Lowering Health Costs by Predicting Chronic Disease Cases

By Ava Masucci

Background and Problem Significance

Non-communicable, or chronic diseases could cost $47 trillion by 2030 (Duff-Brown). The global burden of disease is slowly shifting from communicable or infectious diseases to non-communicable diseases (NCD's). This shift is particularly evident in India, the second most populated country where approximately 18% of the world's population resides. Deaths from noncommunicable diseases such as diabetes, heart, and lung disease, now surpass deaths from infectious diseases in India. However, among the 29 states and 7 territories of India, the burden of disease varies widely from state to state. Identifying the states that carry the highest burden is important because noncommunicable diseases tend to be expensive and long-lasting. Additionally, many chronic diseases are largely preventable. The costs of NCD's can be reduced with adequate population health programs. Data analysis and predictive modeling can determine which states are heavily burdened with chronic disease. This information can inform policy changes that employ the most cost-effective solutions to lessen the burden of disease. In India, and many other parts of the world, health inequalities exist between gender, geography, and socioeconomic status. In my analysis, I am focusing on women's health and respective chronic disease burden.

Data Acquisition and Cleaning

I found a large health data set called "Health Analytics" on Kaggle.com. The original data set had 642 variables (columns) and 284 rows. The data is from the 2012-13 Annual Health Survey of India. The survey was conducted in the nine states of Uttarakhand, Rajasthan, Uttar Pradesh, Bihar, Jharkhand, Odisha, Chhattisgarh, Madhya Pradesh and Assam. These states account for approximately 48 percent of the total population, 59 percent of Births, 70 percent of Infant Deaths, 75 percent of Under 5 Deaths and 62 percent of Maternal Deaths (Kaggle.com). I found the women's health data particularly interesting and important--including marriage, fertility, family planning, and postpartum statistics. To begin cleaning my data, I began eliminating some of the columns:

The column names begin with letters such as "AA", "BB", "CC", etc. I deleted strings of columns with the following code:

```
drop <- Indiahealth[, -grep("AA", colnames(Indiahealth))]
```

After narrowing my columns down to less than 100, I used the following code to delete more:

```
> View(drop)
> India28 <- India[, c(2,3,4,6,8,11,12,13,18,24,25,29,38,40,55,56,57,58,61,76:85)]
```

I created binary columns for each state:

```
> IndiaMean$State_Category_Assam <- ifelse(IndiaMean$State_Name == "Assam", 1, 0)
> IndiaMean$State_Category_Bihar <- ifelse(IndiaMean$State_Name == "Bihar", 1, 0)
> IndiaMean$State_Category_Chhattisgarh <- ifelse(IndiaMean$State_Name == "Chhattisgarh", 1, 0)
> IndiaMean$State_Category_Jharkhand <- ifelse(IndiaMean$State_Name == "Jharkhand", 1, 0)
> IndiaMean$State_Category_Madhya_Pradesh <- ifelse(IndiaMean$State_Name == "Madhya Pradesh", 1, 0)
> IndiaMean$State_Category_Odisha <- ifelse(IndiaMean$State_Name == "Odisha", 1, 0)
> IndiaMean$State_Category_Rajasthan  <- ifelse(IndiaMean$State_Name == "Rajasthan", 1, 0)
> IndiaMean$State_Category_Uttar_Pradesh <- ifelse(IndiaMean$State_Name == "Uttar Pradesh", 1, 0)
> IndiaMean$State_Category_Uttarakhand <- ifelse(IndiaMean$State_Name == "Uttarakhand", 1, 0)
>
```

Next, I deleted the text categories of state and district names:

```
> IndiaMean <- IndiaMean[, -c(1,2)]
```

I replaced the NA's (missing data) with the mean for each column. Deleting the missing data resulted in deleting too much of my data set:

```
IndiaMean[] <- lapply(IndiaMean, function(x) {
```

```
+     x[is.na(x)] <- mean(x, na.rm = TRUE)
+     x
+ })
Warning message:
In mean.default(x, na.rm = TRUE) :
  argument is not numeric or logical: returning NA
> View(IndiaMean)
> row.has.na <- apply(IndiaMean, 1, function(x){any(is.na(x))})
> sum(row.has.na)
[1] 0
```

I saved my data set as the following:

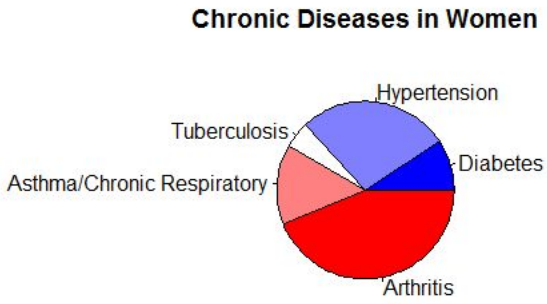
```
> saveRDS(IndiaMean, file="IndiaMean.Rda")
```

For further analysis, I converted the column "Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_Any_Kind_Of_Chronic_Illness_Female_Total" to a percentage column, Percent_Chronic_Illness, with the following code:

```
> IndiaRegRF$Percent_Chronic_Illness <-
(IndiaRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_Any_Kind_Of_Chronic_Illness_Female_Total/100000)*100
```

Exploratory Data Analysis

In my data set, the chronic diseases column is representative of five chronic illnesses:



(In this data set, tuberculosis is classified as a chronic disease. Tuberculosis is often considered a chronic disease, despite being infectious)

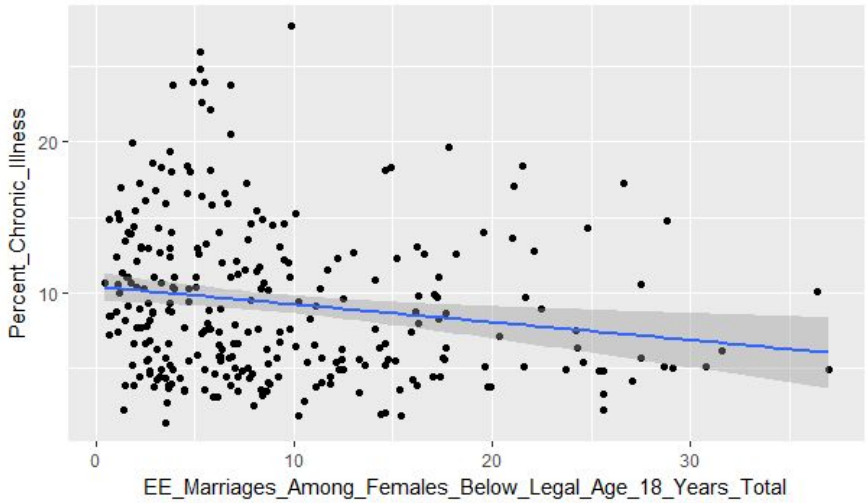
To begin further analysis, I attempted to find correlation between variables. Which variables are correlated with the number (or percentage) of women diagnosed with chronic illnesses? I used the summary() function to learn more about this variable:

```
summary(IndiaMean$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_Any_Kind_Of_Chronic_Illness_Female_Total)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1441    5164    8239    9343   12539   27701
```

I made several scatterplots to visualize the correlation between independent variables and the dependant variable:

- Are the percentages of women married below the age of 18 and those diagnosed with chronic illnesses correlated?

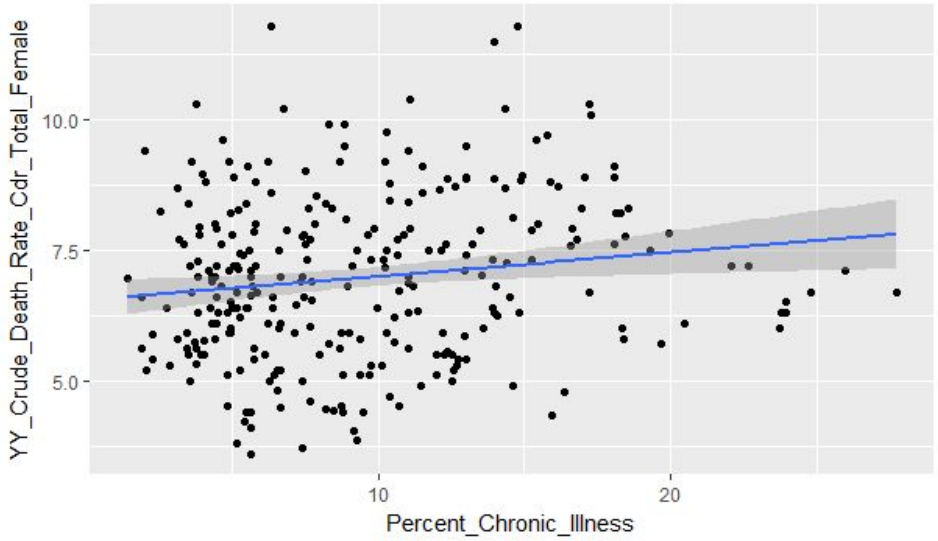
```
> MarriagevsChronic <- ggplot(IndiaPercent,
aes(EE_Marriages_Among_Females_Below_Legal_Age_18_Years_Total,
Percent_Chronic_Illness)) + geom_point() + geom_smooth(method='lm')
> print(MarriagevsChronic)
```



The scatterplot above shows a slight negative correlation between chronic illness and the percent of marriages among females below the legal age of 18 years.

- Is the Crude Death Rate correlated with chronic illnesses?

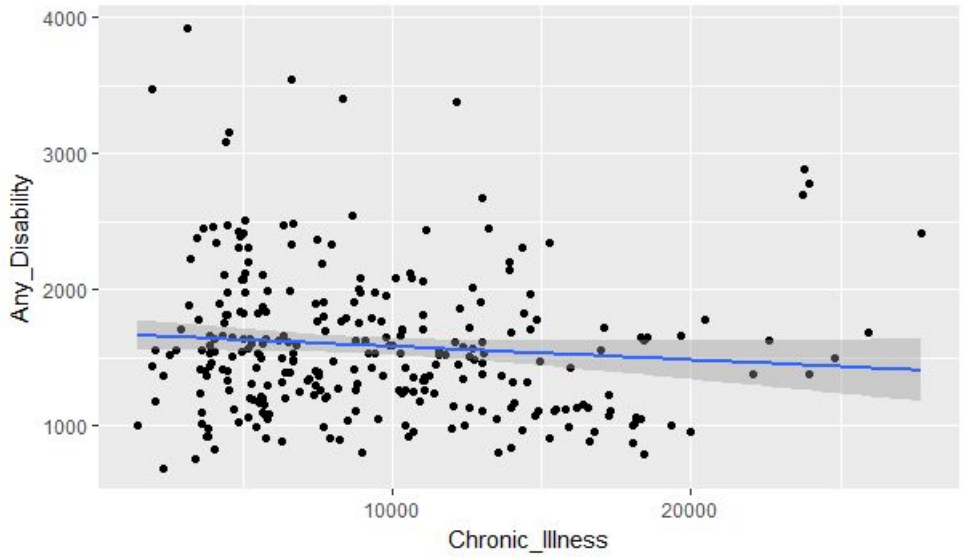
```
CDRvsChronic <- ggplot(IndiaPercent, aes(Percent_Chronic_Illness,
YY_Crude_Death_Rate_Cdr_Total_Female)) + geom_point() + geom_smooth(method='lm')
> print(CDRvsChronic)
```



This scatterplot shows a slight positive correlation between the percentage of chronically ill women and the Crude Death Rate (CDR).

- Are disabilities correlated with chronic illness?

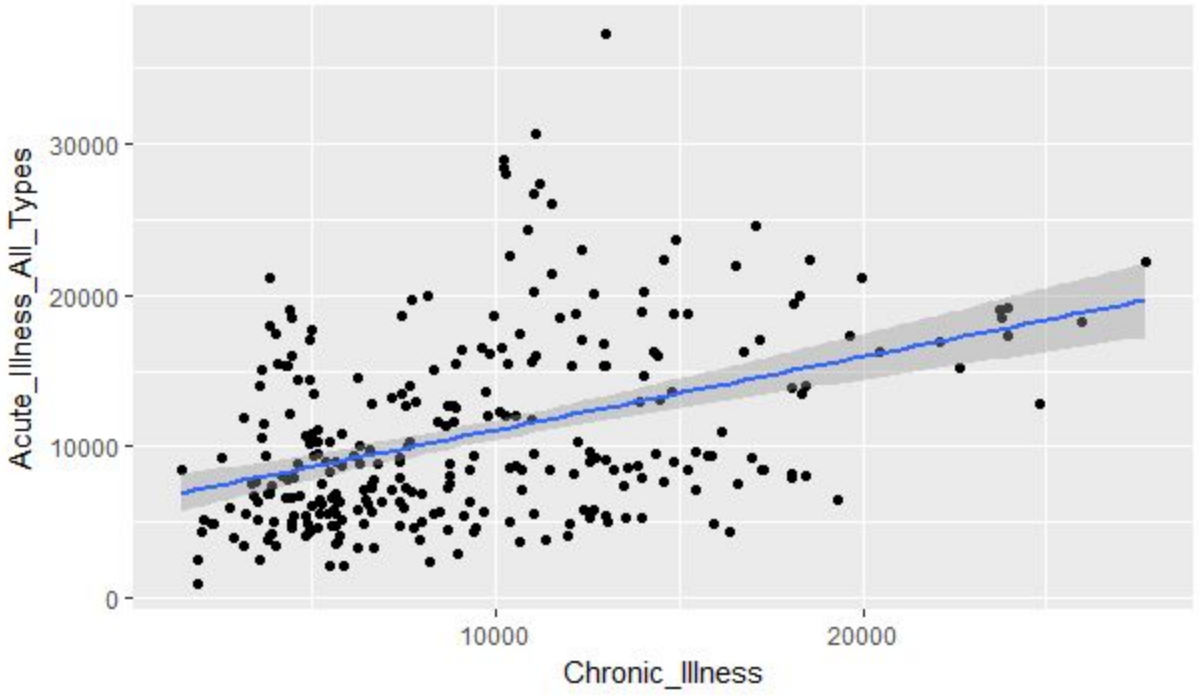
```
DisabilityvsChronic <- ggplot(IndiaMean, aes(Chronic_Illness, Any_Disability)) + geom_point() +
geom_smooth(method='lm')
> print(DisabilityvsChronic)
```



From the above scatterplot, there's an extremely slight negative correlation between chronic illnesses and having a disability in women.

- Are acute illness cases correlated with chronic illness cases?

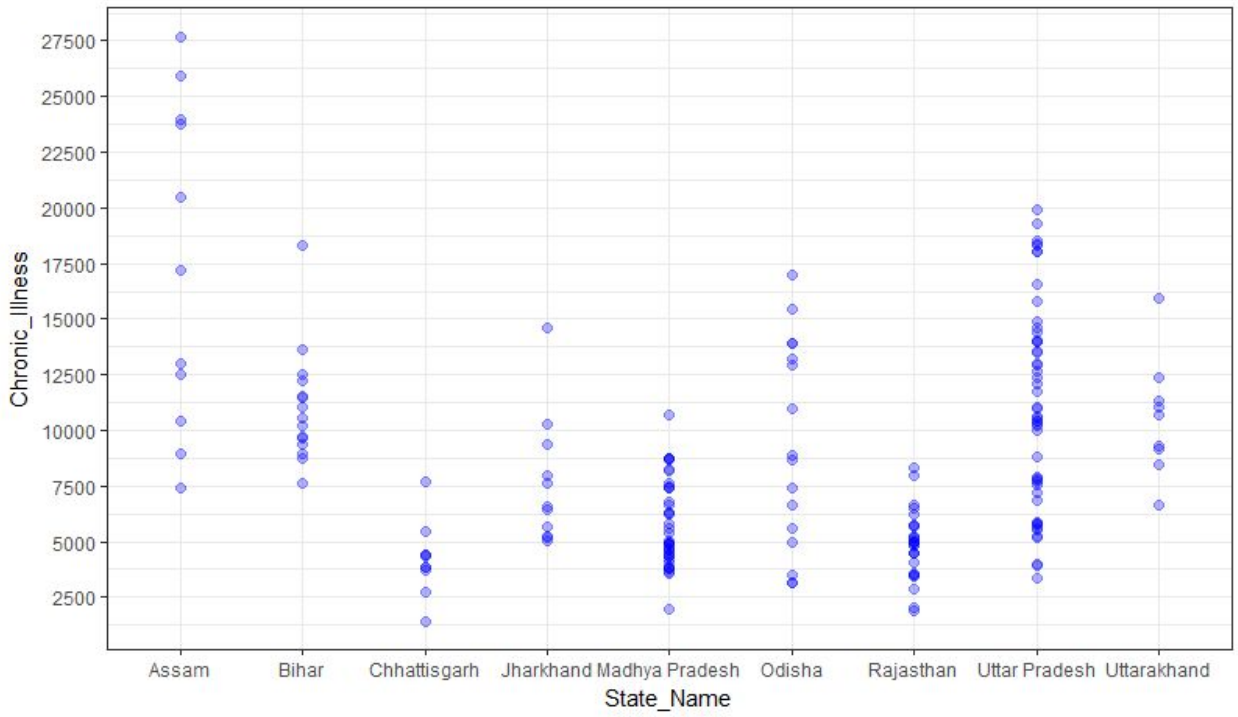
```
AcutevsChronic <- ggplot(IndiaMean, aes(Chronic_Illness, Acute_Illness_All_Types)) +
geom_point() + geom_smooth(method='lm')
> print(AcutevsChronic)
```



Cases of acute illness are positively correlated with chronic illness. Logically, this makes sense because chronic illnesses frequently weaken the immune system.

- How do the number of chronic illnesses vary by state?

```
StatevsChronic <- ggplot(IndiaHealthnoNas, aes(State_Name, Chronic_Illness)) +
geom_point(alpha = 3/10, colour = "blue", size = 2) + theme_bw() +
scale_y_continuous(breaks=seq(0,30000,2500))
> print(StatevsChronic)
```



The above plot illustrates the amount of women diagnosed with chronic illnesses per state. States are divided into districts and health data is provided per district. Districts in the states of Assam and Uttar Pradesh generally have more cases of chronic illness than the other states. Districts in Chhattisgarh and Rajasthan are generally below the median of diagnosed chronic illnesses (8239 per 100,000). Assam, Bihar, and Uttarakhand are generally above the median. State seems to be a reliable predictor of the number of women diagnosed with chronic illnesses.

Data visualization tools help frame my problem as a classification problem. Logically, some of the variables I examined should be highly correlated with chronic illnesses. However, the plots reveal that the independent variables are uncorrelated with my dependent variable, with the exception of acute illness cases. This data set is not well suited for a linear or random forest regression models. Logistic regression and random forest classification models will provide more accurate predictions.

Model Analysis

The variable I'm interested in analyzing is the number of women diagnosed with chronic illness. This is a supervised machine learning problem. From the exploratory data analysis, a classification model seems to be the best model to fit the data. To better analyze this variable, I created a new binary column, GreaterThanMedianChronicIllness:

```
summary(IndiaMean$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_Any_Kind_Of_Chronic_Illness_Female_Total)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1441    5164    8239    9343   12539   27701
> IndiaMean$GreaterThanMedianChronicIllness <-
ifelse(IndiaMean$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_Any_Kind_Of_Chronic_Illness_Female_Total > 8239, 1, 0)
>
> summary(IndiaMean$GreaterThanMedianChronicIllness)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0     0.0     0.5     0.5     1.0     1.0
```

From the above summary, it looks like this is a balanced classification problem. I proceeded to fit the data in a logistic regression model. However, I received the following: "Warning messages: 1: glm.fit: algorithm did not converge 2: glm.fit: fitted probabilities numerically 0 or 1 occurred". There was no way to resolve these warning messages so I began building a random forest classification model. My first attempt at building this model was not as accurate as it could have been, because I did not have enough data. I included about fifty more columns in my data set and re-fitted the model. I also tested varying thresholds between 0.5 and 0.9. Below is a summary of my model's rates:

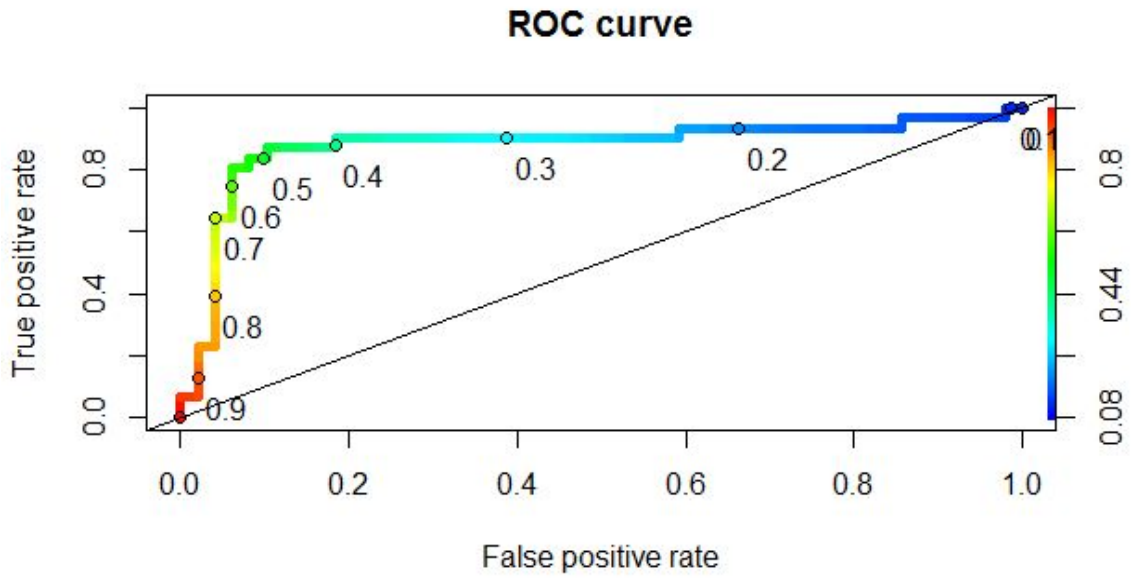
Threshold	True Positive Rate	False Positive Rate	Area Under the Curve	Accuracy
0.5	83.87097	8.163265	0.879526	88.75%
0.6	70.96774	6.122449	0.881501	85%
0.7	61.29032	4.081633	0.8765635	82.5%
0.8	38.70968	4.081633	0.8847926	73.75%
0.9	12.90323	2.040816	0.8874259	65%

(For model code, see appendix following works cited.)

The confusion matrix for threshold 0.5 is:

```
   0  1
0 45  5
1  4 26
```

This model with a threshold set at 0.5 has a high accuracy rate of 88.75%. In my analysis, it is the best model for this task because of it's high accuracy, high true positive rate, and area under the curve value. In this case, the increase in the false positive rate is outweighed by the model's accuracy. Other applications might favor a lower false positive rate with minor decreases in accuracy, such at thresholds 0.6 and 0.7. Below is the ROC curve for threshold 0.5:



The model performed well on the test set, with over 88% accuracy. Despite this model's performance, I wanted to see if I could predict the number, or percent, of women diagnosed with chronic illnesses. To accomplish this I decided to train a random forest regression model. In my first attempt, I predicted the number of women diagnosed with chronic illnesses per 100,000 population. This model returned large RMSE of 2455.413, and very large RSS of 482324395. These large numbers mean there are significant differences between the model predictions and observed data. To improve the model, I converted number of diagnosed illnesses to the percentage of diagnosed illnesses. The returned a RMSE of 2.48 and an RSS of 492.377. While this model was improved by converting the case numbers to percentages, the best model is the random forest classification model with a threshold of 0.5.

Potential clients such as global health organizations should consider the following recommendations:

1. Focus preventative medical interventions in the states, and more specifically districts, that have a greater than median number of chronic diseases.

2. Lower health care costs by distributing resources (medical personnel and supplies) efficiently across states.
3. Determine the reasons why some states have a low number of cases of chronic illness. What are the key differences between these states? How significant are environmental factors such as levels of air pollution? What about gender inequalities and socioeconomic status? Further research into these areas is warranted.

Future Analysis and Limitations

These models could predict incidences of chronic disease in other states and districts of India. However, even the best performing model will most likely perform poorly with data from other countries, particularly the United States. It would be interesting to test these models on health data from the Chinese provinces. Another limitation of the data is that it is lacking in the number of rows and time variables. A data set that included data from various years for time related predictions would be very interesting and informative. Future research could determine costs and disease burdens for specific chronic illnesses such as diabetes, heart disease, hypertension, chronic respiratory diseases, and arthritis. Further research including environmental factors and socioeconomic determinants will produce more accurate predictions.

Works Cited

Balarajan, Yarlini, S Selvaraj, and S V Subramanian. “Health Care and Equity in India.” Lancet 377.9764 (2011): 505–515. PMC. Web. 30 Nov. 2017.

Dandona, Lalit et al. “Nations within a nation: variations in epidemiological transition across the states of India, 1990–2016 in the Global Burden of Disease Study”. *The Lancet*, Volume 0, Issue 0. http://www.thelancet.com/journals/lancet/article/PIIS0140-6736(17)32804-0/fulltext 13 November 2017.

Duff-Brown, “Non-communicable disease could cost $47 trillion by 2030” Stanford University: Freeman Spogli Institute for International Studies. http://fsi.stanford.edu/news/non-communical-disease-could-cost-47-trillion-2030 7 March 2017.

Health Analytics: 26 health indicators (642 variables) from 9 states and 284 districts of India https://www.kaggle.com/rajanand/key-indicators-of-annual-health-survey

Appendix

```
#Random forest classification model code
> modelRF.8 = randomForest(as.factor(GreaterThanMedianChronicIllness) ~ . ,
+                          data=TrainM,
+                          importance=TRUE,
+                          ntree=500)
> preds.8 <- predict(modelRF.8, newdata=TestM,type="prob")
> res.8 = data.frame(preds.8)
> res$sol.8 = ifelse(res.8$X1>0.8,1,0)
> confmat.8 = table(res$sol.8, TestM$GreaterThanMedianChronicIllness)
> confmat.8

    0  1
  0 47 19
  1  2 12
> TP.rate = confmat.8[2,2]/(confmat.8[1,2]+confmat.8[2,2])
> TP.rate*100
[1] 38.70968
> FP.rate = confmat.8[2,1]/(confmat.8[1,1]+confmat.8[2,1])
> FP.rate*100
[1] 4.081633
> predAUC.8 <- prediction(res.8$X1, TestM$GreaterThanMedianChronicIllness)
> perf.8 <- performance(predAUC.8, measure = "tpr", x.measure = "fpr")
> plot(perf.8, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate",
+      xlab="False positive rate")
> abline(0, 1) #add a 45 degree line
>
> AUC.8 = as.numeric(performance(predAUC.8, "auc")@y.values)
> AUC.8
[1] 0.8847926
> modelRF.9 = randomForest(as.factor(GreaterThanMedianChronicIllness) ~ . ,
+                          data=TrainM,
+                          importance=TRUE,
+                          ntree=500)
> preds.9 <- predict(modelRF.9, newdata=TestM,type="prob")
> res.9 = data.frame(preds.9)
> res$sol.9 = ifelse(res.9$X1>0.9,1,0)
```

```
> confmat.9 = table(res$sol.9, TestM$GreaterThanMedianChronicIllness)
> confmat.9

    0  1
  0 48 27
  1  1  4
> TP.rate.9 = confmat.9[2,2]/(confmat.9[1,2]+confmat.9[2,2])
> TP.rate.9*100
[1] 12.90323
> FP.rate.9 = confmat.9[2,1]/(confmat.9[1,1]+confmat.9[2,1])
> FP.rate.9*100
[1] 2.040816
> predAUC.9 <- prediction(res$X1, TestM$GreaterThanMedianChronicIllness)
> perf.9 <- performance(predAUC.9, measure = "tpr", x.measure = "fpr")
> plot(perf.9, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate", xlab="False positive
rate") abline(0, 1) #add a 45 degree line
Error: unexpected symbol in "plot(perf.9, colorize=T, lwd=5,
print.cutoffs.at=seq(0,1,0.1), text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive
rate", xlab="False positive rate") abline"
> plot(perf.9, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate",
+      xlab="False positive rate")
> abline(0, 1) #add a 45 degree line
> AUC.9 = as.numeric(performance(predAUC.9, "auc")@y.values)
> AUC.9
[1] 0.8874259
> modelRF.7 = randomForest(as.factor(GreaterThanMedianChronicIllness) ~ . ,
+                          data=TrainM,
+                          importance=TRUE,
+                          ntree=500)
> preds.7 <- predict(modelRF.7, newdata=TestM,type="prob")
> res.7 = data.frame(preds.7)
> res$sol.7 = ifelse(res.7$X1>0.7,1,0)
> confmat.7 = table(res$sol.7, TestM$GreaterThanMedianChronicIllness)
> confmat.7

    0  1
  0 47 12
```

```
  1  2 19
> TP.rate.7 = confmat.7[2,2]/(confmat.7[1,2]+confmat.7[2,2])
> TP.rate.7*100
[1] 61.29032
> FP.rate.7 = confmat.7[2,1]/(confmat.7[1,1]+confmat.7[2,1])
> FP.rate.9*100
[1] 2.040816
> predAUC.7 <- prediction(res.7$X1, TestM$GreaterThanMedianChronicIllness)
> plot(perf.7, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate",
+      xlab="False positive rate")
Error in plot(perf.7, colorize = T, lwd = 5, print.cutoffs.at = seq(0,  :
  object 'perf.7' not found
> abline(0, 1) #add a 45 degree line
> perf.7 <- performance(predAUC.7, measure = "tpr", x.measure = "fpr")
> plot(perf.7, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate",
+      xlab="False positive rate")
> abline(0, 1) #add a 45 degree line
> AUC.7 = as.numeric(performance(predAUC.7, "auc")@y.values)
> AUC.7
[1] 0.8765635
> modelRF.6 = randomForest(as.factor(GreaterThanMedianChronicIllness) ~ . ,
+                     data=TrainM,
+                     importance=TRUE,
+                     ntree=500)
> preds.6 <- predict(modelRF.6, newdata=TestM,type="prob")
> res.6 = data.frame(preds.6)
> res$sol.6 = ifelse(res.6$X1>0.6,1,0)
> confmat.6 = table(res$sol.6, TestM$GreaterThanMedianChronicIllness)
> confmat.6

    0  1
  0 46  9
  1  3 22
> TP.rate.6 = confmat.6[2,2]/(confmat.6[1,2]+confmat.6[2,2])
> TP.rate.6*100
[1] 70.96774
> FP.rate.6 = confmat.6[2,1]/(confmat.6[1,1]+confmat.6[2,1])
```

```
> Fp.rate.6*100
Error: object 'Fp.rate.6' not found
> FP.rate.6*100
[1] 6.122449
> FP.rate.7*100
[1] 4.081633
> predAUC.6 <- prediction(res.6$X1, TestM$GreaterThanMedianChronicIllness)
> perf.6 <- performance(predAUC.6, measure = "tpr", x.measure = "fpr")
> plot(perf.6, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate",
+      xlab="False positive rate")
> abline(0, 1) #add a 45 degree line
> AUC.6 = as.numeric(performance(predAUC.6, "auc")@y.values)
> AUC.6
[1] 0.881501
> modelRF.5 = randomForest(as.factor(GreaterThanMedianChronicIllness) ~ . ,
+                     data=TrainM,
+                     importance=TRUE,
+                     ntree=500)
> preds.5 <- predict(modelRF.5, newdata=TestM,type="prob")
> res.5 = data.frame(preds.5)
> res$sol.5 = ifelse(res.5$X1>0.5,1,0)
> confmat.5 = table(res$sol.5, TestM$GreaterThanMedianChronicIllness)
> confmat.5

    0  1
  0 45  5
  1  4 26
> TP.rate.5 = confmat.5[2,2]/(confmat.5[1,2]+confmat.5[2,2])
> TP.rate.5*100
[1] 83.87097
> FP.rate.5 = confmat.5[2,1]/(confmat.5[1,1]+confmat.5[2,1])
> FP.rate.5*100
[1] 8.163265
> predAUC.5 <- prediction(res.5$X1, TestM$GreaterThanMedianChronicIllness)
> perf.5 <- performance(predAUC.5, measure = "tpr", x.measure = "fpr")
> plot(perf.5, colorize=T, lwd=5, print.cutoffs.at=seq(0,1,0.1),
text.adj=c(-0.2,1.7),main="ROC curve", ylab="True positive rate",
+      xlab="False positive rate")
```

```
> abline(0, 1) #add a 45 degree line
> AUC.5 = as.numeric(performance(predAUC.5, "auc")@y.values)
> AUC.5
[1] 0.879526
>

#Random forest regression code

Random Forest Regression

IndiaRegRF <- IndiaMean[, -c(79)]
>
> splitRegRF =
sample.split(IndiaRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_P
opulation_Any_Kind_Of_Chronic_Illness_Female_Total, SplitRatio = 0.7)
>
> TrainRegRF= subset(IndiaRegRF, split == TRUE)
> > TestRegRF = subset(IndiaRegRF, split == FALSE)
Error: unexpected '>' in ">"
>
> TestRegRF = subset(IndiaRegRF, split == FALSE)

> IndiaRegForest =
randomForest(KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_A
ny_Kind_Of_Chronic_Illness_Female_Total ~., data = TrainRegRF, nodesize=25,
ntree=500)
>
> RF.pred = predict(IndiaRegForest, newdata = TestRegRF)
> RF.sse = sum((RF.pred -
TestRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_An
y_Kind_Of_Chronic_Illness_Female_Total)^2)
>
> RF.sse
[1] 482324395
> RMSE <-
(sum((RF.pred-TestRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_
Population_Any_Kind_Of_Chronic_Illness_Female_Total)^2)/length(TestRegRF$KK_H
aving_Diagnosed_For_Chronic_Illness_Per_100000_Population_Any_Kind_Of_Chronic
_Illness_Female_Total))^(1/2)
```

```
>
> RMSE
[1] 2455.413
>
print(RMSE/mean(TestRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_1000
00_Population_Any_Kind_Of_Chronic_Illness_Female_Total))
[1] 0.303491
>
RSS2 =
sum((TestRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Populatio
n_Any_Kind_Of_Chronic_Illness_Female_Total - RF.pred)^2)
> RSS2
[1] 482324395
```

Percentage Chronic Illness Regression

```
> IndiaRegRF$Percent_Chronic_Illness <-
(IndiaRegRF$KK_Having_Diagnosed_For_Chronic_Illness_Per_100000_Population_A
ny_Kind_Of_Chronic_Illness_Female_Total/100000)*100
> IndiaPercent <- IndiaRegRF[, -c(18)]
>
> SplitPercent = sample.split(IndiaPercent$Percent_Chronic_Illness, SplitRatio = 0.7)
>
> TrainPercent= subset(IndiaPercent, split == TRUE)
>
> TestPercent = subset(IndiaPercent, split == FALSE)
>
> IndiaPercentForest = randomForest(Percent_Chronic_Illness ~., data = TrainPercent,
nodesize=25, ntree=500)
>
> percent.pred = predict(IndiaPercentForest, newdata = TestPercent)
>
> RF.sse.percent = sum((percent.pred - TestPercent$Percent_Chronic_Illness)^2)
>
> RF.sse.percent
[1] 492.377
> RMSE.percent <- (sum((percent.pred -
```

```
TestPercent$Percent_Chronic_Illness)^2)/length(TestPercent$Percent_Chronic_Illness)
)^(1/2)
>
> RMSE.percent
[1] 2.480869
> print(RMSE.percent/mean(TestPercent$Percent_Chronic_Illness))
[1] 0.3066374
>
> RSS.percent = sum((TestPercent$Percent_Chronic_Illness - percent.pred)^2)
>
> RSS.percent
[1] 492.377
```