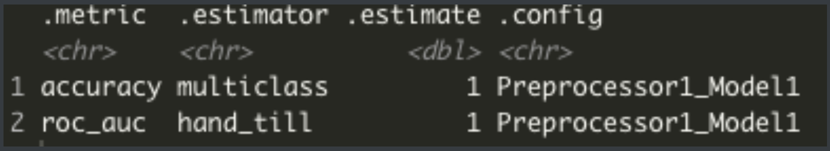
Classifying All Data

First Stage Classifier

This random forest model classifies the shape of the data based on x to y ratio, number of squares, fractal dimension and density skewness.



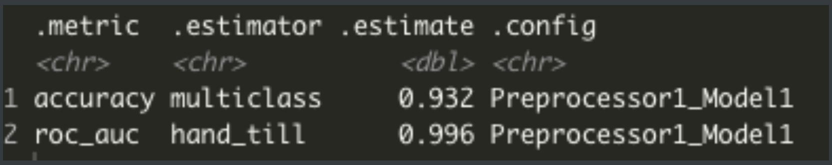
```
  .metric  .estimator .estimate .config
  <chr>    <chr>          <dbl> <chr>
1 accuracy multiclass         1 Preprocessor1_Model1
2 roc_auc  hand_till          1 Preprocessor1_Model1
```

Using this model, we have obtained a set of labeled data which can now be used to train a model which predicts shape based on parameter values.

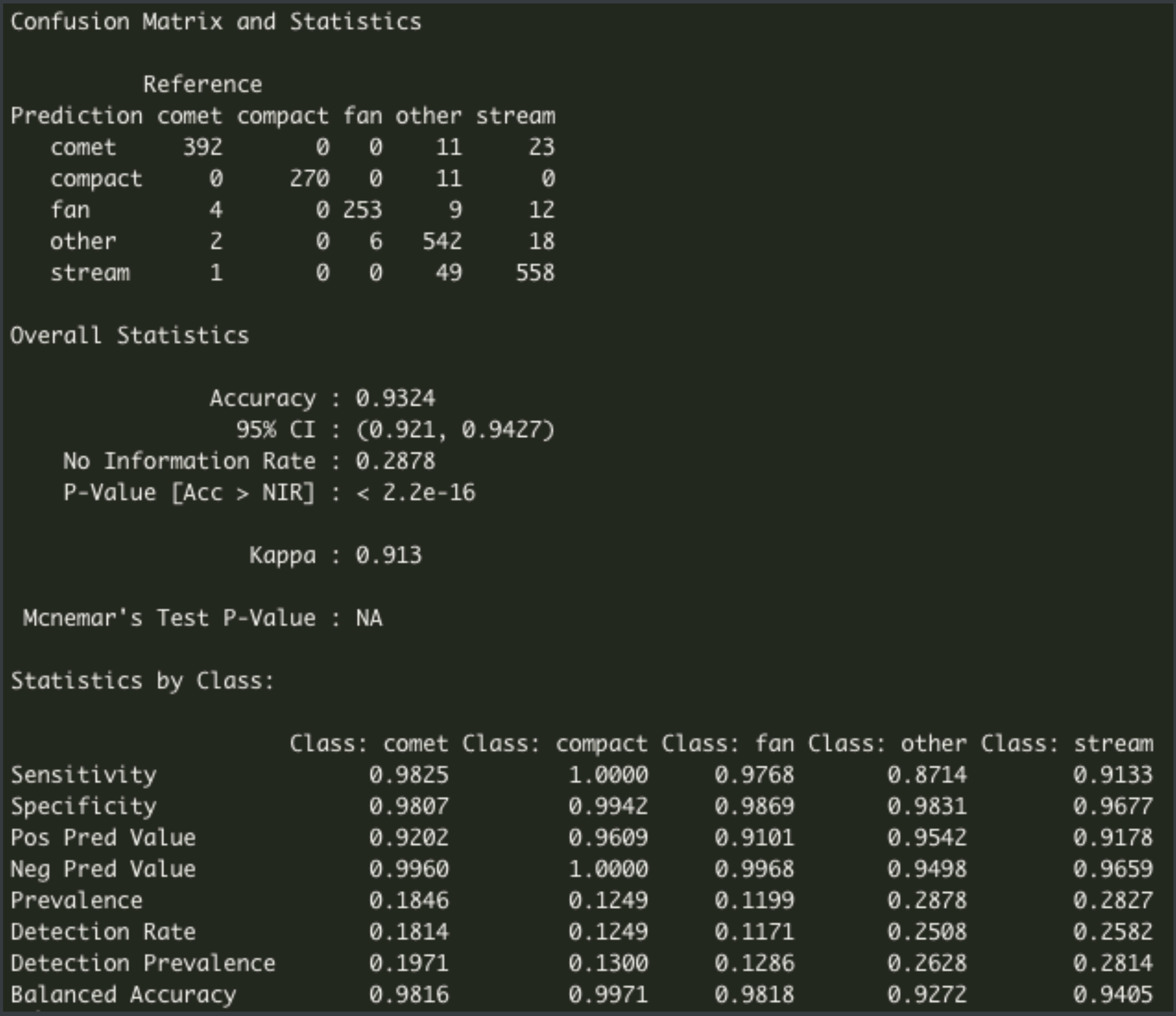
Second Stage Classifier

This random forest model classifies the shape based on parameter values. I tried this treating the parameters as factors and as numeric variables.

Treating params as **factors**:

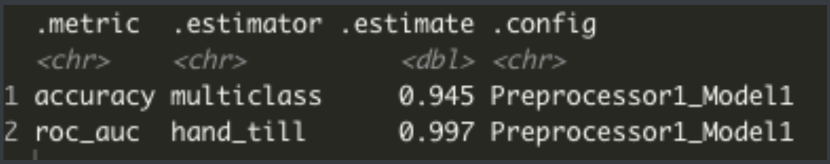


```
  .metric  .estimator .estimate .config
  <chr>    <chr>          <dbl> <chr>
1 accuracy multiclass     0.932 Preprocessor1_Model1
2 roc_auc  hand_till      0.996 Preprocessor1_Model1
```

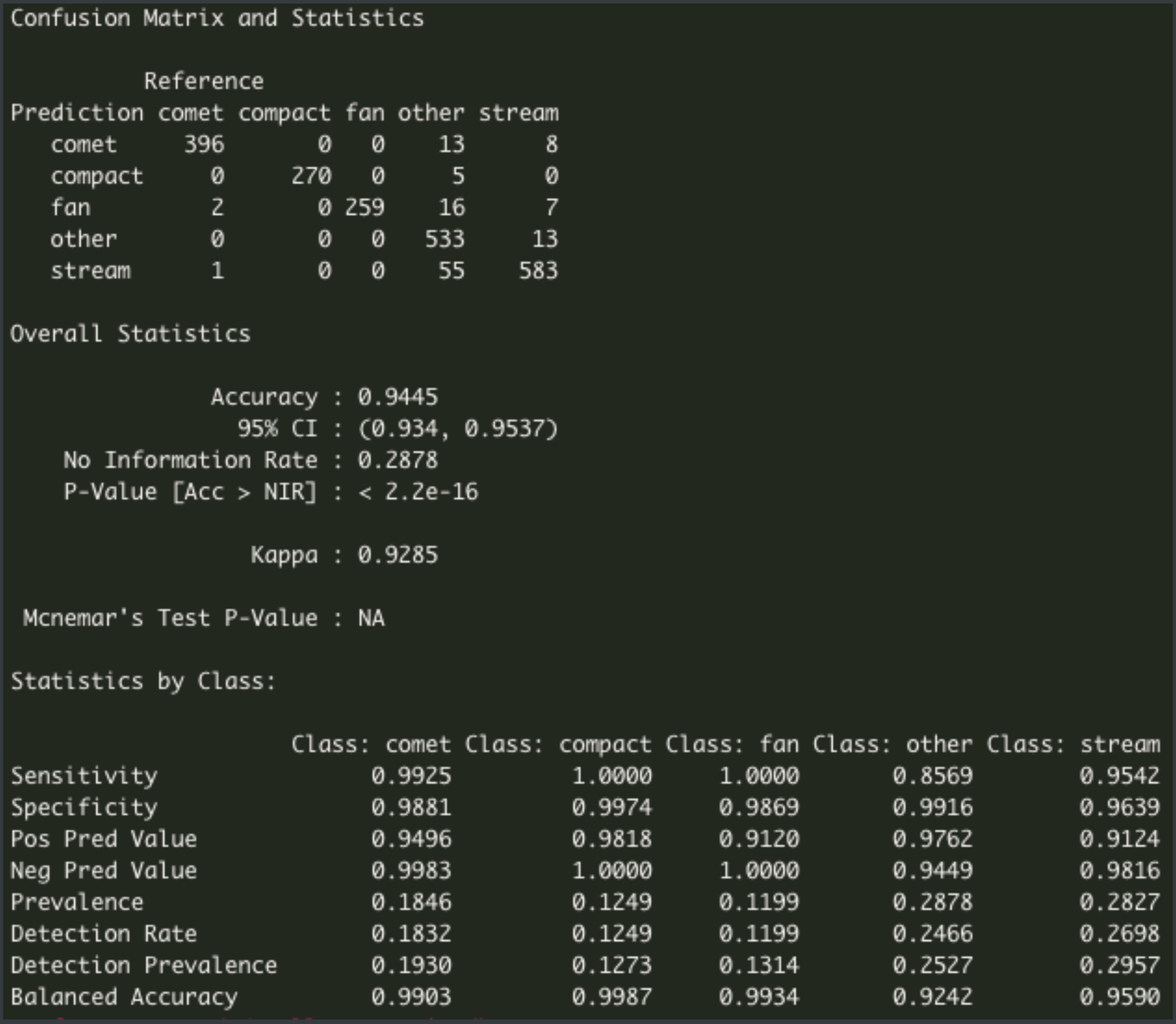


```
Confusion Matrix and Statistics

          Reference
Prediction comet compact fan other stream
   comet     392       0   0    11     23
   compact     0     270   0    11      0
   fan         4       0 253     9     12
   other       2       0   6   542     18
   stream      1       0   0    49    558

Overall Statistics

               Accuracy : 0.9324
                 95% CI : (0.921, 0.9427)
    No Information Rate : 0.2878
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.913

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: comet Class: compact Class: fan Class: other Class: stream
Sensitivity                0.9825         1.0000     0.9768       0.8714        0.9133
Specificity                0.9807         0.9942     0.9869       0.9831        0.9677
Pos Pred Value             0.9202         0.9609     0.9101       0.9542        0.9178
Neg Pred Value             0.9960         1.0000     0.9968       0.9498        0.9659
Prevalence                 0.1846         0.1249     0.1199       0.2878        0.2827
Detection Rate             0.1814         0.1249     0.1171       0.2508        0.2582
Detection Prevalence       0.1971         0.1300     0.1286       0.2628        0.2814
Balanced Accuracy          0.9816         0.9971     0.9818       0.9272        0.9405
```

Treating params as **numeric**:



```
  .metric  .estimator .estimate .config
  <chr>    <chr>          <dbl> <chr>
1 accuracy multiclass     0.945 Preprocessor1_Model1
2 roc_auc  hand_till      0.997 Preprocessor1_Model1
```



```
Confusion Matrix and Statistics

          Reference
Prediction comet compact fan other stream
   comet     396       0   0    13      8
   compact     0     270   0     5      0
   fan         2       0 259    16      7
   other       0       0   0   533     13
   stream      1       0   0    55    583

Overall Statistics

               Accuracy : 0.9445
                 95% CI : (0.934, 0.9537)
    No Information Rate : 0.2878
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9285

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: comet Class: compact Class: fan Class: other Class: stream
Sensitivity                0.9925         1.0000     1.0000       0.8569        0.9542
Specificity                0.9881         0.9974     0.9869       0.9916        0.9639
Pos Pred Value             0.9496         0.9818     0.9120       0.9762        0.9124
Neg Pred Value             0.9983         1.0000     1.0000       0.9449        0.9816
Prevalence                 0.1846         0.1249     0.1199       0.2878        0.2827
Detection Rate             0.1832         0.1249     0.1199       0.2466        0.2698
Detection Prevalence       0.1930         0.1273     0.1314       0.2527        0.2957
Balanced Accuracy          0.9903         0.9987     0.9934       0.9242        0.9590
```

The performance is quite good overal, though it is important to note that these metrics only capture one layer of predictive accuracy, as we are using predicted labels as our ground truth for this model. We see that in each case, the poorest performance is in the *other* and *stream* classes.