A Longitudinal Framework for Predicting Nonresponse in Panel Surveys

Christoph Kern[1] Bernd Weiß[2] Jan-Philipp Kolb[2]

[1]University of Mannheim

[2]GESIS - Leibniz Institute for the Social Sciences

ESRA Conference 18.07.2019

Introduction

Motivation

- Panel studies often suffer from drop-outs over time
  - Biased estimates, decreasing sample size
- Prediction
  - Recent work studies the usage of machine learning (ML) to predict nonresponse (Klausch 2017; Lugtig and Blom 2018; Kern et al. 2019)
- Adaptive designs
  - May benefit from prediction perspective due to accurate targeting

Objective: Extend ML approach to account for longitudinal data structure

- Train and test prediction models with multiple panel waves
- Evaluate potential intervention based on prediction models

Data

GESIS Panel[1]

- Probability-based mixed-mode panel of the general population in Germany
- Recruitment in 2013, bi-monthly surveys since 2014 ($\sim$4900 panelists)
- $\sim$20min each wave, includes external studies and longitudinal core study
- Online (web surveys) and offline (mail) mode
  - About 62% online and 38% offline respondents

$\rightarrow$ **Outcome: Non-participation in (each) next wave**

- Complete or partial interview with sufficient information (0) vs. else (1)
- Sample: Excluding "ineligible" panelists per wave

[1] https://www.gesis.org/en/gesis-panel/

Features for each wave

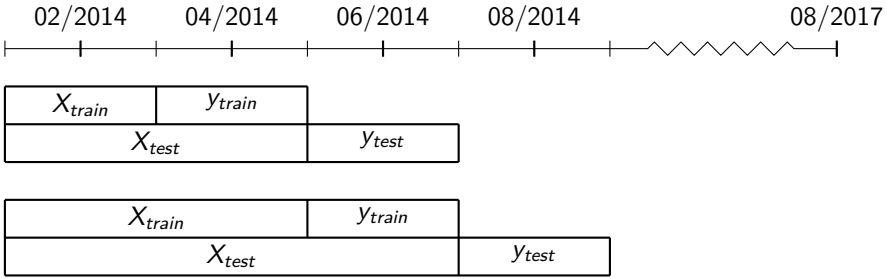
- Block I: Time-invariant
  - Respondent/ socio-demographic characteristics from welcome survey
  - Survey cooperation in welcome survey
- Block II: Time-variant
  - Response status, survey evaluation and participation in last wave
- Block III: Time-variant (aggregated)
  - Response status, survey evaluation and participation over the last three waves
- Block IV: Time-variant (aggregated)
  - Response status, survey evaluation and participation over all previous waves

→ Feature group strategies: `all`, `leave-one-in`

Temporal CV

Longitudinal configuration

- Compare methods/ performance by repeatedly mimicking usage of model in real world
- Temporal Cross-Validation via `triage (Python)`[2]



→ 20 train and 20 test matrices

[2] `https://github.com/dssg/triage`

Methods

- Penalized Logistic Regression
  - Logit regression plus lasso/ ridge penalty on model complexity (Tibshirani 1996)
- Decision Trees
  - Split predictor space into subregions τ_m with associated constants γ_m (Breiman et al. 1984)

$$\mathcal{T}(x;\Theta) = \sum_{m=1}^{M} \gamma_m I(x \in \tau_m)$$

- Random Forest, ExtraTrees
  - Grow an ensemble of decorrelated trees (Breiman 2001, Geurts et al. 2006)

$$\hat{f}_B(x) = \frac{1}{B} \sum_{b=1}^{B} \mathcal{T}_b(x;\Theta_b)$$

- Extreme Gradient Boosting (XGBoost)
  - Build a sum-of-trees ensemble in a sequential manner (Chen and Guestrin 2016)

$$\hat{f}_T(x) = \sum_{t=1}^{T} \mathcal{T}_t(x;\Theta_t)$$

Tuning parameters

Table 1: Tuning grids

Method	Hyperparameter	Values
Logistic Regression	penalty	l1, l2
	C	0.05, 0.1, 1, 1000
Decision Trees	max_depth	3, 5, 10
	max_features	null, sqrt
Random Forest, Extra Trees	max_features	sqrt, log2
	min_samples_leaf	1, 10
	n_estimators	500
XGBoost	max_depth	3, 5, 10
	n_estimators	250, 500, 1000
	learning_rate	0.1, 0.05
	subsample	0.8

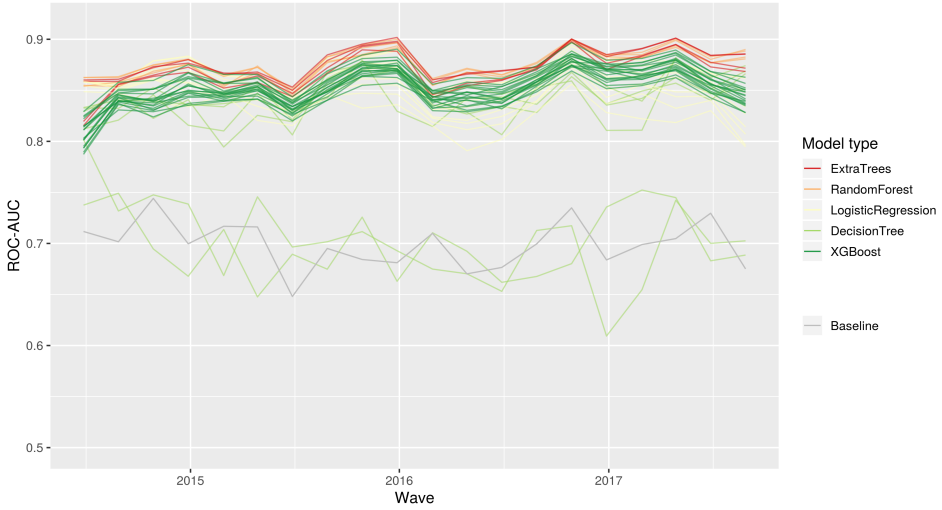
$\rightarrow$ 4000 models to train ($20 \times 5 \times 40$)

Model selection and evaluation

1. Find the optimal hyperparameter-feature group combination for each method over all waves
   - Highest mean ROC-AUC over time
2. Evaluate performance of selected/ best models in most recent wave
   - ROC, PR curves
3. Evaluate potential intervention in most recent wave

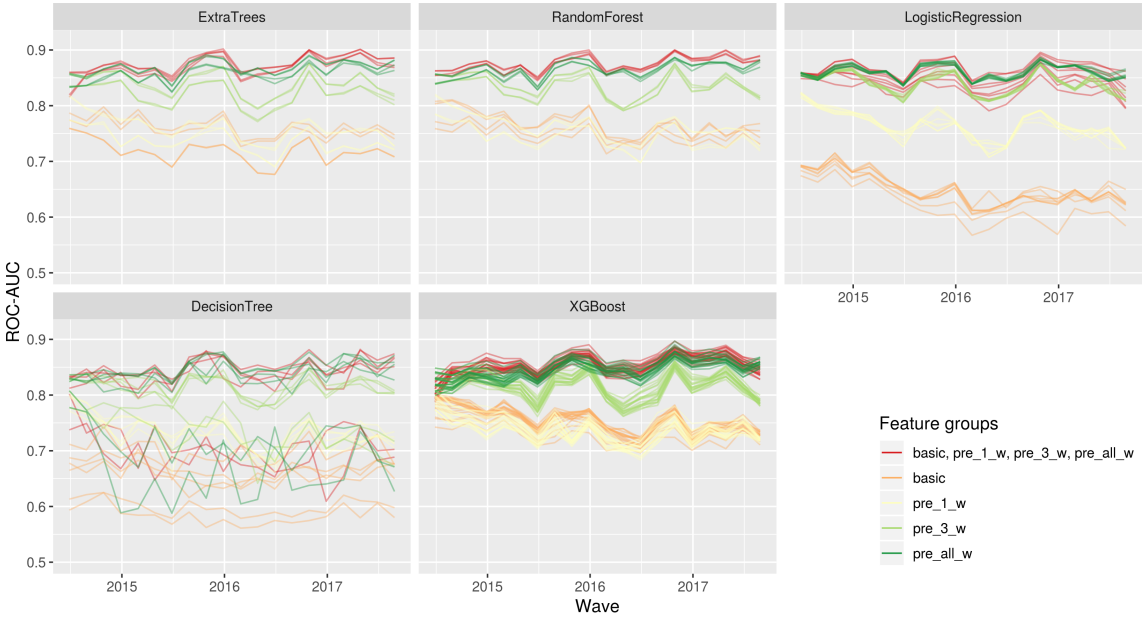
All waves

Figure 1: ROC-AUCs for all waves and models with all feature blocks[3]



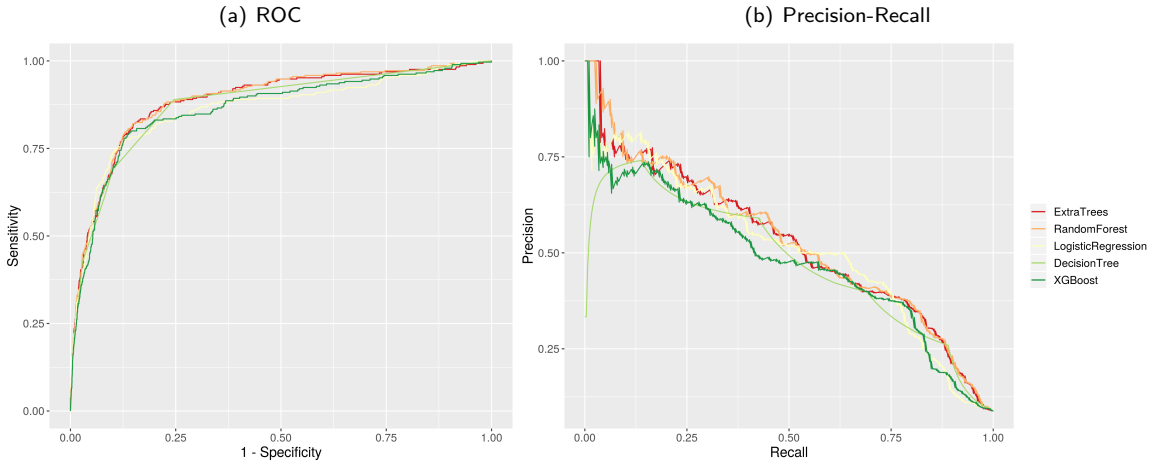
[3] https://ckern.shinyapps.io/predicting-nonresponse/

Figure 2: ROC-AUCs by model type and feature group



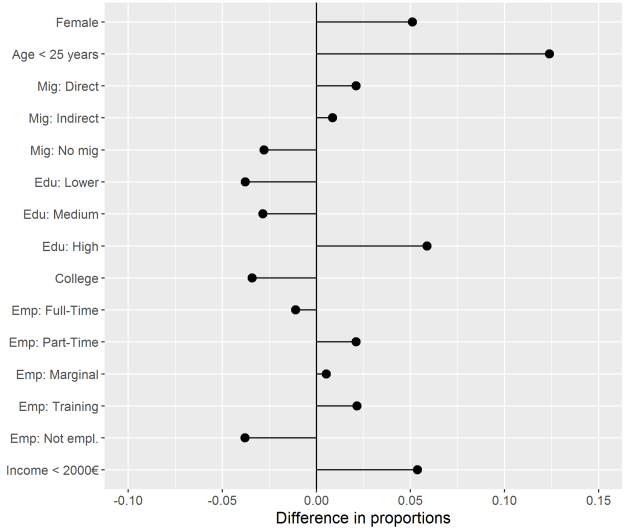
Most recent wave

Figure 3: Performance curves of best models for most recent test wave



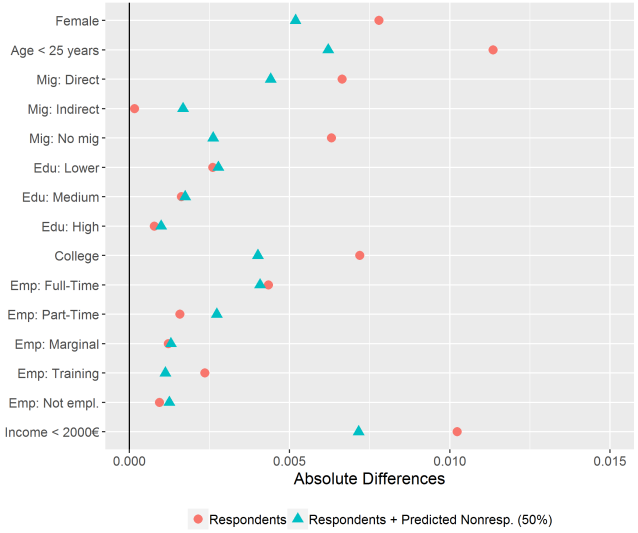
Most recent wave

Figure 4: Differences of high risk vs. low risk observations (top 10%, RF)



Most recent wave

Figure 5: Differences between active panel population, respondents and potential respondents (RF)



Discussion

- Investigating the usage of ML with panel data needs a longitudinal train-test setup
  - Repeatedly predict nonresponse in next wave using information from previous wave(s)
- GESIS Panel: General results
  - Promising prediction performance
  - Increased performance when aggregating features over multiple waves
  - Robust results over time with ExtraTrees, Random Forests
- GESIS Panel: Intervention
  - Targeting predicted nonrespondents may reduce systematic nonresponse

Contact: c.kern@uni-mannheim.de

References

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Monterey, CA: Brooks/Cole Publishing.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Technical report, `https://arxiv.org/abs/1603.02754`.

Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, 63(1):3–42.

Kern, C., Klausch, T., and Kreuter, F. (2019). Tree-based machine learning methods for survey research. *Survey Research Methods*, 13(1):73–93.

Klausch, T. (2017). Predicting panel attrition using panel-metadata: A machine learning approach. Paper presented at the ESRA Conference, Lisbon, Portugal.

Lugtig, P. and Blom, A. (2018). It's the process stupid! Using machine learning to understand the relation between paradata and panel dropout. Paper presented at the MOLS 2 Conference, Essex, Great Britain.

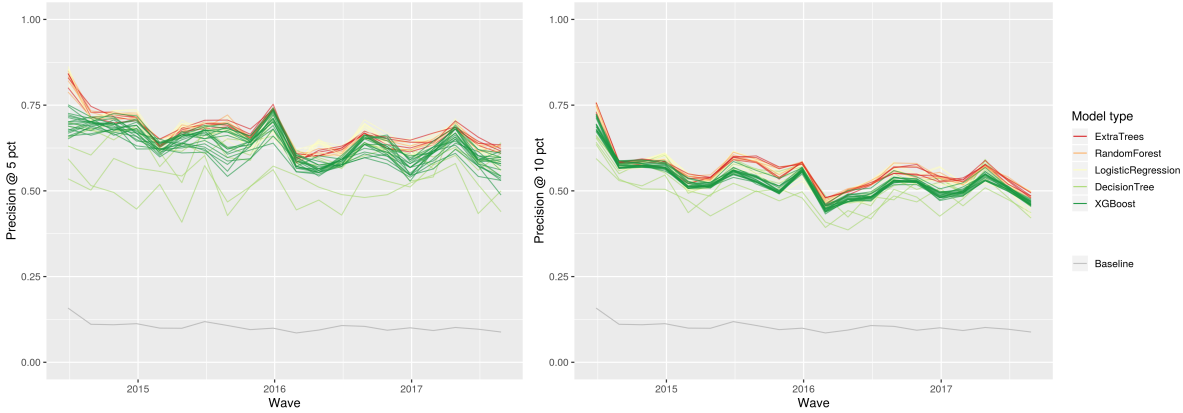
Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

Appendix

Figure 6: Precision at top K for all waves and models with all feature blocks

(a) Precision @ 5 pct

(b) Precision @ 10 pct



Appendix

Figure 7: Recall at top K for all waves and models with all feature blocks

(a) Recall @ 5 pct

(b) Recall @ 10 pct