Report

Feature Detection/Methodology & Findings:

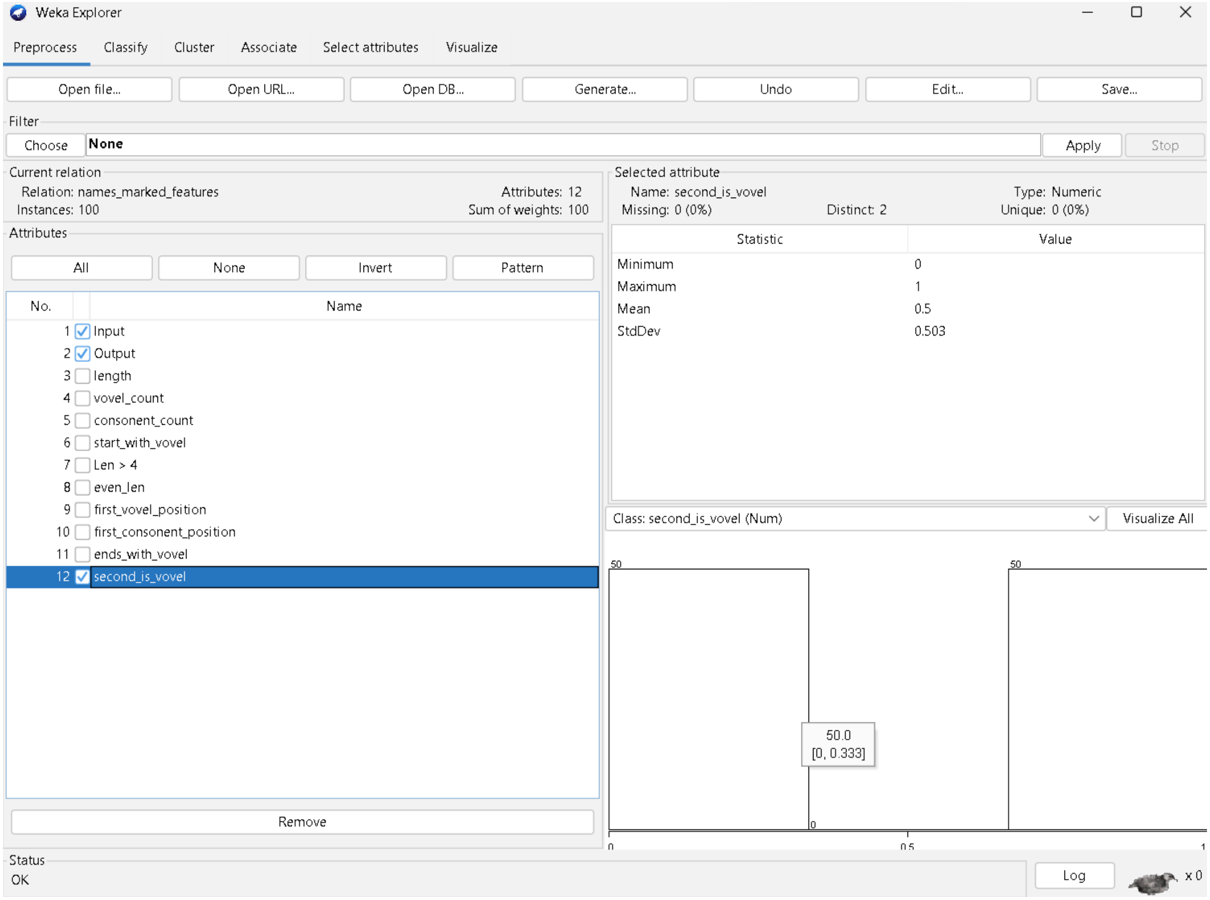
I took the provided data and the data labelled earlier and listed features in a single csv. Added features and attributes to the same file and loaded the file in weka for visualization with the feature set. I tried to take the attributes with truth/False values as I estimated that the classes are distributed in two. I was looking for the feature(s)/set of feature(s) that gives me the similar distribution of data as the number of classes.

Extracted Feature List: [length, vovel_count, consonent_count, start_with_vovel, Len>4, even_len, first_vovel_position, first_consonent_position, ends_with_vovel, second_is_vovel]

I checked repeatedly with many different combinations. But the distribution was nowhere near the number of classes for combinations. I also tried different classifiers with different feature combinations but most of them gave 50-100% error.

I found out on manual analysis that the data I included was either differently mapped or wrong. So I took the original dataset provided in the assignment google drive and grouped the class instances together to check the similarity and I easily found out that the feature seems to be the second vowel alphabet.

Then I checked the features separately. Adding closest feature distribution map here:



Filter: None

Current relation
Relation: names_marked_features   Attributes: 12
Instances: 100   Sum of weights: 100

Selected attribute
Name: second_is_vovel   Type: Numeric
Missing: 0 (0%)   Distinct: 2   Unique: 0 (0%)

Statistic	Value
Minimum	0
Maximum	1
Mean	0.5
StdDev	0.503

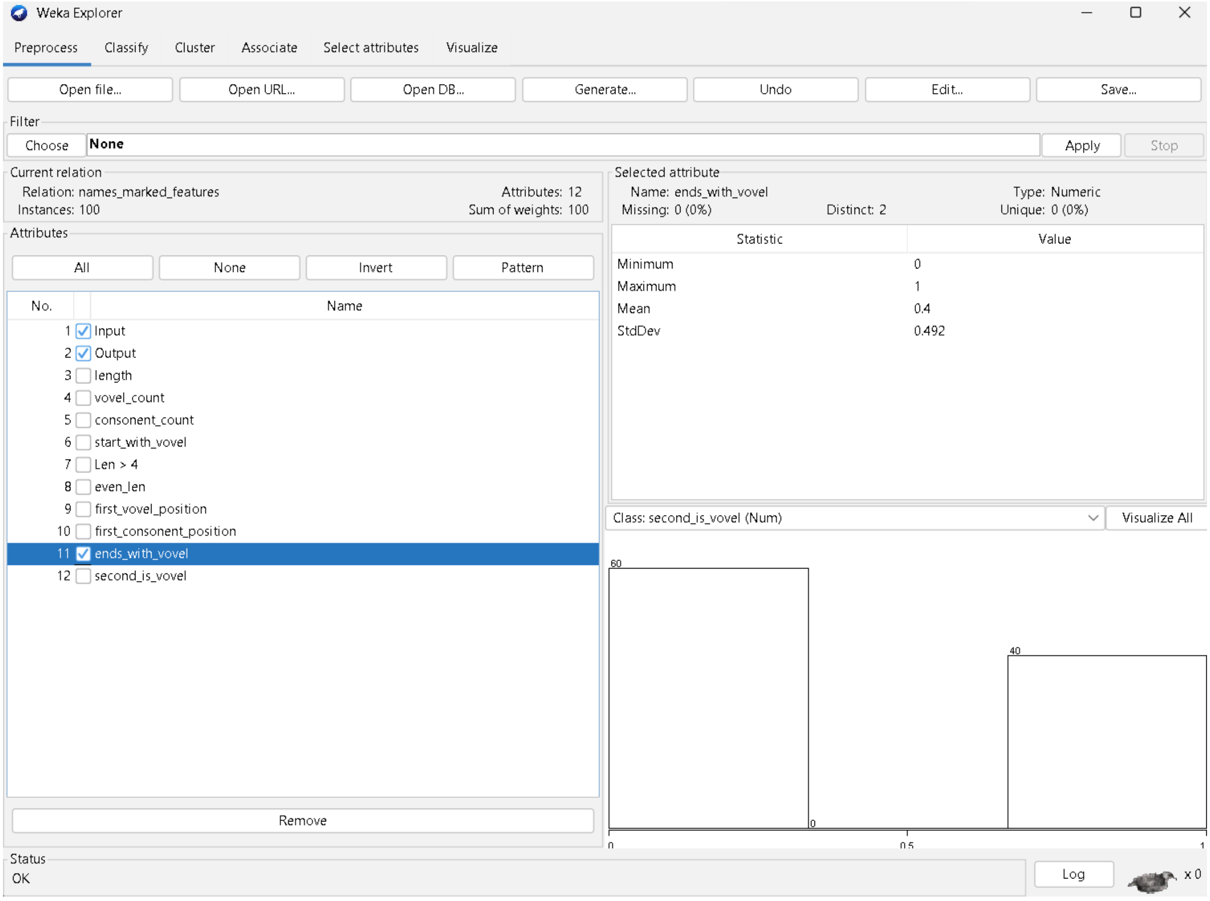
Attributes

No.		Name
1	☑	Input
2	☑	Output
3	□	length
4	□	vovel_count
5	□	consonent_count
6	□	start_with_vovel
7	□	Len > 4
8	□	even_len
9	□	first_vovel_position
10	□	first_consonent_position
11	□	ends_with_vovel
12	☑	second_is_vovel

Class: second_is_vovel (Num)

Status
OK

Figure 1 The correct Feature as per distribution



Filter: None

Current relation
Relation: names_marked_features
Instances: 100
Attributes: 12
Sum of weights: 100

Attributes

No.	Name
1	☑ Input
2	☑ Output
3	□ length
4	□ vovel_count
5	□ consonent_count
6	□ start_with_vovel
7	□ Len > 4
8	□ even_len
9	□ first_vovel_position
10	□ first_consonent_position
11	☑ ends_with_vovel
12	□ second_is_vovel

Selected attribute
Name: ends_with_vovel
Missing: 0 (0%)
Distinct: 2
Type: Numeric
Unique: 0 (0%)

Statistic	Value
Minimum	0
Maximum	1
Mean	0.4
StdDev	0.492

Class: second_is_vovel (Num)

Status
OK

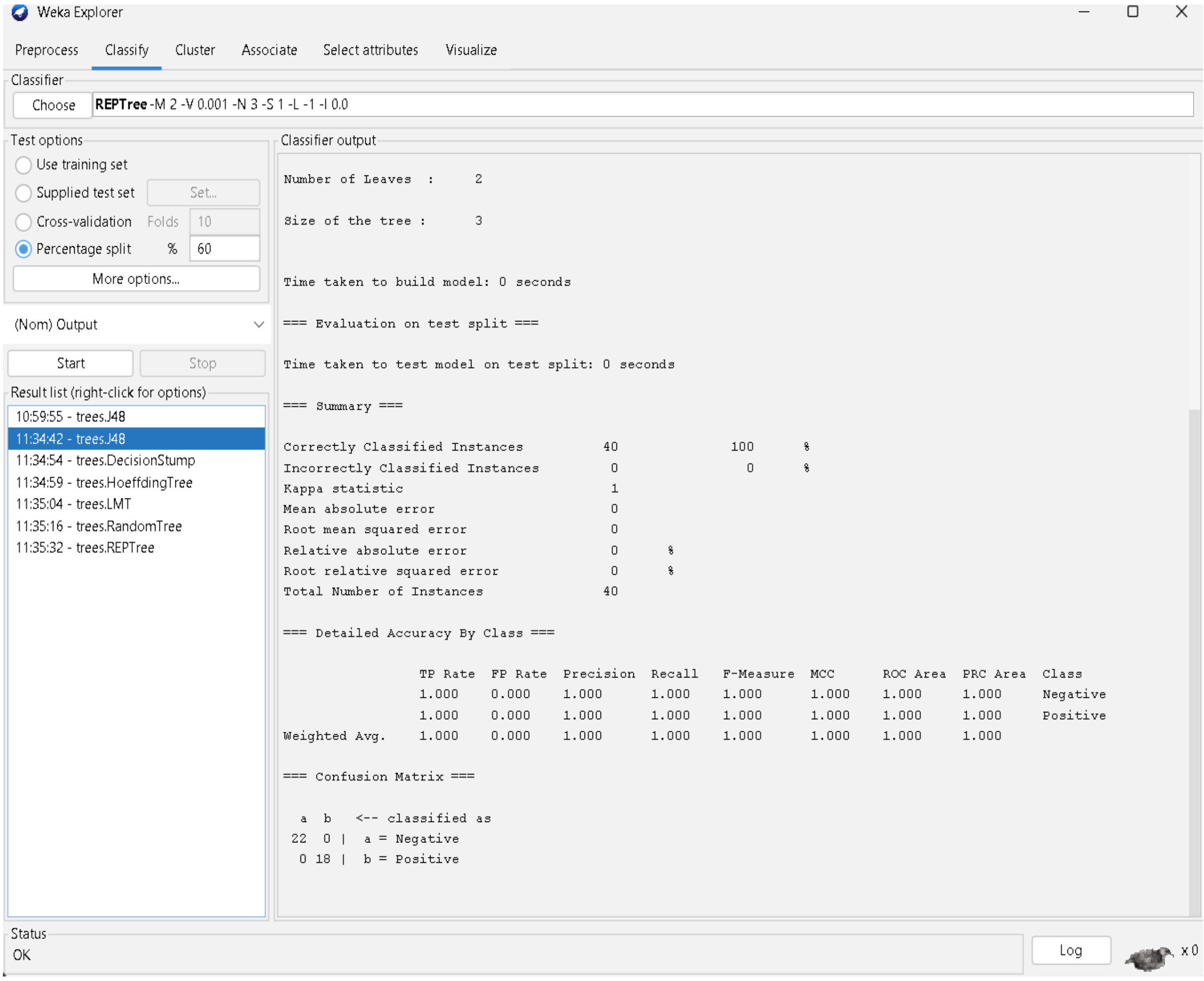
Figure 2 Checking for features

For my analysis, I checked and analyzed other feature data as per my own understanding and shortlisted some other features to experiment with. Although the results were expected, but there were some other relations as well, like first_vowel and consonant_count had distribution near that of results.

Results:

Since the feature detected was right, the classification result should be 100%. And with general decision tree classifiers, most of them gave 100% result.

Attaching the result of final tries of the classification algorithm implementation below:



Classifier: REPTree -M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0

Test options

○ Use training set
○ Supplied test set Set...
○ Cross-validation Folds 10
● Percentage split % 60

(Nom) Output

Result list (right-click for options)

- 10:59:55 - trees.J48
- 11:34:42 - trees.J48
- 11:34:54 - trees.DecisionStump
- 11:34:59 - trees.HoeffdingTree
- 11:35:04 - trees.LMT
- 11:35:16 - trees.RandomTree
- 11:35:32 - trees.REPTree

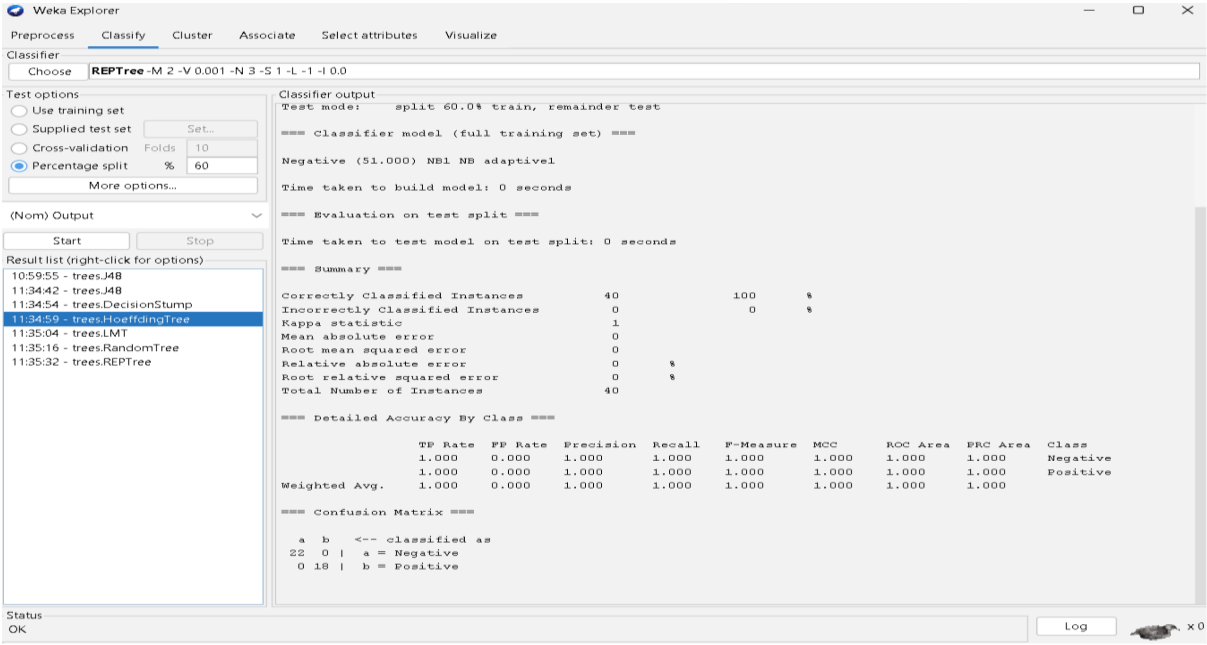
Classifier output

```
Number of Leaves  :     2

Size of the tree :      3


Time taken to build model: 0 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

Correctly Classified Instances          40              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances               40

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Negative
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Positive
Weighted Avg.    1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

  a  b   <-- classified as
 22  0 |  a = Negative
  0 18 |  b = Positive
```

Status
OK

Figure 3Final Result from J48 algorithm

Other algorithm with 100% results are as below:



Classifier: REPTree -M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0

Test options:
○ Use training set
○ Supplied test set
○ Cross-validation Folds 10
● Percentage split % 60

(Nom) Output

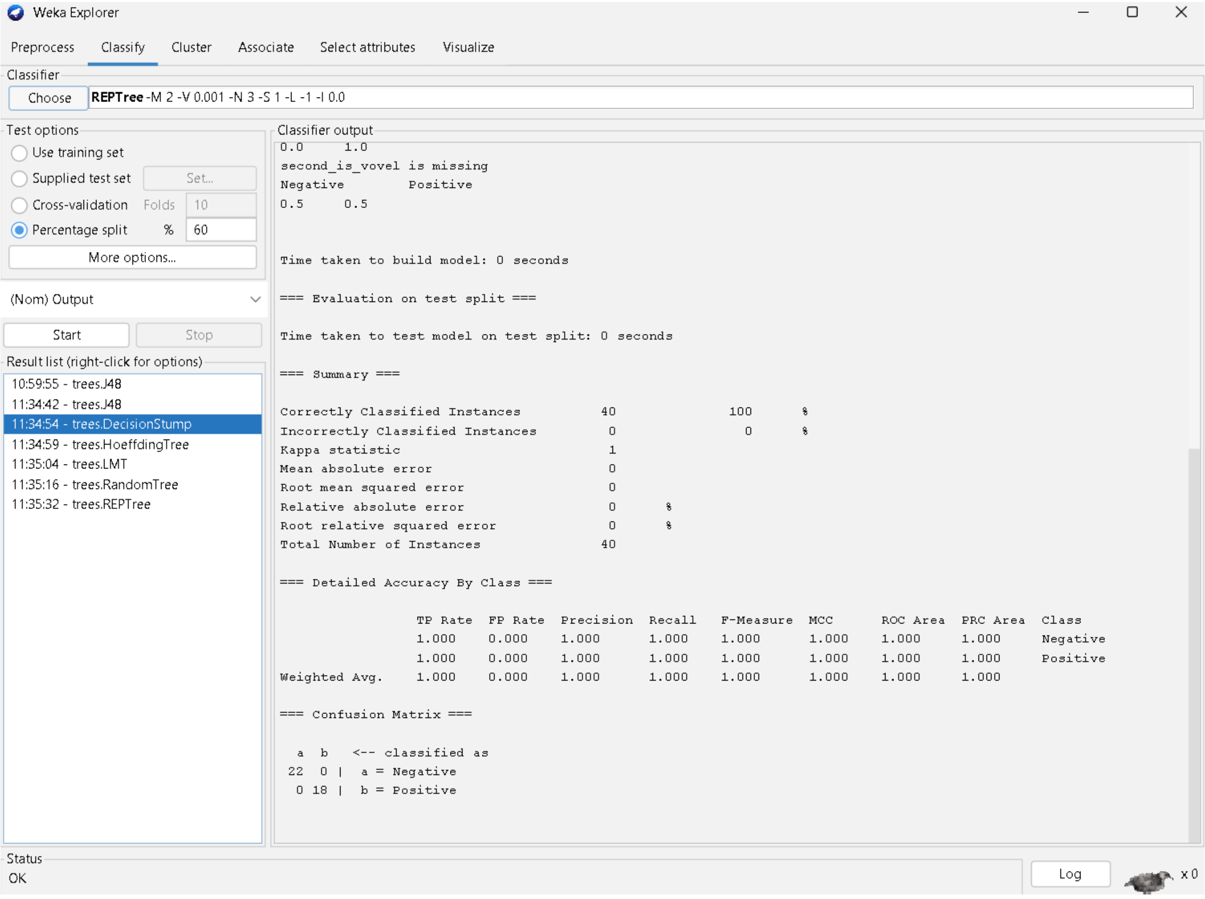
Result list (right-click for options)
- 10:59:55 - trees.J48
- 11:34:42 - trees.J48
- 11:34:54 - trees.DecisionStump
- 11:34:59 - trees.HoeffdingTree
- 11:35:04 - trees.LMT
- 11:35:16 - trees.RandomTree
- 11:35:32 - trees.REPTree

Classifier output

```
Test mode:    split 60.0% train, remainder test

=== Classifier model (full training set) ===

Negative (51.000) NB1 NB adaptive1

Time taken to build model: 0 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

Correctly Classified Instances          40              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances               40

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Negative
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Positive
Weighted Avg.    1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

  a  b   <-- classified as
 22  0 |  a = Negative
  0 18 |  b = Positive
```

Status
OK

Figure 4Hoeffding Tree Results



Classifier: REPTree -M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0

Test options:
○ Use training set
○ Supplied test set
○ Cross-validation Folds 10
● Percentage split % 60

(Nom) Output

Result list (right-click for options)
- 10:59:55 - trees.J48
- 11:34:42 - trees.J48
- 11:34:54 - trees.DecisionStump
- 11:34:59 - trees.HoeffdingTree
- 11:35:04 - trees.LMT
- 11:35:16 - trees.RandomTree
- 11:35:32 - trees.REPTree

Classifier output

```
0.0     1.0
second_is_vovel is missing
Negative        Positive
0.5     0.5


Time taken to build model: 0 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

Correctly Classified Instances          40              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances               40

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Negative
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Positive
Weighted Avg.    1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

  a  b   <-- classified as
 22  0 |  a = Negative
  0 18 |  b = Positive
```

Status
OK

Figure 5 DecisionStump results

Other algorithms like LMT, REPTree and RandomTree gave 26-100% errors in different tries and split