

#### 4.4.10 Ordered One-vs.-All

As stated in Section 4.4.2, the key problem of this competition is to reduce the high rate of misclassification of cover types 1 and 2. Figure 10 shows a confusion matrix that is produced by an Extra Trees model. It shows that type 1 and type 2 are still hard to separate and some type 1s are also misclassified as type 5 and 7; some type 2s are misclassified as type 4, 5 or 6. Because of the high prediction accuracy of types 7, 6, 5, 4, I used an ordered One-vs.-All approach for classes 7, 6, 5, 4 to prevent inaccurate prediction for types 1 and 2. The classification process is demonstrated in Table 9.

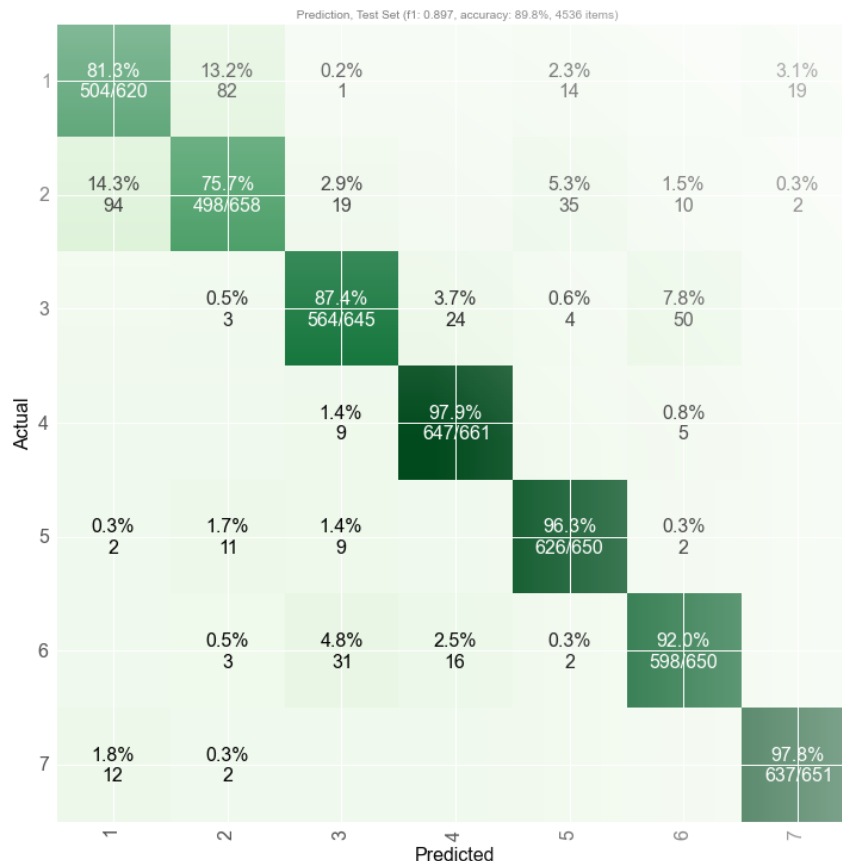


Figure 10: Confusion matrix of extra trees

Table 9: Ordered One-vs.-All

Cover type	Training set	Test set	Classified instances
7	$x_7$ vs. $(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$	$y$	$y_7$
6	$x_6$ vs. $(x_5 + x_4 + x_3 + x_2 + x_1)$	$y - y_7$	$y_6$
5	$x_5$ vs. $(x_4 + x_3 + x_2 + x_1)$	$y - y_7 - y_6$	$y_5$
4	$x_4$ vs. $(x_3 + x_2 + x_1)$	$y - y_7 - y_6 - y_5$	$y_4$
3, 2, 1	$x_3 + x_2 + x_1$	$y - y_7 - y_6 - y_5 - y_4$	$y_3, y_2, y_1$

Let  $x_i$  be the subset of the training data, which is labeled with cover type  $i$ , and  $y_j$  be the subset of test data  $y$ , which is predicted to be cover type  $j$ .

For example, to predict type 7, complete training dataset is used in the training phase, but the instances of cover type 7 are labeled as positives and all other instances are labeled as negatives to prevent other types (especially type 1) from being misclassified as type 7. After prediction, the test instances classified as cover type 7 are labeled "7" and are removed from the test set. The classification process for cover type 6 is the same as that of 7, but the training instances of cover type 7 are removed from the training set. Repeat the same approaches for cover types 5 and 4 and we get the classified instances  $y_7$ ,  $y_6$ ,  $y_5$  and  $y_4$ . Types 1, 2, 3 are classified using the general multi-class classification method.

The Extra Trees algorithm, with parameters selected from the previous step, was used in each step to get a submission, and it scored 0.82269 on the leaderboard.

#### 4.4.11 Final result

Up to now the competition is closed, my final submission scored 0.82269 and ranked 29th out of 1694 teams, which is quite a satisfactory result. An overview of the leaderboard is shown in Table 10. The row in bold is my score.

Table 10: Public leaderboard overview of first competition

Ranking	Team Name	Score
1	antgleb	1.00000
2	Ashish Singh	0.99999
3	ucbw207_2_forest	0.99751
...	...	...
27	Michiel	0.82408
28	DATS36G	0.82281
<b>29</b>	<b>Ying Dong</b>	<b>0.82269</b>
30	hderksen	0.81963
31	Jack Dempsey	0.81950
...	...	...
1692	soumyajyoti	0.00003
1693	Ethan Rosenthal	0.00000
1694	Hiokei Chan	0.00000