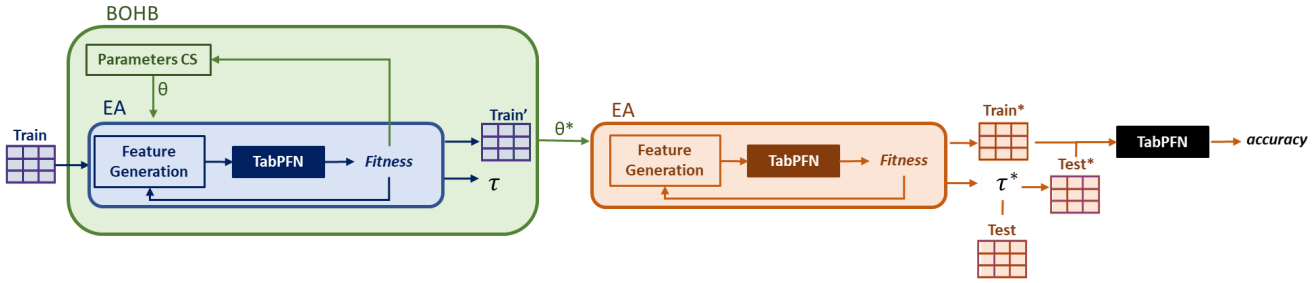
Summary

Introduction

Research Question: Can automated feature engineering using evolutionary algorithm increase the predictive accuracy of TabPFN model for non-categorical tabular data?

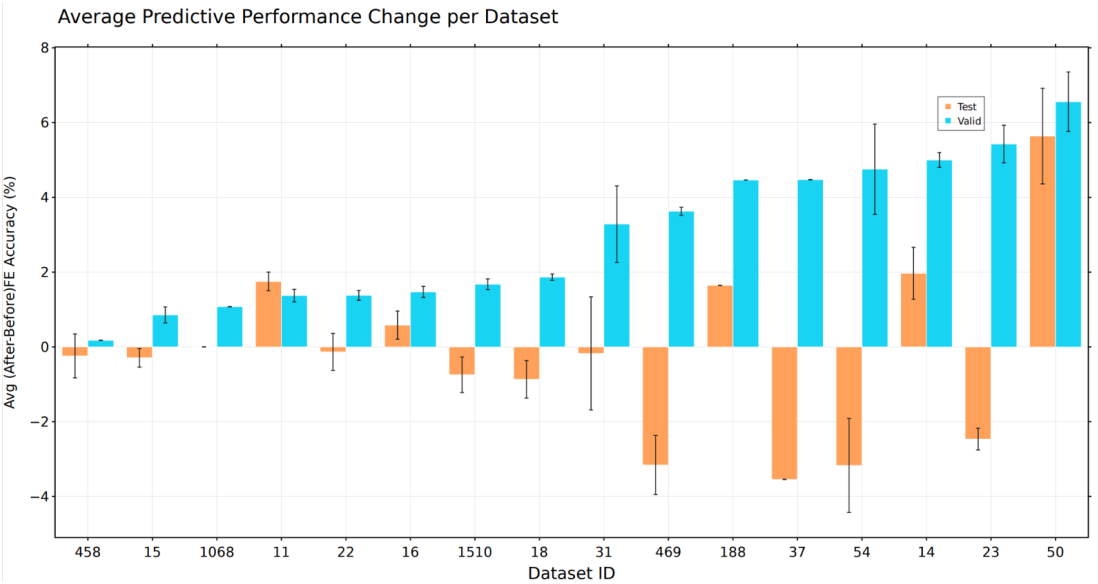
Methodology



- Supports 17 Mutation operations and 3 Recombination operations
- Analysed on 16 different non-categorical datasets, number of samples are less than or equal to 2000 and with number of features less than 100
- Average validation score of TabPFN from K-fold, where k=5, on training subset is the fitness score of a member
- BOHB used to find optimal hyperparameter for EA based on the fitness score
- EA runs 3 times with best hyperparameter found, θ*.
- Accuracy of TabPFN on the new set of features found is the evaluation metric.

Results

- The difference in validation (valid) and test score before and after FE of each dataset is computed



- Validation score increases with FE application for all datasets

- Three trends observed for difference in test scores:
    - For 5 datasets, test score can increase after FE application (e.g., dataset id 50)
    - For 7 datasets, test score can decrease after FE application (e.g., dataset id 469)
    - For 4 datasets, test score can either increase or decrease with a high variance (e.g. dataset id 31)

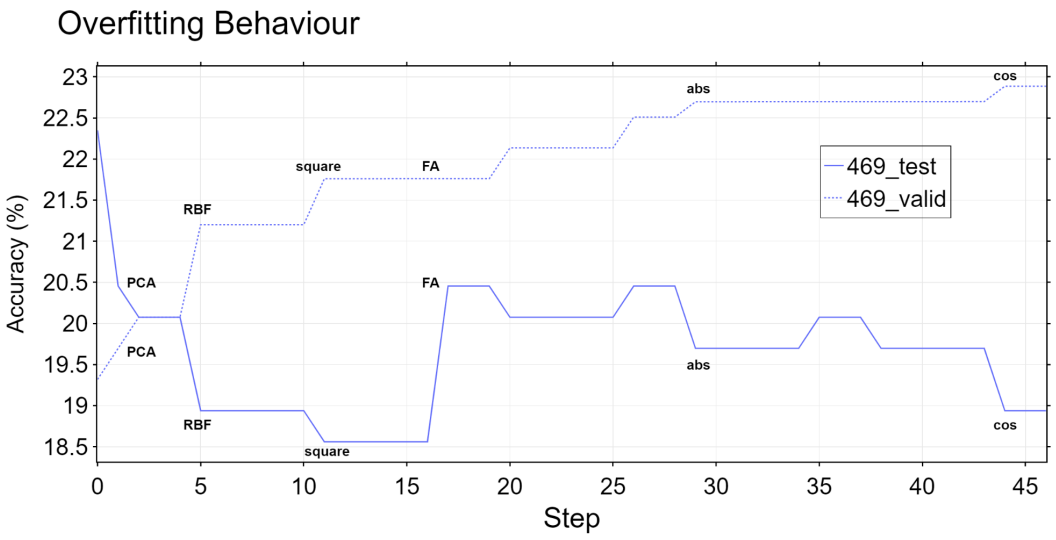
Discussion

Reasons for decrease in test score

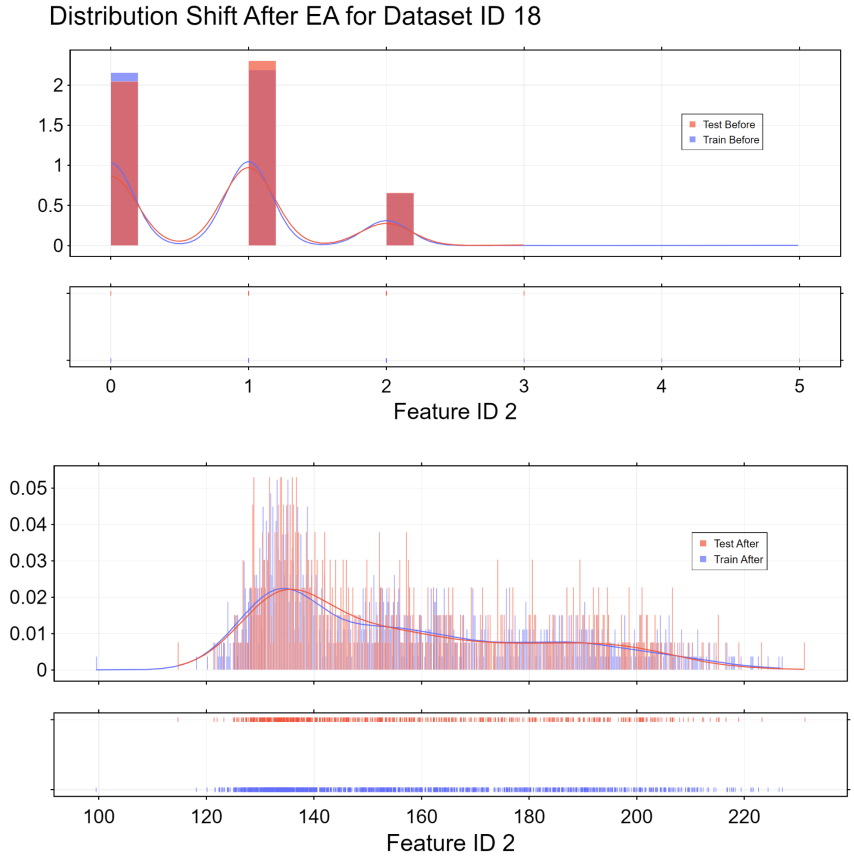
Overfitting behaviour observed (e.g. dataset id 469,18, 54) where validation score increases but test score decreases. Two observed reason:

- Drastic increase in feature space, e.g. dataset id 469 went from 50 features to 100 features after applying RBF Sampler operator
- Drastic increase in feature value, e.g. dataset 18 feature id 2 range value increased from [0,5] to [100,250] after applying Feature Agglomerative operation. And dataset 1510 feature id 23 after applying sine range value decreased from [450,2500] to [-1,1]. The prior feature range is due to Feature Agglomerative operation that resulted in an increase in validation score but no effect on test score.

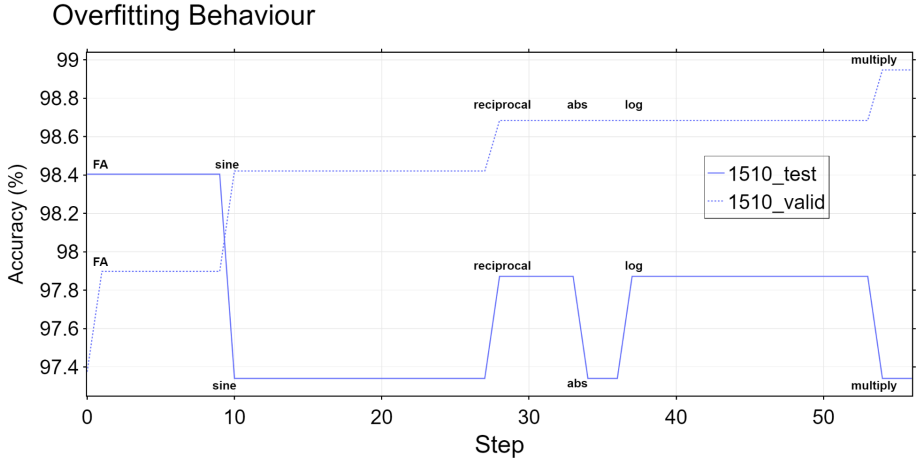
Operation trajectory of dataset 469:



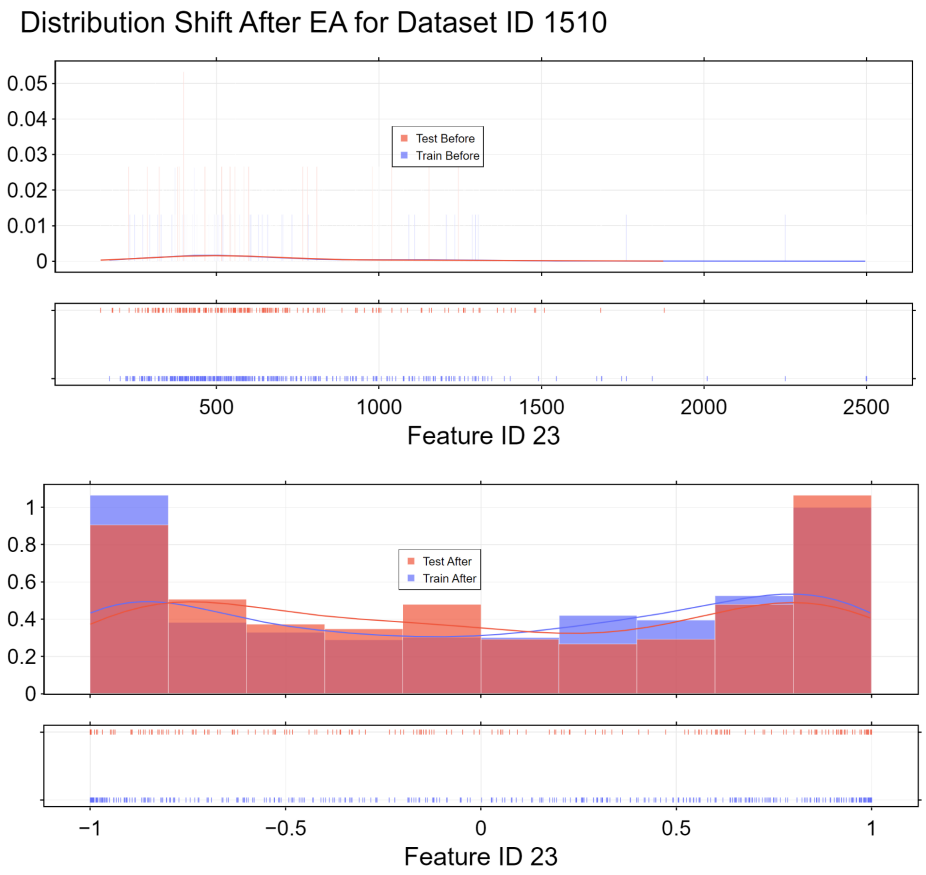
Feature Distribution and range for dataset 18 after Feature Agglomerative Operation:



Operation Trajectory of dataset 1510:



Feature Distribution and range for dataset 1510 after Sine operation:



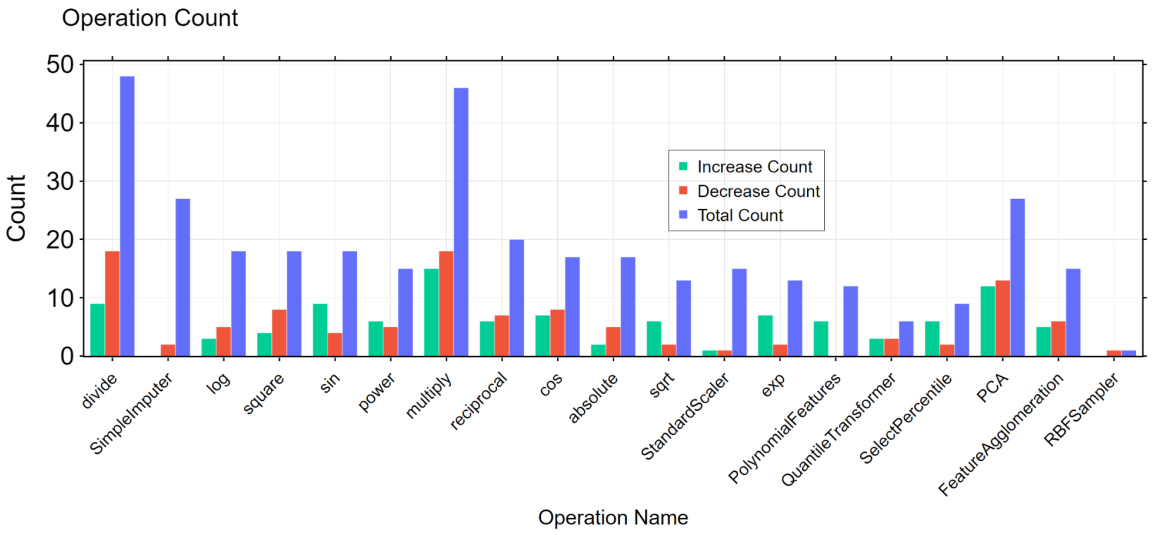
Reasons for high variance in test score

Two reasons:

- Test performance already high, for example dataset 458 already had a test score of 99.6 before EA
- Very different trajectories found when running EA 3 times, for example dataset 22. A suggested way to decrease variance is to run EA for more times.

Extras:

- No clear pattern why some datasets have better performance than others when it comes to number of features.
- Divide operation seems to mostly decrease the performance or have no effect on it. Polynomial Features always increased or had no effect but never worsened the performance. Feature Agglomerative which causes overfitting seems to almost increase performance as much as decrease it.
- Runtime of algorithm is dependent on the hyperparameters found by BO: . It can range from 30 mins per run to 1 day per run with the optimal configuration found. And BO is also dependent on the sampled hyperparameters.but the maximum runtime per sample is set to 3 hours..



Conclusion

- 31% of the datasets showed improvement
- 44% of the datasets showed disimprovement due to overfitting. This overfitting can be reduced by avoiding members with drastic change in number of features or in the feature range.
- 25% of the datasets showed a chance for improvement with some solutions found by the algorithm but also a disimprovement.. Running EA more number of times can overcome this variance but with trading computation time,
- The divide operation and Simple Imputer can be removed as it only worsens performance or doesn't affect it. Sine works better than the Cosine function. Polynomial Features is best recombination operator.
- Overall, Automated FE can increase performance of TabPFNs but we need to pay attention to drastic changes as it can result to overfitting.

Future Work

- Apply BO on operation parameters instead of default parameters
- Include categorical variables by one.hot encode them
- Design a fitness function that penalises drastic changes in number of features and feature ranges and wether mitigating such drastic changes can help improve performance
- Identify common operation sequences to reduce the search space and improve efficiency
- In some scenarios validation score plateaus. Early stopping can then be applied.