Predicting Contraceptive Use in Indonesia

Iliana Maifeld-Carucci. Alexa Giftopoulos, Shilpa Rajbhandari

April 16, 2018

---

1 Introduction

Contraceptive is a method or device used to prevent pregnancy. Although contraceptives have been around since ancient times, the accessibility or use may vary due to demographics, socio-economic factors, government policy, religious, or political reasons in certain cultures. Specifically, in Indonesia, a heavily Islamic, developing country, where the population totals about 261 million (fourth largest worldwide), we wanted to see if certain socio-economic factors contributed to a womens contraceptive use. To research this topic further, we will be analyzing a subset of data from the 1987 National Indonesia Contraceptive Prevalence Survey and predict women's preferred method of contraceptive use based on their respective demographic and socio-economic characteristics. We found our data from the UCI Machine Learning Repository[1].

2 Data Background and Pre-processing

Elaborating on the dataset mentioned above, the samples describe 1,473 married women in Indonesia, who were either not pregnant or did not know they were pregnant at the time of the interview. Our data contains a total of ten attributes: Contraceptive Method Used, Wife Age, Wife Education, Husband Education, Number of Children, Wife Religion, "Wife Working", Husband Occupation, Standard-of-Living, Media Exposure, and "Contraceptive Method Used". We had two numerical variables (Wife Age and Number of Children), three binary variables (Wife Religion, Media Exposure, and Wife Working), and the rest were ordinal (Wife Education, Husband Education, Husband Occupation, Contraceptive Method Used, and Standard of Living).

We began our analysis with a few data pre-processing steps, to ensure our data was clean and structured properly so our analysis would be unbiased and we would obtain accurate results. First, we checked that there were no null values in our data, and when that was successful, we made sure our variables were encoded properly and began to separate and assign our target and predictor variables. Thankfully our variables were already encoded and we didn't have any conflict between ordinal and nominal categorical variables. Our target variable in this analysis is Contraceptive Method Used, with the remaining nine features assigned as our predictor variables. Next, we randomly chose 70% of our data for training and 30% for testing, indicating a random state of zero, so our results are reproducible, and stratified our data by the class labels in our Contraceptive Method Used target variable so it structurally remains consistent.

[1]Data can be found here: https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice

To comply with assumptions of specific classification models we will be using in our analysis, we standardized our training and testing predictor variables (except for Random Forest) and calculated the correlation coefficients of our predictor variables to assess multicollinearity. Finally, to obtain the significant features in our data, to build more efficient models in our analysis, we trained a random forest classifier and utilized the feature importance attribute to estimate the significance of each predictor variable (Figure 1). Once we retrained the model with our sorted feature importance, the results showed that a model with only four features- Wife Age, Number of Children, Wife Education, and Standard of Living- had the most predictive power for explaining our target variable (Figure 2). To see the differences feature selection has on accuracy, we trained each model with all of our features and also with only these top four.

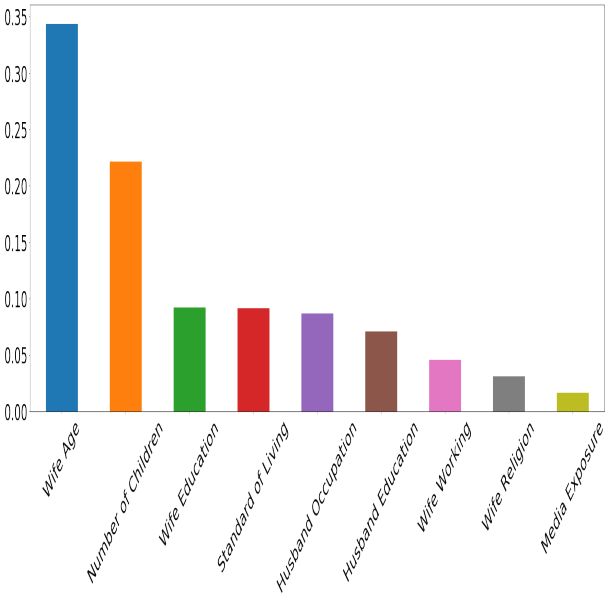

Figure 1

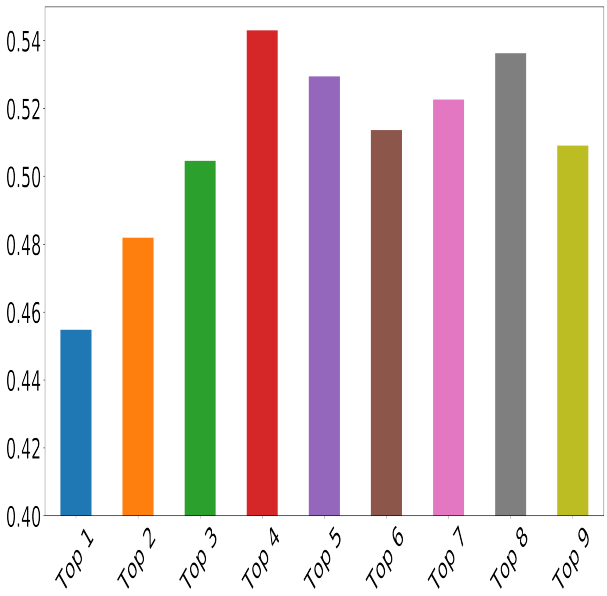

Figure 2

3 Our Models

To ensure we are obtaining the most accurate results possible, we will complete a robust analysis, applying various classification models to our data and selecting the model with the highest predictive accuracy. In the following pages, we will take you through our evaluation process for each model, discussing model assumptions, specifications, results, and why the respective model might or might not be a good fit for our data.

3.1 Random Forest

The first model we fit was a Random Forrest Classifier. This classifier is relatively robust, so it is not sensitive to multicollinearity, outliers, or susceptible to overfitting. Taking these assumptions about the algorithm into consideration, there was not much data preparation to do before training and testing the model.

When we originally fit our model using all of our features, we got a training accuracy of 0.96 and a testing accuracy/fscore of 0.52. When we re-trained our model using only the top four significant features estimated in the section above, it resulted in a training accuracy of 0.85 and a testing accuracy/fscore of 0.54.

As you can see, only using the significant features in our model provided us with a better fit and higher predictive power. It is interesting to note that the high training accuracy of both models seems unusual, as random forest classifiers in their nature are supposed to mitigate overfitting. This is one of the key reasons we use random forests instead of decision trees, because they estimate a large amount of decision trees and then output the mode of the results, decreasing the likelihood of overfitting. However, with may attempts to dissect what issue may be occurring, we still yet to have come to a conclusion. Most importantly, though, our predicting accuracy for both models is in line with the other models we will discuss below, so as long as the model can predict at the average standard, we can look past the training accuracy issue for now.

3.2 Logistic Regression

Next, we fit a multinomial logistic regression model to our data. While scaling the data is not necessary for logistic regression, it does help to speed up the convergence when using gradient descent to fit our model, so we standardized our X_train and X_test data before training the model. Logistic regression models are also robust to small noise, and not affected by mild cases of multicollinearity, which can be further tamed using L2 regularization.

In specifying the parameters of our model, taking into consideration the above assumptions, we implemented an L2 regularization parameter at an inverse regularization strength of 1.0, which is pretty strict, because our predictor variables, Wife Age and Number of Children were mildly correlated at 0.52, and Wife Education and Husband Education were mildly correlated at 0.62. These correlation values are no cause for concern but implementing the regularization parameter just plays it safe. In addition, we implemented a multinomial multi-class parameter because we have three target classes instead of two, so we needed to adjust our model for this additional class.

When we fit and predicted our first model containing all predictor variables, we received a training accuracy of 0.54 and a testing accuracy/fscore of 0.50. We also added a cross-validation test to reassure we were not under or overfitting our model, and we received a result of 0.52, which is very similar to our train and test accuracy, so we can conclude under and overfitting is not present in our model. Next, we decided to fit and predict our model containing only the four significant features estimated by the random forest classifier. This resulted in a training accuracy of 0.53 and a testing accuracy/fscore of 0.50. In comparing the two models, you can see the training accuracy actually dropped by about 0.01. Although this is a relatively small amount, limiting our model to the top four important features actually decreased any potential high variance during model training.

3.3 K-Nearest Neighbor

For our K-nearest neighbor algorithm we first standardized the data since it calculates the distance between data points as an indicator of a "typical" target category. We want to guard against different variances negatively affecting the learning of the algorithm and have all our features be unit independent. This algorithm is insensitive to data that may have outliers and is also useful for non-linear data, but it is sensitive to irrelevant features as we will see below.

In our application of k-nearest neighbor we chose k to be 36 because it is close to the rule of thumb of $k = \sqrt(n)$ where n is the number of samples in our training set. This value of k also maximized our testing accuracy while minimizing the difference in accuracy between our testing and training sets. Furthermore, a higher k is more resilient to outliers.

Next we fit and predicted our k-nearest neighbor model on all the predictor variable to find a training accuracy of .552 and a testing accuracy/ fscore of .514. We also applied a cross-validation test to check that we were not overfitting our model and recieved an accuracy score of .51. Because this is all in the same range we can conclude that while our model has generalized well, it didn't predict contraceptive use well. We proceeded to fit and predict our model on the top four features and received a training accuracy of .59 with a testing accuracy of .532. It is clear that for the k-nearest neighbor algorithm, it is indeed sensitive to irrelevant features and only selecting the top four features helps to improve the accuracy of the model noticeably.

3.4 Support Vector Machine

The final model that we tested was a Support Vector Machine (SVM). In this algorithm standardization of data is critical because input variables are combined via a distance function in a RBF kernel. In other words, the contribution of an input will depend heavily on its variability relative to other inputs. If the variance of one input variable is much greater than another, it may dominate the objective function and make the estimator unable to learn from other features correctly as expected. Aside from a need for standardization, SVM is a very robust algorithm that works well on smaller, cleaner datasets and can be more efficient because it uses a subset of the training points. Furthermore, because of convex optimization, the solution will definitely be the global minimum and not the local minimum. SVM is useful for both linearly separable and non-linearly separable data and in the case of non-linearly separable data we are able to use the kernel trick so that the data may be linearly separable in higher dimensional space. However, it is less effective when the data is noisier and the classes may overlap.

To apply SVM to our data we chose to use the RBF kernel because it maps to infinite dimensional space and is best for non-linearly separable data. In our case we chose gamma to be .08 and C to be 1 because we want the impact of a single training example to be great and a smoother decision surface. These choices also represent an effective trade off between the two values in terms of maximizing the accuracy of our model.

When we trained and predicted our first SVM model on all the features we found a training accuracy of .626 and a testing accuracy/ fscore of .55. Running cross-validation to guard against overfitting we got an accuracy score of .57. We then retrained our model on the top four features to receive a training accuracy of .587 and a testing accuracy/ fscore of .536. While our testing accuracy actually decreased slightly, our training accuracy also decreased, meaning that our model is generalizing better and has a lower variance when only running it on the top four features.

4 Best Classifier

The fscore results below show the predictive accuracy of each of our classifier models, and SVM predicts contraceptive use the best using all the explanatory variables (Figure 3).

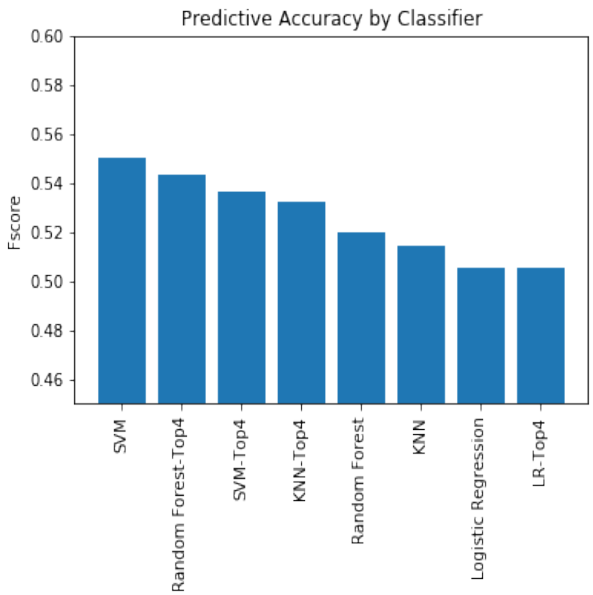

Figure 3

5 Conclusion

As our best classifier section shows, SVM did the best at predicting our testing set. One of the big reasons that SVM did well is because it can deal with non-linearly separable data unlike logistic regression. Furthermore, because we have several dimensions, SVM does better with high dimensional space. The non-linearity of our data is also a reason why random forest performed well because it does not expect a linear relationship between features. Logistic regression was our worst performing model because it assumes linearly separable data, linear relationships, and it doesn't perform well with binary and categorical data which we mostly have. Because k-nearest neighbor didn't perform as well as SVM it suggests that our classes are not easily

separable. However, when isolated to the top 4 features, KNN and SVM perform very similarly since KNN does better in low-dimensions.

Ultimately though, all of our models were very consistent and generalized well since our training and testing accuracies were almost exclusively close. Our models didn't fall into the trap of overfitting, but in fact were underfit since we had high bias, but low variance. This means that our features do not predict contraceptive use well and to get better predictions, we would need more and different features. This is a data issue and is not something that can be fixed by using machine learning techniques.