香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Explore Forest Cover Type Classification

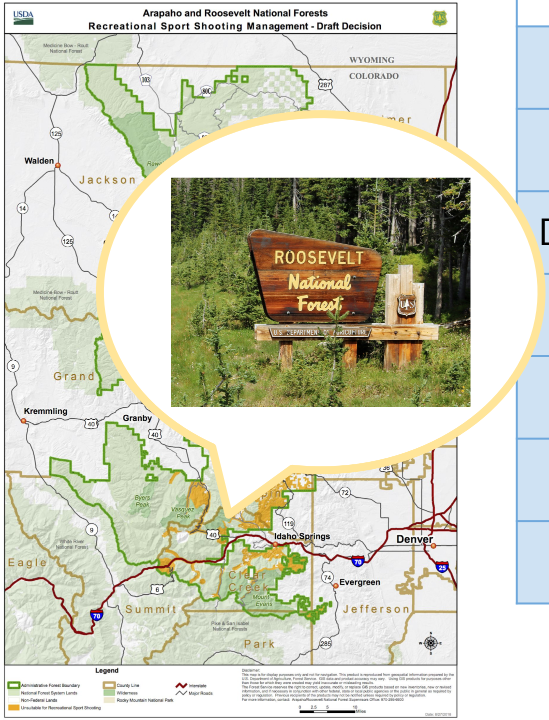
Group 25: Li, Zhuoyuan (20740917)
Zhang, Xinyue (20750194)
Qiao, Shuyu (20747563)

2020/12/01

Content

- **Introduction**
- **Preliminary Works**
- **Models**
  - Logistic Regression
  - Random Forest
  - LightGBM
  - Neural Network
  - Distributed decision tree
- **Conclusion**
- **Contribution**

Introduction

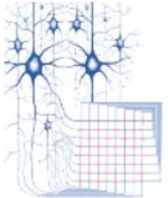


Input Features
Elevation
Slope
Distance to Hydrology
Hillshade
Wilderness_Area
Soil_Type
…

Classification Model

Labels
Spruce/Fir
Lodgepole Pine
Ponderosa Pine
Cottonwood/Willow
Aspen
Douglas Fir
Krummholz

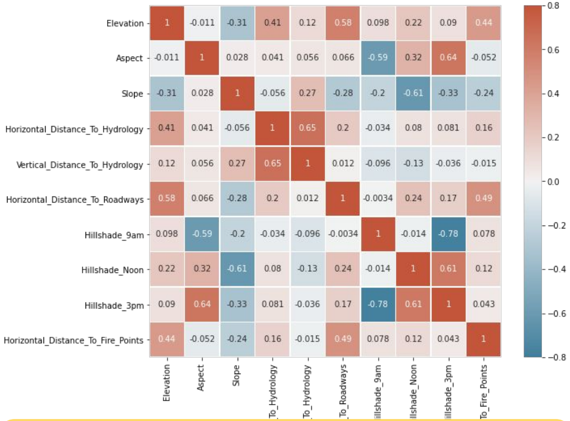
scikit learn

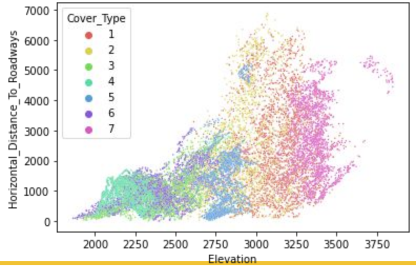


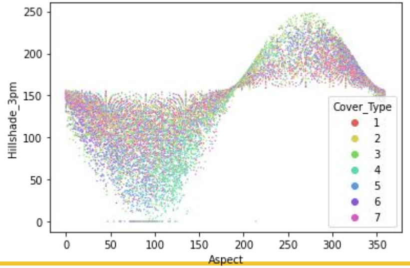
APACHE Spark

Data Overview

Check Missing Value	None
Drop non-informative Columns	Soil type 7 and 15 Drop
Correlaion Analysis (threshold = ±0.5)	1) **Horizontal distance to hydrology** - **vertical distance** (+) 2) **Hill-shade at 3pm** - **aspect** (+) 3) **Hill-shade at noon** - **hill-shade at 3pm** (+) 4) **Hill-shade at noon** - **slope** (-) 5) **Hill-shade at 9am** - **aspect** (-) 6) **Horizontal distance to the roadway** - **elevation** (+)

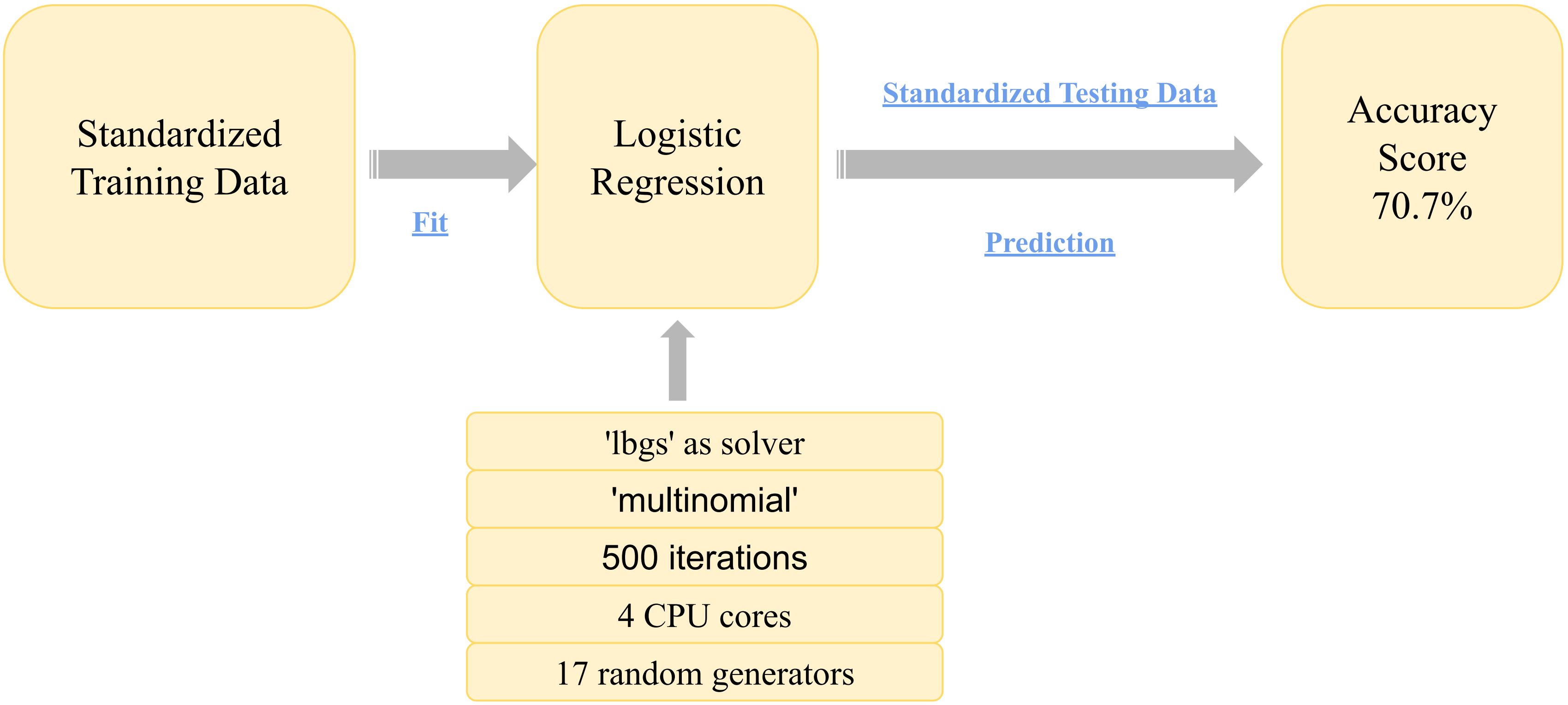




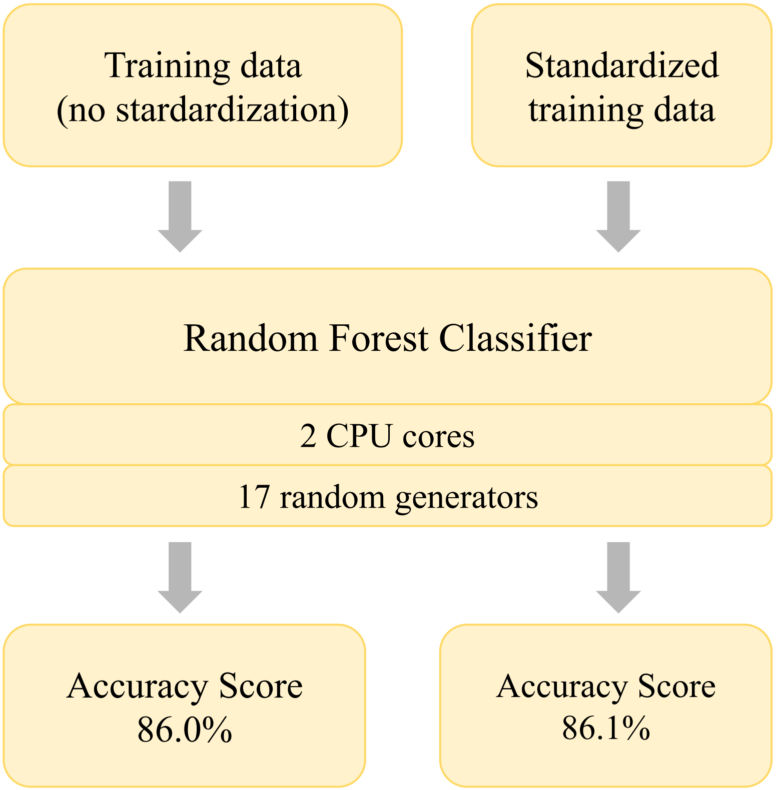


StandardScaler	Map all data to the same scale. Normalize the test set with train set's mean and variance

***Models** - Logistic Regression*



***Models** - Random Forest*



- Feature Importance Ranking

	Importance
Elevation	0.221297
Horizontal_Distance_To_Roadways	0.093678
Horizontal_Distance_To_Fire_Points	0.073004
Horizontal_Distance_To_Hydrology	0.062592
Hillshade_9am	0.052744
Vertical_Distance_To_Hydrology	0.052035
Aspect	0.050237
Hillshade_3pm	0.047294
Hillshade_Noon	0.045997
Wilderness_Area4	0.038577

***Models** - LightGBM*

What we do in LightGBM ?

	XGBoost	LightGBM
Tree growth algorithm	Level-wise good for engineering optimization but not efficient to learn model	Leaf-wise with max depth limitation get better trees with smaller computation cost, also can avoid overfitting
Split search algorithm	Pre-sorted algorithm	Histogram algorithm
memory cost	2*#feature*#data*4Bytes	#feature*#data*1Bytes (8x smaller)
Calculation of split gain	O(#data* #features)	O(#bin *#features)
Cache-line aware optimization	n/a	40% speed-up on Higgs data
Categorical feature support	n/a	8x speed-up on Expo data

- ❑ Faster training speed and higher efficiency.
- ❑ Lower memory usage.
- ❑ Better accuracy.
- ❑ Support of parallel and GPU learning.
- ❑ Capable of handling large-scale data.

Training data → LightGBM Classifier (17 random generators) → Accuracy Score 23.1%

Improvement

Tuning Paramters 87%

Models *- Feedforward Neural Network*

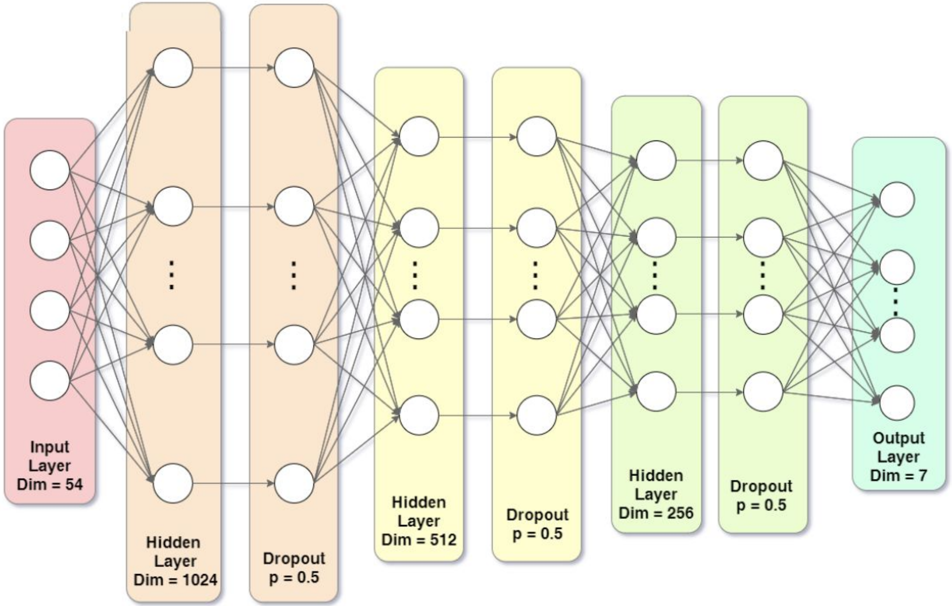
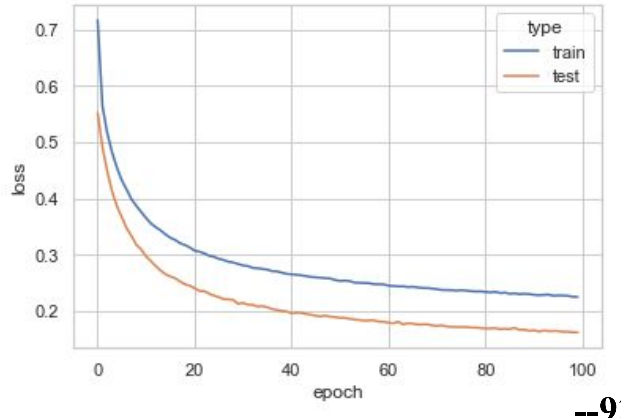
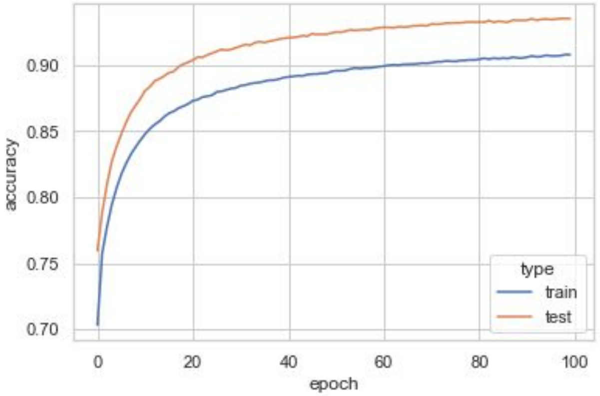


Fig. Our final FNN strcuture



--93.5%



Models *- Distributed decision tree*

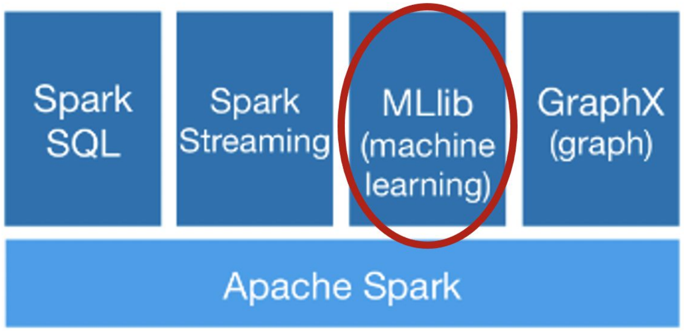


Fig. Spark strcuture

Performance

High-quality algorithms, 100x faster than MapReduce.

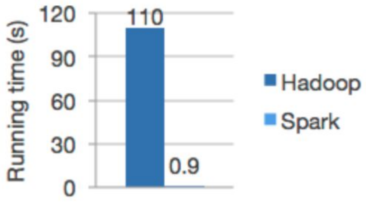
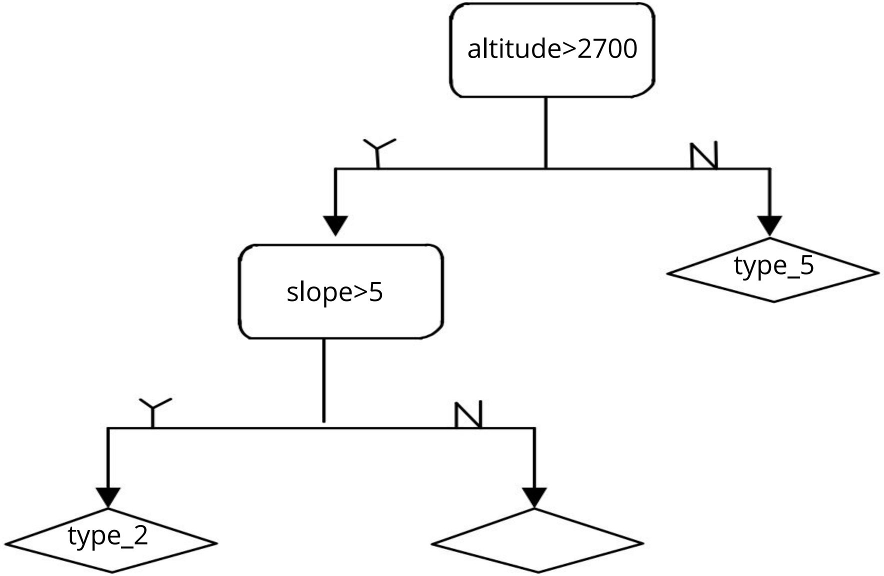


Fig. Logistic regression in Hadoop and Spark

Data is represented by RDD - Resiliennt Distributed Datasets
Algorithms are called on distributed data sets
Those algorithms which perform well on clusters

Models *- Distributed decision tree*



	Elevation	Slope	Cover_Type
Id			
1	2596	3	5
2	2590	2	5
3	2804	9	2
4	2785	18	2
5	2595	2	5

Fig. Sample of decision tree

Easy to understand intuitively

Models *- Distributed decision tree - default model*

- ❑ **LabeledPoint** - Vector containing multiple feature values & label
- ❑ **Split data**

```
val Array(trainData, cvData, testData) =
  data.randomSplit(Array(0.8, 0.1, 0.1))
trainData.cache()
cvData.cache()
testData.cache()
```

- ❑ **Default model**

```
val model = DecisionTree.trainClassifier(
  trainData, 7, Map[Int,Int](), "gini", 4, 100)
```

```
(0.6879105188005712,0.6724202102912441)
(0.7224695369618197,0.7905422222222223)
(0.6314069838676741,0.8600834492350486)
(0.3257328990228013,0.398406374501992)
(0.9411764705882353,0.016194331983805668)
(0.0,0.0)
(0.6995073891625616,0.4190850959173635)
```

covtype (2).info

```
Code Designations:

Wilderness Areas:       1 -- Rawah Wilderness Area
                        2 -- Neota Wilderness Area
                        3 -- Comanche Peak Wilderness Area
                        4 -- Cache la Poudre Wilderness Area

Soil Types:             1 to 40 : based on the USFS Ecological
                        Landtype Units (ELUs) for this study area:

  Study Code USFS ELU Code                      Description
        1         2702        Cathedral family - Rock outcrop complex, extremely stony.
        2         2703        Vanet - Ratake families complex, very stony.
        3         2704        Haploborolis - Rock outcrop complex, rubbly.
        4         2705        Ratake family - Rock outcrop complex, rubbly.
        5         2706        Vanet family - Rock outcrop complex complex, rubbly.
        6         2717        Vanet - Wetmore families - Rock outcrop complex, stony.
        7         3501        Gothic family.
        8         3502        Supervisor - Limber families complex.
        9         4201        Troutville family, very stony.
       10         4703        Bullwark - Catamount families - Rock outcrop complex,
rubbly.
       11         4704        Bullwark - Catamount families - Rock land complex, rubbly.
       12         4744        Legault family - Rock land complex, stony.
       13         4758        Catamount family - Rock land - Bullwark family complex,
rubbly.
       14         5101        Pachic Argiborolis - Aquolis complex.
       15         5151        unspecified in the USFS Soil and ELU Survey.
       16         6101        Cryaquolis - Cryoborolis complex.
       17         6102        Gateview family - Cryaquolis complex.
```

Fig. Covtype.info

Models *- Distributed decision tree - parameters tuning*

- ❑ **Impurity**

Gini formula:

$$I_G(p) = 1 - \sum_{i=1}^{N} p_i^2. \qquad (1)$$

Entropy formula:

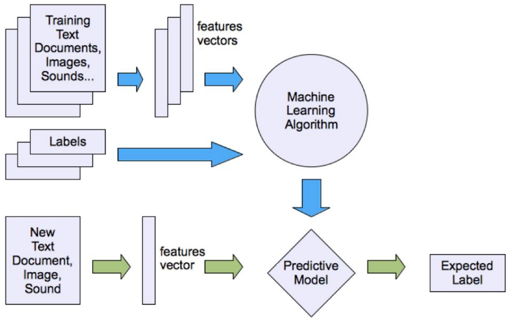
$$I_E(p) = \sum_{i=1}^{N} p_i \log\left(\frac{1}{p}\right) = -\sum_{i=1}^{N} p_i \log(p_i). \qquad (2)$$

- ❑ **maxDepth**
- ❑ **maxBins**
- ❑ **Adjusted model**

```
((entropy,20,300),0.9380098861985638)
((gini,20,300),0.9319721451536285)
((entropy,20,10),0.9273681094366382)
((gini,20,10),0.9195954644654499)
((gini,1,10),0.633916339077334)
((gini,1,300),0.6335772755123819)
((entropy,1,300),0.48759922342395684)
((entropy,1,10),0.48759922342395684)
```

✓ Test set **91.6%**

93.8% - 91.6%
Acceptable



- ❖ A LabeledPoint type RDD is accepted as input
- ❖ Hyperparameters are selected by dividing the input data into <u>training set</u>, <u>cross-validation set</u>, <u>test set</u>.

Comparison & Conclusion

- Preprocessing
  - Visualization
  - Data Cleaning

- Models
  - Logistic Regression
  - Random Forest
  - LightGBM
  - FNN
  - Distributed Decision Tree (Spark)

- Comparison
  - Accuracy
  - Effiency
    - Time cost
    - Construction cost

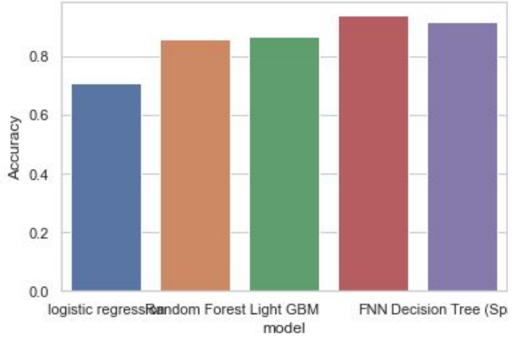


Fig.1 Accuracy Comparison between 5 models

TABLE I
FIVE MODELS' PERFORMANCE

Accuracy / Model	Default	After Tuning
Logistic Regression	70.7%	None
Random Forest	86.0%	None
LightGBM	84.7%	87.0%
Neural Network	50.0%	93.5%
Desicion Tree on Spark	70%	91.6%

None only one result.

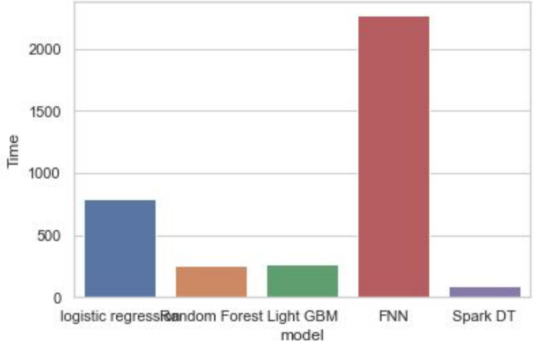


Fig.2 Time for training between 4 models

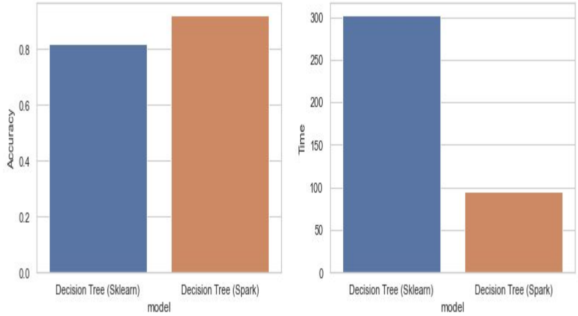


Fig.3 Accuracy (Left) and Time (Right) for training between Spark DT and Sklearn DT

Contributions

- ★ **Zhuoxuan**
  - Literature Review, Neural network model Setup/Coding, Data Visualization, Report writing, Video Presentation.
- ★ **Xinyue**
  - Literature Review, Data Preprocessing, Decision tree model Setup/Coding, Report writing, Video Presentation.
- ★ **Shuyu**
  - Literature Review, Sklearn models Setup/Coding, Report writing, Video Presentation.

Github

Click green title ➔

README.md

MSBD5012-Forest-type-prediction-exploration

A group project of HKUST BDT 20FALL 5012 course.

- Project details are in the following content.
- http://archive.ics.uci.edu/ml/datasets/Forest+type+mapping

Task Description:

In this project, we predict the forest cover type (the predominant kind of tree cover) from strictly cartographic variables (opposed to remotely sensed data). To finish the classification task, we first analyze the dataset to equip the future data pre-processing and application. Then we select multiple machine learning algorithms including Logistic Regresion, Random Forests, LightGBM, Decision Tree on Spark and FNN from various machine learning packages such as Sklearn, Keras and Spark MLlib , to compare their performance. In this report, the process of data analysis, data cleaning, data normalization and hyperparameter tuning will be described to show how they affect the final classification accuracy.

Author	Xinyue Zhang	Zhuoxuan Li	Shuyu Qiao
ID	20750194	20740917	20747563
Email	xzhangfa@connect.ust.hk	zlify@connect.ust.hk	sqiaoac@connect.ust.hk

---

Content

- Data
  - covtype.info
  - train.csv
  - test.csv
- Data explore : EDA.ipynb
- Final code : FNN.ipynb
- Final code : Decision tree on Spark.ipynb
- Final code : LR-RF-LightGBM.ipynb
- Final code : Decesion tree.ipynb
- Final repo : 5012_Final-report.pdf

Reference

[1] “Covertype Data Set”,http://archive.ics.uci.edu/ml/datasets/Covertype/, 1998.

[2] J.A.Blackard, “Description of The Forest CoverType Dataset”, http://ftp.ics.uci.edu/pub/machinelearningdatabases/covtype/covtype.info, 2001.

[3] S. A. Eschrich, “Learning From Less: A Distributed Method for Machine Learning ”, Dissertation, U. of South Florida, 2003.

[4] A. Lazarevic and Z. Obradovic, “The distributed boosting algorithm”, Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 311-316, 2001.

[5] J. R. Quinlan, “C4.5: Programsfor Machine Learning”, Morgan Kaufmann Publishers, Inc., pp. 35-43, 1993.

[6] M. Riedmiller and H. Braun, “A direct adaptive method for faster backpropagation learning: The RPROP algorithm”, Proceedings of IEEE Int’l Conference on Neural Networks, pp. 586-591, 1993.

[7] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. Annals of statistics, pages 1189–1232, 2001.

[8] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". Advances in Neural Information Processing Systems 30 (NIPS 2017), pp. 3149-3157.

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Thank you for your listening!

Li, Zhuoyuan (zlify@connect.ust.hk)
Zhang, Xinyue (xzhangfa@connect.ust.hk)
Qiao, Shuyu (sqiaoac@connect.ust.hk)