Solution_Q3

Shristi, Dhwanit, Bao, Akssay

3/12/2020

Classifying mashable articles as viral and not viral based on given features.

Apart from the trivial null model of predicting every article as non viral, two models are built:

1. Model 1: regress to fit log(shares) and then threshold it to classify as viral or not viral and,

2. Model 2: directly do a logistic regression to predict whether it is viral or not.

For both models, lasso regression with cross validation is used to minimize deviance with penalty term while avoiding overfitting and enabling automatic variable selection.

Summary of the results

Confusion matrix of NULL model

	Actual viral	Actual non-viral
Predicted viral	0	0
Predicted non-viral	19563	20082

Confusion matrix of model 1

	Actual viral	Actual non viral
Preicted Viral	17551	15261
Predicted non-viral	2011	4821

Confusion matrix of model 2

	Actual Viral	Actual non-viral
Predicted viral	12374	7492
Predicted non-viral	7188	12590

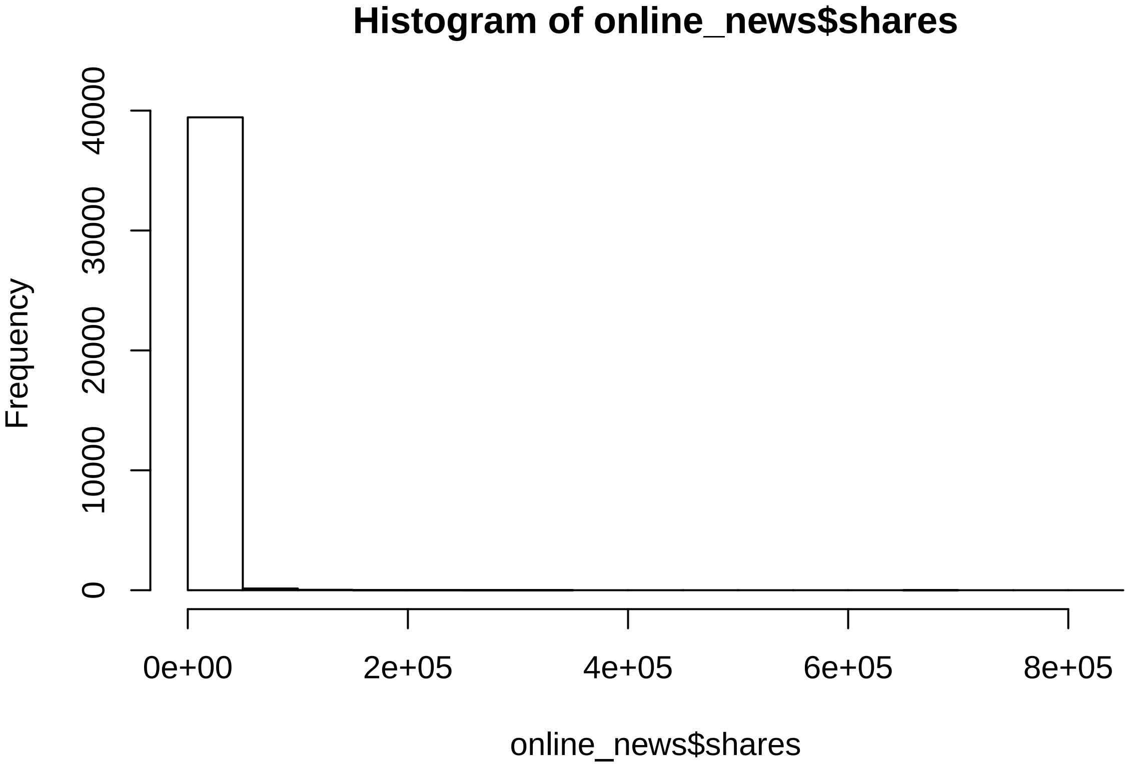
Table of summary of models

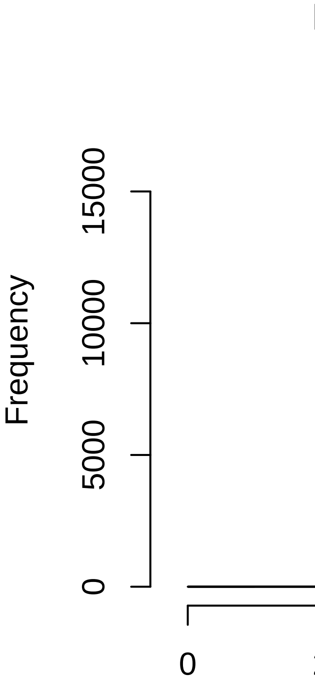
Model	Overall Accuracy	TPR	FPR	FDR
Null	50.66	0	0	undefined
Model 1	56.8	89.7	76	46.5

Model	Overall Accuracy	TPR	FPR	FDR
Model 2	63	63.2	37.3	37.7

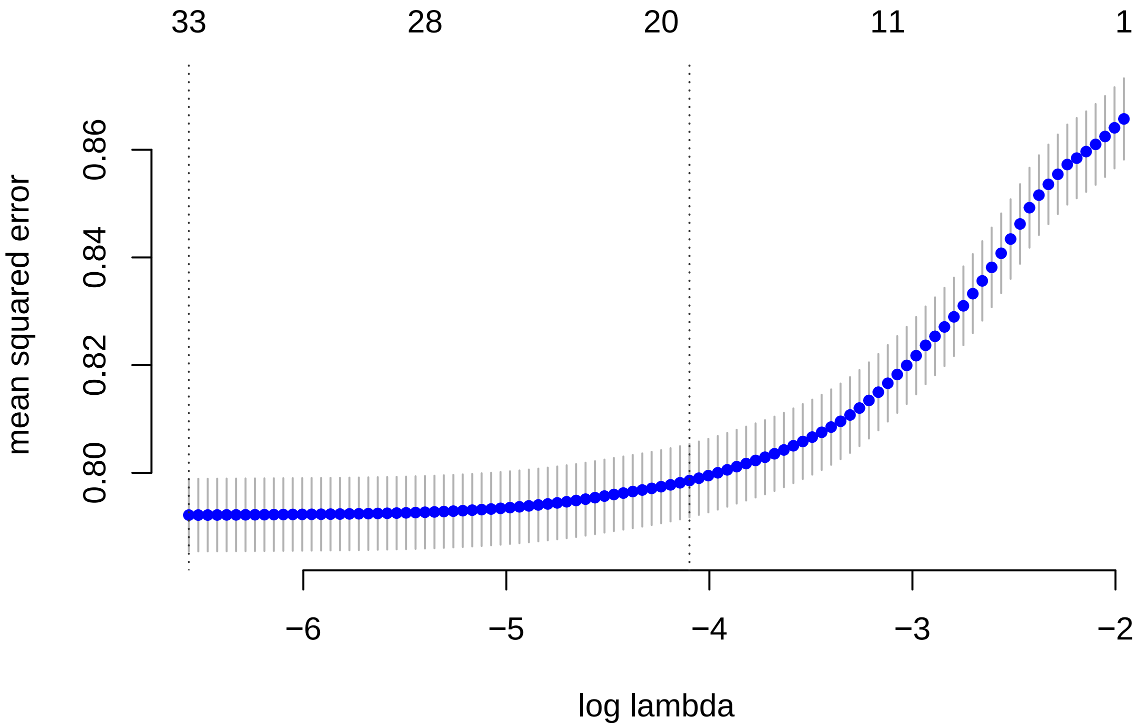
Conclusion: From the table, the overall accuracy rates indicate that **model 2 (direct logistic regression with lasso)** performs better than **model 1 (regress than threshold)** while both perform better than the null model. Model 2 is 6% better than model 1 and 13% better than null model. Althought the TPR for the model 1 is much higher than the model 2, FPR and FDR are much lower for model 2. Thus, overall model 2 performs better than model 1.

Reason: The reason why logistic regression works better for classification is that it handles the imbalance in the data well and is not too sensitive to adding or removing input data. Due to flat tails of the logit link and sharp increase in the middle, it handles the classification problem better than just fitting a linear model and thresholding.





```
## fold 1,2,3,4,5,6,7,8,9,10,done.
```



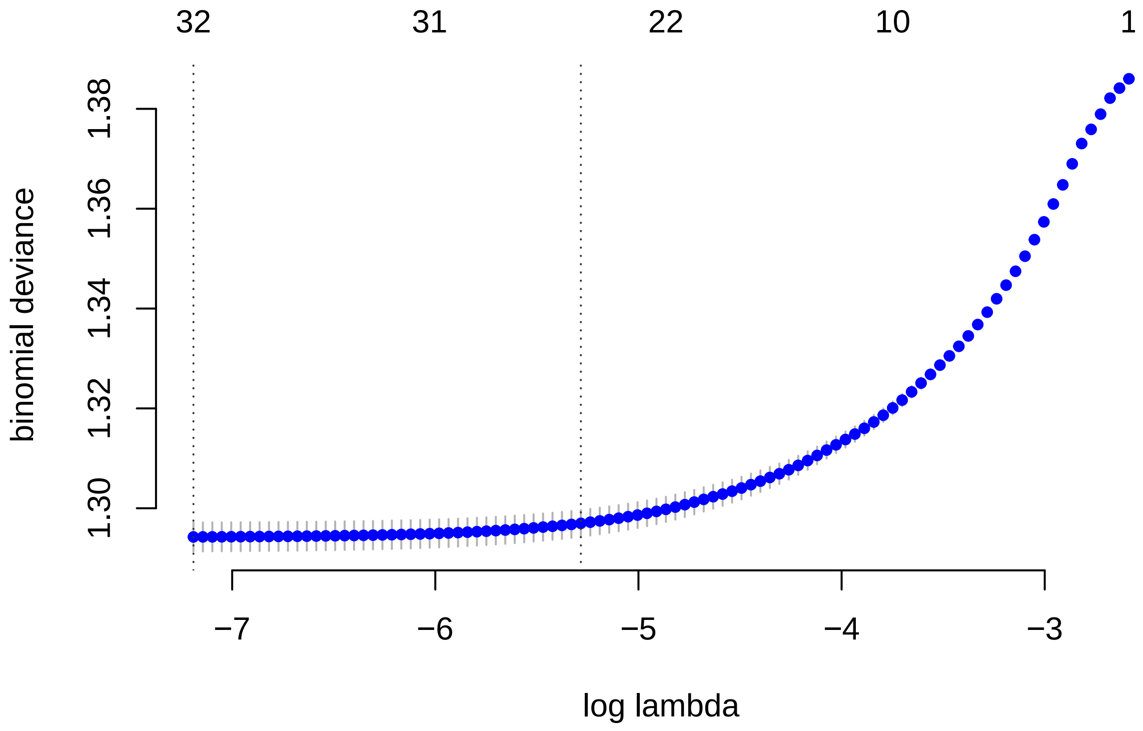
```
## [1] -6.563407
```

```
## [1] 33
```

```
##    yhat
## y       0     1
##   0  5165 14917
##   1  2138 17424
```

```
## [1] 0.5697962
```

```
## fold 1,2,3,4,5,6,7,8,9,10,done.
```



```
## [1] -7.190693
```

```
## [1] 32
```

```
##    yhat
## y       0     1
##   0 12249  7833
##   1  6832 12730
```

```
## [1] 0.6300827
```

```
## viral
##     0     1
## 20082 19562
```

```
## [1] 0.5065584
```

```
##    yhat
## y       0     1
##   0  5165 14917
##   1  2138 17424
```

```
## [1] 0.5697962
```

```
##    yhat
## y       0     1
##   0 12249  7833
##   1  6832 12730
```

```
## [1] 0.6300827
```