ECO481_Assignment1

Alexander Tran - 1006314089

2/12/2023

Exercise 1

1.

- Albert's data has 10,000 features and only 100 observations, so his model is most likely to suffer from the 'Curse of Dimensionality'. This means that the data will be very sparse, which could affect predictive accuracy. There is also a high risk of overfitting, which would give a good in-sample performance, but an abysmal out-of-sample performance. There is also the problem of distance concentration, where the pairwise distances between points converge to the same value.
- To have 50% accuracy means that the proportion of correct predictions is equal to 0.50. That is, the sum of the number of true positives and true negatives, divided by the total sample population, is equal to 0.50. In this context, an accuracy of 50% on the test sample, given that the model had a 99% accuracy on the training sample, suggests that the model is heavily overfitted, as the difference in performance is significantly large when comparing across in-sample and out-of-sample data.
- One step to take in order to fix the overfitting that Albert is having is to reduce the number of features in the model and use a less flexible and complex model. This will bring the number of features closer in line to the number of observations, and should reduce overfitting. An alternative method is to use regularization, which tends to reduce model variance at the cost of an increase in the bias.
- Based on these results, I would recommend that Albert use logistic regression. Given that the average error rate is 9% for the 1-nearest neighbours method, and that $K = 1$ for KNN results in an overly flexible model, the model is overfitted and the error rate is actually 0% for the training data and 18% for the test data. Thus, the error rate is higher for KNN, so we opt for logistic regression.

2. Panel A is most likely to have been generated by KNN, as we can see that the decision border is very flexible and makes (almost) no error in distinguishing between the orange and blue points, indicative of KNN with a small K i.e. $K = 1$. Panel B is most likely to have been generated by logistic regression, as the decision border is linear, and does not classify the points perfectly, which is not unexpected for logistic regression.

3. In a logistic regression classifier, all points are taken into account to compute average marginal effects, which are used to determine a decision boundary. Thus, there is some weight given to this point and Annie is right in saying that removing it will affect the decision boundary, even though the effect may not be significant.

4. $\beta = \infty$ is correct because the data is perfectly separable. The only way for the model to emulate this behavior is if it fails to converge with $\beta = \infty$.

Exercise 2

1.

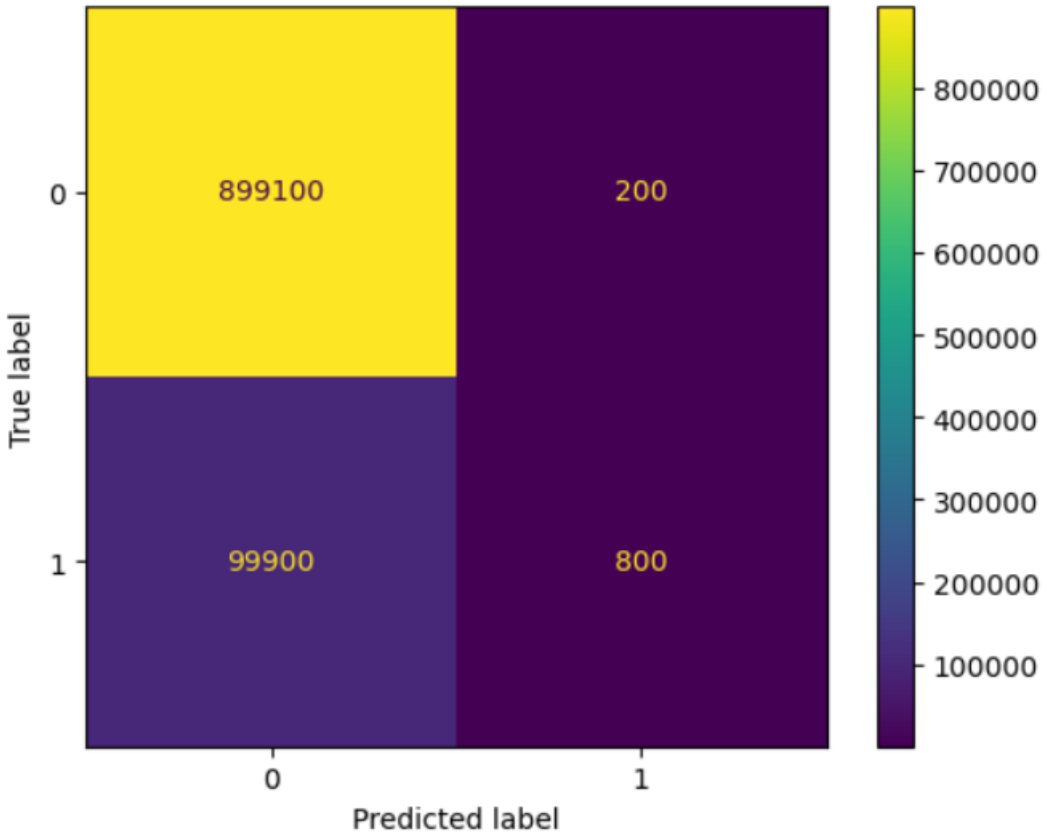

Figure 1: Confusion matrix for 1,000,000 tweets, where a 1 denotes 'fake' and 0 denotes 'non-fake'

Figure 1 was created using Python and can be found in the submitted code (.ipynb).

2. The current algorithm is not performing at an acceptable level. The algorithm's accuracy is currently $0.8999 = \frac{800+899100}{1000+999000} = \frac{TP+TN}{P+N}$, implying an error rate of $1 - 0.8999 = 0.1001$, the precision is equal to $0.8 = \frac{800}{800+200} = \frac{TP}{TP+FP}$, and the false positive rate is equal to $0.0002 = \frac{200}{899100+200} = \frac{FP}{TN+FP}$.
Although these measures all are roughly acceptable, the true positive rate, equal to $0.0079 = \frac{800}{800+99900} = \frac{TP}{TP+FN}$, indicates significant flaws in the algorithm as it is failing to correctly identify 99,900 out of 100,700 cases of fake tweets. One way to improve the model may be to reduce the probability threshold required for a tweet to be classified as 'fake'. While this will increase the TPR, it also has the side effect of also increasing the FPR, which is a good tradeoff given the context of the algorithm.

Exercise 3

1. We begin by computing the entropy for each of the variables, given by $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log(p_{mk})$.

Entropy of *Large household size*:

- If True: $D = -[\frac{1}{3}\log_2(\frac{1}{3}) + \frac{2}{3}\log_2(\frac{2}{3})] = 0.9182958$
- If False: $D = -[\frac{2}{3}\log_2(\frac{2}{3}) + \frac{1}{3}\log_2(\frac{1}{3})] = 0.9182958$
- Overall entropy: $\frac{3}{6}(0.9182958) + \frac{3}{6}(0.9182958) = 0.9182958$

Entropy of *Have a cell phone*:

- If True: $D = -[\frac{2}{2}\log_2(\frac{2}{2}) + \frac{0}{2}\log_2(\frac{0}{2})] = 0$
- If False: $D = -[\frac{1}{4}\log_2(\frac{1}{4}) + \frac{3}{4}\log_2(\frac{3}{4})] = 0.8112781$
- Overall entropy: $\frac{2}{6}(0) + \frac{4}{6}(0.8112781) = 0.5408521$

Entropy of *Have a mattress at home*:

- If True: $D = -[\frac{1}{1}\log_2(\frac{1}{1}) + \frac{0}{1}\log_2(\frac{0}{1})] = 0$
- If False: $D = -[\frac{2}{5}\log_2(\frac{2}{5}) + \frac{3}{5}\log_2(\frac{3}{5})] = 0.9709506$
- Overall entropy: $\frac{1}{6}(0) + \frac{5}{6}(0.9709506) = 0.8091255$

The overall entropy of the *Have a cell phone* node is the lowest and gives the highest information gain at this split, so we choose it as the node for the initial split.

At this point, *M-pesa user* is *True* for all observations in which *Have a cell phone* is *True*. Thus, this node is pure and terminal. We then compute the entropy for each of the remaining variables to further split and attain higher purity in the other node.

Entropy of *Large household size*:

- If True: $D = -[\frac{1}{3}\log_2(\frac{1}{3}) + \frac{2}{3}\log_2(\frac{2}{3})] = 0.9182958$
- If False: $D = -[\frac{1}{1}\log_2(\frac{1}{1}) + \frac{0}{1}\log_2(\frac{0}{1})] = 0$
- Overall entropy: $\frac{3}{4}(0.9182958) + \frac{1}{4}(0) = 0.6887218$

Entropy of *Have a mattress at home*:

- If True: $D = -[\frac{1}{1}\log_2(\frac{1}{1}) + \frac{0}{1}\log_2(\frac{0}{1})] = 0$
- If False: $D = -[\frac{3}{3}\log_2(\frac{3}{3}) + \frac{0}{3}\log_2(\frac{0}{3})] = 0$
- Overall entropy: $\frac{1}{4}(0) + \frac{3}{4}(0) = 0$

The overall entropy of the *Have a mattress at home* node is the lowest and gives the highest information gain at this split, so we choose it as the node for this split.

At this point, *M-pesa user* is *True* for all observations in which *Have a mattress at home* is *True*, and *False* otherwise. Thus, these nodes are pure and terminal, and there no need for further splitting.

Shown on the following page is the resulting decision tree:

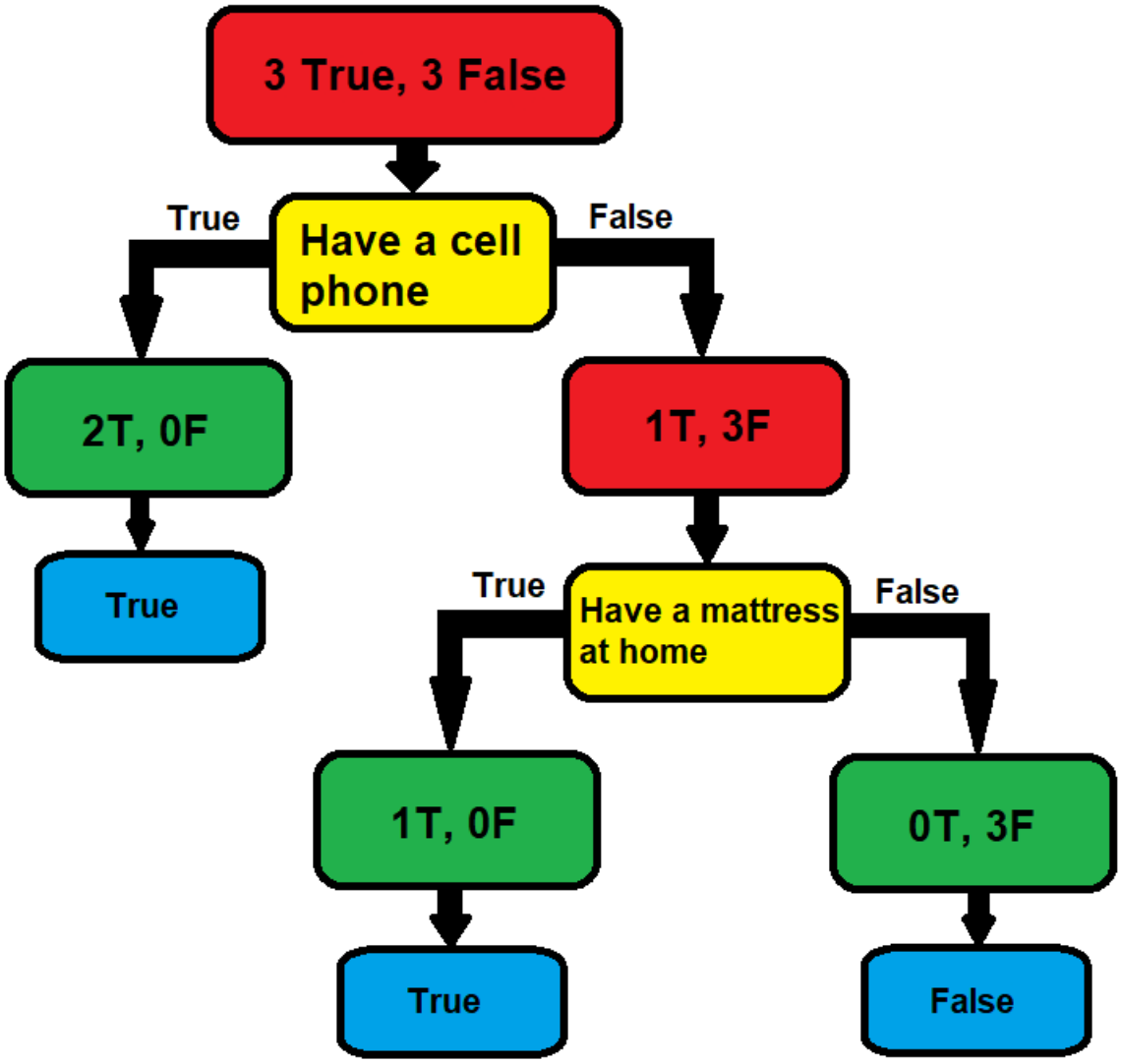

Figure 2: Decision Tree for Classifying 'M-Pesa User'

The prediction generated by the tree for: *Large Household = False*, *Have a cell phone = False* and *Have a mattress at home = True* is *M-pesa user = True*.

The prediction generated by the tree for: *Large Household = True*, *Have a cell phone = True* and *Have a mattress at home = True* is *M-pesa user = True*.

2. Please refer to the submitted code.

3.

	mpesa_user
count	2261.000000
mean	0.737284
std	0.440207
min	0.000000
25%	0.000000
50%	1.000000
75%	1.000000
max	1.000000

Figure 3: Table of Descriptive Statistics for 'M-Pesa User'

As can be seen from Figure 3, there are 2261 complete observations after cleaning. 73.73% of these observations are M-PESA users, consisting of 1667 observations.

Figure 3 was created using Python and can be found in the submitted code (.ipynb).

4.

		cellphone	totexppc	wkexppc	wealth	size	education_years	pos	neg	ag	sick
mpesa_user											
0	count	594.000000	5.940000e+02	594.000000	5.940000e+02	594.000000	594.000000	594.000000	594.000000	594.000000	594.000000
	mean	0.427609	5.423797e+04	28578.544063	7.647869e+04	4.126263	6.489899	0.048822	0.540404	0.139731	0.378788
	std	0.495149	9.342726e+04	28099.286689	2.970897e+05	2.449328	4.624234	0.215676	0.498785	0.347000	0.485494
	min	0.000000	4.800000e+02	0.000000	0.000000e+00	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	25%	0.000000	1.855420e+04	13000.000000	7.062500e+03	2.000000	2.000000	0.000000	0.000000	0.000000	0.000000
	50%	0.000000	3.113607e+04	20297.335000	2.025000e+04	4.000000	7.000000	0.000000	1.000000	0.000000	0.000000
	75%	1.000000	5.375650e+04	32844.500000	5.022500e+04	6.000000	10.000000	0.000000	1.000000	0.000000	1.000000
	max	1.000000	1.576484e+06	237484.000000	4.753200e+06	12.000000	19.000000	1.000000	1.000000	1.000000	1.000000
1	count	1667.000000	1.667000e+03	1667.000000	1.667000e+03	1667.000000	1667.000000	1667.000000	1667.000000	1667.000000	1667.000000
	mean	0.922615	8.447676e+04	35493.869666	2.149231e+05	4.262747	8.373725	0.072585	0.529694	0.108578	0.389922
	std	0.267281	1.029500e+05	27961.784440	1.460980e+06	2.176337	5.287427	0.259533	0.499267	0.311203	0.487879
	min	0.000000	2.306000e+03	1397.500000	0.000000e+00	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	25%	1.000000	3.441190e+04	18412.625000	2.495000e+04	3.000000	5.000000	0.000000	0.000000	0.000000	0.000000
	50%	1.000000	5.720400e+04	27726.400000	5.400000e+04	4.000000	9.000000	0.000000	1.000000	0.000000	0.000000
	75%	1.000000	9.857000e+04	42734.095000	1.126000e+05	6.000000	12.000000	0.000000	1.000000	0.000000	1.000000
	max	1.000000	1.870776e+06	263380.000000	4.720000e+07	13.000000	19.000000	1.000000	1.000000	1.000000	1.000000

Figure 4: Glimpse of a Table of Descriptive Statistics for Various Variables, Conditional on 'M-Pesa User'

Figure 4 was created using Python and the full table can be found in the submitted code (.ipynb).

5. The means of several variables associated with a higher quality of living are considerably higher for users of M-PESA. This includes variables such as *Own Cell Phone*, *Per Capita Consumption*, *Education of Head*, etc. There is also a correlation between negative variables and non M-PESA users.

6. Please refer to the submitted code.

7. Looking at the Python output in the submitted code, the logistic classifier has an accuracy of 0.8234, and area under the curve of 0.8506, both of which are the highest among the three classifiers. By these metrics, The logistic classifier is clearly the best one of the three.

8. Based on the Python output, the top 3 predictors are the 'Education of Head (Years)', 'Main Occ:Public', and 'Receive Remittances', in that order.

9. Based on the Python output, the optimal value of neighbours K is 9, as the KNN classifier with $K = 9$ has the highest cross validated mean accuracy of 0.827069.

10. Using the same two performance metrics, accuracy and area under the curve, the optimal KNN classifier does not outperform the logistic classifier constructed previously. Both the accuracy and area under the curve are lesser than those of the logistic classifier.

11. Based on these findings, the government should mainly focus on increasing education, consumption per capita, and cell phone ownership rates in the population. While increasing education and consumption are complex tasks, the government may be able to subsidize cell phones or provide vouchers to increase ownership.