COS 424 Fragile Families Challenge

Sonia Hashim
Princeton University
shashim@princeton.edu

Viola Mocz
Princeton University
vmocz@princeton.edu

Abstract

The goal of the Fragile Families Challenge is to understand traits that predict how kids in families from primarily unmarried parents fare over time. We analyzed the Fragile Families data set to predict the outcomes *gpa*, *grit*, and *materialHardship* in the 15 year test data. For imputing missing values, we tested median single imputation, K-Nearest Neighbors single imputation, and multiple imputation using the Amelia package in R. We engineered features by taking the mean of matching inputs or by using maximum pooling. We also conducted feature selection by removing features with a low frequency of observations and low variance, features that were highly correlated with each other, and features deemed unimportant in a random forest representation. We modeled *gpa*, *grit*, and *materialHardship* using ordinary linear regression, lasso regression, and ridge regression after conducting imputation, feature engineering, and feature selection. Lasso regression with median imputation and engineered features (with and without feature selection for variance and random forest importance) performed the best overall for each of the three key outcomes.

1 Introduction

The Fragile Families & Child Wellbeing Study attempts to understand how children raised by primarily unmarried parents fare over time. 4242 families were surveyed in 5 waves over the course of 15 years. For the Fragile Families Challenge, participants analyze over 12,000 features given the data of the families from the birth of the child to year 9, as well as some training data at 15 years to predict six key outcomes in the 15 year test data: *gpa*, *grit*, *materialHardship*, *eviction*, *jobLoss*, and *jobTraining*. The goal is to understand the features that are most predictive for each of these outcomes and to find children that have beaten the odds to target for subsequent interviews. For our analysis, we focused on the three outcomes of *gpa*, *grit*, and *materialHardship*.

2 Methods

2.1 Imputation

The Fragile Families Challenge data set had several forms of missing data that were coded with negative values, "Other," or simply NA. We performed some simple preprocessing - for instance, removing features with all missing values or only one observation (no variability). Another method used was our creation of binary features. For every feature that had only one value other than missing values, we replaced that feature with a binary feature where 0 indicated a missing value and 1 indicated a valid observation. In order to impute missing values, we employed three main methods.

1. *Single Imputation - Median* (M): For every missing value, we computed the median of the observations for any given feature that was then used in place of missing values. We selected the median, because it is more robust to outliers than the arithmetic mean. However, this approach reduces the variability of the data, and because it ignores the dependency relationships between variables, diminishes correlation estimates.
2. *Single Imputation - K-Nearest Neighbors* (KNN): Next, we used the K-Nearest Neighbors of any given feature vectors to impute missing values. We used the R package Visualization and Imputation of Missing Values - VIM [2]. We did not rely on previously imputed values to prevent a dependency on location in the data set for each sample. Because of the computational cost involved, we first performed feature selection so that our KNN feature sets had 200 features. We attempted imputation on both the features in original the data set

and additionally on engineered features. Again due to the the computational cost, we used the $K = 3$ nearest neighbors.

3. *Multiple Imputation - using Amelia* (MI): Lastly, we used the R package Amelia to perform multiple imputations [1]. Amelia assumes that missing values are Missing At Random (MAR). By using multiple imputations, we were able to posit more than one value for each missing entry and therefore create models that would on average account for uncertainty in estimating the missing values. In theory, this would provide a more likely estimate for outcome variables, because predictions could be averaged across models built on the different imputations.

2.2 Feature Engineering

For feature engineering, we used two methods to build features. First, we identified which features could be combined. We combined features across categories (for instance, average response to questions regarding income in wave 3 answered by the father). Features were engineered by taking the mean value across inputs or by using maximum pooling (maxPool) to amplify the strength of the signal suggested by a given set of features. We tested our models on data sets with and without feature engineering to evaluate its efficacy. We denote feature engineering by the abbreviation eng.

2.3 Feature Selection

We tried three main combinations of the following four feature selection methods: eliminating features with low information or a certain percentage of missing data, eliminating features with low variance, eliminating features that were highly correlated with each other, and using random forest importance as a score to use for cutoff selection.

1. *Low Information and Low Variance* (fsVar): Firstly, we eliminated features with a low frequency of observations. That is, we choose to not include features with 50% or more of missing data in our models. Then, we used variance within each feature as a measure of how informative a feature might be. We scored each feature according to the variance of its values and removed features with less than the first quartile of variance (4.0).
2. *High Correlation* (fsHC): We created a correlation matrix for all of the features and we removed all features that had a correlation greater than 0.95 with any other feature.
3. *Random Forest Importance* (fsRF): We also used random forest importance where importance is defined as the mean decrease in accuracy of the random forest constructed when the feature in question is left out [4]. Every feature was withheld and a random forest was constructed using the remaining feature sets. Features that provided the least gain in accuracy were then omitted. We used a cut-off to create a feature set with 2000 features.

On account of the computational cost of multiple imputations, we created a reduced feature set with only 100 features using the three feature selection methods detailed above. We ran 10 imputations with 100 features through our models and averaged the resulting predictions.

2.4 Regression Models

Because we chose to focus on the continuous outcome variables, namely *gpa*, *grit*, and *materialHardship*, we sought to test the efficacy of different regression models. We used cross validations to set our hyperparameters for regularization and used the Scikit-Learn Python Library implementation of regression models, in some cases including cross validation [4].

1. *Ordinary Linear Regression* (OLR) - In order to estimate continuous variables, we relied on standard regression models. Ordinary linear regression tries to minimize the residual sum of squares between samples and assumes that features (terms) are independent [5]. When features are not dependent, the effect of errors due to uncertainty in the sample or imputation are greatly magnified as a result of this assumption. On account of this independence assumption, we tried feature selection to minimize dependencies between terms.
2. *Lasso Regression* (LR): *using l1 penalty* - In order to avoid over-fitting and to identity the most useful features for predicting outcome variables, we used Lasso regression. Lasso

achieves this effect by minimizing residual sum of squares with an *l1* penalty. The implementation used coordinate descent to fit the coefficients [5]. Because our goal is to identify the most useful features for predicting *gpa*, *grit*, and *materialHardship*, we used Lasso to perform an ultimate round of feature selection and have shared our results of the top features generated by each model. The regularization parameter α, was set by cross validation.

3. *Ridge Regression* (RR): *using l2 penalty* - To avoid over-fitting in our smaller feature sets, we used ridge regression. Because ridge regression uses an *l2* penalty none of the coefficients are zero [5]. Therefore, each feature is used to predict the outcome variable. We used ridge regression on our smallest feature sets where we had already used feature selection to encode the maximum number of dependencies. Again, the shrinkage parameter, α, was set by cross validation.

2.5 Predicting Prediction Results

Before submitting to the Fragile Families Challenge platform, we predicted the performance of our predictions by using cross validation scoring on internal validation sets. We procured two scores - namely, the coefficient of determination (R2) to assess the ability of our model to predict outcomes on future samples [6] and the mean squared error (MSE) to provide a measure of accuracy on the available data set and because MSE is the metric used by FFC. To this end, we used the implementation in Scikit Learn of a cross validation scorer and used 5 training and test splits to ascertain R2 and MSE for each split [4]. We report our results as the accuracy of R2 and MSE by taking the mean across five splits and the 95% confidence interval (two times the standard deviation across scores).

3 Spotlight Model: Ridge Regression with Multiple Imputations

One of our models that worked well for predicting grit was using ridge regression on the multiple imputation sets that had 100 features. We performed the fsVar and fsHiCor feature selection to reduce the original 12,000 feature set down to about 2,800 features. Then we used fsRF feature selection to further reduce the feature set to only 100 features. There were still missing values in this reduced feature set so we did multiple imputation with Amelia (details described previously) to create 10 complete data sets. Ridge regression was performed on each data set and the 10 different estimates were averaged.

Ordinary linear regression seeks to minimize the sum of the squared residuals

$$||X\beta - y||^2$$

where X is the features, y is the true result, and β is the estimate of regression coefficients. Ridge regression adds a regularization term so that we minimize

$$||X\beta - y||^2 + ||\Gamma\beta||^2$$

where Γ is a multiple of the identity matrix (i.e. $\Gamma = \alpha I$). Then the maximum a posteriori (MAP) estimation of the regression coefficients is

$$\hat{\beta}_{MAP} = (X^T X + \alpha I)^{-1} X^T y$$

The regularization term is the l2 loss function. It helps prevent overfitting and ensures that we can find a solution if the features are not linearly independent [3]. The regularization parameter, α, was set by leave-one-out cross validation by values 1e-15, 1e-10, 1e-6, 0.0001, 0.001, 0.01, 1, 5, 10, and 20 using the RidgeCV method in Scikit-learn [4]. Ridge regression works especially well on data sets with a small number of features (in the hundreds) so we used this model for the imputed data sets that had 100 features.

After fitting each data set with the ridge regression model we found the 10 most important features for each data set, which we determined to be the features with the largest coefficients. The most consistently important features for grit across the data sets were "t5e6", "cm2age", "f4b8b", "m4h1o", "m4i20", "f5a6e", "f2h8c2", "m3c23", and "m3a12." "t5e6" asks for the highest level of education completed by the teacher's aide. "cm2age" is the age of the mother when the child is 1 year old. "f4b8b" asks how many hours per week the child attends a primary program when the child is 5 years old. "m4h1o" asks whether the father's parents currently live together when the child is 5 years old. "m4i20" asks whether there was a time in the past 12 months the mother thought they might be

eligible for food stamps when the child is 5 years old. "f5a6e" asks whether the biological children who do not live with the father all live in same place when the child is 9 years old. "f2h8c2" asks how much the father received help from food stamps in the last month when the child is 1 years old. "m3c23" asks whether the mother has an informal agreement with the father for financial support of the child when the child is 3 years old. "m3a12" asks if the mother has children with someone other than the father when the child is 3 years old. It appears that grit is strongly connected with proper financial and educational support for the child and whether or not the family is "nuclear."

4 Results

Model	*gpa* MSE	*gpa* r^2	*grit* MSE	*grit* r^2	*mH* MSE	*mH* r^2
olr_M	3.869±2.792	-7.943±7.311	1.354±0.246	-4.774±1.201	0.156±0.056	-5.528±2.756
olr_M_eng	4.755±2.313	-9.930±6.397	1.636±0.36	-5.984±1.791	0.166±0.046	-5.926±2.165
olr_M_fsVar	5.318±2.458	-11.084±5.590	1.843±0.368	-6.859±1.733	0.178±0.065	-6.394±3.132
olr_M_fsVar_fsEng	2.435±1.684	-4.652±4.769	1.217±0.206	-4.190±1.041	0.116±0.022	-3.857±1.522
lr_M	0.445±0.042	-0.007±0.013	0.235±0.013	0.000±0.002	0.024±0.003	-0.000±0.001
lr_M_eng	0.444±0.041	-0.005±0.008	0.235±0.013	0.000±0.002	0.024±0.003	-0.000±0.001
lr_M_eng_fsVar	0.444±0.041	-0.005±0.008	0.235±0.013	0.000±0.002	0.024±0.003	-0.000±0.001
lr_M_eng_fsVar_fsRF	0.443±0.039	-0.003±0.001	0.237±0.011	0.008±0.022	0.024±0.003	-0.000±0.001
olr_KNN_fsVar	0.558±0.092	-0.263±0.124	0.289±0.054	-0.228±0.183	0.029±0.004	-0.218±0.115
olr_KNN_eng_fsVar	0.593±0.068	-0.350±0.263	0.337±0.101	-0.443±0.536	0.033±0.009	-0.360±0.257
lr_KNN_fsVar	0.444±0.034	-0.005±0.011	0.235±0.012	0.002±0.003	0.024±0.003	-0.000±0.001
lr_KNN_eng_fsVar	0.444±0.040	-0.004±0.005	0.235±0.012	-0.000±0.002	0.024±0.003	-0.000±0.001

Predicting Prediction Results. Mean Squared Error (MSE) and the coefficient of determination (r^2) reported as the mean averaged across 5 test / train splits with a 95% confidence interval.

Model	*gpa* (MSE)	*grit* (MSE)	*mH* (MSE)
olr_M	2.40337	0.75968	0.09216
olr_M_eng	2.83103	0.92572	0.11599
olr_M_fsVar	2.74103	0.93427	0.10924
olr_M_fsVar_fsEng	1.70245	0.91509	0.12705
lr_M	0.39273	0.22033	0.0288
lr_M_eng	0.39273	0.21975	0.0288
lr_M_eng_fsVar	0.39273	0.21975	0.0288
lr_M_eng_fsVar_fsRF	0.39273	0.21997	0.0288
olr_KNN_fsVar	0.45772	0.25843	0.03175
lr_KNN_fsVar	0.39273	0.22098	0.0288
lr_KNN_eng_fsVar	0.39273	0.21987	0.0288
olr_MI_fsVar_fsRF_fsHC	0.44167	0.22898	0.03065
rr_MI_fsVar_fsRF_fsHC	0.44155	0.22891	0.03064

Fragile Families Challenge Evaluation Scores for *gpa*, *grit*, and *materialHardship*. The col. to the right of each variable indicates if the score was within the range we predicted for the prediction results (please see above). Our ids on the submission result site are shashim, vmocz, agentv.

Top Feat.	lr_M	lr_M_eng	lr_M_eng_fsVar	lr_M_eng_fsVar_fsRF	lr_KNN_fsVar	lr_KNN_eng_fsVar
1	f2h3	f2h_maxPool	f2h_maxPool	m5f_maxPool	m4i3	f2h_maxPool
2	challengeID	m3i_maxPool	m3i_maxPool	ffcc_centsurvey_c_maxPool	f4i3	m3i_maxPool
3	f3d4b	f2h3	f2h3	f3b4l	mothid2	m5f_maxPool
4	f3e2b	m5f_maxPool	m5f_maxPool	f3b6a	f2k7f	m4citywt_rep3
5	f3e2a1	ffcc_centobs_fca_maxPool	ffcc_famsurvey_f_maxPool	f3b6c	m3k13	f4c_maxPool
6	f3e2	f3i29	m3citywt_rep3	f3b6d	m4k13	m5e_maxPool
7	f3e1	f3i31c2	m3citywt_rep4	f3b8a_11	m3c38a	f2l_mean
8	f3d8	f3i31b	m3citywt_rep5	f3b8a_12	m2k15	m3l_mean
9	f3d6	f3i31a1	m3citywt_rep6	f3b9	fathid4	f2fc_maxPool
10	f3d5	f3i31a	m3citywt_rep7	f3b12	mothid4	m4l_mean
11	f3d4a	f3i31	m3citywt_rep8	f3b17	fathid3	f2f_maxPool
12	f3e2d	f3i30a	m3citywt_rep9	f3b17p	mothid3	m2h_mean
13	f3d4	f3i30	m3citywt_rep10	f3b23	fathid2	m2g_maxPool
14	$f3d3a_7$	f3i27c	fathid3	f3b32a	fathid1	f3c_maxPool
15	$f3d3a_6$	f3i27d2	cf3mint	f3b32b	m2k18e	m4c_maxPool

Top 15 features in decreasing order to predict grit as selected by Lasso regression.

5 Discussion

Our high scores for olr on a M imputation clearly indicate over-fitting. In fact, the models using the largest data sets (olr_M, olr_M_eng) have the highest scores even with preliminary feature selection which strongly motivates the need for regularization. The best performing regression model overall

is lr. For our KNN and MI imputations, this is more unclear as over-fitting is also prevented by the small size of the feature set. We can conclude that for gpa, lr with median imputation is tied with lr with KNN imputation, suggesting that lr is in fact the best model. Feature engineering and selection do not seem to effect the outcome. For grit, we can conclude that lr with engineered features is the best model which suggests that engineered features do provide more useful information for prediction. Finally, we can conclude for material hardship that lr on a M imputation performed best. However, again, this may be due to the smaller feature set used and may only reflect the better performance of lr and not necessarily the imputation.

The KNN and MI imputations did not outperform the M imputation. We suspect that this is due to the limited feature set in our KNN and MI imputations. It is notable that lr_KNN_eng_fsVar performed almost as well as our best model (lr_M_eng, lr_M_eng_fsVar) despite a vastly different number of features (200 and 2000, respectively). The proximity in scores suggests that KNN imputation may be more effective than M imputation. Furthermore, our imputations assume that data is missing at random (MAR). However, it might be worthwhile to perform a regression imputation modelling specific observed dependencies on other features.

Adding feature engineering increases the scores for olr (implying a worse fit). Because feature engineering combines information, engineered features encode more information than their standalone counterparts. Because more dependencies in a restricted feature set can be modeled, over-fitting increases on an already over-fitted model. However, for lr, adding feature engineering decreases the scores. This suggests that feature engineering does provide a more useful way to distill information from the original feature set.With regards to feature selection, we observe that lasso with and without fsVar feature selection (variance) has the same performance across all three outcome variables. This suggests that the preliminary feature selection eliminates features that are not vital for prediction. Objectively, our models perform well. For instance, grit which is scaled from 1 to 4 has a MSE of less than 0.25. However, the generalization error will probably be higher on account of the fact that we do not consider uncertainty from the sample itself (i.e. fluctuations if the same sample provided the same answers to the same questions).

Finally, we can identify key features predicting grit using our best model, Lasso, and feature selection. One key feature was the average response on the second wave for the father's environment and program. In particular "f2h3," asking for how much a father would sell his current home, stood out. Another notable feature from this wave was an engineered feature of the mean response to questions about his income (category L). For the mother, the third wave was the most significant. The most important questions centered on civic and community engagement (category I) and a mean engineered feature of questions regarding her income. Across wave 2, wave 3, and wave 4, the following features were significant for lr_KNN "m2k15", "m3k13", and "m4k14," and each asked the mother's income in the past year. Of the mother's data in the third wave, it is interesting to note the city weight replicates that appear. We suspect this is on account of their high standard deviations which led our inadvertent selection of these features during preliminary feature selection before the model was applied. Other questions that seem to be important for the mother in both wave 2 and wave 5 are the mother's background and support. The father's wave 3 is also significant. The features noted revolve around the father's marital status ("f3e2b", "f3e2a1") , whether or not the father and mother were romantically involved ("f3d8", "f3d6"), the quality of the father's relationships ("f3d4b", f3d4a", "f3d4", "f3d3a7"), the father's prison history ("f3i29", "f3i31c2", "f3i31b", "f3i31a1", "f3i31a", "f3i30a", "f3i30", "f3i27c", "f3i27d2") and finally the father's reactions to being a parent ("f3b4l", "f3b6a", "f3b6c", "f3b6d").

6 Conclusion

Overall, lasso regression with median imputation and engineered features (with and without feature selection for variance and random forest importance) performed the best for gpa, grit, and material hardship. However, we suspect KNN imputation will outperform median imputation. One possible extension is to use a randomized lasso regression model which subsamples the training data and minimizes the l1 loss function for each of these subsamples. This often removes features that are highly correlated and the number of selected samples is not limited by sample size [7]. In regards to feature engineering, we could build models based on the best performing features we found in the current analysis. We could also build models out of one wave at a time instead of combining all of the waves at once. Further, feature engineering by matching across waves instead of matching across questions may yield more accurate results.

Acknowledgments

We would like to thank Prof. Engelhardt and Prof. Li and the rest of the course staff for the introduction to imputation and regression models, assignment specification, and for supplementary materials. We would also like to thank Matt and Ian for introducing us to the Fragile Families Challenge and providing a data set to work with.

Honor Code

We pledge our honour that this paper represents our own work in accordance with University regulations.

[s] Viola Mocz, Sonia Hashim

References

Honaker James, King Gary, Blackwell Matthew *Amelia II: A Program for Missing Data* In: Journal of Statistical Software, vol. 15, pp. 1 - 47; 2011. http://www.jstatsoft.org/v45/i07/

Kowarik Alexander and Templ Matthias *Imputation with the R Package VIM* In: Journal of Statistical Software, vol. 74, pp. 1 - 16; 2016.

Murphy, Kevin P: *Adaptive Computation and Machine Learning : Machine Learning : A Probabilistic Perspective.* Cambridge, US: MIT Press, 2012. ProQuest ebrary.

Pedregosa F, Varoquauz G, Gramford A, Michel V, Thirion B, Grisel O, Blondel M, pretternhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E: *Scikit-learn: Machine Learning in Python* In: Journal of Machine Learning Research, vol. 12, pp. 2825 - 2830; 2011.

Pedregosa F, Varoquauz G, Gramford A, Michel V, Thirion B, Grisel O, Blondel M, pretternhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E: *Scikit-learn: Machine Learning in Python User Guide: 1.1 Generalized Linear Model* In: Journal of Machine Learning Research, vol. 12, pp. 2825 - 2830; 2011. http://scikit-learn.org/stable/modules/linear_model.html

Pedregosa F, Varoquauz G, Gramford A, Michel V, Thirion B, Grisel O, Blondel M, pretternhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E: *Scikit-learn: Machine Learning in Python User Guide: 3.3 Model Evaluation: Quantifying the Quality of Predictions* In: Journal of Machine Learning Research, vol. 12, pp. 2825 - 2830; 2011. http://scikit-learn.org/stable/modules/model_evaluation.html

Wang S, Nan B, Rosset S, Zhu J: *Random Lasso* In: The Annuls of Applied Science, vol. 5, no. 1, pp. 468 - 485. 2011.