Performance Variability of Machine Learning Models using Limited Data for Collusion Detection: A Case Study of the Brazilian Car Wash Operation

Everton Schneider dos Santos[1], **Matheus Machado dos Santos**[1], **Márcio Castro**[1], **Jonata Tyska Carvalho**[1]

[1]Postgraduate Program in Computer Science
Federal University of Santa Catarina (UFSC)
P.O. Box 476 – 88.040-370 – Florianópolis – SC – Brazil

{e.schneider.s,matheus.m.santos}@posgrad.ufsc.br,
{marcio.castro,jonata.tyska}@ufsc.br

Abstract. *Fraudulent companies form illegal agreements, like collusion and cartels, to circumvent the impartiality and competitiveness of the public procurement auctions. These types of fraud can cause significant financial losses and erode trust in the public sector. Therefore, building reliable methods for early detection of frauds is a priority for public organizations. This study uses an enriched version of the "Operation Car Wash" dataset to evaluate the collusion detection capabilities of different machine learning algorithms. Using cross-validation techniques, the methodology proposed in our work was able to improve the collusion detection rate of the learning models used in this work, outperforming the results of other works found in the literature.*

1. Introduction

Public procurement is a common method through which governments allocate funds to obtain necessary goods and services. This process employs a systematic competitive bidding approach [Curtis and Maines 1973], which is fundamental to transparent governance and procurement strategies. This competition is beneficial, as it enables governments and taxpayers to obtain better value for their money, enhancing the procurement process's efficiency and fairness [García Rodríguez et al. 2022].

However, this process is not without its challenges. One significant issue is collusion, also known as bid-rigging, where competing entities engage in illicit agreements to increase their profits. This behavior often leads to non-competitive price increases, typically coordinated by cartels, undermining the procurement process's integrity [Porter and Zona 1993, García Rodríguez et al. 2022]. It is estimated that global economic output loses between 2% and 5% annually due to corruption [Velasco et al. 2021].

Recent investigations in Brazil have unveiled numerous instances of corruption across government sectors and corporations. The "Operation Car Wash" [Signor et al. 2020b] was a anti-corruption investigation, responsible for uncovering a group of companies that colluded in several infrastructure projects from Petrobras.

Detecting collusion, especially in capital works procurement, which are among the most costly acquired items, remains a challenge for public institutions. While criminal investigations are frequently initiated to address such issues, proving a collusive action is complex and challenging [García Rodríguez et al. 2022].

A reliable mechanism for early detection of fraud in public procurement auctions could significantly aid authorities in mitigating the adverse effects of these activities. Machine Learning (ML) algorithms, by analyzing data patterns, can play a crucial role in detecting anti-competitive behaviors [Velasco et al. 2021], although their success depends on the availability of comprehensive and reliable historical data [Signor et al. 2020a].

However, historical data is not publicly available in most cases. Moreover, much of the information about procurement processes is confidential, resulting in datasets with limited features. Although it is sometimes possible to access a large amount of data through government transparency programs, obtaining labels (i.e., information about whether a tender is collusive or non-collusive) is challenging. This is because each tender must be investigated by an auditor. When dealing with small datasets for training supervised ML algorithms, issues such as lack of feature representation, class imbalance, and poor model evaluation can arise.

This study builds upon a previous research [García Rodríguez et al. 2022], which evaluated eleven ML algorithms for detecting collusion in public tenders across various countries. Our research delves into the challenges associated with training ML algorithms with limited dataset size. Overall, the key contributions of this paper include:

1. An enriched version of the "Operation Car Wash" dataset, adding to the existing data new features related to the companies participating in the tenders;
2. An evaluation on how features related to internal aspects about the firms participating in public procurement auctions can help improve the collusion detection rate of machine learning models; and
3. An in-depth analysis of ML models' behavior in detecting collusion, expanding on previous work [García Rodríguez et al. 2022]. This includes a study about the variability of the learning models when using different data split strategies and how this might impact decision-making related to collusion detection.

Our results shed light on the dataset challenges, specially when limited data is available, and demonstrate that our approach resulted in a significant improvement in balanced accuracy of the evaluated ML algorithms. Our methodology also decrease the variability in the results when compared to other methodologies.

The remainder of the paper is organized as follows. Section 2 discusses related work. Section 3 presents the original dataset used in [García Rodríguez et al. 2022]. Section 4 presents the methodology proposed in this paper. Then, Section 5 shows the experimental setup. Finally, Section 6 presents the obtained results and Section 7 concludes this paper.

2. Related Work

In recent years, the intersection of data science and ML has driven notable advancements in identifying and addressing the detection of collusive behavior in the procurement of goods and services for the public sector. The following synthesis presents recent studies that contribute to this field.

[Wallimann et al. 2023] used summary statistics, calculated with the screening variables for all possible subgroups of three and four bids, as input for ML models used to predict the risk of collusion. This proposed methodology is especially useful in the presence of competitive bids that distort the statistical signals of collusive behavior and were shown to be superior to other methods used to detect incomplete cartels.

[García Rodríguez et al. 2022] delve into the application of eleven distinct ML algorithms to identify collusion across public procurement auctions in diverse geographical contexts. Their research shows the efficacy of ensemble ML methods in pinpointing collusive behaviors, underscoring the transformative potential of ML in fostering transparency and equity in public procurement.

A novel approach, combining machine learning and screening methods, has been proposed by [Imhof and Wallimann 2021] to detect collusion in public procurement auctions. This methodology calculated screens based on coalitions of firms to detect the

presence of cartels. The effectiveness of this method was evaluated on data from different types of auctions in various countries.

[Wallimann and Sticher 2023] used machine learning models to detect collusion in railway infrastructure tenders. Using an novel approach to adapt and implement price-based screens, this work developed a tool that allows the sequential and decentralized screening of suspicious tenders.

Despite the good results reported, the repeated holdout is the preferred data split strategy of the works that use machine learning models to detect collusion. In this method, the data is divided into training and test sets, repeating this process with multiple random samples to define a final estimate. The variance of this process can be useful in estimating statistical confidence intervals on the error. However, these variations can have a significant impact when there is a class imbalance [Aggarwal et al. 2015].

This problem can be present in collusion detection applications, since collusive datasets are highly unbalanced. This makes the choice of data splitting strategy an important step when comparing multiple models and choosing the optimal model for collusion detection.

The use of features about the firms participating in the tenders is limited in the literature of collusion and cartel detection, with works focusing more on data about the bids and the auctions. However, several works used data about the bidders to detect other types of corruption.

[Villamil et al. 2024] tested firm-level network measures to study the relationship between ownership links and bidding behavior in procurement markets. [Lyra et al. 2021] use network science to study the co-bidding relationships between firms in Brazil with the goal of finding organizations that are more susceptible to frauds. [Decarolis and Giorgiantonio 2022] combined red flags and companies corruption measures to obtain a new measure for fraud in public procurement.

3. The "Operation Car Wash" Collusive Dataset

Between 2002 and 2013, a group of the biggest construction firms in Brazil formed secret and illegal agreements to embezzle billions of dollars from the state-owned Brazilian Oil Company Petrobras. In 2014, the "Operation Car Wash" was responsible for revealing instances of bid-rigging perpetrated by these companies in several of Petrobras procurement auctions [Signor et al. 2020b, Signor et al. 2021].

A version of the dataset, containing information about tenders during the period when the bid-rigging cartel responsible for defrauding Petrobras was active, was made available by [García Rodríguez et al. 2022]. The data contains information regarding the number of bids in the tender, the pre-tender cost estimate (PTE), the value of each bid in the tender, the percentage difference between each bid in the tender and the PTE, the date of the tender, the construction site of the tender and the Brazilian federative unit where the site is located, a flag to indicate the winning bid in the tender and a flag to indicate if the tender is collusive or competitive.

The dataset used by [García Rodríguez et al. 2022] also contains screening variables, statistical features about bids in a tender that can be used to improve the efficiency of collusion detection rate of ML methods. Table 1 shows the descriptive statistics for all numeric features in the dataset.

Screens, or screening variables, are statistical methods used to flag markets or firms for collusion investigation. Simple screens are a type of behavioral screen used to determine whether firms depart from competitive behavior. These statistics are build

Table 1. Descriptive statistics of the "Operation Car Wash" dataset.

	Mean	Std	Min	25%	Median	75%	Max
Bid Value	7.5x10^8	8.4x10^8	5.9x10^7	2.6x10^8	5.1x10^8	9.1x10^8	6.7x10^9
Pre-Tender Estimate	6.0x10^8	5.5x10^8	1.0x10^8	2.4x10^8	4.4x10^8	7.7x10^8	3.4x10^9
Difference Bid/PTE	0.20	0.32	-0.78	-0.01	0.15	0.35	1.96
Nr. Bids	9.33	5.08	2.00	5.00	9.00	12.00	21.00
CV	0.16	0.09	0.03	0.09	0.16	0.22	0.60
SPD	0.74	0.79	0.06	0.27	0.59	0.99	6.05
SKEW	0.30	0.98	-1.73	-0.39	0.29	0.65	3.40
DIFFP	0.10	0.08	0.00	0.04	0.08	0.13	0.50
RD	0.71	0.49	0.00	0.25	0.65	1.07	1.66
KSTEST	0.32	0.13	0.16	0.21	0.27	0.38	0.70

from the distribution of bids in a tender and each of them capture a different aspect of this distribution. The combination of multiple screens can help detect different types of manipulation [Huber and Imhof 2019].

The original dataset contains the Coefficient of Variation (CV) screen, used to evaluate the changes in the dispersion of bids. It also contains the Skewness (SKEW) screen, used to find how collusion changes the symmetry of bids [Huber and Imhof 2019].

The next screen used is the Spread (SPD), defined in Eq. 1. This feature describes the relative difference between the maximum bid and the lowest bid:

$$SPD_t = \frac{max_t - low_t}{low_t} \tag{1}$$

where, for each tender *t*, max_t is the most expensive bid in *t* and low_t is the cheapest bid in *t*.

The next screen used is the Difference Between the Two Lowest Bids (DIFFP), shown in Eq. 2, that measures the relative difference between the two lowest bids:

$$DIFFP_t = \frac{2nd_t - low_t}{low_t} \tag{2}$$

where, for each tender *t*, $2nd_t$ is the second-lowest bid in *t* and low_t is the cheapest bid in *t*.

Another used is the Relative Distance (RD), defined in Eq. 3. This feature calculates the relative distance in a tender using the ratio of the difference between the two lowest bids and the standard deviation of the losing bids:

$$RD_t = \frac{2nd_t - low_t}{sdl_t} \tag{3}$$

where, for each tender *t*, $2nd_t$ is the second-lowest bid in *t*, low_t is the cheapest bid in *t* and sdl_t represents the standard deviation of all losing bids in *t*.

The last screen is the Kolmogorov-Smirnov Test (KSTEST), defined in Eq. 4, that is used to check if the bid values of a tender follow a uniform distribution:

$$KSTEST_t = max\left(D_t^+, D_t^-\right), \\ D_t^+ = max_i\left(\frac{b_{it}}{sd_t} - \frac{i_t}{n_t+1}\right), D_t^- = max_i\left(\frac{i_t}{n_t+1} - \frac{b_{it}}{sd_t}\right) \quad (4)$$

where, for each tender *t*, b_{it} is the bid for the i^{th} rank in *t*, sd_t is the standard deviation of all bids in *t*, n_t is the number of bids in *t* and i_t is the rank of a bid in *t*.

Several works in the literature used screens to improve the results of collusion and cartel detection using learning models [García Rodríguez et al. 2022, Imhof and Wallimann 2021, Huber et al. 2022]. However, screening variables are restricted to be calculated within a tender. Combining the screens with features related to the companies bidding in the tenders might help improve the detection of collusive behavior in public procurement [Wallimann et al. 2023].

4. Methodology

Our methodology, described in the Figure 1, consists of the following steps: feature extraction to enrich the original dataset used in our work, definition of cross-validation strategies, definition of the learning models, definition of a set of values for the hyperparameters, definition of a pipeline to execute hyperparameter optimization and model fitting process and evaluation of the fitted models.

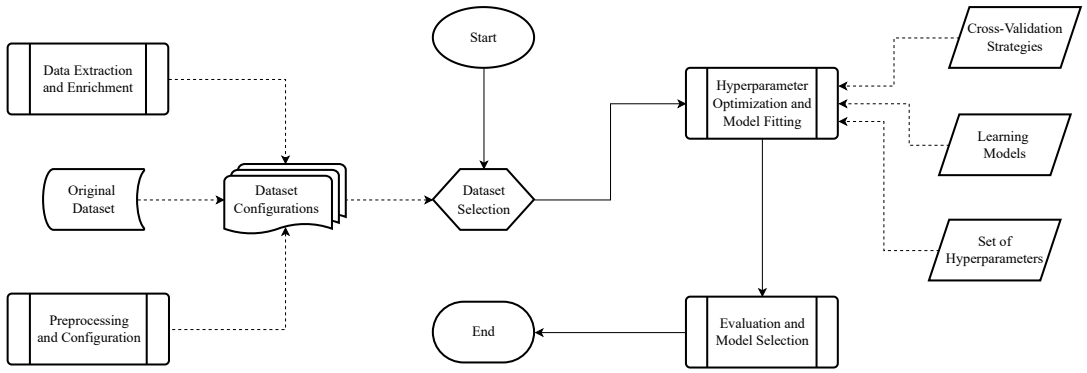


Figure 1. Steps describing our proposed methodology.

4.1. Data

Our work used a subset of dataset provided by [García Rodríguez et al. 2022] to evaluate our proposed methodology. This data contains information about public procurements from different countries and time periods. We chose to use only the Brazilian subset of the data because we want to extract information about the participating companies from other sources, enrich the data, and compare the prediction results using the original and the enriched versions of the dataset.

4.2. Enrichment Features

The first step in the enrichment process was to find a data source containing information about the companies in the dataset. We chose the National Register of Judicial Person[1], a

[1]https://dados.gov.br/dados/conjuntos-dados/cadastro-nacional-da-pessoa-juridica---cnpj

database maintained by the Special Department of Federal Revenue of Brazil that stores information about organizations and legal entities. In this database, every entity is given a unique identification number, commonly called CNPJ, that is used to retrieve the company's name, creation date, size and other information.

The original dataset, however, did not have the CNPJ of the bidders. Based on the Access Information Law – a federal law that guarantees the right to access information generated or stored by public entities in Brazil – we used the federal government Transparency Portal[2] to access the files containing the data regarding the public procurement in the original dataset. We executed 131 access information requests in the Transparency Portal, to obtain all tender reports. In total we received 100 reports where:

- 63 reports containing the companies CNPJs, in a legible way;
- 8 reports containing the companies CNPJs, but illegible (scanned images from documents);
- 10 reports without the companies CNPJs;
- The physical file of 13 reports could not be found;
- 6 reports that do not have a tender report;
- The identification code of one tender does not exist;

We manually extracted the companies CNPJs from all the reports and inserted them in electronic spreadsheets. After that, we verified that 127 bids did not have the CNPJ of the bidding company. To fix this, we checked if the CNPJ existed in other tenders that the company also participated. We ended up with only eight bids without the company's CNPJ after this process.

We made a manual search in the CNPJ database using the complete or partially company name to try to find the CNPJ of the remaining companies. When the search returned more than one result, we selected the company with the largest capital investment. We adopted this criterion because the tenders in our dataset involve huge amounts of money, and typically smaller companies do not compete in these types of tenders.

After defining the CNPJ for all companies, we used this information to extract the following features from the CNPJ database: legal nature, capital invested, founding date, number of CNAEs[3], and the number of partners in the company. These features are briefly described in Table 2. We then incorporated these features into our dataset.

Table 2. Description of the enrichment features.

Feature	Description
Legal Nature	Type of structure of the firm. This code classifies the firm among the existing types in the legislation and determines how it operates
Capital Invested	Total value of resources and assets owned by the firm that were invested by partner or shareholders
Founding Date	Date on which the company started operating
Number of CNAEs	Number of economic activities carried out by the company
Number of Partners	Number of partners in the company

After that, the next step in our experimental setup involved transforming certain features and the excluding of elements based on specific conditions. This process is described in the next section.

[2]https://falabr.cgu.gov.br/web/home

[3]CNAE is the national classification of economic activities in Brazil, used for administrative, tax, and statistical purposes to standardize and organize economic data

4.3. Data Preprocessing

The first step during preprocessing was to remove all tenders with less than three bids. We implemented this to continue utilizing the RD screens, showed in Eq. 3, that needs at least three bids to be calculated. Since only one tender in our dataset had two or fewer bids, we opted to remove only this single tender instead of eliminating the RD feature entirely.

The "Difference Bid/PTE" feature represents the percentage difference between the pre-tender estimate value and the bid value. This feature was modified to the total difference (in BRL) between the pre-tender estimate value and the bid value. This adjustment was made to maintain the same scale between all monetary values in the dataset.

In the original dataset, approximately 94% of all tenders are either entirely free of collusion (competitive) or are completely collusive. About 6% of all tenders have partial collusion, where only a portion of the competitors engage in collusive behavior during the process. Because of this imbalance, we conducted experiments using all tenders in the dataset and experiments excluding the partially collusive tenders. We made these experiments to evaluate the impact of these tenders on the results of the predictive models.

5. Experimental Setup

We utilized popular data science and ML libraries such as Pandas [McKinney 2010] and Scikit-Learn [Pedregosa et al. 2011] to conduct the experiments in this work. Detailed configurations of the methods used in the experiments are described in the next sections. The code and data used in the experiments can be found in the Supplementary Data section at the end of this document.

In our work we tested the learning models with four configurations for the input data: the original dataset with all tenders (C1), the original dataset without partial collusion (C2), the enriched dataset with all tenders (C3) and the enriched dataset without partial collusion (C4).

We chose to apply our methodology to the models that achieved the best results found in [García Rodríguez et al. 2022]. We utilized the tree-based classifiers Extra-Trees and Random Forest (RF), the boosting-based classifiers AdaBoost and Gradient Boosting. We also used the neural network based classifier Multi-Layer Perceptron (MLP).

Given that we are addressing a classification problem, we employed the following metrics to evaluate the results of the predictive models: Accuracy, Balanced Accuracy, Precision, Recall, and F1. For brevity, we only report the balanced accuracy values in our results. This decision was driven by the fact that the minority class represents only 15,62% of the records in the dataset. In this case, the balanced accuracy might be a better evaluation metric since it assigns a greater weight to the minority classes when compared to the accuracy that tends to favour the majority class [Grandini et al. 2020].

We tested three cross-validation strategies: K-Fold, Repeated K-Fold, and Nested K-Fold. All K-fold strategies use the Scikit-Learn default value of 5 splits. In the Repeated K-Fold strategy, the split is repeated 10 times with different data shuffling each time. All cross-validation methods use the tender group to split the data, this approach ensures that all bids from a single tender are not split between the training and test datasets, maintaining the integrity of individual tender data.

5.1. Hyperparameter Optimization Configuration

Regardless of the applied method, all cross-validation strategies were executed during the hyperparameter optimization step of our methodology. We used a randomized method for searching the optimal set of hyperparameter across all possible sets in the search space.

This search step had 50 iterations, and each model had its own set of hyperparameters being tested. Only the scaling and encoding parameters were shared by all models. Table 3 shows the hyperparameter optimization configuration from each model.

Table 3. Hyperparameter configuration for each model.

ExtraTrees	RandomForest	AdaBoost	Grad. Boost.	MLP
n_estimators	n_estimators	estimator	n_estimators	layer_sizes
max_features	max_features	n_estimators	learning_rate	activation
criterion	criterion	learning_rate	max_depth	solver
min_samples_split	min_samples_split			max_iter

The proposed methodology consists of a series of ordered steps to be executed. Our pipeline has the following steps: encoder, scaler, and estimator. The process starts with the encoding of the categorical features in the dataset, continues with the scaling of all features, and finishes with the creation and fitting of a predictive model based on the set of hyperparameters chosen by the randomized search in each iteration.

We used the following encoders from the Category Encoders[4] library as hyperparameter values during the encoding step: BinaryEncoder, QuantileEncoder, OneHotEncoder, RankHotEncoder, HashingEncoder, and CountEncoder. The original dataset contains two categorical features: site and brazilian state. The enriched dataset also includes the legal nature as a categorical feature.

After the encoding of the categorical features, the next step in our pipeline is the scaling. In this step, the scaler method chosen during the randomized search is applied to all features in the dataset. We used the following scalers from the Scikit-Learn library during the scaling step: StandardScaler, MinMaxScaler, MaxAbsScaler, RobustScaler, Yeo-Johnson PowerTransformer and QuantileTransformer.

The last step in our pipeline is the estimator. This step is responsible for randomly selecting a set of values to be used in the configuration of the model. After that, the model is fitted and all evaluation metrics are calculated on the hold out data. The process is repeated 50 times, and each iteration returns the best set of hyperparameters for each model. Here, the best set of hyperparameters is the set that achieved the highest balanced accuracy value.

6. Results and Discussion

As mentioned earlier, the objective of this work is to develop a methodology that employs techniques such as cross-validation and hyperparameter optimization to enhance the collusion detection rate of ML methods, compared with other methodologies documented in the literature. Using [García Rodríguez et al. 2022] as our baseline work, we applied their methodology and compared their results with our solution. Table 4 shows the results of applying the baseline methodology to all the models tested in our work.

Table 4 showcases that, despite the good prediction results, the baseline solution presents a high variance. This variance, indicated by the high standard deviation values when using different train/test splits, raises questions related to model selection when using our methodology to detect collusion in public procurements. Our work aims to evaluate whether feature enrichment and cross-validation are valid solutions to decrease the variance of the results, especially when limited data is available.

Our baseline work, just like other similar works, did not reported the standard deviation (variability) in the results of the models tested. The use of repeated holdout in

[4]https://contrib.scikit-learn.org/category_encoders/index.html

Table 4. Results of the baseline methodology [García Rodríguez et al. 2022] for all tested models.

Model	Balanced Accuracy	Standard Deviation
AdaBoost	84.11	11.02
ExtraTrees	85.63	10.98
GradientBoosting	85.74	9.61
MLP	75.17	10.60
RandomForest	86.86	10.27

these works raised questions about the reliability of model selection, specially on new data. We tried to solve this problem using different cross-validation techniques and evaluating the changes in the variability of the results when using different data split strategies.

The first cross-validation method we used was the Group K-Fold with 5 splits. Figure 2 shows the results of this strategy across all models. The RandomForest and AdaBoost models, using both dataset configurations but excluding tenders with partial collusion, achieved the highest balanced accuracy values among all the models we tested.

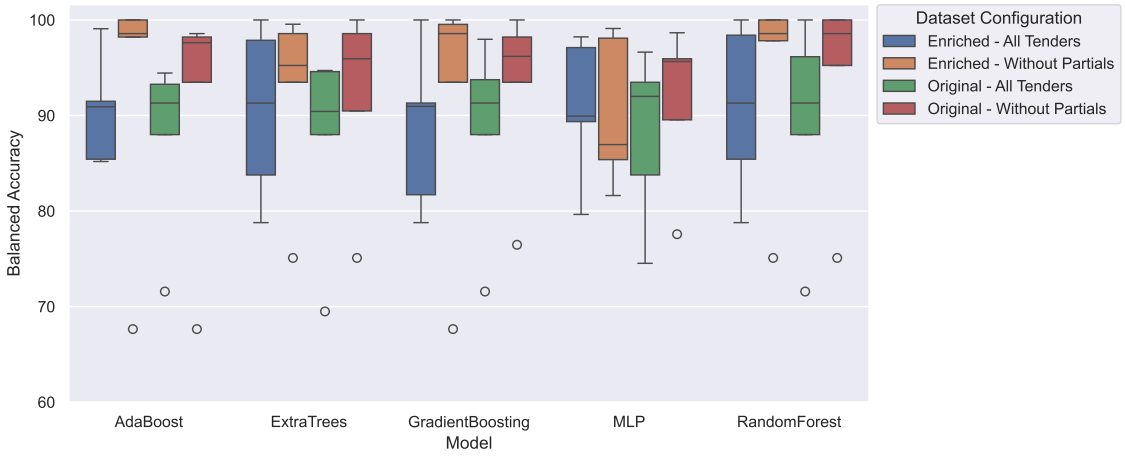


Figure 2. Results for Group K-Fold with 5 splits for all models using different dataset configurations.

However, the AdaBoost and MLP models achieved the lowest standard deviation between all models tested using the enriched dataset version with all tenders. This indicates that with the K-Fold cross-validation strategy, using a dataset containing tenders without partial collusive tenders might improve the results of the predictive models, but it might not necessarily reduce the variance in the results.

Subsequently, we tested the Group K-Fold cross-validation with 5 splits and 10 repetitions. Figure 3 shows the results for all the tested models, using different seeds, grouped by dataset configuration. With this repetition strategy, the tree-based models achieved the results using both dataset configurations without partial collusion. In terms of variance, the use of the enriched dataset with all tenders showed the best results, regardless of the model being used. The MLP model had the smallest standard deviation, followed by the boosting-based models and then the tree-based models.

We also tested the nested cross-validation strategy in our work. This method starts with the data spplitting using Group K-Fold with 5 splits. Then, an outer conditional loop

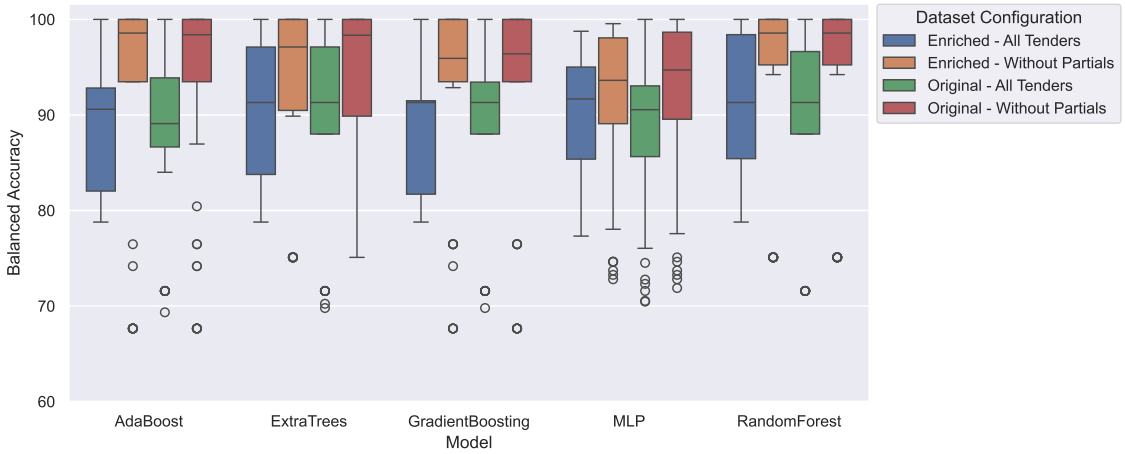


Figure 3. Repeated K-Fold results for all models using different dataset configurations.

iterates over all splits in the data. Within this loop, all training splits are used in a hyperparameter optimization process using the RandomizedSearchCV and another Group K-Fold with 5 splits. The model with the best results during this operation was then evaluated using the test split (holdout). This process is repeated until all data splits are used to evaluate the best model found in the hyperparameter optimization process. Figure 4 shows the average results for the test splits using all models.

It is important to note that, as shown by Figure 4, the configuration using the MLP model, the enriched dataset with all tenders, and the nested cross-validation achieved the least variation in the results when compared to the previous strategies using the same configuration. With this configuration, the MLP model achieved a mean balanced accuracy of 88.37% and a standard deviation of 2.15%, a difference of 2.48% when compared to the cross-validation strategy that achieved the best results when using the same configuration, but an increase of 5.3% when the same configuration.

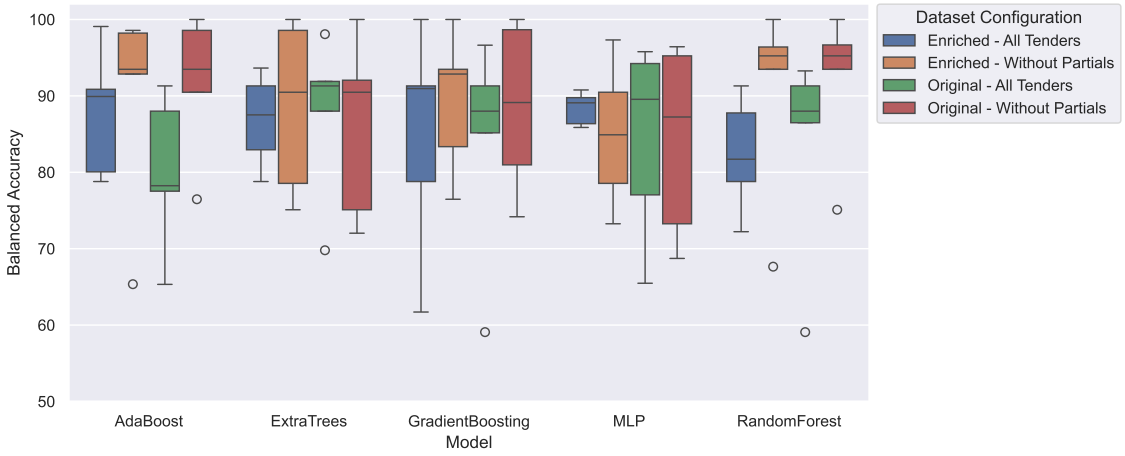


Figure 4. Average results for test splits in the nested cross-validation using all models tested.

Cross-validation using limited data, like the data used in our work, creates a trade-off between generalization and fitting a better model [Korjus et al. 2016]. Our results, however, show that a trade between the model with better results for a model that might

generalize better with new data could be indicated if the quality of the results only decreases between 2% and 3%. A summary of the best results achieved with the cross-validation strategies tested in our work is presented in Table 5. The tree-based models using the datasets without partial collusion achieved the best results in our experiments.

Regarding the variance in the results of our methodology, Table 5 shows that the configurations that achieved the overall best results in our work are not the ones that achieved the smallest variance. The configurations with the smallest standard deviation are the ones using the MLP model and the enriched version of the dataset with all tenders.

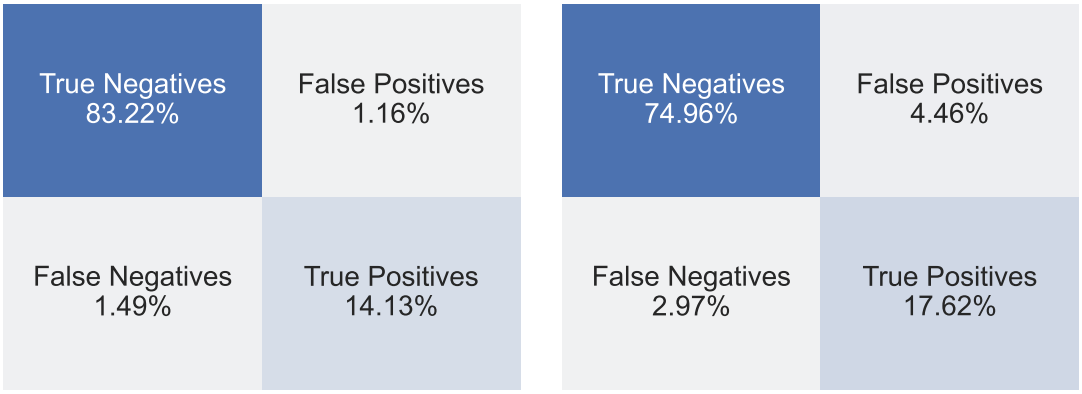
Table 5. Comparison between the results of our methodology and the base article

	Our Methodology			**Base Article**		**Difference**		
CV Strategy	Model (Dataset)	Bal. Acc.	Std	Bal. Acc.	Std	Bal. Acc.	Std	p-value
K-Fold	RF (C4)	**94.30**	10.77	**86.86**	**10.27**	7.44	0.50	0.31
	AdaBoost (C3)	90.41	5.67	84.11	11.02	6.30	-5.34	0.06
Repeated	RF (C4)	93.79	9.57	**86.86**	**10.27**	6.93*	-0.69	$5.1x10^{-5}$
	MLP (C3)	90.22	6.07	75.18	10.60	**15.04***	-4.53	$1.8x10^{-15}$
Nested	RF (C2)	92.10	9.80	**86.86**	**10.27**	5.24	-0.47	0.44
	MLP (C3)	88.37	2.15	75.18	10.60	13.19	**-8.45**	0.06

* Using the Wilcoxon signed-rank test with Bonferroni correction, the difference was found to be statistically significant.

Overall, the best models for each CV strategy defined in our methodology achieved better results when compared with the baseline models. Regardless of the evaluation, the models trained using the CV strategies defined in our work improved the results by up to 15% when compared to the base article. However, only the Repeated K-fold strategy had a statistically significant difference in the results.

For a more detailed evaluation of the models trained using the repeated k-fold strategy, we build the confusion matrix for the results of the models that had a statistically significant difference when compared to the baseline paper. The results presented in Figure 5 are the average for all test folds and show that the model with a higher variability was able to produce less false positives when compared to the more robust model using the same data split strategy.



(a) RF (C4) model **(b) MLP (C3) model**

Figure 5. Confusion matrix achieved using the repeated k-fold strategy

The results from Table 5 showed that the models that had the better results also had a higher variability between the balanced accuracy values from each test fold. However, as shown in Figure 5, these less robust models had more success when differentiating between collusive and competitive tenders. This shows that a choice must be made between

a more powerful model or a more reliable model, underscoring the complexity of model selection for the detection of collusion in public procurement.

To conclude, an analysis of all the results presented in this section shows that the best results in our experiments were achieved, for the most part, using the enriched dataset. The results showed that audit agencies and other decision makers in the public sector should consider implementing data enrichment and model selection with hyperparameter optimization and the correct data split strategy to their current pipelines of screening tenders for possible collusion.

7. Conclusion

This work highlights the potential of machine learning algorithms in detecting and mitigating fraud in public procurement processes. By enriching the dataset from the "Operation Car Wash" investigation and employing machine learning techniques such as hyperparameter optimization and cross-validation, this research was able to provide new insights into improving ML models for detecting collusion in procurement processes. The proposed methodology for model selection achieved better results compared to previous studies, and all code and dataset developed are made publicly available.

The findings underscore the crucial importance of data quality and the careful selection of machine learning methodologies in enhancing fraud detection rates. This advancement is particularly vital for public organizations seeking to uphold transparency and integrity in procurement processes, ultimately contributing to more effective governance and the prevention of substantial financial losses.

Future work should focus on refining these methods by identifying the dataset features that most significantly influence the models and investigating new approaches such as graph-based neural networks (GNN) to further bolster the fight against corruption in public procurement. Another area of focus for future works also include the use of AutoML tools to automate the model selection process.

Supplementary Data. The supplementary data and code used in this research can be found online at: https://zenodo.org/records/11491748

Acknowledgments. This research was funded by the Coordination for the Improvement of Higher Education Personnel Foundation (CAPES) and the Public Prosecutor's Office of Santa Catarina (MPSC).

References

Aggarwal, C. C. et al. (2015). *Data mining: the textbook*, volume 1. Springer.

Curtis, F. and Maines, P. (1973). Closed competitive bidding. *Omega*, 1(5):613–619.

Decarolis, F. and Giorgiantonio, C. (2022). Corruption red flags in public procurement: new evidence from italian calls for tenders. *EPJ Data Science*, 11(1):16.

García Rodríguez, M. J., Rodríguez-Montequín, V., Ballesteros-Pérez, P., Love, P. E., and Signor, R. (2022). Collusion detection in public procurement auctions with machine learning algorithms. *Automation in Construction*, 133:104047.

Grandini, M., Bagli, E., and Visani, G. (2020). Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*.

Huber, M. and Imhof, D. (2019). Machine learning with screens for detecting bid-rigging cartels. *International Journal of Industrial Organization*, 65:277–301.

Huber, M., Imhof, D., and Ishii, R. (2022). Transnational machine learning with screens for flagging bid-rigging cartels. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 185(3):1074–1114.

Imhof, D. and Wallimann, H. (2021). Detecting bid-rigging coalitions in different countries and auction formats. *International Review of Law and Economics*, 68:106016.

Korjus, K., Hebart, M. N., and Vicente, R. (2016). An efficient data partitioning to improve classification performance while keeping parameters interpretable. *PloS one*, 11(8):e0161788.

Lyra, M. S., Curado, A., Damásio, B., Bação, F., and Pinheiro, F. L. (2021). Characterization of the firm–firm public procurement co-bidding network from the state of ceará (brazil) municipalities. *Applied Network Science*, 6:1–10.

McKinney, W. (2010). Data structures for statistical computing in python. In van der Walt, S. and Millman, J., editors, *Proceedings of the 9th Python in Science Conference*, Proceedings of the Python in Science Conference, pages 56–61. SciPy.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Porter, R. H. and Zona, J. D. (1993). Detection of bid rigging in procurement auctions. *Journal of political economy*, 101(3):518–538.

Signor, R., Ballesteros-Pérez, P., and Love, P. E. (2021). Collusion detection in infrastructure procurement: A modified order statistic method for uncapped auctions. *IEEE transactions on engineering management*, 70(2):464–477.

Signor, R., Love, P. E., Belarmino, A. T., and Alfred Olatunji, O. (2020a). Detection of collusive tenders in infrastructure projects: Learning from operation car wash. *Journal of Construction Engineering and Management*, 146(1):05019015.

Signor, R., Love, P. E., and Ika, L. A. (2020b). White collar crime: Unearthing collusion in the procurement of infrastructure projects. *IEEE Transactions on Engineering Management*, 69(5):1932–1943.

Velasco, R. B., Carpanese, I., Interian, R., Paulo Neto, O. C., and Ribeiro, C. C. (2021). A decision support system for fraud detection in public procurement. *International Transactions in Operational Research*, 28(1):27–47.

Villamil, I., Kertész, J., and Fazekas, M. (2024). Collusion risk in corporate networks. *Scientific Reports*, 14(1):3161.

Wallimann, H., Imhof, D., and Huber, M. (2023). A machine learning approach for flagging incomplete bid-rigging cartels. *Computational Economics*, 62(4):1669–1720.

Wallimann, H. and Sticher, S. (2023). On suspicious tracks: machine-learning based approaches to detect cartels in railway-infrastructure procurement. *Transport Policy*, 143:121–131.