Anshul Dabas and Anndi Russell

Professor Cathy Durso

10 June 2020

An Exploration into World-View and Happiness Using Extreme Gradient Boosting

Executive Summary

Our research explores whether we can predict the happiness score of a country a person lives in based on their answer to a series of survey questions relating to their perception and opinion of certain aspects of their society. We used Extreme Gradient Boosting with the "xgboost" R package to help answer our question. The model we produced using xgboost was not a good fit. We were unable to predict the happiness score based on these survey responses with any accuracy.

Data source and definitions

Our research involved two data sets. The first is from the Pew Research Center's Global Attitudes and Trends survey conducted in Spring 2018. The second is data from the World Happiness Report. This report has been published over multiple years; we chose the year 2018 to most closely match with our Pew data.

The Pew survey was a survey given via phone to approximately 30,000 respondents in 27 different countries. The survey first asked questions relating to a person's view of the country they live in, then asked more questions about a respondent's opinion about global politics and relationships between countries. For this research, we were only concerned with the first set of questions. We've listed the questions below. For the sake of brevity we have not listed the response options here as response options often can be inferred intuitively from the question itself (see Appendix A for a full list of questions with their responses).

- *Thinking about our economic situation, how would you describe the current economic situation in (survey country) - is it very good, somewhat good, somewhat bad, or very bad?*
- *When children today in (survey country) grow up, do you think they will be better off, or worse off financially than their parents?*
- *How satisfied are you with the way democracy is working in our country – very satisfied, somewhat satisfied, not too satisfied, or not at all satisfied?*
- *Compared with 20 years ago, do you think the financial situation of average people in (survey country) is better, worse, or do you think there has been no change?*
- *Thinking about the ethnic, religious, and racial makeup of (survey country), over the past 20 years do you think (survey country) has become more diverse, less diverse, or do you think there has been no change?*
    - *Follow-up for those who did not respond "I don't know" or "refused": Do you think this is a good thing or a bad thing for (survey country)?*
- *Over the past 20 years, do you think equality between men and women in (survey country) has increased, decreased, or do you think there has been no change?*

    - *Do you think this is a good thing or a bad thing for (survey country)?*
- *Compared to 20 years ago, do you think religion has a more important role in (survey country), a less important role, or do you think there has been no change?*
    - *Do you think this is a good thing or a bad thing for (survey country)?*
- *Over the past 20 years, do you think family ties in (survey country) have become stronger, weaker, or do you think there has been no change?*
    - *Do you think this is a good thing or a bad thing for (survey country)?*

The World Happiness Report offers a wide variety of information relating to what makes people happy. For our analysis, we were concerned with the "happiness score" produced for each country. Researchers used data from the Gallup World Poll to compile this score. The Gallup World Poll asked respondents to think of a ladder, with the best life for them being a 10 and the worst being a 0. Respondents are then asked to rank their current life on that 0 to 10 scale. Researchers use survey weights to make this representative of the country, and calculate a mean "happiness score" for every country. This is the number we use in our research.

We combined these two datasets by pulling the "happiness score" for a country and attaching it to all respondents from that country. It's important to emphasize again that this is not a happiness score for each individual person; rather, it is a happiness score for the country in which they live.

Exploratory Data Analysis

Our data set has 30109 rows. This is sufficiently large for the method we are using. An evaluation of missing values revealed that the only missing values we have are in the follow-up questions; this is expected, as people who answered "Don't know" or "Refused" to the first question were not asked the follow-up. However, we did discover that 61 people in Mexico were mistakenly asked at least one follow-up question. To correct this, we simply replaced these responses with "NA" values, as was consistent with how the rest of the data was reported.

The number of responses was approximately equal across all countries, except for India which had about twice as many responses as other countries. We left this in, as it would not negatively affect our model and we wanted to have as much data to train as possible.

It was important that we saw a variety of response distributions between countries. If all countries tended to have very similar responses, this would not be an interesting investigation. To check this, we plotted the density of responses for all countries as seen in Figure 1.

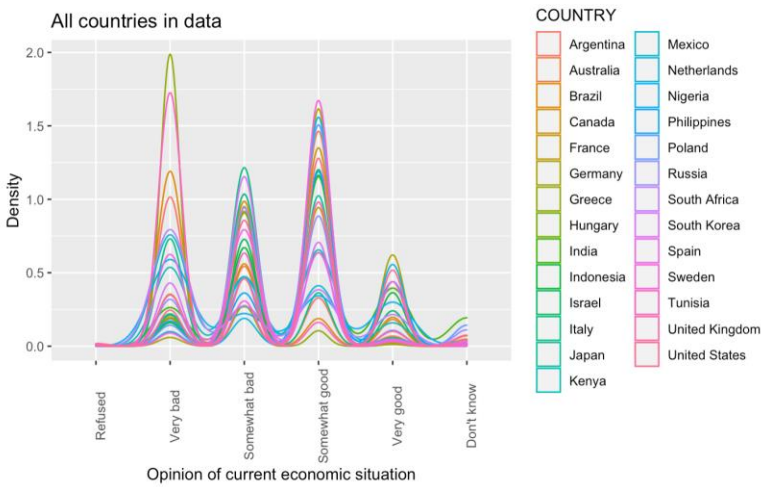


Figure 1: Density of responses to economic situation question by country.

Additionally, we selected one country that had a high happiness score (Netherlands), one with a medium happiness score (Poland), and one with a low happiness score (Tunisia), and compared their distributions of responses to these questions. The Netherlands would not be representative of all happy countries here (same with Poland and Tunisia), but it did allow us to further explore what differences we may find between countries of varying happiness scores. We did this same exploration for each of the 4 stand-alone questions in our data.

One important step we took with our data was to create a new column composed of responses to a question and its follow up (using the paste functionality). The follow-up question does not hold value for us on its own; if a person responded “Good thing”, the meaning can only be captured when combined with their response to the question before it. This resulted in 20 paired responses, with a sample density plot for the question regarding diversity in Figure 2.

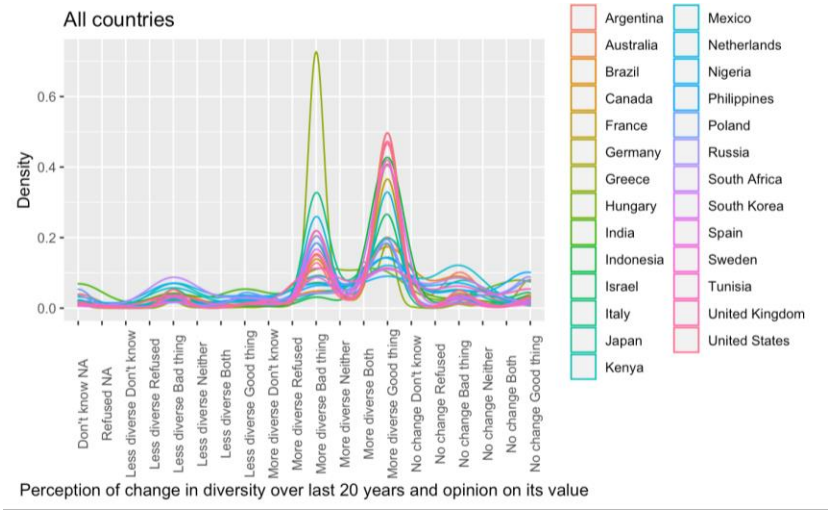


Figure 2: Density of responses to diversity question plus follow-up by country.

Again in this plot, as in Figure 1, we see a spread of responses between countries. We explored this same distribution between a high happiness, medium, and low happiness country as we did above to get a cursory glance at what differences our model may be able to pick up. We did this same exploration for all paired questions, in addition to exploring responses to just the first of the paired questions on their own.

Xgboost Method Explained

Now that we have explored the data, a reminder of our research question is important. We are going to use survey responses to predict the happiness score of the country a person lives in. Xgboost will build a model so we can predict the happiness score for future survey respondents, and we will also be able to explore which variables are the best predictors of our outcome.

Xgboost (extreme gradient boosting), uses decision trees to build a model for our data. The main principle of xgboost is that it fits a new model on the residuals from the previous model, and combines these models together. It does this continuously until the model performance is not improved by fitting more models (using root mean square error as an evaluation metric for this).

Xgboost will first build a naive model (F0), then calculate the residuals of that model. It will build a model predicting the residuals (h1) then combine them together into a new model (F1):

F1(x)=F0(x)+h1(x)

It then calculates the residuals on F1, builds a model (h2) to predict the residuals, then combines these together (F2):

F2(x)=F1(x)+h2(x)

This continues until there is no improvement on model performance by increasing the complexity, or until some stopping criterion for the number of trees built.

The full “TreeBoost” algorithm is defined as follows:

$$F_m(x) = F_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} \mathbf{1}_{R_{jm}}(x), \quad \gamma_{jm} = \arg\min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, F_{m-1}(x_i) + \gamma).$$

Citation: Wikipedia

Data Requirements for xgboost

Xgboost requires a particular data input format. To prepare for this, we first dropped any columns that were unnecessary (survey number, country, follow-up questions), then did a 70/30 train/test split. Since all of our predictors were categorical, we used one-hot-encoding with the sparse.model.matrix() function in R. This turns every survey response option into its own column, with a 1 indicating an answer, and a “.” otherwise. This sparse matrix only includes the predictor variables, approximately 120 of them (xgboost can handle large sets like this, and we

have 30,000 rows in our data so that number of variables is not a problem) (Fig. 3). The outcome variable (happiness score) is put into its own list. These are together put into xgboost's "DMatrix" (Fig. 4).

Create sparse matrix of just predictors

```{r}
sparse_matrix_train <- Matrix::sparse.model.matrix(happiness_score ~ ., data = dat_train_df,drop.unused.levels = FALSE)[,-1]
sparse_matrix_test <- Matrix::sparse.model.matrix(happiness_score ~ ., data = dat_test_df,drop.unused.levels = FALSE)[,-1]
```

XGBoost input

```{r}
dat_train <- xgb.DMatrix(data = sparse_matrix_train,label = output_train)
dat_test <- xgb.DMatrix(data = sparse_matrix_test,label = output_test)
```

Figures 3 and 4: Sparse matrix creation and xgboost DMatrix creation

Application of xgboost

We used the following libraries to apply xgboost modelling - library(xgboost), library(haven), library(car), library(SHAPforxgboost), library(Seurat).

First, we prepared a list of parameters to be passed while fitting the xgboost model on our training dataset. The outcome variable in our case was a continuous variable i.e the happiness score, so we performed a regression using xgboost. So, our booster = 'gbtree' and objective = 'reg::linear'. Further, we used subsample and colsample_bytree parameters to deal with overfitting for our model. Both of these parameters randomly sample the data and variables from the training set for different iterations, hence it will work on the overfitting aspect of the model. We used 'rmse' root mean square error as the evaluation metric (to evaluate regression performance).

```{r}
param_trees <- list(booster = "gbtree"
                , objective = "reg:linear"
                , subsample = 0.7
                , max_depth = 5
                , colsample_bytree = 0.7
                , eta = 0.037
                , eval_metric = 'rmse'
                , base_score = 0.012
                , min_child_weight = 100)
```

Figure 5: Parameters for xgboost

After setting up the required parameters appropriately, we performed cross-validation on our dataset by using xgboost internal cross-validation method xgb.cv with relevant parameters as follows:

```
Run xv
```{r}
target <- output_train
foldsCV <- createFolds(target, k=7, list=TRUE, returnTrain=FALSE)
xgb_cv <- xgb.cv(data=dat_train,
                 params=param_trees,
                nrounds=100,
                prediction=TRUE,
                maximize=FALSE,
                folds=foldsCV,
                gamma=0,
                early_stopping_rounds = 30,
                print_every_n = 5)
```
```

Figure 6: Cross validation for xgboost

After performing cross-validation on the training set, we got the best number of rounds as 100 that we further used as maximum number of trees (nrounds parameter) in our model.

Finally, we ran the xgboost model on our training dataset using xgb.fit with the appropriate parameters as follows:

```
nrounds <- xgb_cv$best_iteration
xgb.fit <- xgb.train(params = param_trees
                  , data = dat_train
                  , nrounds = nrounds
                  , verbose = 1
                  , print_every_n = 5
                 )
```

Figure 7: Fitting xgboost model

Additionally, we applied the predict method on the test dataset to evaluate the predictions using the above model.

Method Results and Analysis

We used the following two evaluation techniques to interpret or analyze the results produced by xgboost model - Importance matrix plot and SHAP (SHaply Additive exPlanation) values plot for the predictors or features.The importance matrix is a table with the first column including the names of all the features actually used in the boosted trees, the second column resulting in ‘importance’ values calculated with importance metrics as weight or gain(default) or cover. We used gain metric as it is the most relevant metric to interpret the relative importance of each feature.It is the improvement in accuracy brought by a feature to the branches it is on. As we had ~120 features in the importance plot, so for clarity, we subsetted the top 10 features as follows:

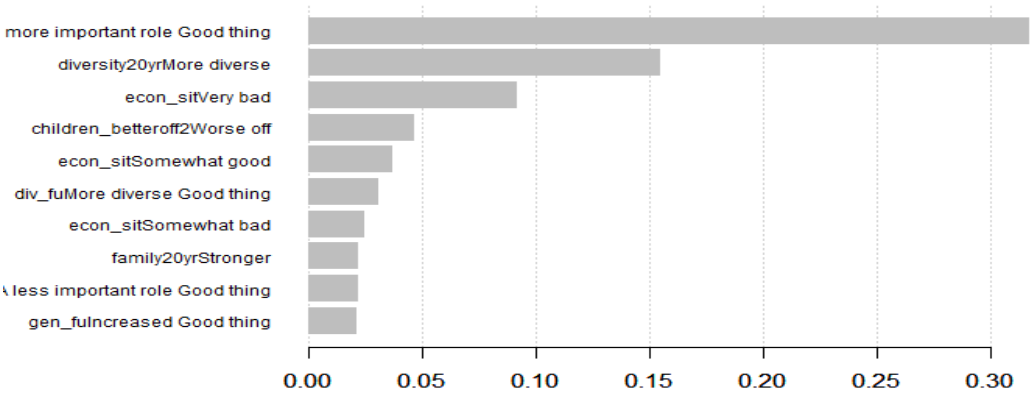
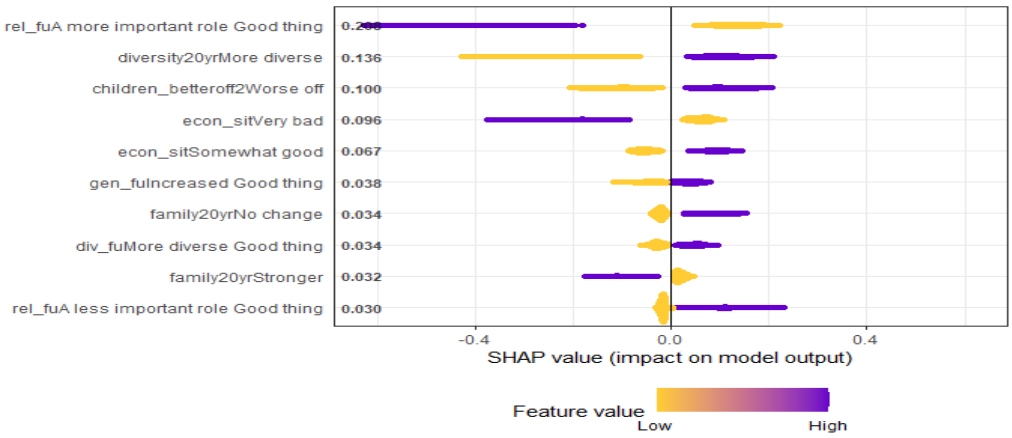


Figure 8: Importance matrix for top 10 predictors

According to the above graph, only three features(religion role importance is a good thing, diversity increase, economic situation being bad) had their importance values > 0.05, however to be considered as a good predictor the importance value should be close to 1 that is not reflected in our case.

Next, we used SHAP (SHaply Additive exPlanation) values, similar to the importance matrix; it is a different way to calculate the most important predictors. It calculates the importance of a feature by comparing what a model predicts with and without the feature. Since the order in which a model sees features can affect its predictions, this is done in every possible order, so that the features are fairly compare



d.

Figure 9: SHAP values for top 10 predictors

According to the SHAP values plot, we got top features as the major contributors in the happiness score prediction, the same as the importance matrix plot. Also, the highest shap value is ~0.2, which is pretty low to consider it as a good predictor. Additionally, SHAP values indicate how much is the change in log-odds and correlation of a predictor with the outcome

variable. For instance, religious importance has a high and negative SHAP value that indicates it to be negatively correlated with the happiness score.

Subsequently, to check the accuracy of our model we evaluated r-squared value as 0.28, that implied very low accuracy henceforth not a well-performing model. Also, we did not look into the happiness score variation within a country. The low r-squared value may have indicated variability within a country however it seems to be the future scope of work for our project to evaluate the variability.

Therefore, considering all the above interpretations and evaluations we concluded that religious importance increase being good, diversity increase, very bad economic situation, gender equality increase being good and children's future finance situation in comparison with parents are the most influential contributor features in the model to predict happiness score for a country by using people's survey. Although, the model accuracy was pretty low hence we cannot consider any of the model-suggested predictors as good predictors for the outcome variable.

On the whole, societal views and opinions of people cannot be considered to be good predictors for the happiness score of that particular country, which is an answer to our research question.

Citations:

https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/

https://en.wikipedia.org/wiki/Gradient_boosting

https://blog.datascienceheroes.com/how-to-interpret-shap-values-in-r/

https://www.analyticsvidhya.com/blog/2016/01/xgboost-algorithm-easy-steps/

https://www.youtube.com/watch?v=3CC4N4z3GJc

Data:
https://www.pewresearch.org/global/2019/04/22/a-changing-world-global-views-on-diversity-gender-equality-family-life-and-the-importance-of-religion/

https://worldhappiness.report/ed/2018/

Appendix A:

Survey questions and responses

Full list of questions with responses and variable names ("DO NOT READ") indicates that the survey reader did not read those options aloud:

Thinking about our economic situation, how would you describe the current economic situation in (survey country) – is it very good, somewhat good, somewhat bad, or very bad?

1 Very good
2 Somewhat good
3 Somewhat bad
4 Very bad
8 Don't know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: econ_sit

When children today in (survey country) grow up, do you think they will be better off, or worse off financially than their parents?

1 Better off
2 Worse off
3 Same (DO NOT READ)
8 Don't know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: children_betteroff2

How satisfied are you with the way democracy is working in our country – very satisfied, somewhat satisfied, not too satisfied, or not at all satisfied?

1 Very satisfied
2 Somewhat satisfied
3 Not too satisfied
4 Not at all satisfied
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: satisfied_democracy

Compared with 20 years ago, do you think the financial situation of average people in (survey country) is better, worse, or do you think there has been no change?

1 Better
2 Worse
3 No change
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: financial20yr

Thinking about the ethnic, religious, and racial makeup of (survey country), over the past 20 years do you think (survey country) has become more diverse, less diverse, or do you think there has been no change?

1 More diverse
2 Less diverse
3 No change
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: diversity20yr

Follow-up (if not 8/9 response): Do you think this is a good thing or a bad thing for (survey country)?

1 Good thing
2 Bad thing
3 Both (DO NOT READ)
4 Neither (DO NOT READ)
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: diversity20yr_fu

Over the past 20 years, do you think equality between men and women in (survey country) has increased, decreased, or do you think there has been no change?

1 Increased
2 Decreased
3 No change
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: gender20yr

Follow-up (if not 8/9 response): Do you think this is a good thing or a bad thing for (survey country)?

1 Good thing
2 Bad thing
3 Both (DO NOT READ)
4 Neither (DO NOT READ)
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: gender20yr_fu

Compared to 20 years ago, do you think religion has a more important role in (survey country), a less important role, or do you think there has been no change?

1 A more important role
2 A less important role
3 No change
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: religion20yr

Follow-up (if not 8/9 response): Do you think this is a good thing or a bad thing for (survey country)?

1 Good thing
2 Bad thing
3 Both (DO NOT READ)
4 Neither (DO NOT READ)
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: religion20yr_fu

Over the past 20 years, do you think family ties in (survey country) have become stronger, weaker, or do you think there has been no change?

1 Stronger
2 Weaker
3 No change
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)
Ref: family20yr

Follow-up (if not 8/9 response): Do you think this is a good thing or a bad thing for (survey country)?

1 Good thing
2 Bad thing
3 Both (DO NOT READ)
4 Neither (DO NOT READ)
8 Don’t know (DO NOT READ)
9 Refused (DO NOT READ)

Ref: family20yr_fu

Appendix B:

Complete R code

```{r}
library(foreign)
library(caret)
library(ggplot2)
library(dplyr)
library(xgboost)
library(devtools)
library(usethis)
library(readr)
library(stringr)
library(caret)
library(tidyverse)
library(haven)
library(car)
library(SHAPforxgboost)
library(Seurat)
source("shap_Func.R")
```

Code to change original sav file to csv:

```
write.table(read.spss("globalattitudes.sav"), file="globalattitudes.csv",
quote = FALSE, sep = ",")
```

Import data

```{r}
dat_global <- read.csv(file = 'globalattitudes.csv')
happiness_dat <- read.csv(file = 'WorldHappiness2018_data.csv')
```

Create table of happiness score and ranks

```{r}

countries_dat_global<-unique(dat_global$COUNTRY)

happiness_score<-c()

rank<-c()

for (country in countries_dat_global){

  x<-happiness_dat$Score[which(happiness_dat$Country==country)]

  r<-happiness_dat$Rank[which(happiness_dat$Country==country)]

  happiness_score<-c(happiness_score,x)

  rank<-c(rank,r)

}


happiness_countries_score<-cbind(countries_dat_global,happiness_score,rank)

colnames(happiness_countries_score)<-c("country","score","rank")

happiness_countries_score<-data.frame(happiness_countries_score)

```


Add happiness score to dat_global based on country. Commented portions are to add in the country rank and grouping. We're not using this in our current project.

```{r}

dat_global$happiness_score <-
happiness_countries_score$score[match(dat_global$COUNTRY,
happiness_countries_score$country)]

#dat_global$happiness_rank <-
happiness_countries_score$rank[match(dat_global$COUNTRY,
happiness_countries_score$country)]

#dat_global$happiness_cat <- cut(as.numeric(dat_global$happiness_rank), c(-
Inf,30,70,Inf), c("high", "medium", "low"))

```

Happiness score should be numeric

```{r}
```

```
dat_global$happiness_score<-
as.numeric(as.character(dat_global$happiness_score))
```

Plot of the variation in happiness score.

```{r}
ggplot(data=happiness_countries_score, aes(x=country, y=score))+geom_point()+
theme(axis.text.x = element_text(angle = 90))+ggtitle("Happiness Score by
Country")+xlab("Country")+ylab("Happiness Score")
```

Drop "Survey" column

```{r}
dat_global<-dat_global[-2]
```

Count of respondents in each country

```{r}
table(dat_global$COUNTRY)
```

Number of respondents

```{r}
nrow(dat_global)
```

Remove the string "DO NOT READ" (this indicated that the survey reader would not read these responses). Not necessary for our analysis; no value added.

```{r}
dat_global <- data.frame(lapply(dat_global, function(x) {
                gsub("\\ \\(DO NOT READ\\)", "", x)
          }))
```

The apostrophe character is odd. Let's replace it with a normal apostrophe everywhere it shows up.

```{r}

dat_global <- data.frame(lapply(dat_global, function(x) {

                gsub("\\’", "\\'", x)

          }))

```

Happiness score was changed to character in those manipulations. Back to numeric.

```{r}

dat_global$happiness_score<-
as.numeric(as.character(dat_global$happiness_score))

unique(dat_global$happiness_score)

```

Examine where we find missing values. Only in the follow-up questions, as expected.

```{r}

sapply(dat_global, function(x) sum(is.na(x)))

```

Create follow-up indicator variable. 1 if they should be asked the follow-up and 0 if not. We'll use this to verify that the correct number were asked the follow-up.

```{r}

div <- ifelse(dat_global$diversity20yr %in% c("Refused","Don't know"),0,1)

dat_global<-tibble::add_column(dat_global, diversity_fu_indicator = div,
.after = "diversity20yr")

gen <- ifelse(dat_global$gender20yr %in% c("Refused","Don't know"),0,1)

dat_global<-tibble::add_column(dat_global, gender_fu_indicator = gen, .after
= "gender20yr")

rel <- ifelse(dat_global$religion20yr %in% c("Refused","Don't know"),0,1)

dat_global<-tibble::add_column(dat_global, religion_fu_indicator = rel,
.after = "religion20yr")

```

```
fam <- ifelse(dat_global$family20yr %in% c("Refused","Don't know"),0,1)

dat_global<-tibble::add_column(dat_global, family_fu_indicator = fam, .after
= "family20yr")
```

Verify that the count of "Don't know" and "Refused" in the frist question matches the count of NA in the follow-up column:

```{r}
sum(dat_global$diversity_fu_indicator==0)

sum(dat_global$family_fu_indicator==0)

sum(dat_global$gender_fu_indicator==0)

sum(dat_global$religion_fu_indicator==0)
```

The above values do not match the NA counts from the earlier table. We have answers to follow-up questions when we shouldn't.

Below we verify that everyone who should be asked a follow-up was indeed asked.

```{r}
sum(dat_global$diversity_fu_indicator[which(is.na(dat_global$diversity20yr_fu
))])

sum(dat_global$family_fu_indicator[which(is.na(dat_global$family20yr_fu))])

sum(dat_global$gender_fu_indicator[which(is.na(dat_global$gender20yr_fu))])

sum(dat_global$religion_fu_indicator[which(is.na(dat_global$religion20yr_fu))
])
```

We will first collect the IDs and countries for which there was a follow-up question discrepancy for the diversity question. This happened 24 times.

```{r}
nrow(filter(dat_global, (diversity_fu_indicator==0 &
!is.na(diversity20yr_fu))))

errorIDs_div<-as.character(filter(dat_global, (diversity_fu_indicator==0 &
!is.na(diversity20yr_fu)))$ID)
```

```
errorcountries<-as.character(filter(dat_global, (diversity_fu_indicator==0 &
!is.na(diversity20yr_fu)))$COUNTRY)
```

Same for gender (8 issues), religion (28 issues), and family (11 issues).

```{r}
nrow(filter(dat_global, (gender_fu_indicator==0 & !is.na(gender20yr_fu))))

errorIDs_gen<-as.character(filter(dat_global, (gender_fu_indicator==0 &
!is.na(gender20yr_fu)))$ID)

errorcountries<-c(errorcountries,as.character(filter(dat_global,
(gender_fu_indicator==0 & !is.na(gender20yr_fu)))$COUNTRY))

nrow(filter(dat_global, (religion_fu_indicator==0 &
!is.na(religion20yr_fu))))

errorIDs_rel<-as.character(filter(dat_global, (religion_fu_indicator==0 &
!is.na(religion20yr_fu)))$ID)

errorcountries<-c(errorcountries, as.character(filter(dat_global,
(religion_fu_indicator==0 & !is.na(religion20yr_fu)))$COUNTRY))

nrow(filter(dat_global, (family_fu_indicator==0 & !is.na(family20yr_fu))))

errorIDs_fam<-as.character(filter(dat_global, (family_fu_indicator==0 &
!is.na(family20yr_fu)))$ID)

errorcountries<-c(errorcountries,as.character(filter(dat_global,
(family_fu_indicator==0 & !is.na(family20yr_fu)))$COUNTRY))
```

All errors were in Mexico

```{r}
errorcountries<-unique(errorcountries)

errorcountries
```

Below, we will insert an "NA" into the follow-up question. This should have been the original value since the follow-up question should not have been asked.

```{r}
errorIDs_div<-as.numeric(errorIDs_div)
```

```
dat_global$diversity20yr_fu[dat_global$ID %in% errorIDs_div]<-NA

errorIDs_rel<-as.numeric(errorIDs_rel)

dat_global$religion20yr_fu[dat_global$ID %in% errorIDs_rel]<-NA

errorIDs_gen<-as.numeric(errorIDs_gen)

dat_global$gender20yr_fu[dat_global$ID %in% errorIDs_gen]<-NA

errorIDs_fam<-as.numeric(errorIDs_fam)

dat_global$family20yr_fu[dat_global$ID %in% errorIDs_fam]<-NA
```

Now we can see that the counts of NA in the follow-up and the counts of 0 in the indicator column match as expected.

```{r}
sum(dat_global$diversity_fu_indicator==0)

sum(dat_global$family_fu_indicator==0)

sum(dat_global$gender_fu_indicator==0)

sum(dat_global$religion_fu_indicator==0)

sapply(dat_global, function(x) sum(is.na(x)))
```

For the 4 paired questions, combine a question with its follow-up into new column:

```{r}
dat_global$div_fu <-
paste(dat_global$diversity20yr,dat_global$diversity20yr_fu)

dat_global$rel_fu <-
paste(dat_global$religion20yr,dat_global$religion20yr_fu)

dat_global$gen_fu <- paste(dat_global$gender20yr,dat_global$gender20yr_fu)

dat_global$fam_fu <- paste(dat_global$family20yr,dat_global$family20yr_fu)
```

We expect 20 possible outcomes for this paired column. Verified below.

```{r}
length(unique(dat_global$div_fu))
```

```
length(unique(dat_global$fam_fu))

length(unique(dat_global$gen_fu))

length(unique(dat_global$rel_fu))
```

Data exploration

Economic situation:

Thinking about our economic situation, how would you describe the current economic situation in (survey country) – is it very good, somewhat good, somewhat bad, or very bad?

Very good,Somewhat good,Somewhat bad,Very bad,Don’t know,Refused

Ref: econ_sit

```{r}
ggplot(data=dat_global, aes(x=econ_sit,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries in data")+xlab("Opinion of
current economic situation")

ggplot(data=dat_global, aes(x=econ_sit,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries in data")+xlab("Opinion of
current economic situation")+scale_x_discrete(limits=c("Refused", "Very bad",
"Somewhat bad","Somewhat good","Very good", "Don't know"))

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',],
aes(x=econ_sit)) +geom_histogram(stat='count',
fill='black')+theme(axis.text.x = element_text(angle = 90))+ggtitle("United
States")+xlab(NULL)+ylab("Count")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',],
aes(x=econ_sit)) +geom_histogram(stat='count', fill='blue')+theme(axis.text.x
= element_text(angle = 90))+ggtitle("Netherlands")+xlab(NULL)+ylab(NULL)

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=econ_sit))
+geom_histogram(stat='count', fill='purple')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Poland")+xlab("Opinion of current economic
situation")+ylab("Count")

d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=econ_sit))
+geom_histogram(stat='count', fill='red')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Tunisia")+xlab("Opinion of current
economic situation")+ylab(NULL)
```

```
a+b+c+d
```

```

Chidren better off

When children today in (survey country) grow up, do you think they will be better off, or worse off financially than their parents?

Better off,Worse off,Same,Don’t know,Refused

Ref: children_betteroff2

```{r}

ggplot(data=dat_global, aes(x=children_betteroff2,group=COUNTRY, color=COUNTRY)) +geom_density(position='dodge')+theme(axis.text.x = element_text(angle = 90))+ylab("Density")+ggtitle("All countries in data")+xlab("Opinion of children's prospective futures as compared to their parents")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',], aes(x=children_betteroff2)) +geom_histogram(stat='count', fill='black')+theme(axis.text.x = element_text(angle = 90))+ggtitle("United States")+xlab(NULL)+ylab("Count")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',], aes(x=children_betteroff2)) +geom_histogram(stat='count', fill='blue')+theme(axis.text.x = element_text(angle = 90))+ggtitle("Netherlands")+xlab(NULL)+ylab(NULL)

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=children_betteroff2)) +geom_histogram(stat='count', fill='purple')+theme(axis.text.x = element_text(angle = 90))+ggtitle("Poland")+xlab("Opinion of children's prospective futures")+ylab("Count")

d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=children_betteroff2)) +geom_histogram(stat='count', fill='red')+theme(axis.text.x = element_text(angle = 90))+ggtitle("Tunisia")+xlab("Opinion of children's prospective futures")+ylab(NULL)


a+b+c+d

```

```

Satisfied with democracy

How satisfied are you with the way democracy is working in our country – very satisfied, somewhat satisfied, not too satisfied, or not at all satisfied?

Very satisfied,Somewhat satisfied,Not too satisfied,Not at all satisfied,Don’t know,Refused

Ref: satisfied_democracy

```{r}

ggplot(data=dat_global, aes(x=satisfied_democracy,group=COUNTRY, color=COUNTRY)) +geom_density(position='dodge')+theme(axis.text.x = element_text(angle = 90))+xlab("Satisfaction with state of democracy")+ylab("Density")+ggtitle("All countries in data")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',], aes(x=satisfied_democracy)) +geom_histogram(stat='count', fill='black')+theme(axis.text.x = element_text(angle = 90))+ggtitle("United States")+xlab(NULL)+ylab("Count")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',], aes(x=satisfied_democracy)) +geom_histogram(stat='count', fill='blue')+theme(axis.text.x = element_text(angle = 90))+ggtitle("Netherlands")+xlab(NULL)+ylab(NULL)

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=satisfied_democracy)) +geom_histogram(stat='count', fill='purple')+theme(axis.text.x = element_text(angle = 90))+ggtitle("Poland")+xlab("Satisfaction with state of democracy")+ylab("Count")

d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=satisfied_democracy)) +geom_histogram(stat='count', fill='red')+theme(axis.text.x = element_text(angle = 90))+ggtitle("Tunisia")+xlab("Satisfaction with state of democracy")+ylab(NULL)

a+b+c+d

```

Financial

Compared with 20 years ago, do you think the financial situation of average people in (survey country) is better, worse, or do you think there has been no change?

Better,Worse,No change,Don’t know,Refused

Ref: financial20yr

```{r}

ggplot(data=dat_global, aes(x=satisfied_democracy,group=COUNTRY,
color=COUNTRY)) +geom_density(position='dodge')+theme(axis.text.x =
element_text(angle = 90))+ylab("Density")+ggtitle("All countries in
data")+xlab("Opinion of financial situation of average person compared to 20
years ago")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',],
aes(x=financial20yr)) +geom_histogram(stat='count',fill='black')+
theme(axis.text.x = element_text(angle = 90))+ggtitle("United
States")+xlab(NULL)+ylab("Count")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',],
aes(x=financial20yr))
+geom_histogram(stat='count',fill='blue')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Netherlands")+xlab(NULL)+ylab(NULL)

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',],
aes(x=financial20yr))
+geom_histogram(stat='count',fill='purple')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Poland")+xlab("Opinion of financial
situation compared to 20 years ago")+ylab("Count")

d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',],
aes(x=financial20yr))
+geom_histogram(stat='count',fill='red')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Tunisia")+xlab("Opinion of financial
compared to 20 years ago")+ylab(NULL)



a+b+c+d

```

Diversity

Thinking about the ethnic, religious, and racial makeup of (survey country), over the past 20 years do you think (survey country) has become more diverse, less diverse, or do you think there has been no change?

More diverse,Less diverse,No change,Don’t know,Refused

Ref: diversity20yr

Follow-up:Do you think this is a good thing or a bad thing for (survey country)?

Good thing,Bad thing,Both,Neither,Don’t know,Refused

Ref: diversity20yr_fu

```{r}

ggplot(data=dat_global, aes(x=diversity20yr))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All
countries")+xlab("Perception of change in diversity over last 20 years")

ggplot(data=dat_global, aes(x=div_fu))
+geom_histogram(stat='count')+theme(axis.text.x = element_text(angle =
90))+ylab("Count")+ggtitle("All countries")+xlab("Perception of change in
diversity over last 20 years and opinion on its value")+theme(axis.text.x =
element_text(angle = 90))

ggplot(data=dat_global, aes(x=diversity20yr,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
diversity over last 20 years")

ggplot(data=dat_global, aes(x=div_fu,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
diversity over last 20 years and opinion on its value")

ggplot(data=dat_global[dat_global$diversity20yr=='More diverse',],
aes(x=diversity20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents
who said 'More Diverse'")+xlab("Opinion on diversity
increasing")+ylab("Count")

ggplot(data=dat_global[dat_global$diversity20yr=='Less diverse',],
aes(x=diversity20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents
who said 'Less Diverse'")+xlab("Opinion on diversity
decreasing")+ylab("Count")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',],
aes(x=div_fu)) +geom_histogram(stat='count',fill='black')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("United
States")+ylab("Count")+xlab("Perception of change in diversity over last 20
years and opinion on its value")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',], aes(x=div_fu))
+geom_histogram(stat='count',fill='blue')+theme(axis.text.x =
element_text(angle =
90))+ggtitle("Netherlands")+ylab("Count")+xlab("Perception of change in
diversity over last 20 years and opinion on its value")

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=div_fu))
+geom_histogram(stat='count',fill='purple')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Poland")+ylab("Count")+xlab("Perception of
change in diversity over last 20 years and opinion on its value")
```

```
d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=div_fu))
+geom_histogram(stat='count',fill='red')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Tunisia")+ylab("Count")+xlab("Perception
of change in diversity over last 20 years and opinion on its value")

a+b+c+d
```

Gender

Over the past 20 years, do you think equality between men and women in (survey country) has increased, decreased, or do you think there has been no change?

Increased,Decreased,No change,Don’t know,Refused

Ref: gender20yr

Do you think this is a good thing or a bad thing for (survey country)?

Good thing,Bad thing ,Both,Neither,Don’t know,Refused

Ref: gender20yr_fu

```{r}
ggplot(data=dat_global, aes(x=gender20yr))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All
countries")+xlab("Perception of change in gender equality over last 20
years")

ggplot(data=dat_global, aes(x=gen_fu))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All
countries")+xlab("Perception of change in gender equality over last 20 years
and opinion on its value")+theme(axis.text.x = element_text(angle = 90))

ggplot(data=dat_global, aes(x=gender20yr,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
gender equality over last 20 years")

ggplot(data=dat_global, aes(x=gen_fu,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
gender equality over last 20 years and opinion on its value")

ggplot(data=dat_global[dat_global$gender20yr=='Increased',],
aes(x=gender20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents who
said 'Increased'")+xlab("Opinion on gender equality
increasing")+ylab("Count")
```

```
ggplot(data=dat_global[dat_global$gender20yr=='Decreased',],
aes(x=gender20yr_fu)) +geom_histogram(stat='count')+ ggtitle("Respondents who
said 'Decreased'")+xlab("Opinion on gender equality
decreasing")+ylab("Count")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',],
aes(x=gen_fu)) +geom_histogram(stat='count',fill='black')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("United
States")+ylab("Count")+xlab("Perception of change in gender equality over
last 20 years and opinion on its value")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',], aes(x=gen_fu))
+geom_histogram(stat='count',fill='blue')+theme(axis.text.x =
element_text(angle =
90))+ggtitle("Netherlands")+ylab("Count")+xlab("Perception of change in
gender equality over last 20 years and opinion on its value")

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=gen_fu))
+geom_histogram(stat='count',fill='purple')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Poland")+ylab("Count")+xlab("Perception of
change in gender equality over last 20 years and opinion on its value")

d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=gen_fu))
+geom_histogram(stat='count',fill='red')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Tunisia")+ylab("Count")+xlab("Perception
of change in gender equality over last 20 years and opinion on its value")

a+b+c+d

```

Religion

Compared to 20 years ago, do you think religion has a more important role in (survey country), a less important role, or do you think there has been no change?

A more important role,A less important role,No change,Don’t know,Refused

Ref: religion20yr

Do you think this is a good thing or a bad thing for (survey country)?

Good thing,Bad thing ,Both,Neither,Don’t know,Refused

Ref: religion20yr_fu

```{r}

ggplot(data=dat_global, aes(x=religion20yr))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All


```

```
countries")+xlab("Perception of change in importance of religion over last 20
years")

ggplot(data=dat_global, aes(x=rel_fu))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All
countries")+xlab("Perception of change in importance of religion over last 20
years and opinion on its value")+theme(axis.text.x = element_text(angle =
90))

ggplot(data=dat_global, aes(x=religion20yr,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
importance of religion over last 20 years")

ggplot(data=dat_global, aes(x=rel_fu,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
importance of religion over last 20 years and opinion on its value")




ggplot(data=dat_global[dat_global$religion20yr=='A more important role',],
aes(x=religion20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents
who said 'A more important role'")+xlab("Opinion on religion playing a more
important role")+ylab("Count")

ggplot(data=dat_global[dat_global$religion20yr=='A less important role',],
aes(x=religion20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents
who said 'A less important role'")+xlab("Opinion on religion playing a less
important role")+ylab("Count")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',],
aes(x=rel_fu)) +geom_histogram(stat='count',fill='black')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("United
States")+ylab("Count")+xlab("Perception of change in importance of religion
over last 20 years and opinion on its value")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',], aes(x=rel_fu))
+geom_histogram(stat='count',fill='blue')+theme(axis.text.x =
element_text(angle =
90))+ggtitle("Netherlands")+ylab("Count")+xlab("Perception of change in
importance of religion over last 20 years and opinion on its value")

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=rel_fu))
+geom_histogram(stat='count',fill='purple')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Poland")+ylab("Count")+xlab("Perception of
change in importance of religion over last 20 years and opinion on its
value")
```

```
d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=rel_fu))
+geom_histogram(stat='count',fill='red')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Tunisia")+ylab("Count")+xlab("Perception
of change in importance of religion over last 20 years and opinion on its
value")

a+b+c+d
```

Family

Over the past 20 years, do you think family ties in (survey country) have become stronger, weaker, or do you think there has been no change?

Stronger,Weaker,No change,Don’t know,Refused

Ref: family20yr

Do you think this is a good thing or a bad thing for (survey country)?

Good thing,Bad thing ,Both,Neither,Don’t know,Refused

Ref: family20yr_fu

```{r}
ggplot(data=dat_global, aes(x=family20yr))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All
countries")+xlab("Perception of change in strength of family ties over last
20 years")+theme(axis.text.x = element_text(angle = 90))

ggplot(data=dat_global, aes(x=fam_fu))
+geom_histogram(stat='count')+ylab("Count")+ggtitle("All
countries")+xlab("Perception of change in strength of family ties over last
20 years and opinion on its value")+theme(axis.text.x = element_text(angle =
90))

ggplot(data=dat_global, aes(x=family20yr,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
strength of family ties over last 20 years")

ggplot(data=dat_global, aes(x=fam_fu,group=COUNTRY, color=COUNTRY))
+geom_density(position='dodge')+theme(axis.text.x = element_text(angle =
90))+ylab("Density")+ggtitle("All countries")+xlab("Perception of change in
strength of family ties over last 20 years and opinion on its value")

ggplot(data=dat_global[dat_global$family20yr=='Stronger',],
aes(x=family20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents who
said 'Stronger'")+xlab("Opinion on family ties being stronger")+ylab("Count")
```

```
ggplot(data=dat_global[dat_global$family20yr=='Weaker',],
aes(x=family20yr_fu)) +geom_histogram(stat='count')+ggtitle("Respondents who
said 'Weaker'")+xlab("Opinion on family ties being weaker")+ylab("Count")

a<-ggplot(data=dat_global[dat_global$COUNTRY=='United States',],
aes(x=fam_fu)) +geom_histogram(stat='count',fill='black')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("United
States")+ylab("Count")+xlab("Perception of change in strength of family ties
over last 20 years and opinion on its value")

b<-ggplot(data=dat_global[dat_global$COUNTRY=='Netherlands',], aes(x=fam_fu))
+geom_histogram(stat='count',fill='blue')+theme(axis.text.x =
element_text(angle =
90))+ggtitle("Netherlands")+ylab("Count")+xlab("Perception of change in
strength of family ties over last 20 years and opinion on its value")

c<-ggplot(data=dat_global[dat_global$COUNTRY=='Poland',], aes(x=fam_fu))
+geom_histogram(stat='count',fill='purple')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Poland")+ylab("Count")+xlab("Perception of
change in strength of family ties over last 20 years and opinion on its
value")

d<-ggplot(data=dat_global[dat_global$COUNTRY=='Tunisia',], aes(x=fam_fu))
+geom_histogram(stat='count',fill='red')+theme(axis.text.x =
element_text(angle = 90))+ggtitle("Tunisia")+ylab("Count")+xlab("Perception
of change in strength of family ties over last 20 years and opinion on its
value")

a+b+c+d

```

Now that we've explored out data, we will prepare the data frame to start our analysis. This starts by dropping a couple columns that are no longer necessary.

Drop ID and COUNTRY and Drop follow-up indicators

```{r}

dat_global<-dat_global[-c(1:2)]

drops <-
c("diversity_fu_indicator","gender_fu_indicator","family_fu_indicator","relig
ion_fu_indicator")

dat_global<-dat_global[ , !(names(dat_global) %in% drops)]

```

At this point, the dataframe is complete with the categorical varibles.

We have 4 questions that stand-alone:

econ_sit,children_betteroff2,satisfied_democracy,financial20yr

As well as 4 sets of questions with follow-ups:

diversity20yr,diversity20yr_fu,gender20yr,gender20yr_fu,religion20yr,religion20yr_fu,family20yr,family20yr_fu

Plus our outcome of interest:

happiness_score

Drop follow-up columns (they don't hold value for us on their own)

```{r}

drops <-
c("diversity20yr_fu","gender20yr_fu","family20yr_fu","religion20yr_fu")

dat_global<-dat_global[ , !(names(dat_global) %in% drops)]

```

Verify that our dataframe is correct up until this point

```{r}

head(dat_global,2)

```

Move happiness score to beginning

```{r}

dat_global<-
dat_global[,c(which(colnames(dat_global)=="happiness_score"),which(colnames(dat_global)!="happiness_score"))]

```

Train/test split

```{r}

set.seed(12345)

train_index <- sample.int(n = nrow(dat_global), size =
floor(.7*nrow(dat_global)), replace = F)

dat_train_df <- dat_global[train_index, ]

dat_test_df  <- dat_global[-train_index, ]

```

Characters to factors

```{r}

dat_global[sapply(dat_global, is.character)] <-
lapply(dat_global[sapply(dat_global, is.character)],

                                       as.factor)

str(dat_global)

```

Create sparse matrix of just predictors

```{r}

sparse_matrix_train <- Matrix::sparse.model.matrix(happiness_score ~ ., data
= dat_train_df,drop.unused.levels = FALSE)[,-1]

sparse_matrix_test <- Matrix::sparse.model.matrix(happiness_score ~ ., data =
dat_test_df,drop.unused.levels = FALSE)[,-1]

```

Check column names

```{r}

all.equal(colnames(sparse_matrix_train), colnames(sparse_matrix_test))

```

Find the error

```{r}

setdiff(colnames(sparse_matrix_train), colnames(sparse_matrix_test))

```

Drop that column from train

```{r}

nm <- c("gen_fuDecreased Refused")

sparse_matrix_train<-sparse_matrix_train[,!colnames(sparse_matrix_train) %in%
nm]

```

Check column names- good to go.

```{r}
```

```
all.equal(colnames(sparse_matrix_train), colnames(sparse_matrix_test))
```

Set the output values, "labels"

```{r}
output_train = dat_train_df$happiness_score

output_test = dat_test_df$happiness_score
```

XGBoost input

```{r}
dat_train <- xgb.DMatrix(data = sparse_matrix_train,label = output_train)

dat_test <- xgb.DMatrix(data = sparse_matrix_test,label = output_test)
```

Set parameters

```{r}
param_trees <- list(booster = "gbtree"
              , objective = "reg:linear"
              , subsample = 0.7
              , max_depth = 5
              , colsample_bytree = 0.7
              , eta = 0.037
              , eval_metric = 'rmse'
              , base_score = 0.012
              , min_child_weight=100)
```

Run xv

```{r}
target <- output_train
foldsCV <- createFolds(target, k=7, list=TRUE, returnTrain=FALSE)
xgb_cv <- xgb.cv(data=dat_train,
                 params=param_trees,
                nrounds=100,
                prediction=TRUE,
                maximize=FALSE,
                folds=foldsCV,
                gamma=0,
                early_stopping_rounds = 30,
                print_every_n = 5)
```

Select best nrounds and fit model

```{r}
nrounds <- xgb_cv$best_iteration
nrounds
xgb_cv$evaluation_log[xgb_cv$best_iteration,]
xgb.fit <- xgb.train(params = param_trees
                 , data = dat_train
                 , nrounds = nrounds
                 , verbose = 1
                 , print_every_n = 5
                )
```

Display importance matrix

```{r}

importancematrix <- xgb.importance(model=xgb.fit)

xgb.plot.importance((importance_matrix=importancematrix))

#head(importancematrix,10)

xgb.plot.importance(importance_matrix[1:10,])

preds <- predict(xgb.fit,dat_test)

#importance_matrix[importance_matrix$Feature=='happiness_score']

```

Plot outcomes

```{r}

plot(preds)

points(dat_test_df$happiness_score, col='red')


```

Predicted vs actual

```{r}

plot(preds, dat_test_df$happiness_score, pch=16, col="blue", cex=0.75,
xlab="Predicted happiness_score", ylab="Observed happiness_score", main=
"XGBOOST: Observed vs. Predicted")

lines(preds,

lm(a~b, data=data.frame(a=dat_test_df$happiness_score,
b=preds))$fitted,lwd=2, col="red")

```

Evaluation metrics

```{r}

actual<-dat_test_df$happiness_score

rss <- sum((preds - actual) ^ 2)  ## residual sum of squares

tss <- sum((actual - mean(actual)) ^ 2)  ## total sum of squares

rsq <- 1 - rss/tss

```

```
rsq

residuals = actual-preds

RMSE = sqrt(mean(residuals^2))

RMSE
```

```
SHAP values
```

```{r}
#Calculate shap values

shap_result = shap.score.rank(xgb_model = xgb.fit,

                              X_train =sparse_matrix_train,

                              shap_approx = F

                              )

# `shap_approx` comes from `approxcontrib` from xgboost documentation.

#Plot var importance based on SHAP

var_importance(shap_result, top_n=10)

#Prepare data for top 10 variables

shap_long = shap.prep(shap = shap_result,

                          X_train = sparse_matrix_train ,

                          top_n = 10

                          )

# Plot shap overall metrics

plot.shap.summary(data_long = shap_long)

xgb.plot.shap(data = sparse_matrix_train, # input data
```

```
              model = xgb.fit, # xgboost model

              features = names(shap_result$mean_shap_score[1:10]), # only top
10 var

              n_col = 3, # layout option

              plot_loess = T # add red line to plot

              )
```

We also used an outside function as part of our shap analysis (citation:

```
# functions for plot
# return matrix of shap score and mean ranked score list
shap.score.rank <- function(xgb_model = xgb_mod, shap_approx = TRUE,
                            X_train = mydata$train_mm){
  require(xgboost)
  require(data.table)
  shap_contrib <- predict(xgb_model, X_train,
                          predcontrib = TRUE, approxcontrib = shap_approx)
  shap_contrib <- as.data.table(shap_contrib)
  shap_contrib[,BIAS:=NULL]
  cat('make SHAP score by decreasing order\n\n')
  mean_shap_score <- colMeans(abs(shap_contrib))[order(colMeans(abs(shap_contrib)),
 decreasing = T)]
  return(list(shap_score = shap_contrib,
              mean_shap_score = (mean_shap_score)))
}

# a function to standardize feature values into same range
std1 <- function(x){
  return ((x - min(x, na.rm = T))/(max(x, na.rm = T) - min(x, na.rm = T)))
}


# prep shap data
shap.prep <- function(shap  = shap_result, X_train = mydata$train_mm, top_n){
  require(ggforce)
  # descending order
  if (missing(top_n)) top_n <- dim(X_train)[2] # by default, use all features
  if (!top_n%in%c(1:dim(X_train)[2])) stop('supply correct top_n')
  require(data.table)
  shap_score_sub <- as.data.table(shap$shap_score)
  shap_score_sub <- shap_score_sub[, names(shap$mean_shap_score)[1:top_n], with = F]
  shap_score_long <- melt.data.table(shap_score_sub, measure.vars =
 colnames(shap_score_sub))

  # feature values: the values in the original dataset
  fv_sub <- as.data.table(X_train)[, names(shap$mean_shap_score)[1:top_n], with = F]
  # standardize feature values
  fv_sub_long <- melt.data.table(fv_sub, measure.vars = colnames(fv_sub))
  fv_sub_long[, stdfvalue := std1(value), by = "variable"]
```

```
  # SHAP value: value
  # raw feature value: rfvalue;
  # standarized: stdfvalue
  names(fv_sub_long) <- c("variable", "rfvalue", "stdfvalue" )
  shap_long2 <- cbind(shap_score_long, fv_sub_long[,c('rfvalue','stdfvalue')])
  shap_long2[, mean_value := mean(abs(value)), by = variable]
  setkey(shap_long2, variable)
  return(shap_long2)
}

plot.shap.summary <- function(data_long){
  x_bound <- max(abs(data_long$value))
  require('ggforce') # for `geom_sina`
  plot1 <- ggplot(data = data_long)+
    coord_flip() +
    # sina plot:
    geom_sina(aes(x = variable, y = value, color = stdfvalue)) +
    # print the mean absolute value:
    geom_text(data = unique(data_long[, c("variable", "mean_value"), with = F]),
              aes(x = variable, y=-Inf, label = sprintf("%.3f", mean_value)),
              size = 3, alpha = 0.7,
              hjust = -0.2,
              fontface = "bold") + # bold
    # # add a "SHAP" bar notation
    # annotate("text", x = -Inf, y = -Inf, vjust = -0.2, hjust = 0, size = 3,
    #          label = expression(group("|", bar(SHAP), "|"))) +
    scale_color_gradient(low="#FFCC33", high="#6600CC",
                         breaks=c(0,1), labels=c("Low","High")) +
    theme_bw() +
    theme(axis.line.y = element_blank(), axis.ticks.y = element_blank(), # remove axis line
          legend.position="bottom") +
    geom_hline(yintercept = 0) + # the vertical line
    scale_y_continuous(limits = c(-x_bound, x_bound)) +
    # reverse the order of features
    scale_x_discrete(limits = rev(levels(data_long$variable))
    ) +
    labs(y = "SHAP value (impact on model output)", x = "", color = "Feature value")
  return(plot1)
}

var_importance <- function(shap_result, top_n=10)
{
  var_importance=tibble(var=names(shap_result$mean_shap_score),
 importance=shap_result$mean_shap_score)

  var_importance=var_importance[1:top_n,]

  ggplot(var_importance, aes(x=reorder(var,importance), y=importance)) +
    geom_bar(stat = "identity") +
    coord_flip() +
    theme_light() +
    theme(axis.title.y=element_blank())
}
```