Predicting Agricultural Output

Karan Sarkar

December 7, 2018

Introduction

Agricultural is the backbone of human civilization. All countries need sustainable sources of food. Moreover, some countries may desire excess food for security. However, predicting exact agricultural yields can be quite difficult because of random effects. In this report, We seek to explore agricultural data from the Food and Agriculture Organization of the United Nations. In particular, we want to answer several important questions:

1. How can the total agricultural production be predicted.
2. What factors can improve agricultural production and by how much?

We would like to explore these questions using machine learning methods. The advantages of machine learning methods are twofold. First, we can handle several variables simulataneously. This assures us that the observed effect of a single variable is corrected for the correlated effects of other variables. Second, we can view nonlinear effects of a variable.

These questions will help governments decide how to allocate spending to improve agricultural yield. In particular, we hope to isolate the effects of individual variables. Understanding these individual effects will allow governments to choose the most cost-effective way to boost agricultural yield. For example, we anticipate that certain variables will have diminishing returns as they are increased. We hope to be able to determine at what point a factor no longer dramatically influences agricultural yields.

Another object of our investigation is to gain a qualitative understanding of the data. We would like to explore characteristics of data. For example, are there clusters or branches in the data. Moreover, what do these clusters correspond to in the real world. What additional variables explain the existence of these clusters?

Data Description

We will be getting our data from the Food and Agriculture Organization (FAO), the UN's agricultural organization. The data is freely available at www.fao.org. Our data consists agricultural data from 1991-2002 for several countries. We are trying to predict the total production quantity in tons. Our predicts are the area harvested, amount of fertilizer used, amount of pesticides used and the total machinery used. We will be considering each pair of country and year to be a unique entry.

```
data <- read.csv("data.csv")
head(data)
```

```
##   COUNTRY YEAR AREA_HARVESTED PRODUCTION_QUANTITY FERTILIZER PESTICIDES
## 1 Albania 1991         215339              446152      31300     121.00
## 2 Albania 1992         198934              428949      23500     121.00
## 3 Albania 1993         247936              665720      17890     121.00
## 4 Albania 1994         262515              646026      14770     201.00
## 5 Albania 1995         227137              645400      11000     251.00
## 6 Albania 1996         205526              503714       6500     313.96
##   MACHINERY
## 1      9469
## 2      9000
## 3      9049
## 4      9100
## 5      8938
## 6      8313
```

Each row consists of a string represented the country and an integer for the year. Area harvested is measured in hectares. Production quantity and fertilize/pesticide consumption is measured in tons. On the other hand, machinery is simply measured by the count of agricultural heavy machinery.

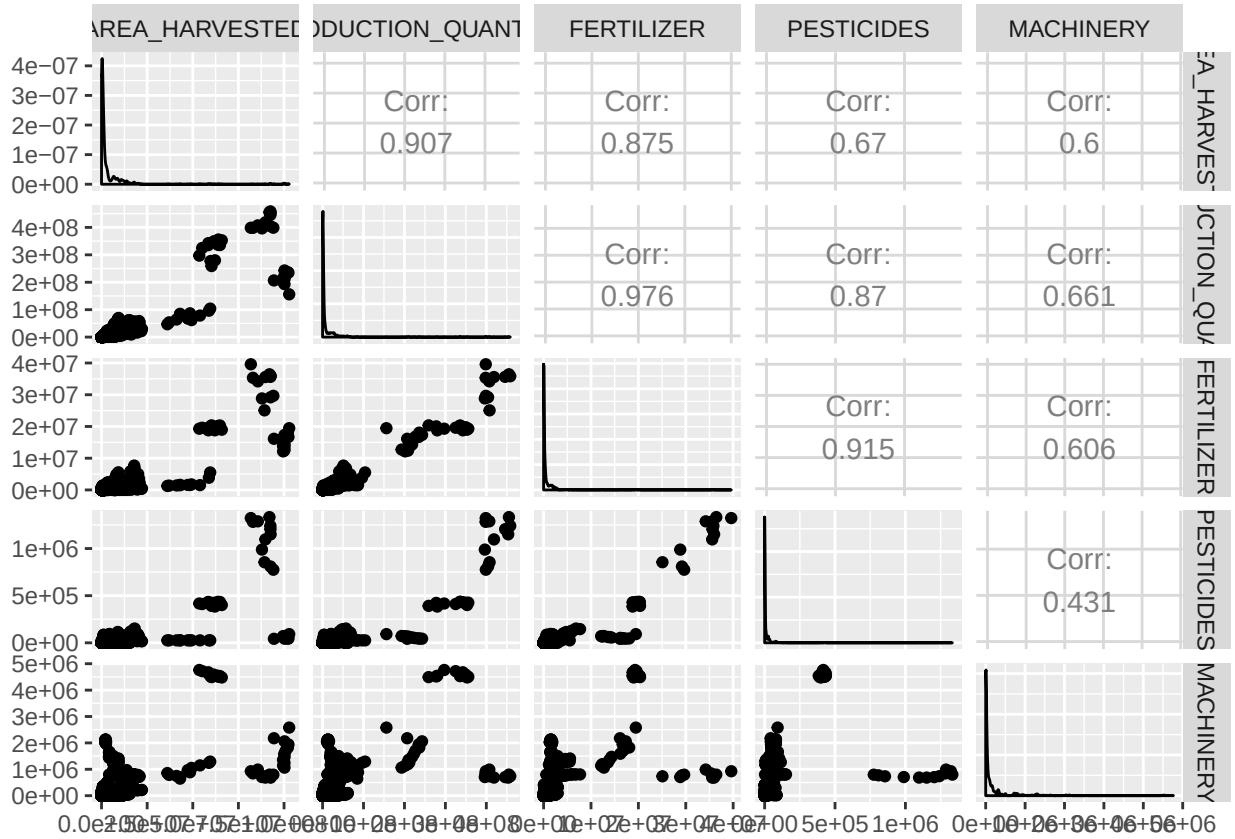
This data required a significant amount of cleaning. Each predictor was found in a different spreadsheet. Each spread sheet included different years and countries. We chose the years so that most of the spreadsheets available would be usable. Moreover, we had to only use the subset of countries that had all of their data available. It look a significant amount of time to remove all of the countries that had just partial data. We stored the unified dataset into a comma separate value file for easy usage.

Analysis

```
library(GGally)
```

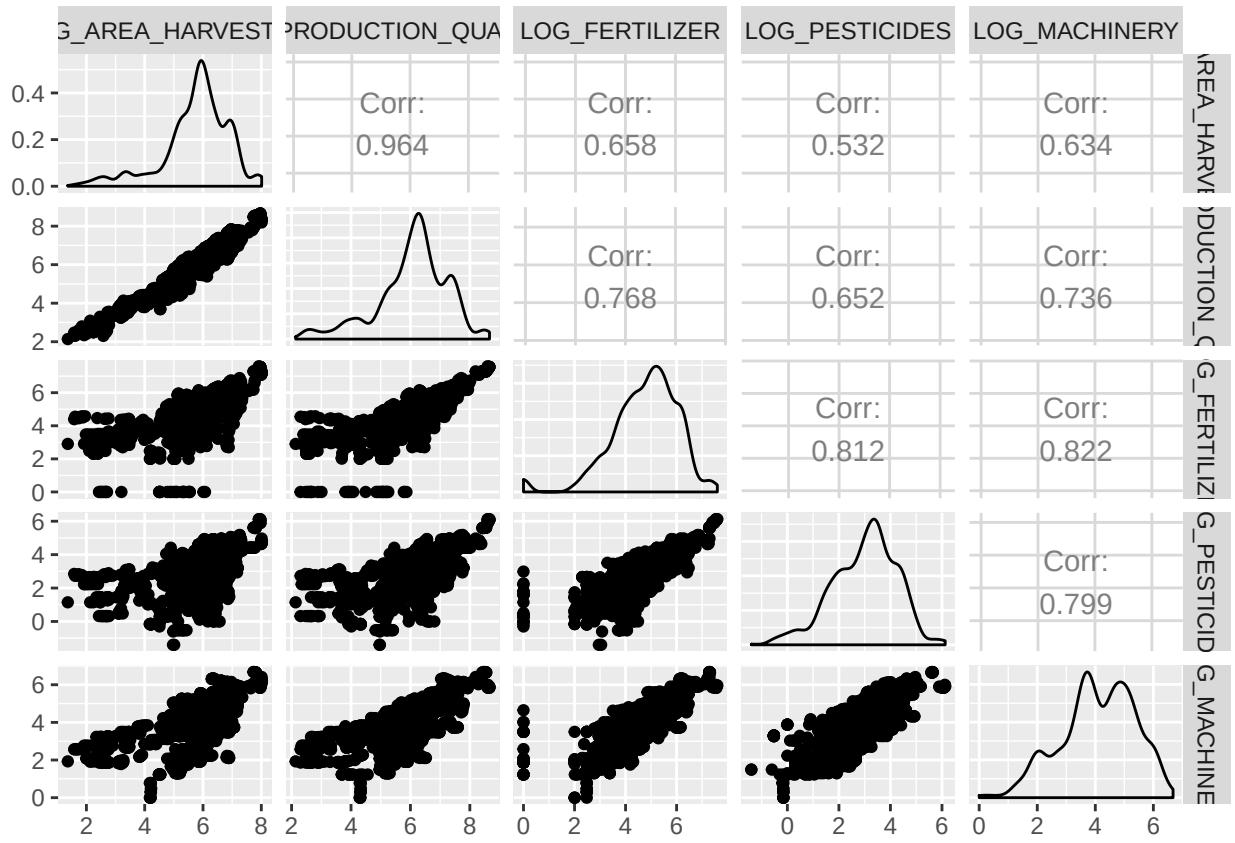
```
## Loading required package: ggplot2
```

```
ggpairs(data, columns = 3:7, progress = FALSE)
```



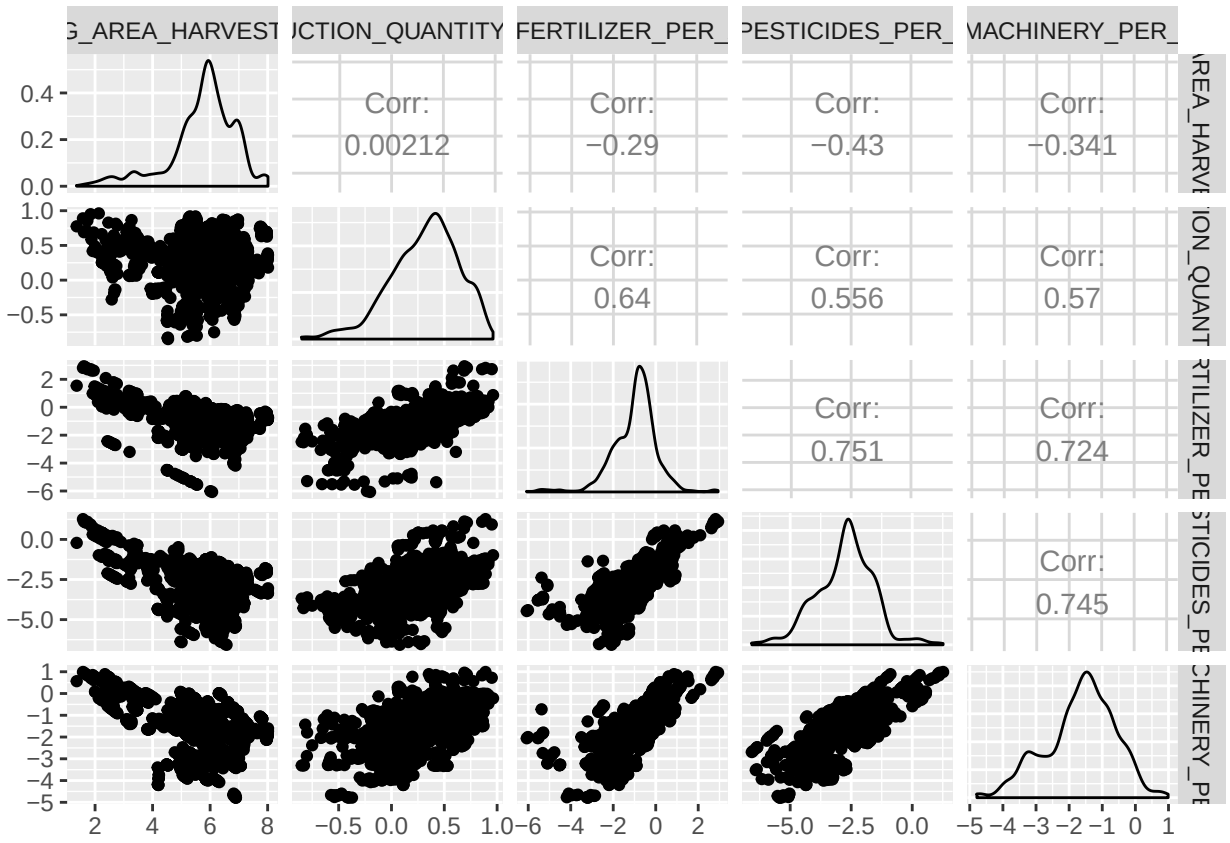
To get a handle on the data, we have created a scatterplot matrix. The diagonal shows density plots of the variables. We can easily see that all of the variables are heavily skewed to the lower numbers. Therefore, it may make sense to try a logarithm re-expression. Having more evenly dispersed data will make it easier to analyze.

```
data$LOG_AREA_HARVESTED <- log10(data$AREA_HARVESTED)
data$LOG_PRODUCTION_QUANTITY <- log10(data$PRODUCTION_QUANTITY)
data$LOG_FERTILIZER <- log10(data$FERTILIZER + 1)
data$LOG_PESTICIDES <- log10(data$PESTICIDES)
data$LOG_MACHINERY <- log10(data$MACHINERY + 1)
ggpairs(data, columns = 8:12, progress = FALSE)
```



After the logarithm re-expression, all of the variables have roughly unimodal and symmetric distributions. We can now observe that all of the variables are positively correlated with each other. This makes sense because all of the these variables are total quantities over a country. Countries with large amounts of arable land have high scores in all variables. What we are really interested in though is how different countries allocate resources differently to these factors. Therefore, we will subtract log area harvested from all of these variables to obtain the logarithm of each factor per unit area. For example, rather than log total fertilizer, we will use log fertilizer per area.

```
data$LOG_PRODUCTION_QUANTITY_PER_AREA <- data$LOG_PRODUCTION_QUANTITY - data$LOG_AREA_HA
data$LOG_FERTILIZER_PER_AREA <- data$LOG_FERTILIZER - data$LOG_AREA_HARVESTED
data$LOG_PESTICIDES_PER_AREA <- data$LOG_PESTICIDES - data$LOG_AREA_HARVESTED
data$LOG_MACHINERY_PER_AREA <- data$LOG_MACHINERY - data$LOG_AREA_HARVESTED
ggpairs(data, columns = c(8,13:16), progress = FALSE)
```



Here is the scatterplot matrix with the per area metrics shown. The per area metrics are all positively correlated with each other as expected. One interesting observation is that countries with smaller areas harvested use greater proportions of machinery, pesticides and fertilizer and have subsequently greater production rates. This makes sense for two reasons. First, smaller countries are often more densely populated and need to invest more into agriculture. Second, smaller countries have less land and have an easier time splurging on agricultural investments.

One potential source of basis in our analysis is that we are only considering the production of cereal crops. However, we are considering total amounts of fertilizer, pesticdes and machinery. This is an unfortunate occurence due to the nature of the data resources. However, according to the FAO, cereals account for about 60% of total agricultural output, so we hope our results are still applicable. Another potential problem, is climate driven changes in agricultural output. For example, droughts one year could mean decreased output. We hope that these yearly fluctuations balance out and do not bias our data.

Model Development

Random Forest

We will start by using a random forest model to predict the production quantity per area. We choose the random forest model because it handles quantitative variables and potential

nonlinear relationships very well. We will first tune our hyperparameters namely mtry and ntree. Mtry refers to the number of variables considered and ntree is the number of individual trees. We want to determine what values of both maximize our accuracy.

```
library(randomForest)
```

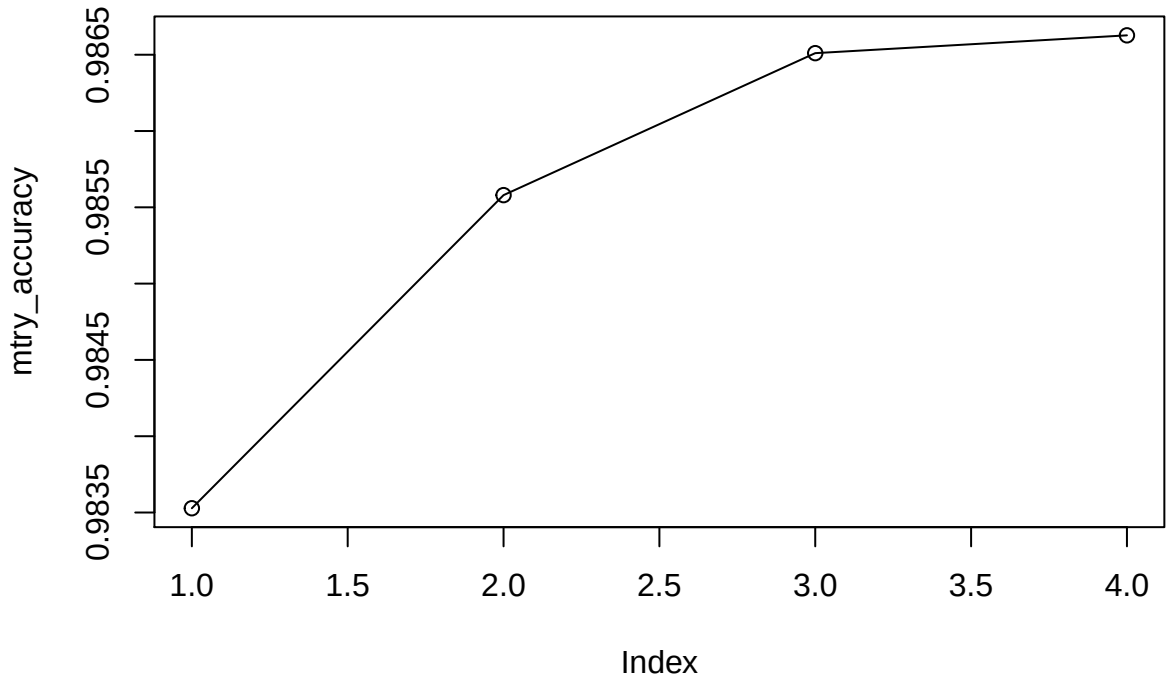
```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

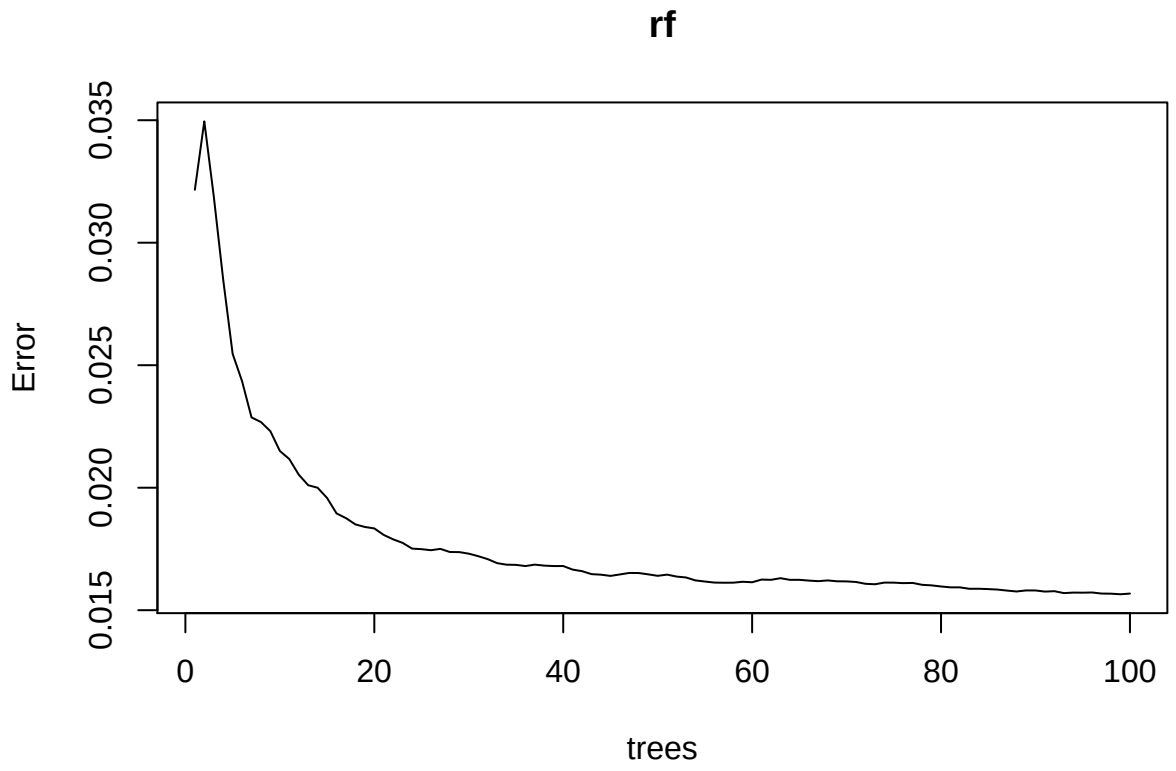
```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
mtry_accuracy <- numeric(4)
for (i in 1:4){
    rf_test <- randomForest(
    data$LOG_PRODUCTION_QUANTITY_PER_AREA ~ data$LOG_AREA_HARVESTED +
    data$LOG_FERTILIZER_PER_AREA + data$LOG_PESTICIDES_PER_AREA +
    data$LOG_MACHINERY_PER_AREA,
    data, keep.inbag = TRUE, importance = TRUE,  mtry = i, ntree = 100)
    preds <- predict(rf_test, data)
    mtry_accuracy[i] <- cor(preds, data$LOG_PRODUCTION_QUANTITY_PER_AREA)
}
plot(mtry_accuracy)
lines(mtry_accuracy)
```



The line plot above clearly shows that 3 is the optimal value of mtry. We can see that the random forest underfits for mtry = 1 or 2. This means that for small mtry values, the random forest model is not complex enough to fully fit the data. On the other hand, when mtry = 4, the accuracy drops due to overfitting. This means that there are too many degrees of freedom allowing the model to inadvertently memorize random fluctuations in the data as opposed to trends. We will be continuing will mtry = 3 from now on.

```
rf = randomForest(
  data$LOG_PRODUCTION_QUANTITY_PER_AREA ~ data$LOG_AREA_HARVESTED +
    data$LOG_FERTILIZER_PER_AREA + data$LOG_PESTICIDES_PER_AREA +
    data$LOG_MACHINERY_PER_AREA,
  data, keep.inbag = TRUE, importance = TRUE,  mtry = 3, ntree = 100,
)
plot(rf)
```

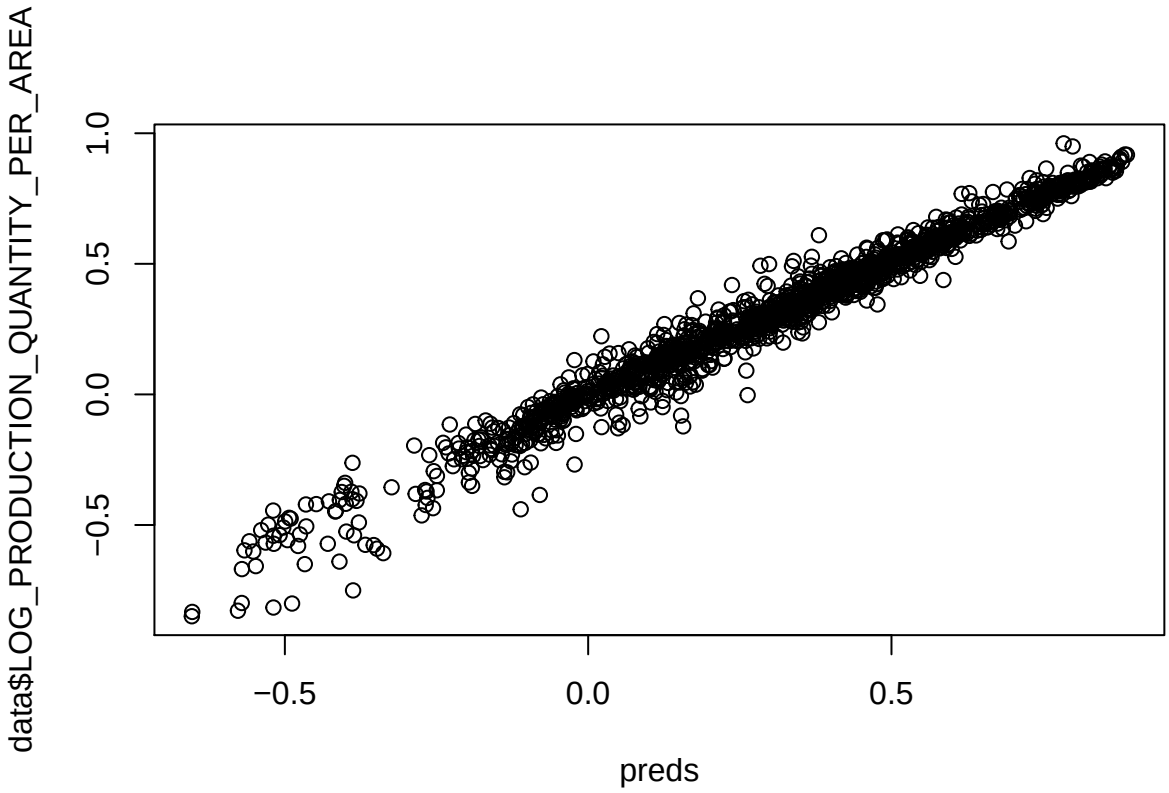


The plot of ntree vs. error shows us that 100 trees is sufficient. In the plot, the error rates drops rapidly at first and subsequently levels off. Because 100 trees is well past the point at which error levels off, we can be confident that a further increase in the number of trees will not influence accuracy significantly. Therefore, we will be maintaining our ntree value of 100.

```
rf
```

```
##
## Call:
##  randomForest(formula = data$LOG_PRODUCTION_QUANTITY_PER_AREA ~      data$LOG_AREA_HA
##                Type of random forest: regression
##                      Number of trees: 100
## No. of variables tried at each split: 3
##
##           Mean of squared residuals: 0.01567897
##                     % Var explained: 84.58
```

```
preds <- predict(rf, data)
plot(preds, data$LOG_PRODUCTION_QUANTITY_PER_AREA)
```

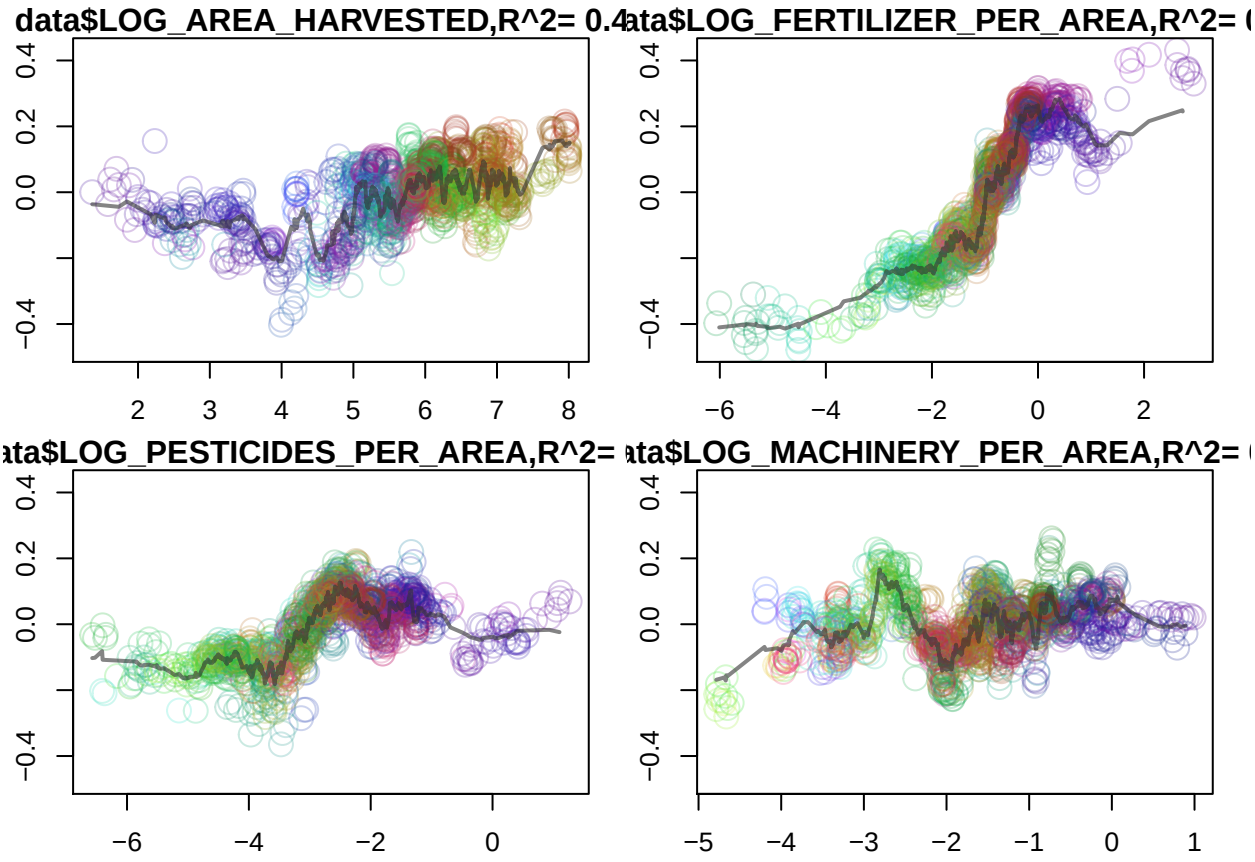


Our model explained 85% of the variance in the production quantity of cereals per unit area. Therefore, our model is an accurate description of how factors such as pesticides, machinery and fertilizer influence productivity. We will now display a scatterplot of predicted yield vs actual yield in order to visualize the accuracy. Note that because random forest uses sample to train each tree, we do not need a validation phase. Moreover, from the scatterplot, we can see that the random forest model is very accurate. There is a very strong positive relationship between the predicted value and the actual value.

We will now use the forestFloor package to generate contribution plots for the random forest. In particular, we are interested in seeing which variables are the most important

```
library(forestFloor)
ff = forestFloor(
  rf = rf ,
  X = data[,c(8,13:16)],
  calc_np = FALSE,
  binary_reg = FALSE
)
plot(ff,  col = fcol(ff,orderByImportance=FALSE), orderByImportance=FALSE , cex = 2)
```

```
## [1] "compute goodness-of-fit with leave-one-out k-nearest neighbor(guassian weighting
```



We can see that that fertilizer is the best predictor of yield while factors like area harvested and machinery are less important. Additionally, we can identify critical points for different variables. In particular, at a certain point an increase in a variable does not produce an increase in productivity. For example, fertilizer does not have a significant effect at levels above 1 ton per hectare. Similarly, pesticides do not have a tangible influence above 0.01 tons per hectare. Therefore, we can prescribe optimal levels for different agricultural components.

Linear Regression

One problem with random forests is that we cannot precisely quantify the effects of individual variables. Using linear regression will allow us to quantify average effect of a variable. The weakness of linear regression though is the inability to handle nonlinear trends. Therefore, linear regression will in general lose predictive accuracy. In order to avoid overfitting, we will use cross-validation and the lasso shrinkage method.

We will begin by using cross validation to pick a value of lambda. Essentially, we are recording test error at different lambda values. We want to pick the value of lambda that minimizes the total error and avoids both overfitting and underfitting.

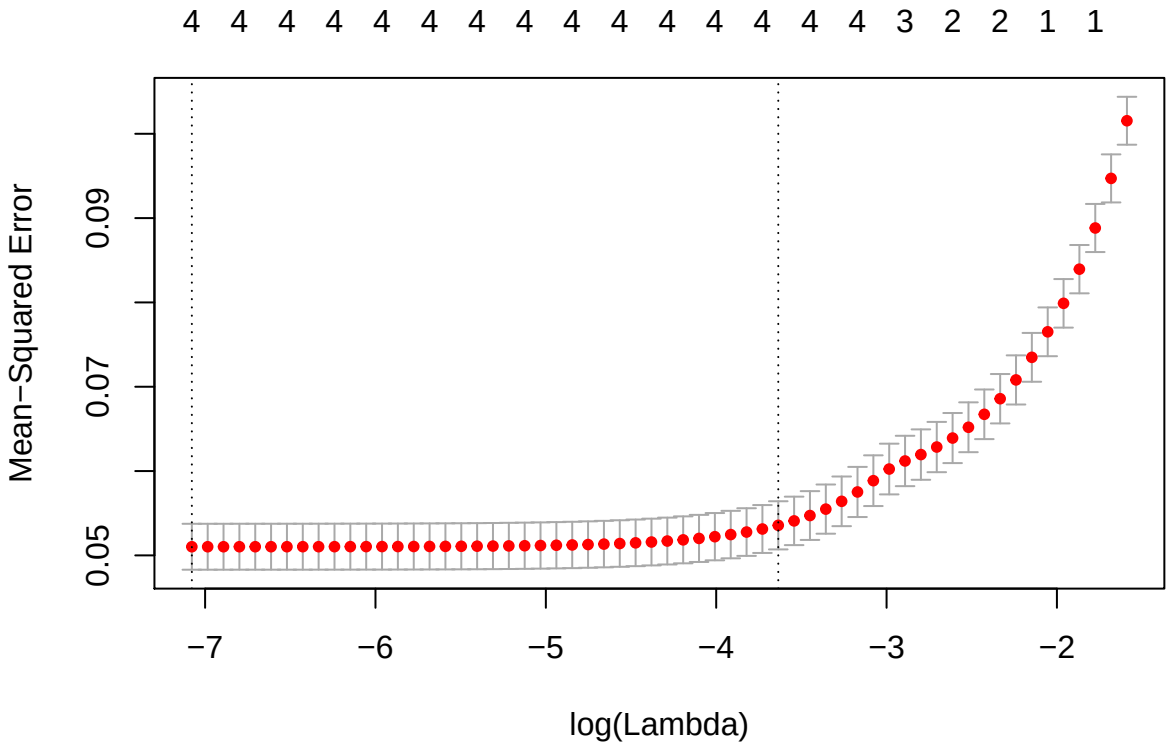
```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach

## Loaded glmnet 2.0-16
```

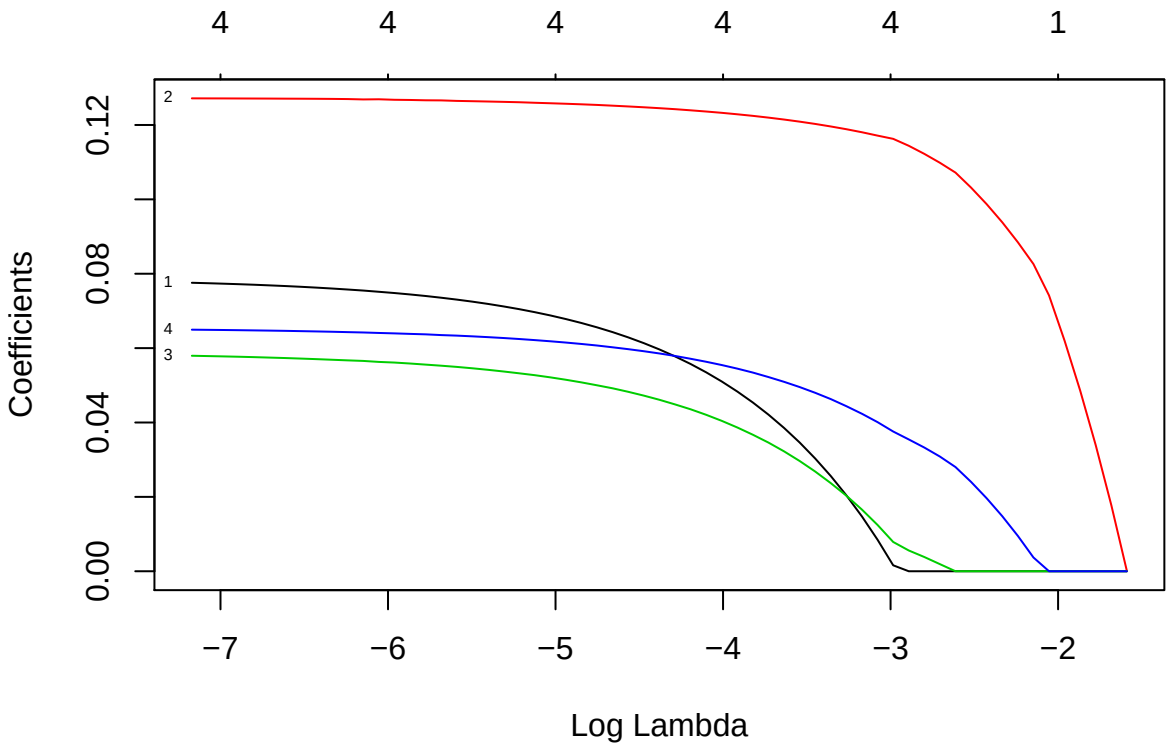
```
model <- cv.glmnet(as.matrix(data[,c(8,14:16)]), data[,13], standardize=TRUE, nfolds = 5
plot(model, 1)
```



We can see that the optimal value of lambda is zero. This means that our model is not having any problems with overfitting. This makes sense because the number of data points is so much greater than the number of predictors. Because the number of degrees of freedom for the linear model is significantly less than the number of data points, overfitting is less of a problem. We will now be using a zero value for lambda.

In order to get a better understanding of the data, we have made variable trace plots. In a variable trace plot, we plot the value of each of the variable coefficients against the value of lambda. As lambda increases, the values of each of the coefficients goes to zero. However, more important variables will have coefficients that take longer to vanish.

```
plot(model$glmnet.fit, "lambda", label=TRUE)
```



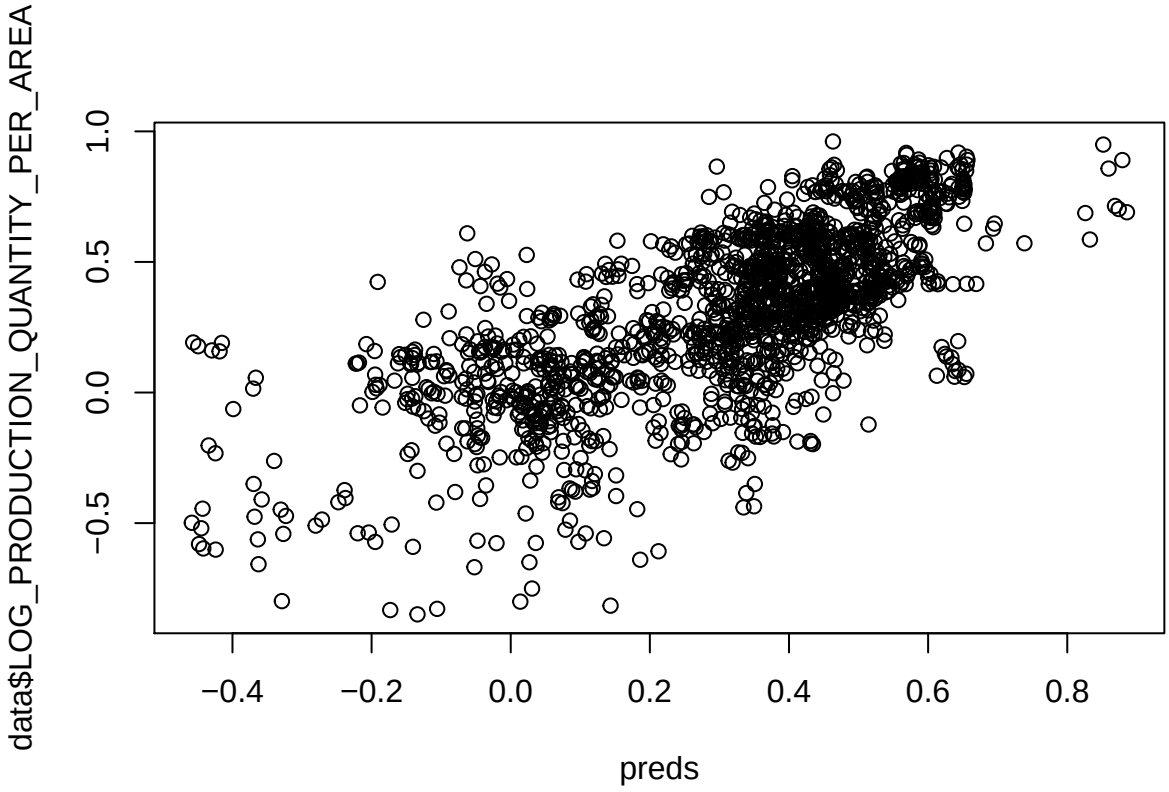
The variables are labeled 1,2,3,4. However, we will add that variable 1 is area harvested. Variable 2 is fertilizer. Variable 3 is pesticides and variable 4 is machinery. We can easily observe from this plot that fertilizer is by far the most important variable with machinery coming in second. On the other hand, area harvested is not as important. One interesting observation is that area harvested has a large coefficient but is the least statistically significant. This might be because there is so much noise with respect to that variable as we observed in the random forest plots.

```
lm <- lm(data$LOG_PRODUCTION_QUANTITY_PER_AREA ~ data$LOG_AREA_HARVESTED +
    data$LOG_FERTILIZER_PER_AREA + data$LOG_PESTICIDES_PER_AREA +
    data$LOG_MACHINERY_PER_AREA,data)
summary(lm)
```

```
##
## Call:
## lm(formula = data$LOG_PRODUCTION_QUANTITY_PER_AREA ~ data$LOG_AREA_HARVESTED +
##     data$LOG_FERTILIZER_PER_AREA + data$LOG_PESTICIDES_PER_AREA +
##     data$LOG_MACHINERY_PER_AREA, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.95947 -0.12726  0.00354  0.16302  0.67217
##
```

```
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.257056   0.028333   9.073  < 2e-16 ***
## data$LOG_AREA_HARVESTED        0.078751   0.005262  14.965  < 2e-16 ***
## data$LOG_FERTILIZER_PER_AREA 0.127273   0.008732  14.575  < 2e-16 ***
## data$LOG_PESTICIDES_PER_AREA 0.058769   0.008309   7.073 2.20e-12 ***
## data$LOG_MACHINERY_PER_AREA  0.065380   0.008519   7.675 2.78e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2253 on 1697 degrees of freedom
## Multiple R-squared:  0.5023, Adjusted R-squared:  0.5011
## F-statistic: 428.2 on 4 and 1697 DF,  p-value: < 2.2e-16
```

```
preds <- predict(lm, data)
plot(preds, data$LOG_PRODUCTION_QUANTITY_PER_AREA)
```



We have now computed the linear model with a lambda value of zero. The linear model has an R^2 of 0.5 meaning that it explains 50% of the variance. This is significantly less that the 85% explained by the random forest. This is corroborated by the scatterplot which shows significantly more scatter than the one for the random forest model. The advantage of the linear model is that we have quantitative measures of the influence of a variable and can test for statistical significance. For example, a 1% increase in the amount of fertilizer on average

leads to a 0.13% increase in the amount of yield. Moreover, all of our variables have been determined to be statistically significant.

Discussion

We will make our conclusions based on both the linear model and the random forest. We conclude that fertilizer per unit area is the most important factor for agricultural yields. It reaches an optimal values at around 1 ton per hectare. On average, an additional 1% fertilizer per hectare leads to 0.12% additional productivity per hectare. We believe that additional fertilizer up until the threshold of 1 ton per hectare is the best investment to improve yield.

In addition, the least important variable is area harvested. The linear model gives it a large coefficient. In particular, the linear model predicts that on average a 1% increase in area harvested will lead to an additional 0.8% increase in productivity. However, we believe that this correlation is not causative. We believe that countries with smaller amounts of arable land spend more per unit area than countries with larger amounts of land. This belief is supported by the random forest plots which show that the smaller areas harvested tend to have higher amounts of fertilizer and pesticides.

Furthermore, machinery and pesticides are about of equal importance. Pesticides seems to reach an optimal value at around 0.01 tons per hectare. On the other hand, machinery seems to have a clear positive influence over its whole range of values. Therefore, if a country has excess agricultural funds, we would reccomend spending it on pesticides until reaching the threshold of 0.01 tons per hectare. After exceeding that threshold, the best investment is in additional machinery.

Our research did not run into many problems. Our data was exactly the initial data we envisoned and the problem we addresed was what we intended from the start. We chose the random forest in order to model nonlinearities in the data. Another reason was our intention from the start to use contribution plots to determine the optimal values of different variables. We chose the linear model because it is very statistical and would allow us to confidently quantify the relationships between variables. Despite having less accuracy than a random forest, the linear model allowed us to determine statistiscal significance with out any complications.

One problem we did run into was our initial attempt to use bayesian networks. Unfortuantely, our data is quantitative. Bayesian networks shine when they handle categorical variables. In order to use bayesian networks, we can to discretize our variables. However, this was clunky and made the bayesian network very difficult to interpret. Because our primary purpose was to be able to prescribe future agricultural spending, interpretability was very important to us. Therefore, we decided not to continue to use the bayesian network.

For future investigations into agricultural data, we would consider using additional variables. Some variables could include credit given to farmers as well as the year. Moreover, we could use climate data such as rainfall and soil type to correct for potential biases within the data. In addition, we could break down variables like fertilizer, machinery and pesticides into many smaller categories like potassium, tractors and insecticide. This would allow a more fine grained approach to the problem. It would allow even more specific reccomendations for agricultural spending.