Imputation in the sample survey on participation of Polish residents in trips

Organisation: Statistical Office in Rzeszów, Statistics Poland

Author(s): Sebastian Wójcik

Date: 12.06.2020

Version: 2.0

1. Background and why and how this study was initiated

(as many sentences as necessary, as few as possible)

Statistics Poland conducts a quarterly-based households sample survey on participation of Polish residents in trips. With Big Data sources we are able to estimate the number of trips on the low level of aggregation. To use these results in regular production we need to estimate microaggregates of expenditures by expenditure category and country. Various methods are compared how they deal with data imputation on the level of microaggregates.

2. Data

2.1 Input Data (short description)

Quarterly sample survey on participation of Polish residents in trips. Dataset contains 22.3 ths. of records covering three consecutive years 2016-2018. Five continuous variables to predict with a use of 10 categorical variables were selected by specialist in the field of tourism. For each predicted variable, the training set was subsetted for relevant cases and variables.

2.2 Data Preparation

Data cleaning has been already carried out by specialist in the field of tourism. Databases for each quarter were merged, one grouping variable was added. Categorical variables were modelled with dummy variables.

2.3 Feature Selection

Features were selected by specialist in the field of tourism.

2.4 Output data

Set of predictions and their statistics (RMSE, R2, MAE, MAPE) for each tested model with tuned hyperparameters and for each dataset.

3. Machine Learning Solution

3.1 Models tried

Non-Machine Learning models:

- Linear Model (OLS)
- General Linear Model (GLS)
- Robust Linear Model
- LARS
- Predictive Mean Matching (used as a single imputation method, not as a part of MICE algorithm)

Machine Learning models:

- CART
- Random Forest
- Optimal Weighted Nearest Neighbour
- Support Vector Machine (linear and radial kernel)

3.2 Model(s) finally selected and the criterion

(i.e.: which model was why seen being the best?)

Based on five sets of results, it was hard to pick just a one winner. In final, Optimal Weighted Nearest Neighbour was selected. It was in the top two models in terms of RMSE and R2 as well it was in the top five models in terms of MAE and MAPE. CART model achieved the best accuracy with respect to RMSE in three out of five cases. Nevertheless, in other two cases it had a mediocre accuracy. Surprisingly, SVM with linear kernel achieved the best accuracy with respect to MAE and MAPE in all five cases but on the other hand its RMSE and R2 were poor also in all five cases. In fact, except aforementioned models, all of the models achieved very similar results with respect to all accuracy metrics.

3.3 Hardware used

Intel Core i7-4770, 2x3.40 GHz, 64bit

16 GB RAM

3.4 Runtime to train the model

Runtime was contingent upon the number of tested sets of hyperparameters, bootstrap samples and form of the training datasets as well as on the size of the dataset itself. For each method, an R function used is presented in parentheses. Some implementations calculates more intermediate statistics than the others what affects a runtime. For instance, Random Forest implemented in *randomForest* function carries out a selection of some hyperparameters.

It turned out that some functions could deal with factor variables. In such case the **short** version of the dataset was used. Elsewhere, factor variables needed to be converted into dummy variables and the **long** version of the dataset was used.

The table below presents the size of the datasets used.

Dataset	No. records (cases)	No. variables
Expenditures for accommodation	10491	10 (short version), 54 (long version)
Expenditures for restaurants and café	16209	
Expenditures for transport	17123	
Expenditures for commodities	17565	
Other expenditures	7993	

Selection of the optimal set of hyperparameters was based on 200 bootstrap samples. Number of the set of hyperparameters tested with bootstrap method varied. Hence, it must be taken into account when comparing the runtime of the models.

The table below presents the bootstrapping setup.

Model (R function)	No. bootstrap samples	Type of the dataset	No. sets of hyperparameters	Tuned parameters
Linear Model OLS (lm)	200	long	2	constant
General Linear Model GLS (glm)	200	long	2	error distribution

Robust Linear Model (rlm)	200	long	2	constant
LARS (lar)	200	long	54	norm of vector of parameters
Predictive Mean Matching (pmm)	200	short	3	m
CART (rpart)	200	short	6	cp
Random Forest (randomForest)	200	short	5	ntree, cp
Optimal Weighted Nearest Neighbour (kknn)	200	short	15	k
Support Vector Machine (svm)	200	short	2	kernel

The table below presents the runtime with respect to five datasets.

Model (R function)	Accommodation	Restaurants and café	Transport	Commodities	Other
Linear Model OLS (lm)	23.081 secs	38.157 secs	37.128 secs	46.569 secs	21.642 secs
General Linear Model GLS (glm)	34.786 secs	60.435 secs	56.895 secs	1.119 mins	26.607 secs
Robust Linear Model (rlm)	2 mins 12.7secs	3.119 mins	6.728 mins	39.422 secs	1.121 mins
LARS (lar)	36.947 secs	65.845 secs	67.782 secs	1.186 mins	31.645 secs
Predictive Mean Matching (pmm)	1 hour 52.57 mins	46.106 mins	50.516 mins	48.456 mins	18.673 mins
CART (rpart)	3 mins 49.92 secs	3.800 mins	3.788 mins	4.321 mins	1.728 mins
Random Forest (randomForest)	3 hours 10.56 mins	5.702 h	6.482 h	6.199 h	1.561 h
Optimal Weighted Nearest Neighbour (kknn)	15.837 mins	30.833 mins	29.937 mins	36.647 mins	10.023 mins
Support Vector Machine (svm)	1 hour 51.041 mins	6.622h	4.67h	5.742 h	1.020 h

4. Results

(e. g. in terms of RMSE, MAE, distributional accuracy [*], F1 (micro or macro), recall, accuracy, (threshold,) ..., perhaps as a table for different situations (if available))

[*]: If used: How did you measure distributional accuracy? By proportions, moments, quantiles, correlations,

Mixture of Bootstrapping and K-fold Cross Validation was used to find the optimal set of hyperparameters with respect to RMSE for each tested model. Several accuracy measures were calculated for each model with the optimal set of hyperparameters on the complete dataset. Tuning was carried out in the following way:

- Draw B samples without replacement with size amounting 90% of size of the dataset.
- Train model with a given set of hyperparameters
- Make predictions on the 10% remaining cases.
- Calculate R2, RMSE, MAPE and MAE
- Average R2, RMSE, MAPE and MAE over all B draws.

The best set of hyperparameters is used to build model and make the final predictions on the whole data set. The next tables present the results for the tested models.

I Dataset pertaining to expenditures for accommodation

Method (R function)	MAE	MAPE	RMSE	R2
Linear Model OLS (lm)	92.62	1.400	183.05	0.225
General Linear Model GLS (glm)	92.62	1.400	183.05	0.225
Robust Linear Model (rlm)	85.64	1.040	188.56	0.216
LARS (lar)	93.26	1.476	184.35	0.217
Predictive Mean Matching (pmm)	95.27	1.504	184.62	0.214
CART (rpart)	94.15	1.463	185.79	0.203
Random Forest (randomForest)	92.01	1.350	185.13	0.212
Optimal Weighted Nearest Neighbour (kknn)	86.15	1.240	171.13	0.329
Support Vector Machine (svm) radial kernel	91.25	0.956	202.84	0.172
Support Vector Machine (svm) linear kernel	86.03	0.938	192.45	0.203

In a case of Optimal Weighted Nearest Neighbour, predictions were slightly biased - mean prediction was 2% higher than true mean. The same situation occurred with predictive mean matching – 2,9% bias. Other methods did not produced significantly biased predictions (bias up to ±0.5%). Based on Kolmogorov-Smirnoff test, the distribution of predictions significantly differed from the true one for every model (p-value < 10^{-16}).

II Dataset pertaining to expenditures for restaurants and café

Method (R function)	MAE	MAPE	RMSE	R2
Linear Model OLS (lm)	58.28	1.286	125.44	0.099
General Linear Model GLS (glm)	58.28	1.286	125.44	0.099
Robust Linear Model (rlm)	53.70	0.977	127.66	0.093
LARS (lar)	58.56	1.341	126.10	0.092
Predictive Mean Matching (pmm)	61.38	1.286	128.32	0.068
CART (rpart)	54.72	1.207	112.34	0.278
Random Forest (randomForest)	58.13	1.295	126.97	0.084
Optimal Weighted Nearest Neighbour (kknn)	56.27	1.218	116.84	0.225
Support Vector Machine (svm) radial kernel	54.53	0.893	131.65	0.080
Support Vector Machine (svm) linear kernel	53.21	0.854	129.45	0.089

SVM and Robust Linear Model were harshly biased – mean prediction was lower by 19%-32% from the true mean. In a case of other models, bias was up to ±3%). Based on Kolmogorov-Smirnoff test, the distribution of predictions significantly differed from the true one for every model (p-value < 10^{-16}).

III Dataset pertaining to expenditures for transport

Method (R function)	MAE	MAPE	RMSE	R2
Linear Model OLS (lm)	440.05	1.483	853.00	0.418
General Linear Model GLS (glm)	440.05	1.483	853.00	0.418
Robust Linear Model (rlm)	413.35	1.208	874.08	0.402
LARS (lar)	448.23	1.559	892.36	0.368
Predictive Mean Matching (pmm)	434.59	1.666	841.59	0.434
CART (rpart)	373.62	1.149	696.27	0.612
Random Forest (randomForest)	407.10	1.259	820.22	0.462
Optimal Weighted Nearest Neighbour (kknn)	393.03	1.244	774.25	0.533
Support Vector Machine (svm) radial kernel	519.22	1.384	1149.6	0.057
Support Vector Machine (svm) linear kernel	458.63	0.912	1043.7	0.272

SVM and Robust Linear Model were harshly biased with similar magnitude as for the expenditures for restaurants and cafés. Optimal Weighted Nearest Neighbour also produced biased predictions - mean prediction was 6% lower than true mean. Other methods did not produced significantly biased predictions (bias up to ±2.2%). Based on Kolmogorov-Smirnoff test, the distribution of predictions significantly differed from the true one for every model (p-value < 10^{-16}).

IV Dataset pertaining to expenditures for commodities

Method (R function)	MAE	MAPE	RMSE	R2
Linear Model OLS (lm)	69.97	3.527	175.21	0.025
General Linear Model GLS (glm)	69.97	3.527	175.21	0.025
Robust Linear Model (rlm)	68.89	3.485	175.05	0.024
LARS (lar)	69.85	3.539	175.24	0.024
Predictive Mean Matching (pmm)	76.52	4.034	178.59	0.008
CART (rpart)	64.04	2.959	153.69	0.249
Random Forest (randomForest)	70.80	3.439	178.63	0.013
Optimal Weighted Nearest Neighbour (kknn)	68.73	3.238	159.12	0.206
Support Vector Machine (svm) radial kernel	59.94	1.742	181.22	0.005
Support Vector Machine (svm) linear kernel	59.74	1.711	180.87	0.007

SVM predictions were harshly biased (-44%) as well as Predictive Mean Matching (+11%). This time Robust Linear Model and other methods did not produce significantly biased predictions (bias up to ±0.5%). Based on Kolmogorov-Smirnoff test, the distribution of predictions significantly differed from the true one for every model (p-value < 10^{-16}).

V Dataset pertaining to other expenditures.

Method (R function)	MAE	MAPE	RMSE	R2
Linear Model OLS (lm)	53.84	2.857	126.02	0.046
General Linear Model GLS (glm)	53.84	2.587	126.02	0.046
Robust Linear Model (rlm)	46.94	1.669	128.82	0.034
LARS (lar)	53.86	2.747	126.61	0.038
Predictive Mean Matching (pmm)	56.29	2.897	126.50	0.040
CART (rpart)	52.13	2.430	118.99	0.149
Random Forest (randomForest)	54.89	2.532	128.02	0.036
Optimal Weighted Nearest Neighbour (kknn)	51.10	2.284	119.30	0.146
Support Vector Machine (svm) radial kernel	47.09	1.483	131.33	0.021
Support Vector Machine (svm) linear kernel	46.38	1.279	130.68	0.029

Bias of the mean prediction is very similar to the bias for the expenditures for restaurants and cafés. The distribution of predictions significantly differed from the true one for every model (p-value < 10^{-16}).

5. Code/programming language

(e.g. the Python code is stored in GitHub)

R language. Code and input data are stored on local servers.

6. Evolution of this study inside the organisation

(e. g.: Collaboration within the organisation? Has this study advanced ML within the organisation?)

ML is still on the stage of seed. It has been tested just by one of the divisions in the statistical office.

7. Is it a proof of concept or is it already used in production?

(If it is a proof of concept: Was it successful? How will its results prospectively be used in the future?)

It is a proof of concept. It must be embedded in the methodology of the survey on trips and consulted with all institutions (Polish National Bank and Ministry of Development) participating in the survey. Further, R script implementing machine learning solution must be linked to the software already used in production.

7.1 What is now doable which was not doable before?

(**e. g.: Is something faster or cheaper or more exact? What is the added value using this machine learning?**)

All tested machine learning methods produced plausible predictions. Traditional regression models produced negative values except LARS which gives several sets of prediction and the final set can be selected with respect to non-negativity condition. Also, the machine learning methods can deal with “singular” problems since they do not take into account any correlations.

7.2 Is there already a roadmap/service journey available how to implement this?

(as many sentences as necessary, as few as possible)

Not yet.

7.3 Who are the stakeholders?

Statistics Poland, Polish National Bank and Ministry of Development.

7.4 Fall Back

Is a fall back plan in place or planned to mitigate the risk of the ML solution failing in production? Will there be resource left in place to go back to e.g. manual imputation or the use of rule-based scripts? (as many sentences as necessary, as few as possible)

Presented PoC is a part of new methodology stemming from the access to new data sources. Thus, there is nothing to go back to.

7.5 Robustness

What fail checks are in place or planned to ensure that the ML solution is consistently meeting or exceeding the set gold standard? (as many sentences as necessary, as few as possible)

It is very early stage of "discovering" ML solution within our institution. No fails checks are planned at this moment.

8. Conclusions and lessons learned

Machine learning methods are much more powerful than traditional models and they can easily overfit to the dataset. Therefore, estimating the out-of-bag error is one of the relevant way to compare various methods by bootstrapping or cross validation. Nevertheless, the results of e.g. k-fold cross validation may be misleading. Based on empirical studies, when k-fold cross validation was run several times, it lead to confusion about that which model is the optimal model. Thus, bootstrapping is more reliable method for model selection but at the same time is more time-consuming.

It is worth to notice that the model selection cannot be based just on the accuracy measures e.g. MAPE, RMSE etc. without checking distributional accuracy including biasedness. When data is imputed it is hard to expect to impute data perfectly on the individual level. It may be expected to retrieve a true mean level of imputed data with respect to some strata. Then, on average, totals can be calculated correctly. Simulations revealed that SVM produced good predictions in terms of MAPE. But these predictions were harshly biased (30%-40% downward). In a result, estimated totals for true and for imputed values differed significantly.

9. Potential organisation risk if ML solution not implemented

(as many sentences as necessary, as few as possible)

ML solution is an option thus there is no organisation risk if it is not implemented.

10. Has there been collaboration with other NSIs, universities, etc?

(yes/no, if yes: which ones?)

No.

11. Next Steps

(as many sentences as necessary, as few as possible)

In our institution there is a need to develop a relevant knowledge and skills to understand the process of building and testing ML models.