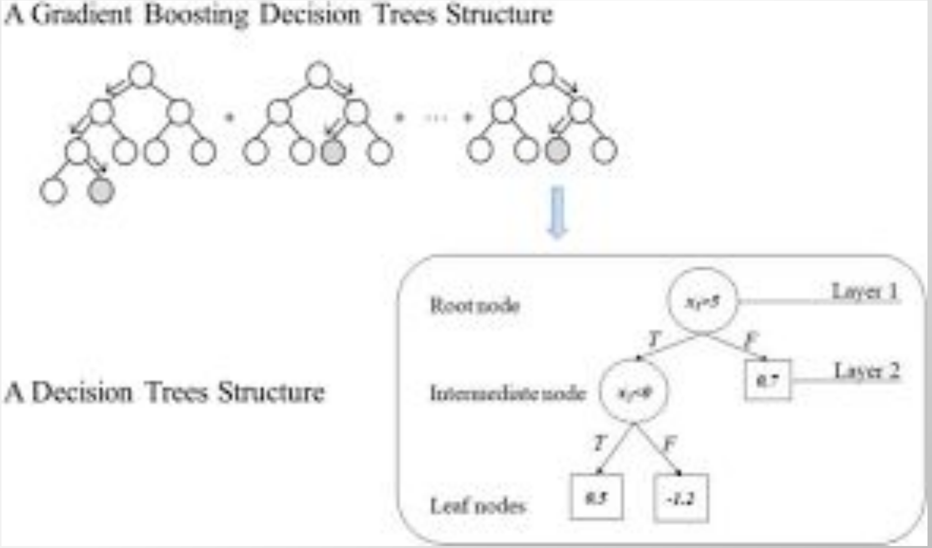
MAXGBoost: A Fast Novel Heuristic Approach to Adaptive Learning Rates in Gradient Boosted Decision Trees

By: Anirudh Chintaluri, Radin Rezanezhad

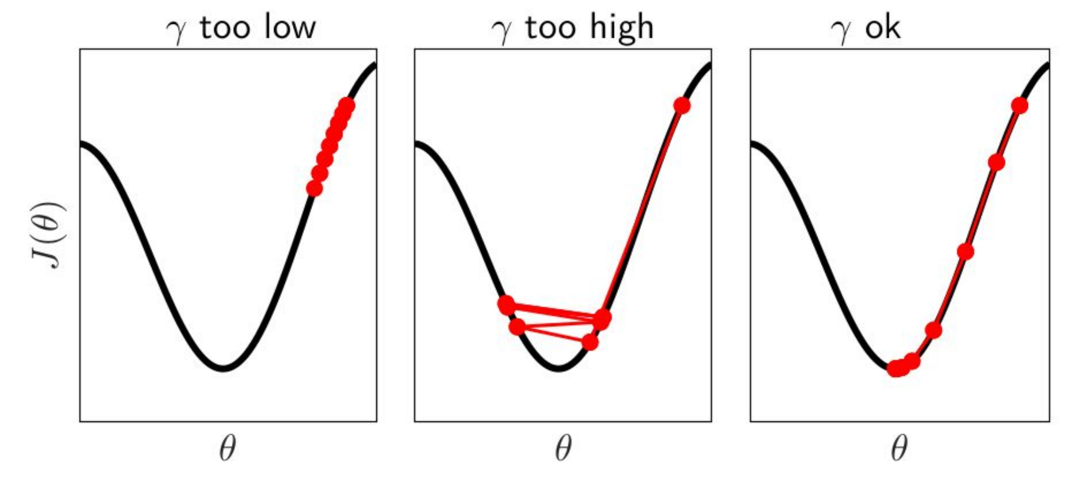
Fundamentals

- GBDTs initialize with a decision tree that's evaluated using a loss function
- The model minimizes loss by moving in the direction of the negative gradient
- This process is similar to neural networks' backpropagation, but creates an ensemble decision tree model



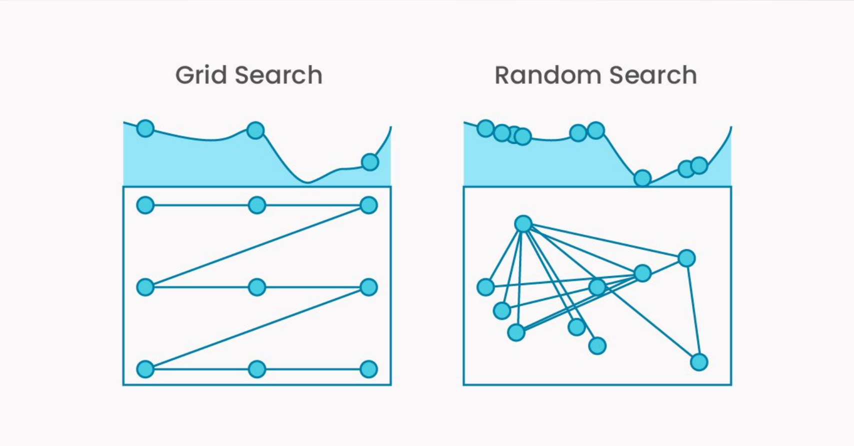
Learning Rate Mechanics

- The algorithm adds new decision trees scaled by a learning rate (η)
- Learning rate determines how far down the loss function the tree will go
- Low η ensures convergence but requires more iterations and computational cost
- High η speeds up convergence but risks overshooting the optimal solution



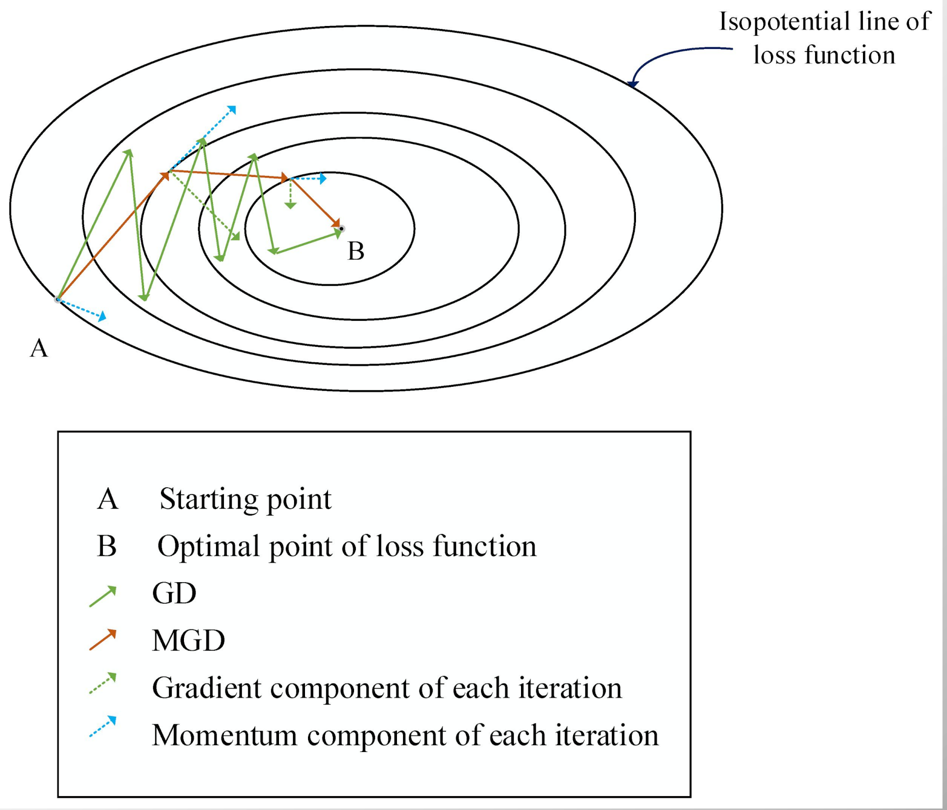
Learning Rate Optimization

- Optimal learning rate depends on both dataset and loss function
- Tuning requires iterative experimentation
- Adaptive strategies help reduce need for extensive hyperparameter searches
- Particularly important for datasets with high sparsity



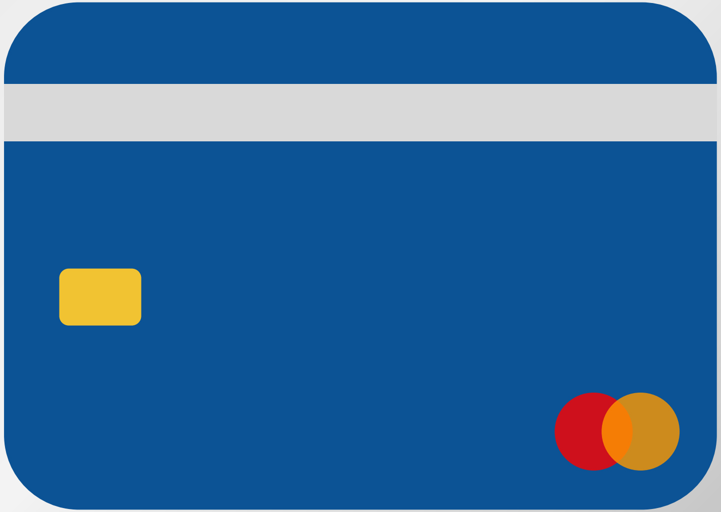
Introduction to MAXGBoost

- Momentum Approximation XGBoost dynamically adjusts learning rate based on loss momentum
- Enables faster convergence
- Reduces need for manual optimization while maintaining high detection accuracy



The Dataset

- Credit Card Fraud Detection
- 284,807 total instances
- 31 total attributes
- Notable Features:
  - Time
  - Amount
  - V1-V38 → Result of PCA
- 492 instances marked as fraud
  - < 0.2% of data
- Dataset already preprocessed



The Algorithm - Traditional Neural Network

$$w_{t+1} = w_t - \eta \nabla L$$

The Algorithm - Neural Network with Momentum

- v = velocity
- β = how influential past velocities are on the new velocity calculated
- Converges faster than with a constant learning rate

$$v_{t+1} = \beta v_t + (1 - \beta)\nabla L$$

$$w_{t+1} = w_t - \eta v_{t+1}$$

The Algorithm - MAXGBoost

- Combines GBDT and momentum-based updates
- Approximates ∇ L
- Using momentum to update the **learning rate**
  - Loss increase = η increase
  - Loss decrease = η decrease

$$v_{t+1} = \beta v_t + (1 - \beta)(L_{t+1} - L_t)$$

$$\eta_{t+1} = \eta_t(1 + v_{t+1})$$

Experiments - Dataset

Splits

- 68%: Training
- 12%: Validation
- 20%: Testing

Analysis

- 5-fold Cross-validation
- Results based on best fold per model

Experiments - Models

Decision Tree	Single Decision Tree
Random Forest	Ensemble Voting Decision Tree Model
Constant η XGBoost	XGBoost with a constant learning rate
Exponential Decay XGBoost	XGBoost with a learning rate multiplied by constant every iteration
MAXGBoost	XGBoost with Momentum Approximation

Experiments - Hyperparameters

Decision Tree	**Max Depth** = 4
Random Forest	**Estimators** = 5
Constant η XGBoost	**Estimators** = 423, **η** = 0.08
Exponential Decay XGBoost	**Estimators** = 423, **x** = 0.9
MAXGBoost	**Estimators** = 423, **Initial η** = 0.89, **β** = 0.99

Results

Model	Accuracy	Precision	Recall	AUC
Decision Tree	0.99946	0.86813	0.80612	0.90296
Random Forest	0.99951	0.77000	0.93902	0.96931
Constant η	0.99977	0.88636	0.96296	0.98139
Exponential decay	0.99979	0.88764	**0.97531**	**0.98757**
MAXGBoost	**0.99980**	**0.93827**	0.92683	0.96337

Discussion - MAXGBoost Strengths

- **Best Precision** in Fraud Detection
  - Exceptional performance in minimizing false positives
  - Effective for high-stakes financial environments
    - Reduces customer friction from false alerts
    - Minimizes operational costs from investigation overhead

Discussion - MAXGBoost Limitations

- Relatively low AUC for Fraud Detection
  - **Rapid** learning rate **decrease** with small loss improvements
  - **Limited exploration of feature space**
  - Over-emphasis on strongly discriminative features
  - Underweighting of subtle fraud patterns

Conclusion

- Decision Tree-based models effective in classifying positive class of fraud **in < 0.2% of data**
  - **Increases the ability for law enforcement** to correctly catch frauds
- MAXGBoost best model comparing accuracy and precision
  - MAXGBoost most effective in **preventing false accusations of fraud** and **increasing customer satisfaction**
- Exponential decay XGBoost most effective in **catching fraudulent cases**, compromising false accusations

Future Work

- **Hybrid Model Development**
  - Combine strengths of EGB and MAXGBoost
  - Create integrated momentum-decay approach
- Feature-specific momentum updates
- Class-aware momentum adjustments

Thanks!

Questions?

References

Andrej Baranovskij. (2019, March 12). *Selecting Optimal Parameters for XGBoost Model Training*. Medium; Towards Data Science. https://medium.com/towards-data-science/selecting-optimal-parameters-for-xgboost-model-training-c7cd9ed5e45e

Beja-Battais, P. (2023, October 6). *Overview of AdaBoost: Reconciling its views to better understand its dynamics*. ArXiv.org. https://doi.org/10.48550/arXiv.2310.18323

Brownlee, J. (2016, September 15). *Tune Learning Rate for Gradient Boosting with XGBoost in Python*. Machine Learning Mastery. https://machinelearningmastery.com/tune-learning-rate-for-gradient-boosting-with-xgboost-in-python/

Chen, T., & Guestrin, C. (2016). XGBoost: a Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, 785–794. https://doi.org/10.1145/2939672.2939785

Credit Card Fraud Detection. (n.d.). Kaggle. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). *LightGBM: A Highly Efficient Gradient Boosting Decision Tree*. Neural Information Processing Systems; Curran Associates, Inc. https://papers.nips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html

Learning rate. (n.d.). Wikipedia. https://en.wikipedia.org/wiki/Learning_rate

Mohan, A. (2021, April 6). *IEEE-CIS Fraud Detection - Top 5% Solution - Towards Data Science*. Medium; Towards Data Science. https://medium.com/towards-data-science/ieee-cis-fraud-detection-top-5-solution-5488fc66e95f

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Müller, A., Nothman, J., Louppe, G., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2018). Scikit-learn: Machine Learning in Python. *ArXiv:1201.0490 [Cs]*. https://arxiv.org/abs/1201.0490

Vitaly Bushaev. (2017, December 4). *Stochastic Gradient Descent with momentum - Towards Data Science*. Medium; Towards Data Science. https://medium.com/towards-data-science/stochastic-gradient-descent-with-momentum-a84097641a5d

Wang, C., Wang, Z., Ouyang, Y., & Soleimani, B. H. (2024, May 27). *Adaptive Learning Rates for Gradient Boosting Machines*. Proceedings of the Canadian Conference on Artificial Intelligence. https://caiac.pubpub.org/pub/py65wd3c/release/1