Can a Decision Tree predict the markets better than linear models?

After testing several linear regressions (OLS, Ridge, Lasso), I wanted to see if a nonlinear model like a Decision Tree Regressor could capture more structure in daily S&P 500 returns.

Methodology

Same dataset and process as before:
Features: lagged returns (Lag_1, Lag_2, Lag_5), historical volatility, RSI (14 days), and log-volume.
Pipeline: preprocessing + scaling + TimeSeriesSplit cross-validation.
Hyperparameter tuning: GridSearchCV with depth, splits, and leaf size.

Results

Baseline: $R^2 \approx 12\%$, RMSE ≈ 1.1%
Optimized: $R^2 \approx 16\%$, RMSE ≈ 1.0%

For comparison, the best regression reached $R^2 \approx 16\%$. A similar level of performance, despite the tree's non-linear flexibility.

Interpretation

The optimized Decision Tree slightly improves over the baseline, reducing error variance and producing a more balanced model between bias and variance.
Its test R^2 of 16%, like Ridge regression, confirms that moving from linear to non-linear models brings only marginal gains at this stage.

The feature importance chart highlights historical volatility and RSI as dominant drivers, as seen in the regression models, these indicators remain the most relevant short-term signals, though their impact remains modest.
Lagged returns again show weak predictive power, consistent with the limited autocorrelation of market returns.

The "Actual vs Predicted" plot shows a horizontal band of points : the model often predicts values near the mean, a sign of underfitting typical for shallow trees.
This behavior reflects the tree's stepwise structure : predictions are constant within each leaf, making it difficult to capture the fine-grained variations in returns.

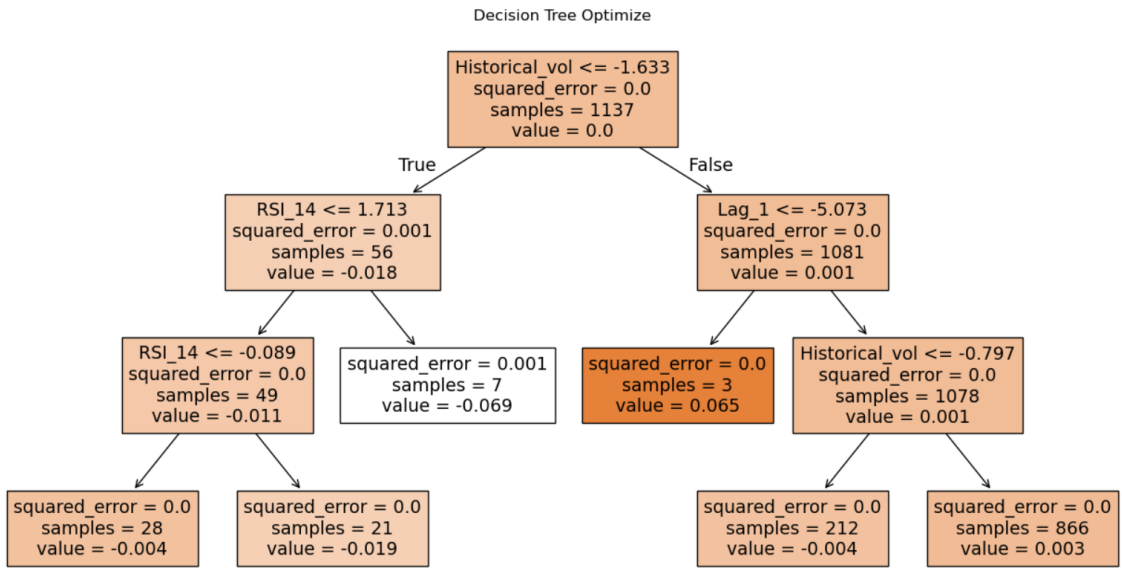
Finally, the residuals are centered around zero, as in the regression models, showing no systematic bias.
However, their wide dispersion highlights the persistent noise in financial data, confirming that a single tree while more flexible than linear models still cannot grasp the market's chaotic nature.

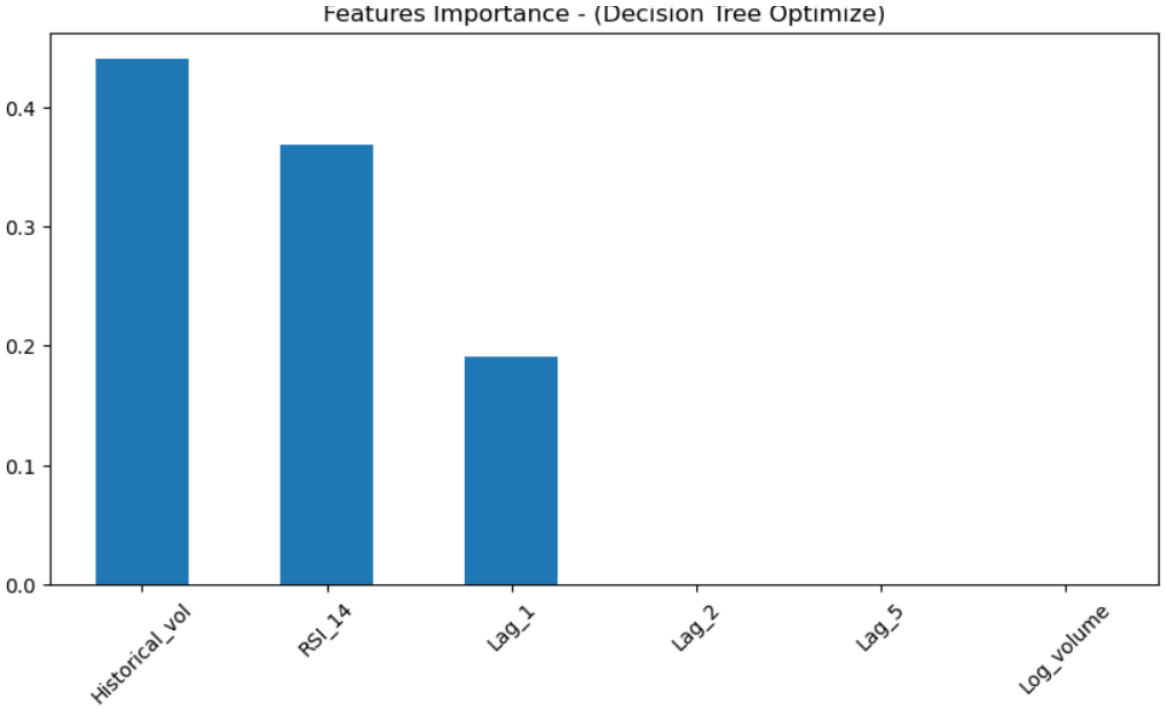
Conclusion

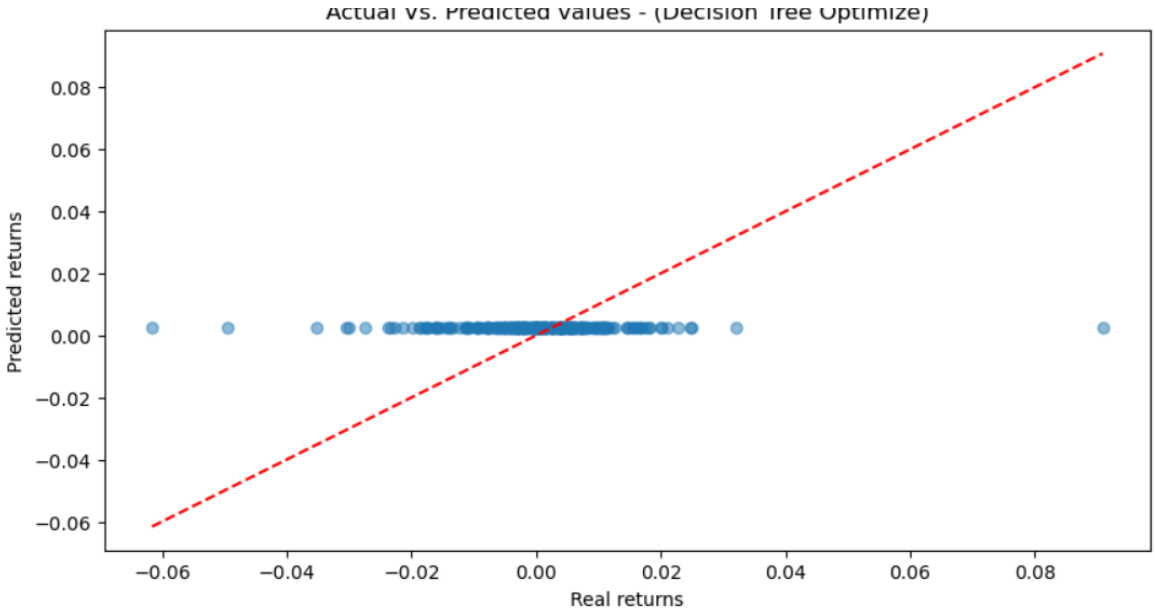
Decision Trees introduce non-linearity and interpretability but remain limited in scope.
To go beyond this, we'll need to aggregate multiple trees : ensemble methods like Random Forests, Bagging, and Extra Trees in order to reduce variance and extract more consistent market signals.

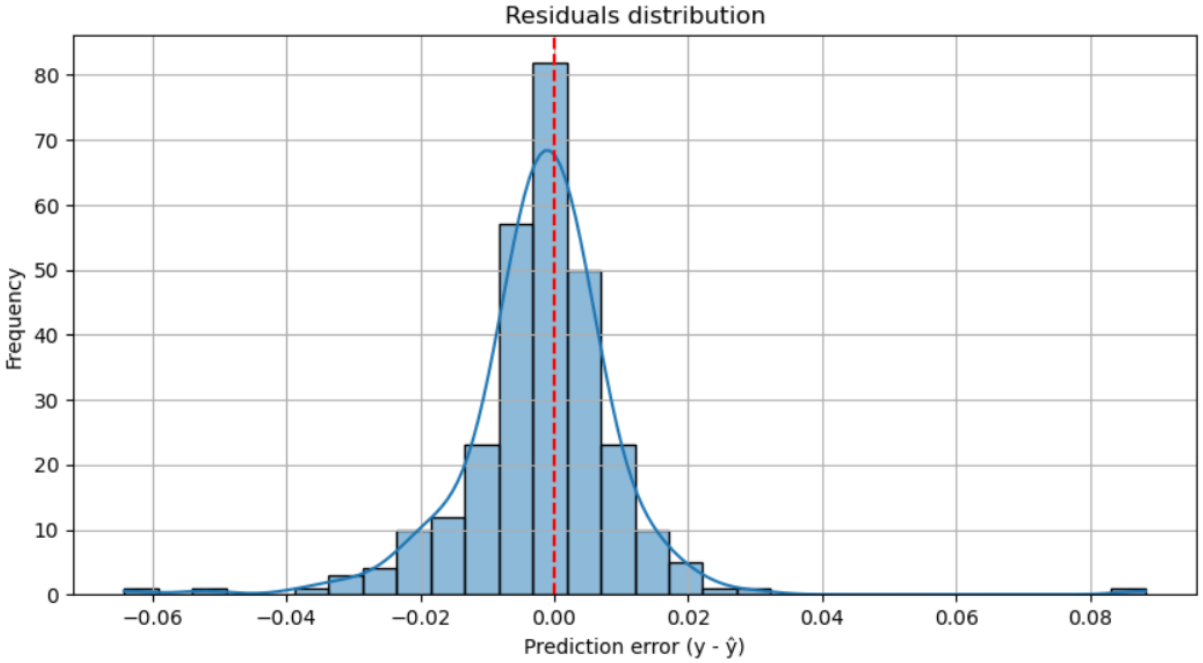
📊 Visuals: tree structure, feature importance, actual vs predicted, residuals
💻 Full notebook on GitHub → [your link]

#MachineLearning #QuantitativeFinance #DataScience #Python #DecisionTree #ScikitLearn #AlgoTrading




Features Importance - (Decision Tree Optimize)
0.4
0.3
0.2
0.1
0.0
Historical_vol
RSI_14
Lag_1
Lag_2
Lag_5
Log_volume


Actual vs. Predicted Values - (Decision Tree Optimize)
Predicted returns
Real returns
0.08
0.06
0.04
0.02
0.00
−0.02
−0.04
−0.06
−0.06
−0.04
−0.02
0.00
0.02
0.04
0.06
0.08


Residuals distribution
Frequency
Prediction error (y - ŷ)
0
10
20
30
40
50
60
70
80
−0.06
−0.04
−0.02
0.00
0.02
0.04
0.06
0.08