SOCIUS
Special Collection: Fragile Families Challenge

Socius: Sociological Research for a Dynamic World
Volume 5: 1–10

DOI: 10.1177/2378023118820418
srd.sagepub.com
SAGE

Winning Models for Grade Point Average, Grit, and Layoff in the Fragile Families Challenge

Daniel E. Rigobon[1,2], Eaman Jahani[1], Yoshihiko Suhara[1], Khaled AlGhoneim[1], Abdulaziz Alghunaim[1], Alex "Sandy" Pentland[1], and Abdullah Almaatouq[1]

Abstract

In this article, the authors discuss and analyze their approach to the Fragile Families Challenge. The data consisted of more than 12,000 features (covariates) about the children and their parents, schools, and overall environments from birth to age 9. The authors' modular and collaborative approach parallelized prediction tasks and relied primarily on existing data science techniques, including (1) data preprocessing: elimination of low variance features, imputation of missing data, and construction of composite features; (2) feature selection through univariate mutual information and extraction of nonzero least absolute shrinkage and selection operator coefficients; (3) three machine learning models: random forest, elastic net, and gradient-boosted trees; and finally (4) prediction aggregation according to performance. The top-performing submissions produced winning out-of-sample predictions for three outcomes: grade point average, grit, and layoff. However, predictions were at most 20 percent better than a baseline that predicted the mean value of the training data for each outcome.

Keywords

Fragile Families Challenge, data science, machine learning

In this article, we describe our individual and team submissions that collectively won first place in three categories in the Fragile Families Challenge (FFC). The challenge was based on the Fragile Families and Child Wellbeing Study (FFCWS) (McLanahan, Garfinkel, and Waller 2000; Waldfogel, Craigie, and Brooks-Gunn 2010), which followed thousands of American households for more than 15 years and collected information about the children and their parents, schools, and environments. Within these data, six key outcomes were identified: (1) grade point average (GPA) and (2) grit of the child, (3) material hardship and (4) eviction of the household, and (5) layoff and (6) job training of the primary caregiver. Given these outcomes for a small portion of households as training data and approximately 12,000 features[1] from birth to age 9 for all households, FFC participants were tasked with predicting the outcomes for all households. Our best performing submissions were ranked 1st in predicting GPA, grit, and layoff, along with 3rd for job training, 8th for material hardship, and 11th for eviction.

The FFCWS data (McLanahan et al. 2000; Waldfogel et al. 2010) have been used in studies attempting to understand causal effects in well-being indicators such as academic standing or material hardship (Carlson, McLanahan, and England 2004; Mackenzie et al. 2011; Wildeman 2010). Our approach neither aimed to develop new insights into causal processes nor created novel data science techniques to analyze social science data. Rather, we made use of existing methods to thoughtfully navigate the steps required in prediction tasks. Our data after preprocessing and engineering of new features included more than 20,000 features while providing training outcomes for only 2,121

[1]Features are also commonly known as covariates or independent variables.

[1]Massachusetts Institute of Technology, Cambridge, MA, USA
[2]Princeton University, Princeton, NJ, USA

Corresponding Author:
Abdullah Almaatouq, Massachusetts Institute of Technology, 20 Ames Street, Cambridge, MA 02142, USA.
Email: amaatouq@mit.edu

households.[2] Therefore, feature selection was a critical step in our approach.[3]

This article is organized as follows. First, we explain our methodology, including preprocessing, engineering, selection of features, and model development. We then describe our results, including model performance and feature importance. Finally, we close with a discussion of insights we obtained from this challenge and some suggestions for future work related to common prediction tasks in the social sciences.

Methodology

Our team elected to pursue a collaborative approach to the FFC by dividing the task of generating predictions into three largely independent subtasks: preparation of the data, development of models, and aggregation of individual predictions. This modular approach enabled our team members to contribute where their strengths lay to build on one another's work.

We performed a single preprocessing of the data but used two techniques for feature selection and three distinct learning algorithms. From these three algorithms, four individual prediction sets were generated,[4] and four aggregations of these predictions were performed.

Feature Engineering

Most machine-learning algorithms are prone to overfitting when their training data contain more features than observations. As this was the case with the raw FFC data set, we needed to extract features that could predict the challenge outcomes and remove those that would not. Figure 1 shows how the data set changed over the course of this study's feature engineering.

Eliminating Features. We removed any feature that had small variance[5] or contained more than 80 percent missing data, which reduced the number of features from 12,942 to 5,168.

Imputation of Missing Data. We treated missing data in continuous and ordinal features differently from that in categorical features.[6]

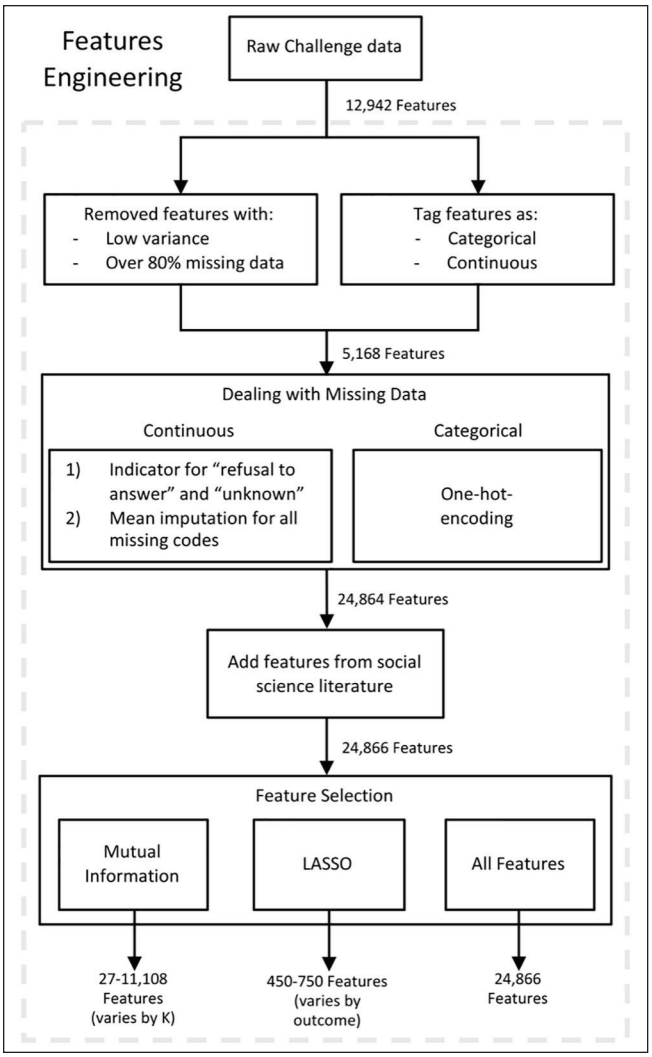

Figure 1. Flowchart of feature engineering with the number of features after every major step of data pre-processing.

Because only a small proportion of continuous and ordinal features contained missing values, we performed a simple mean imputation and additionally added two dummy variables[7] when respondents either refused to answer or did not know the answer to a question.[8]

One-hot encoding[9] was performed on the categorical features. Every unique missing code and possible response for a categorical feature became a new dummy variable, such that

[2]Of the total 4,242 households, only 2,121 had training data supplied. The remaining 2,121 were used by the FFC organizers for leaderboard and hold-out evaluations.

[3]High-dimensional problems, in which the number of features exceeds the number of observations, are not ideal for many machine-learning algorithms.

[4]A single learning algorithm (random forest) was used by two team members to generate predictions using different data.

[5]Features with absolute variance smaller than 0.05.

[6]Our method of identifying features as either continuous or ordinal is found in the supplementary information.

[7]Also identified as binary indicators or Boolean variables.

[8]Both of these missing codes (refusal = −1, do not know = −2) could be indicative of an effect present but not tangibly captured by the continuous or ordinal responses in the data.

[9]One-hot encoding is a process by which features are partitioned into unique response dummy variables. A question with four possible responses (including missing codes) would be replaced with four columns such that the row-wise sum of the resulting variables is exactly one for all observations.

no imputation was necessary. Our use of one-hot encoding significantly increased the number of features in our data set, as each possible response to a categorical question (including every missing code) constituted a new feature. Following this process, the data set contained 24,864 features for each of the original 4,242 households and no missing data in any of the features.[10]

Composite Homelessness Features. Previous research with the FFCWS data uncovered relationships between features and FFC outcomes. In one particular study, Fertig and Reingold (2008) identified factors positively and negatively correlated with homelessness or doubling up (living with someone else). These two sets of features were weighted and aggregated[11] into two composite features that were correspondingly positively and negatively related to homelessness. This resulted in our final, complete data set, with 24,866 features for each of 4,242 households.

Feature Selection. Learning algorithms struggle with high-dimensional data, as was the case at this stage of our methodology, with 6 times as many features (i.e., covariates) as observations (i.e., households). Therefore, we needed to eliminate features that were not predictive of our outcomes. We used two methods to reduce the number of features: (1) univariate feature selection based on mutual information and (2) extraction of nonzero least absolute shrinkage and selection operator coefficients (LASSO)[12] coefficients.

Mutual information (Peng, Long, and Ding 2005) is a measure of predictability from information theory defined as:

$$I(X,Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log\left(\frac{p(x)p(y)}{p(x,y)}\right),$$

which captures the level of information that two random variables share.[13] We calculated the mutual information value for each unique outcome (X) and feature (Y) pair. For each outcome we selected the top K[14] features and merged them to create data for distinct K-values that could be used for model building.

LASSO was our second feature selection method (Kukreja, Löfberg, and Brenner 2006), which admits a penalty parameter (α) that sets coefficients to zero if they are not useful for reducing the model's loss criterion: the sum of squared residuals plus the sum of coefficients' magnitude. Therefore, the LASSO selects features that have predictive power toward the outcome and discards those that do not. The value of α determines the extent of feature selection and was selected such that the resulting regression's R^2 value (variance accounted for) equaled an ad hoc value of 0.4 for each outcome. Such a value was large enough to prevent removing too many important features while still significantly reducing the number of features.

The number of features selected by both methods can be found in the supplementary information. It is important to note that feature selection is not directly indicative of feature importance or out-of-sample predictive power. Importance and predictive power are derived from the learning models that are cross-validated, described in the following section.

Model Building

After feature engineering was completed, we had two databases that could be directly used by learning algorithms to train models and subsequently generate predictions. In making model design choices, we made use of the leaderboard available to FFC participants.

Four individual team members developed models in parallel, which resulted in two broad types of approaches: regularized linear models (in the form of an elastic net) and nonlinear tree-based models (implemented as either random forests[15] or gradient-boosted [GBoost] trees).

We treated the prediction of GPA, grit, and material hardship as a continuous regression task, whereas the remaining three outcomes—eviction, job training, and layoff—were predicted as binary, with an underlying probability. For these binary outcomes, we chose to submit the underlying probability of positive class label (1), as opposed to discrete class labels (in this instance, 0 or 1). Predicting probabilities for the binary outcomes would help improve our performance by lowering the brier loss associated with incorrect predictions.[16]

[10]Missing codes are still present as dummy variables created by one-hot encoding.

[11]The exact weights and methodology behind the construction of these features can be found in the supplementary information.

[12]LASSO involves using an L1 norm penalty term in ordinary least squares regression to penalize nonzero coefficients.

[13]The mutual information, I(X,Y), is equal to zero if X and Y are independent, as in the case of p(X|Y) = p(X). This means that we have no improvement in the knowledge of X from Y. On the other hand, if X and Y are not independent, then I(X,Y) > 0: the knowledge of Y is useful to better understand X.

[14]Several values of K were used and can be found in the supplementary information.

[15]The random forest algorithm was used by two distinct team members to generate two individual prediction sets.

[16]For instance, for an observation with true value 1 for eviction, if we find that this observation has probability 0.4 of being evicted, we are worse off by predicting 0 (brier loss of 1) than by predicting 0.4 (brier loss of 0.36).

[17]L1 regularization penalizes proportional to the sum of coefficient magnitudes, and L2 regularization penalizes proportional to the sum of squared coefficient magnitudes.

The Elastic Net. The elastic net is a regularized linear model that combines LASSO (L1) and ridge (L2) regularization[17] (Zou and Hastie 2005) and achieves the advantages of both methods: sparsity and stability. It can perform additional feature selection by setting coefficients equal to zero, the extent of which is parametrized by the coefficients on the L1 and L2 regularization terms.

In a correctly specified linear model, the relationship between the independent and dependent variables is linear. The inclusion of only raw untransformed features could lead to model misspecification and decrease performance. Therefore, we applied three transformations to the continuous features used by the elastic net—log, square root, and square—and then normalized each transform-feature pair. The increased number of features did not pose a problem because of the elastic net's ability to perform additional feature selection, and in fact, the inclusion of transformed features improved this model's leaderboard performance. Furthermore, we transformed GPA by squaring it, so it exhibited a distribution that was less skewed and closer to normal.[18] Our final model used this GPA transformation, because it improved the model fit compared with the untransformed performance.

The elastic net generated a single set of predictions for the continuous outcomes only, with regularization parameters selected by k-fold cross-validation. It achieved the best leaderboard results when the continuous features and GPA were transformed and the cutoff for the K-mutual information feature selection method was no more than 300.

The Random Forest. The random forest algorithm (Liaw and Wiener 2002) is a nonlinear tree-based model and was used by two individual team members. Two unique sets of predictions were generated because of distinct feature selection and validation methods.

One of our team members trained random forest regressors or classifiers,[19] depending on whether the outcome was continuous or binary. These models were trained on untransformed features selected by mutual information with K = 100.[20] A total of 50 random forests[21] were trained in a nested cross-validation fashion (Cawley and Talbot 2010) by generating a series of training/validation/test splits with the given data. Each forest was fitted to each training split, and its hyperparameters were optimized in the validation splits. Finally, each forest's predictions were averaged according to performance on the test split. Nested cross-validation can help prevent random forests from overfitting, and this model's final predictions performed well on the binary-valued outcomes of the leaderboard.

A second team member trained random forest regressors on the features selected by the LASSO method. No feature transformations were applied, and the model parameters were selected on the basis of traditional k-fold cross-validation. This individual set of predictions did not perform as well as the other individual predictions on the leaderboard.

The Gradient-boosted Tree. The GBoost tree model (Friedman 2001) is a nonlinear tree-based method that learns a new decision tree additively to correct the residual errors from the existing sequence of trees. The GBoost tree is capable of taking into account multiple combinations of features, so we do not have to directly derive combinatorial features manually. Furthermore, the feature subsampling function enables us to skip the computationally expensive feature selection step, because the model's training method inherently avoids the overfitting problem.

For this model, we used the imputed 24,864-dimensional training data without feature selection, transformations, or the composite homelessness features we created from social science literature. We used the XGBoost (Chen and Guestrin 2016; Friedman 2001; https://github.com/dmlc/xgboost) implementation, with XGBRegressor for continuous-valued outcomes and XGBClassifier for binary-valued outcomes. The optimal hyperparameters for GBoost tree's single set of predictions were selected on the basis of threefold cross-validation.

Ensembled Predictions. Four individual sets of predictions had been generated and submitted to the challenge: one from elastic net, two by random forest, and another from the GBoost tree. In an effort to improve generalization, we aggregated our models' predictions in four distinct ways.

First, we performed a simple average of all four predictions, the team average. We averaged all four sets for the continuous outcomes and excluded elastic net for the binary ones.

Second, we experimented with a weighted team average, in which the weights were determined ad hoc by relative ranking on the leaderboard. The weight vector for the top three performing predictions for each outcome was given by [1/2, 1/3, 1/6] for first, second, and third, respectively. Predictions performing worse than 30th on the leaderboard were not included in this averaging.

Finally, we looked into aggregation with other models, using learning algorithms to find optimal weights for combining our individual prediction sets. This was done in two ways: using either linear or logistic regression or random

[18]This transformation of GPA would help prevent problems of model misspecification akin to those for the independent variables.

[19]Regressors predict continuous values, and classifiers predict discrete class labels with associated probabilities.

[20]The K cutoff used was selected on the basis of cross-validation. No significant difference was found with intermediate K values, though extreme values had worse performance in both cross-validation and on the leaderboard.

[21]For the competition, we submitted 200 iterations of a nested random forest. However, in this article we use only 50, which does not require access to high-performance computers and is reproducible in a reasonable amount of time.

forest regressor or classifier. Cross-validation was performed to select the best hyperparameters for these models.

Our submitted team predictions were generated by the weighted team average, weighted by individual predictions' leaderboard performance.

Results

We report the performance of all eight prediction sets.

Individual predictions
- Elastic net
- Random forest with nested cross-validation and mutual information feature selection
- Random forest regressors with LASSO feature selection
- XGBoost implementation of GBoost tree

Aggregated predictions
- Random forest aggregation
- Linear regression aggregation
- Weighted team average
- Simple team average

Model Performance

Model performance for the leaderboard and holdout sets was determined by looking at the improvement over the baseline,[22] or relative accuracy improvement.

The correlation between leaderboard and holdout scores was calculated across outcomes for all models, and for each individual outcome, to assess overfitting to the leaderboard, which was used in developing, evaluating, and aggregating models. The scatterplot of leaderboard versus holdout performance is shown in Figure 2. Notably, layoff and job training exhibited the largest magnitude correlation coefficients, indicating that performance on the leaderboard was strongly correlated with the performance on the holdout data set.

The strong correlations present indicate that performance on the leaderboard was a good proxy for performance on the holdout set. That is to say, the leaderboard was the best judge of performance on the holdout set. The same cannot be said for the relation between in-sample error and holdout performance, as we further explore in the supplementary information.

Feature Importance

Feature importance was determined for the GBoost tree, the best performing of our models. The importance values are derived from the algorithm's ability to partition outcome values depending on feature values. That is, a feature's importance grows as its splits lead to more homogenous subsets of an outcome in subsequent branches. As a result, importance is nearly impossible to interpret for two or more correlated predictors, as they would be equally useful in partitioning outcome values, but the algorithm will only use a single one.

It is important to note that our general approach and use of machine-learning algorithms is not designed to measure causal relationships between features and outcomes. Therefore, the feature importance values for our predictive task should not be confused with the properties we typically associate with parameter estimation tasks. Additional discussion on how to think about these values can be found in Mullainathan and Spiess (2017). The top three features for each outcome, along with their importance (as calculated for the GBoost tree models) and description (as found in the codebook), are provided in Table 1. For the features created through one-hot encoding, the feature description contains both the value of the response, and the question text. Notably, some of the most important features are closely related to the outcomes, but measured in earlier survey waves.

Values of feature importance were aggregated across categories corresponding to whom the question was posed to or when the question was asked. This resulted in overall importance of wave (i.e., the year of the data collection) and respondent (e.g., father, mother) in predicting any given outcome. The results of this aggregation are shown in Figure 3. We find that the most important data comes from wave 5 (last wave), except for material hardship, and the most important respondent is consistently the mother.

Discussion and Conclusion

The best performing model for GPA performed less than 20 percent better than a simple baseline (i.e., predicting the average GPA for everyone), while the competition-winning grit model had less than 10 percent improvement over the baseline. We attribute these modest improvements to three main causes.

First, the FFC data were very high dimensional, with more features than observations, which was exacerbated by our use of one-hot encoding in the preprocessing step. Furthermore, the traditional machine learning algorithms readily available in software packages were designed for scenarios in which there are more data points than features. Therefore, model performance was extremely sensitive to feature selection. In fact, reruns of an identical model repeatedly resulted in very different leaderboard performance, potentially because of the stochasticity in the algorithms that selected different features and optimal parameters. We believe that high-dimensional scenarios, similar to this challenge, are becoming more common in computational social science. Such scenarios present a greater need for research and implementation of high-dimensional statistical methods.

Second, common linear models such as ordinary least squares and its regularized variations (such as LASSO and elastic net) are not ideal for the continuous outcomes in the

[22]The baseline prediction is predicting the simple average value of the training set for the entire sample of households given.

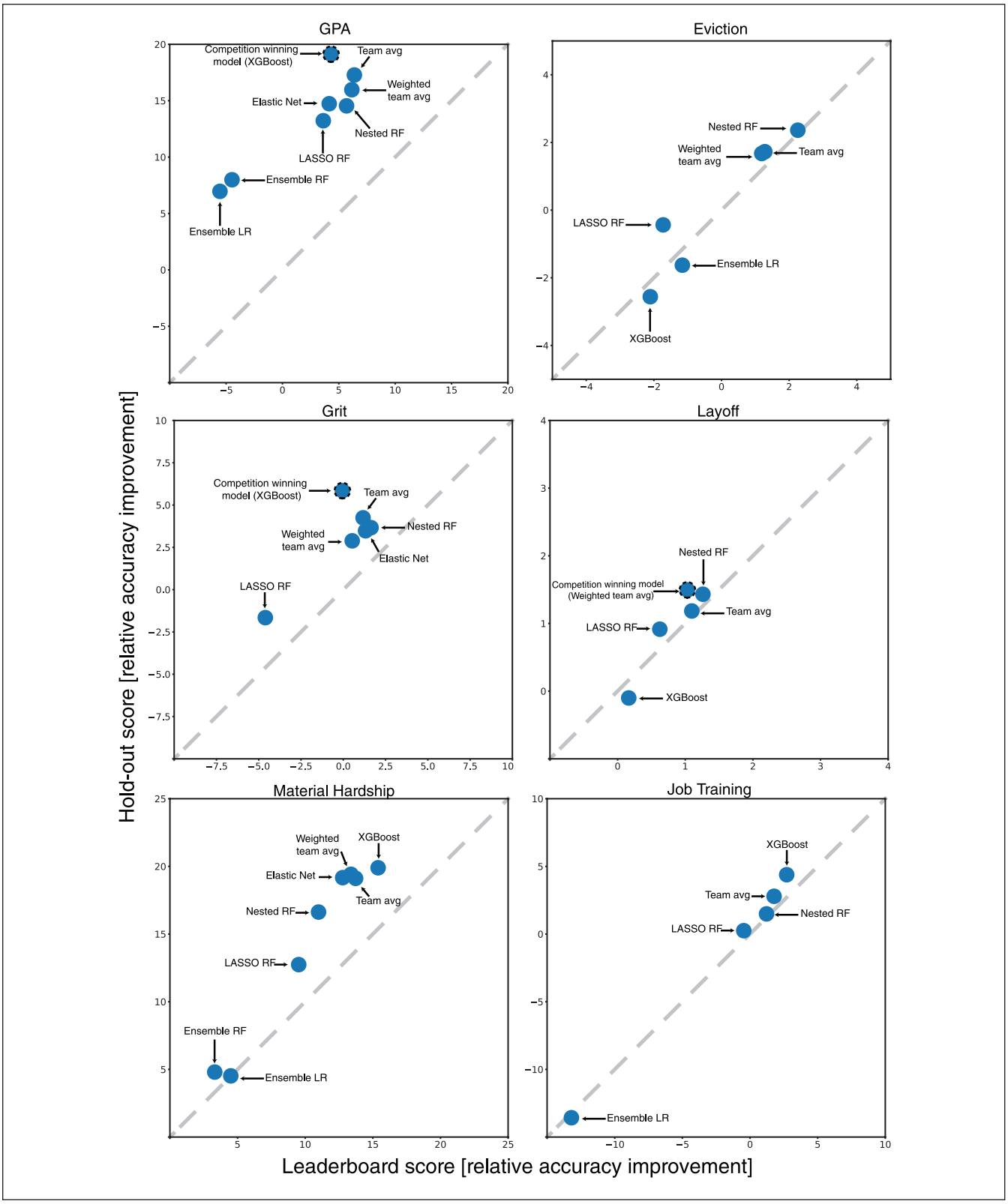

Figure 2. Model performance within the leaderboard and the holdout data sets for each outcome, as relative accuracy improvements over the baseline (average value in the training set). Notable winning and best performing models are highlighted, and the correlation between leaderboard and holdout scores are calculated overall and for each particular outcome. We have omitted models performing more than 25 percent worse than the baseline on either leaderboard or holdout sets. Fully labeled model performance on these sets can be found in the supplementary information.

Table 1. Top Three Most Important Features for the GBoost Tree Model, per Outcome.

Feature Code	Importance	Description
GPA		
hv5_wj10ss	0.01507	Woodcock-Johnson Test 10 standard score
f3b3	0.01004	How many times have you been apart for a week or more?
m2c3j	0.00904	How many days a week does father put child to bed?
Grit		
hv4l47_2	0.01520	Value 2 for "(He/she) stares blankly."
hv4r10a_3_1	0.01520	Value 1 for "Any hazardous condition 3: broken glass"
hv5_wj9raw	0.00946	Woodcock-Johnson Test 9 raw score
Material hardship		
m1lenmin	0.04380	Total length of interview (minutes)
m1citywt	0.03437	Mother baseline city weight (20-cities population)
m1lenhr	0.02110	Total length of interview (hours)
Eviction		
m5f23k_1	0.07216	Value "yes" for "Telephone service disconnected because wasn't enough money in past 12 months."
m5f23c_1	0.05842	Value "yes" for "Did not pay full amount of rent/mortgage payments in past 12 months."
m3i4	0.02062	How much rent do you pay each month?
Layoff		
p5j10	0.01678	Amount of money spent eating out in last month
m3i0q	0.01678	How important is it to serve in the military when at war?
f5i13	0.01678	How much you earn in that job, before taxes
Job training		
m4k3b_1	0.06355	Value "yes" for "In the last 2 years, have you taken any classes to improve your job skills?"
m5i1_1	0.06355	Value "yes" for "You are currently attending any school/trainings program/classes."
m5i3b_1	0.06355	Value "yes" for "You have taken classes to improve job skills since last interview."

Note: The feature importance values do not correspond to causal effects.

FFC, as GPA, grit, and material hardship were bounded. We experimented with Tobit regression (McDonald and Moffitt 1980) and nonlinear models to address this modeling deficiency; however, elastic net still achieved better performance for the continuous outcomes. We believe that bounded regression problems arise in many scenarios and that more attention to developing robust models for bounded regression is warranted. For instance, scikit-learn (Pedregosa et al. 2011), the popular machine-learning library in Python, does not currently provide an implementation of a bounded regression such as Tobit (McDonald and Moffitt 1980).

Third, the deidentification of the data required the omission of information about households' community (e.g., the levels of residential segregation). Previous studies have found that such features can be extremely important for child well-being outcomes. For example, researchers (Chetty et al. 2014) have found that intergenerational mobility varies substantially across geographic areas. This study found that community-level features (e.g., residential segregation, income inequality, family stability, and social capital) were the most predictive of intergenerational mobility (R^2 = .38). Perhaps a second and more secure stage of the challenge that allowed access to geographical or precomputed community indicators would allow models to perform better and provide insight as to how location-variant features may affect the outcomes of children's lives, while preserving the privacy of households.

As illustrated by the FFC organizers, most submitted models captured a very small portion of the variance in the outcomes, with even the best models predicting around the average. We believe these observations indicate poor predictive performance. In addition to the technical reasons above, we speculate that the inherent unpredictability of this setting could serve as a more fundamental reason behind the poor performance of models. This hypothesis becomes more plausible in light of recent focus on the limits to prediction and purely random outcomes, analogous to luck, in complex social settings such as ours (Hofman, Sharma, and Watts, 2017).

We believe that constructing predictive features from raw features may have contributed to our high relative performance. Fortunately, there is a vast body of research knowledge, not just restricted to Fragile Families data but in other similar contexts, that has studied the causal factors that affect the well-being of children. The inclusion of this knowledge in models such as ours could significantly affect predictive performance and improve the ability to verify previously published findings. However, as we experienced, a manual review of such a vast body of knowledge is next to impossible for researchers who lack domain knowledge or expertise in the sociology of fragile families. For those who participate

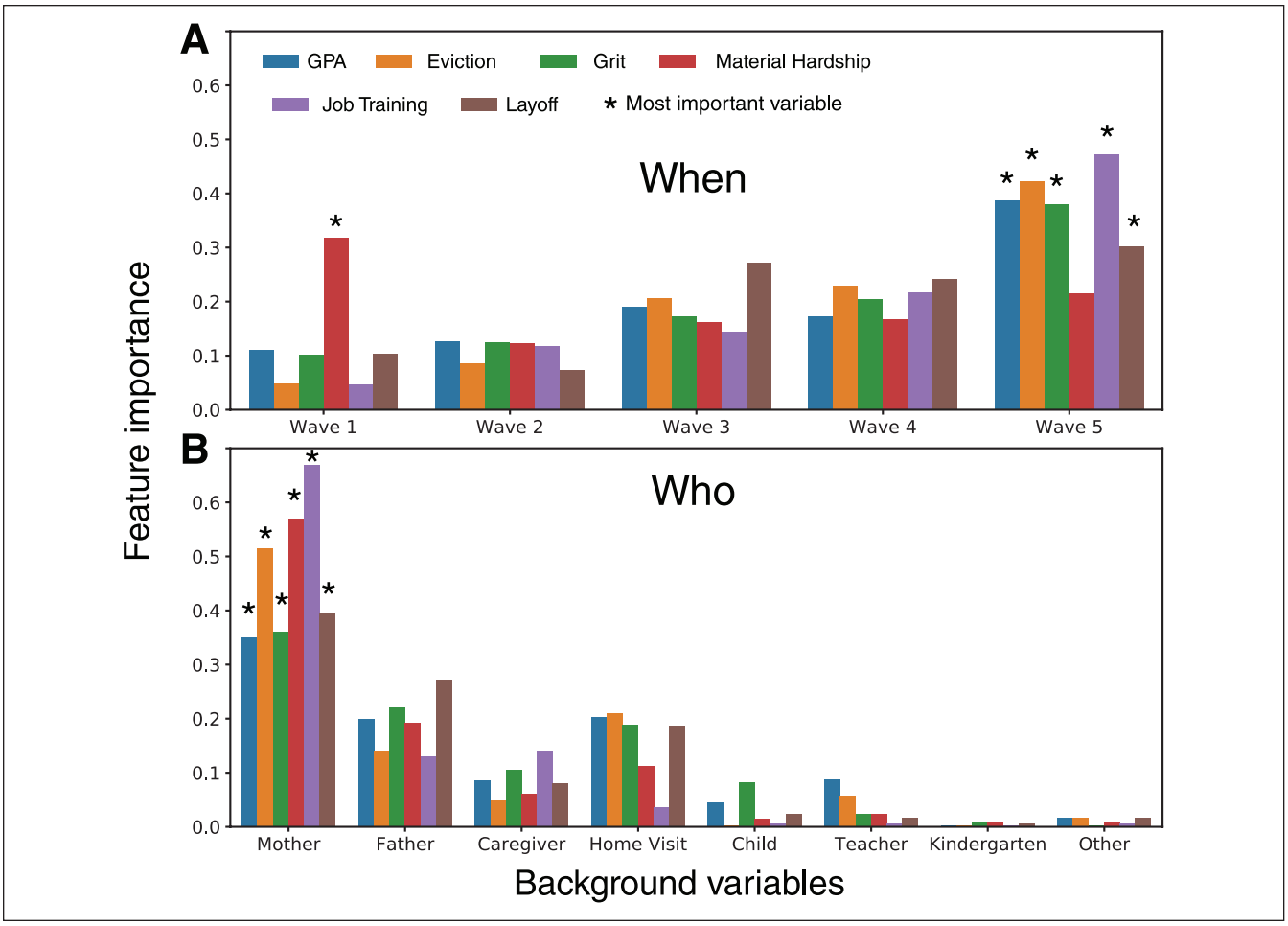

Figure 3. The figure shows the aggregated feature importance of questions asked at particular times (A) or to particular people (B) over the course of the children's lives. These importance values indicate the usefulness of a feature in predicting outcome values and are neither analogous to coefficients nor indicate the presence of causal effects. All of these values come from the gradient-boosted tree model. Stars indicate the highest importance for each outcome.

without extensive domain expertise, we believe the existence of a database incorporating the main results of relevant social science studies in a queryable structure should greatly help performance in prediction tasks, not only for the FFC but for evaluating the effectiveness of interventions in many other problem domains important to policy making.

Acknowledgments

Funding for the FFCWS was provided by the Eunice Kennedy Shriver National Institute of Child Health and Human Development through grants R01HD36916, R01HD39135, and R01HD40421 and by a consortium of private foundations, including the Robert Wood Johnson Foundation. Funding for the FFC was provided by the Russell Sage Foundation. The results in this paper were created with software using Python 3.6.1 (Python Software Foundation 2017) with packages numpy 1.12.1 (Oliphant 2006), scipy 0.19.0 (Jones, Oliphant, and Pearu 2001), matplotlib 2.0.2 (Hunter 2007), seaborn 0.8.1 (Waskom 2014), pandas 0.20.1 (McKinney 2010), scikit_learn 0.18.1 (Pedregosa et al. 2011), statsmodels 0.8.0 (Seabold and Perktold 2010), astropy 1.3.2 (Price-Whelan et al. 2018), XGBoost 0.6 (Chen and Guestrin 2016), R 3.4.3 (R Core Team 2017) with packages data.table 1.10.4-2 (Dowle and Srinivasan 2017) and Amelia 1.6.2 (Honaker, King, and Blackwell 2011).

ORCID iD

Abdullah Almaatouq https://orcid.org/0000-0002-8467-9123

Supplemental Material

Supplemental material for this article is available with the manuscript on the *Socius* website.

References

Carlson, Marcia, Sara McLanahan, and Paula England. 2004. "Union Formation in Fragile Families." *Demography* 41(2):237–61.

Cawley, Gavin C., and Nicola L. C. Talbot. 2010. "On Overfitting in Model Selection and Subsequent Selection Bias

in Performance Evaluation." *Journal of Machine Learning Research* 11(July):2079–2107.

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD '16*. Retrieved December 11, 2018 (http://delivery.acm.org/10.1145/2940000/2939785/p785-chen.pdf?ip=73.98.97.161&id=2939785&acc=CHORUS&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E6D218144511F3437&__acm__=1544561857_371fc743ba7299347ff2e8e410467a72).

Chetty, Raj, Nathaniel Hendren, Patrick Kline, and Emmanuel Saez. 2014. "Where Is the Land of Opportunity? The Geography of Intergenerational Mobility in the United States." *Quarterly Journal of Economics* 129(4):1553–1623.

Dowle, Matt, and Arun Srinivasan. 2017. data.table: Extension of 'data.frame.' R package version 1.10.4-2. (https://CRAN.R-project.org/package=data.table).

Fertig, Angela R., and David A. Reingold. 2008. "Homelessness among At-risk Families with Children in Twenty American Cities." *Social Service Review* 82(3):485–510.

Friedman, Jerome H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." *Annals of Statistics* 29(5):1189–1232.

Hofman, Jake M., Amit Sharma, and Duncan J. Watts. 2017. "Prediction and Explanation in Social Systems." *Science* 355(6324):486–88.

Honaker, James, Gary King, and Matthew Blackwell. 2011. "Amelia II: A Program for Missing Data." *Journal of Statistical Software* 45(7):1–47.

Hunter, John D. 2007. "Matplotlib: A 2D Graphics Environment." *Computing in Science & Engineering* 9:90–95. doi:10.1109/MCSE.2007.55

Jones, Eric, Travis E. Oliphant, and Pearu Peterson. 2001. "SciPy: Open Source Scientific Tools for Python." (http://www.scipy.org/).

Kukreja, Sunil L., Johan Löfberg, and Martin J. Brenner. 2006. "A Least Absolute Shrinkage and Selection Operator (LASSO) for Nonlinear System Identification." *IFAC Proceedings Volumes* 39(1):814–19.

Liaw, Andy, and Matthew Wiener. 2002. "Classification and Regression by randomForest." *R news* 2.3:18–22.

Mackenzie, Michael J., Eric Nicklas, Jeanne Brooks-Gunn, and Jane Waldfogel. 2011. "Who Spanks Infants and Toddlers? Evidence from the Fragile Families and Child Well-being Study." *Children and Youth Services Review* 33(8):1364–73.

McDonald, John F., and Moffitt, Robert A. 1980. "The Uses of Tobit Analysis." *Review of Economics and Statistics* 62(2):318.

McKinney, Wes. 2010. "Data Structures for Statistical Computing in Python." Proceedings of the 9th Python in Science Conference, June 28, Austin, TX. (https://conference.scipy.org/proceedings/scipy2010/pdfs/proceedings.pdf).

McLanahan, Sarah, Irwin Garfinkel, and Maureen Waller. 2000. "Fragile Families and Child Well-being Study." Retrieved December 11, 2018 (http://www.80bola.com.ppic.org/content/pubs/op/OP_1199MWOP.pdf).

Mullainathan, Sendhil, and Jann Spiess. 2017. "Machine Learning: An Applied Econometric Approach." *Journal of Economic Perspectives* 31(2):87–106.

Oliphant, Travis E. 2006. "A guide to NumPy." USA: Trelgol Publishing.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. "scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research* 12(October):2825–30.

Peng, Hanchuan, Fuhui Long, and Chris Ding. 2005. "Feature Selection Based on Mutual Information Criteria of Max-dependency, Max-relevance, and Min-redundancy." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27(8):1226–38.

Price-Whelan, Adrian M., B. M. Sipőcz, H. M. Günther, P. L. Lim, S. M. Crawford, S. Consei, D. L. Shupe, M. W. Craig, N. Dencheva, A. Ginsburg, J. T. VanderPlas, L. D. Bradley, D. Pérez-Suárez, and M. de Val-Borro. 2018. "The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package." *The Astronomical Journal* 156(3):1–19.

Python Software Foundation. 2017. *Python Language Reference, Version 3.6.1*. (http://www.python.org).

R Core Team. 2017. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. (https://www.R-project.org/).

Seabold, Skipper, and Josef Perktold. 2010. "Statsmodels: Econometric and Statistical Modeling with Python." Proceedings of the 9th Python in Science Conference, June 28, Austin, TX. (https://conference.scipy.org/proceedings/scipy2010/pdfs/proceedings.pdf).

Waldfogel, Jane, Terry-Ann Craigie, and Jeanne Brooks-Gunn. 2010. "Fragile Families and Child Wellbeing." *The Future of Children* 20(2):87–112.

Waskom, Michael. 2014. "Seaborn: Statistical Data Visualization." Accessed May 5, 2017 (https://seaborn.pydata.org/).

Wildeman, Christopher. 2010. "Paternal Incarceration and Children's Physically Aggressive Behaviors: Evidence from the Fragile Families and Child Wellbeing Study." *Social Forces* 89(1):285–309.

Zou, Hui, and Trevor Hastie. 2005. "Regularization and Variable Selection via the Elastic Net." *Journal of the Royal Statistical Society, Series B: Statistical Methodology* 67(2):301–20.

Author Biographies

Daniel E. Rigobon is currently a PhD candidate at Princeton University in operations research and financial engineering. He studied his undergraduate years at the Massachusetts Institute of Technology (MIT), during which time he became heavily involved in research with the Media Lab's Human Dynamics group. His current research interests lie in financial markets, network dynamics, and game theory.

Eaman Jahani is a PhD student at the MIT Institute for Data, Systems, and Society. He has a BS and an MS in computer science. His research focuses on the intersection of economics, computer science, and statistics. He is currently investigating the network effects on inequality using empirical methods. The research aims to discover the mechanisms under which social networks can exacerbate intergroup differences in economic outcomes. Before

joining MIT, he was a software engineer at Google for more than four years.

Yoshihiko Suhara, PhD, is a research scientist at Megagon Labs and a visiting scientist at the MIT Media Lab. His research interests include machine learning, natural language processing, and computational social science.

Khaled AlGhoneim is the founder of Hawaz, a company focusing on public-private partnerships and behavioral economics, as well as cofounder of Mozn, a data analytics firm. He received a PhD in electrical and computer engineering from Carnegie Mellon University in 1996 and went on to serve on the faculty of King Saud University, where he worked on signal processing and machine learning. In 2002, he established Elm, the premier Saudi e-government company. He also created Takamol, a company affiliated with the Saudi Ministry of Labor. During summer 2017 he was a visiting researcher with the Human Dynamics group at the MIT Media Lab, where he participated in the FFC.

Abdulaziz Alghunaim is a cofounder of Tarjimly, a technology nonprofit that allows bilinguals to volunteer as translators for refugees. He received his BS and MS degrees in computer science from MIT in 2015, after which he joined Palantir as a software engineer working on data-driven decision making in finance, aircraft manufacturing, and oil and gas. In 2018, he left to focus on Tarjimly, a Y Combinator–backed technology nonprofit that aims to close the communication gap faced by millions of refugees around the world.

Alex "Sandy" Pentland directs the MIT Connection Science and Human Dynamics labs and previously helped create and direct the MIT Media Lab. He has an h-index of 129, and *Forbes* named him one of the "7 most powerful data scientists in the world." He is a founding member of advisory boards for Google, AT&T, Nissan, and the UN Secretary General, a serial entrepreneur, a member of the U.S. National Academy of Engineering, and a leader within the World Economic Forum.

Abdullah Almaatouq is a research assistant with the Human Dynamics group and is pursuing a PhD in computational science and engineering at MIT. Abdullah did dual master's degrees at MIT in computational engineering and media, arts, and sciences. Prior to MIT, Abdullah received his BS from the School of Electronics and Computer Science at Southampton University.