SSC Annual Meeting, May 2021
Proceedings of the Survey Methods Section

A MACHINE LEARNING VALIDATION PROTOCOL CUSTOMIZED FOR AN OFFICIAL STATISTICS USE CASE

Keven Bosa[1], Kenneth C.K. Chu[2]

ABSTRACT

In many official statistics production settings, the underlying data generation mechanism may be non-stationary, i.e., it may change, across production cycles. Indiscriminate use in such settings of many conventional supervised machine learning validation and testing protocols (e.g., hold-out and k-fold cross-validation) implicitly assumes stationarity, and risks temporal data leakage or underestimation of generalization error when the stationarity assumption is violated. In this article, we introduce a validation and testing protocol that remains valid under non-stationarity, and demonstrate its use by applying it to train a crop yield prediction model for the Field Crop Reporting Series [Statistics Canada (2021)].

KEY WORDS: Non-stationarity, temporal data leakage, underestimation of generalization error, rolling window forward validation, crop yield prediction

RÉSUMÉ

Dans de nombreux contextes de production de statistiques officielles, le mécanisme sous-jacent de génération de données peut être non stationnaire, c.-à-d. qu'il peut changer, au cours des cycles de production. Dans de tels contextes, l'utilisation sans discernement de nombreux protocoles classiques de validation et de test de l'apprentissage automatique supervisé (par ex., la validation croisée à k blocs, et la validation à l'aide de groupes exclus) suppose implicitement la stationnarité, et risque de provoquer une fuite de données temporelle ou une sous-estimation de l'erreur de généralisation si l'hypothèse de stationnarité est violée. Dans cet article, nous présentons un protocole de validation et de test qui reste valide en cas de non-stationnarité, et démontrons son utilisation en l'appliquant à l'entraînement d'un modèle de prédiction du rendement des cultures pour la Série de rapports sur les grandes cultures [Statistics Canada (2021)].

MOTS CLÉS : non-stationnarité; fuite de données temporelle; sous-estimation de l'erreur de généralisation; validation de la fenêtre mobile progressive, prédiction du rendement des cultures.

1 INTRODUCTION

Many conventional validation and testing protocols for supervised machine learning, such as k-fold cross-validation and hold-out, implicitly assume that the underlying data generation mechanism of the training, validation and testing data sets is identical to that of the data on which a trained model will be deployed in practice. However, in many official statistics production settings, supervised machine learning models deployed in the current production cycle are almost invariably trained and tested on data from past production cycles. In such settings, error estimates resulting from conventional validation and testing protocols are valid only under a stationarity assumption on the underlying data generation mechanism (i.e., the underlying data generation mechanism is tacitly assumed fixed across production cycles). Violation of this stationarity assumption risks temporal data leakage, or, more concretely, underestimation of validation or generalization error.

There are well estalbished validation protocols to avoid temporal data leakage in the non-stationary time series forecasting literature. In particular, the *rolling window forward validation* (RWFV) protocol [Schnaubelt (2019)]

[1]Keven Bosa, Statistics Canada, R.H. Coats Building, 100 Tunney's Pasture Driveway, Ottawa, Ontario, Canada, K1A 0T6, keven.bosa@statcan.gc.ca

[2]Kenneth C.K. Chu, Statistics Canada, R.H. Coats Building, 100 Tunney's Pasture Driveway, Ottawa, Ontario, Canada, K1A 0T6, kenneth.chu@statcan.gc.ca

is well suited to avoid the type of temporal data leakage mentioned above that may result from indiscriminate use of conventional validation protocols in official statistics production settings.

In this article, we apply RWFV for validation (i.e., hyperparameter tuning) to train a crop yield prediction model using data from the Field Crop Reporting Series (FCRS) [Statistics Canada (2021)], and we demonstrate the use of historical prediction error series for testing (i.e., generalization error estimation).

In Section 2, we give background information about the FCRS and the FCRS data structure. In Section 3, we describe in detail the RWFV protocol as well as how to compute the historical prediction error series. In Section 4, we give the definition of an aggregate error for quantifying the performance of a prediction strategy for a given validation year or production cycle. We then compare the performance of a machine learning prediction technique (XGBoost [Chen and Guestrin (2016)]) and that of an existing model-based statistical technique via historical prediction error series.

2 PROJECT DESCRIPTION

We conducted a project to evaluate the potential of applying supervised machine learning techniques for crop yield prediction for the FCRS by comparing their performances against that of the existing prediction model.

2.1 Field Crop Reporting Series – background

Traditionally, the FCRS publishes annual crop yield estimates at the end of each reference year (shortly after harvest). In addition, full-year crop yield predictions are published several times during the reference year. Farms are contacted in March, June, July, September and November for data collection.

In 2019, for the province of Manitoba, a model-based method – essentially, variable selection via LASSO (Least Absolute Shrinkage and Selection Operator), followed by robust linear regression – was introduced to generate the unit-level July predictions based on longitudinal satellite observations of local vegetation levels as well as region-level weather measurements. We shall refer to this method as LASSO-Robust in what follows. The use of a prediction model allowed the removal of the question about crop yield prediction from the Manitoba FCRS July questionnaire, thereby reducing response burden. The LASSO-Robust method was taken as the benchmark, against which machine learning techniques would be compared.

2.2 Field Crop Reporting Series – available data

Historical FCRS data from 2000 to 2017 for Manitoba were used for the project. Data from 2018 to 2020 were withheld for downstream independent testing by subject matter experts.

The data are in tabular format, with 380,180 rows and 293 columns (variables). Each row contains data from a parcel of land (160 acres) in Manitoba for a particular year; in other words, the primary key of the data table is (Year, Parcel ID). Across years, the number of parcels ranges from about 13,000 to 26,000. The target variable for prediction is Yield, which by definition is crop production per unit harvested area, measured in number of bushels per acre. The variables divide into the following groups:

- crop insurance data: crop yield and insured crop type for each (year, parcel)
- geographical covariates: longitude, latitude, Census Agricultural Region (CAR), eco-region, etc.
- operational data: operator ID, seeded area, harvested area, etc.
- weekly (weeks 16 to 31) parcel-level normalized difference vegetation index (NDVI): variables derived from satellite measurements of the amount of growth of local vegetation.
- weekly (weeks 18 to 31) CAR-level weather data (e.g., total precipitation, average soil water content).
- certain derived variables from the NDVI and weather time series (e.g., totals, maxima, rolling averages)

2.3 Key challenge and main contribution

The key challenges of the project were (1) how to conduct the computational experiments in a way that reflects what would be operationally feasible within the FCRS production environment, and (2) how to evaluate any candidate prediction method meaningfully given the FCRS production context.

The main contribution of the project is the adaptation of rolling window forward validation (RWFV) for hyperparameter tuning, and historical mock production error series as testing error (generalization error estimates). RWFV is a special case of forward validation [Schnaubelt (2019)], a family of validation protocols designed to prevent temporal data leakage for supervised learning on time series data.

3 PROPOSED VALIDATION PROTOCOL & TESTING PERFORMANCE METRIC

3.1 Validation via Rolling Window Forward Validation (RWFV)

We illustrate RWFV with an example. Suppose we are training a prediction model for deployment in the 2021 production cycle, and that data are available up to year 2020. The following schematic illustrates an RWFV scheme with a *training window* of five years, and a *validation window* of three years:

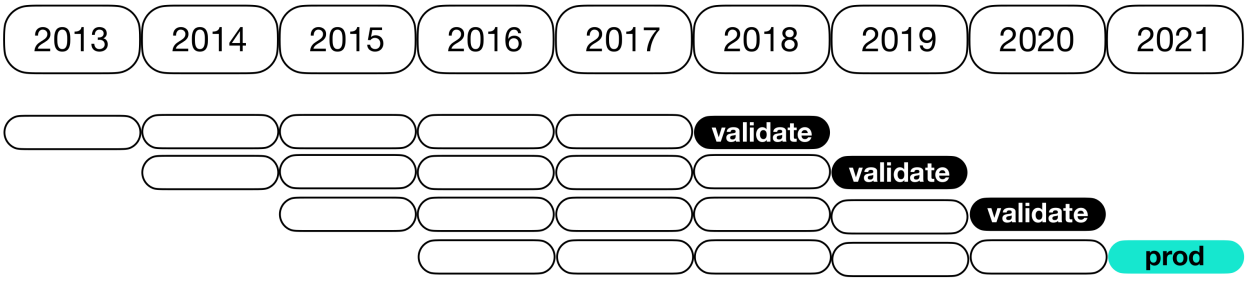

Figure (1) The blue box represents the 2021 production cycle and the five white boxes to its left correspond to the fact that a training window of five years is being used. This means that the training data for the 2021 production cycle will be those from the five years strictly and immediately prior (2016 to 2020). For validation, or hyperparameter tuning, for the 2021 production cycle, the three black boxes correspond to our choice that the validation window is three years.

The RWFV protocol is used to choose the "optimal" configuration from the hyperparameter search space, as follows:

(i) Fix temporarily an arbitrary candidate hyperparameter configuration from the search space.

(ii) Use that configuration to train a model for validation year 2020, using training data from the strictly preceding five years: 2015 to 2019.

(iii) Use that resulting trained model to make predictions for the validation year 2020. Compute accordingly the unit-level prediction errors for 2020.

(iv) Aggregate the unit-level prediction errors down to an appropriate numeric *aggregate error*.
(For the FCRS project, we chose this to be the *harvested-area-weighted relative error* (HAWRE); see Section 4.1.)

(v) Repeat for the rest of the validation years, i.e., 2018 and 2019.

(vi) Compute the average of the aggregate errors across the validation years 2018, 2019 and 2020, and call it the *RWFV error* for the temporarily fixed hyperparameter configuration.

(vii) Repeat for all candidate hyperparameter configurations in the hyperparameter search space to obtain their respective RWFV errors.

(viii) The optimized hyperparameter configuration to actually be deployed in production is the one that yields the smallest RWFV error.

Once we have determined the optimal hyperparameter configuration as described above, we use it to train a prediction model based on training data from 2016, 2017, . . . , 2020. This trained model is then deployed for the 2021 production cycle.

We emphasize again that the above protocol respects the operational constraint that, for the 2021 production cycle, the trained prediction model must have been trained and validated on data from strictly preceding years; in other words, the protocol prevents temporal data leakage.

3.2 Testing via aggregate errors of consecutive mock production cycles

Recall that testing in the context of supervised machine learning refers to estimation of the generalization error of a trained model, which is the expected error of that trained model when it is applied in practice to "never-before-seen" data (i.e., data that were not used during the model building procedure).

For crop yield prediction for the FCRS, the relevant (generalization) error to estimate is thus model prediction error for the current production cycle, using a prediction model trained on data strictly from past production cycles, as this faithfully reflects what is implementable, and what will be implemented, in the official statistics production setting of the FCRS.

We thus evaluate a candidate *prediction strategy* by computing its series of prediction errors – what we called the aggregate error in (iv) in general, and it will be the HAWRE for the FCRS project in particular – that would have resulted had it been deployed in past production cycles, where a prediction strategy refers to a procedure that produces a trained model based on training/validation data. Figure 2 illustrates how to produce the aggregate error series for a number of consecutive mock production cycles that use RWFV as validation (hyperparameter tuning) protocol.

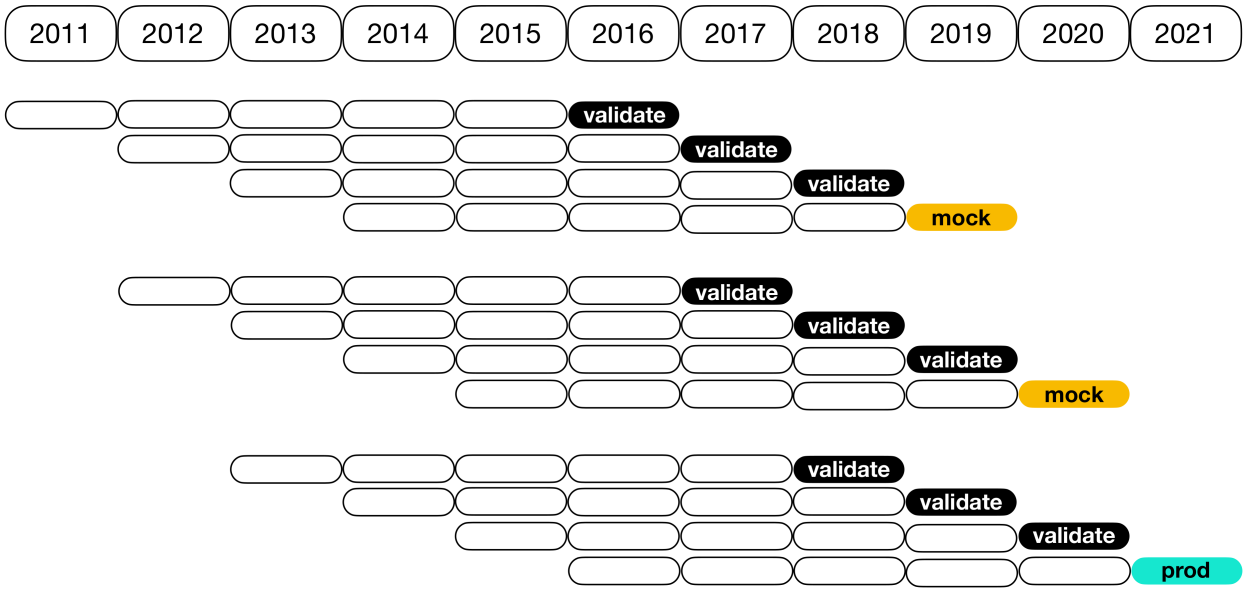

Figure (2) For each past mock production cycle (represented by an orange box), we repeat the model building procedure described in the preceding section (for the current production cycle, represented by the blue box). The difference now is the following: for the current production cycle (blue box, year 2021), it is NOT yet possible to compute the production/prediction errors at time of crop yield prediction (in July 2021) since the 2021 growing season has not ended. However, for the past mock production cycles (the orange boxes), it is possible.

The procedure illustrated in Figure 2 allows us to compute the series of prediction errors for past production cycles that would have resulted had the candidate prediction strategy been deployed during those past production cycles. We take this series of prediction errors as testing error estimates, i.e., generalization error estimates.

4 RESULTS

Our goal was to devise a prediction/model building strategy for crop yield prediction for the FCRS, and evaluate it by estimating its error rate in the FCRS production setting. Note that the strategy itself must be actually implementable within the FCRS production setting; in particular, it must not cause temporal data leakage. The performance metric must convincingly reflect how the strategy would perform in the actual production setting.

We decided to proceed with the following:

- Prediction/model building strategy: Use XGBoost(Linear) as the core prediction technique. Use RWFV with 5-year training window and 5-year validation window as validation (hyperparameter tuning) protocol. Use harvested-area-weighted relative error (HAWRE) as the aggregate error for each (prediction method,

hyperparameter configuration, year); see Section 4.1. The RWFV error for each (prediction method, hyperparameter configuration) is then the across-validation-years average of its HAWRE's.

- Model testing strategy: Use the historical mock production HAWRE series as testing performance metric for each given prediction/model building strategy, which computes the actual series of HAWRE's over consecutive production cycles that would have resulted had that strategy been deployed in the past.

We next give the definition of HAWRE in Section 4.1. We then compare the XGBoost/RWFV and the baseline LASSO-Robust strategies by examining their respective historical mock production HAWRE series.

4.1 Aggregate error – harvested-area-weighted relative error (HAWRE)

We now specify the *aggregate error* mentioned in (iv) that we used for the project. It is the weighted average of (eco-region, crop type)-specific relative errors of predicted crop productions, weighted by harvested area. The predicted crop productions were computed within eco-region and crop type, since eco-region and crop type are strong covariates that influence crop production. Harvested areas were used as "importance" weights that were independent of crop type.

Let $(\,m\,,h\,)$ represent a combination of (prediction method, hyperparameter configuration), and $(\,y,r,c\,)$ a combination of (year, eco-region, crop type). Then, the (actual) crop production $P_{r,c}^{(y)}$ for (y,r,c) and predicted crop production $\widehat{P}_{r,c}^{(m,h,y)}$ for (y,r,c) and (m,h) are respectively given by:

$$P_{r,c}^{(y)} \;:=\; \sum_{l\,\in\,(y,r,c)} \begin{pmatrix}\text{crop}\\ \text{yield}\end{pmatrix}_l \times \begin{pmatrix}\text{harvested}\\ \text{area}\end{pmatrix}_l , \qquad \widehat{P}_{r,c}^{(m,h,y)} \;:=\; \sum_{l\,\in\,(y,r,c)} \begin{pmatrix}(m,h)\text{-predicted}\\ \text{crop yield}\end{pmatrix}_l \times \begin{pmatrix}\text{harvested}\\ \text{area}\end{pmatrix}_l ,$$

where the above summations are taken over all parcels l within ecoregion r growing crop type c during year y. The crop-production-induced relative error $\varepsilon_{r,c}^{(m,h,y)}$ for (m,h) and (y,r,c), the harvested area $A_{r,c}^{(y)}$ and harvested-area-induced weight $w_{r,c}^{(y)}$ for (y,r,c) are respectively given by:

$$\varepsilon_{r,c}^{(m,h,y)} \;:=\; \Big|\, \widehat{P}_{r,c}^{(m,h,y)} \,-\, P_{r,c}^{(y)} \,\Big| \Big/ P_{r,c}^{(y)} , \qquad A_{r,c}^{(y)} \;:=\; \sum_{l\,\in\,(y,r,c)} \begin{pmatrix}\text{harvested}\\ \text{area}\end{pmatrix}_l , \qquad w_{r,c}^{(y)} \;:=\; A_{r,c}^{(y)} \Big/ \sum_{(\xi,\zeta)} A_{\xi,\zeta}^{(y)}$$

Finally, the harvested-area-weighted relative error for (m,h,y) is given by:

$$\text{HAWRE}(m,h,y) \;\;:=\;\; \sum_{(r,c)} w_{r,c}^{(y)} \cdot \varepsilon_{r,c}^{(m,h,y)} ,$$

which is taken as the aggregate error for (m,h,y), as first mentioned in (iv).

4.2 Testing errors – HAWRE's of consecutive mock production cycles

We experimented with a number of prediction techniques, including: random forests, support vector machines, elastic-net regularized generalized linear models, and multilayer perceptrons. Accuracy and computation time considerations led us to focus attention on XGBoost with linear base learner. We report only results for XGBoost(Linear) in what follows.

We used grid search for XGBoost(Linear) hyperparameter tuning, with the following search grid:

$$(\alpha,\lambda) \;\in\; \Theta \;:=\; \Big\{\, 7,9,11,\ldots,33 \,\Big\} \times \Big\{\, 7,9,11,\ldots,33 \,\Big\} ,$$

where α is the L_1 regularization term on weights and λ is the L_2 regularization term on weights. The full search grid therefore had $196 = 14 \times 14$ XGBoost(Linear) hyperparameter configurations.

We computed $\text{HAWRE}\big(\text{XGBoost(Linear)}, h, y\big)$, for each $h = (\alpha,\lambda) \in \Theta$ and each (validation) year $y \in \{2005, 2006, \ldots, 2017\}$. For each mock production cycle $y \in \{2010,\ldots,2017\}$, the optimal hyperparameter configuration was then chosen as $h^*(y) \in \underset{h\in\Theta}{\text{argmin}} \Big\{ \text{RWFV-err}\big(\text{XGBoost(Linear)}, h, y\big) \Big\}$, and the test error of the XGBoost/RWFV prediction strategy was taken to be $\text{HAWRE}\big(\text{XGBoost(Linear)}, h^*(y), y\big)$, where

RWFV-err$\Big($XGBoost(Linear), $h, y\Big) := \frac{1}{5} \cdot \sum_{\xi=y-5}^{y-1}$ HAWRE$\Big($XGBoost(Linear), $h, \xi\Big)$. The results are illustrated in Figure 3, which shows that the XGBoost/RWFV prediction strategy exhibited smaller HAWRE's than the baseline LASSO-Robust method, consistently over consecutive historical production cycles.

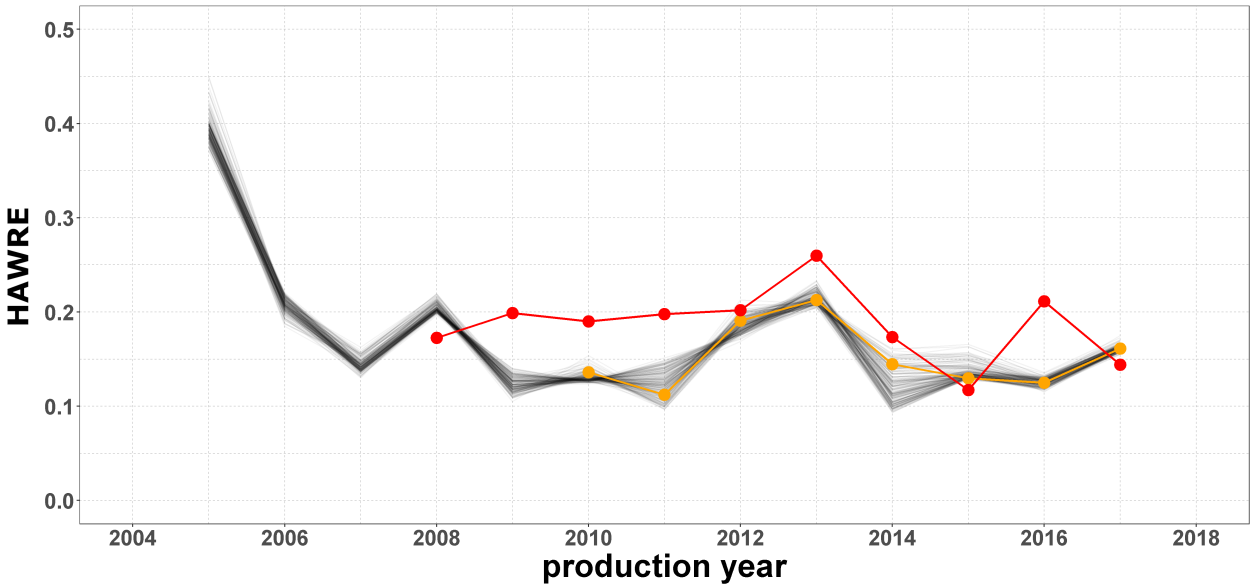

Figure (3) Each of the 196 gray lines illustrates the HAWRE series for a candidate hyperparameter configuration $h = (\alpha, \lambda) \in \Theta$. The orange line illustrates the XGBoost/RWFV strategy testing error series, i.e., HAWRE(XGBoost(Linear), $h^*(y), y$), while the red line illustrates that of the baseline LASSO-Robust model, i.e., HAWRE(LASSO-Robust, $-, y$).

5 CONCLUSIONS

For comparing different techniques for crop yield prediction within the context of the FCRS, we chose not to use a more conventional validation method such as hold-out or k-fold cross-validation, nor a generic generalization error estimate such as prediction error on a testing data set kept aside at the beginning.

These decisions were taken based on our determination that our proposed validation protocol (RWFV) and choice of generalization error estimates (mock production cycle prediction error series) must (a) avert the subtle pitfall of temporal data leakage, and (b) give error estimates that are more relevant with respect to the production context of the FCRS.

We emphasize, however, that the validation and testing protocols we discussed in this article are by no means the only ones that can ensure (a) and (b). Lastly, survey methodologists and machine learning practitioners are encouraged to evaluate carefully whether generic validation protocols or generalization error estimates are indeed appropriate for their use cases at hand, and if not, seek alternatives that are more relevant and methodologically sound within the given context.

REFERENCES

Chen, T. and Guestrin, C. (2016). XGBoost, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* .
URL: `http://dx.doi.org/10.1145/2939672.2939785`

Schnaubelt, M. (2019). A comparison of machine learning model validation schemes for non-stationary time series data, *FAU Discussion Papers in Economics* **11**.

Statistics Canada (2021). Field Crop Reporting Series.
URL: `https://www23.statcan.gc.ca/imdb/p2SV.pl?Function=getSurvey&SDDS=3401`