Clustered Decision Trees

Justin Lee and Abhinav Palikala

Introduction

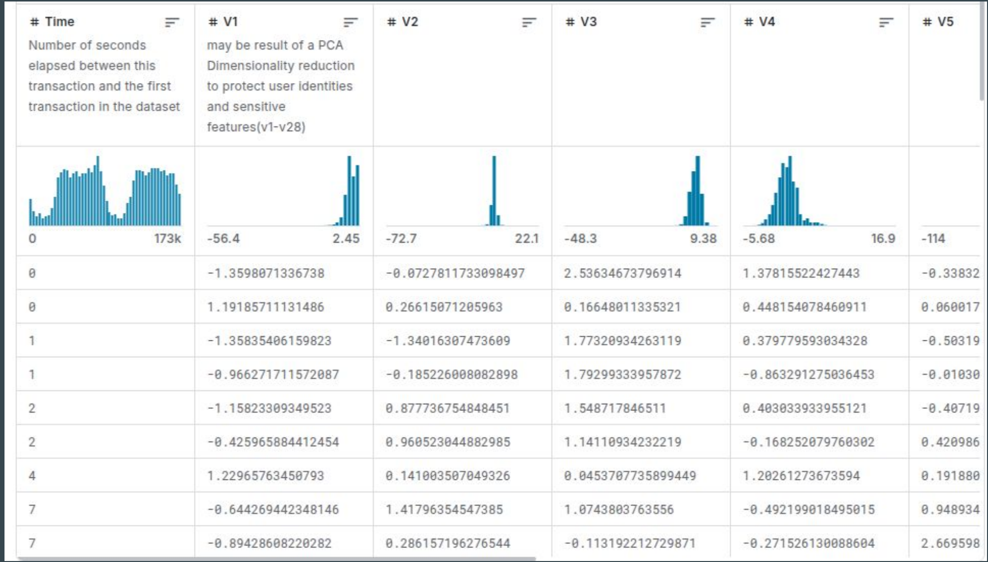
- **Decision Trees:** Classification model that's used in several fields like Healthcare and Business.
- **Feature Selection:** Traditional splitting heuristics like information gain, gain ratio, and Gini impurity act greedily, lacking the ability to optimize for future splits.
- **Research Focus:** We address heuristic limitations with a novel algorithm, comparing its performance against existing methods across multiple datasets.
- **Novel Approach:** The problem was tackled by combining K-means clustering to create "intuitive" groupings while also calculating the traditional entropy.

Related Work

- The most common research on decision tree splitting is to alter the ‘purity’ or entropy measure
    - Traditional InfoGain
    - Gini impurity
    - Power-mean impurity
    - Generalized power series
- None of these address our concerns for decision tree splitting
- More “novel” methods combine several attributes together as part of a “multi-dimensional” selection method.
- In a similar fashion, some splitting methods combine several different algorithms. This is where we decided to work.

Dataset and Features

- Credit Fraud Dataset
  - Heavily unbalanced dataset
  - 492 frauds out of 284,8407 transactions (0.172% frauds)
  - 30 feature attributes
    - Result of PCA on original dataset
- Preprocessing
  - Choose a representative sample
  - Equal width 3 bins
  - Saved discrete and continuous data
  - 80-20 split



# Time	# V1	# V2	# V3	# V4	# V5
Number of seconds elapsed between this transaction and the first transaction in the dataset	may be result of a PCA Dimensionality reduction to protect user identities and sensitive features(v1-v28)				
0 173k	-56.4 2.45	-72.7 22.1	-48.3 9.38	-5.68 16.9	-114
0	-1.3598071336738	-0.0727811733098497	2.53634673796914	1.37815522427443	-0.33832
0	1.19185711131486	0.26615071205963	0.16648011335321	0.448154078460911	0.060017
1	-1.35835406159823	-1.34016307473609	1.77320934263119	0.379779593034328	-0.50319
1	-0.966271711572087	-0.185226008082898	1.79299333957872	-0.863291275036453	-0.01030
2	-1.15823309349523	0.877736754848451	1.548717846511	0.403033933955121	-0.40719
2	-0.425965884412454	0.960523044882985	1.14110934232219	-0.168252079760302	0.420986
4	1.22965763450793	0.141003507049326	0.0453707735899449	1.20261273673594	0.191880
7	-0.644269442348146	1.41796354547385	1.0743803763556	-0.492199018495015	0.948934
7	-0.89428608220282	0.286157196276544	-0.113192212729871	-0.271526130088604	2.669598

Methods behind Clustered Decision Trees

Normal Decision Tree Heuristic:

$$Info = -\sum_{D_i}^{D}\frac{|D_i|}{|D|}\sum_{C_{i,j}}^{C_i}\frac{|C_{i,j}|}{|C_i|}log_2(\frac{|C_{i,j}|}{|C_i|})$$

Clustered Decision Tree Heuristic:

$$ClusteredInfo = -\sum_{D_i}^{D}\frac{|D_i|}{|D|}\sum_{K_{i,j}}^{K_i}\frac{|K_{i,j}|}{|K_i|}\sum_{C_{i,j,k}}^{C_{i,j}}\frac{|C_{i,j,k}|}{|C_{i,j}|}log_2(\frac{|C_{i,j,k}|}{|C_{i,j}|})$$

D: Node after split

K: K-Means cluster

C: Class label

Motivation

Consider the two following datasets:

Feature	Class
1	1
0	1
1	0
0	0

Feature	Class
1	1
1	1
0	0
0	0

Decision trees are greedy algorithms and only consider the instantaneous splitting power. These two datasets are thus treated the same.

Results (K = 2)

$$Sensitivity = \frac{TP}{TP + FN}$$

```
##### Clustered Decision Tree Train #####
Accuracy: 99.99%
Confusion Matrix:
        Actual (1) Actual (0)
Pred (1) | 024       000
Pred (0) | 001       7975

##### Clustered Decision Tree Test #####
Accuracy: 99.85%
Confusion Matrix:
        Actual (1) Actual (0)
Pred (1) | 010       000
Pred (0) | 003       1987
```

```
##### Regular Decision Tree Train #####
Accuracy: 99.99%
Confusion Matrix:
        Actual (1) Actual (0)
Pred (1) | 024       000
Pred (0) | 001       7975

##### Regular Decision Tree Test #####
Accuracy: 99.25%
Confusion Matrix:
        Actual (1) Actual (0)
Pred (1) | 005       007
Pred (0) | 008       1980
```

Sensitivity = 10 / (10 + 3) = 0.769

Sensitivity = 5 / (5 + 8) = 0.385

Conclusion

- Clustered Decision Trees do perform better than normal decision trees on select datasets
- They are capable of capturing more complex interactions along with addressing unbalanced class labels
- Significantly more computationally expensive than traditional decision trees

Future

- Try to make it more efficient (cache clusters)
- Weight different features differently
- Compare the depths of decision trees to find if it makes cheaper (shallower) trees
- Analyze the performance of CDTs on different data distributions

References

Bahnsen, A. C., Aouada, D., & Ottersten, B. (2015). Example-dependent cost-sensitive decision trees. arXiv preprint arXiv:1509.07266.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (2012). Classification and regression trees. Microsoft Research. Retrieved from https://www.microsoft.com/en-us/research/wp-content/uploads/2012/06/1206.4620.pdf

Dal Pozzolo, A., Caelen, O., Johnson, R. A., & Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. arXiv preprint arXiv:1511.08136.

Dugas-Phocion, G., & Bengio, S. (2020). Learning optimal decision trees using reinforcement learning heuristics. arXiv preprint arXiv:2010.08633.

Kaggle. (n.d.). Credit card fraud detection dataset. Retrieved from https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

Keylabs AI. (n.d.). Decision trees: How they work and practical examples. Retrieved from https://keylabs.ai/blog/decision-trees-how-they-work-and-practical-examples/

Lomax, S., & Vadera, S. (2013). A survey of cost-sensitive decision tree induction algorithms. Knowledge Engineering Review, 28(2), 159–188. Retrieved from https://pmc.ncbi.nlm.nih.gov/articles/PMC2701298/

Vation Ventures. (n.d.). Decision trees: Definition, explanation, and use cases. Retrieved from https://www.vationventures.com/glossary/decision-trees-definition-explanation-and-use-cases

Zhang, Y., & Wu, J. (2018). A survey on decision tree learning for heterogeneous data. arXiv preprint arXiv:1801.08310.