county	1	44.9

You can access these public datasets from https://www.ncei.noaa.gov/access/monitoring/climate-at-a-glance/county/mapping and https://weatherspark.com/map?id=145043.

5. Monthly H5N1 Cases by County from Jan. 2022 to Jan. 2023 in the United States

This cleaned data is combined and derived from the 4 source data mentioned above, which is the major dataset we will use in the rest of this report. There are 163,436 observations and 10 variables.

- `fips`: Each FIPS code represents a unique county in the United States, so it is a categorical variable with 3,143 unique values, each unique value has 52 entries.
- `state`: Abbreviation of each state in the United States, so it is a categorical variable with 51 unique values, each states has its own number of counties.
- `county`: The names of counties, independent cities, census areas, and same administrative level regions in the United States, so it is a categorical variable with 3,143 unique values with respect to states, each unique value has 52 entries (note that some sates have some counties with the same name).
- `lat`: The latitude of each county.

- `lng`: The longitude of each county.
- `month.index`: The order of month of avian influenza outbreak from 1 (January 2022) to 13 (January 2023). Each `month.index` has 12,572 entries. Every month.index has the same number of entries because the cleaned dataset contains all counties' H5N1 situations regardless of how many cases they have, if there is no cases in a county, then the case number is just 0.
- `type`: The type of outbreak in a specific county and month, so it is a categorical variable with 4 unique values, including `poultry` (40,859 entries), `non-poultry` (40,859 entries), `wild bird` (40,859 entries), and `captive wild bird` (40,859 entries). Every type has the same number of entries because the cleaned dataset contains all counties' H5N1 situations regardless of how many cases they have, if there is no cases in a county, then the case number is just 0.
- `avg.temp`: The average temperature in a specific county and month in Fahrenheit degree (°F).
- `cases`: The number of H5N1 cases detected in a specific county and month.
- `binary.case`: If the case of a type of outbreak in a specific county and month is 0, then it is marked as `uninfected` (160,993 entries). Otherwise, it is marked as `infected` (2,443).

Table 5: Most and Least 5 Monthly Cases by County in the United States till January 31st 2023

FIPS code	State	County	Month Index	Type	Cases
19021	IA	buena vista county	3	poultry	5486700
19143	IA	osceola county	3	poultry	5011700
42071	PA	lancaster county	4	poultry	3782700
39039	OH	defiance county	9	poultry	3748500
55055	WI	jefferson county	3	poultry	2750700
...	...	...	...	...	...
56037	WY	sweetwater county	5	wild bird	1
56039	WY	teton county	10	wild bird	1
56039	WY	teton county	6	wild bird	1
56039	WY	teton county	9	wild bird	1
56043	WY	washakie county	13	wild bird	1

Table 5 shows the most and least 5 monthly cases by county in the United States till January 31st 2023. We can see that the county that has the most monthly cases was Buena Vista, Iowa with 5,486,700 in March 2022. Its outbreak type is `poultry`. The 5 counties that has the lease monthy cases are all in Wyoming with only 1 case each.

Table 6: Most and Least 5 Cumulative Cases by County in the United States till January 31st 2023

FIPS code	State	County	Cases
8123	CO	weld county	6188782
19021	IA	buena vista county	5606301
19143	IA	osceola county	5011700
42071	PA	lancaster county	3855188
39039	OH	defiance county	3748500
...	...	...	...
55127	WI	walworth county	1
55135	WI	waupaca county	1

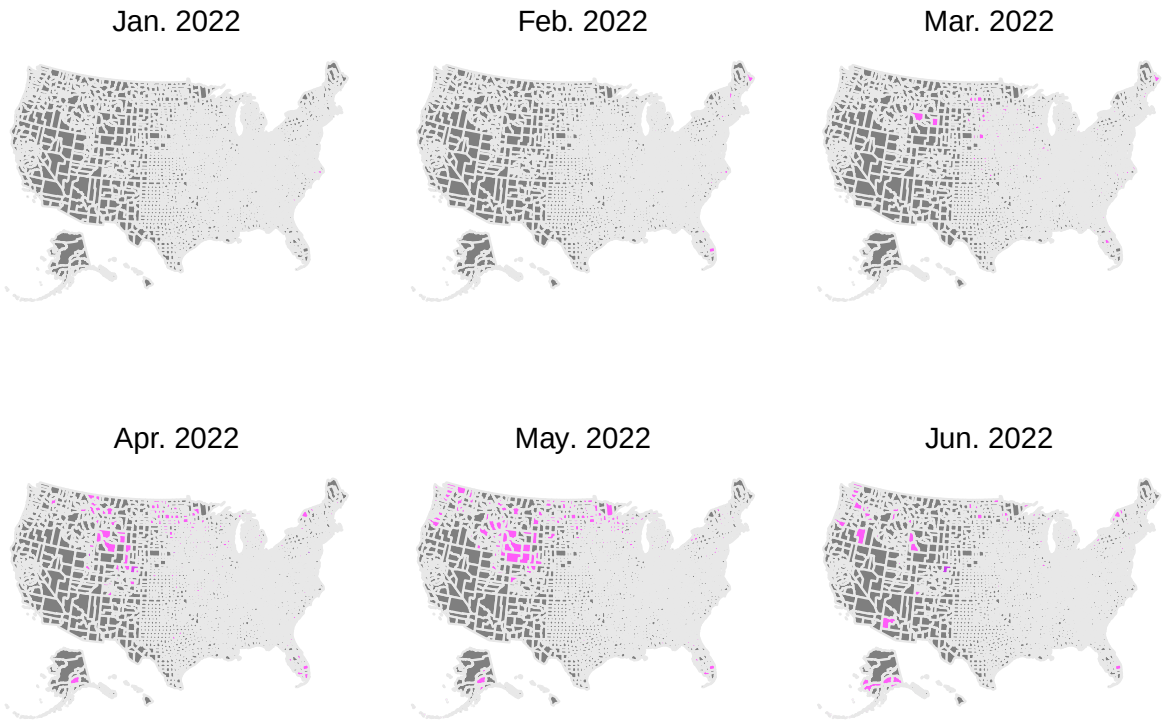
FIPS code	State	County	Cases
56003	WY	big horn county	1
56037	WY	sweetwater county	1
56043	WY	washakie county	1

Table 6 shows the most and least 5 cumulative cases by county in the United States till January 31st 2023. We can see that the county has the most cumulative cases is Weld, Colorado with 6,188,782 cases. Notice that Buena Vista and Osceola in Iowa also have a lot of cumulative cases, 5,606,301 and 5,011,700 respectively. The counties have the least 5 cumulative cases are all in Wisconsin and Wyoming with only 1 case each.

III. Visualization

It is important to visualize the data to understand the patterns and trends that are present in the datasets before building models and doing analyses.

Figure 1: New Cases each Month by County from January to December 2022



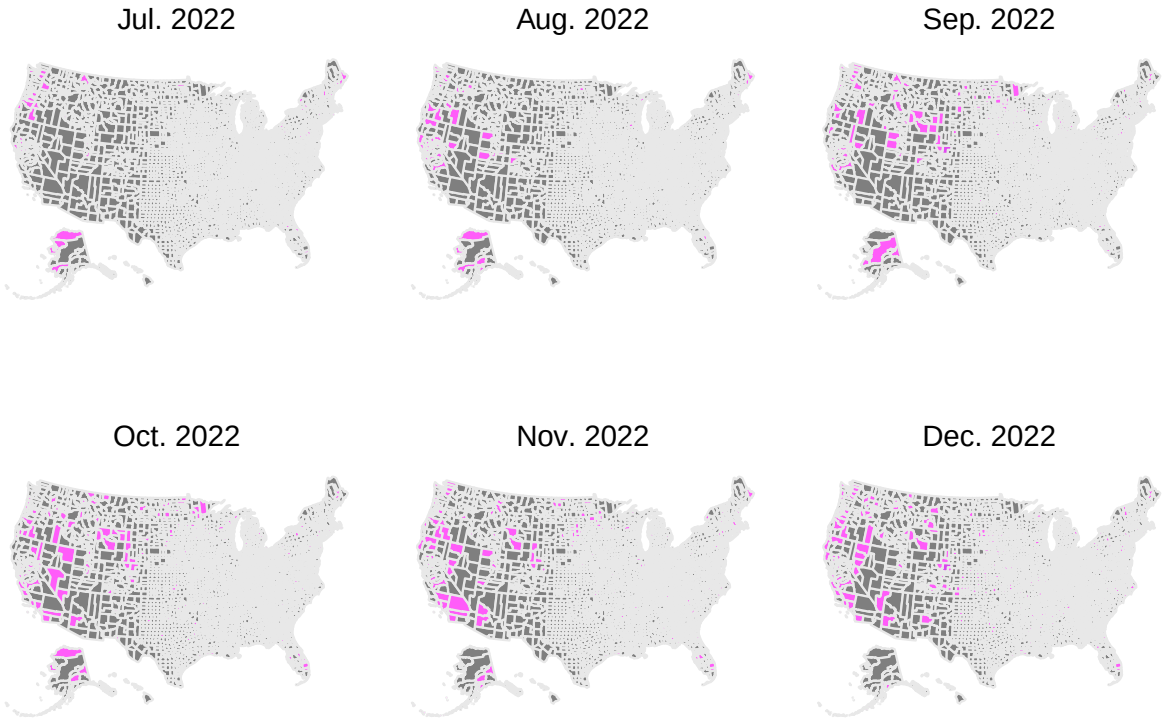


Figure 1 indicates that as the months went by, there were more new cases of the H5N1 virus. The majority of the new cases were in the west and midwest regions. There was a fluctuation of new cases in April, May, and from August to the end of the year.

Figure 2: New Cases each Month by County in January 2023

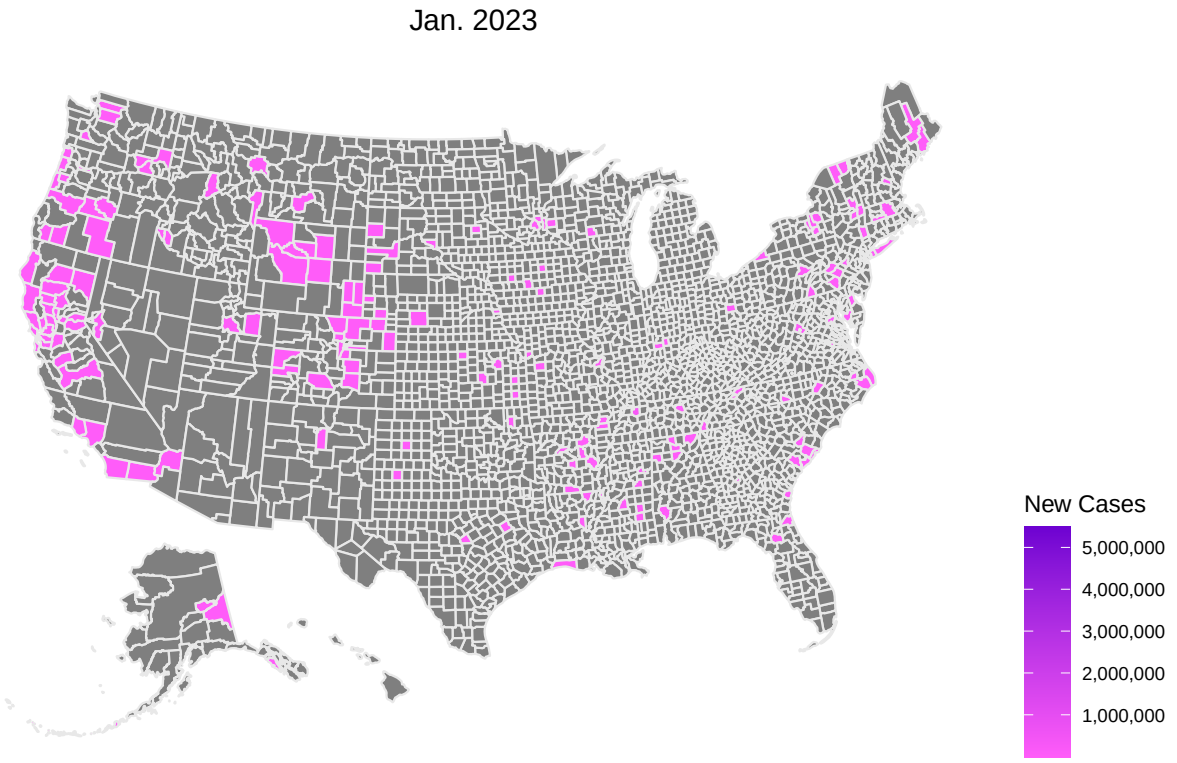
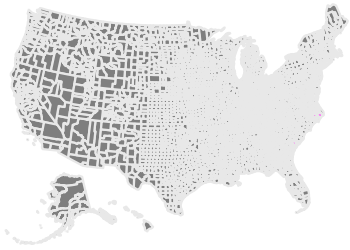


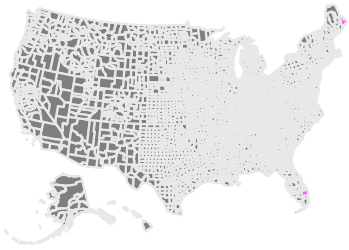
Figure 2 shows the most recent, January 2023, situation of H5N1 across the United States. Based on the colors, new cases did not exceed 1,000,000.

Figure 3: Cumulative Cases each Month by County from January to December 2022

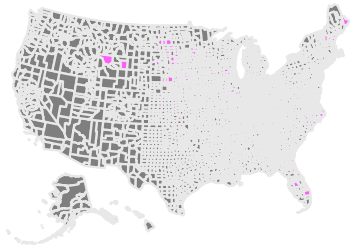
Jan. 2022



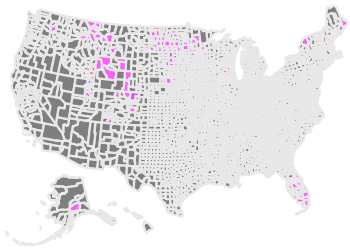
Feb. 2022



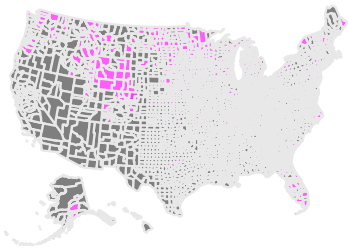
Mar. 2022



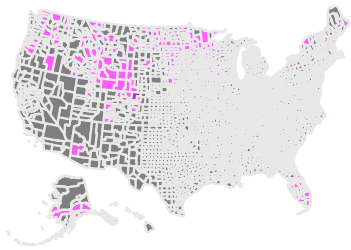
Apr. 2022



May. 2022



Jun. 2022



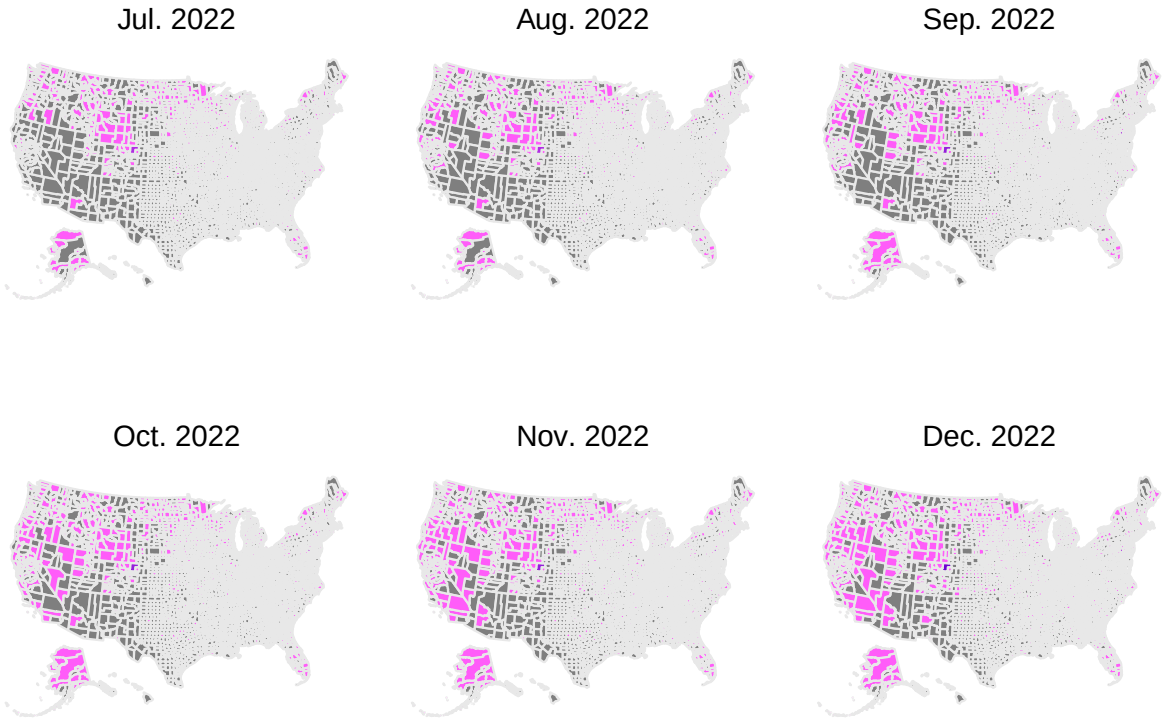
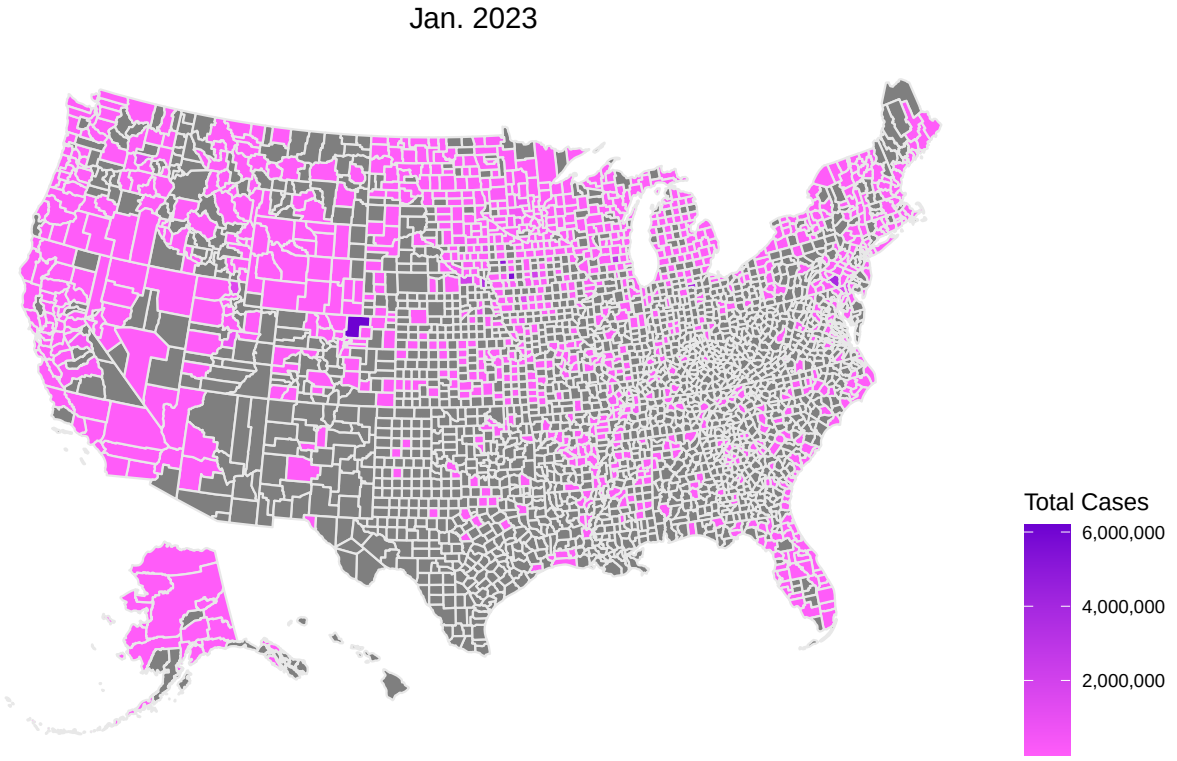


Figure 3 above shows the cumulative cases of the H5N1 virus from January through December 2022.

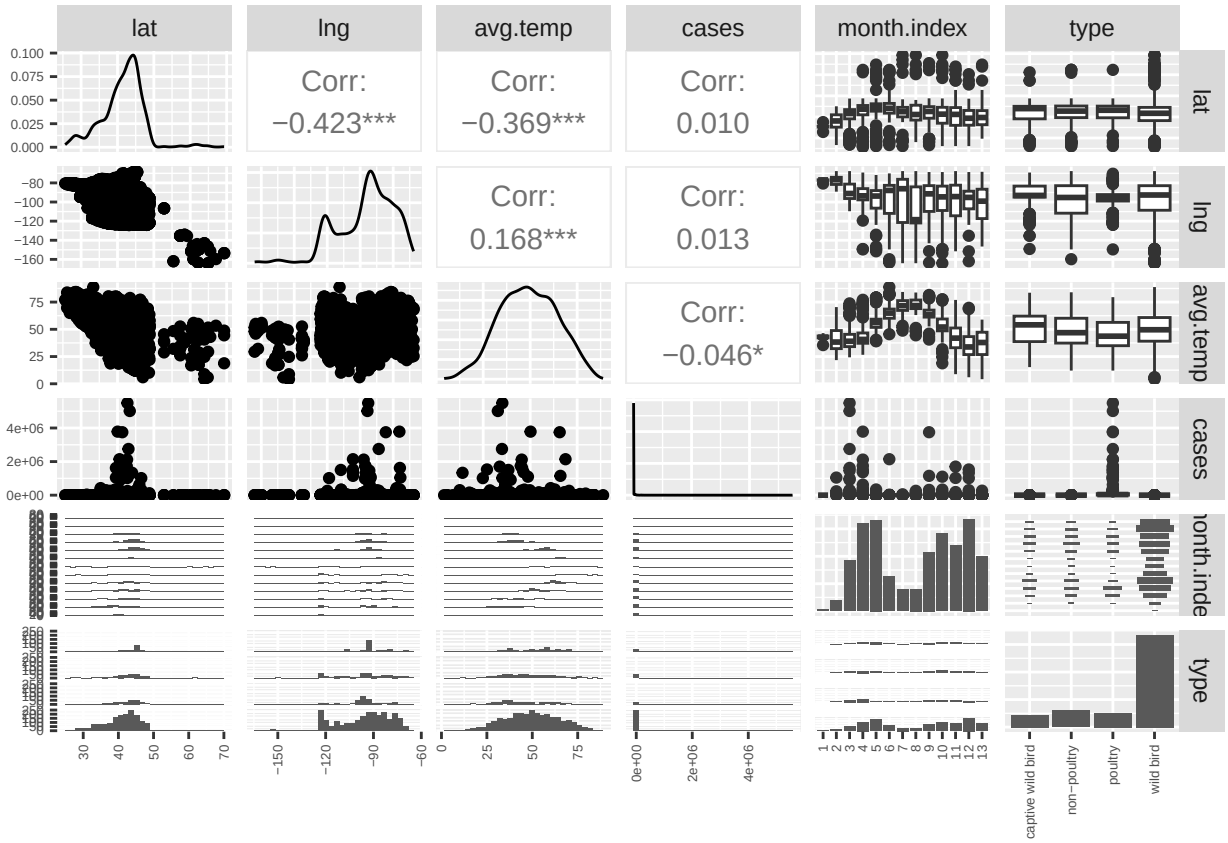
Figure 4: Cumulative Cases each Month by County in January 2023



By the time it is January 2023, most of the United States had cases of this virus. However, most of the cases are fewer than 2,000,000. In the end, there are 9 counties that have cumulative cases greater than 2,000,000 as we can see in Figure 4.

Figure 5: Scatterplot Matrix

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



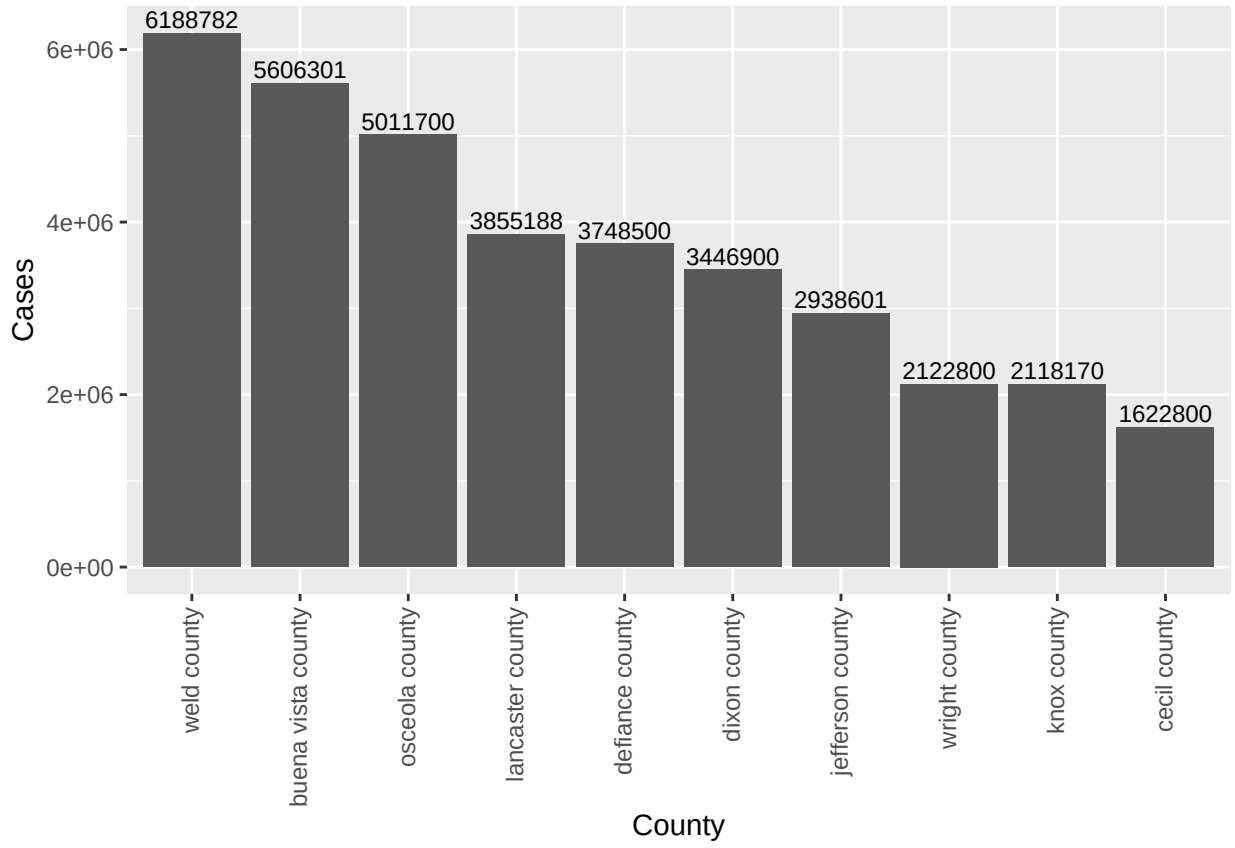
In figure 5, the x-axis represents the variables of the columns and the y-axis represents the variables of the rows.

Among the numerical variables, latitude and longitude have the highest correlation. The correlation of -0.423 indicates that these two variables have a moderate negative relationship with each other. When the latitude is between 35 and 45, there is a surge in cases. Meanwhile, when longitude is between -120 and -70, cases increase dramatically. They indicate the locations in the United States that have a larger number of cases.

The type variable shows that most of the H5N1 cases in the dataset are wild birds.

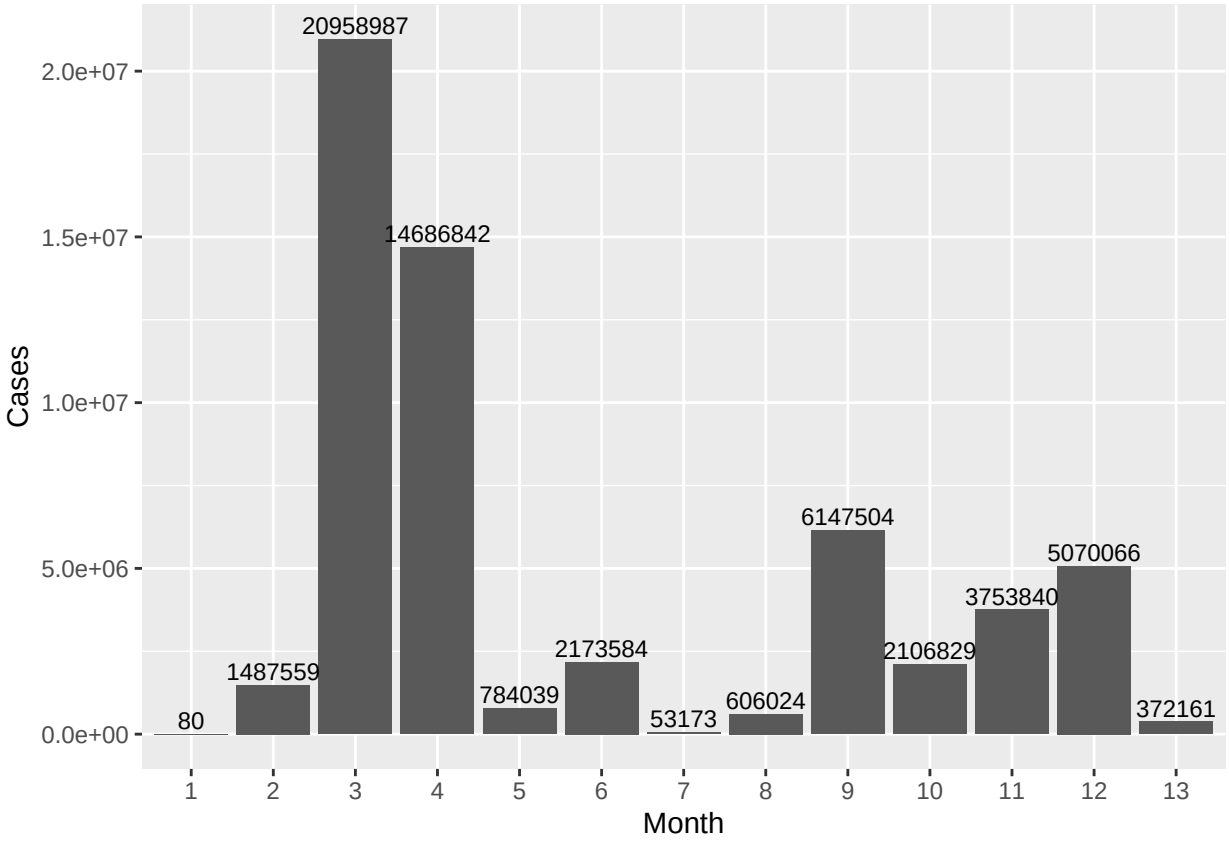
Furthermore, the cases variable indicates that there are not a lot of cases in each outbreak, yet there are many outliers. Possible reasons for this are that although wild birds make up most of the dataset, poultry are in large groups while wild birds are not. Since viruses spread more easily through close contact, most of the cases are poultry.

Figure 6: Bar Chart of Top 10 Counties with the Most Cumulative Cases



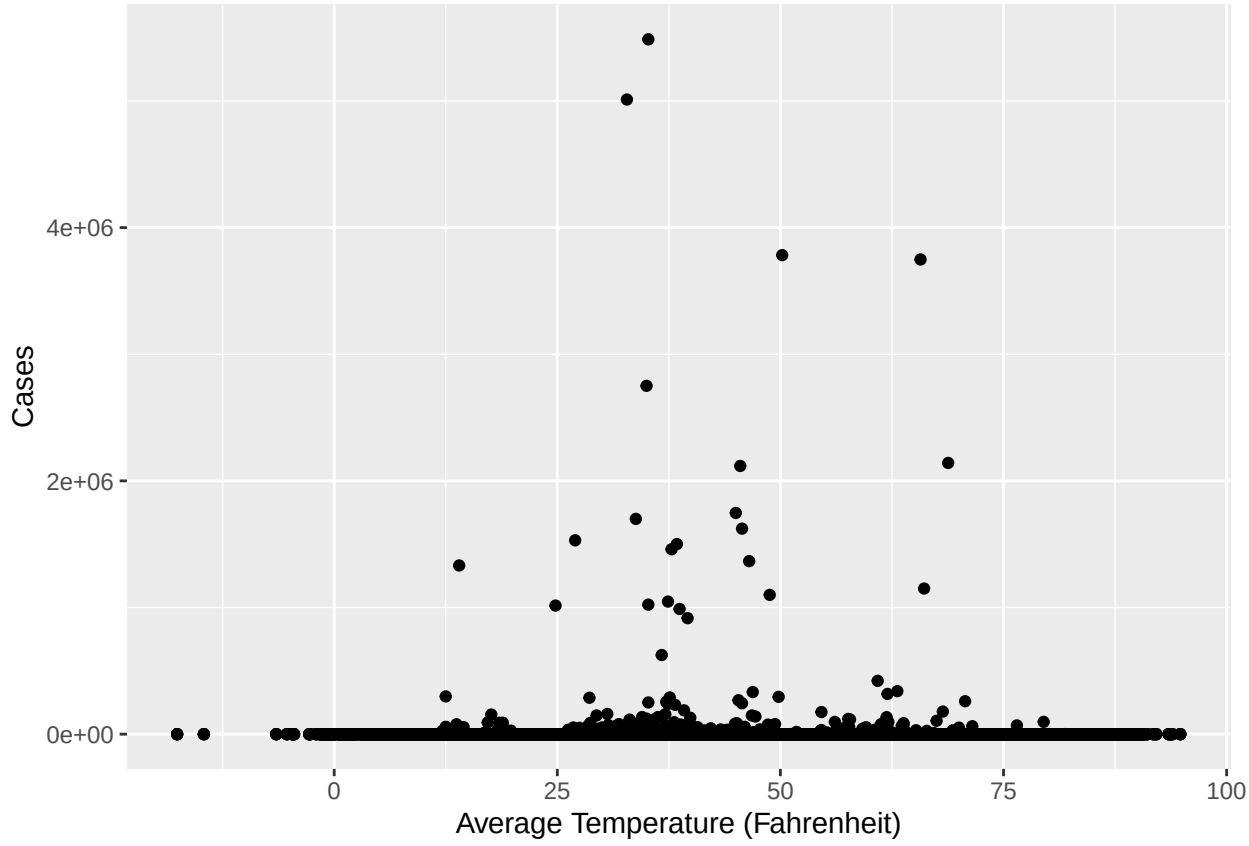
The bar chart as figure 6 shown above indicates the top 10 counties with the most cases of H5N1 from January 2022 to January 2023.

Figure 7: Bar Chart of H5N1 Cases Each Month



This bar chart as figure 7 shown above indicates the number of cases in each month. There is a surge of cases in March and April in 2022.

Figure 8: Scatter Plot of Cases Each Month against Average Monthly Temperature



This scatter plot (figure 8) presents the number of cases against the average temperature for each county in each month. There are more cases when the temperature is between 25 and 50 degrees Fahrenheit.

IV. Modeling and Interpretation

We use the outbreaks in 2022 as our training set, which has 150,864 observations. Moreover, we let the outbreaks happened in January 2023 as the testing set, which has 12,572 observations. The testing set will tell us how well our model performs on predicting which county will have H5N1 cases.

Our model is

$$\begin{aligned} Y =& \beta_0 + \beta_1 X_{\text{lat}} + \beta_2 X_{\text{lng}} + \beta_3 X_{\text{month.index}} + \beta_4 X_{\text{type(non-poultry)}} \\ &+ \beta_5 X_{\text{type(poultry)}} + \beta_6 X_{\text{type(wild bird)}} + \beta_7 X_{\text{avg.temp}} + \beta_8 X_{\text{lat * lng}}, \end{aligned} \tag{1}$$

where β_0 is the intercept of the model, β_1 to β_8 are the coefficients of explanatory variables X_{lat} to $X_{\text{lat * lng}}$.

Moreover, we need to use the sigmoid function

$$p(\boldsymbol{X}) = \frac{1}{1 + e^{-\boldsymbol{X}\boldsymbol{\beta}}},$$

where

$$\boldsymbol{X} = \begin{bmatrix} 1 & X_{1\text{ lat}} & X_{1\text{ lng}} & \cdots & X_{1\text{ lat * lng}} \\ 1 & X_{2\text{ lat}} & X_{2\text{ lng}} & \cdots & X_{2\text{ lat * lng}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{150864\text{ lat}} & X_{150864\text{ lng}} & \cdots & X_{150864\text{ lat * lng}} \end{bmatrix}_{150864\times 9}, \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_8 \end{bmatrix}_{9\times 1}.$$

The sigmoid function guarantees that the predicted probability is in the range $(0, 1)$ and hence allows us to obtain a sensible prediction.

Because this model determines whether a county will have H5N1 case(s) based on each outbreak type, so it is a binary classifier.

Since the result is either `infected` or `uninfected`, so for each observation, its distribution is a Bernoulli Distribution

$$\text{Bern}(p).$$

Since we have 150,864 observations in the training set, the distribution of Y should be a Binomial Distribution

$$\text{Binomial}(150864,\ p).$$

p is the probability of a county, based on each outbreak type, to have H5N1 case(s), which will be obtained by performing the model with the sigmoid function described above.

Now let us start build models to predict potential H5N1 outbreak(s) in the future.

1. Logistic Regression Model

i. Modeling

We decided to use logistic regression to model our data with the response variable as to whether a specific county will get infected by avian influenza.

The log-likelihood function for logistic regression is

$$\ell(\beta) = \sum_{i=1}^{n} Y_i \log(p(X_i)) + (1 - Y_i) \log(1 - p(X_i)),\ n = 150864.$$

In order to estimate the parameters of the logistic regression model, we will apply the method of Maximum Likelihood Estimate (MLE), which solves the objective function

$$\hat{\beta} = \arg\max_{\beta} \Big[\sum_{i=1}^{n} Y_i \log(p(X_i)) + (1 - Y_i) \log(1 - p(X_i)) \Big],\ n = 150864.$$

By performing the logistic regression with the MLE method, we get our estimated coefficients, rounded to five decimals, as table 7 shown below.

Table 7: Estimated Coefficients Generated by Logistic Regression Model

Coefficient	Estimation
beta.0	17.95338
beta.1	-0.28498
beta.2	0.07552
beta.3	-0.09589
beta.4	-0.30772
beta.5	-0.19225
beta.6	-2.05547
beta.7	0.00087
beta.8	-0.00169

As a result, our predicted model is (coefficients rounded to three decimal places)

$$\begin{aligned}\hat{Y} =& 17.953 - 0.285X_{\text{lat}} + 0.076X_{\text{lng}} - 0.096X_{\text{month.index}} - 0.308X_{\text{type(non-poultry)}} \\ & - 0.192X_{\text{type(poultry)}} - 2.055X_{\text{type(wild bird)}} + 0.001X_{\text{avg.temp}} - 0.002X_{\text{lat * lng}}.\end{aligned} \tag{2}$$

ii. Interpretation

The coefficient for `lat` (-0.28498) indicates that a one-unit increase in latitude is associated with a negative change in probability that `lat` multiple by 0.7520293, holding all other predictor variables constant.

The coefficient for `lng` (0.07552) indicates that a one-unit increase in longitude is associated with a positive change in probability that `lng` multiplies by 1.0784448, holding all other predictor variables constant.

The coefficient for `month.index` (-0.09589) indicates that a one-unit increase in `month.index` (which represents the month of the year) is associated with a negative change in probability that month multiplies by 0.908564, holding all other predictor variables constant.

The coefficient for each `type` of outbreak (`poultry`, `non-poultry`, `wild bird`) represents the difference in log odds of the outcome compared to the reference category (in this case, wild bird). The coefficient for `non-poultry` (-0.30772) indicates that `non-poultry` animals are 0.7351211 times less likely than the `wild birds`, holding all other predictor variables constant.

The coefficient for `avg.temp` (8.7×10^{-4}) indicates that a one-unit increase in average temperature is associated with a positive change in the log odds of the outcome by 1.0008704, holding all other predictor variables constant.

The coefficient for the interaction term `lat * lng` (-0.00169) indicates that the effect of latitude on the log odds of the outcome depends on the value of longitude. Specifically, a one-unit increase in latitude is associated with a negative change in the log odds of the outcome by 0.9983114 units for each one-unit increase in longitude.

Overall, this logistic regression model can be used to predict the probability of the binary outcome based on the values of the predictor variables included in the model. The estimated coefficients can also be used to interpret the effects of each predictor variable on the log odds of the outcome, holding all other predictor variables constant.

Because the residual deviance of the logistic regression model is $\sum_{i=1}^{n} d_i^2 = \sum_{i=1}^{n} 2\left[Y_i \log\left(\frac{Y_i}{p(X)}\right) + (1 - Y_i)\log\left(\frac{1-Y_i}{1-p(X)}\right)\right] = 22231$, which is too large, we want to penalize our logistic regression model, in the next two sections, to possibly achieve a better performance on detecting the H5N1 cases in a specific county.

2. Ridge Regression Model for Classification

i. Modeling

In this section, we will try the ridge regression model for classification. Ridge regression attempt to solve the objective function

$$\hat{\beta}^{\text{ridge}} = \arg\max_{\beta} \Big[\ell(\beta) + \lambda \sum_{i=1}^{n} \beta_i^2\Big],\ n = 150864.$$

where $\ell(\beta)$ is the loss function of the original logistic regression model, $\lambda \sum_{i=1}^{n} \beta_i^2$ is the penalty term. λ is called the penalty parameter and $\lambda \in [0, \infty)$.

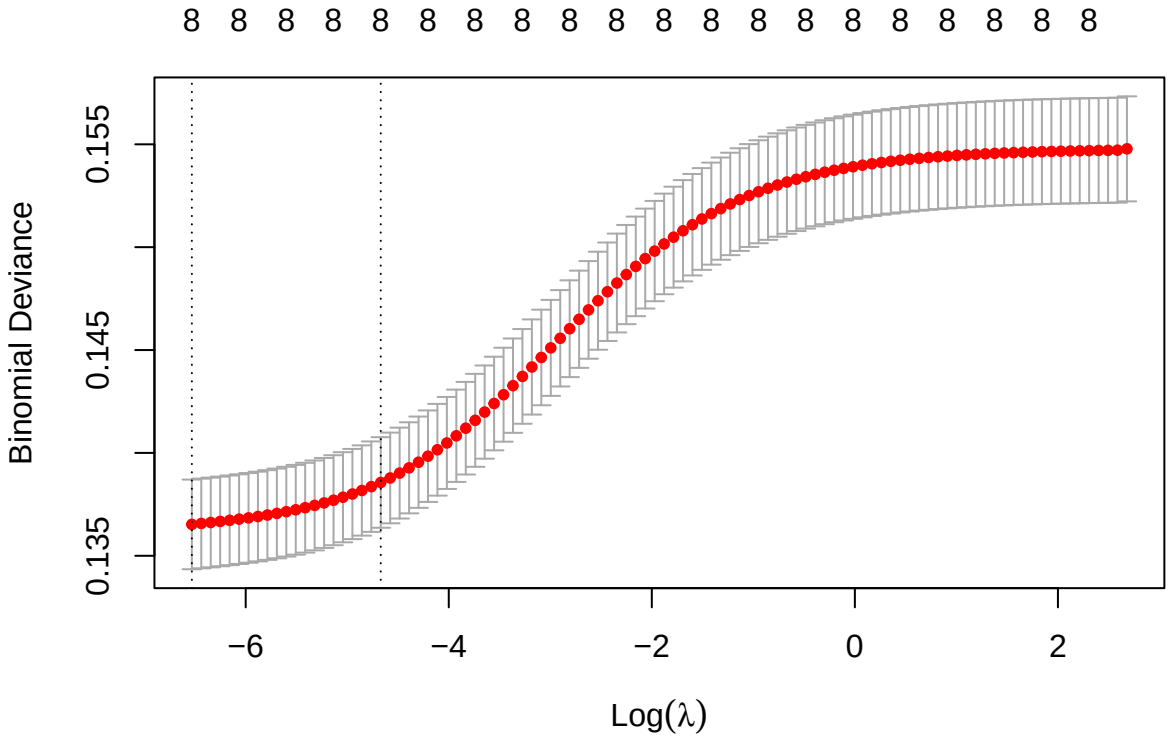
Since we need to find λ to do the ridge regression model, we first use a 5-fold cross validation to find out the λ that yields the smallest deviance

$$CV_{(5)} = \frac{1}{5}\sum_{i=1}^{n} d_i^2 = \frac{1}{5}\sum_{i=1}^{n} 2\left[Y_i \log\left(\frac{Y_i}{p(X)}\right) + (1 - Y_i)\log\left(\frac{1 - Y_i}{1 - p(X)}\right)\right],\ n = 150864.$$

Figure 9 below shows different deviances when using different λ's. The dashline on the left side indicates the λ (0.0014585) producing the smallest deviance (0.1365252). This λ is the one we want for our penalty term to avoid extreme selection (i.e. $\hat{\beta}_i$ becomes 0).

The dashline on the right side of the first one indicates the largest λ (0.009375) at which the deviance is within one standard error of the smallest deviance.

Figure 9: Cross Validation for Proper λ in Ridge Regression Model



After performing the 5-fold cross validation and getting the value of λ we want, we can start building the ridge regression model for classification, which gives us the estimated coefficients as shown below in table 8.

Table 8: Estimated Coefficients Generated by Ridge Regression Model

Coefficient	Estimation
beta.0	-9.02613
beta.1	0.09103
beta.2	-0.00266
beta.3	0.08643
beta.4	0.01607
beta.5	-0.08307
beta.6	1.71345
beta.7	-0.00313
beta.8	6e-05

As a result, our predicted model becomes (coefficients rounded to three decimal places)

$$\begin{aligned}\hat{Y} = &- 9.026 + 0.091X_{\text{lat}} - 0.003X_{\text{lng}} + 0.086X_{\text{month.index}} + 0.016X_{\text{type(non-poultry)}} \\ &- 0.083X_{\text{type(poultry)}} + 1.713X_{\text{type(wild bird)}} - 0.003X_{\text{avg.temp}} + 6 \times 10^{-5}X_{\text{lat * lng}}.\end{aligned} \tag{3}$$

ii. Interpretation

The coefficient for `lat` (0.09103) indicates that a one-unit increase in latitude is associated with a positive change in probability that `lat` multiple by 1.0953019, holding all other predictor variables constant.

The coefficient for `lng` (-0.00266) indicates that a one-unit increase in longitude is associated with a negative change in probability that `lng` multiplies by 0.9973435, holding all other predictor variables constant.

The coefficient for `month.index` (0.08643) indicates that a one-unit increase in `month.index` (which represents the month of the year) is associated with a positive change in probability that month multiplies by 1.090275, holding all other predictor variables constant.

The coefficient for each `type` of outbreak (`poultry`, `non-poultry`, `wild bird`) represents the difference in log odds of the outcome compared to the reference category (in this case, wild bird). The coefficient for `non-poultry` (0.01607) indicates that `non-poultry` animals are 1.0161998 times more likely than the `wild birds`, holding all other predictor variables constant.

The coefficient for `avg.temp` (-0.00313) indicates that a one-unit increase in average temperature is associated with a negative change in the log odds of the outcome by 0.9968749, holding all other predictor variables constant.

The coefficient for the interaction term `lat * lng` (6×10^{-5}) indicates that the effect of latitude on the log odds of the outcome depends on the value of longitude. Specifically, a one-unit increase in latitude is associated with a positive change in the log odds of the outcome by 1.00006 units for each one-unit increase in longitude.

3. Lasso Regression Model for Classification

i. Modeling

In this section, we will try the lasso regression model for classification. Lasso regression attempt to solve the objective function

$$\hat{\beta}^{\text{lasso}} = \arg\max_{\beta}\Big[\ell(\beta) + \lambda\sum_{i=1}^{n}|\beta_i|\Big],\ n = 150864.$$

where $\ell(\beta)$ is the loss function of the original logistic regression model, $\lambda\sum_{i=1}^{n}|\beta_i|$ is the penalty term. λ is called the penalty parameter and $\lambda \in [0, \infty)$.

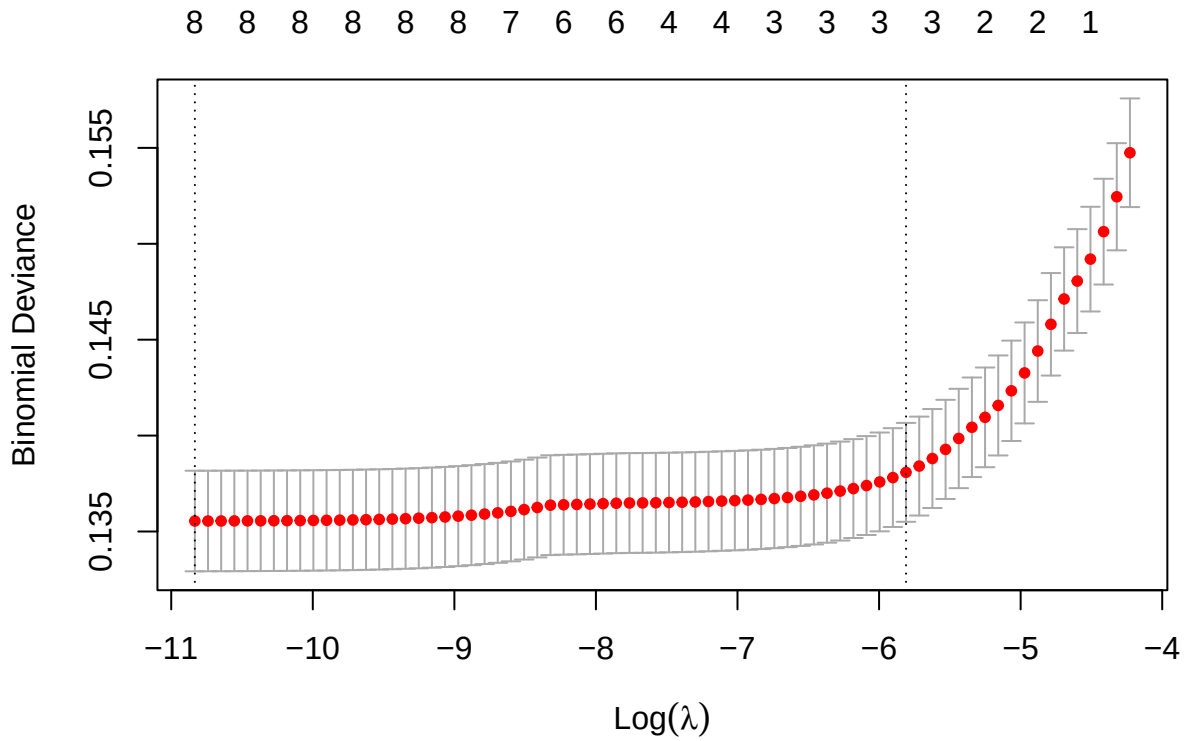
Since we need to find λ to do the ridge regression model, we first use a 5-fold cross validation to find out the λ that yields the smallest deviance

$$CV_{(5)} = \frac{1}{5}\sum_{i=1}^{n} d_i^2 = \frac{1}{5}\sum_{i=1}^{n} 2\left[Y_i \log\left(\frac{Y_i}{p(X)}\right) + (1-Y_i)\log\left(\frac{1-Y_i}{1-p(X)}\right)\right],\ n = 150864.$$

Figure 10 below shows different deviances when using different λ's. The dashline on the left side indicates the λ (1.9733578×10^{-5}) producing the smallest deviance (0.1355459). This λ is the one we want for our penalty term to avoid extreme selection (i.e. $\hat{\beta}_i$ becomes 0).

The dashline on the right side of the first one indicates the largest λ (0.0029993) at which the deviance is within one standard error of the smallest deviance.

Figure 10: Cross Validation for Proper λ in Lasso Regression Model



After performing the 5-fold cross validation and getting the value of λ we want, we can start building the lasso regression model for classification, which gives us the estimated coefficients as shown below in table 9.

Table 9: Estimated Coefficients Generated by Lasso Regression Model

Coefficient	Estimation
beta.0	-17.04739
beta.1	0.26509
beta.2	-0.06755
beta.3	0.09541
beta.4	0.28437
beta.5	0.16841
beta.6	2.0351
beta.7	-0.00096
beta.8	0.00151

As a result, our predicted model becomes (coefficients rounded to three decimal places)

$$\begin{aligned}\hat{Y} = &-17.047 + 0.265X_{\text{lat}} - 0.068X_{\text{lng}} + 0.095X_{\text{month.index}} + 0.284X_{\text{type(non-poultry)}} \\ &+ 0.168X_{\text{type(poultry)}} + 2.035X_{\text{type(wild bird)}} - 0.001X_{\text{avg.temp}} + 0.002X_{\text{lat * lng}}.\end{aligned} \tag{4}$$

ii. Interpretation

The coefficient for `lat` (0.26509) indicates that a one-unit increase in latitude is associated with a positive change in probability that `lat` multiple by 1.3035483, holding all other predictor variables constant.

The coefficient for `lng` (-0.06755) indicates that a one-unit increase in longitude is associated with a negative change in probability that `lng` multiplies by 0.934681, holding all other predictor variables constant.

The coefficient for `month.index` (0.09541) indicates that a one-unit increase in `month.index` (which represents the month of the year) is associated with a positive change in probability that month multiplies by 1.1001098, holding all other predictor variables constant.

The coefficient for each `type` of outbreak (`poultry`, `non-poultry`, `wild bird`) represents the difference in log odds of the outcome compared to the reference category (in this case, wild bird). The coefficient for `non-poultry` (0.28437) indicates that `non-poultry` animals are 1.3289245 times more likely than the `wild birds`, holding all other predictor variables constant.

The coefficient for `avg.temp` (-9.6×10^{-4}) indicates that a one-unit increase in average temperature is associated with a negative change in the log odds of the outcome by 0.9990405, holding all other predictor variables constant.

The coefficient for the interaction term `lat * lng` (0.00151) indicates that the effect of latitude on the log odds of the outcome depends on the value of longitude. Specifically, a one-unit increase in latitude is associated with a positive change in the log odds of the outcome by 1.0015111 units for each one-unit increase in longitude.

V. Analysis

Now we have three models to determine whether a county will have a specific type of outbreak in the future, which are logistics regression model, ridge regression model for classification, and lasso regression model for classification. The best way to see how these model performs is to use the testing set to see the accuracy of their predictions.

We plug in the testing set, which are the cases happened in January 2023, into our three models and find out that all the models produce the exact same result with the threshold probability of `infected` being 0.5 (i.e. when the predicted probability is larger than 0.5, it is classified as `infected`, otherwise `uninfected`).

These models indicate that all counties are `uninfected`. The confusion matrix is shown as 10 below, which represents the counts of all combination of values between the predicted label and the true label.

Table 10: Confusion Matrix of the Test Set

Predicted/True	Infected	Uninfected
Uninfected	198	12374

By looking at the confusion matrix, we get error rate equals to $\frac{198}{12572} = 0.01574928$, and the accuracy equals to $1 - \frac{198}{12572} = 0.9842507$, which means that most of the cases are correctly classified.

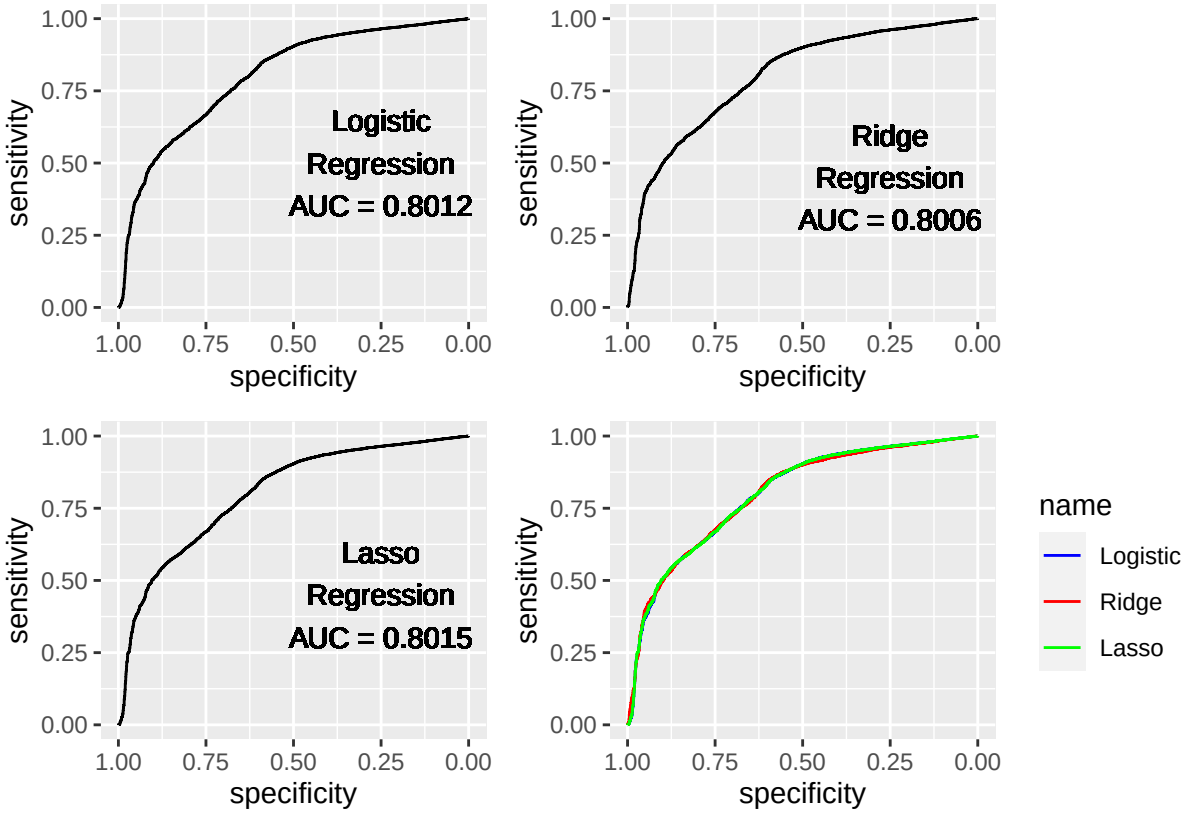
Although the accuracy is very high, but that is not what we really want because all these models classify all the 3,143 counties as `uninfected` in January 2023, whereas there were 198 counties had outbreaks, which cannot bring us any useful information and may bring risks to public health. Table 11 shows the top 10 outbreak cases across the United States in January 2023.

Table 11: Top 10 Outbreak Cases in January 2023

FIPS Code	State	County	Type	Cases
47183	TN	weakley county	poultry	267800
51165	VA	rockingham county	poultry	36000
19021	IA	buena vista county	poultry	27700
6103	CA	tehama county	poultry	23700
20003	KS	anderson county	poultry	8900
20123	KS	mitchell county	poultry	6900
46029	SD	codington county	poultry	140
53067	WA	thurston county	non-poultry	120
8069	CO	larimer county	non-poultry	70
48281	TX	lampasas county	poultry	70

In order to know which model performs better, we use Receiver Operating Characteristic (ROC) curve, which tests the goodness of fit, and compare the Area Under the Curve (AUC). The range of AUC is $(0, 1)$, where higher AUC means the classifier is better. Figure 11 below shows the ROC curves and AUC of logistic regression, ridge regression, and lasso regression models for classification.

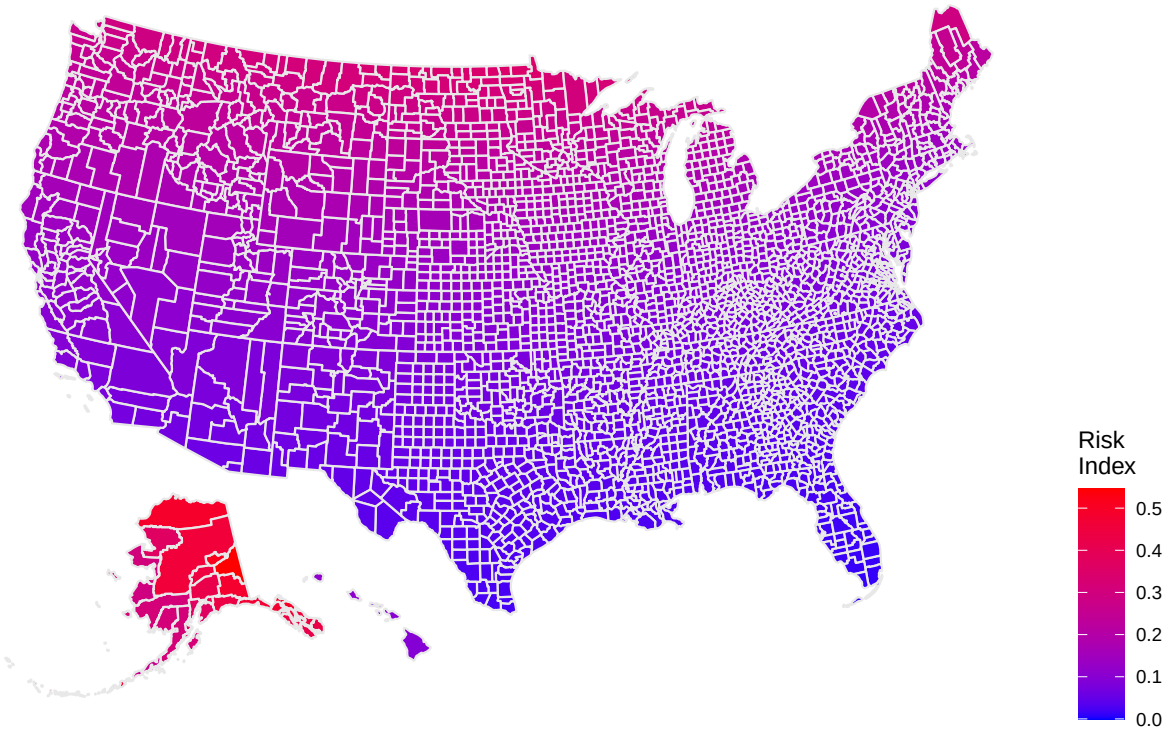
Figure 11: ROC Curves of All Three Models



We can see the all ROC curves look smooth and goes to the left corner, meaning that our classifier works good. By obtaining the value of AUC's, which are 0.8012 for logistic regression model, 0.8006 for ridge regression model, and 0.8015 for lasso regression model, indicating that most counties are correctly classified as `infected` and `uninfected`. Our detection of H5N1 looks good.

Moreover, since the AUC for the lasso regression model is the highest, although the difference is not very large, we think the lasso regression model for classification works the best among all the three models. Figure 12 is a map, which plots out the sum of predicted probabilities of all four possible H5N1 outbreak types (`poultry`, `non-poultry`, `wild bird`, and `captive wild bird`) of all counties in January 2023 calculated by the lasso regression model. Note that redder color means higher risk to have an outbreak. We can interpret the sum of probability of four types of a county as its risk index, with range $[0, 4]$, of having an outbreak.

Figure 12: Map of Risk Index Generated by Lasso Regression Model



As we can see from figure 12, seem the counties in the south are less likely to have outbreaks of H5N1 in January, but those counties in the north and west are more risky. This actually makes sense if we compare with Figure 2: New Cases each Month by County in January 2023.

VI. Conclusion and Suggestion

In conclusion, we have developed three models, logistic regression, ridge regression, and lasso regression, to predict the likelihood of H5N1 outbreaks in different counties of the United States. The models were trained using historical data in 2022, and the results showed that all models had a high accuracy in predicting the occurence of H5N1 outbreaks in January 2023, which is about 98.4%. However, the models failed to predict the occurrence of 198 H5N1 outbreaks in January 2023, if we set the threshold of `infected` to be 0.5, which highlights a limitation in our current models.

Our analyses show that the lasso regression model performed the best among the three models, with an AUC of 0.8015. The map generated based on the lasso regression model indicated that counties in the north and west were at a higher risk of having H5N1 outbreaks in January 2023, which matches the actual result.

Despite the high accuracy of our models, there are still some limitations that need to be addressed. One limitation is that our models only considered the effects of temperature, outbreak types, time, and density on the likelihood of H5N1 outbreaks, but there may be other factors that also affect the spread of the virus, such as migration patterns of birds, human movement, egg production, breeding size, and so on. In addition, the models were based on historical data, and the emergence of new H5N1 strains may lead to changes in the spread of the virus that are not accounted for in our models.

To improve our models, we could incorporate additional data sources such as bird migration patterns, human travel data, egg production, breeding size, and so on. We could also use more sophisticated machine learning

techniques such as deep learning to capture more complex relationships between different variables. We suggest the USDA and CDC to make more detailed data collections, including these factors described above.

In terms of controlling the spread of H5N1, there are several measures that can be taken.

First, it is essential to implement strict biosecurity measures in poultry farms to prevent the spread of the virus. This includes regular cleaning and disinfection of poultry houses, limiting human and vehicle traffic in and out of the farm, and separating sick birds from healthy ones. Preventing the spread of diseases among birds is crucial and can be achieved by implementing the following measures: limiting access to your property, keeping your birds away from other bird species, maintaining cleanliness by washing hands, disinfecting equipment, and handling dead birds properly. Buying healthy birds from reputable sources and keeping them separate for 30 days can also help. It is essential to sanitize equipment and supplies before sharing them with others and to be aware of warning signs of illness in birds. Any sick or dying birds should be reported immediately to the relevant authorities, such as the Cooperative Extension office or a veterinarian, to prevent further spread of disease. Early detection and reporting can help keep birds healthy and prevent the spread of avian diseases.

Moreover, chicken farmers are close contacts of poultry. In order to prevent chicken farmers from being infected with avian influenza, they must develop good hygienic habits. It is best to wear masks and work clothes when working to reduce the chance of direct contact with chickens. The work clothes should be cleaned, disinfect. Wash hands after contact with dirt, and wear gloves when handling manure from chicken farms. When an epidemic occurs, minimize contact with poultry, and wear gloves, masks, and protective clothing when touching poultry.

Second, surveillance programs should be implemented to monitor the spread of the virus in wild birds and poultry farms. This will help identify outbreaks early and prevent further spread of the virus. Early detection of disease can help prevent its spread. Keep an eye on your birds and look for signs of illness or distress. While it can be difficult to identify avian flu, checking your birds frequently can help you identify any issues. If your bird is sick or dying, report it immediately to your local Cooperative Extension office, veterinarian, or state animal/bird diagnostic practice laboratory. Call USDA toll-free at 1-866-536-7593 to find a local contact who can help you. Early reporting can help prevent the spread of disease and protect the health of other birds.

Third, public education campaigns should be launched to raise awareness of the risks of H5N1 and to educate people on how to prevent the spread of the virus.

Finally, there should be coordinated efforts at the national and international level to track and contain the spread of the virus, including sharing information and resources across different countries.

By implementing these measures, we can help control the spread of H5N1 and prevent future outbreaks.

Reference

United States Department of Agriculture (USDA) (2022). Per capita availability of chicken higher than that of beef since 2010. USDA ERS - Chart Detail. Retrieved February 27, 2023, from https://www.ers.usda.gov/data-products/chart-gallery/gallery/chart-detail/?chartId=58312

United Egg Producers (UEP) (2021, March 10). Facts & stats. United Egg Producers. Retrieved February 27, 2023, from https://unitedegg.com/facts-stats/

Centers for Disease Control and Prevention (CDC) (2023, February 22). H5N1 bird flu detections across the United States (backyard and commercial). Centers for Disease Control and Prevention. Retrieved February 27, 2023, from https://www.cdc.gov/flu/avianflu/data-map-commercial.html

Centers for Disease Control and Prevention (CDC) (2023, February 15). H5N1 Bird Flu Detections across the United States (Wild Birds). Centers for Disease Control and Prevention. Retrieved February 27, 2023, from https://www.cdc.gov/flu/avianflu/data-map-wild-birds.html

Iacurci, G. (2023, February 8). Wholesale egg prices have 'collapsed.' why consumers may soon see relief. CNBC. Retrieved February 27, 2023, from https://www.cnbc.com/2023/02/07/wholesale-egg-prices-have-collapsed-from-record-highs-in-december.html

World Health Organization (WHO) (2018, November 13). Influenza (avian and other zoonotic). World Health Organization. Retrieved February 27, 2023, from https://www.who.int/news-room/fact-sheets/detail/influenza-(avian-and-other-zoonotic)

World Organisation for Animal Health (WOAH) (2022, May 6). Health Standards Glossary - WOAH. World Organisation for Animal Health. Retrieved February 27, 2023, from https://www.woah.org/fileadmin/Home/eng/Health_standards/tahc/current/glossaire.pdf

Appendix: R Script

```
knitr::opts_chunk$set(echo = TRUE)
rm(list=ls())
set.seed(77)
library(dplyr)
library(ggplot2)
library(tidyr)
library(usmap)
library(ggpubr)
library(grid)
library(gridExtra)
library(patchwork)
library(sf)
library(knitr)
library("imputeTS")
library(textstem)
library(GGally)
library(pROC)
library(lmtest)
library(car)
library(glmnet)
library(psych)
# Runtime control
show.visual = TRUE
# Table number
table.num = 0
# Figure number
figure.num = 0
# Positive & Negative change
change = "positive"
##############################################################################################
##### II. Data Description ###################################################################
##############################################################################################

##############################################################################################
########## Data 1: us.county #################################################################
##############################################################################################
# Import data
us.county = read.csv("https://iasc2023.gd.edu.kg/dataset/uscounties.csv",
                     header = TRUE)
# Change column location
us.county = us.county %>%
  relocate(county, .after = state)
# Change state name to abbreviation
for (i in 1:dim(us.county)[1]){
  if (us.county$state[i] != "District of Columbia"){
    abb = state.abb[grep(us.county$state[i], state.name)]
    us.county$state[i] = abb
  }
}
dc.index = which(us.county$state == "District of Columbia")
us.county$state[dc.index] = "DC"
```

```
# Table 1: First 5 Observations of US Counties Database
table = us.county
colnames(table) = c("FIPS Code", "State", "County", "Latitude",
                    "Longitude")
kable(head(table), row.names = FALSE)
##############################################################################################
########## Data 2: h5n1.poultry.cdc.o ########################################################
##############################################################################################
# Import data
h5n1.poultry.cdc.o = read.csv("https://iasc2023.gd.edu.kg/dataset/hpai-poultry.csv",
                              header = TRUE)
# Ignore cases after January 31, 2023
h5n1.poultry.cdc = h5n1.poultry.cdc.o %>%
  separate(Outbreak.Date, sep="-", into = c("Month", "Day", "Year")) %>%
  unite("Year.Month", c("Year", "Month")) %>%
  filter(Year.Month != "2023_02") %>%
  relocate(County, .after = State)
# Add "County" after the name of each county
h5n1.poultry.cdc$County = paste(h5n1.poultry.cdc$County, "County")
non.poultry = which(h5n1.poultry.cdc$Flock.Type == "WOAH Non-Poultry")
h5n1.poultry.cdc$Flock.Type[non.poultry] = "Non-Poultry"
h5n1.poultry.cdc$Flock.Type[-non.poultry] = "Poultry"
colnames(h5n1.poultry.cdc) = c("state", "county", "year_month",
                               "day", "type", "cases")
# Table 2: First and Last 5 Outbreaks in the United States till
# Jan. 31st 2023 (Backyard and Commercial)
table = h5n1.poultry.cdc %>%
  arrange(year_month, day)
table = headTail(table, 5, 5, ellipsis=TRUE)
table[6, c(1:5)] = "..."
colnames(table) = c("State", "County", "Year Month", "Day",
                    "Type", "Cases")
kable(table, row.names = FALSE)
##############################################################################################
########## Data 3: h5n1.wild.cdc.o ###########################################################
##############################################################################################
# Import data
h5n1.wild.cdc.o = read.csv("https://iasc2023.gd.edu.kg/dataset/hpai-wild-birds.csv",
                           header = TRUE)
# Seprate year month day
h5n1.wild.cdc = h5n1.wild.cdc.o %>%
  separate(Date.Detected, sep="/", into = c("Month", "Day", "Year"))
# Add 0 in front of months that only has a single
single.num.month = which(as.numeric(h5n1.wild.cdc$Month) < 10)
h5n1.wild.cdc$Month[single.num.month] = paste0("0",
                                               h5n1.wild.cdc$Month[single.num.month])
# Add 0 in front of days that only has a single
single.num.day = which(as.numeric(h5n1.wild.cdc$Day) < 10)
h5n1.wild.cdc$Day[single.num.day] = paste0("0",
                                           h5n1.wild.cdc$Day[single.num.day])
# Ignore cases after January 31, 2023
h5n1.wild.cdc = h5n1.wild.cdc %>%
  unite("Year.Month", c("Year", "Month")) %>%
```

```
  filter(Year.Month != "2023_02")
# Add "County" after the name of each county
h5n1.wild.cdc$County = paste(h5n1.wild.cdc$County, "County")
h5n1.wild.cdc = h5n1.wild.cdc[, c(1, 2, 3, 4, 7)]
colnames(h5n1.wild.cdc) = c("state", "county", "year_month",
                            "day", "type")
h5n1.wild.cdc = h5n1.wild.cdc %>%
  count(state, county, year_month, day, type)
colnames(h5n1.wild.cdc) = c("state", "county", "year_month",
                            "day", "type", "cases")
# Table 3: First and Last 5 Outbreaks in the United States
# till Jan. 31st 2023 (Wild Birds)
table = h5n1.wild.cdc %>%
  arrange(year_month, day)
table = headTail(table, 5, 5, ellipsis=TRUE)
table[6, c(1:5)] = "..."
colnames(table) = c("State", "County", "Year Month", "Day",
                    "Type", "Cases")
kable(table, row.names = FALSE)
##############################################################################################
########## Data 4: Average Temperature by Month in each County ##########################
##############################################################################################
# The combination and cleaning process of this dataset is in Data_Cleaning.R.          #
#                                                                                      #
# Please download from the following Github link for more detail.                      #
#                                                                                      #
# https://github.com/GitData-GA/iasc2023/blob/main/code/Data_Cleaning.R                #
##############################################################################################
# Import Data
avg.temp = read.csv("https://iasc2023.gd.edu.kg/dataset/avg_temp.csv",
                    header = TRUE)
# Table 4: First 5 Observations of Average Temperature in F
# across the United States
rownames(avg.temp) = NULL
table = head(avg.temp)
colnames(table) = c("State", "County", "Month Index", "Average Temperature")
kable(table, row.names = FALSE)
##############################################################################################
########## Data 5: h5n1.cdc.clean derived from data 1 & 2 & 3 & 4 ########################
##############################################################################################
# The combination and cleaning process of this dataset is in Data_Cleaning.R.          #
#                                                                                      #
# Please download from the following Github link for more detail.                      #
#                                                                                      #
# https://github.com/GitData-GA/iasc2023/blob/main/code/Data_Cleaning.R                #
##############################################################################################
# Import Data
h5n1.cdc = read.csv("https://iasc2023.gd.edu.kg/dataset/h5n1_cdc.csv",
                    header = TRUE)
h5n1.cdc.clean = read.csv("https://iasc2023.gd.edu.kg/dataset/h5n1_cdc_clean.csv",
                          header = TRUE)
# Table 5: Most and Least 5 Monthly Cases by County in the United
# States till January 31st 2023
```

```
h5n1.cdc1 = h5n1.cdc %>%
  arrange(desc(cases)) %>%
  filter(cases > 0) %>%
  select(fips, state, county, month.index, type, cases)
table = as.data.frame(headTail(h5n1.cdc1, 5, 5, ellipsis=TRUE))
table$state[6] = "..."
table$county[6] = "..."
table$type[6] = "..."
colnames(table) = c("FIPS code", "State", "County", "Month Index",
                    "Type", "Cases")
kable(table, row.names = FALSE)
# Table 6: Most and Least 5 Cumulative Cases by County in the United
# States till January 31st 2023
options(dplyr.summarise.inform = FALSE)
h5n1.cdc2 = h5n1.cdc.clean %>%
  group_by(fips, state, county, month.index) %>%
  summarise(cases = sum(cases))
for (i in 1:nrow(h5n1.cdc2)){
  if (h5n1.cdc2$month.index[i] != 1){
    h5n1.cdc2$cases[i] = h5n1.cdc2$cases[i] + h5n1.cdc2$cases[i - 1]
  }
}
h5n1.cdc2.final = h5n1.cdc2 %>%
  filter(month.index == 13) %>%
  filter(cases != 0) %>%
  select(fips, state, county, cases) %>%
  arrange(desc(cases))
table = as.data.frame(headTail(h5n1.cdc2.final, 5, 5, ellipsis=TRUE))
table$state[6] = "..."
table$county[6] = "..."
colnames(table) = c("FIPS code", "State", "County", "Cases")
kable(table, row.names = FALSE)
##############################################################################################
##### III. Visualization #####################################################################
##############################################################################################
plot.tltle = c("Jan. 2022", "Feb. 2022", "Mar. 2022", "Apr. 2022",
               "May. 2022", "Jun. 2022", "Jul. 2022", "Aug. 2022",
               "Sep. 2022", "Oct. 2022", "Nov. 2022", "Dec. 2022",
               "Jan. 2023")
plot.name = c("Jan_2022", "Feb_2022", "Mar_2022", "Apr_2022",
              "May_2022", "Jun_2022", "Jul_2022", "Aug_2022",
              "Sep_2022", "Oct_2022", "Nov_2022", "Dec_2022",
              "Jan_2023")
##############################################################################################
########## 1. Map New Case each Month by County ##############################################
##############################################################################################
new.case.month.county = list()
for (i in 1:13){
  data = h5n1.cdc[which(h5n1.cdc$month.index == i),]
  new.case.month.county = append(new.case.month.county, list(data))
}
map.plot = list()
for (i in 1:13){
```

```
  legend.position = "none"
  if (i == 13) {legend.position = "right"}
  plot = plot_usmap(data = new.case.month.county[[i]],
                        values = "cases",
                        color = "#e8e8e8") +
  scale_fill_gradient(low = "#ff5cf9",
                      high = "#6d02d1",
                      name = "New Cases",
                      label = scales::comma,
                      limits = c(1,5486700)) +
  labs(title = plot.tltle[i]) +
  theme(legend.position = legend.position,
        plot.title = element_text(hjust = 0.5))
  map.plot = append(map.plot, list(plot))
}

# REQUIRES: Plot has to be a valid R plot. Prefix is the first part of
#           the file name, which indicates the type of the plot. Name
#           is the latter part of the file name, use index to specify.
#           w is the width of the exported plot. h is the height of the
#           exported plot. isMap is true is the exported plot is a map.
#           ... allows you to add one argument to the plot.
# MODIFIES: Nothing
# EFFECTS:  This function will export a plot as an .svg file to your
#           device.
save.as.svg = function(plot, prefix, name, n, w, h, isMap, ...){
  for (i in 1:n){
    if (isMap){
      fileName = paste0(paste(prefix, name[i], sep = "_"), ".svg")
      svg(fileName, width = w, height = h)
      print(plot[[i]] +
              theme(legend.key.size = unit(10, 'cm'),
                    legend.text = element_text(size = 100),
                    legend.title = element_text(size = 100),
                    plot.title = element_text(hjust = 0.5, size = 200)))
      dev.off()
    }
    else {
      fileName = paste0(paste(prefix, name, sep = "_"), ".svg")
      svg(fileName, width = w, height = h)
      print(plot + ...)
      dev.off()
    }
  }
}
# save.as.svg(map.plot, "map_1", plot.name, 13, 100, 75, TRUE)
((map.plot[[1]] | map.plot[[2]] | map.plot[[3]])/
(map.plot[[4]] | map.plot[[5]] | map.plot[[6]]))
((map.plot[[7]] | map.plot[[8]] | map.plot[[9]])/
(map.plot[[10]] | map.plot[[11]] | map.plot[[12]]))
map.plot[[13]]
################################################################################################
########## 2. Map Total Case each Month by County ######################################
```

```r
##############################################################################################
h5n1.cdc.county = h5n1.cdc.clean
for (i in 2:13){
  pre.case = h5n1.cdc.county[which(h5n1.cdc.county$month.index == i - 1),]$cases
  cur.case = h5n1.cdc.county[which(h5n1.cdc.county$month.index == i),]$cases
  h5n1.cdc.county[which(h5n1.cdc.county$month.index == i),]$cases = pre.case + cur.case
}
options(dplyr.summarise.inform = FALSE)
h5n1.cdc.county.cum = h5n1.cdc.county %>%
  group_by(fips, state, county, lat, lng, month.index) %>%
  summarise(cases = sum(cases)) %>%
  filter(cases > 0)
cum.case.month.county = list()
for (i in 1:13){
  data = h5n1.cdc.county.cum[which(h5n1.cdc.county.cum$month.index == i),]
  cum.case.month.county = append(cum.case.month.county, list(data))
}
map.plot = list()
for (i in 1:13){
  legend.position = "none"
  if (i == 13) {legend.position = "right"}
  plot = plot_usmap(data = cum.case.month.county[[i]],
                    values = "cases",
                    color = "#e8e8e8") +
  scale_fill_gradient(low = "#ff5cf9",
                      high = "#6d02d1",
                      name = "Total Cases",
                      label = scales::comma,
                      limits = c(1,6188782)) +
  labs(title = plot.tltle[i]) +
  theme(legend.position = legend.position,
        plot.title = element_text(hjust = 0.5))
  map.plot = append(map.plot, list(plot))
}
# save.as.svg(map.plot, "map_2", plot.name, 13, 100, 75, TRUE)
((map.plot[[1]] | map.plot[[2]] | map.plot[[3]])/
(map.plot[[4]] | map.plot[[5]] | map.plot[[6]]))
((map.plot[[7]] | map.plot[[8]] | map.plot[[9]])/
(map.plot[[10]] | map.plot[[11]] | map.plot[[12]]))
map.plot[[13]]
h5n1.cdc.clean1 = h5n1.cdc.clean
h5n1.cdc.clean1$month.index = as.factor(h5n1.cdc.clean$month.index)
h5n1.cdc.clean1 = h5n1.cdc.clean1 %>%
  filter(cases > 0)
scatter.matrix = ggpairs(h5n1.cdc.clean1[, c(4, 5, 8, 9, 6, 7)]) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 5),
        axis.text.y = element_text(size = 5))
scatter.matrix
# Save as SVG
# scatter.matrix.exp = ggpairs(h5n1.cdc.clean1[, c(4, 5, 8, 9, 6, 7)])
# save.as.svg(scatter.matrix.exp, "scatter", "matrix", 1, 16, 16, FALSE,
#             theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size = 10),
#                   axis.text.y = element_text(size = 10)))
```

```
#################################################################################################
########## 3. Bar Chart Top 10 Counties ##########################################################
#################################################################################################
h5n1.cdc2.final = h5n1.cdc2.final %>% filter(cases > 1501550)
bar.1 = ggplot(data=h5n1.cdc2.final, aes(x=reorder(county, -cases), y = cases)) +
  geom_bar(stat = "identity") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.4, hjust=1)) +
  xlab("County") +
  ylab("Cases") +
  geom_text(aes(label = cases), vjust = -0.3, size = 3)
bar.1
# Save as SVG
# bar.1.exp = ggplot(data=h5n1.cdc2.final, aes(x=reorder(county, -cases), y = cases)) +
#   geom_bar(stat = "identity") +
#   theme(axis.text.x = element_text(angle = 90, vjust = 0.4, hjust=1)) +
#   xlab("County") +
#   ylab("Cases") +
#   geom_text(aes(label = cases), vjust = -0.3, size = 5)
# save.as.svg(bar.1.exp, "bar", "1", 1, 16, 10, FALSE,
#             theme(axis.text = element_text(size = 20),
#                   axis.title = element_text(size = 25, face="bold")))
cases_by_month <- h5n1.cdc %>% group_by(month.index) %>%
  summarise(cases = sum(cases))
bar.2 = ggplot(data=cases_by_month, aes(x= as.factor(month.index), y = cases)) +
  geom_bar(stat = "identity") +
  xlab("Month") +
  ylab("Cases") +
  geom_text(aes(label = cases), vjust = -0.3, size = 3)
bar.2
# Save as SVG
# bar.2.exp = ggplot(data=cases_by_month, aes(x= as.factor(month.index), y = cases)) +
#   geom_bar(stat = "identity") +
#   xlab("Month") +
#   ylab("Cases") +
#   geom_text(aes(label = cases), vjust = -0.3, size = 5)
# save.as.svg(bar.2.exp, "bar", "2", 1, 16, 10, FALSE,
#             theme(axis.text = element_text(size = 20),
#                   axis.title = element_text(size = 25, face="bold")))
scatter.1 = ggplot(h5n1.cdc.clean, aes(x=avg.temp, y=cases)) +
  geom_point() +
  xlab("Average Temperature (Fahrenheit)") +
  ylab("Cases")
scatter.1
# Save as SVG
# scatter.1.exp = ggplot(h5n1.cdc.clean, aes(x=avg.temp, y=cases)) +
#   geom_point() +
#   xlab("Average Temperature (F)") +
#   ylab("Cases")
# save.as.svg(scatter.1.exp, "scatter", "1", 1, 16, 10, FALSE,
#             theme(axis.text = element_text(size = 20),
#                   axis.title = element_text(size = 25, face="bold")))
#################################################################################################
##### IV. Modeling and Interpretation ###########################################################
```

```
#############################################################################################
# Split training set and testing set
train = h5n1.cdc.clean[h5n1.cdc.clean$month <= 12, ]
test  = h5n1.cdc.clean[h5n1.cdc.clean$month == 13, ]
# Logistic Regression
logistic.model = glm(as.factor(binary.case) ~ lat + lng + lat * lng +
                     month.index + avg.temp + as.factor(type), data = train,
                      family = "binomial")
summary(logistic.model)
Coefficient = c("beta.0", "beta.1", "beta.2", "beta.3", "beta.4", "beta.5",
         "beta.6", "beta.7", "beta.8")
Estimation = unname(round(logistic.model$coefficients[1:4], 5))
Estimation = append(Estimation, unname(round(logistic.model$coefficients[6:8], 5)))
Estimation = append(Estimation, unname(round(logistic.model$coefficients[5], 5)))
Estimation = append(Estimation, unname(round(logistic.model$coefficients[9], 5)))
kable(cbind(Coefficient, Estimation), row.names = FALSE)
x = model.matrix(as.factor(binary.case) ~ lat + lng + lat * lng +
                     month.index + as.factor(type) + avg.temp, train)[,-1]
y = ifelse(train$binary.case == "infected", 1, 0)
# Ridge
cv.ridge = cv.glmnet(x, y , alpha = 0, family = "binomial", nfolds = 50)
# minimum lambda
cv.ridge$lambda.min
# 1 stand error lambda
cv.ridge$lambda.1se
# Plot lambda
plot(cv.ridge)
# Coefficient using minimum lambda
coef(cv.ridge, cv.ridge$lambda.min)[,1]
# Ridge Model
ridge.model = glmnet(x, y, alpha = 0, family = "binomial",
                      lambda = cv.ridge$lambda.min)
Coefficient = c("beta.0", "beta.1", "beta.2", "beta.3", "beta.4", "beta.5",
         "beta.6", "beta.7", "beta.8")
Estimation = unname(round(coef(cv.ridge, cv.ridge$lambda.min)[,1], 5))
kable(cbind(Coefficient, Estimation), row.names = FALSE)
# lasso
cv.lasso = cv.glmnet(x, y , alpha = 1, family = "binomial", nfolds = 50)
# minimum lambda
cv.lasso$lambda.min
# 1 stand error lambda
cv.lasso$lambda.1se
# Plot lambda
plot(cv.lasso)
# Coefficient using minimum lambda
coef(cv.lasso, cv.lasso$lambda.min)
# Lasso Model
lasso.model = glmnet(x, y, alpha = 1, family = "binomial",
                      lambda = cv.lasso$lambda.min)
Coefficient = c("beta.0", "beta.1", "beta.2", "beta.3", "beta.4", "beta.5",
         "beta.6", "beta.7", "beta.8")
Estimation = unname(round(coef(cv.lasso, cv.lasso$lambda.min)[,1], 5))
kable(cbind(Coefficient, Estimation), row.names = FALSE)
```

```
##############################################################################################
##### V. Analysis ############################################################################
##############################################################################################
# Get prediction result based on test set
p = predict(logistic.model, type = "response")
logodds = log(p / (1-p))
# Confusion matrix on test set
p.test.logistic = 1 - predict(logistic.model, type = "response", test)
predicted = ifelse(p.test.logistic > 0.5, "infected", "uninfected")
confusion = table(as.factor(predicted), as.factor(test$binary.case),
                  dnn = c("True", "Predicted"))
confusion
# Ridge
x.test = model.matrix(as.factor(binary.case) ~ lat + lng + lat * lng +
                    month.index + as.factor(type) + avg.temp, test)[,-1]
ridge.prob = ridge.model %>% predict(newx = x.test)
predicted.classes.ridge = ifelse(ridge.prob > 0.5, "infected", "uninfected")
# Model accuracy
observed.classes.ridge = test$binary.case
mean(predicted.classes.ridge == observed.classes.ridge)
# Confusion matrix
confusion = table(as.factor(predicted.classes.ridge),
                  as.factor(test$binary.case),
                  dnn = c("True", "Predicted"))
confusion
p.ridge = ridge.model %>% predict(newx = x)
# Accuracy
1 - sum(diag(confusion)) / sum(confusion)
# Lasso
x.test = model.matrix(as.factor(binary.case) ~ lat + lng + lat * lng +
                    month.index + as.factor(type) + avg.temp, test)[,-1]
probabilities.lasso = lasso.model %>% predict(newx = x.test)
predicted.classes.lasso = ifelse(probabilities.lasso > 0.5, "infected", "uninfected")
# Model accuracy
observed.classes.lasso = test$binary.case
mean(predicted.classes.lasso == observed.classes.lasso)
# Confusion matrix
confusion = table(as.factor(predicted.classes.lasso),
                  as.factor(test$binary.case),
                  dnn = c("True", "Predicted"))
confusion
p.lasso = lasso.model %>% predict(newx = x)
# Accuracy
1 - sum(diag(confusion)) / sum(confusion)
# Error rate
sum(diag(confusion)) / sum(confusion)
# Accuracy
1 - sum(diag(confusion)) / sum(confusion)
kable(head(test %>%
  arrange(desc(cases)) %>%
  select(fips, state, county, type, cases) %>%
  rename(`FIPS Code` = fips, State = state, County = county,
         Type = type, Cases = cases), 10), row.names = FALSE)
```

```
g.logistic = roc(binary.case ~ p, data = train, quiet = FALSE)
# ROC curve
roc.logistic = ggroc(g.logistic) +
  geom_text(x=-0.25, y=0.5, label="Logistic\nRegression\nAUC = 0.8012")
g.logistic$auc
g.ridge = roc(binary.case ~ p.ridge, data = train, quiet = FALSE)
# ROC curve
roc.ridge = ggroc(g.ridge) +
  geom_text(x=-0.25, y=0.45, label="Ridge\nRegression\nAUC = 0.8006")
g.ridge$auc
g.lasso = roc(binary.case ~ p.lasso, data = train, quiet = FALSE)
# ROC curve
roc.lasso = ggroc(g.lasso) +
  geom_text(x=-0.25, y=0.45, label="Lasso\nRegression\nAUC = 0.8015")
g.lasso$auc
rocs = list()
rocs[["g.logistic"]] = g.logistic
rocs[["g.ridge"]] = g.ridge
rocs[["g.lasso"]] = g.lasso
roc.all = ggroc(rocs) +
  scale_color_manual(labels = c("Logistic",
                                "Ridge",
                                "Lasso"),
                     values = c("blue", "red", "green"))
roc.curves = ((roc.logistic | roc.ridge)/
  (roc.lasso | roc.all))
roc.curves
lasso.probability = exp(probabilities.lasso[,1])
test.map = cbind(test, lasso.probability)
test.map = test.map %>%
  group_by(fips, state, county) %>%
  summarise(risk = sum(lasso.probability))
lasso.map = plot_usmap(data = test.map,
           values = "risk",
           color = "#e8e8e8") +
  scale_fill_gradient(low = "blue",
                      high = "red",
                      name = "Risk\nIndex",
                      label = scales::comma,
                      limits = c(0,max(test.map$risk))) +
  theme(legend.position = "right",
        plot.title = element_text(hjust = 0.5))
lasso.map
```