WS5 – Quality Framework for Statistical Algorithms
Final Report

December 2021

Author: Jose Jimenez, INEGI, Mexico

Objectives

The goal is to explore the dimensions of the Quality Framework for Statistical Algorithms (QF4SA) in a consolidated project to analyse the output based on a set of standard metrics and procedures.

Outputs

Even though the model was assessed in an early stage, the initial expectation was to perform a thorough evaluation comprising a set of other metrics and procedures. Therefore, this report is delivered to inform on a set of indicators about the performance of the dimensions specified in the QF4SA.

Project name and previous achievements

The project where the framework was tested on, originated as part of the HLG-MOS 2020 results, the name was **"Occupation and Economic activity coding using natural language processing".** As such, the goal of that project was to leverage Machine Learning models to automate the process of coding regarding such activities. The project concluded in an initial phase without being used in production. The project was aimed to predict an occupation classification based on a set of text data such as the description of the activity and the tasks related with the activity according to NAICS (SCIAN) and SINCO, as shown in Figure 1.

Occupation	SINCO	IND_SINCO	Task	...
Taxi Driver	4586	3	To move people	
Owner	1111	2	Pay, Sell merchandise	

Figure 1. Data for economic activities and their classification

A series of transformations are performed on categorical data belonging to NAICS, after these, another series of transformations to both numeric and text data is done, the goal is to match text labels to the NAICS classification. Several Machine Learning (ML) classifiers are used for classification: SVM, RandomForestClassifier, MLPClassifier, LogisticRegression, DecisionTreeClassifier, GaussianNB, KNeighborsClassifier, XGBClassifier and ExtraTreesClassifier. As expected, the ML algorithms performed differently, showing diverse performance in accuracy and in time. The SVM classifier had the best results with an accuracy of 88.32% done in 87 minutes.

Dimension 1: Explainability

The first approach to show Explainability was to shuffle the values in the columns: col_1, col_2, …col_n, changed to col_n, … col_2, …col_1. These changes were made in text and numerical columns used in the model. The results for SVM were Accuracy: 88.37%, Time: 82 minutes. These results show the classifier can have a different output according to input data. The second approach was to use an adversarial example by changing the values in the text columns, for this, different words were added as prefix of the column. The results for SVM were Accuracy: 88.22%, Time: 97 minutes (execution time was higher). These results show the classifier can have a different output according to input data, therefore, the model reacts to such changes in data.

Dimension 2: Accuracy

These ML classifiers were used for classification: Random Forest Classifier, Extra Trees Classifier, Multi Layer Perceptron Classifier, Logistic Regression and Linear SVM, the latter was the classifier with the higher accuracy: 88.32%. Then several experiments were performed by the team, the goal of such experiments was to increase the initial accuracy achieved by LSVC. The SVM classifier was presented as the best ML algorithm for this task, having the best results in Accuracy and Execution time. These experiments included changes using different Number of classes, Dimensions, ML classifiers, Auxiliar Variables, Dimensionality reduction techniques and Class Balancing. After performing all the referred experiments, only one had a slight increment in accuracy, it was from 88.32 to 88.39%. Some of these results are shown in figure 2 and 3.

	10000 dims						*10000 dims / VAR AUX*					
Freq Class	*f1_macro*	*accuracy*	*recall*	*precision*	*time/secs*	*MODEL*	*f1_macro*	*accuracy*	*recall*	*precision*	*time/secs*	*Freq Class*
>=4	0.5248	0.8447	0.4916	0.6277		RandF	0.5352	0.8504	0.5002	0.6448	X-391.83	>=4
>=4	0.6081	0.8535	0.5955	0.679		ExtraT	0.6104	0.8602	0.595	0.6844	X-1308.16	>=4
>=4	0.6115	0.8541	0.6056	0.6399		MLPC	0.6321	0.867	0.6244	0.6582	X-6535.81	>=4
>=4	0.5361	0.8607	0.5172	0.5976		LR	0.5617	0.87	0.5396	0.6249	X-1742.30	>=4
>=4	0.6426	0.8652	0.6717	0.6367	X-150.07	LSVC	0.6428	0.8748	0.67374	0.6315	X-1397.16	>=4
>=40	0.6142	0.8491	0.5792	0.722	X-966.69	RandF	0.6289	0.854	0.5919	0.7371	39.54	>=40
>=40	0.6785	0.8594	0.664	0.7402	X-2440.83	ExtraT	0.698	0.8663	0.6812	0.7641	85.62	>=40
>=40	0.6762	0.853	0.6677	0.6939	X-8165.80	MLPC	0.6994	0.8674	0.6924	0.7223	5004.11	>=40
>=40	0.6399	0.8654	0.6144	0.7287	X-461.87	LR	0.4538	0.7962	0.4346	0.527	245.29	>=40
>=40	0.7026	0.8663	0.7324	0.6866	X-101.01	LSVC	0.7203	0.8765	0.7507	0.7022	1366.65	>=40
>=100	0.695	0.8519	0.6636	0.7828	X-819.43	RandF	0.711	0.855	0.6747	0.8021	38	>=100
>=100	0.7444	0.8629	0.7299	0.7906	X-2472.81	ExtraT	0.7533	0.8681	0.7363	0.801	74.6	>=100
>=100	0.7472	0.8654	0.739	0.7597	X-7784.17	MLPC	0.7603	0.8732	0.752	0.7751	5249.08	>=100
>=100	0.7225	0.8682	0.6995	0.7946	X-383.32	LR	0.5671	0.8055	0.5418	0.6656	199.69	>=100
>=100	0.7569	0.8708	0.7789	0.7437	X-95.97	LSVC	0.7759	0.8807	0.7935	0.7642	1065.43	>=100

Figure 2. Accuracy Results – 10000 dimensions matrix

Freq Class	25000 dims					MODEL	25000 dims / VAR AUX					Freq Class
	f1_macro	accuracy	recall	precision	time/secs		f1_macro	accuracy	recall	precision	time/secs	
>=4	0.5212	0.8397	0.483	0.6627	116.89	RandF	0.5284	0.8441	0.4904	0.6512	71.62	>=4
>=4	0.6076	0.8513	0.5944	0.6789	306.99	ExtraT	0.614	0.8559	0.5975	0.6993	182.43	>=4
>=4	0.6235	0.8583	0.6112	0.6625	15881.84	MLPC	0.6501	0.8704	0.6475	0.6672	11139.18	>=4
>=4	0.5577	0.8677	0.5346	0.6216	510.07	LR	0.3492	0.7871	0.3379	0.4172	480.08	>=4
>=4	0.6461	0.8671	0.673	0.6398	141.56	LSVC	0.6518	0.8755	0.6744	0.6464	1638.27	>=4
>=40	0.604	0.846	0.5664	0.7236	72.2	RandF	0.6149	0.8503	0.5754	0.7341	71.14	>=40
>=40	0.6754	0.8567	0.6587	0.753	167.76	ExtraT	0.6935	0.8624	0.6723	0.7695	164.05	>=40
>=40	0.6908	0.8597	0.6781	0.7191	1210.87	MLPC	0.6996	0.868	0.6907	0.732	12789.69	>=40
>=40	0.6632	0.872	0.6366	0.7441	410.87	LR	0.429	0.7883	0.4122	0.497	409.14	>=40
>=40	0.7104	0.8706	0.737	0.6962	121.06	LSVC	0.7251	0.8787	0.7506	0.7112	1365.33	>=40
>=100	0.6895	0.8491	0.6546	0.7928	69.35	RandF	0.7032	0.8532	0.6647	0.8111	68.74	>=100
>=100	0.7366	0.8589	0.7192	0.7944	150.92	ExtraT	0.7467	0.8627	0.7268	0.8052	147.28	>=100
>=100	0.754	0.8662	0.7439	0.7704	8693.1	MLPC	0.7584	0.8722	0.7555	0.769	12155.26	>=100
>=100	0.7426	0.8747	0.7204	0.8092	364.09	LR	0.5492	0.7982	0.5259	0.6521	330.76	>=100
>=100	0.7651	0.874	0.7846	0.754	100.26	LSVC	0.7829	**0.8839**	0.7997	0.7718	1202.88	>=100

Figure 2. Accuracy Results – 25000 dimensions matrix

Dimension 3: Reproducibility

"In machine learning, reproducibility is being able to recreate a machine learning workflow to reach the **same conclusions** as the original work" (Preeti Hemant). Currently, many research studies are difficult to reproduce independently. Methods Reproducibility could be an alternative for NSOs when sharing their experience in developing ML models and internal data cannot be shared outside the organization.

Types of Reproducibility

Methods reproducibility: they provide sufficient detail about procedures and data, so same procedures could be repeated exactly. But it has limited availability across the different platforms and their implementations (Python, R, Julia, etc.).

Results reproducibility: they obtain the same results from a study with procedures as closely matched to the original study as possible. It presents high dependance on input data.

Inferential reproducibility: they draw the same conclusions from either an independent replication of a study or a reanalysis of the original study.

Guidelines on Reproducibility

- Commonly, data scientists have a different logic as well as diverse technical skills, therefore, documentation in-code is highly advisable. This documentation practices are also a part of maintainability.
- To document the details of how the model was trained are also useful for future improvements.

- It is also recommended to perform control versioning in both training data and feature generation.
- There must be provided the details of the software used to construct the ML model such as versioning and packages/libraries used, as they change every year.

There was a way for Reproducibility inside INEGI as code and data were shared to new ML teams, and the re results could be replicated or improved.

Dimension 4: Timeliness

According to NSOs, Timeliness is the length of time between the reference period and the availability of information. It should cover the period of time between a need for data and the release of the information to meet that need (QF4SA). Therefore, it is the length of time between the reference period and the availability of information. According to QF4SA, some of the elements to be considered in this dimension are:

- Data cleansing: Sometimes cleaning or preparing data can take a long time. After having the initial dataset, then it may become a repetitive task.
- Informatics infrastructure: To consider if the NSO has the infrastructure needed or new hardware should be acquired.
- Evaluation of data quality: To establish benchmarks on the quality of the data and keep reviewing such quality when new data is aggregated.
- Scalability of the approach: To analyze if the current methods and infrastructure can support taking the ML model to analyze large volumes of information and being able to inform results at their needed time.

Dimension 5: Cost Effectiveness

This dimension is the degree to which results are effective in relation to the costs of obtaining them. In terms of QF4SA, cost effectiveness is defined as the accuracy (measured by the MSE for continuous data and F1 score or similar metrics for categorical data) per unit cost. In table 1, we reflect the elements that must be considered when analyzing this dimension for a ML project.

Table 1. Potential additional fixed and ongoing costs for machine learning adoption.

Cost component	Type	Purpose	Comparison
IT infrastructure	Fixed	Acquiring necessary hardware and software	- No dedicated unit, but is part of our sandbox. - Less computers dedicated to this task
Cloud storage	Ongoing	Acquiring necessary cloud storage space	- *Cloud constraints*
IT maintenance	Ongoing	Maintaining IT infrastructure	- Lot cheaper than maintaining several computers - Code maintenance should be considered
Initial staff training	Fixed	Training current staff on ML; may include hiring new staff	- Cheaper than training new human coders - LCiD Team
Ongoing staff training	Ongoing	Keeping staff up to date withnew ML developments	- This is a cost to consider
Data acquisition	Fixed/ ongoing	Acquiring and processing new data sources	- Because of the relevance of the survey, normal data will be estimated, new sources might be integrated
Quality assurance	Ongoing	Conducting quality assurance and control	- This could be an internal or external control

Conclusions

- More ML projects should be evaluated using QF4SA, not only those from NSOs, traditional metrics (F1, ROC, AUC, etc.) do not report the outcome of a ML from a holistic view.
- The framework must be periodically revised to increase the dimensions, or to improve current ones according to new findings.
- The evaluation of Deep Learning models should be considered in the dimensions and integrate this in the framework.
- Output from ML models can be analyzed using the framework and compared vs their metrics to show the arising differences.