

Building a reusable benchmark for life-outcome prediction: design, modelling approaches, predictors, and reproducibility of the PreFer data challenge

Elizaveta Sivak

e.sivak@rug.nl

University of Groningen

Malvina Nissim

University of Groningen

Gert Stulp

University of Groningen

Research Article

Keywords: machine learning, fertility, data challenge, prediction, reproducibility

Posted Date: October 26th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11231888/v1>

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

[Read Full License](#)

Additional Declarations: No competing interests reported.

Building a reusable benchmark for life-outcome prediction: design, modelling approaches, predictors, and reproducibility of the PreFer data challenge

Elizaveta Sivak^{1,2*}, Malvina Nissim³, Gert Stulp^{1,2}

¹Department of Sociology and ²Inter-University Center for Social Science Theory and Methodology, University of Groningen, Grote Rozenstraat 31, 9712 TS Groningen, The Netherlands

³Center for Language and Cognition Groningen, Faculty of Arts, University of Groningen, Oude Kijk in 't Jatstraat 26, 9712 EK Groningen, The Netherlands

*Corresponding author. Email: e.sivak@rug.nl

Abstract

Benchmarks – shared prediction tasks with fixed data and a common metric – have driven cumulative progress in many fields but remain rare in the social sciences. We document PreFer (Predicting Fertility in the Netherlands), a data challenge in which participants predicted whether Dutch residents aged 18–45 in 2020 would have a child in 2021–2023, using longitudinal LISS panel surveys or full-population administrative registers. About 150 researchers made 69 valid survey-based and 11 register-based submissions. We describe the challenge design and review how teams handled thousands of features, a small and imbalanced survey sample, missing values and longitudinal data. We then analyse the reproducibility of submissions two years after the challenge. Requiring code and trained models rather than predictions in the survey phase made most submissions reproducible: 67 of 69 saved models reproduced their original scores, while retraining from code reproduced the original F1 score to within 0.03 for 52 submissions. Finally, using feature importance and ablation experiments, we identify which topics contributed to predictive performance. The predictive signal was highly concentrated: in the survey data, fertility intentions with partnership and household information came close to full-model performance; the remaining demographic, economic and attitudinal variables added little. In the register data, which lack intentions, partnership and household characteristics each recovered most of the attainable performance. These analyses inform other data challenges in the social sciences and establish PreFer as a reusable benchmark for cumulative fertility research by providing a reference point against which new predictors, data, and methods can be tested.

Keywords: machine learning, fertility, data challenge, prediction, reproducibility

Introduction

In many fields, progress has been organised around benchmarks, or data challenges: competitions with a shared prediction task, a fixed dataset, and a single measure of how well a model predicts the outcome for observations it was not trained on [1–4]. Holding the task and measure constant makes results cumulative: different approaches can be compared on equal terms, improvement becomes measurable, and researchers can learn from the most predictive models [5–7]. It also drives methodological innovation, as researchers compete to reduce the prediction error [1, 4]. This approach is well established in fields such as computer science, medicine, and biology, but remains uncommon in the social sciences, where research has overwhelmingly asked how specific variables relate to an outcome within a sample rather than how well that outcome can be predicted for new observations.

Fertility research is one example. Decades of work have identified dozens of factors associated with childbearing [8], but the measures typically used to report these associations, such as p-values, coefficient sizes, and in-sample R^2 , depend on the model specification and sample. Findings from different studies thus cannot tell which factors matter most or be combined to establish how much of fertility behaviour is explained overall. By using only this approach, we miss the opportunities that prediction tasks and benchmarks provide: learning from the best-performing models and from changes in predictive performance over time, testing whether new data or methods improve prediction, and stimulating innovation.

We provide such a benchmark through PreFer (Predicting Fertility in the Netherlands), a large international and interdisciplinary data challenge, to study how well one specific fertility outcome can be predicted: whether individuals of reproductive age (18–45 years old in 2020) will have a child within the next three years (2021–2023). More than 140 researchers participated in PreFer. A key feature of PreFer is its use of both full-population administrative data and survey data. Administrative data comes from Dutch registers [9] – one of the world's largest social datasets – and includes detailed life-course information relevant for predicting fertility outcomes. The survey data comes from the longitudinal surveys conducted on the LISS panel [10], a representative sample of Dutch households, also covering life-course information, plus a multitude of subjective measures including individuals' fertility intentions. This setting allowed us to study the current predictability of fertility outcomes and what limits it [11].

To support cumulative science, a benchmark needs to be documented and preserved well enough that others can reuse it. Documenting PreFer in this way is the main aim of this paper. First, this paper outlines the methodology of PreFer, which could provide a blueprint for other challenges, methodologically and logistically. Second, we document what teams have already tried: their strategies for feature selection and engineering, how they handled class imbalance, missing values, and algorithm selection, and how they tried to increase the survey sample size. This can also be useful beyond PreFer because the difficulties teams faced – a huge feature space, a relatively small sample of the survey data, an imbalanced outcome, missing values, longitudinal structure – are typical for many social science prediction tasks. This description therefore also acts as a catalogue of options with observed scores for researchers facing similar tasks with other data.

Third, we analyse the reproducibility of submissions. Reproducibility is key to turning the benchmark into a resource for future fertility-prediction research. Achieving reproducibility,

however, is challenging [12]. In the part of PreFer based on survey data, we facilitated reproducibility by requiring participants to submit not their predictions, but their code and model. Here, we assess the reproducibility of PreFer submissions nearly two years after the data challenge ended to learn which submissions are reproducible. Apart from showing how reusable the PreFer submissions are, the analysis of the survey-based submissions also shows how effective the submit-the-code approach is, what the remaining problems are, and how reproducibility can be improved in future data challenges.

Finally, we examine which predictors the submitted models relied on and how much different feature sets (topics) contribute to predictive performance. Because importance is assessed in several independently developed models rather than one, we can distinguish predictors that matter more robustly from those that appear important in a single model. For the benchmark, this analysis documents the search that teams collectively performed over the feature space: which topics carried the most information relevant for predicting the outcome and which topics added little. For fertility research, it provides a ranking of predictors on a single, comparable scale, which the fertility literature has so far lacked.

Methodology of the PreFer data challenge

Data

Survey data

The survey data comes from the LISS Core study and the Background survey. The Core study includes ten longitudinal surveys – covering family, education and work, income, assets, housing, health, religion, social integration and leisure, personality, politics and values – fielded (bi)yearly starting in 2007/08 (see SI Table S1). All household members aged 16+ are eligible to participate in these surveys. The Background survey includes basic demographic and socioeconomic questions about all members of the household, even those not participating in the panel. Our study sample consists of LISS members aged 18-45 in 2020 who participated in at least one Core survey in 2007-2020 ($N = 6,877$); because of attrition, 2021–2023 outcomes were available for 1,382 of them (see [13]).

We randomly split the sample into a training set (available for data challenge participants) and a holdout set (reserved for evaluation and not available to the participants). The training set includes 6,418 people, of whom 987 have a known outcome (70% of the study sample with known outcomes). The holdout set includes the remaining 30% of the study sample with known outcomes; $N = 395$. Data challenges often use two holdout sets to avoid overfitting to the holdout [14]: a leaderboard set for evaluating intermediate models and a final evaluation set to evaluate the final submissions. Due to the limited sample size, we did not use a leaderboard set for the survey data. All the submissions were evaluated on the holdout set. However, we limited the number of submissions to five to minimise the risk of overfitting to the holdout set.

The training-holdout split is done by households to avoid data leakage, which would lead to suboptimal models and overly optimistic performance estimates that do not reflect real predictive ability [15, 16]. For example, partners can have the same values of household-related variables and the same outcome; a model can pick up on these similarities to make predictions even though these variables might be unrelated to fertility outcomes. The split is stratified by the

outcome to ensure the same proportion of observations of each class in the training and holdout data (see details about the split in [13]).

PreFer participants received (1) a main training dataset (merged Core surveys 2007–2020, the outcome variable, and selected Background variables), (2) a supplementary training dataset with the same variables but covering respondents outside the target group (excluding individuals who belong to the same households as people in the holdout to avoid data leakage) to potentially extend the training set (see [17]), (3) a merged monthly Background dataset 2007–2020 for the training set participants and their family members, and (4) machine-readable codebooks. Two equivalent datasets, but for the holdout – the main holdout dataset and monthly Background dataset – were accessible only to the organisers for evaluating submissions. In all prepared datasets, we replaced original IDs with pseudonymised IDs to prevent linking those outcome-window datasets that were available for download from the LISS data archive during PreFer. The sample sizes are presented in Table 1 and in SI section 1.

Register data

Register data comes from Dutch registers on population, taxes, education, employment, social benefits, housing, and neighbourhoods, managed by Statistics Netherlands (CBS) [9]. We selected all datasets potentially relevant to predicting fertility, except those containing highly sensitive personal information (medical prescriptions and treatments, healthcare costs). The selected datasets include longitudinal information – in most cases, with at least 10 years of coverage – about marriages and partnerships, children, education, employment, income, assets and benefits, households, neighbourhoods, and linkages between individuals based on family ties, household membership, co-residence, school attendance, and shared workplaces [18, 19] (see SI Table S2).

The target population is all Dutch residents aged 18–45 on December 31, 2020 ($N = 5,878,530$) (see SI section 2). We randomly split the target population into training (70%) and holdout (30%) sets, ensuring again that entire households were assigned to the same set to reduce the risk of data leakage. If married or registered partners lived in separate households, they were also placed in the same set. After the 70/30 training-holdout split, the holdout set was further divided into two parts (again keeping household members and married/registered partners together): one-third for evaluating intermediate submissions and two-thirds for the final submissions (see Table 1 and SI Figure S2). In addition, 10% of the training set was flagged as an evaluation set so that participants could test their models without replicating the splitting procedure themselves.

PreFer participants could use data up to the end of 2020 from the selected register datasets for everyone, including holdout and out-of-target individuals (younger than 18 or older than 45 in 2020). This was done to enable the construction of network characteristics, as network members of the training set individuals could appear in the holdout. The outcomes were withheld for all but the training set.

The selected datasets have different structures and units of observation, such as individuals, main breadwinners in a household, or addresses. To help participants develop initial models without linking many files, we also provided the PreFer Base dataset. It includes variables selected based on research on fertility determinants, mostly from 2020, on individuals, their

parents, and their married or registered partners. It covers both the target population and out-of-target individuals (see SI section 2).

Table 1. Datasets and participation in the two phases of PreFer

	Survey data (LISS)	Register data
Target population	6,877	5,878,530
Outcome known	1,382	All target group
Training set	6,418 (987 with outcome)	4,109,149
Supplementary training set	10,644 (2,806 with outcome)	11,597,479
Holdout set	395	1,234,843
Outcome prevalence	22%	15%
Teams with at least one valid submission	41	10
Valid submissions	69	11

Prediction task

The task of PreFer is to predict for Dutch residents aged 18–45 in 2020 who will have a new child (by birth or adoption) within the following three years (2021–2023), based on the data up to and including 2020. For the LISS data, we determined the outcome using survey responses on the number of biological and adopted children from 2020–2023. If missing, we supplemented it with Background survey data on children's birthdays to increase the size of the survey data with outcomes. For the register data, we used the data about legal parents (Kindoudertab, [20]) and birthdates [21] to identify whether target group members had at least one child born between 2021 and 2023. If a person has no children born in 2021–2023 listed in the Kindoudertab, we consider the outcome to be zero.

The proportion of people who had a child in 2021–2023 is 22% in the survey sample and 15% in the register sample. The difference arises because, among LISS participants with incomplete responses on the number of children in 2020–2023, we could only determine the outcome for those who had a new child during the years for which information was available, while missing data prevented certainty about non-births, leaving those without a new child underrepresented. For example, imagine a person who had data available only for 2021, and not for 2022 and 2023. If this person had a new child in 2021, the outcome was clear: this person had a child in 2021–2023. If this person did not have a child, we could not establish a definite negative outcome because the person might have had a child in 2022 or 2023.

We evaluate predictions with the F1 score for the positive class. It is the harmonic mean of precision and recall: precision is the proportion of predicted positive cases that are true positives, and recall is the proportion of actual positive cases that are correctly identified by the model. The F1 score helps to find a balance between precision and recall, i.e. identify people who had a child while at the same time trying to minimise the number of false positive predictions, which is especially useful for an imbalanced outcome.

Participation, phases, submissions

Overall, 314 people from different backgrounds (mostly demography, sociology, computational social science, data science, and computer science) registered to participate in PreFer, joined by 17 participants of the ODISSEI Summer School in Computational Social Science.

In the first phase of PreFer (1 April – 16 September 2024), participants predicted the outcome for the LISS holdout. To enable reproducibility, participants submitted a GitHub repository containing their model, preprocessing and prediction code, a file specifying required packages, a setup script (Dockerfile), and a description of their approach. Participants could use either R or Python. Submitted code was run inside a container, generating predictions on the holdout data. Before submission, automated GitHub checks tested whether the code could run successfully on synthetic data matching the structure of the holdout set. This ensured that the code would run in the same environment as the participants used themselves and that it would work on the holdout data.

Participants could submit five times. After each round of submission, participants saw their own score on the holdout data and anonymised scores of other submissions (without team names and links to submissions). In total, 153 participants from 43 teams made at least one submission (89 submissions overall). Of these participants, 147 from 41 teams produced at least one valid submission – one that ran successfully on the holdout data and was not labelled as a test submission. These 41 teams produced 69 valid submissions in the first phase.

The second phase, based on register data, ran from 1 July to 25 October 2024. It began once results from the first three rounds of the first phase were available. Due to strict data access restrictions, participation was limited to five teams selected based on their interim results in phase 1 and the potential of their approaches for the register data, plus seven Summer School teams. Several teams that were not selected continued working on the survey data.

Unlike the first phase, participants submitted predicted classes for the holdout set along with the trained model and code for preprocessing and training. The remote secure environment through which participants could access the register data prevented an approach equivalent to that used for the survey data. Teams submitted only once because most teams used an evaluation subset of the training data to evaluate intermediate models. Overall, 12 teams made 11 valid register-based submissions (two teams did not develop a working submission; another two teams produced one combined submission in addition to models that they produced separately).

Overview of submitted methods

In developing their predictive approaches, teams faced a range of modelling decisions and challenges. Some of these are familiar across most statistical learning tasks – for example, how to handle missing values, select features, and which algorithms to apply. Other decisions and challenges were shaped by the specific design of PreFer (though not unique to it), such as how to work with a relatively small sample size of the survey data, deal with an imbalanced outcome, or transform detailed longitudinal data into formats suitable for statistical modelling.

Teams approached these issues in diverse ways. While some strategies led to differences in prediction scores, many different approaches achieved similarly high performance – a phenomenon known as the Rashomon effect [22, 23], where multiple, distinct models can perform equally well. Some approaches did not achieve the top scores in the PreFer setting, but offer promising and potentially transformative ideas for the social sciences [24–26].

Below, we briefly review the approaches submitted to PreFer based on team descriptions and code. The list of submissions with scores, brief descriptions, and reproducibility results is available in Additional file 1 and SI Table S3. We use team names or GitHub usernames in italics to refer to submissions; team names refer to survey-phase submissions unless we note that a submission is register-based. We structure this review around two common stages in predictive modelling workflows: data preparation (including handling missing data, feature creation and selection) and statistical modelling.

Data preparation

Feature selection

PreFer datasets include thousands of variables, which is why a selection of variables was a first step most teams took. Some teams used a theory-driven approach – selecting variables manually based on their established relevance to fertility outcomes (see *danielvwijk*, *NextFinBaby*). Other teams used a data-driven approach or combined data- and theory-driven approaches.

A data-driven approach was done in different ways. Several teams selected variables based on feature importance from tree-based ensemble algorithms or using penalised logistic regression (e.g. *yannIcks*). *RosemberGuerra* used artificial noise variables to benchmark the importance of each feature. A LightGBM classifier was utilised across multiple parameter settings to evaluate the importance of each feature relative to created noise variables. A feature was considered informative if its importance consistently exceeded both noise variables. *Team Erasmus* identified best-performing features by calculating the information gain of each one when compared against a fertility intention question, highly correlated with the outcome. *Copenhagen Social Complexity Lab* gradually increased the number of variables after initial selection to see how many features are needed to achieve the best predictive performance. *BimBO* applied SWAG [27] – an algorithm for feature selection that follows a forward-step approach, beginning with a few variables and progressively adding more in each step, to identify a minimal set of features with high predictive power. In some submissions, feature selection was implicit: the model learned during training to assign near-zero weights to uninformative features (e.g. *SOC-555-CS*).

Several teams used pre-selection before doing the final data-driven selection: for example, keeping only variables from the most recent wave, removing all zero-variance predictors, predictors with more than a certain percentage of missing values, open-ended questions, date/time, and variables that showed a strong correlation with other variables in the dataset (e.g. *chrisrellama*).

Three teams combined theory- and data-driven approaches to variable selection. They began with hand-picking variables relevant to fertility, then supplemented or refined this set in a data-driven step, sometimes using internal model evaluations such as cross-validation. Examples of this data-driven step include using univariate models for iterating over every feature available to get a list of the most important features (*GAS*), applying regularised regression to identify other predictive variables (*Stork Oracle*, [17]), or using feature importance scores from an initial model to filter variables (*LYC-MRK*, [28]).

Finally, one team in the LISS phase and three teams in the register-based phase explored novel data transformation and statistical learning approaches that do not require variable selection. One of these approaches involves transforming tabular data into sequences representing individual life courses in a symbolic language, in which life events and their attributes become tokens ordered chronologically, which are then processed by neural networks to generate vector representations (embeddings) used for prediction. This approach was developed by *SudaMax* in the survey phase and by the *Copenhagen Social Complexity Lab* (not developed in full) and *SBU EUI* in the register phase. The second approach (*Cruiff*; [25, 26]) involves fine-tuning an openly available pre-trained large language model on "books of life" – natural-language life histories generated from the register data, covering demographic information and household and address spells. The main rationale shared by both approaches is that they can exploit detailed longitudinal data without reducing life courses to a limited set of features.

Creating features

Several teams generated scales to reduce dimensionality for conceptually related variables by averaging these features together or by using factor analysis. Generated scales include, for example, sexism, feelings toward current child, traditional attitudes about marriage/sexuality/reproduction, attitudes about traditional fertility, traditional motherhood and fatherhood, traditional marriage, and working mothers (*Stork Oracle*); religiosity, average net income, language proficiency, and partner status over time (*Bernoulli Stork Squad*); five components of personality according to the Big Five model (*GAS*); depression (*Pasprefer*); gender values, marital values, and anxiety (*treadmill*).

Teams also created additional features, often to make use of the longitudinal data without adding variables from many different years. New features can be grouped into:

Family and household structure and composition, and demographic and economic characteristics: examples from the survey submissions include age of the youngest child, household income per capita, average income, same-sex couples, or most recent values from the monthly Background dataset; from the register submissions – years spent in a household type with and without children, additional assets and house ownership variables.

Changes over time: in their survey-based submission, *GAS* included changes in marital status and income; *KinderComputer* used changes in drinking and smoking behaviour (possibly as proxies for fertility intentions), and *teamSGBD* focused on changes in housing situation, household income, education, and occupation. In the register-based submissions, *teamSGBD* included income trends and educational changes over time.

Responses and/or other data from the partner: for households where both partners participated in the survey, *Stork Oracle* merged in the partner's fertility intentions from 2019 and 2020 and the partner's answers to questions about how many children they have, which improved model performance [17]. In the register-based phase, information about partners came from linked administrative records – for example, partners' age, gender, country of birth, education, and employment (*ConceptionCoders*, *JustThe2OfUs*); one team additionally identified cohabiting partners who are neither married nor in a registered partnership (*Stork Oracle*).

Network features. Most register-based submissions used population network data in their submissions to construct different network features. *TeamSGBD* [29] constructed three personal networks per individual – neighbours (a random sample of 20 people living within 200 metres of the person's address), family members of the same generation (full, half-, and step-siblings and cousins), and up to 100 geographically closest colleagues under age 45. They transformed the variables of the PreFer Base file into network-level features by averaging them within each network (for continuous variables, the mean was used; for categorical variables, the proportion of network members in a given category). Other network features included counts of siblings and of siblings' or colleagues' young children (*DEPLearning*, *JustThe2OfUs*, *Code babies*) and second-order ties such as children of colleagues of siblings and of siblings of colleagues (*Code babies*). *Stork Oracle* included counts of different ties in the family network (parents, grandparents, siblings, nieces and nephews, relatives by marriage, etc.). *SBU EUI* included counts of network ties to family, household members, neighbours, schoolmates, and colleagues in the life sequences used to pre-train a transformer model.

Individuals' residential environment. Several register-based submissions used municipality-level information: election results, used to proxy political values and religiosity of the local context (*ConceptionCoders*, *DEPLearning*, *JustThe2OfUs*, *data storks*, *teamSGBD*), and municipal statistics such as population density and the proportion of residents aged 15-44 (*DEPLearning*), as well as proximity to childcare facilities (*ConceptionCoders*).

Dealing with low sample size and imbalanced data

Small training samples and a low share of positive cases are familiar problems in predictive modelling, and both were present in PreFer. With few observations, a model may fail to capture the underlying patterns [30], while class imbalance encourages a model that predicts "no child" for nearly everyone. Teams responded by enlarging the training data, rebalancing it, or adjusting how the model was trained and how its probabilities were converted into predictions.

Stork Oracle [17] enlarged the training sample through what they call time-shift data augmentation. Because panel surveys ask the same questions every year, the prediction task can be reconstructed for an earlier window: who had a child in 2018–2020 based on the information up to the end of 2017. The team computed the outcome for 2018–2020, dropped any features measured after 2017, and appended those observations as new rows. Some of the time-shifted data

rows represent earlier observations of the members of the original training set. Other rows are 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). This approach nearly tripled the original number of training observations with an available outcome.

YannIcks explored self-training and tri-training to exploit the large share of the sample without an observed outcome. Both methods assign provisional labels to unlabelled cases and add the most confident of them to the training set, iteratively [31, 32]. The approach scored poorly in internal evaluation and was not part of the final submission, but remains a promising route for using unlabelled cases.

Other teams rebalanced the training data directly using random oversampling of the minority class (*Ttliu00*, *ConceptionCoders*, *DEPLearning*, *data storks*), SMOTE [28] and ADASYN (*PRC Team*). Both latter approaches generate synthetic minority cases by interpolating between existing ones and their nearest neighbours, with ADASYN concentrating on regions where imbalance is most severe [33, 34].

Some teams dealt with the class imbalance during training or when converting probabilities into predictions rather than by modifying the training data. For example, *Copenhagen Social Complexity Lab* and *JustThe2OfUs* assigned greater weight to the positive class during training, which encourages the model to pay more attention to underrepresented cases. Several teams lowered the classification threshold below the default 0.5 to increase recall – in the register phase (*Cruijff*, *Stork Oracle*) or in both phases (*TeamSGBD*).

Missing values

Missing data was mainly an issue in the survey phase, though it also occurs in the register data: educational information, for instance, is recorded at lower rates for older people and for migrants. Most teams imputed missing values, except teams that used XGBoost, CatBoost, or LightGBM, which handle missing values internally.

The simplest strategies were mean, median or mode imputation and, for categorical variables, an explicit missing category. Several teams instead predicted missing values from other variables, using k-nearest neighbours, MissForest [35] (*BimBO*), MICE [36] (*tredmill*), IterativeImputer [37] (*Kawaii-Killer*), or truncated singular value decomposition (*UU Human data science*).

Two approaches were tailored to the data. *Chrisrellama* plotted selected features against age, found the relationship close to linear, and filled gaps by regressing on age. Several teams exploited the longitudinal structure of the survey by carrying forward the most recent non-missing value of the same variable (*GAS*, *Pasprefer*, *Team Erasmus*), sometimes adjusting for elapsed time, for example, as in the question about how many years from now the respondent wants a child.

Approaches to statistical learning

Model selection

In the survey phase, many teams used XGBoost, including the winning model (*Stork Oracle*); the rest were divided between other tree ensembles, logistic regression, and neural

architectures for tabular data such as ExcelFormer [38]. In the register phase, the winning model used CatBoost (*Stork Oracle*); two teams submitted the sequence- and language-model approaches described above (*Cruijff*, *SBU EUI*), and the remaining submissions again used tree ensembles, alongside penalised logistic regression and a voting ensemble combining several algorithms.

Teams made model selection choices in different ways: by using knowledge of an algorithm's properties (e.g. in the register phase, *Stork Oracle* chose CatBoost because it handles categorical variables well and requires little preprocessing), by submitting alternative models and comparing holdout scores, or through an internal validation set. *Bearly Predictable* automated the last of these with the R *tidymodels* package, tuning each model by repeated cross-validation, ranking models by mean F1 across folds, and evaluating the best on a held-out set. *Cruijff* and *SBU EUI* chose approaches that could leverage detailed longitudinal data [24–26].

BimBO used different models for different rows of the holdout data. They created an ensemble of random forest models. At the prediction stage, they checked for each holdout row which models could be applied: a model can be applied if none of the needed variables is missing. If no models could be applied, they used the imputed version of the row. The final prediction was the most frequent class among predictions from the applicable models.

Hyperparameter tuning

Approaches to hyperparameter tuning ranged from zero-shot tuning, which sets hyperparameters from dataset characteristics [39] (*yannlcks*), to Bayesian optimisation [40] (*chrisrellama*). Where tuning used cross-validation, folds were typically stratified on the outcome, so that the small number of positive cases was distributed evenly rather than concentrated in a few folds. Tuning did not always pay off: for the best-performing register-based model, exploring tuning options produced no meaningful gains and the final submission used default hyperparameters (*Stork Oracle*).

Strategies of internal evaluation before submission

Most teams evaluated models internally with k-fold cross-validation and stratified on the outcome. Some teams also grouped on household ID so that members of the same household stayed within a single fold. Internal scores were often optimistic relative to the holdout, likely because of overfitting or a lucky random split. Teams that noticed the gap between the cross-validated and holdout scores responded in different ways: *Copenhagen Social Complexity Lab* added L2 regularisation to XGBoost to avoid overfitting; *GAS* drew ten independent random splits, each holding out 30% of the data for validation while preserving the class proportions, and found the resulting scores matched the holdout score more closely, and *Bearly Predictable* used cross-validation for tuning and a distinct internal validation set for model comparison.

Reproducibility of the PreFer submissions

Survey-based submissions

In the survey-based part of PreFer, we facilitated reproducibility by requiring participants to submit their code and model instead of predictions (see Participation, phases, submissions). By

reproducibility we mean “computational reproducibility” [41], or, in our case, a researcher’s ability to recreate the model’s score using the raw data and code provided by the original author.

The submit-the-code approach required additional effort from participants. They had to ensure that their code would run on holdout data to which they had no access, relying only on the training data and synthetic data (with the same variables and set of unique values as in the holdout) for debugging. Participants also had to structure their files so that the whole prediction pipeline could be executed, rather than simply submitting a file with predictions. The approach also required effort from the organisers, such as preparing a template GitHub repository, submission instructions, and developing a submission platform (as existing competition platforms such as CodaLab could not support our workflow with the data that is not publicly available).

Whether the submit-the-code approach actually guarantees reproducibility remains an open question. Even when code, models, and computational environments are preserved, reproducibility may deteriorate over time, for example, due to incomplete environment specifications, changes in operating systems, libraries becoming deprecated, or random differences during model training (e.g. due to random initialisation of model parameters or random sampling of training data).

To assess the extent to which the submit-the-code approach did in fact result in reproducible submissions, we analyse two kinds of computational reproducibility. The first kind is whether the submitted model still produces predictions on the holdout data, and whether these predictions result in the same score as during the original PreFer evaluation. We refer to this kind as inference reproducibility. The second kind is whether a submission can be rerun from scratch: whether the training script produces a model that generates predictions on the holdout data, and whether these predictions again match the original score. We refer to this as training reproducibility.

During PreFer, all submissions were evaluated by Eyra (the software company that developed the submission platform) using the saved model in Docker containers. We reran all survey-based submissions in June-September 2026 – nearly two years after the end of the data challenge (September 2024). We used a script that recreates the original evaluation process: for each submission, it builds a Docker image from the submitted Dockerfile, creates a container, and runs the submission inside that container. Submissions were retrieved by cloning each participant’s GitHub repository at the exact commit hash that was submitted; two repositories that had since been deleted could not be cloned at their commit and were downloaded manually. Because inference and training can be non-deterministic, each submission was run five times for the test of inference reproducibility and three times for training reproducibility (fewer than for inference because retraining requires much more time). On each run, we recorded whether the submission ran successfully or failed; for successful runs, we recorded the score and compared it to the original leaderboard score reported by Eyra.

Both analyses of inference and training reproducibility were carried out in stages. In the first stage, every submission was run exactly as submitted. In the second stage, submissions that had failed to run were rerun after fixes to the individual repositories. If a submission ran but produced a different score, we checked the code for potential reasons. We recorded every fix and classified them as light or substantive. Light fixes were confined to code outside the submitted functions for data preprocessing, prediction, and model training. There were four kinds of light

fixes: (1) pinning package versions or the R version to those used at the time of development (based on the versions current in April 2024 at the start of PreFer); (2) removing absolute or local paths from the training script and from the environment specification; (3) removing preprocessing or prediction code that did not belong in the training script; and (4) adding missing package imports to the training script. Substantive fixes were changes inside the preprocessing, training, or prediction functions themselves – for example, removing rows with a missing outcome from the training data after they threw an error. We made such fixes only when they followed logically from the rest of the code. For instance, we removed rows with a missing outcome from the training data because they cannot be used for supervised training and there was no code provided that dealt with missing values in the outcome. We list all the fixes to individual submissions along with the reproducibility status and scores in Additional file 1; full error messages and training logs are available in the replication package on Dataverse.

We label a submission as not reproducible in two cases: (1) when the fixes required to make a submission run would have forced us to make our own modelling decisions that do not follow from the rest of the code and (2) when the training code was missing.

Figure 1 shows the results of the reproducibility analysis. Inference reproducibility was good: 58 out of 69 submissions ran as submitted; 8 submissions ran after light fixes; 2 after substantive fixes. All these submissions, except one with a substantive fix, reproduced the original F1 score (that is, the original and reproduced scores were identical to four decimal places). One submission did not run, despite the fact that it worked and produced a score at the original evaluation.

Training reproducibility was worse, both in terms of making the submission run and in terms of reproducing the exact F1 score. Only 22 submissions ran as submitted; 23 ran after light fixes and 15 after substantive fixes. Four submissions did not reproduce because modelling decisions needed to be changed to make the submission run, and 5 because training code was missing.

Out of the 60 submissions that ran (either as submitted or after fixes), 23 reproduced with a different score; 4 of them had non-deterministic training (seeds not set for data preprocessing or training), and one had non-deterministic inference. In other submissions that did not reproduce the original score, the training script provided likely differed from the training script used for producing the saved model; or the difference in score could have resulted from an environment drift (difference in package versions, hardware, system architecture used for training the original model vs used for retraining the model). Submissions requiring substantive fixes were less likely to reproduce their original score: 5 of 22 submissions that ran as-is produced a different score, 7 of 23 after light fixes, and 11 of 15 after substantive fixes. The winning model was also affected – the score of the retrained model was higher than the score of the saved model. Together with its authors, we established that this was likely caused by a different seed used to train the original model.

However, most of these differences were small. Among the 23 retrained submissions whose score changed, the median absolute difference from the original F1 score was 0.017 (Figure 2), and 52 of the 60 submissions that ran reproduced the original score either exactly (37) or within 0.03 (15). A few gaps were large: F1 changed by more than 0.05 for five submissions, and by more than 0.10 for three of them. Two of these had an original F1 of zero but scored 0.12 and 0.66 after

retraining. Run-to-run variation was small for four of the five non-deterministic submissions, but for one of these submissions performance ranged from 0.52 to 0.66 across identical retraining runs, illustrating that differences caused by non-determinism in training can be quite large. Despite some changes after retraining, the winning submission was the same.

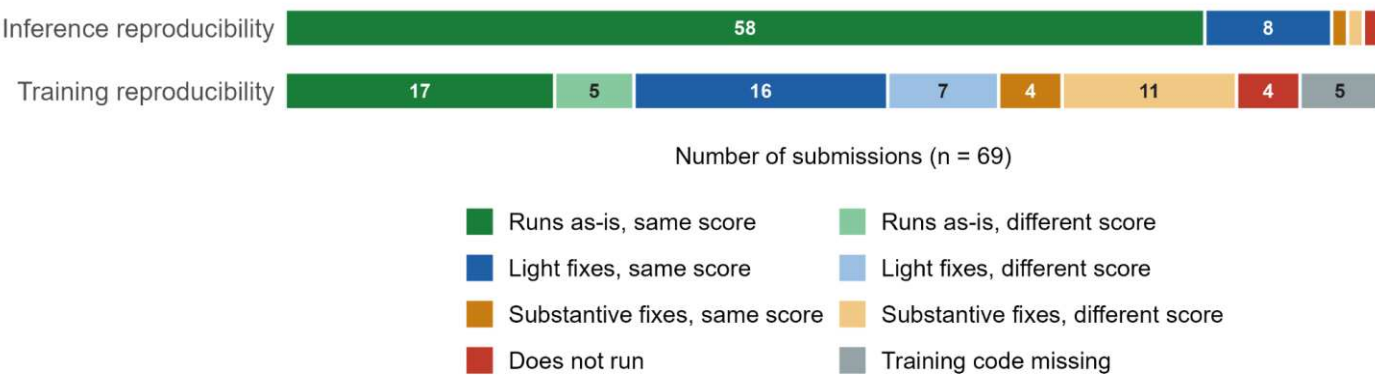


Figure 1. Number of survey-based submissions with different reproducibility statuses. Lighter shade means the submission was reproduced but with a different score. The total number of survey-based submissions is 69.

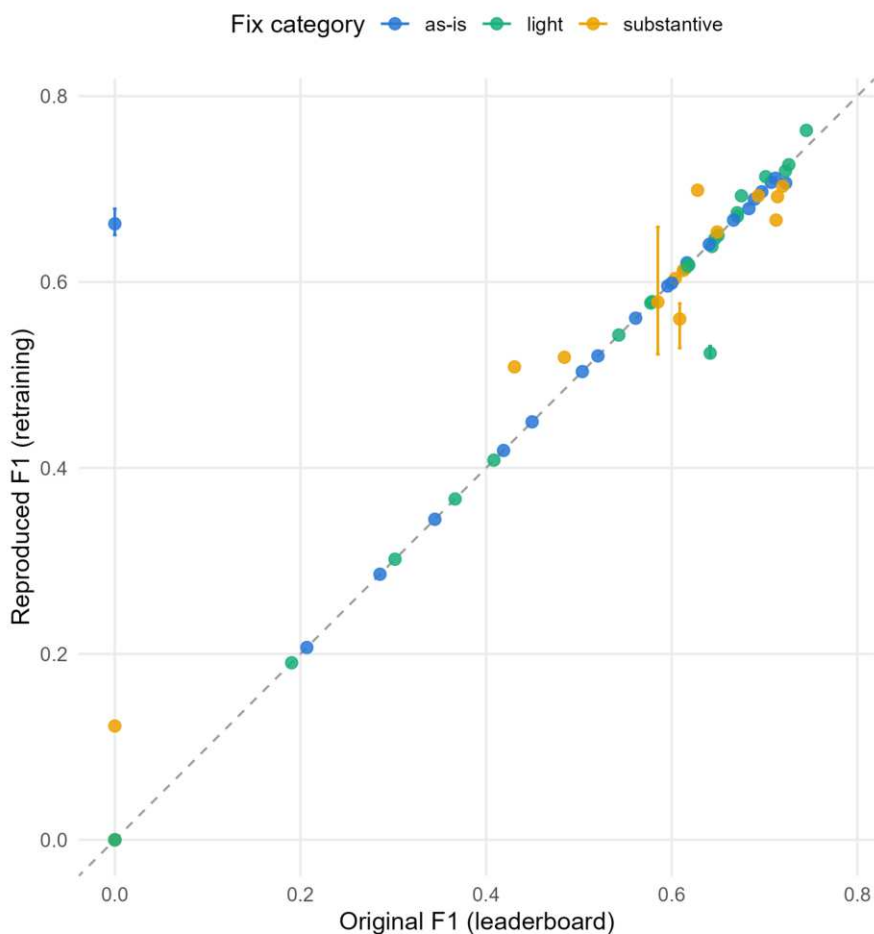


Figure 2. F1 scores of retrained models versus original scores for the 60 survey-based submissions that could be retrained. Submissions on the dashed line are reproduced exactly, and the vertical distance from the dashed line is the reproducibility gap. Colours represent the fix required to make the submission run: as submitted (as-is), light, or substantive. Submissions whose three training runs gave identical scores are shown as a single point; for the five submissions with varied scores, the point shows the mean across three runs and the whisker shows the minimum and maximum across the three runs.

That training reproducibility was worse than inference reproducibility is unsurprising: during the challenge, the training script was never executed and never checked. We did not verify, for example, whether a training script present in the repository differed from the template, or whether the required packages were imported. Even so, most submissions reproduced from the training script with a score identical or close to the original, which we consider a reasonably good outcome.

Part of this likely owes to the design of the submission itself. We provided a template for the training script and asked participants to implement a specific standardised model-training

function (which most teams did) rather than just providing a training script in any form, to fit the model by calling that function, and to commit the resulting saved model to the repository.

Training reproducibility can be improved by running evaluation from scratch – using each submitted training script to refit the model at evaluation time. In our reproducibility study, retraining every model once took 18 hours in total (on a machine with an Intel Core i5-12500T, 16 GB RAM, no GPU, with models trained sequentially), or about 18 minutes per model on average, which is manageable. If the number of submissions is much larger, retraining at evaluation time might become prohibitive. In Discussion, we list much less time-consuming measures to substantially improve training reproducibility.

Register-based submissions

In the register-based part, participants submitted their code, their saved model, and their predictions, and it was the submitted predictions that were scored. Inference reproducibility therefore means something slightly different here than for the survey-based submissions: it means testing whether the saved model, run on the holdout data, produces predictions whose score matches the score of the originally submitted predictions. Training reproducibility has the same meaning as before – whether the training script produces a model that, run on the holdout data, reproduces the original score.

Three features of the register-based submissions complicate this analysis. First, 3 out of 11 submitted models required the ODISSEI secure supercomputing environment, access to which we could not arrange for this reproducibility study. Second, 6 models developed at the ODISSEI Summer School (see Participation, phases, submissions) were originally trained and scored on datasets that resulted from an earlier train-test split. In that first split, individuals who were married or in a registered partnership but lived in different households could be assigned to different sets; we later corrected this so that partners living in different households are kept together, avoiding potential data leakage. The original scores for the affected submissions were therefore computed on data constructed slightly differently from the data used here, so their original scores are not a clean target for comparison. Third, for the same 6 submissions, the originally submitted models are no longer available because of a transition to a different project space in the CBS remote access environment. This does not allow testing inference reproducibility for these models.

We analysed reproducibility of 8 of the 11 register-based submissions. The reproducibility results with details are provided in SI section 4. In total, of the eight submissions analysed, three reproduced without code changes – the winning submission with an identical score, one with a practically identical score, and one with a substantially higher score – and five reproduced after minor changes, with scores identical or close to the original (differences up to 0.03).

Reproducing the register-based submissions required more effort than the survey-based ones. We recommended but did not enforce a submission format (except for the function that produced predictions), so different teams used different ways of preprocessing the data and training the model. Because of this, we could not run all submissions automatically as we did with the survey-based models – each training script then had to be run separately. For most submissions, we first had to locate the scripts needed to produce the additional variables and run them.

Reusing PreFer

The fact that most PreFer submissions remain reproducible, at least for now, is what makes them reusable. Reusing PreFer, however, must follow several recommendations. With the holdout data downloadable and no enforceable rules on the number of submissions, the holdout is no longer a holdout in the strict sense: nothing prevents evaluating many model variants against it and reporting the best, which can lead to overfitting. We recommend that future users develop and select models on the training data using internal validation (following the way we originally split the data, e.g. by households to avoid leakage), and evaluate on the holdout once; if evaluated several times, report the number of holdout evaluations performed. Improvements can also be tested on future data for robustness: new waves of LISS and new cohorts in the register data permit constructing new evaluation sets with later prediction windows on which improvements can be tested.

Predictors and topics that drove predictive performance

Teams in PreFer took very different approaches to feature selection – theory-driven, purely data-driven, and mixed (see Overview of submitted methods). This gives an opportunity to study the importance of predictors across independently developed models rather than in a single one. Because feature importance depends on modelling choices, assessing it across several models gives a more robust picture than any one model can [23, 42, 43]. Comparing models with different feature sets, however, requires additional preprocessing.

We study two related questions. First, which features did the models rely on when making predictions? We answer this with global SHAP values, computed for a large set of submissions. Second, how much did each group of features contribute to predictive performance? We answer this with ablation experiments – retraining models with topics added or removed. Because this analysis requires retraining the model many times on different feature sets, we restricted it to the top-3 models for each data source.

Reliance on features: SHAP values

SHAP values decompose a model's prediction into per-feature contributions; they are model-agnostic, relatively fast to compute, and widely used to quantify feature contributions in machine-learning models [44]. Among submissions for which SHAP values could be computed, we selected all survey-based submissions that outperformed a fertility intentions baseline (a logistic regression with a single fertility-intentions variable) ($N = 21$ out of 69 survey submissions), and the top-5 register-based submissions. We restricted the survey analysis to submissions beating the fertility intentions baseline, as a model that does not outperform this baseline has most likely not learned from other variables, so its feature importances are uninformative about which features predict fertility. The register data have no comparably strong baseline, so we took roughly the same proportion of submissions.

Feature importances are not directly comparable across submissions for two reasons. First, the overall magnitude of global SHAP values depends on the model (e.g. whether the model outputs probabilities or log-odds). We therefore normalised each feature's global SHAP value within its submission by dividing by the sum over all of that submission's features. For each submission, this produced a set of importance shares summing to one, which are comparable across different models. Second, because teams selected different feature sets, the same construct might

appear together with different numbers of correlated variables, and importance can be split among such correlated features. We therefore grouped features into broader topics (mainly based on the dataset they came from) – 27,224 variables in the selected survey-based submissions into 16 topics, 2,336 variables in the five register-based submissions into 13 topics – and analysed importance at the topic level. A topic's importance in a submission is the sum of the normalised importances of its features. We then ranked topics by their mean share across submissions.

Figure 3 shows the topics the models most relied on, ordered by mean share of importance across submissions, and the importance of individual features within each topic. In both survey- and register-based models, most topics showed the same shape: a dense base of features with little or no importance and a short tail of a few high-importance features.

In the survey-based submissions, fertility intentions dominated: this topic accounted on average for more than a third of total importance (mean share 0.35), nearly twice the share of any other topic. Partnership was second (0.19); it contained all variables on partners and partnerships including the variables that also describe household composition, e.g. household types (single household, partnered household with children, etc.). Age & gender, income, household composition, social integration and leisure formed a third group with similar shares (0.05–0.07); the remaining topics contributed less.

For the register-based submissions, the order was similar to the survey ranking (but excluding fertility intentions). Partnership and household characteristics came first (with mean shares 0.25 and 0.21). The second group includes several topics with comparable average importance – age and gender (0.14), education and job (0.11), income and assets (0.10), family network (0.10). Other topics contributed on average smaller shares.



Figure 3. Predictive importance of individual features across submissions, grouped by topic. Each point is a feature used in at least one submission; its horizontal position is the maximum importance across the models that included it, and its size is proportional to the number of models in which it was included. Topics are ordered by their average importance across submissions. (A) Survey submissions: 21 submissions that outperformed the fertility-intentions baseline and for which we could calculate SHAP are included; (B) Register submissions: top 5 submissions (among those for which we could calculate SHAP) are included.

Figure 4 shows the 30 individual features with the highest maximum importance across the submissions that included them. In the survey data, this set is dominated by measures of fertility intentions and by structural, life-stage aspects of partnership (having a partner, cohabiting, years living together), together with basic demographics and income. In the register data, the top features were similarly mostly timing and structure variables: person's and partner's age, the start dates of

the most recent civil status and of the first household with a partner, the partner's birth year, the number of co-parents, household composition, and the age of the youngest child. Two less expected features appear: the number of nieces and nephews, especially those under three years old, which is suggestive of social influence or biological predispositions, and municipality code. Municipality characteristics such as religious voting, by contrast, were not important in any model. This suggests place matters, but the aggregate characteristics included in these models were not the most relevant predictors.

The figure also shows large variation in the importance of the same variable across models. Part of this variation might reflect different preprocessing – variables with the same name may have been transformed differently by different teams – but it also illustrates why feature importance is better studied across several models than in one.

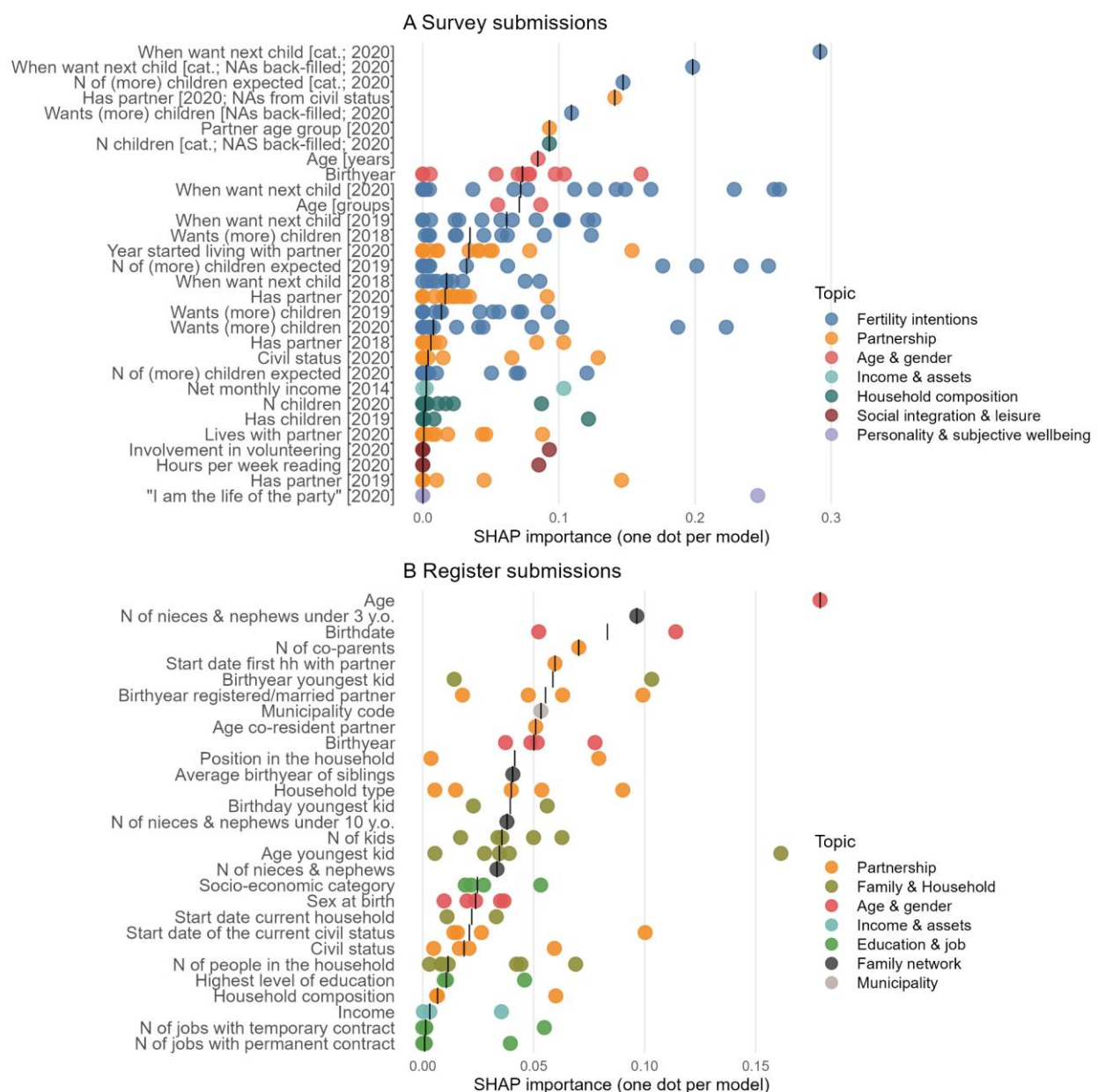


Figure 4. The 30 features with the highest maximum importance across the submissions that included them: (A) survey submissions, (B) register submissions. Each row is a feature; dots show its normalised global SHAP importance in each submission that included it, and the vertical bar shows the median across submissions. Features are coloured by topic and ordered by median importance.

Contribution to predictive performance: ablation

SHAP values show which variables a model relied on; they do not show how much a group of variables actually contributes to predictive performance. To measure this, we ran two ablation analyses on the top-3 retrainable models for each data source. In the first analysis, we added topics one at a time to a minimal model containing only age and gender, so that each retrained model contains two topics. This measures the total signal a topic carries, including whatever it shares with other topics. In the second, we removed topics one at a time from the full model. This measures a topic's unique contribution.

Survey-based models

The predictive signal is concentrated in three topics: fertility intentions, partnership, and family and household composition. In the best survey-based model, adding fertility intentions alone to a minimal model with age and gender raises the F1 score from 0.21 to 0.70 – within 0.06 of the full model (0.76); partnership raises it to 0.52 and family and household information to 0.45. Housing, health, migration background, and education and job form a group of smaller contributions.

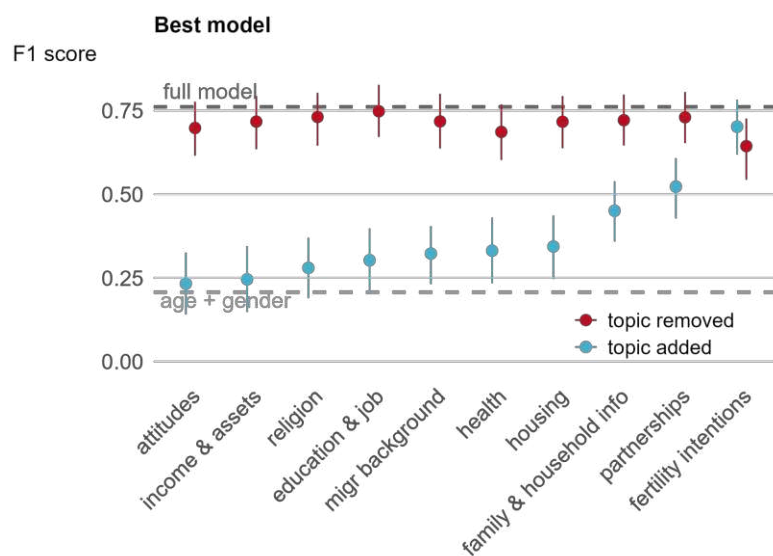
Once intentions are in the model, no other topic contributes much: removing any single topic other than intentions from the full model leaves performance nearly unchanged, even for topics with substantial total contributions such as partnership (their outcome-relevant information seems to be contained in the intentions measures). Intentions are also the only topic with a substantial unique contribution: removing them from the full model reduces the F1 score from 0.76 to 0.65, the largest drop for any topic. Intentions thus contain information about the outcome that is not fully recoverable from the other variables included in the model.

Health showed a moderate unique contribution (a drop from 0.76 to 0.69), which may reflect a close proxy of the outcome in the health module, such as a question about gynaecologist visits, rather than health status itself. Attitudes (feelings toward one's children, gendered religiosity, and attitudes to gender roles, marriage, parenthood, working mothers, and sexism) contributed little when added to age and gender but showed a drop when removed from the full model (0.76 to 0.70); this asymmetry may be noise, or may indicate that attitudinal variables become informative only in combination with contextual information such as partnership.

We also analysed the minimum set of easiest-to-measure topics in the best model that gets close to the full model's performance. We retrained the model but only with age, gender, persons' and partners' fertility intentions, structural partnership characteristics (marital/partnership status, duration), partners' characteristics (age, gender), and number of children in the household. This gave an F1 score of 0.74 – compared to 0.70 for intentions plus age and gender and 0.763 for the full model (score achieved at retraining). This means that all remaining variables (mostly economic and attitudinal) together add only around 0.023.

The second- and third-best models confirm the conclusions about intentions: the fertility-intentions-plus-age-and-gender model is the closest to the full model; removing any other topic barely changes performance when intentions are kept; and no other topic fully compensates when intentions are removed. Partnership and family and household also rank second and third in these models, though their contributions are more similar to those of other topics – probably because the best model includes more partnership variables, including the partner's own responses [17].

The ranking of the remaining topics, and whether they add anything beyond age and gender, differs across models. Housing improves on the age-and-gender baseline in the best model but not in the second-best. This may reflect the variables included: the best model used the number of rooms, dwelling type (owned, rented, etc.) and urbanity of the place of residence, whereas the second-best used dwelling type only. Differences also arose where near-identical variables were included: education and job variables improved predictive performance beyond age and gender in the best model but not in the third-best, although the employment variable was the same and the education variable almost the same. One explanation is the larger training sample of the best model (see [17]): the contribution of education and job is small, and a weak signal is easier to detect with more observations. Alternatively, these variables may have interacted with other features present in one model but not the other.



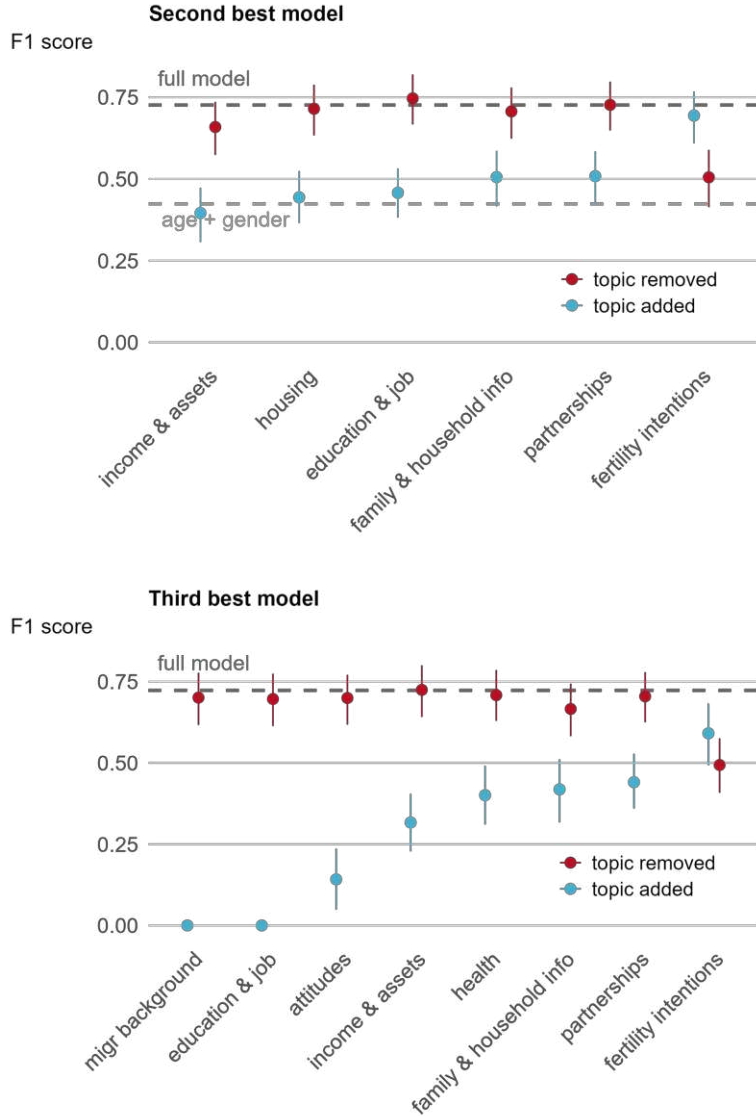


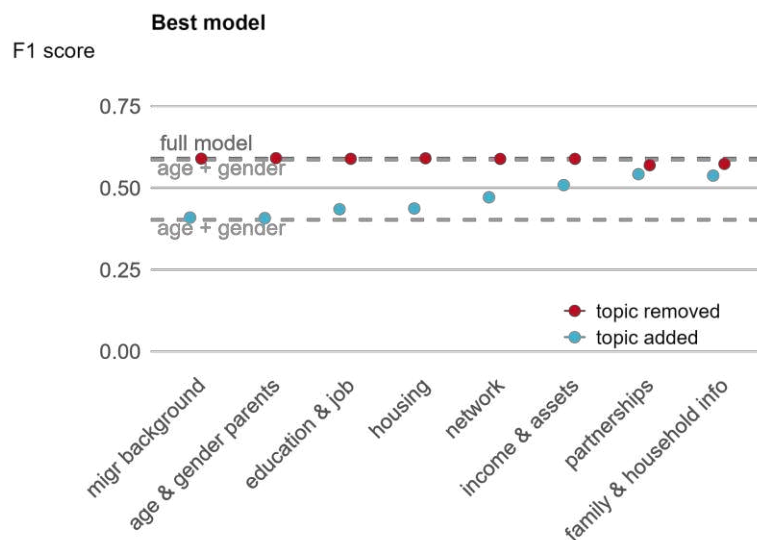
Figure 5. Contribution of different topics to predictive performance in the top-3 survey models. Blue colour shows the performance of the model that only includes age and gender plus one additional topic. Red colour shows predictive performance when one topic is removed from a full model. The third-best model does not show the F1 score for the age and gender model because it is zero. Error bars show 95% confidence intervals calculated via bootstrapping of the holdout.

Register-based models

In the register-based models, the predictive signal is concentrated in partnership, family and household, and income and assets (Figure 6). In the best register-based model, adding either family and household or partnership variables alone to the age-and-gender model raises the F1 score from 0.40 to 0.54 – within 0.05 of the full model (0.59); income and assets raise it to 0.51, and family network to 0.47. The remaining topics form a band of small or near-zero total contributions.

Removing any single topic from the full best model leaves performance nearly unchanged, even for partnership and family and household: the information they carry about the outcome is also present in other topics (for example, having a partner is recorded in household composition too). No topic has a large unique contribution, unlike fertility intentions in the survey data. The minimal set of topics is small, as in the survey data: retraining the best model with only age, gender, partnership and partner characteristics, and household characteristics gave an F1 score of 0.57 (vs 0.59 for the full model).

The second- and third-best models confirm that partnership, household characteristics, and income contribute most. As in the survey data, the ranking of other topics differs across models – most noticeably for family network. Family network variables contributed noticeably in the best model but almost nothing in the other two. The likely cause is that different models included different family network features. The best model included the family network size (counts of many ties, such as parents, grandparents, grandchildren, siblings, stepsisters and stepbrothers, nieces and nephews, aunts and uncles, and relatives by marriage). The third-best model included only features about siblings and their children, and the second-best included mainly network composition (e.g. mean income of the network) rather than size.



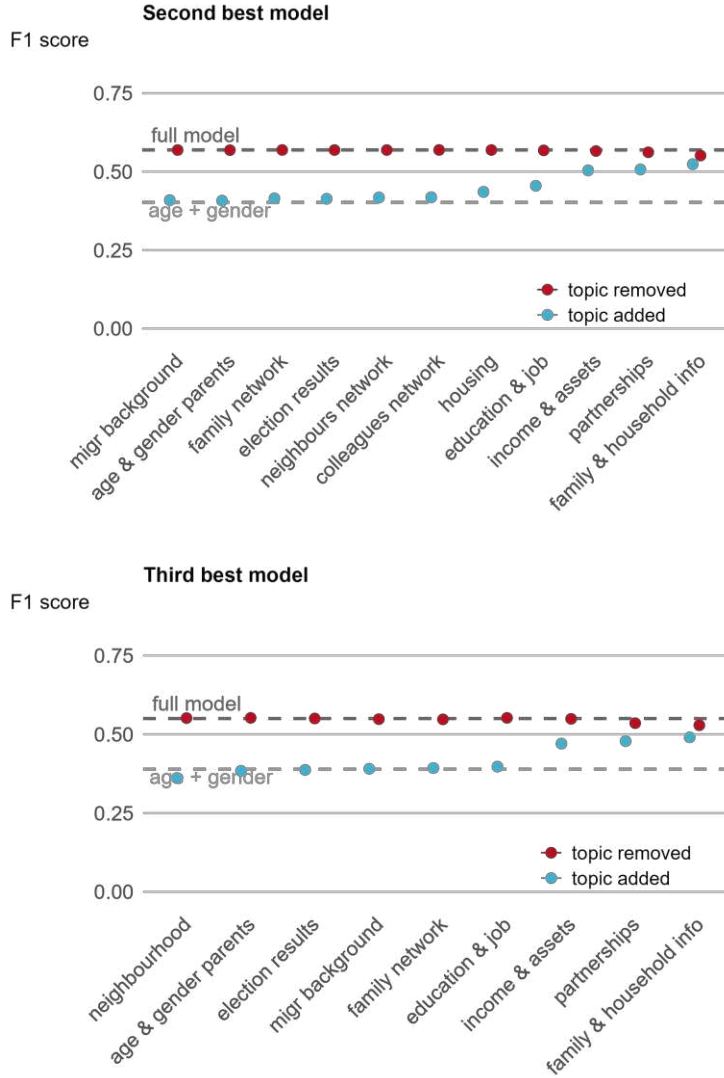


Figure 6. Contribution of different topics to predictive performance in the top-3 register models. Blue colour shows the performance of the model that only includes age and gender plus one additional topic. Red colour shows predictive performance when one topic is removed from a full model; 95% confidence intervals were calculated via bootstrapping of the holdout but are not visible because they are very small.

Discussion and conclusions

This paper documents the PreFer benchmark as an instrument for cumulative fertility research. We described the methodology, reviewed the approaches that teams took to deal with the common challenges of predictive modelling in social science, analysed which submissions still run and can be built upon, and characterised which topics of predictors carried most of the predictive signal and which contributed only marginally.

Reproducibility analysis showed that the submit-the-code approach produced submissions that remained largely reproducible nearly two years after the challenge closed. When the saved

model was used (inference reproducibility), nearly every submission ran and returned its original score. Reproducibility was not achieved for all submissions even in this case, though – when code, models, and environment specifications are preserved – which illustrates how difficult computational reproducibility remains in practice. Rerunning submissions from scratch (training reproducibility) was substantially less reliable: a minority retrained without any changes, a substantial share of the rest required fixes, and a sizable group produced a different score once retrained. The asymmetry between inference and training reproducibility is caused by the fact that the saved model and prediction pipeline were executed and checked during the challenge, but the training script never was. Encouragingly, about half of the failures of training reproducibility resulted from a small set of preventable causes: unpinned package versions, missing package imports, and local paths. The fact that most submissions were reproducible either as-is or after light fixes enabled us to do many different kinds of analysis in this paper and in [11].

Taken together, the results support requiring code and models rather than predictions alone. The approach imposes real costs on participants and organisers and does not deliver reproducibility automatically – but the gap between what it delivered and full reproducibility was largely attributable to identifiable and preventable causes.

Several recommendations for improving reproducibility follow from our analysis. First, most of the issues that caused submissions to fail to run as-is can be detected automatically. A validation step run at submission time could flag missing imports, local paths, and a training script identical to the template (which signals that the real script is missing), returning a warning or error. Second, improving the templates and instructions would prevent many failures. Most light fixes involved pinning R or package versions, which participants likely omitted because our R template did not include them; imports were omitted because the template training script did not import any packages. We also did not document how training scripts would be executed, which is why some participants might have left local paths in them; the instructions should state the execution context of the training script explicitly. More detailed instructions on setting seeds for both preprocessing and training could reduce non-determinism. Finally, one full training run can be executed at submission, but this requires much more time.

The feature importance analyses showed that in the survey-based submissions, predictive signal was highly concentrated. Across submissions built on different feature sets, fertility intentions together with basic partnership and household information contributed the most predictive ability – performance was close to the full model – and the remaining rich set of demographic, economic, and attitudinal variables added comparatively little. Our analysis also showed that fertility intentions capture the signal contained in most other variables: removing any single topic other than intentions from the full model leaves performance nearly unchanged. This pattern is consistent with the standard theoretical view of intentions as a proximate determinant that integrates upstream circumstances – partnership situation, existing children, housing, background [45]. Despite the instability and measurement error of reported intentions [46], they remain by far the strongest predictor of short-term fertility outcomes available. At the same time, predictions remain far from perfect, highlighting the gap between intending a child and having one even for a horizon as short as three years.

Register-based models offer a complementary setting for the question of relative predictor strength: register data lack intentions, so these models show which observable life-course topics carry the predictive load when intentions are unavailable. Here the signal was also concentrated.

Partnership and family and household characteristics – having a partner, the duration and legal form of the partnership, the number and ages of existing children – each alone recovered most of the attainable performance, with income and assets third and family network contributing in the best model. Age, gender, partnership and household information together were almost as good as the full model. Two further register-based findings deserve mention. Municipality identity was predictive while aggregate municipality characteristics were not, suggesting local heterogeneity in fertility behaviour that the available contextual variables do not capture. The contribution of family network likely depended on how the network was measured – size, composition, or a subset of ties – which illustrates the importance of studying many different network measures [47].

The main limitation of the analysis of predictive contribution is that predictive contribution is not causal importance. That a topic shows no unique signal means that it carries no additional predictive value beyond other variables, not that it plays no role in fertility. Likewise, that a topic predicts little on its own does not make it irrelevant: its role may be small, confined to particular subgroups, dependent on combinations with variables not in the model, or detectable only in a larger sample. Feature-importance results in general – whether a topic ranks high, or improves over a baseline – depend on the model, the other variables present, and the data at hand. Our own analysis provides an illustration: education and job variables improved performance over the baseline in one model but not in another, even though the two models used similar variables from this topic. This is a reminder that feature importance needs to be interpreted with caution and ideally studied based on several models.

These results carry several implications. For demography and fertility research, we show that intentions together with partnership form the minimal set that recovers most of the attainable predictive performance; this can be useful for short-term fertility forecasting. This can also be useful for policy and planning, as it shows the value of measuring fertility intentions. We also identified the most and least predictive topics; topics with low predictive ability are unlikely to be major independent drivers of who has a child within three years in this population. For survey design, the same finding can be useful for constructing a very short instrument that captures only the strongest predictors of short-term fertility.

The predictability we established in PreFer may change as methods, data, cohorts, and contexts change. Approaches that did not win here may improve. For example, representing life courses as sequences and processing them with language models did not outperform gradient boosting in our setting, but these methods might improve with more data and computing resources in a way that tree ensembles do not. New determinants of fertility, as they are proposed, can be tested by their out-of-sample gain over the PreFer results, which is a far more demanding test than whether a coefficient deviates from zero. New LISS waves and later register cohorts allow the same task to be repeated for later prediction windows, showing whether births are becoming easier or harder to predict as fertility behaviour itself changes. PreFer gives a common standard to make all such changes measurable.

Declarations

Availability of data and materials

The LISS panel data is 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 SANE <https://odissei-data.nl/facility/secure-analysis-environment-sane/>. The register data is not publicly available; access is only granted for scientific purposes to vetted researchers affiliated with authorised institutions. Academic institutions can apply for authorisation through Statistics Netherlands. The code to produce the outcome variable, reproduce the train-test split, and prepare the Base dataset is available at the same page in the LISS data archive. The PreFer Base dataset is available via the CBS Data Storage <https://odissei-data.nl/facility/cbs-data-storage/>. The submission files, scores and metadata for all PreFer submissions are available at Dataverse <https://doi.org/10.34894/WA6XSE> [to be published in October 2026].

Code availability

Code and instructions to reproduce the analysis presented in this paper are available at Dataverse <https://doi.org/10.34894/WA6XSE> [to be published in October 2026].

Conflict of interests

There is no conflict of interest to declare.

Funding

This work is supported by a VIDI grant (VI.Vidi.201.119) from the Netherlands Organization for Scientific Research (NWO) to GS. In this paper, we use non-public microdata from Statistics Netherlands (CBS). We also use data from the LISS (Longitudinal Internet Studies for the Social Sciences) panel administered and managed by Centerdata (Tilburg University, The Netherlands) and from the LISS panel data linked to the CBS microdata by Statistics Netherlands. Funding for the panel's ongoing operations comes from the Domain Plan SSH and ODISSEI since 2019. The initial set-up of the LISS panel in 2007 was funded through the MESS project by the Netherlands Organization for Scientific Research (NWO). Access to the CBS microdata, the ODISSEI Benchmark Platform, the ODISSEI-SICSS Summer School, and the development of the LISS harmonised dataset are financed by the ODISSEI Roadmap Project financed by NWO, grant number 184.035.014.

Ethical approval and consent to participate

This research was approved by the ethical committee of sociology at the University of Groningen (SOC-2425-S-0002).

Consent for publication

Not applicable.

Author contributions

All authors contributed to the design and implementation of the study. Elizaveta Sivak and Gert Stulp conducted the analysis and wrote the paper. All authors reviewed the manuscript.

Acknowledgements

We are grateful to all PreFer participants. We thank SURF (www.surf.nl) for the support in using the National Supercomputer Snellius. This work was supported by the Netherlands eScience Center. Results based on calculations by Elizaveta Sivak in project number 9424 using non-public

microdata from Statistics Netherlands. Under certain conditions, these microdata are accessible for statistical and scientific research. For further information: Microdata: Conducting your own research <https://www.cbs.nl/en-gb/our-services/customised-services-microdata/microdata-conducting-your-own-research>.

Abbreviations

CBS: Centraal Bureau voor de Statistiek (Statistics Netherlands)
LISS: Longitudinal Internet Studies for the Social Sciences
MICE: Multivariate Imputation by Chained Equations
NWO: Nederlandse Organisatie voor Wetenschappelijk Onderzoek (Dutch Research Council)
ODISSEI: Open Data Infrastructure for Social Science and Economic Innovations
PreFer: Predicting Fertility in the Netherlands (the data challenge)
RAM: random-access memory
SANE: Secure ANalysis Environment
SHAP: SHapley Additive exPlanations
SICSS: Summer Institute in Computational Social Science
SMOTE: Synthetic Minority Over-sampling Technique
SSH: Social Sciences and Humanities
SWAG: Sparse Wrapper AlGorithm

References

1. Jumper J, Evans R, Pritzel A, et al (2021) Highly accurate protein structure prediction with AlphaFold. *Nature* 596:583–589. <https://doi.org/10.1038/s41586-021-03819-2>
2. Lam R, Sanchez-Gonzalez A, Willson M, et al (2023) Learning skillful medium-range global weather forecasting. *Science* 382:1416–1421. <https://doi.org/10.1126/science.adi2336>
3. Litjens G, Kooi T, Bejnordi BE, et al (2017) A survey on deep learning in medical image analysis. *Med Image Anal* 42:60–88. <https://doi.org/10.1016/j.media.2017.07.005>
4. Russakovsky O, Deng J, Su H, et al (2015) ImageNet Large Scale Visual Recognition Challenge. *Int J Comput Vis* 115:211–252. <https://doi.org/10.1007/s11263-015-0816-y>
5. Agrawal M, Peterson JC, Griffiths TL (2020) Scaling up psychology via Scientific Regret Minimization. *Proc Natl Acad Sci* 117:8825–8835. <https://doi.org/10.1073/pnas.1915841117>
6. Hofman JM, Watts DJ, Athey S, et al (2021) Integrating explanation and prediction in computational social science. *Nature* 595:181–188. <https://doi.org/10.1038/s41586-021-03659-0>
7. Verhagen MD (2024) Incorporating Machine Learning into Sociological Model-Building. *Sociol Methodol* 54:217–268. <https://doi.org/10.1177/00811750231217734>
8. Balbo N, Billari FC, Mills M (2013) Fertility in Advanced Societies: A Review of Research. *Eur J Popul Rev Eur Démographie* 29:1–38. <https://doi.org/10.1007/s10680-012-9277-y>

9. Bakker BFM, Rooijen J van, Toor L van (2014) The system of social statistical datasets of Statistics Netherlands: an integral approach to the production of register-based social statistics. *Stat J U N ECE* 30:411–424. <https://doi.org/10.3233/SJI-140803>
10. Scherpenzeel A, Das JWM (2010) “True” longitudinal and probability-based internet panels: Evidence from the Netherlands. In: Das JWM, Ester P, Kaczmirek L (eds) *Social and Behavioral Research and the Internet*. Taylor & Francis, Boca Raton, pp 77–103
11. Sivak E, Cantrell EM, Emery T, et al (2026) Births are difficult to predict even with rich survey and full-population register data. <https://arxiv.org/abs/2609.01194>
12. Liu DM, Salganik MJ (2019) Successes and Struggles with Computational Reproducibility: Lessons from the Fragile Families Challenge. *Socius* 5:. <https://doi.org/10.1177/2378023119849803>
13. 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*. <https://doi.org/10.1007/s42001-024-00275-6>
14. Neto EC, Hoff BR, Bare C, et al (2016) Reducing overfitting in challenge-based competitions. <https://arxiv.org/abs/1607.00091>
15. Kapoor S, Narayanan A (2023) Leakage and the reproducibility crisis in machine-learning-based science. *Patterns* 4:100804. <https://doi.org/10.1016/j.patter.2023.100804>
16. Kaufman S, Rosset S, Perlich C, Stitelman O (2012) Leakage in data mining: Formulation, detection, and avoidance. *ACM Trans Knowl Discov Data* 6:15:1-15:21. <https://doi.org/10.1145/2382577.2382579>
17. Ren H, Cantrell EM (2026) You Have More Data Than You Think: Optimizing Machine Learning in Tabular Social Science Datasets Through Augmentation and Linkage. <https://www.researchsquare.com/article/rs-10528921/v1>
18. Bokányi E, Heemskerk EM, Takes FW (2023) The anatomy of a population-scale social network. *Sci Rep* 13:9209. <https://doi.org/10.1038/s41598-023-36324-9>
19. van der Laan J, de Jonge E, Das M, et al (2023) A Whole Population Network and Its Application for the Social Sciences. *Eur Sociol Rev* 39:145–160. <https://doi.org/10.1093/esr/jcac026>
20. Statistiek CB voor (1995) *In de Gemeentelijke Basisadministratie Persoonsgegevens (GBA) ingeschreven personen en hun juridische ouders*
21. Statistiek CB voor (2021) *Persoonskenmerken van alle in de Gemeentelijke Basis Administratie (GBA) ingeschreven personen, gecoördineerd*
22. Breiman L (2001) Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author). *Stat Sci* 16:199–231. <https://doi.org/10.1214/ss/1009213726>
23. Rudin C, Zhong C, Semenova L, et al (2024) Amazing Things Come From Having Many Good Models. <https://arxiv.org/abs/2407.04846>

24. Pial T, Macanovic A, Sage L, et al (2026) Fertility Prediction with Foundation Models Trained on Population Registry Data (forthcoming)
25. Satish V, Hafner F, Kapoor S, et al (2026) Language Models for Life Course Prediction (forthcoming)
26. Verhagen MD, Stroebl B, Liu T, et al (2025) The Book of Life approach: Enabling richness and scale for life course research. <https://arxiv.org/abs/2507.03027>
27. Molinari R, Bakalli G, Guerrier S, et al (2021) SWAG: A Wrapper Method for Sparse Learning. <https://arxiv.org/abs/2006.12837>
28. Chang LY, Rostami Kandroodi M Who Will Have a Child? Predicting Three-Year Childbearing Outcomes Using Machine Learning (forthcoming)
29. Rotman S, Hendrickson A, Cassani G (2026) Using population social network data to improve predictions in administrative data (forthcoming)
30. Lundberg I, Brown-Weinstock R, Clampet-Lundquist S, et al (2024) The origins of unpredictability in life outcome prediction tasks. *Proc Natl Acad Sci* 121:e2322973121. <https://doi.org/10.1073/pnas.2322973121>
31. Amini M-R, Feofanov V, Pauletto L, et al (2025) Self-training: A survey. *Neurocomputing* 616:128904. <https://doi.org/10.1016/j.neucom.2024.128904>
32. Zhou Z-H, Li M (2005) Tri-training: exploiting unlabeled data using three classifiers. *IEEE Trans Knowl Data Eng* 17:1529–1541. <https://doi.org/10.1109/TKDE.2005.186>
33. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: Synthetic Minority Over-sampling Technique. *J Artif Intell Res* 16:321–357. <https://doi.org/10.1613/jair.953>
34. He H, Bai Y, Garcia EA, Li S (2008) ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). pp 1322–1328
35. Stekhoven DJ, Bühlmann P (2012) MissForest—non-parametric missing value imputation for mixed-type data. *Bioinformatics* 28:112–118. <https://doi.org/10.1093/bioinformatics/btr597>
36. Azur MJ, Stuart EA, Frangakis C, Leaf PJ (2011) Multiple imputation by chained equations: what is it and how does it work? *Int J Methods Psychiatr Res* 20:40–49. <https://doi.org/10.1002/mpr.329>
37. IterativeImputer. In: Scikit-Learn. <https://scikit-learn/stable/modules/generated/sklearn.impute.IterativeImputer.html>. Accessed 24 Mar 2025
38. Chen J, Yan J, Chen Q, et al (2024) ExcelFormer: A neural network surpassing GBDTs on tabular data. <https://arxiv.org/abs/2301.02819>
39. Wang C, Wu Q, Weimer M, Zhu E (2021) FLAML: A Fast and Lightweight AutoML Library.

<https://arxiv.org/abs/1911.04706>

40. Akiba T, Sano S, Yanase T, et al (2019) Optuna: A Next-generation Hyperparameter Optimization Framework. <https://arxiv.org/abs/1907.10902>
41. Stodden V, Seiler J, Ma Z (2018) An empirical analysis of journal policy effectiveness for computational reproducibility. *Proc Natl Acad Sci* 115:2584–2589. <https://doi.org/10.1073/pnas.1708290115>
42. Ewald FK, Bothmann L, Wright MN, et al (2024) A Guide to Feature Importance Methods for Scientific Inference. <https://arxiv.org/abs/2404.12862>
43. Ragodos R, Wang T, Feng L, Hu YJ (2024) From Model Explanation to Data Misinterpretation: Uncovering the Pitfalls of Post Hoc Explainers in Business Research. <https://arxiv.org/pdf/2408.16987>
44. Lundberg SM, Lee S-I (2017) A Unified Approach to Interpreting Model Predictions. In: *Advances in Neural Information Processing Systems*. Curran Associates, Inc.
45. Ajzen I, Klobas J (2013) Fertility intentions: An approach based on the theory of planned behavior. *Demogr Res* S16:203–232. <https://doi.org/10.4054/DemRes.2013.29.8>
46. Bhrolcháin MN, Beaujouan É (2019) Do People Have Reproductive Goals? Constructive Preferences and the Discovery of Desired Family Size. In: Schoen R (ed) *Analytical Family Demography*. Springer International Publishing, Cham, pp 27–56
47. Stulp G, Top L, Xu X, Sivak E (2023) A data-driven approach shows that individuals' characteristics are more important than their networks in predicting fertility preferences. *R Soc Open Sci* 10:230988. <https://doi.org/10.1098/rsos.230988>
48. Prins K (2017) *Population Register Data, Basis for the Netherlands Population Statistics*
49. Cantrell E, Ren H, Kapoor S, et al (2026) More Data, Modest Gains: Assessing Limits of Prediction in Multi-domain Population-Scale Social Data (forthcoming)

Figure Legends

Figure 1. Number of survey-based submissions with different reproducibility statuses. Lighter shade means the submission was reproduced but with a different score. The total number of survey-based submissions is 69.

Figure 2. F1 scores of retrained models versus original scores for the 60 survey-based submissions that could be retrained. Submissions on the dashed line are reproduced exactly, and the vertical distance from the dashed line is the reproducibility gap. Colours represent the fix required to make the submission run: as submitted (as-is), light, or substantive. Submissions whose three training runs gave identical scores are shown as a single point; for the five submissions with varied scores, the point shows the mean across three runs and the whisker shows the minimum and maximum across the three runs.

Figure 3. Predictive importance of individual features across submissions, grouped by topic. Each point is a feature used in at least one submission; its horizontal position is the maximum importance across the models that included it, and its size is proportional to the number of models in which it was included. Topics are ordered by their average importance across submissions. (A) Survey submissions: 21 submissions that outperformed the fertility-intentions baseline and for which we could calculate SHAP are included; (B) Register submissions: top 5 submissions (among those for which we could calculate SHAP) are included.

Figure 4. The 30 features with the highest maximum importance across the submissions that included them: (A) survey submissions, (B) register submissions. Each row is a feature; dots show its normalised global SHAP importance in each submission that included it, and the vertical bar shows the median across submissions. Features are coloured by topic and ordered by median importance.

Figure 5. Contribution of different topics to predictive performance in the top-3 survey models. Blue colour shows the performance of the model that only includes age and gender plus one additional topic. Red colour shows predictive performance when one topic is removed from a full model. The third-best model does not show the F1 score for the age and gender model because it is zero. Error bars show 95% confidence intervals calculated via bootstrapping of the holdout.

Figure 6. Contribution of different topics to predictive performance in the top-3 register models. Blue colour shows the performance of the model that only includes age and gender plus one additional topic. Red colour shows predictive performance when one topic is removed from a full model; 95% confidence intervals were calculated via bootstrapping of the holdout but are not visible because they are very small.

Table Legends

Table 1. Datasets and participation in the two phases of PreFer

Supplementary Files

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

- [Additionalfile1surveysubmissionsreproducibility.xlsx](#)
- [supplementaryinformationPreFer.docx](#)