

You Have More Data Than You Think: Optimizing Machine Learning in Tabular Social Science Datasets Through Augmentation and Linkage

Hanzhang Ren

ren1999@stanford.edu

Stanford University

Emily M. Cantrell

Princeton University

Research Article

Keywords: machine learning, data augmentation, feature engineering, sample size, longitudinal survey, fertility

Posted Date: August 7th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10528921/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

You Have More Data Than You Think: Optimizing Machine Learning in Tabular Social Science Datasets Through Augmentation and Linkage

Hanzhang Ren^{1*†} and Emily M. Cantrell^{2†}

¹Department of Sociology, Stanford University, 450 Jane Stanford Way,
Building 120, Stanford, CA 94305, USA.

²Department of Sociology, Princeton University, Wallace Hall,
Princeton, NJ 08544, USA.

*Corresponding author(s). E-mail(s): ren1999@stanford.edu;

Contributing authors: e.cantrell@princeton.edu;

[†]These authors contributed equally to this work.

Abstract

What strategies improve machine learning performance in tabular social science datasets? The Predicting Fertility (PreFer) data challenge invited dozens of researchers around the world to predict births and adoptions using Dutch survey data. Our model produced the most accurate predictions in the competition. This paper describes two strategies behind the model, in which we engineered the training data to create effectively “more” data. First, we outline a strategy we call “time-shift data augmentation,” which artificially increases the sample size by repurposing data across time periods. Second, we demonstrate “partner linkage,” which increases the number of features by linking data from each focal person with data from their spouse or cohabiting partner. Our findings provide a case study of how these two data engineering strategies improve predictive performance. Both strategies can be implemented on a wide range of social data, including longitudinal surveys and administrative records, and may improve predictive performance in a variety of settings.

Keywords: machine learning, data augmentation, feature engineering, sample size, longitudinal survey, fertility

1 Introduction

Machine learning offers new tools to improve our description, understanding, and forecasting of the social world by identifying complex interactions and nonlinear patterns [10, 12, 41]. However, even with increasingly sophisticated machine learning techniques, predictive performance remains low in many social science applications [20, 29, 31, 35, 45]. This is especially true when working with small sample sizes, as is common in many social science data sources such as surveys, experiments, and studies of small subgroups or rare events.

Given these limitations, what strategies can improve predictive accuracy in social science contexts? This question was at the core of the Predicting Fertility (PreFer) data challenge. In this international competition, social scientists and data scientists competed to predict births and adoptions using Dutch survey data and administrative register data [33–35]. Our models produced the most accurate predictions in both datasets. A key driver of this performance was the data engineering strategies we used to transform the available data into effectively “more” data: either by increasing the sample size, increasing the number of features, or both. In the survey dataset, we artificially increased the sample size through a strategy we call “time-shift data augmentation,” which repurposes observations from earlier years of the longitudinal data to create additional cases. In both the survey and register datasets, we increased the number of features by manipulating the data to extract additional information about spouses or cohabiting partners, a strategy we refer to as “partner linkage.”

Prior literature on the predictability of life outcomes decomposed prediction error into two sources. Our data engineering strategies address both. First, “learning error” is the gap between predictions and the true conditional mean for people who are identical according to the features observed in the data [24]. Learning error occurs when the model fails to perfectly learn the relationship between the available features and the outcome. The time-shift augmentation strategy reduces learning error by

28 increasing sample size, thus allowing the model to learn the relationship between
 29 the features and the outcome more effectively. Second, “irreducible error” is the gap
 30 between the true conditional mean and the true outcomes for people who are identical
 31 according to the features observed in the data [24]. It is called “irreducible” because
 32 it can only be reduced by changing the prediction task (e.g., by changing the set
 33 of features). The partner linkage strategy reduces irreducible error by adding more
 34 features to the data.

35 This paper provides a case study of how time-shift data augmentation and partner
 36 linkage affect predictive performance using data from the Longitudinal Internet studies
 37 for the Social Sciences (LISS), the survey dataset used in the PreFer challenge [6]. For
 38 a visual summary of the core concepts of this paper, see Figure 1. For a description of
 39 our strategy in the register dataset where we also utilized partner linkage, see our other
 40 paper in this special issue [4]. By analyzing how these strategies improve predictive
 41 accuracy in fertility modeling, we aim to inform broader efforts to improve machine
 42 learning performance in social science contexts.

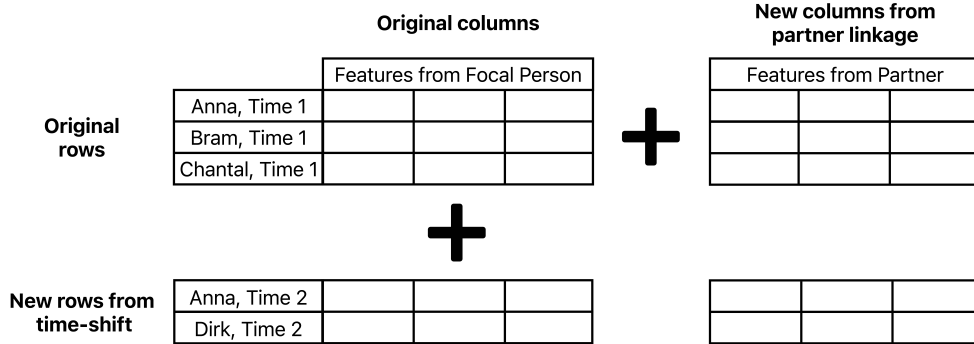


Fig. 1 Visual abstract: adding more rows and columns to improve predictive performance

2 Time-shift data augmentation

2.1 Introduction

To increase the sample size of the training data, we implemented a data engineering strategy we call “time-shift data augmentation.” This is a form of the more general concept of “data augmentation,” in which one modifies the original data to generate additional training observations. Although data augmentation is most commonly associated with image recognition techniques, like flipping or rotating images to create additional training samples [27, 28], similar principles can be applied to tabular data such as the survey data used here. For example, methods like the Synthetic Minority Oversampling Technique (SMOTE) and other techniques for generating artificial samples based on modifications of the raw data can improve model performance, particularly when the sample size is small¹ or when certain groups are underrepresented [26, 43].

Our time-shift data augmentation strategy relies on the fact that many social science datasets, such as panel surveys and government administrative data, repeatedly measure the same construct at regular intervals. Social scientists have long harnessed the statistical power of such iterative designs through panel regression methods and techniques that aggregate exposures across individuals over time as entity-time-periods (e.g., “person-years”) [1, 32]. Our work builds on this modeling tradition by using the “person-years” concept to increase sample size for predictive performance optimization. For examples of other studies using entity-time-period data with predictive modeling, see Stock and Watson [37], Ginsberg et al. [9], and Kessler et al. [17].

Some datasets are inherently organized in the entity-time-period structure, but many datasets—like the survey used in the PreFer challenge—require data wrangling to achieve this structure. These datasets come in a “wide” format where each row refers to an individual and contains multiple years of data. To increase the sample size

¹The value of additional samples decreases as sample size increases; for example, see [4].

69 of a training set in this type of data, we can pivot the dataset into a “long” format
70 where each row represents an entity-time-period, thus “shifting” data from past time
71 periods to align with data from the most recent period.

72 **2.2 Implementation**

73 The outcome for the PreFer challenge was a binary indicator of whether the respondent
74 had a new child by birth or adoption in a three-year period (2021–2023), and the fea-
75 tures were derived from survey questions asked prior to this time period (2007–2020).
76 The survey asks many of the same questions annually, allowing for the observation of
77 the same outcomes and features at various points in time. Adapting the code PreFer
78 organizers used to calculate the official outcomes, we calculated the outcomes for an
79 earlier three-year period (2018–2020) and added these outcome observations to the out-
80 come vector in the training data, creating new rows. For these newly appended rows,
81 we adjusted the features to maintain the same temporal relationship between predic-
82 tors and outcome as in the original data. This involved two steps: First, we removed
83 any features measured after 2017, since these would have occurred during or after the
84 2018–2020 outcome period and therefore could not serve as predictors. Second, we
85 systematically renamed the remaining features to align their temporal positions with
86 the original data structure. We did this by “time-shifting” the features forward by
87 three years. For example, we renamed the 2017 fertility intention variable (cf17j130)
88 to match the 2020 fertility intention variable (cf20m130), because both represented
89 fertility intentions measured one year before their respective outcome periods. Simi-
90 larly, 2016 features were renamed to match 2019 feature names (two years before the
91 outcome), and so on. This renaming ensured that features representing the same tem-
92 poral distance from the outcome were consistently located in the same columns across
93 both original and time-shifted data rows.

For some of the time-shifted variables, we edited not only their names but also their values. First, we added three years to any time-shifted feature that referenced a year, such as birth year or marriage year, to account for the three-year shift. By recoding the birth year of someone born in 1987 as 1990, we informed our machine learning model that this person would be the same age as a non-time-shifted person born in 1990 at the time their outcome time window began. Second, we used Consumer Price Index data from Statistics Netherlands, the Dutch national statistical agency, to adjust for inflation when time-shifting features containing monetary values, such as income [36].

Some of the time-shifted data rows represented earlier observations of the members of the original training set. Other rows were additional people who were not in the 2021–2023 outcome window because they had missing data or were older than the target population (ages 18–45). Figure 2 depicts the size of each of these groups in our time-shift augmentation strategy.

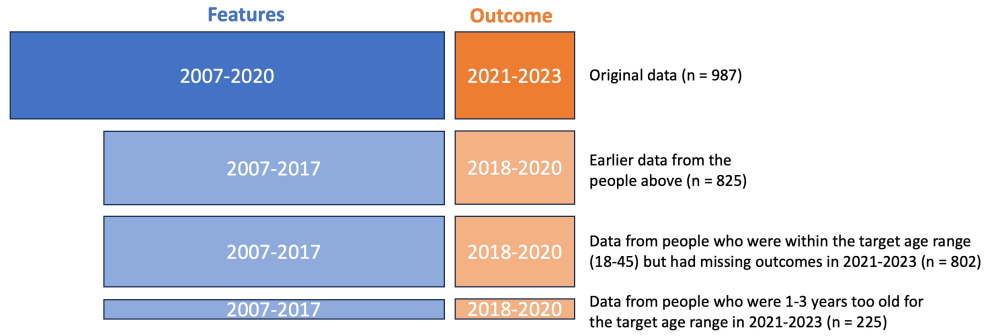


Fig. 2 Time-shift augmentation in the PreFer challenge. We created new rows using the data from earlier years and “time-shifted” the new rows to temporally align with the original data, thereby increasing our sample size. The heights of the bars in the figure are scaled to represent the sample sizes

The time-shift augmentation strategy nearly tripled our sample size from 987 to 2,839. Table 1 presents the breakdown of sample size by outcome class for original data and time-shifted data. The time-shift strategy could be applied to additional time periods in this dataset to further expand the sample size. However, due to time

constraints in the PreFer challenge and complexities in calculating the outcome, we implemented time-shift augmentation for just one time period.

Table 1 Distribution of Outcomes in Original and Time-Shifted Data

Data source	Outcome = 0 (no new child)	Outcome = 1 (new child)	<i>n</i>
Original data (2021–2023)	775	212	987
Time-shifted data (2018–2020)	1484	368	1852
Total <i>n</i>	2259	580	2839

2.3 Sample size versus temporal drift trade-off

Time-shift data augmentation presents a trade-off: while it increases the sample size, it may introduce “temporal drift.” In machine learning, “temporal drift” (also known as “temporal distribution shift”) refers to changes in data patterns over time, which can reduce model accuracy when the model is applied to future time periods [22, 44]. For example, the COVID-19 pandemic may have altered fertility behavior in the 2021–2023 period compared to the 2018–2020 period, such that the patterns learned from the time-shifted 2018–2020 observations are less useful for predicting 2021–2023 outcomes.

Another possible form of temporal drift is a higher level of missingness in time-shifted rows, if the span of the time-shift reaches beyond the earliest time point observed in the dataset. Indeed, after feature engineering but before one-hot encoding, the mean missingness rate for features in the original data was 41%, while the mean missingness rate for features in the time-shifted data was 57% (see Figure A1 [Additional file 1]). Much of this additional missingness occurred because some participants recently joined the LISS panel during sample refreshment. Due to the difference in missingness, variables helpful for predicting fertility in the original data might not perform as well in the time-shifted data.

130 On the other hand, the original data and time-shifted data had similar base rates
131 for the outcome, which is a sign that temporal drift might be low: 21% of participants
132 had a new child in the original data (2021–2023), compared to 20% with a new child
133 in the time-shifted data (2018–2020).

134 Uncertain of the severity of temporal drift, we included a binary feature indicating
135 whether each row came from the original data or the time-shifted data. This feature
136 allowed our model to learn systematic differences between time periods, to the extent
137 that we had sufficient data across periods for the differences to emerge.

138 **2.4 Train-test splits with time-shift data augmentation**

139 Time-shift data augmentation introduces two complications when splitting data into
140 train and test sets. First, observations of the same person from different time periods
141 must be kept within the same split to prevent data leakage, which occurs when the
142 training set contains information about someone in the test set. Leakage can inflate
143 test set performance, misrepresenting the generalizability of the model [3, 15, 30].
144 Second, researchers should carefully consider whether to include time-shifted data in
145 the test set [3]. This decision depends on the prediction task and substantive context.
146 In most scenarios, if the goal is to forecast future outcomes and reduce the impact
147 of temporal drift, it may be preferable to include only the most recent time period
148 in the test set. Please see Section 4.7.2 for further discussion and a diagram of our
149 cross-validation splits.

150 **3 Partner linkage**

151 **3.1 Introduction**

152 Many tabular social science datasets include information on individuals within the
153 same household but store each person’s data in a separate row. This structure prevents
154 machine learning models from incorporating data on other household members when

155 predicting outcomes for a given individual, since the model uses inputs from just
156 one row when generating a prediction. The format is common in both government
157 administrative data and household-based surveys, including the two datasets used in
158 the PreFer challenge: the Dutch administrative register and the LISS panel survey. In
159 such datasets, researchers can enrich the data by linking information about household
160 members, thereby increasing the number of features available for each row.

161 3.2 Implementation

162 In the LISS survey, which sampled at the household level, participants’ spouses or
163 cohabiting partners often participated as well. From here onward, we use the term
164 “partner” to refer to any cohabiting partner, regardless of marital status. Since part-
165 ners typically make fertility decisions jointly [2, 39, 40], we sought to incorporate the
166 partner’s survey responses into each focal person’s row, when available. Specifically,
167 we incorporated the partner’s responses to fertility intentions questions, as well as
168 several other features that we expected to be relevant to fertility.

169 We used several variables to identify partners. The survey assigned each respondent
170 a household ID and a position within the household. We began by identifying all
171 possible pairings in which two people shared a household ID and were each labeled with
172 the position “household head,” “wedded partner,” or “unwedded partner.” To reduce
173 incorrect pairings, we then applied two filters: (1) at least one member of the pair must
174 have reported cohabiting with a partner in the most recent wave they completed, and
175 (2) the reported gender and birth year of each individual had to match their partner’s
176 report, and vice versa, unless someone’s gender or birth year was missing.² These
177 steps helped guard against errors due to changes in relationship status, inconsistencies

²We also implemented an additional filter: we conducted partner linkage only if the partner had a non-missing outcome. We added this stipulation because the PreFer challenge training set includes people for whom the fertility outcome is missing, but the holdout set under evaluation only includes people for whom the outcome is nonmissing. As a result, members of the training set could theoretically be matched to their partner even if the partner had a missing outcome, but members of the holdout set could only be matched to their partner if the partner’s outcome was nonmissing. By only making linkages when both partners had a nonmissing outcome, we roughly equalized the availability of partner data across training and holdout sets, to help ensure the model we trained would generalize to the holdout set.

in labeling, or other edge cases. An example of the data used for linking partners is depicted in Figure 3. We linked time-shifted focal person data with time-shifted partner data, and non-time-shifted focal person data with non-time-shifted partner data. We found partner data for about 35% of all person-years with partners, which accounted for about 72% of person-years across the original and time-shifted datasets.

Person ID	Household ID	Outcome Available	Household Position	Live with Partner	Birth Year	Partner Birth Year	Gender	Partner Gender	Data to Link to Partners	Data Linked from Partners
Person 1	Household 1	Yes	Household Head	Yes	1990	1989	Female	Male	Feature Vector A	Feature Vector B
Person 2	Household 1	Yes	Wedded Partner	Yes	1989	1990	Male	Female	Feature Vector B	Feature Vector A

Fig. 3 Partner linkage. We used the variables with colored cells in this figure to link partners together, as described in the main text. Once we identified these links, we merged a variety of features about each person into their partner’s row (“feature vectors” in the diagram)

Once partners were successfully linked, we merged selected features from the partner’s survey responses to the focal person’s row. As a result, some data from people whose partner participated in the survey appeared twice in our engineered dataset: once in the row where they were the focal person, and once in the row where they were the partner. The merged partner features included information on birth year, fertility intentions and history, and gynecologist visits.

4 Methods³

4.1 Data

We used the LISS dataset, a panel survey representative of the Dutch population aged 16 and older (see Appendix Section A.1.2 [Additional file 1] for sampling details) that covers a variety of social scientific topics [6]. Multiple members of a sampled household may participate in the survey. We used a version of the LISS data provided by the

³We document methods following the guidance of the **Recommendations for Machine-learning-based Science (REFORMS)** checklist [16]; see Appendix C [Additional file 1].

195 PreFer challenge organizers; additional details are available in the PreFer challenge
196 announcement paper [34].

197 4.2 Outcome

198 The outcome was a binary indicator for whether a respondent had at least one new
199 child via birth or adoption in a three-year period (2021–2023), as calculated by the
200 organizers of the PreFer challenge based on respondents’ answers to several sur-
201 vey questions. The population of interest for this outcome was Dutch residents of
202 childbearing age, aged 18–45.

203 4.3 Features

204 With 31,633 possible features (predictor variables) and only 987 cases in the main LISS
205 training dataset, automated feature selection methods could easily pick up spurious
206 correlations that do not generalize beyond the training data. To mitigate this “curse
207 of dimensionality,” we selected features manually rather than automatically [14]. We
208 chose features that we believed were likely to be relevant to fertility based on our
209 practical understanding and prior domain knowledge; we did not review literature on
210 fertility during the feature selection process. We also incorporated three features that
211 colleagues identified using an automated feature selection approach for the PreFer
212 challenge, after assessing the features’ substantive relevance.⁴

213 Sometimes having too many features worsens model performance, particularly in
214 small sample sizes [13, 14, 42]. Therefore, after identifying an initial set of features
215 to try, we removed some of those features through trial and error, based on model
216 performance in cross-validated folds within the training set. Following feature engi-
217 neering but prior to one-hot encoding, our model contained 84 features: 67 from
218 the focal person’s survey, 14 from the partner’s survey, 1 partner birth year variable

⁴Thank you to Sayash Kapoor and Benedikt Ströbl!

219 combining information from focal person and partner surveys, and 2 indicator vari-
220 ables on whether the row contained time-shifted or partner-linked data. See Table A1
221 [Additional file 1] for more details.

222 This final set of features covered a range of topics, including age, gender, partner-
223 ship status, length of current partnership, ages of current children, finances, education,
224 housing, attitudes and beliefs, fertility intentions from recent years, and recent con-
225 tact with a gynecologist. The most important feature was the focal person’s response
226 to a question about fertility intentions in the year before the outcome time window:
227 “Within how many years do you hope to have your [first / next] child? If [you and
228 your partner / you] are currently expecting a child, please enter a zero (0).”⁵

229 4.4 Feature engineering

230 We employed two main feature engineering strategies: (1) constructing “scales” and
231 (2) “coalescing” data over time.

232 First, we constructed several scales, a common method in social sciences for repre-
233 senting abstract attitudes and beliefs. To generate a scale, we calculated the mean of a
234 respondent’s answers to a series of related survey questions rated on a Likert scale (e.g.,
235 “fully disagree” to “fully agree”). This strategy aimed to smooth out response vari-
236 ability across individual survey questions and provide an estimate of the respondent’s
237 beliefs about a core underlying construct. The scales in our model include feelings
238 toward one’s own current child, gendered religiosity, traditional fertility attitudes, tra-
239 ditional motherhood attitudes, traditional fatherhood attitudes, traditional marriage
240 attitudes, opposition to working mothers, and sexism. We derived these scales by iden-
241 tifying clusters of related survey questions; they were not necessarily psychometrically
242 validated.

⁵Importance is measured by XGBoost “gain,” indicating the proportion of log loss reduction attributable to each feature. The feature associated with this question is called cf20m130. Our model also includes features for the same question from 2019 and 2018, as well as the partner’s answers to this question in 2018, 2019, and 2020 for anyone whose partner participated in the survey.

243 Second, we “coalesced” data from groups of features that represented the same
244 construct, meaning we selected the most recent value from across those features. For
245 example, respondents reported their partner’s birth year only once, in the first survey
246 wave after the partnership began, so we coalesced the responses to this question across
247 waves to create a single feature containing the most recent partner’s birth year, regard-
248 less of when that birth year was reported. In addition to coalescing data across waves,
249 we also coalesced children’s birth years within waves, so that data about the first-born
250 child, second-born child, and so on were collapsed into a single feature representing
251 the birth year of the youngest child.

252 We also implemented a few other feature engineering tasks, such as special handling
253 of small categories in categorical variables and rule-based imputation for features with
254 structural missingness [25].

255 4.5 Model

256 We used XGBoost with hyperparameter tuning via grid search. We chose XGBoost
257 because it performed well in other settings and offered a simple out-of-the-box model
258 for testing our time-shift and partner linkage strategies [8]. As a boosted tree model,
259 XGBoost utilizes a series of decision trees, each of which predicts the remaining vari-
260 ation in the outcome not accounted for by earlier trees. We tuned the total number of
261 decision trees, the depth of the trees, the size of the random samples of data the indi-
262 vidual trees were trained on, and the learning rate or shrinkage rate applied to each
263 tree’s representation of remaining variation in the data. For details on how we selected
264 the values in the hyperparameter tuning grid, see Appendix Section A.2 [Additional
265 file 1]. Our algorithm employed a logistic loss function.

266 4.6 Data preparation

267 We one-hot encoded unordered categorical features. We did not apply imputation
268 (other than the rule-based imputation mentioned above) or rescale the data, as these
269 are not necessary for XGBoost.

270 4.7 Train-test splits

271 4.7.1 Official PreFer challenge splits

272 The organizers of the PreFer challenge created a dataset of LISS panel members aged
273 18–45 with a non-missing “new child” outcome, and split this data into a 70% train-
274 ing set and 30% holdout set (training set: $n = 987$, holdout set: $n = 395$). They kept
275 members of the same household within the same split to prevent information leakage
276 between the holdout set and the training set. They provided the training set to partic-
277 ipants in the PreFer challenge, along with supplementary data about people outside
278 the 18–45 age range and people with missing outcomes. Participants trained models
279 with the training and supplementary data and submitted these models for evaluation
280 on the holdout set [34]. For a diagram of the train-test splits for the official PreFer
281 challenge, please see Sivak et al. [34].

282 4.7.2 Cross-validation without drift or leakage

283 Within the official PreFer training set, our team utilized 5-fold cross-validation for
284 hyperparameter tuning and model selection. As discussed in Section 2.4, the time-
285 shift data augmentation strategy requires special considerations for train-test splits.
286 We wanted our cross-validated test sets to resemble the official PreFer holdout set as
287 closely as possible, so to avoid temporal drift in the test sets, we used time-shifted
288 data only in the training sets. To achieve this, we first generated cross-validation splits
289 within the original (i.e., not time-shifted) PreFer training set. We then appended

time-shifted data to the training splits of these cross-validation folds. These splits are depicted in Figure 4.

To prevent leakage, we applied two additional constraints in the creation of these splits (not depicted in the diagram). First, we assigned all members of the same household to the same fold. Second, since the time-shift augmentation introduced earlier observations of the same individuals, we excluded any time-shifted row from a training set if it was an earlier observation of someone in the test set.

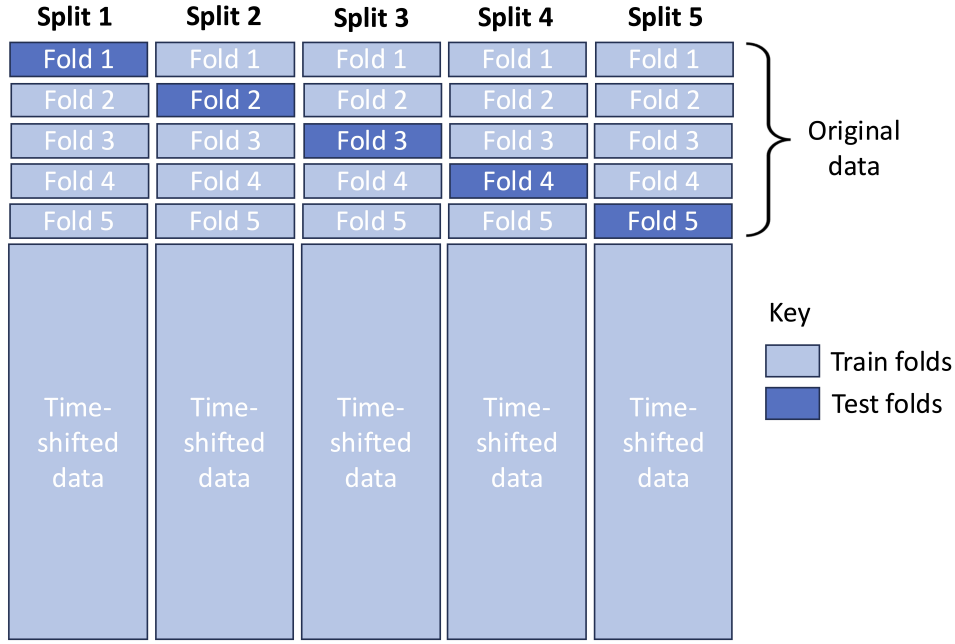


Fig. 4 Cross-validation splits. In our cross-validation splits, we included time-shifted data only in training sets, not in test sets. We also applied additional criteria to prevent leakage (not depicted), as described in the main text

4.7.3 Assessing performance

The results in the manuscript submitted to the journal for review are based on cross-validated performance. Following existing practice [23], we are reserving the holdout

300 set to evaluate performance at the end of the journal review process, so that we main-
301 tain a clean holdout set in case reviewer feedback alters any of our methodological
302 decisions. Note that the performance of our model that combines time-shift aug-
303 mentation and partner linkage was already evaluated on the holdout set during the
304 PreFer challenge as part of the competition, but we have not utilized the holdout set
305 specifically to evaluate the impact of each of these strategies.

306 The sample size of the holdout set is small ($n = 395$), so if the holdout set results
307 are consistent with the cross-validated test set results in terms of the impact of time-
308 shift augmentation and partner linkage, we may pool the holdout and cross-validation
309 results together in the main paper for more precise performance estimates, and report
310 results for the holdout set alone in the appendices as a robustness check.

311 We note that during the PreFer challenge, we aimed to maximize cross-validation
312 performance when choosing predictive features and tuning hyperparameters, poten-
313 tially overfitting our model on the cross-validation data. However, in Section 5.2, we
314 show that even if we did no hyperparameter tuning or simply used a set of baseline
315 features recommended by PreFer organizers, time shift data augmentation and part-
316 ner linkage together still improved cross-validation performance. Furthermore, we did
317 not use cross-validation performance to guide any decisions about time-shift data aug-
318 mentation, so cross-validation performance should not systematically overestimate the
319 impact of this strategy.

320 4.8 Metrics

321 We optimized our model to maximize the F1 score because F1 was the primary metric
322 for the PreFer challenge. We report our main results in terms of R_{CV}^2 (Equation 1) due
323 to its interpretability. $R_{CV}^2 = 1$ indicates perfect predictions, while $R_{CV}^2 = 0$ indicates
324 that the model’s predictions are no better than if the mean of the training folds were
325 used as the prediction for everybody in the test fold. As discussed in Section 4.7.3, after

we receive feedback from reviewers, we will update this metric to R_{Holdout}^2 (Equation 2) or a pooled $R_{\text{CV, Holdout}}^2$ (Equation A1 [Additional file 1]) to reflect holdout set performance.

$$R_{\text{CV}}^2 = 1 - \frac{\sum_{i \in \text{All Folds}} (y_i - \hat{y}_i)^2}{\sum_{i \in \text{All Folds}} (y_i - \bar{y}_{\text{Training}_j})^2} \quad (1)$$

Eq. 1 Formula for cross-validated (“CV”) R^2 computed across all five folds. For each of the five cross-validated test folds, $\bar{y}_{\text{Training}_j}$ refers to the mean outcome across the four training folds associated with the test fold

$$R_{\text{Holdout}}^2 = 1 - \frac{\sum_{i \in \text{Holdout}} (y_i - \hat{y}_i)^2}{\sum_{i \in \text{Holdout}} (y_i - \bar{y}_{\text{Training}})^2} \quad (2)$$

Eq. 2 Formula for R^2 computed on the holdout set

When using R_{CV}^2 , R_{Holdout}^2 , or $R_{\text{CV, Holdout}}^2$ with time-shift data augmentation, the researcher must decide whether the mean of the training set in the denominator should refer to the mean of the entire training set or the mean of just the original (non-time-shifted) portion of the training set. We adopted the latter approach because if the mean of the training set includes time-shifted data, temporal distribution shifts in fertility rates might inflate the denominator in the equations above and thus inflate the performance of the time-shifted model.

To illustrate the robustness of our estimates to random perturbations within the given test data, we generated 95% confidence intervals (CIs) for model performance using 2,000 bootstraps of the cross-validated test sets. In this bootstrap procedure, we resampled individuals in the cross-validated test sets with replacement 2,000 times, calculated R_{CV}^2 on each of these samples, and used the resulting distribution of R_{CV}^2 to construct the CI. We also generated bootstrapped CIs for *improvements* in model

performance (e.g., for the improvement between a model trained on the original data and a model that utilizes time-shift augmentation). For each of the 2,000 resamples, we calculated the difference in R_{CV}^2 between models and used the resulting distribution of the differences to construct a CI. The intervals for performance improvement tended to be narrower than those for performance, because much of the sampling variation in prediction error applied equally to two models evaluated on the same bootstrap sample and was canceled out when we took the difference between performance scores.

To aid comparison with other literature, Appendix B [Additional file 1] reports results in terms of several other metrics: $R_{\text{Traditional}}^2$,⁶ mean squared error (MSE), log loss, F1, precision, recall, accuracy, and the area under the receiver operating characteristic curve (AUC). R_{CV}^2 , R_{Holdout}^2 , $R_{CV,\text{Holdout}}^2$, $R_{\text{Traditional}}^2$, MSE, log loss, and AUC are calculated for predicted probabilities, while F1, precision, recall, and accuracy are calculated for predicted classes.

5 Results

5.1 Improvements in XGBoost performance

When we applied either time-shift augmentation or partner linkage, cross-validated performance tended to improve. Figure 5 shows the performance of a model trained on the original training set, alongside the performance of models with each data engineering strategy. Figure 6 shows the improvement in R_{CV}^2 achieved by each strategy.

The figures show that with the original training set, $R_{CV}^2 = .543$, 95% CI [.477, .608] (see Section 4.8 for how we constructed the confidence intervals). Partner linkage raised R_{CV}^2 to .557 [.484, .629], while time-shift augmentation (without partner linkage) raised R_{CV}^2 to .581 [.512, .648]. The respective improvements over the original

⁶Unlike R_{Holdout}^2 , $R_{\text{Traditional}}^2$ uses the mean of the test set rather than the training set in the denominator.

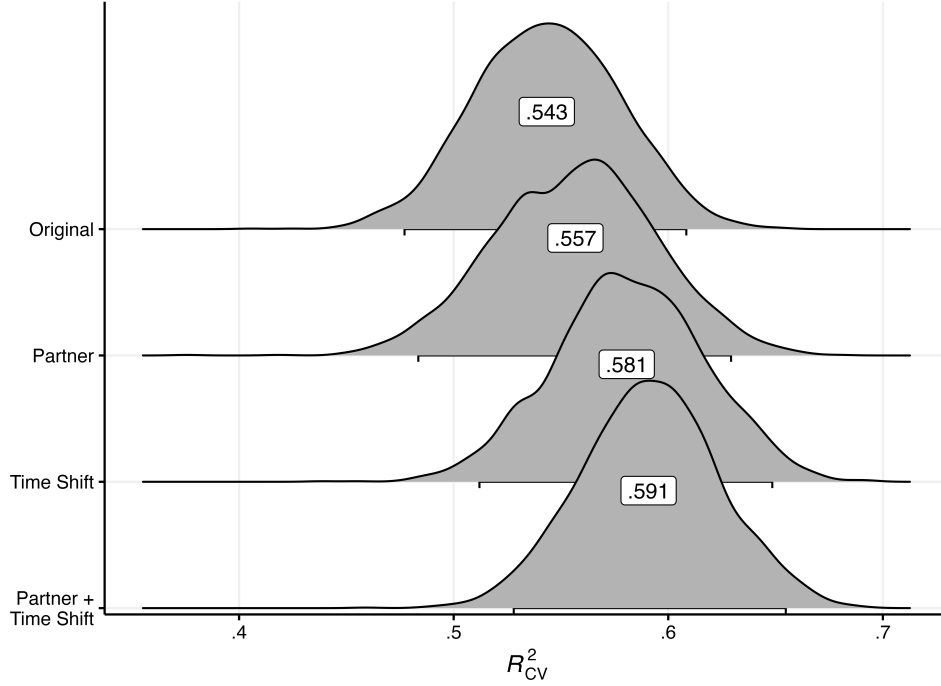


Fig. 5 R^2_{CV} with original data versus engineered data (XGBoost). Boxed numbers are the performance on the pooled cross-validation test folds. Ridges show the smoothed distribution of performance across 2,000 bootstrap resamples. Tick marks denote 95% CIs

performance, .013 $[-.023, .048]$ and .037 $[-.002, .074]$, suggest modest gains, although the CIs included zero. The small gains from partner linkage in part reflect the fact that we could only merge in partner data for about 35% of partnered individuals. More complete data coverage would likely further improve performance.

Applying partner linkage and time-shift augmentation together raised R^2_{CV} to .591 $[.528, .655]$, an improvement of .047 $[.012, .081]$ compared to the original data. Other metrics generally produced similar results in terms of the impact of our data engineering strategies, with the exception that for recall, only the time-shift strategy improved performance (see Table B1 [Additional file 1]). These results demonstrate that time-shift augmentation and partner linkage both have the potential to improve predictive

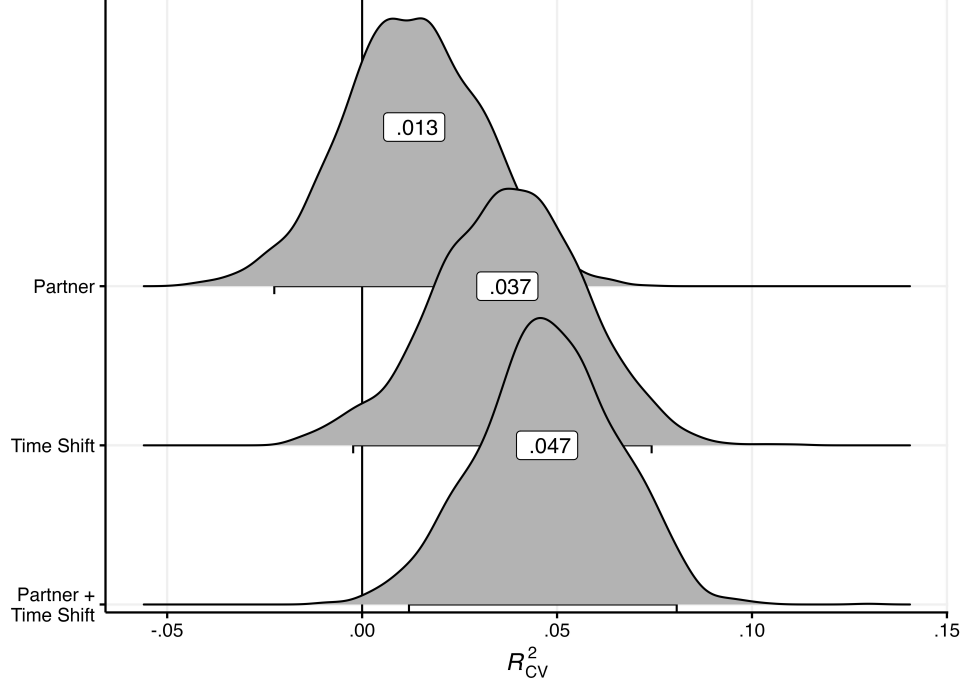


Fig. 6 Change in R^2_{CV} compared to original data (XGBoost)

performance in tabular social science data, and that they can achieve even better performance when combined.

5.2 Improvements in logistic regression performance

Time-shift data augmentation and partner linkage have the potential to improve performance not only for XGBoost but also for other model types. As a simple proof-of-concept, Figures 7 and 8 respectively show the impact of our data engineering strategies for two additional models: a logistic regression model with seven baseline features hand-chosen by the organizers of the PreFer challenge,⁷ and a logistic regression model with all the features from our winning PreFer submission. When used together, our two data engineering strategies improved performance in both of these

⁷We combined the “demographic” baseline and “fertility intentions” baseline from Sivak et al. [35] into a single model.

390 additional models across all nine metrics. Furthermore, the impacts of time-shift aug-
 391 mentation alone or partner linkage alone were mostly in the expected direction (see
 392 Tables B3 and B5 [Additional file 1] for all metrics).

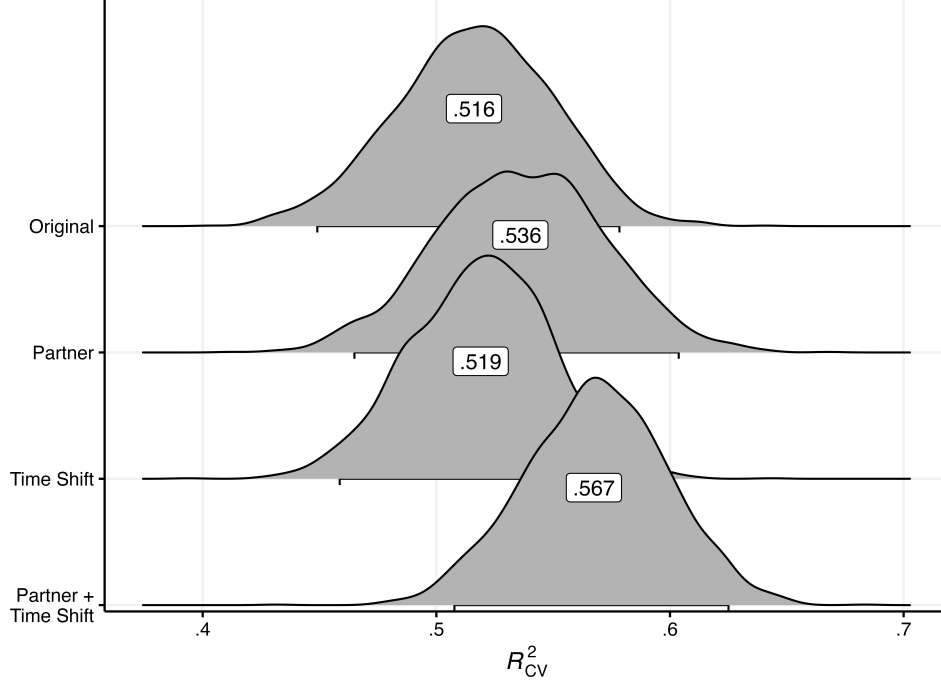


Fig. 7 R^2_{CV} with original data versus engineered data (logistic regression, baseline features)

393 5.3 Putting the magnitude of the improvements in context

394 To contextualize the magnitude of the improvements from time-shift data augmenta-
 395 tion and partner linkage, we compared these improvements to performance differences
 396 between our winning model and other models.

397 First, we compared the impact of our data engineering strategies to the perfor-
 398 mance gap between our winning model and a simple “fertility intentions” baseline.
 399 The “fertility intentions” baseline was an XGBoost model with two features indicating

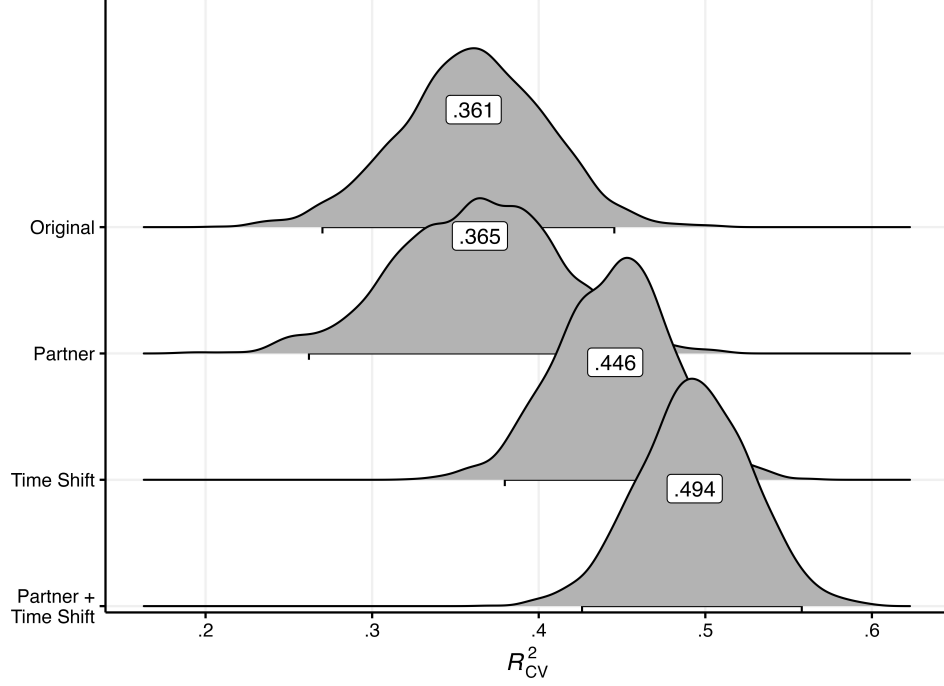


Fig. 8 R^2_{CV} with original data versus engineered data (logistic regression, all features from winning model)

400 whether and when the focal person intended to have a new child in the future, based
 401 on their survey responses in the year before the outcome was measured (or the equiv-
 402 alent time-shifted year).⁸ As results for the “fertility intentions” model in Figure 9
 403 show, our final model with 84 features outperformed this fertility intentions baseline
 404 by an R^2_{CV} difference of .156, 95% CI [.116, .196]. In comparison, the joint benefit
 405 of time-shift augmentation and partner linkage yielded a smaller R^2_{CV} improvement
 406 of .047 [.012, .081]. Thus, the two data engineering strategies had an impact equiva-
 407 lent to about 30% of what our winning model achieved beyond the fertility intentions
 408 baseline.

⁸This baseline is inspired by a similar baseline in the PreFer community paper. The two features are cf20m128 (“Do you think you will have (more) children in the future?”) and cf20m130 (“Within how many years do you hope to have your [first / next] child? If [you and your partner / you] are currently expecting a child, please enter a zero (0).”). Since we used time-shift augmentation when training this baseline model, the model also contains a binary feature for whether a row comes from the original data or time-shifted data, as discussed in Section 2.3.

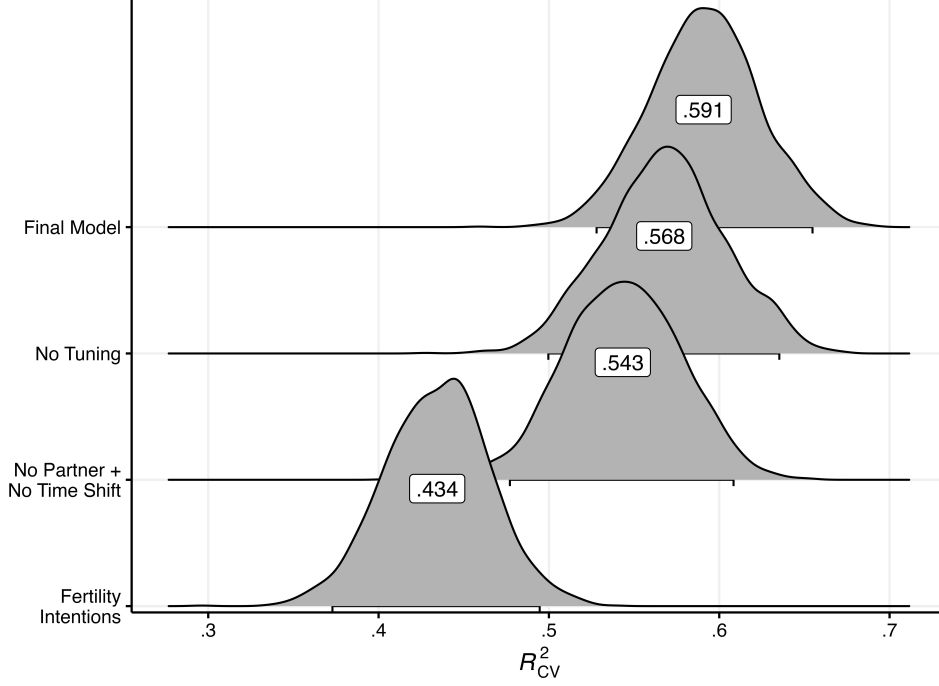


Fig. 9 R^2_{CV} with final model versus less engineered models (XGBoost)

Next, we compared the impact of our data engineering strategies to the impact of hyperparameter tuning. As results for the “no tuning” model in Figure 9 show, using default hyperparameter settings in our full model instead of the best hyperparameters selected through hyperparameter grid tuning reduced R^2_{CV} by about .023, 95% CI [.006, .041].⁹ Therefore, in terms of R^2_{CV} , the joint benefit of time-shift augmentation and partner linkage was larger than the gain from hyperparameter tuning. We obtained similar results for $R^2_{\text{Traditional}}$, MSE, log loss, AUC, and precision. However, regarding F1, recall, and accuracy, hyperparameter tuning helped more than our two

⁹During hyperparameter tuning, we determined and adjusted the hyperparameter values in the search grid by referring to recommendations in model documentation, consulting machine learning textbooks [11], and inspecting cross-validated performance. Our final model had 119 trees with a maximum depth of 6. Each tree modeled 80% of the training data, and the full model had a learning rate of 0.03. By default, XGBoost in the `tidymodels` package has 15 trees with a maximum depth of 6 [19]. Each tree models 100% of the training data, and the full model has a learning rate of 0.3. **Note to reviewer:** These performance measures are based on cross-validated performance and are likely an overestimate of the benefits of hyperparameter tuning because we conducted tuning by referring to cross-validation performance. We will have a more accurate estimate of the impact of hyperparameter tuning when we test on the holdout set after the review process.

417 engineering strategies (see Table B7 [Additional file 1]). Taking all evidence together,
418 we conclude that the payoffs of our combined strategies are roughly comparable to
419 those of hyperparameter tuning in this context.

420 Finally, we note that in terms of F1 score, which was the primary metric used
421 in the PreFer challenge, our performance with time-shifted and partner-linked data
422 exceeded the performance of a model trained on the original data by .030, 95% CI
423 [.000, .059]. This difference was nearly as large as the F1 difference of .037 between
424 our model and the model that won second place in the PreFer challenge [7].

425 6 Recommendations for data and metadata structure

426 Given the benefits of time-shift data augmentation and partner linkage for improving
427 predictive performance, we encourage data stewards to organize data in machine-
428 parsable formats that not only enable these techniques but also support a wide range
429 of other machine learning applications.

430 To facilitate time-shift data augmentation, panel surveys and other datasets that
431 contain repeated measures of the same construct over time should be structured such
432 that variables can be automatically matched across waves. For example, in the LISS
433 survey, time-shift data augmentation is possible due to consistent variable naming
434 conventions: the stem of the variable name remains the same across waves, while a
435 section of the name varies to indicate the year [38]. Alternatively, variable matching
436 across time points could be made even easier by providing metadata that indicates the
437 year of each variable and identifies which variables capture the same construct [18].

438 To facilitate partner linkage and other household member linkage, datasets with
439 information on multiple members of the same household should provide a household
440 ID variable, plus data on the relationships among household members. For example,
441 LISS provides a variable that indicates each person’s “position” in the household, such

442 as “household head,” “wedded partner,” or “child living at home” [5]. In data envi-
443 ronments that are exceptionally rich, such as administrative registers, data stewards
444 could also denote social network linkages beyond the household to facilitate research
445 that leverages broader social ties, potentially by providing additional variables whose
446 values refer to other individuals’ IDs [21].

447 7 Discussion

448 Dozens of teams around the world explored a wide range of machine learning strategies
449 to predict births and adoptions with Dutch survey data. Our advantage in outper-
450 forming the other submissions was not an innovative model: like 13 other teams in
451 the PreFer challenge, we used a boosted tree model readily available on major public
452 code repositories. Rather, our advantage came mainly from creative data engineering:
453 we transformed the existing data into effectively “more” data.

454 Both of our data engineering strategies can be implemented in a variety of con-
455 texts. Time-shift data augmentation can be used to *increase sample size* in any tabular
456 dataset where the same constructs are repeatedly measured at regular time intervals,
457 such as government administrative records or longitudinal panel surveys, or in con-
458 tinuously updated event data, such as administrative records that are updated every
459 time a birth (or other event of interest) occurs. Meanwhile, partner linkage (or link-
460 age with other household members) can be used to *increase the number of features*
461 in any dataset where members of the same household are stored in separate rows or
462 files, which is common in government administrative records and surveys sampled at
463 the household level.

464 The impact of time-shift data augmentation and partner linkage likely depends on
465 the specific prediction task. In studies of processes that change substantially over time,
466 temporal drift may reduce the utility of time-shift data augmentation [22, 44]. If the
467 sample size is already large, time-shift augmentation is also unlikely to greatly improve

468 performance [4, 42]. For outcomes where partner characteristics are less relevant, part-
469 ner linkage might lead models to overfit on noise in the additional variables, possibly
470 reducing predictive performance [13, 14, 42]. We note, however, that this “curse of
471 dimensionality” might be alleviated by the increased sample size afforded by time-
472 shift data augmentation. Given the predictive benefits our winning model achieved
473 from the two strategies, we encourage other data analysts working with small datasets
474 on longstanding family processes to consider combining the two strategies and testing
475 the resulting models on a validation set.

476 Other methods of data augmentation for social science data are also promising. For
477 example, team LYC-MRK (Li Ya Chang and Mojtaba Rostami Kandroodi) achieved
478 the second-highest score with survey data in the PreFer challenge by using a different
479 form of data augmentation, SMOTE, to artificially increase the number of observations
480 with positive outcomes [7]. SMOTE or other data augmentation techniques could be
481 combined with time-shift augmentation and partner linkage to potentially achieve
482 even higher predictive accuracy. Future research should test time-shift augmentation
483 and partner linkage in other social science datasets, as data engineering may improve
484 predictive performance in a wide range of applications.

Declarations

Availability of data and materials

The LISS panel data are available at <https://www.lissdata.nl/how-it-works-archive>. Scripts for creating PreFer datasets from LISS data are available on the project page in the LISS data archive at <https://doi.org/10.57990/f3ge-3a61>. Prepared PreFer survey datasets are available via the Secure Analysis Environment at <https://odisseei-data.nl/facility/secure-analysis-environment-sane/>.

Code availability

Code and instructions for replicating the results in this paper are available at <https://github.com/HanzhangRen/fertility-prediction-challenge-2024>.

Conflict of interests

There is no conflict of interest to declare.

Funding

In this paper we make use of data of the LISS panel administered by Centerdata (Tilburg University, the Netherlands). Funding for the panel's ongoing operations has come from the Domain Plan Social Sciences & Humanities (Domain Plan SSH) and the Open Data Infrastructure for Social Science and Economic Innovations (ODISSEI) since 2019. The initial set-up of the LISS panel in 2007 was funded through the Measurement and Experimentation in the Social Sciences (MESS) project by the Netherlands Organization for Scientific Research (NWO). The ODISSEI Benchmark Platform and the development of the LISS harmonized dataset are financed by the ODISSEI Roadmap Project financed by NWO.

Ethics approval and consent to participate

The PreFer challenge study was approved by the Ethical Committee of Sociology at the University of Groningen (SOC-2425-S-0002).

Consent for publication

Not applicable.

Author contributions

Both authors contributed equally to the conception and implementation of the study, including designing modeling strategies for the PreFer challenge, coding, analysis, and manuscript writing. The work was completed collaboratively and iteratively.

Acknowledgments

We express enormous thanks to Elizaveta Sivak and Gert Stulp for leading the PreFer challenge and for answering our many questions along the way. We also thank Jeremy Freese, Matthew Salganik, Varun Satish, Elizaveta Sivak, Brandon Stewart, and Gert Stulp for their feedback on drafts of this manuscript.

Abbreviations

- **PreFer:** Predicting Fertility Data Challenge
- **LISS:** Longitudinal Internet studies for the Social Sciences
- **SMOTE:** Synthetic Minority Oversampling Technique
- **REFORMS:** Recommendations for Machine-learning-based Science
- **CI:** confidence interval
- **MSE:** mean squared error
- **AUC:** area under the receiver operating characteristic curve
- **Domain Plan SSH:** Domain Plan Social Sciences & Humanities
- **ODISSEI:** Open Data Infrastructure for Social Science and Economic Innovations
- **MESS:** Measurement and Experimentation in the Social Sciences
- **NWO:** Netherlands Organization for Scientific Research (Nederlandse Organisatie voor Wetenschappelijk Onderzoek)

References

- [1] Allison PD (1982) Discrete-time methods for the analysis of event histories. *Sociol Methodol* 13:61–98. doi:10.2307/270718
- [2] Bauer G, Kneip T (2014) Dyadic fertility decisions in a life course perspective. *Adv Life Course Res* 21:87–100. doi:10.1016/j.alcr.2013.11.003
- [3] Bergmeir C, Benítez JM (2012) On the use of cross-validation for time series predictor evaluation. *Inf Sci* 191:192–213. doi:10.1016/j.ins.2011.12.028
- [4] Cantrell EM, Ren H, Kapoor S, et al (2026) More data, modest gains: Assessing limits of prediction in multi-domain population-scale social data. *EPJ Data Science* (manuscript under minor revision)
- [5] Centerdata (2026) Background variables. <https://www.lissdata.nl/background-variables>. Accessed 24 July 2026
- [6] Centerdata (2026) Home. <https://www.lissdata.nl/>. Accessed 24 July 2026
- [7] Chang LY, Kandroodi MR (2026) Predicting individual’s fertility outcomes: A data science approach (working paper)
- [8] Chen T, Guestrin C (2016) XGBoost: a scalable tree boosting system. In: Krishnapuram B, Shah M, Smola AJ, et al (eds) *KDD ’16: proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. Association for Computing Machinery, New York, pp 785–794
- [9] Ginsberg J, Mohebbi MH, Patel RS, et al (2009) Detecting influenza epidemics using search engine query data. *Nature* 457(7232):1012–1014. doi:10.1038/nature07634

- [10] Grimmer J, Roberts ME, Stewart BM (2021) Machine learning for social science: an agnostic approach. *Annu Rev Political Sci* 24:395–419. doi:10.1146/annurev-polisci-053119-015921
- [11] Hastie T, Tibshirani R, Friedman J (2009) *The elements of statistical learning: data mining, inference, and prediction*, 2nd edn. Springer, New York
- [12] Hofman JM, Watts DJ, Athey S, et al (2021) Integrating explanation and prediction in computational social science. *Nature* 595(7866):181–188. doi:10.1038/s41586-021-03659-0
- [13] Hughes GF (1968) On the mean accuracy of statistical pattern recognizers. *IEEE Trans Inf Theory* 14(1):55–63. doi:10.1109/TIT.1968.1054102
- [14] James G, Witten D, Hastie T, et al (2021) *An introduction to statistical learning: with applications in R*, 2nd edn. Springer, New York
- [15] Kapoor S, Narayanan A (2023) Leakage and the reproducibility crisis in machine-learning-based science. *Patterns* 4(9). doi:10.1016/j.patter.2023.100804
- [16] Kapoor S, Cantrell EM, Peng K, et al (2024) REFORMS: consensus-based recommendations for machine-learning-based science. *Sci Adv* 10(18). doi:10.1126/sciadv.adk3452
- [17] Kessler RC, Hwang I, Hoffmire CA, et al (2017) Developing a practical suicide risk prediction model for targeting high-risk patients in the Veterans health Administration. *Int J Methods Psychiatr Res* 26(3). doi:10.1002/mpr.1575
- [18] Kindel AT, Bansal V, Catena KD, et al (2019) Improving metadata infrastructure for complex surveys: insights from the Fragile Families Challenge. *Socius* 5. doi:10.1177/2378023118817378

- [19] Kuhn M, Vaughan D (2026) Boosted trees via xgboost.
https://parsnip.tidymodels.org/reference/details_boost_tree_xgboost.html.
 Accessed 24 July 2026
- [20] Kuikka S (2024) The (un)predictability of early (un)employment: a machine learning approach. *Socius* 10. doi:10.1177/23780231241286655
- [21] van der Laan DJ (2022) A person network of the Netherlands.
<https://www.cbs.nl/en-gb/background/2022/20/a-person-network-of-the-netherlands>. Accessed 24 July 2026
- [22] Logeswaran Y, Oliver D (2025) It’s about time: why we need to consider temporal drift when developing and implementing clinical prediction models. *Biol Psychiatry: Cogn Neurosci Neuroimaging* 10(7):677–678. doi:10.1016/j.bpsc.2025.05.003
- [23] Ludwig J, Mullainathan S (2024) Machine learning as a tool for hypothesis generation. *Q J Econ* 139(2):751–827. doi:10.1093/qje/qjad055
- [24] Lundberg I, Brown-Weinstock R, Clampet-Lundquist S, et al (2024) The origins of unpredictability in life outcome prediction tasks. *Proc Natl Acad Sci U S A* 121(24). doi:10.1073/pnas.2322973121
- [25] Luzi O, di Zio M, Guarnera U, et al (2007) Recommended practices for editing and imputation in cross-sectional business surveys. <https://ec.europa.eu/eurostat/documents/64157/4374310/30-Recommended+Practices-for-editing-and-imputation-in-cross-sectional-business-surveys-2008.pdf>. Accessed 24 Jul 2026
- [26] Machado P, Fernandes B, Novais P (2022) Benchmarking data augmentation techniques for tabular data. In: Hujun Y, Camacho D, Tiño P (eds) *Intelligent*

- data engineering and automated learning – IDEAL 2022 23rd international conference, IDEAL 2022, Manchester, UK, November 24–26, 2022, proceedings, part i. Lecture notes in computer science, vol 13756. Springer, Cham, pp 104–112
- [27] Maharana K, Mondal S, Nemade B (2022) A review: data pre-processing and data augmentation techniques. *Glob Transit Proc* 3(1):91–99. doi:10.1016/j.gltp.2022.04.020
 - [28] Mumuni A, Mumuni F (2022) Data augmentation: a comprehensive survey of modern approaches. *Array* 16. doi:10.1016/j.array.2022.100258
 - [29] van de Rijt A, Bernardi F, Foley W, et al (2026) Luck and predictability in the life course. *Annu Rev Sociol* 52. doi:10.1146/annurev-soc-031324-115112
 - [30] Roberts DR, Bahn V, Ciuti S, et al (2017) Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography* 40(8):913–929. doi:10.1111/ecog.02881
 - [31] Salganik MJ, Lundberg I, Kindel AT, et al (2020) Measuring the predictability of life outcomes with a scientific mass collaboration. *Proc Natl Acad Sci USA* 117(15):8398–8403. doi:10.1073/pnas.1915006117
 - [32] Singer JD, Willett JB (2003) *Applied longitudinal data analysis: modeling change and event occurrence*. Oxford University Press, New York
 - [33] Sivak E, Stulp G (2026) Introduction to the special issue on the PreFer data challenge (working paper)
 - [34] Sivak E, Pankowska P, Mendrik A, et al (2024) Combining the strengths of Dutch survey and register data in a data challenge to predict fertility (PreFer). *J Comput Soc Sci* 7(2):1403–1431. doi:10.1007/s42001-024-00275-6

- [35] Sivak E, Cantrell EM, Emery T, et al (2026) A data challenge with rich survey and full-population register data reveals the limits of predicting births (manuscript submitted for publication)
- [36] Statistics Netherlands (2022) Inflation rate up to 14.5 percent in September. <https://www.cbs.nl/en-gb/news/2022/40/inflation-rate-up-to-14-5-percent-in-september>. Accessed 25 July 2026
- [37] Stock JH, Watson MW (2002) Macroeconomic forecasting using diffusion indexes. *J Bus Econ Stat* 20(2):147–162. doi:10.1198/073500102317351921
- [38] Stulp G, Sivak E (2024) A guide on PreFer datasets [LISS]. <https://stulp.gmw.rug.nl/prefer/posts/posts/2024-03-20-prefer-datasets.html>. Accessed 24 July 2026
- [39] Thomson E (1997) Couple childbearing desires, intentions, and births. *Demography* 34(3):343–354. doi:10.2307/3038288
- [40] van Tintelen AMG, Stulp G (2024) Explaining uncertainty in women’s fertility preferences. *Heliyon* 10(6). doi:10.1016/j.heliyon.2024.e27610
- [41] Verhagen MD (2022) A pragmatist’s guide to using prediction in the social sciences. *Socius* 8. doi:10.1177/23780231221081702
- [42] Viering T, Loog M (2023) The shape of learning curves: a review. *IEEE Trans Pattern Anal Mach Intell* 45(06):7799–7819. doi:10.1109/TPAMI.2022.3220744
- [43] Volkova S (2024) An overview on data augmentation for machine learning. In: Gibadullin A (ed) Digital and information technologies in economics and management: proceedings of the international scientific and practical conference ”Digital and information technologies in economics and management” (DITEM2023).

Lecture notes in networks and systems, vol 942. Springer, Cham, pp 143–154

- [44] Yao H, Choi C, Cao B, et al (2023) Wild-time: a benchmark of in-the-wild distribution shift over time. In: Koyejo S, Mohamed S, Agarwal A, et al (eds) Advances in neural information processing systems 35: 36th conference on neural information processing systems (NeurIPS 2022), vol 14. Curran Associates, Red Hook, pp 10309–10324

- [45] Zheng H, Cheng S (2025) Social rigidity across and within generations: a predictive approach. *Sociol Methods Res* 54(4):1683–1725. doi:10.1177/00491241251347984

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [RenCantrellAppendix.zip](#)