Classification of Forest Cover

John Courville

Abstract

This writeup outlines the procedures and results of classification of forest cover type. The cover types are [Spruce/Fir, Lodgepole Pine, Cottonwood/Willow, Aspen, Douglas-fir, Krummholz]. Features were collected from data obtained from the US Geological Survey and USFS. The studied area is in the Roosevelt National Forrest of northern Colorado.

Data Management

The goal was to predict the class of cover based on the features of a 30x30 meter cell of land in the studied area. The data was already cleaned but did need a little manipulation. I separated the 'class' column from the label set and removed the label column from the data to produce the features set. All the features were numerical, but some were categorical, thus all other features were scaled using Z-score Normalization.

Training Data

The training data contained the labels of each 30x30 meter cell and the 54 features. These features included but were not limited to elevation, slope, distance to water, distance to roads, etc. The labels were numerically classified 1-7 and were unevenly distributed with two of the classes making more than 80% of the data. Before removing portions of the majority data, I wanted to see how the model would perform without this change, however I did stratify the train and test sets to have equal proportions of each class.

Model Choices

Because of the size of the data(581012 entries and 55 features each), the original plan was to use a neural network but, I chose to first use a simple random forest classification and see if I could improve upon the performance.

Results

When running the random forest model, I achieved very good results. (Image 1). I then performed the same model using 5-fold stratification and produced results that were poor in comparison. (Image 2). In response to the poor performance, I ran another model wich shuffled the samples before stratification and it performed similarly to the first model (Image 3

Image 1

	precision	recall	f1-score	support
0	1.00	1.00	1.00	0
1	0.97	0.93	0.95	63552
2	0.95	0.97	0.96	84991
3	0.95	0.94	0.95	10726
4	0.93	0.79	0.86	824
5	0.95	0.70	0.81	2848
6	0.95	0.86	0.90	5210
7	0.98	0.94	0.96	6153
micro avg	0.96	0.95	0.95	174304
macro avg	0.96	0.89	0.92	174304
weighted avg	0.96	0.95	0.95	174304
samples avg	0.96	0.95	0.95	174304

Image 2

```
Maximum Accuracy That can be obtained from this model is: 63.74933305795081 %

Minimum Accuracy: 55.31660384502849 %

Overall Accuracy: 59.47777488038665 %

Standard Deviation is: 0.038780215611878915
```

Image 3

```
Maximum Accuracy That can be obtained from this model is: 95.64895612812172 %

Minimum Accuracy: 95.42524719671609 %

Overall Accuracy: 95.54157241159224 %

Standard Deviation is: 0.0008858774809053015
```

The neural net produced poorer results without stratification (Image 4) but slightly better when stratifying and shuffling before fitting(Image 5)

Image 4

	precision	recall	f1-score	support
1	0.92	0.89	0.91	63552
2	0.91	0.94	0.93	84991
3	0.90	0.91	0.90	10726
4	0.87	0.74	0.80	824
5	0.81	0.65	0.72	2848
6	0.82	0.80	0.81	5210
7	0.91	0.92	0.92	6153
accuracy			0.91	174304
macro avg	0.88	0.84	0.85	174304
weighted avg	0.91	0.91	0.91	174304

Image 5

	precision	recall	f1-score	support
1	0.95	0.97	0.96	42368
2	0.97	0.96	0.97	56660
3	0.97	0.95	0.96	7151
4	0.89	0.83	0.86	549
5	0.91	0.85	0.88	1898
6	0.93	0.94	0.93	3474
7	0.99	0.90	0.94	4102
accuracy			0.96	116202
macro avg	0.94	0.92	0.93	116202
weighted avg	0.96	0.96	0.96	116202

Conclusions and Afterthoughts

- Shuffling and stratifying the training sets can drastically affect the performance. Especially when the sample has unbalanced label sets.
- I would suspect that we could improve the speed of these models by removing features that are not as impactful. One could run simple classification models on each feature against the label set and eliminate ones that fail at providing any predictive use.