ANALYSIS OF SURVEY RESULT

COVID 19 - CONSPIRACY OR NOT ??

Rahul Chaturvedi
v20rahch@du.se

`

Abstract

This document analyses 2 basic questions which are supposed to be answered as per the dataset provided. The dataset is related to the survey conducted in US population and based on various parameters to analyse the thinking of US population. It measures the influence of various news sources along with capturing the various details of person like age, gender, education and sundry.

Keywords: Covid-19, conspiracy theory, prediction, US survey result.

`

Analysis of Dataset prepared using Survey results

The survey dataset provided is a detailed survey dataset based on response of various people who took part in the survey and provides a view of overall view of the users. The dataset is based on typical feedback of users on various scales and the objective of this exercise leads to understand in details what is the general view of the public regarding covid-19 disease. In this report, we will analyse the results and try to answer the questions. The questions are 1. Does any certain type of individual believe more in conspiracy theory? You may take cons_biowpn_dummy as the class variable and 2. Build a suitable prediction model to predict an individual's degree of belief in conspiracy theory. You may make the prediction in accordance with the class coding of "cons_biowpn" variable. In the first part of the question, we will explore the relation between the dependent variable and the independent variables. However, major analysis will be focused towards estimating the people's believe if they think covid-19 is a bioweapon or not.

1. Dataset exploration and cleaning

The dataset provided consists 1009 observations in 31 columns. There were many missing values in the dataset 835 were complete entries and 174 observations have a few values missing. Since the data was mostly in the form of numerical values mostly, in various categories of survey answers, I used median values(average will be different in survey categories and add a new complexity)to fill the dataset in most missing observations. I removed the survey weight column as well, as it was advised in the instructions and pending missing values as well. Finally I was left with 1004 observations without any missing values. There isn't any need for scaling of dataset or removing outliers as most of the survey data is within ranges. I changed variables age, hispanic,gender,white,pid3,pid2,and idlg to factors because these observations are not in any order, they are just different. However, I found that other variables have an order in different values hence it made sense to leave them as numerical values.

On dataset variables analysis, it was found that dummy variables are encoded as per different levels, hence they should be very correlated with other variables. However, analysis based on correlation of variables isn't practical based on dataset with big number of variables. The dependent variable for task 1 is a binary dummy variable while most of the dataset is in different levels of numeric values only. Same is the case with task 2 dependent variable but with 4 levels. Also, both questions are classification questions.

I will provide the major results in the report, additional results in Appendix and entire script available for verification as add-on with this report.

`

2. TASK 1 -

Method

This is a binary dependent variable (cons_biowpn_dummy) for which logistic regression is best suited for this inference task. Logistic regression is better choice for this kind of this binary classification problem then LDA, QDA and KNN because of the relevance. Hence I decided to proceed with Logistic regression. Classifier variable cons_biowpn_dummy is also balanced here with almost half of variables in a class. I changed variable cons_biowpn_dummy to factors here.

There are many variables in the dataset. With removal of cons_biowpn (which has linear relation with the dependant variable),AIC was **1022.9** in logistic regression model. This result is present in **Appendix as Table 5.1.**

I tried to look at correlation between the variables and finally select preferred variables as well, but this approach will not work here because of many variables.

To reduce the variables , I used best subset selection method on the selected model. I used BIC to figure out the relevant number of variables for my logistic regression model. The result came with **5 variables** to be selected. The plot is shown in **Appendix as Figure 5.1**

Making the model with only these variables led to AIC reduction to the value **990.74.** Since we know that AIC is better when value is lower, I can deduce that I selected relevant model with less variables. The snapshot of the result is present in the Appendix as Figure 5.2

Result and Analysis –

Coefficients of Logistic regression model

	Estimate	Std. Error	z value	Pr(>|z|)	
(Intercept)	-5.79306	0.45207	-12.815	< 2e-16	***
populism_5	0.35104	0.08673	4.048	5.17E-05	***
populism_1	0.4639	0.10532	4.405	1.06E-05	***
cons_covax	1.05821	0.08942	11.834	< 2e-16	***
pid22	0.77425	0.16891	4.584	4.57E-06	***
md_fox	0.36263	0.06963	5.208	1.91E-07	***

Table 2.1 Coefficients of logisitic regression model with reduced variables

By looking at above result, we can interpret that selected variables in the table affect the probability they impact the reasoning of a certain person believing more or less in the conspiracy theory. The p value indicate the statistical significance of the variable influence while estimate value informs us how much probability of the variable provide the influence on the thinking of a certain individual.

`

However, we should not forget that we are not looking at linear regression which has direct relation with the Estimate value in the model. In logistic regression, the values present in the estimate are the **log-odds** and one unit increase in one variable is associated with an increase in the log-odds of dependant variable by those number of units and vice versa[2]pg 134.

We can also look at the direct relation by using the exponential values of the coefficients, because of the definition of logistic regression In contrast, in a logistic regression model, increasing X by one unit changes the log odds by β1, or equivalently it multiplies the odds by e^β1.[2]Pg 132.

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X.$$

But since this relation is not a straight line, β1 do not corresponds to change in probability of p(X) associated with one unit increase in Estimate.[2]pg 133.Taking the exponential values help makes below table which changes log-odds to odds for simplicity.

Coefficients table of variables

	Estimate	exp(Estimate)
(Intercept)	-5.79306	0.00304864
populism_5	0.35104	1.42054415
populism_1	0.4639	1.59026394
cons_covax	1.05821	2.88120901
pid22	0.77425	2.16896479
md_fox	0.36263	1.43710403

Table 2.2 Coefficients of logistic regression model with exponential values

But since this relation is not a straight line, β1 do not corresponds to change in probability of p(X) associated with one unit increase in Estimate.[2]pg 133. So with other variables as constant values, I can say increase in populism 5 increases the probability of dependant variable by 1.42 times. And all the selected variables are statistically significant. If there is a negative value in the independent variable, which is not in this case, it will lead to conclusion that it has adverse effect on the probability of dependant variable.

3. TASK 2- Best model Selection using cons_biowpn as class variable

Method

This is a classification problem with 4 classes for given dependant variable - cons_biowpn. These variables are also balanced.

In this question, I will try to make the model with QDA and KNN while trying to analyse the results. Multinomial logistic regression may also be tried but I will not analyse the same in the current scope of analysis. I can see many variables as categorical, so LDA is not valid option because the data is not normally distributed, however I will change the selected variables to numerical for the sake of analysis. For LDA, I will change the selected variables to numerical

`

values for them to hold normality. I will use k-fold cross-validation method in resampling of the training data. I made training data set and test set in 80-20 ratio and I will use them in below methods. This gave me 803 observations in training set and 201 observations in test set.

1. Using Quadratic Discriminant Analysis QDA with k-fold Crossvalidation

First I will select the variables with new dependant variable. I will be using best subset selection again with cons_biowpn variable with my dataset and select the relevant variables to use in the prediction model. Using lowest BIC – 6, I selected 6 variables for the analysis.

Lowest BIC for model selection

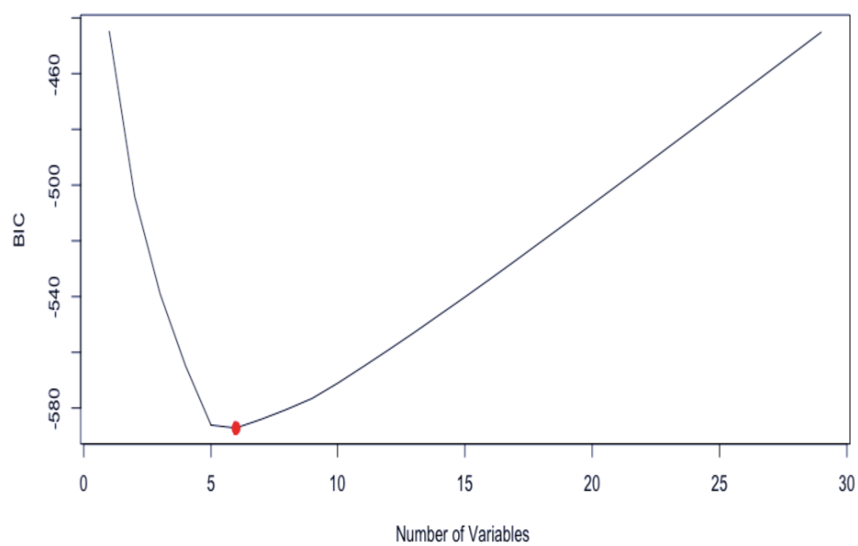

Figure 3.1 BIC plot for variable selection

Coefficients of best subset

Intercept	populism_5	populism_1	cons_covax	pid33	md_localpap	md_fox
0.07195566	0.15536811	0.20287367	0.47261668	0.35904527	-0.0817208	0.17289736

Table 3.1 Selected variables from best subset selection

Now I will apply QDA Model with **10-fold** cross validation on these variables. The snapshot of result is in Appendix Fig 5.3

Results -

Average accuracy of the prediction
> mean(qda.pred==test_set[,12])
[1] 0.5074627

`

Accuracy of individual prediction on Test set – Proportion Table

	Actual test set values				
		1	2	3	4
QDA Predicted values	1	0.76363636	0.33333333	0.2	0.05
	2	0.09090909	0.33333333	0.30909091	0.025
	3	0.07272727	0.21568627	0.32727273	0.3
	4	0.07272727	0.11764706	0.16363636	0.625

From QDA model, we can see that average accuracy is 50.7%, while the model is able to predict correct prediction for value of 1(least agreement with covid-19 as bioweapon) as 76%, 2-33.3%,3-32.7% and 4(most agreement with theory) – 62.5% as correct prediction.

2. **Using K- Nearest Neighbor KNN with k fold-cross validation**

   We can check other prediction models with the same set of dependent and independent variables with different models. So I will proceed with same 6 variables to apply the KNN model on the same training and test set. The snapshot of model application are shown in Appendix Fig 5.4

Result -

> mean(knnPredict==test_set[,12])
[1] 0.5223881

Accuracy of individual prediction on Test set – Proportion Table

	Actual test set values				
		1	2	3	4
KNN Predicted Values	1	0.67272727	0.15686275	0.10909091	0.05
	2	0.12727273	0.47058824	0.38181818	0.15
	3	0.16363636	0.29411765	0.41818182	0.275
	4	0.03636364	0.07843137	0.09090909	0.525

From KNN model, we can see that average accuracy is 52.2%, while the model is able to predict correct prediction for value of 1(least agreement with covid-19 as bioweapon) as 67%, 2- 47%,3-41.8% and 4(most agreement with theory) – 52.5% as correct prediction.

3. Using LDA

Result –

> mean(lda.pred==test_set[,12])
[1] 0.5074627

`

Accuracy of individual prediction on Test set – Proportion Table

	Actual test set values				
	1	2	3	4	
	1	0.78181818	0.43137255	0.18181818	0.05
knnPredict	2	0.12727273	0.19607843	0.16363636	0.05
	3	0.05454545	0.35294118	0.45454545	0.3
	4	0.03636364	0.01960784	0.2	0.6

From LDA Model, we can see that average accuracy is 50.7%, while the model is able to predict correct prediction for value of 1(least agreement with covid-19 as bioweapon) as 78%, 2- 19%,3- 45.5% and 4(most agreement with theory) – 60% as correct prediction.

4. Conclusion

As requested in Task 1, we were able to analyse the question and conclude as per our analysis the model which may predict the probability of a certain group of person influenced by the conspiracy theory more than other keeping other variables at constant and we analysed the statistical inferences in detailed discussion. However, in Task 2 , we analysed the accuracy of KNN, LDA and QDA models with 10 fold cross validation on test set, the accuracy was little better than 50% only, which is not better than making any guess, on best subset selected models. KNN performed better than other models at k=5 selected by cross validation.

`

References

1. https://stackoverflow.com/questions/40080187/kfold-cross-validation-for-knn-text-classifier-in-r

2. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning: With applications in R*. Springer.

5. Appendix

1. The BIC value plot with number of variables.

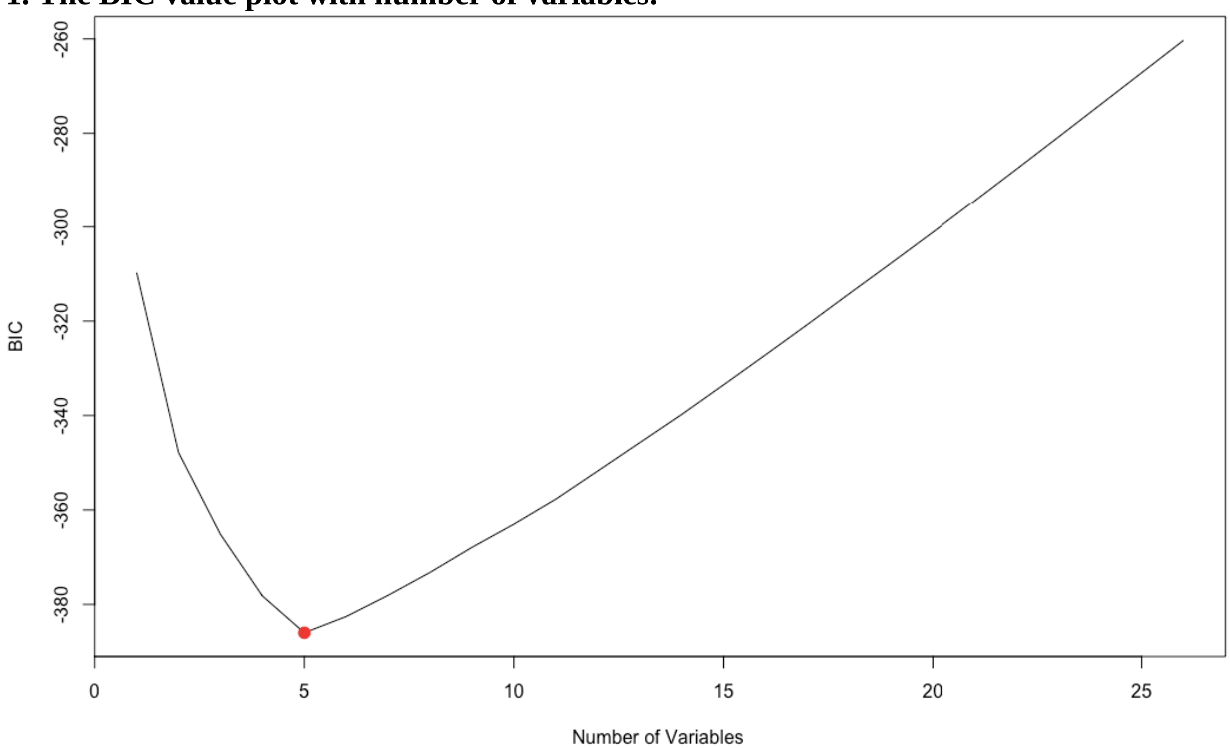

Figure 5.1 The minimum BIC value with number of variables

2. Logistic regression model on all variables

Coefficients of model before Variable reduction methods

	Estimate	Std. Error z	value	Pr(>|z|)	
(Intercept)	-5.93E+00	7.63E-01	-7.763	8.30E-15	***
trust_1	7.83E-02	1.15E-01	0.679	0.497294	

`

populism_5	3.65E-01	9.86E-02	3.704	0.000212	***
populism_4	-8.56E-02	1.12E-01	-0.765	0.444434	
populism_3	1.09E-01	9.96E-02	1.093	0.274178	
populism_2	-3.92E-02	1.18E-01	-0.334	0.738578	
populism_1	4.47E-01	1.23E-01	3.646	0.000266	***
age	-9.29E-04	5.85E-03	-0.159	0.873939	
gender2	-2.31E-01	1.68E-01	-1.378	0.168124	
hhi	-7.97E-03	1.39E-02	-0.575	0.56496	
hispanic1	8.44E-02	2.62E-01	0.322	0.747637	
cov_beh_sum	3.67E-03	1.61E-02	0.229	0.81912	
cons_covax	1.09E+00	9.81E-02	11.06	< 2e-16	***
white1	-5.96E-02	2.08E-01	-0.287	0.774449	
highered	-1.14E-01	1.79E-01	-0.638	0.523646	
idlg2	1.87E-01	3.81E-01	0.491	0.623717	
idlg3	2.68E-01	4.03E-01	0.665	0.506282	
idlg4	-7.61E-02	3.36E-01	-0.227	0.820536	
idlg5	-2.00E-02	4.11E-01	-0.049	0.961149	
idlg6	1.13E-01	4.02E-01	0.282	0.778038	
idlg7	3.70E-01	4.31E-01	0.859	0.390469	
pid32	6.55E-02	2.45E-01	0.267	0.789422	
pid33	4.75E-01	3.74E-01	1.269	0.204526	
pid22	4.19E-01	3.16E-01	1.327	0.184359	
md_radio	1.33E+05	4.01E+05	0.331	0.740447	
md_national	1.33E+05	4.01E+05	0.331	0.740447	
md_broadcast	1.33E+05	4.01E+05	0.331	0.740446	
md_localpap	1.33E+05	4.01E+05	0.331	0.740447	
md_localtv	1.33E+05	4.01E+05	0.331	0.740446	
md_fox	3.00E-01	8.89E-02	3.379	0.000728	***
md_agg	1.33E+05	4.01E+05	0.331	0.740446	
md_con	1.49E-01	1.03E-01	1.443	0.149034	
ms_news	-7.97E+05	2.41E+06	-0.331	0.740447	

Table 5.1 – Logistic regression model with all variables of dataset

3. Snapshot of final Logistic regression model for inference

```
Call:
glm(formula = cons_biowpn_dummy ~ populism_5 + populism_1 + cons_cov
ax +
    pid2 + md_fox, family = binomial, data = df1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5267  -0.7557  -0.3553   0.7939   2.5051

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.79306    0.45207 -12.815  < 2e-16 ***
populism_5   0.35104    0.08673   4.048 5.17e-05 ***
populism_1   0.46390    0.10532   4.405 1.06e-05 ***
cons_covax   1.05821    0.08942  11.834  < 2e-16 ***
pid22        0.77425    0.16891   4.584 4.57e-06 ***
md_fox       0.36263    0.06963   5.208 1.91e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1388.71  on 1003  degrees of freedom
Residual deviance:  978.74  on  998  degrees of freedom
AIC: 990.74

Number of Fisher Scoring iterations: 5
```

Figure 5.2 – Snapshot of selected Logisitic regression model with cons_biowpn_dummy as dependant variable.

4. QDA model snapshot

```
> qda.fit.cv
Quadratic Discriminant Analysis

803 samples
  6 predictor
  4 classes: '1', '2', '3', '4'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 722, 723, 723, 722, 723, 723, ...
Resampling results:

  Accuracy   Kappa
  0.5006327  0.3315185

> qda.pred=predict(qda.fit.cv,test_set)
> table(qda.pred,test_set[,12])

qda.pred  1  2  3  4
       1 42 17 11  2
       2  5 17 17  1
       3  4 11 18 12
       4  4  6  9 25
> #mean(qda.pred!=test_set[,12])
> mean(qda.pred==test_set[,12])
[1] 0.5074627
> cm = prop.table(table(qda.pred,test_set[,12]),2)
> cm

qda.pred          1          2          3          4
       1 0.76363636 0.33333333 0.20000000 0.05000000
       2 0.09090909 0.33333333 0.30909091 0.02500000
       3 0.07272727 0.21568627 0.32727273 0.30000000
       4 0.07272727 0.11764706 0.16363636 0.62500000
```

Figure 5.3 – Snapshot of QDA model and testing on test data set to calculate accuracy.

5. KNN model application

`

```
> knnFit
k-Nearest Neighbors

803 samples
  6 predictor
  4 classes: '1', '2', '3', '4'

Pre-processing: centered (7), scaled (7)
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 723, 722, 722, 724, 724, 722, ...
Resampling results across tuning parameters:

  k  Accuracy   Kappa
  5  0.5031485  0.3317572
  7  0.5018827  0.3301549
  9  0.4906157  0.3141506

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was k = 5.
> knnPredict <- predict(knnFit,newdata = test_set )
> confusionMatrix(knnPredict, test_set[,12] )
Confusion Matrix and Statistics

          Reference
Prediction  1  2  3  4
         1 37  8  6  2
         2  7 24 21  6
         3  9 15 23 11
         4  2  4  5 21

Overall Statistics

               Accuracy : 0.5224
                 95% CI : (0.451, 0.5931)
    No Information Rate : 0.2736
```

`

```
               Accuracy : 0.5224
                 95% CI : (0.451, 0.5931)
    No Information Rate : 0.2736
    P-Value [Acc > NIR] : 8.362e-14

                  Kappa : 0.358

 Mcnemar's Test P-Value : 0.6339

Statistics by Class:

                     Class: 1 Class: 2 Class: 3 Class: 4
Sensitivity            0.6727   0.4706   0.4182   0.5250
Specificity            0.8904   0.7733   0.7603   0.9317
Pos Pred Value         0.6981   0.4138   0.3966   0.6562
Neg Pred Value         0.8784   0.8112   0.7762   0.8876
Prevalence             0.2736   0.2537   0.2736   0.1990
Detection Rate         0.1841   0.1194   0.1144   0.1045
Detection Prevalence   0.2637   0.2886   0.2886   0.1592
Balanced Accuracy      0.7816   0.6220   0.5892   0.7283
> table(knnPredict, test_set[,12] )

knnPredict  1  2  3  4
         1 37  8  6  2
         2  7 24 21  6
         3  9 15 23 11
         4  2  4  5 21
> prop.table(table(knnPredict,test_set[,12]),2)

knnPredict          1          2          3          4
         1 0.67272727 0.15686275 0.10909091 0.05000000
         2 0.12727273 0.47058824 0.38181818 0.15000000
         3 0.16363636 0.29411765 0.41818182 0.27500000
         4 0.03636364 0.07843137 0.09090909 0.52500000
> mean(knnPredict==test_set[,12])
[1] 0.5223881
>
```

Figure 5.4 – Snapshot of KNN model and testing on test data set to calculate accuracy.

`

6. LDA model application to the test set

```
> lda.fit.cv
Linear Discriminant Analysis

803 samples
  5 predictor
  4 classes: '1', '2', '3', '4'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 722, 723, 722, 723, 723, 725, ...
Resampling results:

  Accuracy  Kappa
  0.5154    0.348087

> lda.pred=predict(lda.fit.cv,test_set)
> table(lda.pred,test_set[,12])

lda.pred  1  2  3  4
       1 43 22 10  2
       2  7 10  9  2
       3  3 18 25 12
       4  2  1 11 24
> #mean(lda.pred!=test_set[,12])
> mean(lda.pred==test_set[,12])
[1] 0.5074627
>
```

Figure 5.5 – Snapshot of LDA model and testing on test data set to calculate accuracy.