Week 6 Assignment (80 Points)

Do Classification Tree posted with this assignment (20 Points)

Week 6 Assignment: Naïve Bayes Algorithm (80 Points)

1. **Discuss the Exact Bayes and Naïve Bayes algorithm. Explain which one is used in data mining technologies such as R and why the other one not. (10 Points)**

**Bayes algorithm (Exact):**

* In this algorithm, classification for a new record is based on finding other records in the dataset that have the same predictor profile as the new record. This means considering only records that match the new record's predictor values.
* For these matching records, the algorithm determines the classes they belong to and identifies which class is most prevalent among them.
* The algorithm assigns the most prevalent class to the new record. It's essentially a "peer group" classification, where the classification decision is based solely on records with identical predictor values.
* It's accurate but computationally expensive, making it less practical for large datasets.

**Naïve Bayes:**

* In the Naïve Bayes algorithm, classification for a new record isn't restricted to records that match the predictor profile of the new record. Instead, it considers the entire dataset.
* Similar to the Bayesian Classifier (Exact), the algorithm determines what classes the records belong to and identifies the most prevalent class across the entire dataset.
* The algorithm assigns the most prevalent class to the new record, without exclusively relying on the predictor profile of the new record. It assumes independence between predictor variables, which simplifies the computation of conditional probabilities.
* In data mining technologies like R, Naïve Bayes is commonly used because it's efficient and also works well, while Exact Bayes is not typically used directly due to its computational complexity, unless it's specifically required for a particular problem.

1. **Considering the following sample dataset. I compute in class the Complete Bayesian and Naïve Bayes probability for Class = fraudulent (F), Prior Legal Trouble = yes (y) and Company size = small (s). manually compute the rest of the Naïve Bayes probabilities.**

**Just a number for each case is not acceptable. You have to provide Naïve Bayesian formula for each case, and your work thru up to the final result for this particular data. (20 points)**

|  |  |  |  |
| --- | --- | --- | --- |
| Company | Prior Legal Trouble | Company Size | Status |
| 1 | Yes | Small | Truthful |
| 2 | No | Small | Truthful |
| 3 | No | Large | Truthful |
| 4 | No | Large | Truthful |
| 5 | No | Small | Truthful |
| 6 | No | Small | Truthful |
| 7 | Yes | Small | Fraudulent |
| 8 | Yes | Large | Fraudulent |
| 9 | No | Large | Fraudulent |
| 10 | Yes | Large | Fraudulent |

Naïve Bayesian probability (Pnb) for all other classes.

**Fraudulent Class:**

**Truthful Class:**

1. **Use the above dataset and compute the complete (Exact) Bayesian classifier probability for Class = fraudulent, Prior Legal Trouble = No, and Company Size = Large**

**(10 Points)**

The Bayesian formula for the calculation of this probability that a record belong to class i (i.e. Ci) is as follow.

**(Exact) Bayesian classifier probability:**

1. Predicting flight delay can be useful to people and variety of organizations. If this information were provided in advance in real-time, organizations would benefit from benefit from some advanced notice about flights that are likely to be delayed.

I prepared a csv file (FlightDelays.csv) with 2201 records. Build a Naïve Bayes model on the training dataset and apply the model on test (or validation) part. Make sure all attributes are categorical. Check the solution page for the textbook Naïve Bayes assignment for converting numeric values to categorical.

Your work should include the following activities. (20 Points)

1. Load the file into R
2. Partition it to 60% training, and 40% test
3. Build a Naïve Bayes model on the training dataset
4. Extract the posterior and prior probability of your model
5. Use the model and predict the flight status on test dataset.
6. Create a confusion matrix for prediction and actual values of flight status for the test dataset
7. Create a data frame for the test dataset which has the actual and predicted values.
8. Save the data frame in a csv file.
9. Interpret the outcome. In detail.

Do not submit your R code.

Submit

* the posterior and prior probabilities
* Confusion matrix of the training and test datasets
* The test dataset with additional column of flight status predictions
* Your interpretation of the model performance

**Solution: The posterior and prior probabilities:**

A prior is a probability distribution that represents what the model knows before seeing the data.

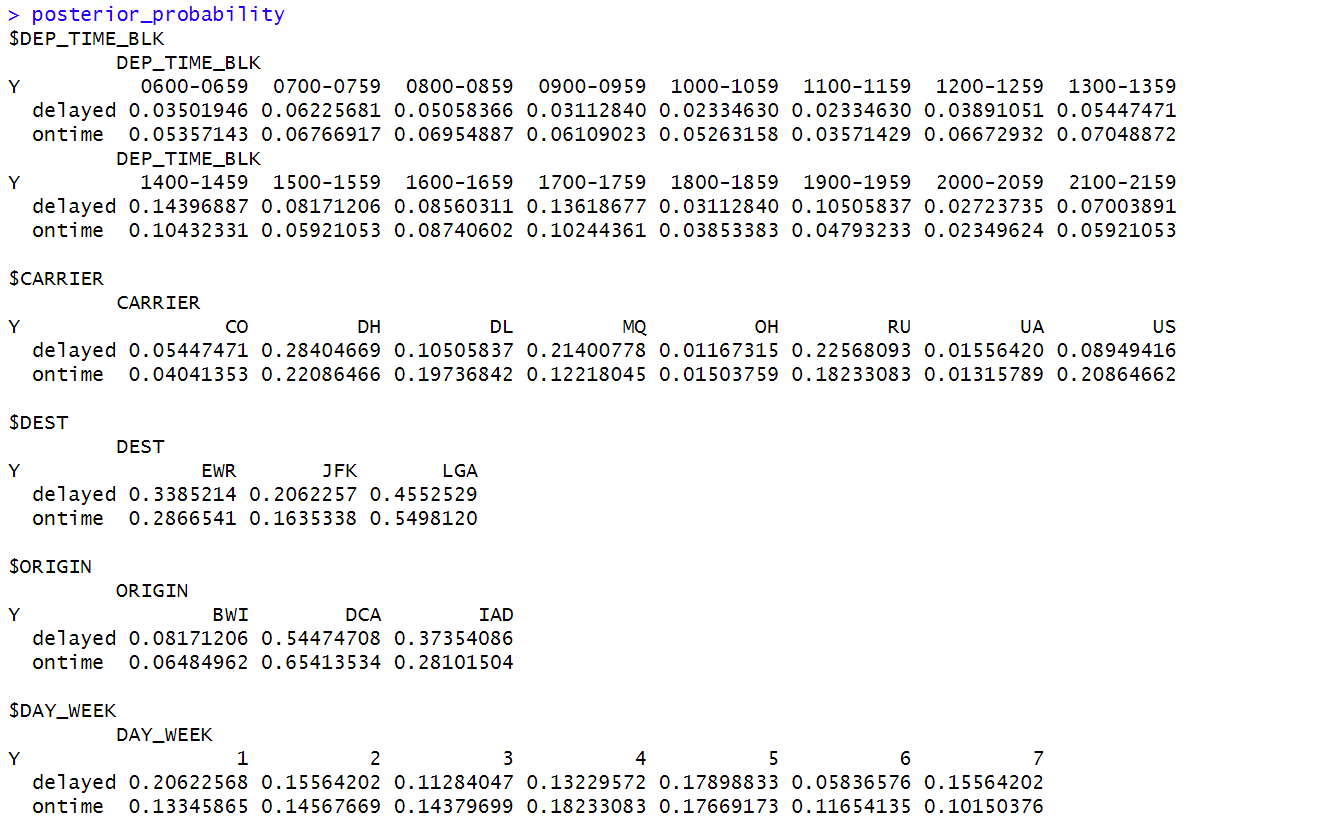
> prior\_probability

Y

delayed ontime

257 1064

A posterior is a probability distribution that represents what the model knows after having seen the data.



**Confusion matrix of the training and test datasets:**  
**Train data:**

**Confusion Matrix and Statistics**

Reference

Prediction delayed ontime

delayed 18 15

ontime 239 1049

Accuracy : 0.8077

95% CI : (0.7854, 0.8286)

No Information Rate : 0.8055

P-Value [Acc > NIR] : 0.4337

Kappa : 0.0836

Mcnemar's Test P-Value : <2e-16

Sensitivity : 0.07004

Specificity : 0.98590

Pos Pred Value : 0.54545

Neg Pred Value : 0.81444

Prevalence : 0.19455

Detection Rate : 0.01363

Detection Prevalence : 0.02498

Balanced Accuracy : 0.52797

'Positive' Class : delayed

**Test dataset:**

**Confusion Matrix and Statistics**

Reference

Prediction delayed ontime

delayed 7 9

ontime 164 700

Accuracy : 0.8034

95% CI : (0.7756, 0.8292)

No Information Rate : 0.8057

P-Value [Acc > NIR] : 0.5876

Kappa : 0.043

Mcnemar's Test P-Value : <2e-16

Sensitivity : 0.040936

Specificity : 0.987306

Pos Pred Value : 0.437500

Neg Pred Value : 0.810185

Prevalence : 0.194318

Detection Rate : 0.007955

Detection Prevalence : 0.018182

Balanced Accuracy : 0.514121

'Positive' Class : delayed

**Your interpretation of the model performance:**

The model's performance on both the training and test datasets is characterized by relatively high accuracy but low sensitivity (recall) for predicting delayed flights. In the training dataset, the model achieved an accuracy of approximately 80.77%, meaning it correctly classified a majority of the flights. But when focusing on predicting delayed flights, it had a low sensitivity of around 7.00%, indicating that it missed a significant number of actual delayed flights. On the test dataset, the model's accuracy was also reasonably high at about 80.34%, but its sensitivity for predicting delayed flights remained low at approximately 4.09%.

The above results says that the model can effectively classify on-time flights, it struggles to identify and capture delayed flights.