MS&E 226 Project

Predicting Depression Among Female Safety Net Program Participants

Abstract

In this study, we are looking to better understand depression in a low income country. The goal of this analysis is to provide insights to help allocate their poverty reduction funds in order to improve wellbeing.

I. Exploring data

i. Dataset: Psychosocial & socio- demographic variables from a safety net program survey

The data was collected through a survey for a large randomized experiment of a national safety net program for ultra poor households. This analysis uses psychological, social, and cultural variables as well as select sociodemographic variables. This sample is of recipients of this safety net program who have an average of 0-1 years of education and are in the lowest income households in each village. We are interested to generate an interpretable model to predict depression severity and depression.

Continuous response variable: We use a score (0-30) on a validated depression questionnaire called the CESD-R-10. Per protocol, these scores are the sum of responses to 10 questions about experiences of depressive symptoms over the past week (on a scale from 0 “None of the time / less than 1 day per week” to 3 “Most of the time / 5-7 days per week”). We have chosen this variable because depressive symptoms are a key indicator of mental health and wellbeing, an important policy outcome.

Binary response variable: Here we will use “depressed” / “not depressed”, where classification as Depressed = 1 for any score above the threshold of 13 and Depressed = 0 otherwise. This threshold is based on other research in African countries showing that it achieves the best balance of sensitivity and specificity.

ii. Using this dataset, we are interested in answering the following questions:

Question 1: Better understand how depression manifests in this context via its correlations with other psychological and cultural variables.

Question 2: Identify which sociodemographic factors are most strongly associated with depression to consider candidate causal drivers of depression (causal inference to be examined in other studies).

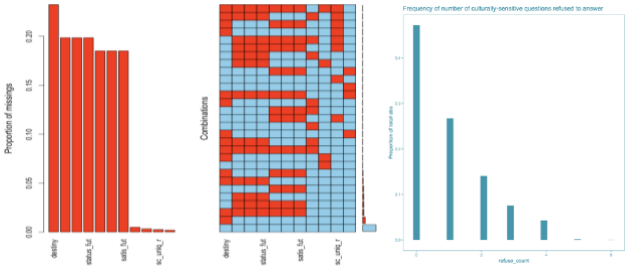
iii. Data cleaning

Concern: We have some concerns over the extent of missingness on some variables. Additionally, it appears that there is significant noise on many variables, given that pairwise correlations are low, even on variables that are similarly worded and from the same psychological scale. Thus, the absolute value of our prediction error may be higher than we might expect compared to, for example, a more educated population more familiar with surveys.

Insight: 53% of observations have at least one missing on 1 of 8 culturally-sensitive questions about the future or one's obligations to others. Piloting in some of these communities suggested that this missingness was not random. Specifically, we found that some respondents considered it arrogant to presume to know one's fate or to know what others think and thus refused to respond.

iv. Strategies for dealing with missingness:

First, our strategy is to create a new variable that is a count of refusals to these 8 questions per observation (refuse_count, see Fig 1&2). This variable ranges from 0-6. Instead of adding indicator variables for missingness for each variable, we will use this count variable. Statistically, it allows us to control for missingness on these variables and, conceptually, it may provide more individual-level information on our respondents (e.g. extent of collectivism / traditionalism).



FIGURES 1&2. *Missingnesssensitive questions and & frequency of new ‘refuse_count’ variable*

Insight: We find that the more of these questions respondents refused, the higher they scored on measures of collectivism, lower on measures of individualism, and thus variable missingness is not random. Interestingly, we did not see evidence for bias in missingness for these variables with visualization methods (e.g. matrixplot, marginplot) (Fig 3). If we

correlate missingness with all variables in the dataset, the correlations are very low.

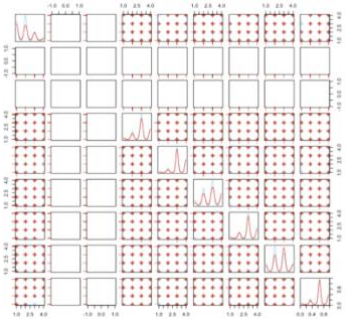


FIGURE 3. *Visualizing missing vs observed distributions on sensitive questions with scattmatrixMiss*

However, we have reason to believe that the missingness for these culturally-sensitive questions is non-random and the refuse_count variable better allows us to see this small, but significant, non-randomness (see Fig 4).

Second, luckily, for a few of these variables, the missingness is trivial (<0.5% of the data). We follow Gelman (2006) to impute the mean of non-missing observations for the trivial number of missing values for these select variables. Even though this method can create bias in our estimates by overweighting observations that may differ from those with missing data and by constraining variance, we do not think it will create bias in this case because the number of missings is low and because the refuse_count variable will absorb some of this bias.

Third, for the categorical variables that have non-trivial amount of missingness, we create an additional level of the factor called "refused" (Gelman, 2006).

Lastly, for the variable close_husb, the missing data mechanism is known (the women have no husband), thus the value is set to 0, and we create an indicator variable (have_husb) to interact with close_husb.

We remove 'commune' as it is collinear with 'region'.

II. Data transformation & Prediction

i. Correlations: Covariate selection

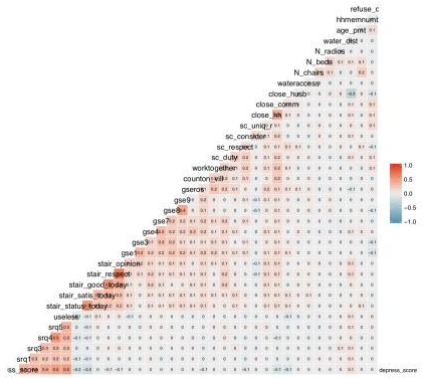


FIGURE 4. *Examining pairwise associations*

We see that about 27 of the variables appear associated with depression scores and they mostly fall into three conceptual categories: functioning, collectivism and closeness to community, and poverty.

Insight: We see that depression is negatively correlated to performance (work functioning), and it appears that this relationship is buffered by physical health. In other words, if you are depressed but your digestive system is still functioning well, your work performance does not suffer as much. We expect to see interactions between physical vs work functioning on depression. Indeed we see that, in our model with all interactions, many of the significant interactions ($p<0.01$) show that better physical health (i.e. lower srq1,3) is associated with an attenuated relationship between work functioning (srq4,5) and depression.

Insight: We see that when we regress all variables in our base model, the coefficients on the self-efficacy variables (gse) become positive, which is the opposite direction of association as we see in the West (i.e. typically has negative association). We believe this occurs because self-efficacy variables are positively related to one's subjective social status and satisfaction, and these two outcomes soak up much of the variance in depression scores, leaving less residual variance for the gse variables.

Insight: We see that the third of women who have cell phones have lower depression scores. However, this association is likely due to a *transitive relationship* with poverty level, given that being poorer is associated with not having a phone and also with depression.

ii. Variable transformation: Due to skewness, we log transform age_pmt, hhmemnumber, N_beds, N_chairs, N_radios, water_dist (after adding 0.5 to each value). We log transform variables that are non-normally distributed. While these changes do not affect

the in-sample RMSE, we will retain them as they help meet inferential assumptions of regression.

In addition, we see that poverty_pmt, a composite measure of poverty level, has two very distinct clusters of values, one around zero and the other around 1e+14. These are not interpretable values. We suspect that this variable was used for targeting a public program to the most poor individuals in a village and not those who were less poor. Thus, for interpretability, we decide to create a binary variable called poverty_cat being "very poor" and "less poor".

iii. Prediction: Our goal is to build a model that is interpretable, meaning parsimonious, and reduces the CV RMSE. Our dataset is large, so we first use visual inspection and statistical tests of all pairwise correlations with the continuous outcome to identify non-zero correlations and use lasso with a low penalty to retain a lesser number of variables (27) in subsequent models.

iv. Model building steps and findings

Base model: We choose our base model to be a complete case analysis including all possible covariates in an additive linear model. We chose this because excluding cases with missings is a common practice in development economics. We also think including all covariates is a good first step as it makes use of all of the available information from predictors in this dataset.

Model 1: All interactions: We include all possible interactions in a model from the reduced covariate set.

Model 2: All significant interactions and higher order terms: Scatterplots show some variables to have a negative quadratic relationship with depression score (srq1-5, useless, age) (Fig 5). For model 3, we include interactions significant (at the 5% level) from model 2 and add negative second order terms. All variables in this model were from the reduced covariate set.

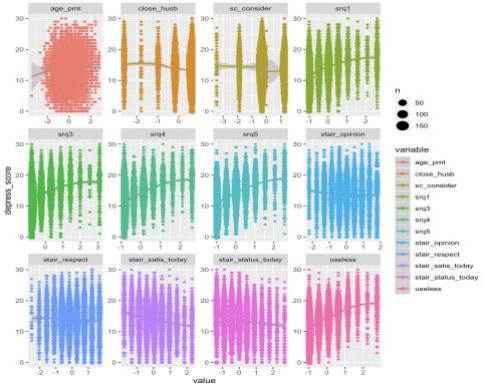


FIGURE 5. *Choosing higher order terms*

Model 3: Backward stepwise regression: We conduct backward stepwise selection from model 1. The model with the lowest AIC has 17 coefficients and an R-squared value of 0.44.

Model 4: Lasso with all interactions: We run lasso on a model with pairwise interactions on the reduced covariate set. We obtain a model with the minimum lambda (0.105) which retains 88 coefficients.

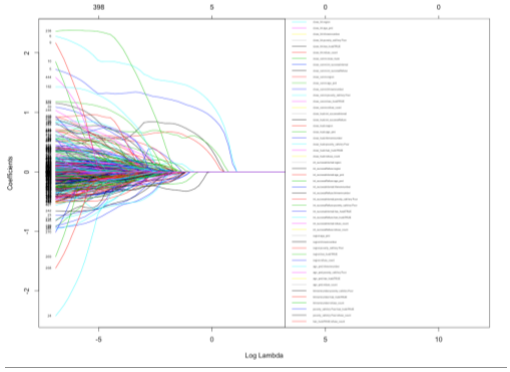
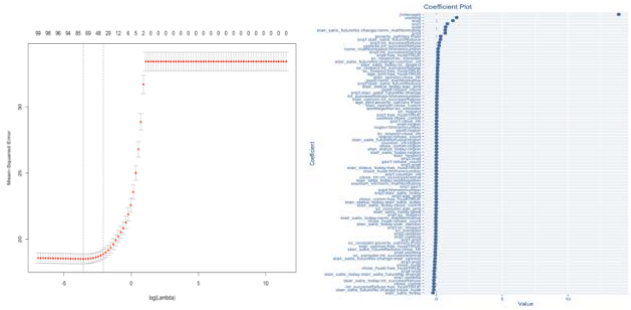


FIGURE 6. *Shrinkage of lasso model coefficients*



FIGURES 7 & 8. *Lasso: choosing the lambda value and examining coefficient shrinkage for the best model*

Model 5: Ridge regression with all interactions

We conduct model 4 but with ridge (alpha=0) which results in a model with 461 coefficients.

III. Model Comparison: Best model

TABLE 1*. Comparison of models, continuous outcome*

	CV RMSE on Train Set
Base Model	4.434
All Pairwise Interactions	6.228
Ridge with all interactions	4.448
Lasso with all interactions	4.385
Backward Stepwise Selection	4.347
*Best: Significant Interactions and Hi Order Terms	4.322

We choose the model with significant interactions and higher order terms because it generates the lowest CV RMSE. Compared to the base model, this model is also more interpretable to some extent as it has fewer covariates. A benefit of a more parsimonious model is that, if the program for instance wanted to target those at risk for depression, they would need to collect fewer variables. Compared to a model with all interactions, this model is less likely to be overfit to the training data and includes a select few interactions that are significant and interpretable (e.g. between physical and work functioning variables, as noted above).

IV. Predicting Binary Variable: Depressed

Base model: As with the continuous outcome, our base logistic regression model includes all covariates from the full train set in a complete case analysis. We run basic logistic regression with all covariates and obtain a 0-1 loss of 0.256, CV 0-1 loss of 0.273, and AUC of 0.832.

	Predicted 0	Predicted 1	Total
Actual 0	717	243	960
Actual 1	279	797	1076
Total	996	1040	2036

All models use the same subset of variables used for the models on the continuous outcome.

Model 1: Interactions & higher order terms: Our model with significant interactions and higher order terms produced 0-1 loss of 0.244, CV 0-1 loss of 0.255 and an AUC of 0.839. This model used variables from the reduced covariate set.

	Predicted 0	Predicted 1	Total
Actual 0	1269	435	1704
Actual 1	452	1478	1930
Total	1721	1913	3634

Model 2: Forward stepwise selection: We used forward stepwise selection based on the AIC model score, which penalizes excessive model complexity, from a model with all interacting predictors. The model with the lowest AIC value of 3742 had 37 coefficients. The model produced 0-1 loss of 0.252, CV 0-1 loss of 0.259 and AUC of 0.831.

Model 3: Lasso model: We construct a lasso model with all interactions. Our lasso model from minimum lambda (0.11) has CV 0-1 loss 0.256 and AUC 0.821.

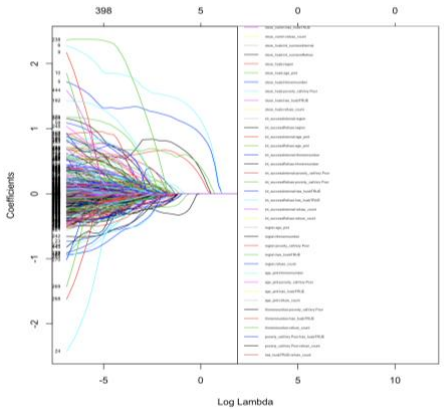


***FIGURE 9**. Shrinkage of lasso model coefficients*

Model 4. K Nearest Neighbors

We run a knn analysis with all 48 numeric predictors. While we tested values for k of 1, 5, 10, 20, 50, 100, and 200, we find the lowest CV 0-1 loss of 0.280 where k=50. We do not proceed with this model because it underperforms compared to others analyzed.

V. Model Comparison: Best Model

***TABLE 2**. Comparison of models, binary outcome*

	CV 0-1 Loss on Train Set
Base Model	0.268
KNN	0.280
Forward Stepwise Selection	0.259
Lasso with all interactions	0.256
Significant Interactions and Hi Order Terms	0.255

We see that the best model, which has the lowest CV 0-1 Loss (0.255) is the same model as that for the continuous outcome -- the model with significant interactions and higher order terms. We compute the area under the curve (AUC) for this best model to be 0.839 which outperforms that for the baseline of 0.832.

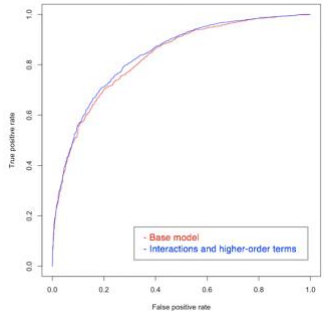


***FIGURE 10.** AUC for the base and best models*

Conclusion and interpretation: We believe that the best models for the continuous and binary outcomes, which include significant interactions and higher order terms, are preferable to the base models because they account for missingness (e.g. by adding a count for refusals, creating levels for missingness). For this reason, we believe they should be less biased and more representative of the true population than a model which ignores this information. We also think that, given our available data and compared to other models, the bias and variance of the best model will be relatively minimal and well balanced. This is because we selected the full subset of covariates that were associated with depression, which can help reduce bias, and removed the variables that were not associated with depression and thus could add noise to the model and increase variance. However, we still think that our best model's CV error might be an overestimate of the test error because it includes many higher order terms and interactions and thus is at risk of being overfit to the training data.

Lastly, while we are interested to predict depression with the greatest degree of accuracy, we are also interested in an interpretable model that can help us think about which variables may be important risk factors for depression, which we will explore in Part 2. For this reason, we prefer the relative parsimony of our best model that includes a subsample of the covariates and interaction terms.

Code: MSE226_Project_Code_Part1

Load Relevant Libraries and Functions

```
library(pacman)
p_load(tidyverse, ggplot2, forcats, ggthemes, glmnet, psych, knitr, Hmisc, corrplot,
GGally, psy, MASS, cvTools, car, here, VIM, mice, class, coefplot, pROC, plyr,
regclass, boot, ROCR)
cost_function = function(y, y_hat) {
    mean(as.integer(y_hat > 0.5) != y)
}
here::here()
df.train <- read_csv("df.train.csv", col_names=TRUE, na = c("", "NA"))
df.train <- subset(df.train, select = -X1)
df.train.base.cont <- subset(df.train, select = -depressed)
df.train.base.bin <- subset(df.train, select = -depress_score)
```

Data Cleaning

Missing data

```
colSums(is.na(df.train)) # destiny 845, stair_status_future 722, stair_satis_future
673, sc_duty 18, sc_consider 12, sc_uniq_r 9, sc_respect 7, close_husb 346

# create new variable for extent of missings
df.train$refuse_count <- is.na(df.train$destiny) +
is.na(df.train$stair_status_future) + is.na(df.train$stair_satis_future) +
is.na(df.train$sc_duty) + is.na(df.train$sc_consider) + is.na(df.train$sc_uniq_r) +
is.na(df.train$sc_respect) + as.logical(df.train$int_success=="Refuse")

# known missingness mechanism
df.train$close_husb[is.na(df.train$close_husb)] <- 0
df.train$has_husb <- df.train$close_husb>0  # new indicator var

#trivial missings
df.train$sc_duty[is.na(df.train$sc_duty)] <- mean(df.train$sc_duty, na.rm=T)
df.train$sc_consider[is.na(df.train$sc_consider)] <- mean(df.train$sc_consider,
na.rm=T)
df.train$sc_uniq_r[is.na(df.train$sc_uniq_r)] <- mean(df.train$sc_uniq_r, na.rm=T)
df.train$sc_respect[is.na(df.train$sc_respect)] <- mean(df.train$sc_respect, na.rm=T)
df.train <- subset(df.train, select = -destiny)

# categorical vars
df.train$stair_status_future[is.na(df.train$stair_status_future)] <- "Refused"
df.train$stair_satis_future[is.na(df.train$stair_satis_future)] <- "Refused"

# improving inteerpretation
df.train$poverty_cat = ifelse(df.train$poverty_pmt>1.0e+14, "Less Poor", "Very Poor")
df.train <- subset(df.train, select = -poverty_pmt)

# variable transformation
df.train$N_chairs <- df.train$N_chairs + 0.5
df.train$N_beds <- df.train$N_beds + 0.5
df.train$N_radios <- df.train$N_radios + 0.5
df.train$water_dist <- df.train$water_dist + 0.5
log_cols = c("age_pmt", "hhmemnumber", "N_beds","N_chairs", "N_radios", "water_dist")
df.train[log_cols] = log(df.train[log_cols])
```

```
# Standardize vars
# Continous dataset
df.train.cont <- df.train %>% dplyr::select(-depressed) %>% mutate_at(vars(-
depress_score, -region, -stair_status_future, -stair_satis_future, -int_success, -
poverty_cat, -norm_mat, -norm_fert, -norm_save, -is_head,-has_toilet, -onoutskirts, -
telephone, -has_husb), funs(scale))

# Binary dataset
df.train.bin <- df.train %>% dplyr::select(-depress_score) %>% mutate_at(vars(-
depressed, -region, -stair_status_future, -stair_satis_future, -int_success, -
poverty_cat, -norm_mat, -norm_fert, -norm_save, -is_head,-has_toilet, -onoutskirts, -
telephone, -has_husb), funs(scale))
```

Model Building for the Continuous Outcome

Model 0: Base Model - Linear regression with all covars

```
df.train.base.cont <- na.omit(df.train.cont)
mod.base <- lm(depress_score ~ ., data=df.train.base.cont)
rmse.base = sqrt(mean(mod.base$residuals^2)) # 4.361613, R-squared 0.4309
cv.base = cvFit(mod.base, data=df.train.base.cont,
y=df.train.base.cont$depress_score, K = 10, seed=123) # 4.434165
```

Choosing a subselection of variables

```
Y.train_all <- df.train.cont$depress_score
X.train_all <- model.matrix(depress_score ~ ., df.train.cont)
model_select <- glmnet(X.train_all, Y.train_all, alpha = 1, lambda = .105,
standardize=TRUE) # mimics results of pairwise correlations
coef(model_select) # put these vars in sub datasets below

df.train.cont.sub <- df.train.cont %>% dplyr::select(depress_score, srq1, srq3, srq4,
srq5, useless, stair_status_today, stair_satis_today, stair_satis_future,
stair_respect, stair_opinion, gse1, gse8,  counton_vill, worktogether, sc_respect,
sc_consider, norm_mat, close_hh, close_comm, close_husb, int_success, region,
age_pmt, hhmemnumber, poverty_cat, has_husb, refuse_count)
df.train.bin.sub <- df.train.bin %>% dplyr::select(depressed, srq1, srq3, srq4, srq5,
useless, stair_status_today, stair_satis_today, stair_satis_future, stair_respect,
stair_opinion, gse1, gse8,  counton_vill, worktogether, sc_respect, sc_consider,
norm_mat, close_hh, close_comm, close_husb, int_success, region, age_pmt,
hhmemnumber, poverty_cat, has_husb, refuse_count)
```

Model 1: All interactions

```
mod.all.int <- lm(depress_score ~ . + .:., data=df.train.cont.sub)
rmse.all.int = sqrt(mean(mod.all.int$residuals^2)) # 3.966521, R-squared 0.5293
cv.all.int = cvFit(mod.all.int, data=df.train.cont, y=df.train.cont$depress_score, K
= 10, seed=123) # 6.228431
```

Model 2: Significant interactions plus higher order terms

```
mod.int.hiorder <- lm(depress_score ~ . - I(srq1^2) - I(srq3^2) - I(srq4^2) -
I(srq5^2) - I(useless^2) - I(age_pmt^2) + srq1*useless + srq1*close_husb +
srq1*has_husb + srq1*poverty_cat + srq3*srq5 + srq3*gse1 + srq4*srq5 + srq4*norm_mat
+ srq4*close_comm + srq5*worktogether + srq5*sc_respect + useless*int_success +
useless*region + useless*poverty_cat + stair_status_today*age_pmt +
stair_status_today*stair_satis_future + stair_satis_today*stair_satis_future +
stair_satis_today*region + stair_satis_future*counton_vill
```

```
+stair_satis_future*worktogether+ stair_satis_future*sc_respect+ +
stair_satis_future*norm_mat + stair_satis_future*close_husb +
stair_satis_future*region + stair_respect*gse1 + stair_respect*gse8
+stair_respect*norm_mat+ stair_opinion*gse8 + stair_opinion*int_success+
stair_opinion*age_pmt + gse1*close_hh+ gse1*region + gse8*norm_mat+ gse8*region+
gse8*poverty_cat+ gse8*refuse_count+ worktogether*refuse_count+
sc_respect*sc_consider+sc_respect*close_hh+sc_respect*refuse_count+
close_comm*close_husb + close_husb*refuse_count + int_success*hhmemnumber +
age_pmt*poverty_cat + has_husb*close_husb, data=df.train.cont.sub)
rmse.int.hiorder = sqrt(mean(mod.int.hiorder$residuals^2)) # 4.201945, R-squared:
0.5293
cv.int.hiorder = cvFit(mod.int.hiorder, data=df.train.cont.sub,
y=df.train.cont.sub$depress_score, K = 10, seed=123) # 4.322006
```

Model 3: Backward Stepwise Selection

```
lm.step.bic.bw = stepAIC(mod.int.hiorder, direction="backward",
k=log(nrow(df.train.cont.sub)))
rmse.bw = sqrt(mean(lm.step.bic.bw$residuals^2)) # 4.325365; R-squared 0.4403
cv.bw <- cvFit(lm.step.bic.bw, data=df.train.cont.sub,
y=df.train.cont.sub$depress_score, K = 10, seed=123) # 4.347284
```

Model 4: Lasso with interactions

```
set.seed(123)
Y.train <- df.train.cont.sub$depress_score
lambdas <- 10^seq(-3, 5, length.out = 100)
X.train <- model.matrix(depress_score ~ . + .:., df.train.cont.sub)
fm.lasso_cv <- cv.glmnet(X.train, Y.train, alpha = 1, lambda = lambdas,
                         standardize = TRUE, thresh = 1e-12, nfolds=10)
lambda_cv <- fm.lasso_cv$lambda.min # 0.1047616
model_cv <- glmnet(X.train, Y.train, alpha = 1, lambda = lambda_cv)
y_hat_cv <- predict(model_cv, X.train)
rmse.lasso = sqrt(mean((y_hat_cv - Y.train)^2)) # 4.268273
cv.lasso <- sqrt(fm.lasso_cv$cvm[fm.lasso_cv$lambda == fm.lasso_cv$lambda.min]) #
4.385421
```

Model 5: Ridge with interactions

```
set.seed(123)
Y.train <- df.train.cont.sub$depress_score
lambdas <- 10^seq(-3, 5, length.out = 100)
X.train <- model.matrix(depress_score ~ . + .:., df.train.cont.sub)
fm.ridge_cv <- cv.glmnet(X.train, Y.train, alpha = 0, lambda = lambdas,
                         standardize = TRUE, thresh = 1e-12, nfolds=10)
lambda.ridge_cv <- fm.ridge_cv$lambda.min # 3.593814
model.ridge_cv <- glmnet(X.train, Y.train, alpha = 0, lambda = lambda.ridge_cv)
y_hat_ridge_cv <- predict(model.ridge_cv, X.train)
rmse.ridge = sqrt(mean((y_hat_ridge_cv - Y.train)^2)) # 4.148169
cv.ridge <- sqrt(fm.ridge_cv$cvm[fm.ridge_cv$lambda == fm.ridge_cv$lambda.min]) #
4.447674
```

Performance of all models for continuous outcome

```
vec.cvrmse <- c(cv.base$cv, cv.all.int$cv, cv.ridge, cv.lasso, cv.bw$cv,
cv.int.hiorder$cv)
mat <- cbind(vec.cvrmse)
colnames(mat) <- c("CV RMSE on Train Set")
```

```
rownames(mat) <- c("Base Model", "All Pairwise Interactions",  "Ridge with all
interactions", "Lasso with all interactions", "Backward Stepwise Selection", "*Best:
Significant Interactions and Hi Order Terms")
df.mat <- as.data.frame(mat)
knitr::kable(df.mat, format = "pandoc", digits=round(3))
```

Model Building for the Binary Outcome

Base Model - Logistic regression with all covars

```
df.train.base.bin <- na.omit(df.train.base.bin)
mod.base.glm = glm(depressed ~ ., family = binomial(), data = df.train.base.bin) #
AIC: 2171.6
fitted.base.glm = fitted(mod.base.glm)
confusion_matrix(mod.base.glm)
#          Predicted 0 Predicted 1 Total
# Actual 0         717         243   960
# Actual 1         279         797  1076
# Total            996        1040  2036
roc_base_data = data.frame(fit = fitted.base.glm, obs = df.train.base.bin$depressed)
my_base_roc = roc(roc_base_data$obs ~ roc_base_data$fit, plot = FALSE)
base_auc <- round(pROC::auc(my_base_roc), digits=3) # AUC =  0.829
mod.base.error = mean(as.integer(mod.base.glm$fitted.values > 0.5) !=
df.train.base.bin$depressed) # 0.2563851
set.seed(123)
base.bin.cv = cv.glm(data=df.train.base.bin, glmfit=mod.base.glm, cost=cost_function,
K=10)
base.bin.cv.error = base.bin.cv$delta[1] # 0.2725933
```

Model 1: Significant interactions and higher order terms

```
mod.bin.int.hiorder <- glm(depressed ~ . - I(srq1^2) - I(srq3^2) - I(srq4^2) -
I(srq5^2) - I(useless^2) - I(age_pmt^2) + srq1*useless + srq1*close_husb +
srq1*has_husb + srq1*poverty_cat + srq3*srq5 + srq3*gse1 + srq4*srq5 + srq4*norm_mat
+ srq4*close_comm + srq5*worktogether + srq5*sc_respect + useless*int_success +
useless*region + useless*poverty_cat + stair_status_today*age_pmt +
stair_status_today*stair_satis_future + stair_satis_today*stair_satis_future +
stair_satis_today*region + stair_satis_future*counton_vill
+stair_satis_future*worktogether+ stair_satis_future*sc_respect+ +
stair_satis_future*norm_mat + stair_satis_future*close_husb +
stair_satis_future*region + stair_respect*gse1 + stair_respect*gse8
+stair_respect*norm_mat+ stair_opinion*gse8 + stair_opinion*int_success+
stair_opinion*age_pmt + gse1*close_hh+ gse1*region + gse8*norm_mat+ gse8*region+
gse8*poverty_cat+ gse8*refuse_count+ worktogether*refuse_count+
sc_respect*sc_consider+sc_respect*close_hh+sc_respect*refuse_count+
close_comm*close_husb + close_husb*refuse_count + int_success*hhmemnumber +
age_pmt*poverty_cat + has_husb*close_husb, family=binomial, data=df.train.bin.sub) #
AIC: 3788.3
fitted.int.hiorder.glm = fitted(mod.bin.int.hiorder)
confusion_matrix(mod.bin.int.hiorder)
#          Predicted 0 Predicted 1 Total
# Actual 0        1269         435  1704
# Actual 1         452        1478  1930
# Total           1721        1913  3634
roc_int.hiorder_data = data.frame(fit = fitted.int.hiorder.glm, obs =
df.train.bin.sub$depressed)
my_int.hiorder_roc = roc(roc_int.hiorder_data$obs ~ roc_int.hiorder_data$fit, plot =
FALSE)
```

```
int.hiorder_auc <- round(pROC::auc(my_int.hiorder_roc), digits=3) # AUC =  0.8386
mod.int.hiorder.error = mean(as.integer(mod.bin.int.hiorder$fitted.values > 0.5) !=
df.train.bin.sub$depressed) # 0.2440837
set.seed(123)
int.hiorder.bin.cv = cv.glm(data=df.train.bin.sub, glmfit=mod.bin.int.hiorder,
cost=cost_function, K=10)
int.hiorder.bin.cv.error = int.hiorder.bin.cv$delta[1] # 0.255366
```

Model 2: Forward Stepwise Selection

```
full_model = glm(depressed ~ . + .:., family = binomial(), data = df.train.bin.sub)
summary(full_model)
null_model = glm(depressed ~ 1, family = binomial(), data = df.train.bin.sub)
summary(null_model)
step(null_model, list(upper = full_model), direction = 'forward') # 3742
best_AIC_model_forward = glm(formula = depressed ~ useless + srq5 + srq1 + srq3 +
srq4 +
    stair_satis_today + close_comm + close_husb + sc_respect +
    hhmemnumber + poverty_cat + gse1 + sc_consider + close_hh +
    srq1:srq3 + stair_satis_today:close_comm + useless:poverty_cat +
    srq1:poverty_cat + stair_satis_today:sc_respect + useless:gse1 +
    sc_respect:sc_consider + useless:sc_consider + useless:stair_satis_today +
    srq1:sc_consider + useless:srq1 + srq3:poverty_cat + srq5:srq3 +
    stair_satis_today:hhmemnumber + useless:close_husb + srq3:close_husb +
    useless:srq5 + srq1:sc_respect + stair_satis_today:poverty_cat +
    gse1:close_hh + close_comm:gse1 + sc_respect:close_hh, family = binomial, data =
df.train.bin)
fitted_model_forward = fitted(best_AIC_model_forward)

roc_fw_data = data.frame(fit = fitted_model_forward, obs = df.train.bin$depressed)
my_fw_roc = roc(roc_fw_data$obs ~ roc_fw_data$fit, plot = FALSE)
fw_auc <- round(pROC::auc(my_fw_roc), digits=3) # AUC =  0.831
mod.fw.error = mean(as.integer(best_AIC_model_forward$fitted.values > 0.5) !=
df.train.bin$depressed) # 0.2517887
set.seed(123)
fw.bin.cv = cv.glm(data=df.train.bin, glmfit=best_AIC_model_forward,
cost=cost_function, K=10)
fw.bin.cv.error = fw.bin.cv$delta[1] # 0.2592185
```

Model 3: Lasso

```
set.seed(123)
Y.train.bin <- as.factor(df.train.bin.sub$depressed)
lambdas <- 10^seq(-3, 5, length.out = 100)
X.train.bin <- model.matrix(depressed ~ . + .:., df.train.bin.sub, family=binomial)
fm.lasso.bin_cv = cv.glmnet(X.train.bin, Y.train.bin, alpha =1, family = "binomial",
type.measure = "class", lambda = lambdas, standardize = TRUE, thresh = 1e-12,
nfolds=10)
lambda.bin_cv <- fm.lasso.bin_cv$lambda.min # 0.01123324
cv.bin.lasso <- fm.lasso.bin_cv$cvm[fm.lasso.bin_cv$lambda ==
fm.lasso.bin_cv$lambda.min] # 0.2564667
fm.lasso.auc_cv = cv.glmnet(X.train.bin, Y.train.bin, alpha =1, family = "binomial",
type.measure = "auc",lambda = lambdas, standardize = TRUE, thresh = 1e-12, nfolds=10)
cv.auc.lasso <- fm.lasso.auc_cv$cvm[fm.lasso.auc_cv$lambda ==
fm.lasso.auc_cv$lambda.min] # 0.821675
```

Model 4: K-Nearest-Neighbors

```
df.train.bin.num <-  df.train.bin %>% dplyr::select(-region, -stair_status_future, -
stair_satis_future , - int_success, - poverty_cat, - norm_mat, -onoutskirts, -
has_husb, -norm_mat, -norm_fert, -norm_save, -commune, -is_head, -has_toilet, -
onoutskirts, -telephone, -has_husb)

## take out a fold. basically do CV.
set.seed(123)
n = nrow(df.train.bin.num)
val_idx = base::sample(n, floor(0.2*n))
data_tr = df.train.bin.num[-val_idx,]
data_val = df.train.bin.num[val_idx,]

dtX = dplyr::select(data_tr, -depressed)
dvX = dplyr::select(data_val, -depressed)

set.seed(110)
ks = c(1, 5, 10, 20, 50, 100, 200)
knns = lapply(ks, function(x) {
  knn(dtX, dvX, data_tr$depressed, k = x)
})

loss01 = function(preds, response) {
  mean(preds != response)
}

response = data_val$depressed
l01_knn = vapply(knns, function(x) {
  loss01(x, response)
}, 0.0)
names(l01_knn) = ks

#         1         5        10        20        50       100       200
# 0.3870523 0.3319559 0.3250689 0.2947658 0.2796143 0.2823691 0.2851240
knn.err <- 0.2796143
```

Performance of all models for binary outcome

```
vec.cv01loss <- c(base.bin.cv.error, knn.err, fw.bin.cv.error, cv.bin.lasso,
int.hiorder.bin.cv.error)
mat <- cbind(vec.cv01loss)
colnames(mat) <- c("CV 0-1 Loss on Train Set")
rownames(mat) <- c("Base Model","KNN", "Forward Stepwise Selection", "Lasso with all
interactions", "*Best: Significant Interactions and Hi Order Terms")
df.mat <- as.data.frame(mat)
knitr::kable(df.mat, format = "pandoc", digits=round(3))
```

MS&E 226 Project - Part 2

Predicting Depression Among Female Safety Net Program Participants

Summary of Part 1

In this study, we are looking to generate an interpretable model to predict depression and understand its manifestation among the poorest households. The goal of this analysis is to provide insights to help allocate their poverty reduction funds in order to improve wellbeing. This analysis uses psychological, social, cultural variables and select sociodemographic variables. Our focus:

Question 1: *Better understand how depression manifests in this context via its correlations with other psychological and cultural variables.*

Question 2: *Identify which sociodemographic factors are most strongly associated with depression to consider candidate causal drivers of depression (causal inference to be examined in other studies).*

In Part 1, we performed diagnostics on missingness and developed strategies for different types of missing data, e.g. adding a count of refusals variable. We log transformed certain variables that had negative quadratic relationships with depression. Our best model was developed by choosing significant predictors and interaction terms, according to pairwise associations, lasso, and full data regression models. This model had lower CV RMSE and CV 0-1 loss for linear and logistic models, respectively, compared to models with all covariates, all interaction terms, forward/backward stepwise selection, K Nearest Neighbors (KNN), lasso, and ridge.

VI. Prediction on the test set

	Train Set	Test Set
CV RMSE	4.322	4.632
CV 0-1 Loss	0.255	0.477

We see for both linear and logistic regression models that error is much higher on the test data than we estimated (even using cross-validation) on the train set.

➶ Insight: *Why are we observing this?*

- First, by selecting covariates and interactions to include based on lasso, pairwise associations with the continuous outcome, and models with all possible terms, we favorably biased our models to the noise in the train set.
- Second, we included a large number of interaction terms, which seems to have led to overfitting to the train data.
- Third, by selecting covariates primarily based on associations with the linear outcome, the discrepancy between the train and test error was worse for the binary outcome than for the continuous outcome.

VII. Inference

i. Statistical significance of coefficients:

We are choosing the linear regression model for the inference task. In our chosen model, we see that the following single variables are significantly associated with depressive score both in the chosen multivariate regression model and in pairwise associations: srq3, srq4, srq5, useless, region, stair_satis_today, stair_statuts_today, stair_opinion, counton_vill, sc_consider, close_comm, close_husb, poverty_cat.

In this context, statistical significance means that the test statistic had an associated p-value less than 0.05. This p-value gives you the probability that we would have observed data as extreme from the null of zero as what we saw, given a true null. Running linear regression with a *t* distribution for hypothesis tests assumes that the data come from a normal distribution with equal variance. The test value is the ratio of the coefficient to the standard error and the coefficients in multivariate regression are adjusted for all other covariates included in the model.

We are not inclined to put much faith in the significance of results for a few reasons. First, the coefficient terms and their significance jump around from model to model. For instance, we see that many of the coefficients in the chosen model are no longer significant for variables that had a significant bivariate correlation with depression or in models where we included all possible interactions. (One reason for this can be that certain covariates are not associated with the *residual* variation in the outcome given variation explained by other covariates in the model.) Note that we also may not trust results of bivariate associations as they don't control for possible confounders that are included in the multivariate regression.

Second, the p-values are likely to be overly optimistic given that we selectively refined the model based on p values for associations in the train data. Specifically, we dropped covariates and interaction terms that were not significant in preliminary models. Thus, we predict the coefficients to not be as significant on the test data. Third, see multiple hypothesis section below.

ii. Our model on the test data

While for the chosen model in the train set 47.9% of coefficients are significant, in the test set only 10.6% of the 95 coefficients are significant. Part of the reason for this is, as described above, that the p-values are larger on average in the test set (i.e. the p-values are underestimates in the train set). On the train data, the mean p-value is 0.217 while in the test data it is 0.433, about twice as large. We see that most of the variables which had larger significant p-values (between 0.01 and 0.05) on the train data are no longer significant in the test data.

Visually, we see in Figures 11 & 12 that (a) the variance of the coefficients in the test set are wider (i.e. the intervals contain zero) and (b) the average magnitude of the coefficients is slightly closer to zero in the test set (Mean $Coef_{test}$ = 0.24) than the train set (Mean $Coef_{train}$ = 0.29). Because the standard errors are larger and the coefficient magnitudes are slightly smaller in the test set than the train set, fewer coefficients are significant.

➶ **Insight:** *Why are we observing this?* One reason why the p-values may have been underestimated in the train set compared to the test set is that the model was developed in a way that was favorable to the noise of particular data in the train set (e.g. selection of covariates & interactions). Second, the test set (n=907) is much smaller than the train set (n=3634). Less data leads to more noise and larger standard errors.

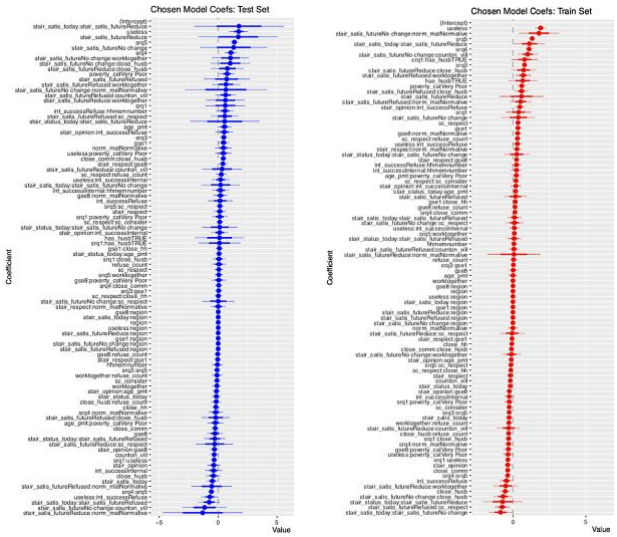


***Figure 11 & 12**: Magnitude of coefficients of chosen model on train set and test set*

iii. Bootstrapped confidence intervals

When we compute the bootstrapped test statistics and confidence intervals (R=2000), we see that the bootstrapped 95% confidence interval appear slightly wider on average than in the standard regression. Indeed, we see that the average standard error across all coefficients in the standard model was optimistically low (SE_{std} = 0.44) compared to the bootstrapped model (SE_{boot} = 0.48). This difference it not concerningly large, being about a 9% increase. The average bias, i.e. the difference between the test statistics of the standard and bootstrapped models, is relatively small as well (Mean Bias = 0.002).

	Original		Bootstrap-Normal		Bootstrap-Percentile	
	2.5%	**97.5%**	**2.5%**	**97.5%**	**2.5%**	**97.5%**
Intercept	11.17	15.43	11.24	15.43	11.15	15.29
srq1	-1.06	2.26	-1.24	2.51	-1.36	2.43
srq2	0.11	0.84	0.04	0.84	0.13	0.91
srq3	0.58	1.58	0.57	1.63	0.53	1.61
srq5	1.00	1.75	0.95	1.81	0.94	1.81
useless	0.94	2.53	0.82	2.61	0.89	2.64
stairstatus	-0.55	0.23	-0.53	0.23	-0.54	0.23

***Figure 13.** Original & bootstrapped normal & percentile 95% CI for the first 7 coefficients of chosen model*

	R	original	bootBias	bootSE
(Intercept)	2000	13.3000244	-0.0382845	1.0689797
srq1	2000	0.6007710	-0.0354065	0.9581033
srq3	2000	0.4757858	0.0363607	0.2040126
srq4	2000	1.0758250	-0.0228091	0.2697038
srq5	2000	1.3744851	-0.0076113	0.2199779
useless	2000	1.7368398	0.0235812	0.4558135
stair_status_today	2000	-0.1555673	-0.0019882	0.1943767
stair_satis_today	2000	-0.4941373	-0.0059582	0.2865634

***Figure 14.** Bootstrapped regression estimates for the first 7 coefficients of the chosen model*

➶ **Insight:** *Why are we observing this?* The reason that the bootstrapped confidence intervals are slightly larger is likely due to some of the assumptions of the OLS model not being met. If we test assumptions of

OLS using `regclass:: check_regressions` command, we see that the issue seems to be that the assumption of linearity was not met (see Figure 15). We see in plots of the fitted values vs the residuals of the chosen model and vs the outcome variable that there are slightly negative quadratic relationships rather than linear ones.

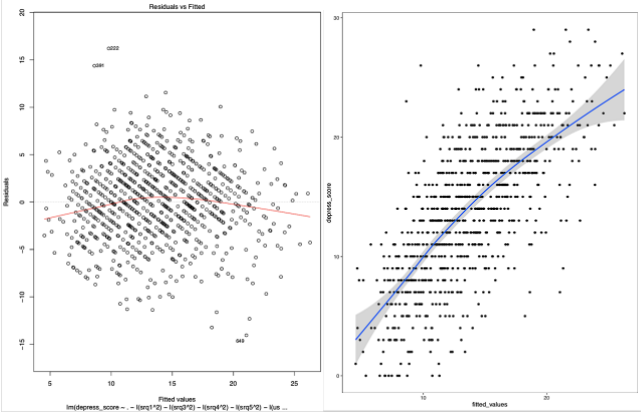


***Figure 15**: Non-linearity of standard chosen model, shown in plots of fitted values vs residuals and vs observed values of depress_score*

Other assumptions appear to be met: we cannot reject the null hypothesis that the residuals had equal variance and were normally distributed. Because assumptions were only slightly violated, the standard and bootstrapped models differ only slightly.

iv. Model Comparison: All vs Chosen

If we include all 48 possible covariates additively in a linear model in the test set, we see that the proportion of significant coefficients is higher (15.7%), compared to the chosen model with select covariates (10.6%). We see even variables that were consistently positively associated with the outcome variable in many different models and in pairwise correlations are no longer significant in the model with all possible covariates, including region, stair_status_today, stair_satis_today.

Insight: *Why are we observing this?* When we take out covariates and leave a subset, we push mass of unexplained variation in outcome into the remaining covariates. For this reason, the coefficients are more likely to be larger in a model that includes only those covariates found to have a significant association with the outcome. And conversely, they are likely to be smaller in a model in a model with more covariates because the mass of explained variation in Y spans a greater number of covariates. We thus see that which covariates and the number of covariates we include in the model changes the significance of particular coefficients and thus must be careful about interpreting the coefficients in multivariate regression.

v. Potential Analysis Problems

Multiple Hypothesis testing

Given that we have 95 coefficients in our chosen model, we are certainly at risk for have inflated p-values and Type I errors (i.e. selecting non-significant covariates as significant. With a 5% cut-off, we are possibly accepting 5% false positives. So, among our 95 terms included in the model, we should expect that almost 5 (=0.05x95) to be Type I errors. We can apply multiple testing corrections like Bonferroni and Benjamini-Hochberg to ensure that the probability of false positives is no higher than 5%. Using Bonferroni, we determine hypothesis where the p-value is lower than 0.05/p as significant (p is the # of hypothesis tests being carried). In our study, p = 95, therefore the Bonferroni correction enforces alpha to be 0.0005

(=0.05/95). If we apply a Bonferroni correction, we find that only 5 covariate coefficients are now significant (including the intercept), representing 5% of all p-values.

Given that the Bonferroni test is overly conservative, we also apply the Benjamini-Hochberg correction (BH). BH ensures that the rate of false discovery is no more than 0.05 by: ordering all p values from smallest to largest, finding the largest order position q_j such that $q_j \leq \alpha_j$ / p, and rejecting all hypotheses up to that position. When we apply BH, we find that $q_j = 5$ and we reject the same 5 covariate coefficients as we did using Bonferroni. As displayed in Figure 16, when not adjusted, p-values are evenly spread between 0 and 1 and 10.6% are < 0.05. In contrast, when we adjust the p-values for the number of hypotheses using BH, many of the p-values are pushed upwards towards 1, and now only 5.3% of p-values are less than 0.05 (which is what we want to see).

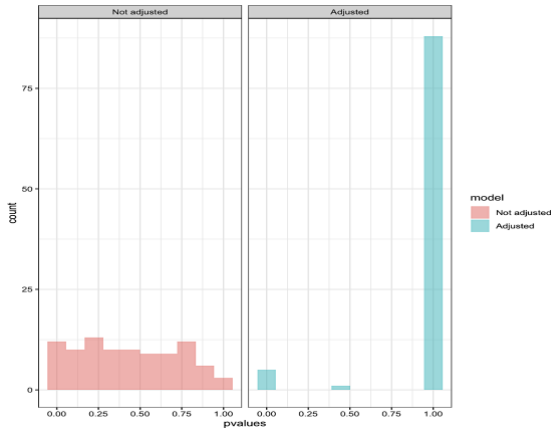


Figure 16. *Distributions of p values that are not adjusted vs. adjusted using Benjamini-Hochberg*

Post-selection inference: As we selected our covariates and the best model on the noise in the train set, we favorably biased our selection of p-values (see section VII.i). Therefore, on the test set, the coefficients found to be significant will change and overall fewer coefficients will be significant. We should be careful while interpreting significance in our final model.

Causality: Another potential issue is our causality interpretation of coefficients. For our causation analysis to be correct, we need all covariates to be independent (ideally, orthogonal) of other variables in order to ascertain exogeneity.

vi. Causality

➶ ***Pitfalls***: Correlation ≠ Causation. In comparing various models (full vs chosen model, test vs train data), we see that the significance of coefficients varies quite a lot, and thus we should not rely on a single multivariate regression model to make inferences about a particular covariate.

➶ ***Pitfalls***: Reverse causality. It is difficult to determine the direction of causality for psychological and health symptoms like happiness, low self-worth, and disturbances in sleep given they can be causes and consequences of depression. For more distal factors like poverty and region, we can most likely rule out that these are downstream consequences, but we do not know whether they are upstream causes or not.

➶ ***Pitfalls***: Confounders and omitted variable bias. For a more distal variable (to depression) like region, we can believe that there is likely no causal relationship but rather there is an association between region and depression due to confounders like ethnicity or access to education which are associated with both region and depression.

➶ ***Possible strategy forward***: Identifying candidates for further testing. We do see that a select few proximal variables are associated with depressive score across most models, including srq1-5, useless, close_comm, and close_husb, with a few of these holding up even under the Benjamini-Hochberg correction. We may think that such covariates which have consistently strong relationships with the outcome across many models and when controlling the false discovery rate are stronger *candidates* than others in having a causal relationship with depressive symptoms. Such candidates should then be assessed using causal inference methods to determine causality.

➶ Insight: *Possible strategy forward*: Existing literature. We will also draw on the literature to help us think about likely causal relationships. For instance, social isolation is a known contributor to depression; thus, we may believe that closeness (vs. isolation) to one's husband and community (close_husb, close_comm) could be a cause of depression in this population.

VIII. Discussion

i. Our model in the real world: Prediction or Inference?

We think that these models could be used in a couple ways. First, they could be used to target people at risk for moderate and severe levels of depression. Then additional, evidence-based psychosocial support and financial resources could be disbursed to these at-risk populations in order to improve wellbeing. Second, they could be used for inference in the sense that they demonstrate that certain covariates are so consistently correlated with depression in this cultural context (e.g. impairments in both physical and work functioning, estrangement from one's husband and one's community) that they may represent context-specific causal drivers or consequences of depression; as such, these variables should be included in assessments of risk for depression and recovery from depression. Experiments could be used to manipulate physical health by e.g. providing treatment or closeness to one's husband through e.g. couples therapy to assess causal relationships with depression.

ii. Hold up over time

These models would need to be updated every few years because this country is a somewhat rapidly changing society with an increase in access to information, different cultural influences, and resources as well as significant environmental threats that could all influence both the prevalence and manifestations of depression.

iii. Notes to other model users

We found in the test set that we violated the OLS assumption of linearity. This non-linearity is slightly surprising given that we log transformed variables that had non-linear relationships with the outcome and added relevant higher order terms to the model. However, we would advise future users to log transform the outcome variable as well. As for data cleaning, we would advise users to not ignore missing data in an available case analysis but would advise them to include a count measure of refusals, as we did, which captures an additional source of information on a respondent and likely their cultural orientation.

iv. Thoughts on data collection process

If we were to collect the data again:

First, we'd want to better understand why respondents refused to answer certain questions and then to adapt questions to be more culturally appropriate. This would have avoided some of the missing data issues we saw.

Second, we would have wanted to collect a measure of religiosity as we posit that it would help explain response patterns like refusals to future-oriented questions.

Third, we would have wanted to randomize the order of questions. It appears that variables cluster together and correlate more strongly with other variables that were proximal to each other in the questionnaire. Thus, to prevent order bias, we would have wanted to randomize the order of different constructs as well as particular questions within constructs. Relatedly, fourth, we would have wanted to have the same response scale for all questions to the extent possible.

Fifth, the coefficients of variation are quite large for many of the variables. We would have wanted to take multiple measurements of the same person and averaged them in order to reduce measurement error, increased the sample size further, or worked to improve clarity of the questions being asked in local languages to reduce any confusion.

Finally, we would have wanted to collect more variables. We would collect ethnicity and language to understand whether these factors moderate the relationship between certain variables and depression or are highly associated with depression. We know that certain ethnic groups have more restrictive gender norms, and we believe that closeness to one's husband and to one's community (which were highly predictive of depression) would vary by ethnic group. Relatedly, we would have wanted to collect additional variables on social cohesion, including tensions within households and communities, and the root causes of those tensions. If these variables were predictive of depression, this could help point to other types of economic and social interventions, in addition to psychological ones, that could improve wellbeing. We'd also have wanted to collect a more refined measure of poverty to understand the shape of the relationship between depression and poverty, which could help improve prediction.

v. What would we do differently?

If we were to attack the same dataset again:

First, we would take further measures to address non-linearity in order to meet OLS assumptions, beyond log transforming certain predictors and adding higher order terms. For example, we could have log transformed the outcome variable as well.

Second, in Part 1, we discussed our suspicion that our best model's CV error might be an overestimate of the test error because it includes many higher order terms and interactions and thus is at risk of being overfit to the training data. In Part 2 we confirmed that we have an issue with overfitting. If we were to go through the same analysis again, we would not overfit the model so strongly to the training data, in particular we would have included fewer interaction terms.