Method 1

At first the J48 algorithm was applied with a confidence factor of 0.25 and a minimum number of instances per leaf set to 2. The dataset was preprocessed by using the following filters:

1. Remove (all but num_of_helpful_votes, title and text)
2. StringToNominal
3. MathExpression
4. NumericToBinary
5. StringToWordVector (options for stopwords, tokenization, and stemming)

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3106               99.6151 %
Incorrectly Classified Instances        12                0.3849 %
Kappa statistic                         -0.0006
Mean absolute error                      0.0072
Root mean squared error                  0.0619
Relative absolute error                 98.1373 %
Root relative squared error            104.4601 %
Total Number of Instances             3118

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    1,000    0,996      1,000    0,998      -0,001   0,468     0,996     0
                 0,000    0,000    0,000      0,000    0,000      -0,001   0,468     0,003     1
Weighted Avg.    0,996    0,996    0,993      0,996    0,995      -0,001   0,468     0,993

=== Confusion Matrix ===

    a    b   <-- classified as
 3106    1 |    a = 0
   11    0 |    b = 1
```

Analysis:

As the decision three only has one leaf and the size of one, it essentially means no splits were made and a stump was created that always predicts the majority class. Even if the model achieved a high correct classification rate, the Kappa statistic is ~0, indicating that neither the low mean absolute error nor root mean squared error are informative.

As a summary: the high classification accuracy is misleading and the model is predicting the majority class for all instances as a result of the imbalance in the dataset instead of learning to discriminate between classes. As the model cannot correctly classify any instances of the minority class (b= >15) the model fails to meet the business needs despite is high accuracy.

Method 2

CostSensitiveClassifier is used to balance the classes before training the model to handle the imbalance in the data along with another model “RandomForest”. A cost matrix was applied where a false negative had a cost of 10 and a false positive a cost of 1.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3107               99.6472 %
Incorrectly Classified Instances        11                0.3528 %
Kappa statistic                          0
Mean absolute error                      0.0087
Root mean squared error                  0.0594
Relative absolute error                117.9095 %
Root relative squared error            100.1547 %
Total Number of Instances             3118

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    1,000    0,996      1,000    0,998      ?        0,804     0,999     0
                 0,000    0,000    ?          0,000    ?          ?        0,804     0,020     1
Weighted Avg.    0,996    0,996    ?          0,996    ?          ?        0,804     0,995

=== Confusion Matrix ===

    a    b   <-- classified as
 3107    0 |    a = 0
   11    0 |    b = 1
```

The results were very similar to the previous try, indicating the need of fine tuning. Next the cost matrix will be fine tuned and model parameters experimented. The cost was varied and finally up to 50. However, the results still showed no improvement.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3107               99.6472 %
Incorrectly Classified Instances        11                0.3528 %
Kappa statistic                          0
Mean absolute error                      0.0096
Root mean squared error                  0.0598
Relative absolute error                129.8134 %
Root relative squared error            100.8355 %
Total Number of Instances             3118

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    1,000    0,996      1,000    0,998      ?        0,663     0,998     0
                 0,000    0,000    ?          0,000    ?          ?        0,663     0,046     1
Weighted Avg.    0,996    0,996    ?          0,996    ?          ?        0,663     0,994

=== Confusion Matrix ===

    a    b   <-- classified as
 3107    0 |    a = 0
   11    0 |    b = 1
```

Even with the cost at 80 and by varying parameters for random forest, the results showed no improvement. As the value of 80 is already very high, it should not be increased further, as it may lead to the model not learning anything due to the cost being too prohibitive.

Thus, I moved on to try several other models. The best performing was to use a cost sensitive classifier along with J48.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3079               98.7492 %
Incorrectly Classified Instances        39                1.2508 %
Kappa statistic                          0.0438
Mean absolute error                      0.0122
Root mean squared error                  0.1086
Relative absolute error                164.6445 %
Root relative squared error            183.227  %
Total Number of Instances             3118

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,991    0,909    0,997      0,991    0,994      0,050    0,541     0,997     0
                 0,091    0,009    0,033      0,091    0,049      0,050    0,541     0,021     1
Weighted Avg.    0,987    0,906    0,993      0,987    0,990      0,050    0,541     0,993

=== Confusion Matrix ===

    a    b   <-- classified as
 3078   29 |    a = 0
   10    1 |    b = 1
```

Here the Kappa statistic is indicating at least a slight agreement beyond chance. In addition the model had a very high TPR for the below 15, but only 0.091 for the other part. However, it seems to manage to correctly identify a small number of instances of the above 15 class as well. This can be seen as an improvement of the previous tries.

While the model shows a clear improvement, it can still be improved to better identify the minority class. Let's go ahead and try to increase the cost to 90 as a test and adjust the confident factor and minimum number of instances per leaf in J48. We will also lower the threshold below 0.5 to increase the sensitivity for the minority class.

By setting the cost to 90, min number of instances to 50 and the confident factor to 0.1, we received the following result:

```
Time taken to build model: 2.21 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        2891               92.7197 %
Incorrectly Classified Instances       227                7.2803 %
Kappa statistic                          0.0192
Mean absolute error                      0.0591
Root mean squared error                  0.2191
Relative absolute error                800.5507 %
Root relative squared error            369.4486 %
Total Number of Instances             3118

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,930    0,727    0,997      0,930    0,962      0,047    0,598     0,997     0
                 0,273    0,070    0,014      0,273    0,026      0,047    0,598     0,011     1
Weighted Avg.    0,927    0,725    0,994      0,927    0,959      0,047    0,598     0,994

=== Confusion Matrix ===

    a    b   <-- classified as
 2888  219 |    a = 0
    8    3 |    b = 1
```

This shows a slight decrease in the Kappa statistic, but an improvement in the TPR for the minority.

To further receive a better result, some other models or more advanced sampling techniques could be utilized.