Classification Competition

William G. Hatcher, Kevin McNamara
Department of Computer and Information Sciences
Towson University, Maryland, USA 21252, USA
Emails: whatch2@students.towson.edu, kmcnamara@towson.edu

***Abstract*—Given labeled training and unlabeled testing datasets of forest cover types, we have conducted supervised learning to predict the class label "Cover_Type". The data includes over 50 attributes of binary and quantitative numeric values. We have tested many different learning algorithm with a variety of settings, determining the Random Forest to perform the best. Our test accuracy, using a 70%/30% training/testing split of the *training* dataset, was approximately 95%.**

***Index Terms*—Data mining, Machine learning, Classification**

I. Classification Methodology

We tested K-Nearest-Neighbors, Decision Tree, Random Forest, and Naive Bayes algorithms, which we denote as *Basic Learning* and *Improved Basic Learning* algorithms. Results can be seen in Tables I through IV. In Tables I and II, we have K-Nearest-Neighbors initialized with 3 neighbors (KNN-3), Decision Trees with gini (DT-gini) and entropy (DT-entropy) as attribute splitting mechanisms, Random Forest with 10 nodes (RF-10), and Naive Bayes (NB-gaussian). In Tables III and IV, we have K-Nearest-Neighbors initialized with 7 neighbors (KNN-7), Decision Trees with entropy and minimum leeaves set to 5 (DT-entropy, min-leaf-5), Extra Tree with entropy (ET-entropy) as the attribute splitting mechanisms, Random Forest with 100 nodes (RF-100), and Multinomial Naive Bayes (NB-multinomial).

In testing these Basic Learning mechanisms, because they trained quite quickly, we were able to train different models with Full Sets of the data (Full Set 1 and Full Set 2), with only the Binary attributes (Binary Only 1), such as 2702, 2703, 2704, etc., and with only Numeric attributes (Numeric Only 1 and Numeric Only 2). We then compared these different models to determine the impacts on our accuracy results. For example, from Table I, we see that the accuracy was the highest in the full sets of attributes. We also applied Min-Max normalization in our *Improved Basic Learning* models for comparison, with only slight improvements.

In addition to the basic learning models, we also analyzed *Advanced Learning* and *Improved Advanced Learning* models. These include Logistic Regression (LogReg), Neural Networks (NN-30x3), SVM, Bagging, and Boosting algorithms. The results of these models can be seen in Tables V and VI. Clearly, these models did not perform nearly as well as the basic algorithms, and also took significantly longer to train.

Finally, based on the results noted above, we determined the Random Forest algorithm to perform the best. Thus we tested multiple versions of the RF algorithm with a variety of settings. In Table VII we can see the results of this study. Here, the "-#" indicates the number of estimators in the forest. So, for example, RF-250 has 250 estimators. In addition, we were also able to apply entropy as the decision criterion, denoted as "-entropy" in the table. Finally, the RF-250-entropy model in **red** performed the best, and was used to create our final *Prediction.csv* file on the unlabeled test data.

TABLE I. Basic Learning - Accuracy

ACCURACY	Full Set 1	Full Set 2	Binary Only 1	Numeric Only 1	Numeric Only 2
KNN-3	0.596	0.596	0.415	0.596	0.596
DT-gini	0.927	0.927	0.540	0.897	0.897
DT-entropy	**0.930**	**0.930**	0.540	0.910	0.909
RF-10	0.925	0.927	**0.546**	**0.918**	**0.917**
NB-gaussian	0.634	0.634	0.488	0.634	0.634

TABLE II. Basic Learning - F1-Score

ACCURACY	Full Set 1	Full Set 2	Binary Only 1	Numeric Only 1	Numeric Only 2
KNN-3	0.590	0.590	0.400	0.590	0.590
DT-gini	**0.930**	**0.930**	0.540	0.900	0.900
DT-entropy	**0.930**	**0.930**	0.540	0.910	0.910
RF-10	0.920	**0.930**	**0.550**	**0.920**	**0.920**
NB-gaussian	0.640	0.640	0.320	0.640	0.640

On an additional note, we applied TensorFlow and Keras machine learning libraries, as denoted in Table VIII to train dense neural networks of shapes 250-100-10-7 and 500-250-10-7. These were trained on a variety of epochs and batch sizes, reported as "_epochs_batch" in the table. Again, we trained the models on only binary data and only multivariate data ("-bin" and -"multi") to determine the performance of data subsets. These again performed worse. It is also notable that the final accuracies calculated, denote by Accuracy*, we not the same as the training accuracy reported by the TensorFlow API, and it is not clear what the difference is. For this reason, this method was not ultimately used.

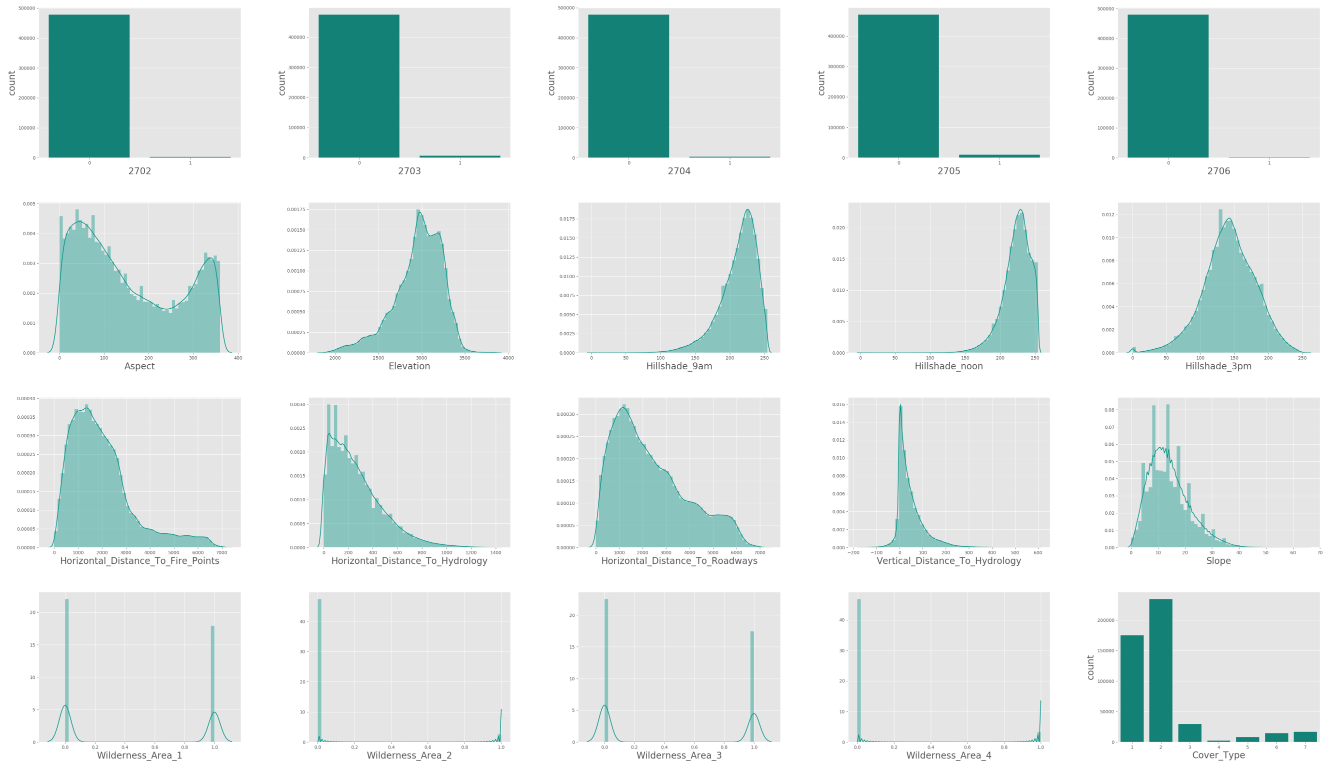

Fig. 1. Bar charts, histograms, and KDE plots of some of the dataset attributes.

TABLE III. Improved Basic Learning - Accuracy

ACCURACY	Full Set 1	Full Set 2	Full Set 1 min-max	Full Set 2 min-max
KNN-7	0.596	0.596	0.875	0.875
DT-entropy, min-leaf-5	0.921	0.921	0.921	0.921
ET-entropy	0.825	0.816	0.841	0.822
RF-100	**0.942**	**0.942**	**0.943**	**0.943**
NB-multinomial	-	-	0.642	0.642

TABLE IV. Improved Basic Learning - F1-Score

ACCURACY	Full Set 1	Full Set 2	Full Set 1 min-max	Full Set 2 min-max
KNN-7	0.590	0.590	0.870	0.870
DT-entropy, min-leaf-5	0.920	0.920	0.920	0.920
ET-entropy	0.820	0.820	0.840	0.820
RF-100	**0.940**	**0.940**	**0.94**0	**0.940**
NB-multinomial	-	-	0.620	0.620

TABLE V. Advanced Learning - Accuracy

ACCURACY	Full Set	Binary Only	Numeric Only	Numeric Only min-max
LogReg	0.717	0.647	**0.660**	0.665
NN-30x3-sgd	**0.851**	**0.652**	0.487	**0.804**
NN-30x3-lbfgs	0.810	0.651	0.481	0.744
SVM-rbf	0.790	0.651	-	0.770
Bagging	0.113	0.126	-	0.655
Boosting	0.418	0.613	-	0.613

TABLE VI. Advanced Learning - F1-Score

ACCURACY	Full Set	Binary Only	Numeric Only	Numeric Only min-max
LogReg	0.700	0.630	**0.630**	0.640
NN-30x3-sgd	**0.850**	**0.640**	0.240	**0.800**
NN-30x3-lbfgs	0.810	0.640	0.370	0.740
SVM-rbf	0.810	0.640	-	0.740
Bagging	0.660	0.740	-	0.640
Boosting	0.580	0.620	-	0.610

TABLE VII. Top Basic Learning - Random Forests

FULL DATA min-max	Accuracy	Precision	Recall	F1-Score
RF-250	0.943	0.94	0.94	0.94
RF-500	0.943	0.94	0.94	0.94
RF-1000	0.943	0.94	0.94	0.94
RF-100-entropy	0.945	**0.95**	**0.95**	0.94
RF-250-entropy	**0.951**	**0.95**	**0.95**	**0.95**

TABLE VIII. Top Advanced Learning - Dense Neural Networks with TensorFlow

FULL DATA min-max	Accuracy*	Precision	Recall	F1-Score
250-100-10-7_1-64	0.937*	0.78	0.77	0.76
250-100-10-7_1-64-bin	0.899*	0.62	0.63	0.61
250-100-10-7_1-64-mult	0.917*	0.69	0.7	0.69
250-100-10-7_32-16	**0.968***	**0.90**	**0.88**	**0.89**
500-250-100-10-7_32-16	0.958*	0.84	0.85	0.84