CSP571 DPA ProjectPrediction of Rice price per district in Andhra Pradesh

Santosh Reddy Edulapalle Jobin Joyson Venkata Siva Rupesh Akurati

2022-11-07

Loading Dataset

```
df.rice.farm.harvest.price <- read.csv(unz("Datasets.zip","Datasets/Rice_farm_harvest_price.csv"))

df.rice.area.production.yield <- read.csv(unz("Datasets.zip","Datasets/Rice_area_production_yield.csv"))

df.irrigation.sourcewise.irrigated.area <- read.csv(unz("Datasets.zip","Datasets/Irrigation_sourcewise_

df.irrigation.rice.area <- read.csv(unz("Datasets.zip","Datasets/Irrigation_rice_area.csv"))

df.inputs.wages <- read.csv(unz("Datasets.zip","Datasets/Inputs_wages.csv"))

df.inputs.season.fertilizer.consumption <- read.csv(unz("Datasets.zip","Datasets/Inputs_season_fertiliz

df.inputs.fertilizer.consumption <- read.csv(unz("Datasets.zip","Datasets/Inputs_fertilizer_consumption

df.environment.temperature.minimum <- read.csv(unz("Datasets.zip","Datasets/Environment_temperature_mini

df.environment.temperature.maximum <- read.csv(unz("Datasets.zip","Datasets/Environment_temperature_maxi

df.environment.precipitation <- read.csv(unz("Datasets.zip","Datasets/Environment_precipitation*.csv"))

df.environment.evapotranspiration.potential <- read.csv(unz("Datasets.zip","Datasets/Environment_evapoti

df.environment.evapotranspiration.actual <- read.csv(unz("Datasets.zip","Datasets/Environment_evapotrans

df.biophysical.soil.type1 <- read.csv(unz("Datasets.zip","Datasets/Biophysical_soil_type.csv"))

df.biophysical.monthly.rainfall <- read.csv(unz("Datasets.zip","Datasets/Biophysical_monthly_rainfall.cs

df.biophysical.length.of.growing.period <- read.csv(unz("Datasets.zip","Datasets/Biophysical_length_of_g

df.biophysical.landuse <- read.csv(unz("Datasets.zip","Datasets/Biophysical_landuse.csv"))
```

Data Processing

Initially, district codes 503,504 were missing in some tables because these districts are formed after state division. So, a lot of NAs were formed while performing joins. Now we gathered latest possible information for all data sets.

Deleting rows < 1990

Other than environment, entire data sets starts from 1990. So, to get fair results, we are deleting observations from environment data sets that are recorded in years < 1990.

```
#deleting rows less than 1990

del.1990 <- function(df.name){
  df.name <- df.name[df.name$Year >= 1990,]
  return(df.name)
}

df.environment.evapotranspiration.actual <- del.1990(df.environment.evapotranspiration.actual)

df.environment.evapotranspiration.potential <- del.1990(df.environment.evapotranspiration.potential)

df.environment.precipitation <- del.1990(df.environment.precipitation)
df.environment.temperature.maximum <- del.1990(df.environment.temperature.maximum)
df.environment.temperature.minimum <- del.1990(df.environment.temperature.minimum)
```

Deleting rows > 2015

So, some tables end with 2015 while others end with 2017. Our target table end with Year 2016. So, to get fair results, we are deleting observations from data sets that are recorded in years > 2015.

```
#deleting rows more than 2015

del.2015 <- function(df.name){
  df.name <- df.name[df.name$Year <= 2015,]
  return(df.name)
}

df.biophysical.landuse <- del.2015(df.biophysical.landuse)
df.biophysical.monthly.rainfall <- del.2015(df.biophysical.monthly.rainfall)
df.inputs.fertilizer.consumption <- del.2015(df.inputs.fertilizer.consumption)
df.inputs.season.fertilizer.consumption <- del.2015(df.inputs.season.fertilizer.consumption)
df.inputs.wages <- del.2015(df.inputs.wages)
df.irrigation.rice.area <- del.2015(df.irrigation.rice.area)
df.irrigation.sourcewise.irrigated.area <- del.2015(df.irrigation.sourcewise.irrigated.area)
df.rice.area.production.yield <- del.2015(df.rice.area.production.yield)
df.rice.farm.harvest.price <- del.2015(df.rice.farm.harvest.price)
```

Handling missing values

(In the ICRISAT website, we are said that missing values are coded as -1).

Process:

1. Initially, we skimmed thourgh all datasets and deleted the records which have -1. Later after combining all the datasets to single df.all dataframe, we were able to see lot of NAs. After going thorough careful investigation, we found out that since we deleted couple of observations in few datasets, those years-observations were causing NA while joining with other datasets. So, we Handled missing values by taking appropriate years and handling df.input.wages as it is largest missing data. Additionally, we did not have districts 503,504 for few tables, so we got new and updated datasets.
2. Now we are taking summaries of all tables, and checking if any table has min:-1, so we are handling those specific columns by performing mean of each district in that specific year gap (10 year average

for missing districts using filter())

3. After doing step1,2 and combining the tables, we still got few NAs after checking summary of each join, we found that - df.rice.farm.price table has 2 missing values for district 503,504 for year 1995. and this was not coded as -1. Instead it is completely missing. same case for df.input.wages for all districts for years 2010,2011.

df.irrigation.sourcewise.irrigated.area$OTHER.WELLS.AREA..1000.ha. = has missing values for districts 50,51 for years 2007.

df.rice.farm.harvest.price$PADDY.HARVEST.PRICE..Rs.per.Quintal. = has missing values for districts 50,51,52,53 for years 1998.

df.inputs.season.fertilizer.consumption$NITROGEN.KHARIF.CONSUMPTION..tons. -> (All the column values of these given districts is missing for these two particular years - 1998,1999 Srikakulam, Vishakapatnam, East Godavari, West Godavari, Krishna, Guntur, S.P.S.Nellore, Kurnool, Ananthapur, Kadapa YSR, Chitoor, Prakasam, Vijianagaram.)

df.inputs.wages –> There are 132 missing values. (almost 25% data is missing)

Handling NA values for df.irrigation.sourcewise.irrigated.area

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#replace -1 by NA

df.irrigation.sourcewise.irrigated.area$OTHER.WELLS.AREA..1000.ha. <- ifelse(df.irrigation.sourcewise.i

#create a test data frame with 0 rows and 13 columns
test.df <- data.frame(Dist.Code = integer(),
                      Year = integer(),
                      State.Code = integer(),
                      State.Name = character(),
                      Dist.Name = character(),
                      CANALS.AREA..1000.ha. = double(),
                      TANKS.AREA..1000.ha.  = double(),
                      TUBE.WELLS.AREA..1000.ha. = double(),
                      OTHER.WELLS.AREA..1000.ha. = double(),
                      TOTAL.WELLS.AREA..1000.ha. = double(),
                      OTHER.SOURCES.AREA..1000.ha. = double(),
                      NET.AREA..1000.ha.  = double(),
                      GROSS.AREA..1000.ha.   = double(),
                      stringsAsFactors = F
                      )
dist <- c(44,45,46,47,48,49,50,51,52,53,54,503,504)
for(y in dist){
  #we only have missing value for the 2007
```

```
  #round1 - 1990:1999
  x <- filter(df.irrigation.sourcewise.irrigated.area, df.irrigation.sourcewise.irrigated.area$Dist.Cod

  x$OTHER.WELLS.AREA..1000.ha. <- ifelse( is.na(x$OTHER.WELLS.AREA..1000.ha.), mean(x$OTHER.WELLS.AREA.
  test.df <- bind_rows(test.df,x)
    #round2 - 2000:2009
  x1 <- filter(df.irrigation.sourcewise.irrigated.area, df.irrigation.sourcewise.irrigated.area$Dist.Co

  x1$OTHER.WELLS.AREA..1000.ha. <- ifelse( is.na(x1$OTHER.WELLS.AREA..1000.ha.), mean(x1$OTHER.WELLS.ARE
  test.df <- bind_rows(test.df,x1)
    #round3 - 2010:2019
  x2 <- filter(df.irrigation.sourcewise.irrigated.area, df.irrigation.sourcewise.irrigated.area$Dist.Co

  x2$OTHER.WELLS.AREA..1000.ha. <- ifelse( is.na(x2$OTHER.WELLS.AREA..1000.ha.), mean(x2$OTHER.WELLS.ARE
  test.df <- bind_rows(test.df,x2)
}
df.irrigation.sourcewise.irrigated.area <- test.df
rm(test.df)
rm(x)
rm(x1)
rm(x2)
```

Handling NA values for df.rice.farm.harvest.price

```
# Manualy adding 503,504 data


test.df1 <- data.frame(Dist.Code = as.numeric(c(504,503)),
                       Year = as.numeric(1995),
                       State.Code = as.numeric(1),
                       State.Name = "Andhra Pradesh",
                       Dist.Name = c("Prakasam","Vizianagaram"),
                       PADDY.HARVEST.PRICE..Rs.per.Quintal. = as.numeric(-1),
                       stringsAsFactors = F
                       )
df.rice.farm.harvest.price <- bind_rows(df.rice.farm.harvest.price,test.df1)

rm(test.df1)
#replace -1 by NA

df.rice.farm.harvest.price$PADDY.HARVEST.PRICE..Rs.per.Quintal. <- ifelse(df.rice.farm.harvest.price$PA

#create a test data frame with 0 rows and 6 columns
test.df <- data.frame(Dist.Code = integer(),
                       Year = integer(),
                       State.Code = integer(),
                       State.Name = character(),
                       Dist.Name = character(),
                       PADDY.HARVEST.PRICE..Rs.per.Quintal. = double(),
                       stringsAsFactors = F
                       )
```

```
dist <- c(44,45,46,47,48,49,50,51,52,53,54,503,504)
for(y in dist){
  #we only have missing value for the 1998,1995
  #round1 - 1990:1999
  x <- filter(df.rice.farm.harvest.price, df.rice.farm.harvest.price$Dist.Code == y & df.rice.farm.harv

  x$PADDY.HARVEST.PRICE..Rs.per.Quintal. <- ifelse( is.na(x$PADDY.HARVEST.PRICE..Rs.per.Quintal.), mean
  test.df <- bind_rows(test.df,x)
    #round2 - 2000:2009
  x1 <- filter(df.rice.farm.harvest.price, df.rice.farm.harvest.price$Dist.Code == y & df.rice.farm.har

  x1$PADDY.HARVEST.PRICE..Rs.per.Quintal. <- ifelse( is.na(x1$PADDY.HARVEST.PRICE..Rs.per.Quintal.), mea
  test.df <- bind_rows(test.df,x1)
    #round3 - 2010:2019
  x2 <- filter(df.rice.farm.harvest.price, df.rice.farm.harvest.price$Dist.Code == y & df.rice.farm.har

  x2$PADDY.HARVEST.PRICE..Rs.per.Quintal. <- ifelse( is.na(x2$PADDY.HARVEST.PRICE..Rs.per.Quintal.), mea
  test.df <- bind_rows(test.df,x2)
}
df.rice.farm.harvest.price <- test.df
rm(test.df)
rm(x)
rm(x1)
rm(x2)
```

Handling NA values of df.input.wages

```
#manually adding all districts data.



rowd1 <-(df.biophysical.soil.type1$Dist.Name)

dist <- c(44,45,46,47,48,49,50,51,52,53,54,503,504)

test.df1 <- data.frame(Dist.Code = as.numeric(dist),
                       Year = as.numeric(rep(c(2010,2011),times = 13)),
                       State.Code = as.numeric(1),
                       State.Name = "Andhra Pradesh",
                       Dist.Name = rowd1,
                       DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day. = as.numeric(-1),
                       DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day. = as.numeric(-1),
                       STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. = as.numeric(-1),
                       STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. = as.numeric(-1),
                       stringsAsFactors = F
                       )


df.inputs.wages <- bind_rows(df.inputs.wages,test.df1)

rm(test.df1)
#replace -1 by NA
```

```
df.inputs.wages$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day. <- ifelse(df.inputs.wages$DISTRICT.MALE.FIELD.LA

df.inputs.wages$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day. <- ifelse(df.inputs.wages$DISTRICT.FEMALE.FIEL

df.inputs.wages$STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse(df.inputs.wages$STATE.MALE.AVERA

df.inputs.wages$STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse(df.inputs.wages$STATE.FEMALE.A


#filter() from dplyr

#create a test data frame with 0 rows and 9 columns
test.df <- data.frame(Dist.Code = integer(),
                      Year = integer(),
                      State.Code = integer(),
                      State.Name = character(),
                      Dist.Name = character(),
                      DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day. = double(),
                      DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day. = double(),
                      STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. = double(),
                      STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. = double(),
                      stringsAsFactors = F
                      )


dist <- c(44,45,46,47,48,49,50,51,52,53,54,503,504)
for(y in dist){
  #round1 - Years 1990:1999
  x <- filter(df.inputs.wages, df.inputs.wages$Dist.Code == y & df.inputs.wages$Year >=1990 & df.inputs

  x$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day.)

  x$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Da

  x$STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x$STATE.MALE.AVERAGE.FIELD.LABOUR..Rs

  x$STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x$STATE.FEMALE.AVERAGE.FIELD.LABOUR

  test.df <- bind_rows(test.df,x)

  #round2 - Years 2000:2009
  x1 <- filter(df.inputs.wages, df.inputs.wages$Dist.Code == y & df.inputs.wages$Year >=2000 & df.input

  x1$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x1$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day

  x1$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x1$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per

  x1$STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x1$STATE.MALE.AVERAGE.FIELD.LABOUR..

  x1$STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x1$STATE.FEMALE.AVERAGE.FIELD.LABOU

  test.df <- bind_rows(test.df,x1)
```

```
   #round3 - Years 2010:2019
  x2 <- filter(df.inputs.wages, df.inputs.wages$Dist.Code == y & df.inputs.wages$Year >=2010 & df.inputs

  x2$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x2$DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day

  x2$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x2$DISTRICT.FEMALE.FIELD.LABOUR..Rs.per

  x2$STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x2$STATE.MALE.AVERAGE.FIELD.LABOUR..

  x2$STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. <- ifelse( is.na(x2$STATE.FEMALE.AVERAGE.FIELD.LABOU

  test.df <- bind_rows(test.df,x2)

}

df.inputs.wages <- test.df
#removing test.df from environment
rm(test.df)
rm(x)
rm(x1)
rm(x2)
```

Handling NA values of df.inputs.season.fertilizer.consumption

```
#replace -1 by NA

df.inputs.season.fertilizer.consumption$NITROGEN.KHARIF.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$NITROGEN.KHARIF.CONSUMPTION..tons. == -1,NA,df.inputs.s

df.inputs.season.fertilizer.consumption$NITROGEN.RABI.CONSUMPTION..tons.<-
  ifelse(df.inputs.season.fertilizer.consumption$NITROGEN.RABI.CONSUMPTION..tons. == -1,NA,df.inputs.sea


df.inputs.season.fertilizer.consumption$PHOSPHATE.KHARIF.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$PHOSPHATE.KHARIF.CONSUMPTION..tons. == -1,NA,df.inputs


df.inputs.season.fertilizer.consumption$PHOSPHATE.RABI.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$PHOSPHATE.RABI.CONSUMPTION..tons. == -1,NA,df.inputs.s


df.inputs.season.fertilizer.consumption$POTASH.KHARIF.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$POTASH.KHARIF.CONSUMPTION..tons. == -1,NA,df.inputs.sea


df.inputs.season.fertilizer.consumption$POTASH.RABI.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$POTASH.RABI.CONSUMPTION..tons. == -1,NA,df.inputs.seaso


df.inputs.season.fertilizer.consumption$TOTAL.KHARIF.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$TOTAL.KHARIF.CONSUMPTION..tons. == -1,NA,df.inputs.sea
```

```
df.inputs.season.fertilizer.consumption$TOTAL.RABI.CONSUMPTION..tons. <-
  ifelse(df.inputs.season.fertilizer.consumption$TOTAL.RABI.CONSUMPTION..tons. == -1,NA,df.inputs.seasor


#create a test data frame with 0 rows and 13 columns
test.df <- data.frame(Dist.Code = integer(),
                      Year = integer(),
                      State.Code = integer(),
                      State.Name = character(),
                      Dist.Name = character(),
                      NITROGEN.KHARIF.CONSUMPTION..tons. = double(),
                      NITROGEN.RABI.CONSUMPTION..tons. = double(),
                      PHOSPHATE.KHARIF.CONSUMPTION..tons. = double(),
                      PHOSPHATE.RABI.CONSUMPTION..tons. = double(),
                      POTASH.KHARIF.CONSUMPTION..tons. = double(),
                      POTASH.RABI.CONSUMPTION..tons. = double(),
                      TOTAL.KHARIF.CONSUMPTION..tons. = double(),
                      TOTAL.RABI.CONSUMPTION..tons. = double(),
                      stringsAsFactors = F
                      )

dist <- c(44,45,46,47,48,49,50,51,52,53,54,503,504)
for(y in dist){
  # here we only have missing data for 1998 & 1999
  #round1 - 1990:1999
  x <- filter(df.inputs.season.fertilizer.consumption, df.inputs.season.fertilizer.consumption$Dist.Cod

  x$NITROGEN.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x$NITROGEN.KHARIF.CONSUMPTION..tons.), mean(x$NI

  x$NITROGEN.RABI.CONSUMPTION..tons. <- ifelse( is.na(x$NITROGEN.RABI.CONSUMPTION..tons.), mean(x$NITRO

  x$PHOSPHATE.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x$PHOSPHATE.KHARIF.CONSUMPTION..tons.), mean(x$

  x$PHOSPHATE.RABI.CONSUMPTION..tons. <- ifelse( is.na(x$PHOSPHATE.RABI.CONSUMPTION..tons.), mean(x$PHOS

  x$POTASH.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x$POTASH.KHARIF.CONSUMPTION..tons.), mean(x$POTASH

  x$POTASH.RABI.CONSUMPTION..tons. <- ifelse( is.na(x$POTASH.RABI.CONSUMPTION..tons.), mean(x$POTASH.RAI

  x$TOTAL.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x$TOTAL.KHARIF.CONSUMPTION..tons.), mean(x$TOTAL.KH

    x$TOTAL.RABI.CONSUMPTION..tons. <- ifelse( is.na(x$TOTAL.RABI.CONSUMPTION..tons.), mean(x$TOTAL.RABI

  test.df <- bind_rows(test.df,x)
  #round2 - 2000:2009
   x1 <- filter(df.inputs.season.fertilizer.consumption, df.inputs.season.fertilizer.consumption$Dist.Co

  x1$NITROGEN.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x1$NITROGEN.KHARIF.CONSUMPTION..tons.), mean(x1

  x1$NITROGEN.RABI.CONSUMPTION..tons. <- ifelse( is.na(x1$NITROGEN.RABI.CONSUMPTION..tons.), mean(x1$NIT

  x1$PHOSPHATE.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x1$PHOSPHATE.KHARIF.CONSUMPTION..tons.), mean
```

```
  x1$PHOSPHATE.RABI.CONSUMPTION..tons. <- ifelse( is.na(x1$PHOSPHATE.RABI.CONSUMPTION..tons.), mean(x1$

  x1$POTASH.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x1$POTASH.KHARIF.CONSUMPTION..tons.), mean(x1$PO

  x1$POTASH.RABI.CONSUMPTION..tons. <- ifelse( is.na(x1$POTASH.RABI.CONSUMPTION..tons.), mean(x1$POTASH

  x1$TOTAL.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x1$TOTAL.KHARIF.CONSUMPTION..tons.), mean(x1$TOTA

    x1$TOTAL.RABI.CONSUMPTION..tons. <- ifelse( is.na(x1$TOTAL.RABI.CONSUMPTION..tons.), mean(x1$TOTAL.

  test.df <- bind_rows(test.df,x1)
  #round3 - 2010:2019
   x2 <- filter(df.inputs.season.fertilizer.consumption, df.inputs.season.fertilizer.consumption$Dist.C

  x2$NITROGEN.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x2$NITROGEN.KHARIF.CONSUMPTION..tons.), mean(x

  x2$NITROGEN.RABI.CONSUMPTION..tons. <- ifelse( is.na(x2$NITROGEN.RABI.CONSUMPTION..tons.), mean(x2$NI

  x2$PHOSPHATE.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x2$PHOSPHATE.KHARIF.CONSUMPTION..tons.), mean

  x2$PHOSPHATE.RABI.CONSUMPTION..tons. <- ifelse( is.na(x2$PHOSPHATE.RABI.CONSUMPTION..tons.), mean(x2$

  x2$POTASH.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x2$POTASH.KHARIF.CONSUMPTION..tons.), mean(x2$PO

  x2$POTASH.RABI.CONSUMPTION..tons. <- ifelse( is.na(x2$POTASH.RABI.CONSUMPTION..tons.), mean(x2$POTASH

  x2$TOTAL.KHARIF.CONSUMPTION..tons. <- ifelse( is.na(x2$TOTAL.KHARIF.CONSUMPTION..tons.), mean(x2$TOTA

    x2$TOTAL.RABI.CONSUMPTION..tons. <- ifelse( is.na(x2$TOTAL.RABI.CONSUMPTION..tons.), mean(x2$TOTAL.

  test.df <- bind_rows(test.df,x2)
}

df.inputs.season.fertilizer.consumption <- test.df
#removing test.df from environment
rm(test.df)
rm(x)
rm(x1)
rm(x2)
```

Merging Reductant columns

Since there are many columns in dataset, we like to reduce some redundant columns by talking average of months by general agricultural seasons in India.

Kharif - July - October

Rabi - November - April

Zaid - March - June

But now for analysis reasons, we consider Rabi to be only November to February.

```
#calculating means of columns and created 3 new columns.
#df.biophysical.monthly.rainfall
df.biophysical.monthly.rainfall$Zaid.rainfall.mm <- rowMeans(df.biophysical.monthly.rainfall[,c("MARCH.
```

```
df.biophysical.monthly.rainfall$Kharif.rainfall.mm <- rowMeans(df.biophysical.monthly.rainfall[,c("JULY
df.biophysical.monthly.rainfall$Rabi.rainfall.mm <- rowMeans(df.biophysical.monthly.rainfall[,c("NOVEMBE

#df.environment.evapotranspiration.actual
df.environment.evapotranspiration.actual$Zaid.evapotranspiration.actual.mm<-
  rowMeans(df.environment.evapotranspiration.actual[,c("MARCH.ACTUAL..Millimeters.","APRIL.ACTUAL..Milli
df.environment.evapotranspiration.actual$Kharif.evapotranspiration.actual.mm<-
  rowMeans(df.environment.evapotranspiration.actual[,c("JULY.ACTUAL..Millimeters.","AUGUST.ACTUAL..Milli
df.environment.evapotranspiration.actual$Rabi.evapotranspiration.actual.mm<-
  rowMeans(df.environment.evapotranspiration.actual[,c("NOVEMBER.ACTUAL..Millimeters.","DECEMBER.ACTUAL


#df.environment.evapotranspiration.potential
df.environment.evapotranspiration.potential$Zaid.evapotranspiration.potential.mm<-
  rowMeans(df.environment.evapotranspiration.potential[,c("MARCH.POTENTIAL..Millimeters.","APRIL.POTENTI
df.environment.evapotranspiration.potential$Kharif.evapotranspiration.potential.mm<-
  rowMeans(df.environment.evapotranspiration.potential[,c("JULY.POTENTIAL..Millimeters.","AUGUST.POTENTI
df.environment.evapotranspiration.potential$Rabi.evapotranspiration.potential.mm<-
  rowMeans(df.environment.evapotranspiration.potential[,c("NOVEMBER.POTENTIAL..Millimeters.","DECEMBER.P

#df.environment.precipitation
df.environment.precipitation$Zaid.percipitation.mm <- rowMeans( df.environment.precipitation[,c("MARCH.P
df.environment.precipitation$Kharif.percipitation.mm <- rowMeans( df.environment.precipitation[,c("JULY
df.environment.precipitation$Rabi.percipitation.mm <-rowMeans( df.environment.precipitation[,c("NOVEMBER


#df.environment.temperature.maximum

df.environment.temperature.maximum$Zaid.temp.max.c <- rowMeans(df.environment.temperature.maximum[,c("MA
df.environment.temperature.maximum$Kharif.temp.max.c <- rowMeans(df.environment.temperature.maximum[,c(
df.environment.temperature.maximum$Rabi.temp.max.c <- rowMeans(df.environment.temperature.maximum[,c("NO

#df.environment.temperature.minimum

df.environment.temperature.minimum$Zaid.temp.min.c <- rowMeans(df.environment.temperature.minimum[,c("MA
df.environment.temperature.minimum$Kharif.temp.min.c <- rowMeans(df.environment.temperature.minimum[,c(
df.environment.temperature.minimum$Rabi.temp.min.c <- rowMeans(df.environment.temperature.minimum[,c("NO
```

Deleting columns

Deleting all other columns (January to December), state name ,and district name.

```
library(dplyr)
#dropping all other columns
df.biophysical.monthly.rainfall <- select(df.biophysical.monthly.rainfall,1:3,19:21)
df.environment.evapotranspiration.actual <- select(df.environment.evapotranspiration.actual,1:3,18:20)

df.environment.evapotranspiration.potential <- select(df.environment.evapotranspiration.potential,1:3,18
df.environment.precipitation <- select(df.environment.precipitation,1:3,18:20)
df.environment.temperature.maximum <- select(df.environment.temperature.maximum,1:3,18:20)
df.environment.temperature.minimum <- select(df.environment.temperature.minimum,1:3,18:20)
```

```
df.rice.area.production.yield <- select(df.rice.area.production.yield, -c("State.Name","Dist.Name"))
df.rice.farm.harvest.price <- select(df.rice.farm.harvest.price,-c("State.Name","Dist.Name"))
df.irrigation.rice.area <- select(df.irrigation.rice.area,-c("State.Name","Dist.Name"))
df.irrigation.sourcewise.irrigated.area <- select(df.irrigation.sourcewise.irrigated.area,-c("State.Nam
df.inputs.wages <- select(df.inputs.wages,-c("State.Name","Dist.Name"))
df.inputs.season.fertilizer.consumption <- select(df.inputs.season.fertilizer.consumption,-c("State.Nam
df.inputs.fertilizer.consumption <- select(df.inputs.fertilizer.consumption,-c("State.Name","Dist.Name"

df.biophysical.landuse <- select(df.biophysical.landuse,-c("State.Name","Dist.Name"))
df.biophysical.length.of.growing.period <- select(df.biophysical.length.of.growing.period,-c("State.Nam

df.biophysical.soil.type1 <- select(df.biophysical.soil.type1,-c("State.Name","Dist.Name"))
```

Data Transformation

Merging all data frames

Now that our datasets are ready, we will try to combine all those data sets to a single DF.

We are going to use **joins from dplyr package** here. We are going to use full join because we don't want to loose any data. we will plan to handle the missing values later.

We are getting lot of rows due to cartesian product. As per discussion with Professor,k decided to move on with join on Year and concentrate on new data first.

```
combine1 <- full_join(df.rice.area.production.yield,df.rice.farm.harvest.price,by = c("Dist.Code","State
combine2 <- full_join(combine1,df.irrigation.sourcewise.irrigated.area,by = c("Dist.Code","State.Code",
combine3 <- full_join(combine2,df.irrigation.rice.area,by = c("Dist.Code","State.Code","Year"))
combine4 <- full_join(combine3,df.inputs.wages,by = c("Dist.Code","State.Code","Year"))
combine5 <- full_join(combine4,df.inputs.season.fertilizer.consumption,by = c("Dist.Code","State.Code",
combine6 <- full_join(combine5,df.inputs.fertilizer.consumption,by = c("Dist.Code","State.Code","Year"))
combine7 <- full_join(combine6,df.environment.temperature.minimum,by = c("Dist.Code","State.Code","Year
combine8 <- full_join(combine7,df.environment.temperature.maximum,by = c("Dist.Code","State.Code","Year
combine9 <- full_join(combine8,df.environment.precipitation,by = c("Dist.Code","State.Code","Year"))
combine10 <- full_join(combine9,df.environment.evapotranspiration.potential,by = c("Dist.Code","State.C
combine11 <- full_join(combine10,df.environment.evapotranspiration.actual,by = c("Dist.Code","State.Cod
combine12 <- full_join(combine11,df.biophysical.monthly.rainfall,by = c("Dist.Code","State.Code","Year"
combine13 <- full_join(combine12,df.biophysical.landuse,by = c("Dist.Code","State.Code","Year"))
combine14 <- full_join(combine13,df.biophysical.length.of.growing.period,by = c("Dist.Code","State.Code
df.all <- full_join(combine14,df.biophysical.soil.type1,by = c("Dist.Code","State.Code"))

#dropping Year and Year.y and State.Code(because we only have one state)
df.all = subset(df.all, select = -c(Year,Year.y,State.Code))
```

Data Processing Notes:

Initially, our final dataframe consists of 884 observations with 137 predictors. But by taking averages of columns season-wise, and deleting repeated columns(columns after taking averages), we reduced predictors from 137 to 80 and 780 observations. By taking data starting from years 1990 we reduced no.of observations from 780 to 364. After handling NAs,our observations got down to 338.

Maximum NAs have been produced by the only reason that 5 tables were starting from 1950's while every other table is starting from 1990.

After handling NULL values, Now after checking df.all we have 2 null values in df.rice.farm.price-paddy for 503,504(1995). and 26 NUll values in wages for all districts for years(2010,2011)

All NAs are handled.

These are not present before, they are formed during joins. need to check joins individually.

Normalization : Standard scaling method

As we can see from above dataframes, we have each column in different numerical range. Some are in range of 0-100 while some are in range or 10k - 100k. So this affects for future analysis, we will bring them to some range.

We are using regular scaling method

$z = \frac{x-\bar{x}}{s}$ to scale

$z = (x \times s) + \bar{x}$ for re-scaling back.

```
#creating a data frame to hold the values of mean and sd for every column, the same vlaues will be util

df.stats <- data.frame(
  names = character(),
  m = double(),
  s = double(),
  stringsAsFactors = F
)

# creating statistics function
stats = function(x){
  m = mean(x)
  s = sd(x)
  return(list(m,s))
}

#collecting colnames of dataFrame.
names.list <- names(df.all)

#trying to add col names as rows for new df.
for (i in names.list){
  if(i != c("Dist.Code") & i != c("Year.x")){
    st = stats(unlist(df.all[i]))
    df.stats[nrow(df.stats)+1,]<- c(i,st[[1]],st[[2]])
  }
}
```

```
#scaling function.

#standardizing
for (i in names.list){
  x = 1
  while(x <= nrow(df.stats)){
  if(i == df.stats[x,'names']){
    #print(i)
    m = df.stats[1,'m']
    s = df.stats[1,'s']
    k = unlist(df.all[i])
    z <- (k - as.numeric(m)) / as.numeric(s)
    df.all[i] <- z
  }
```

```
    x = x+1
  }
}
```

Factorising data

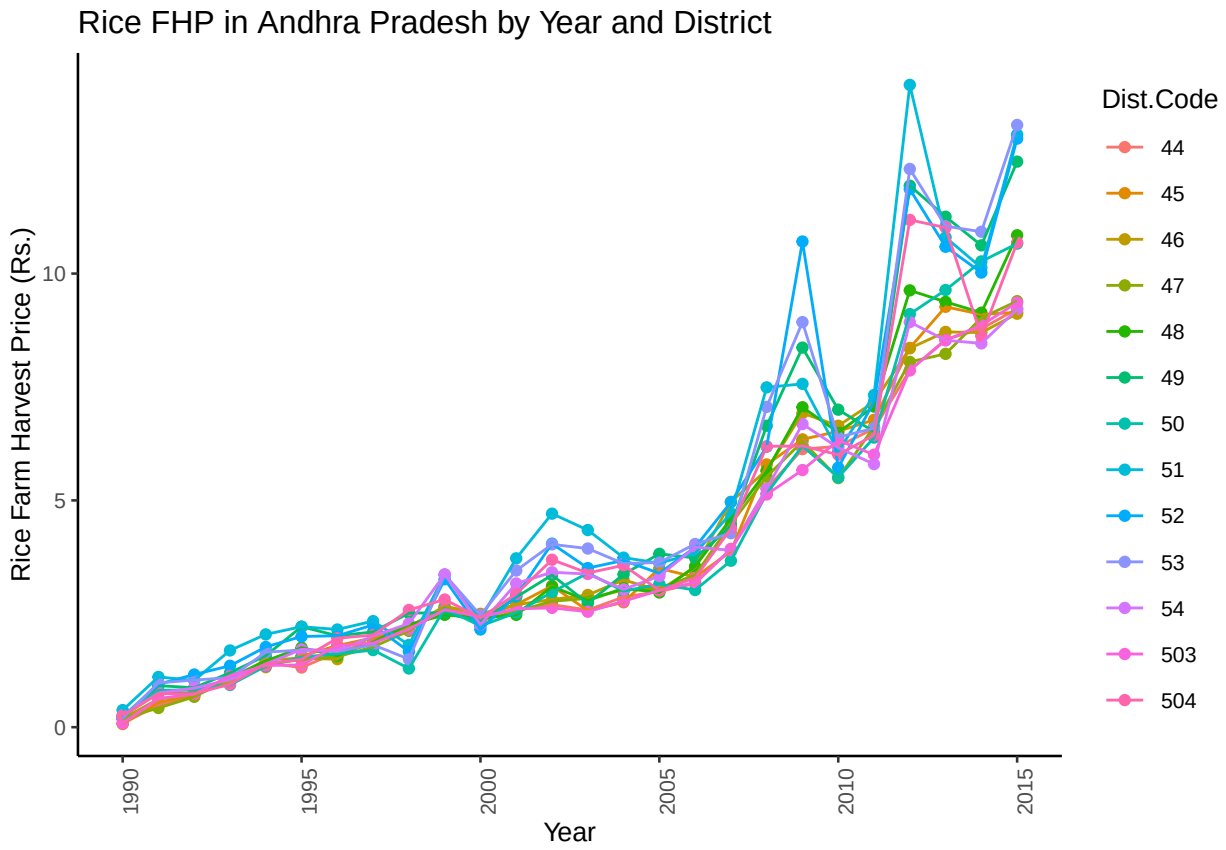
```
#factorising
df.all$Dist.Code <- as.factor(df.all$Dist.Code)
#df.all$Year.x <- as.factor(df.all$Year.x)
#df.all$State.Code <- as.factor(df.all$State.Code)
```

finding corelation: because of too many variables, we could not find this correlation to be much usefull.

```
# library(caret)
# library(corrplot)
# corMatrix <- cor(df.all)
# corrplot(corMatrix)
```

Data Visualization

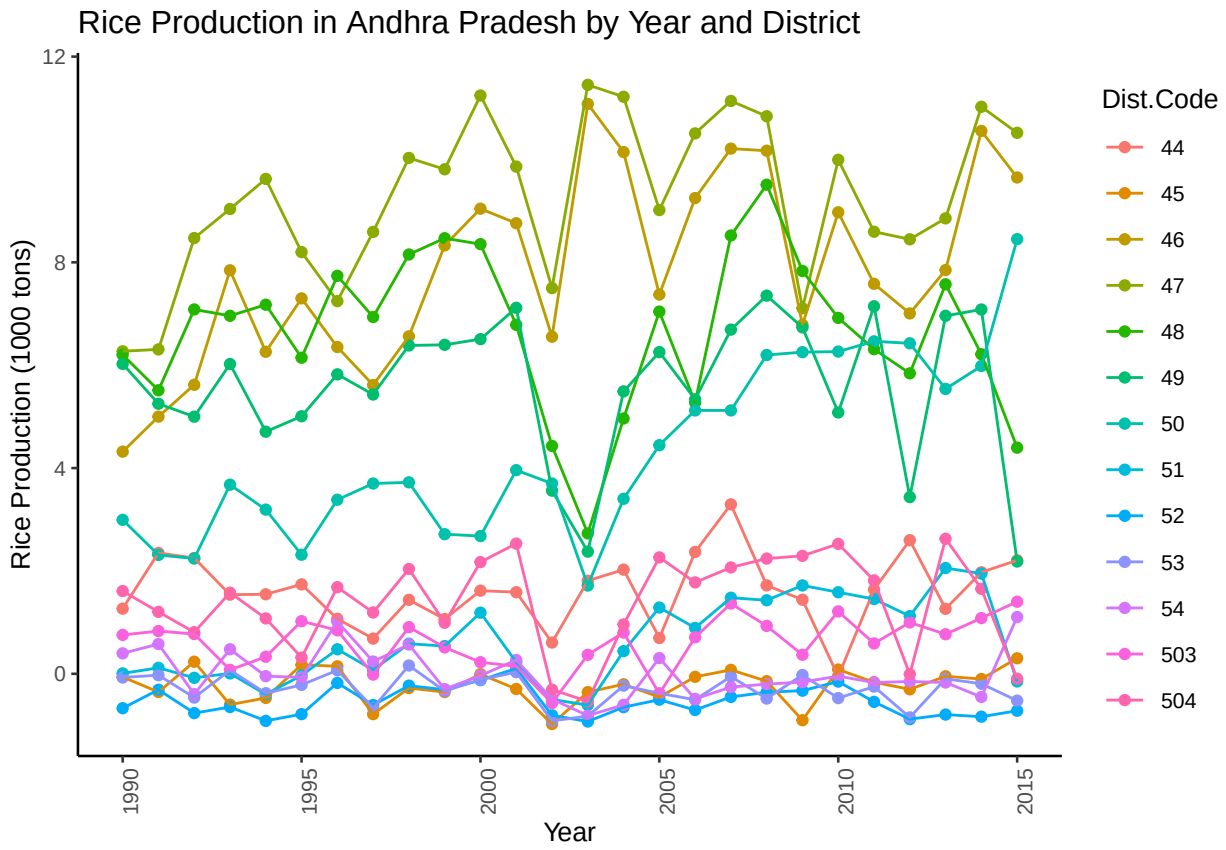
```
library(ggplot2)
library(ggsci)
ggplot(data = df.all, aes(x = Year.x, y = PADDY.HARVEST.PRICE..Rs.per.Quintal., group = Dist.Code, col =
  geom_point() +
  geom_line() +
  theme_classic(base_size = 10) +
  theme(axis.text.x = element_text(angle = 90))+
  ylab("Rice Farm Harvest Price (Rs.)") +
  xlab("Year") +
  ggtitle("Rice FHP in Andhra Pradesh by Year and District")
```



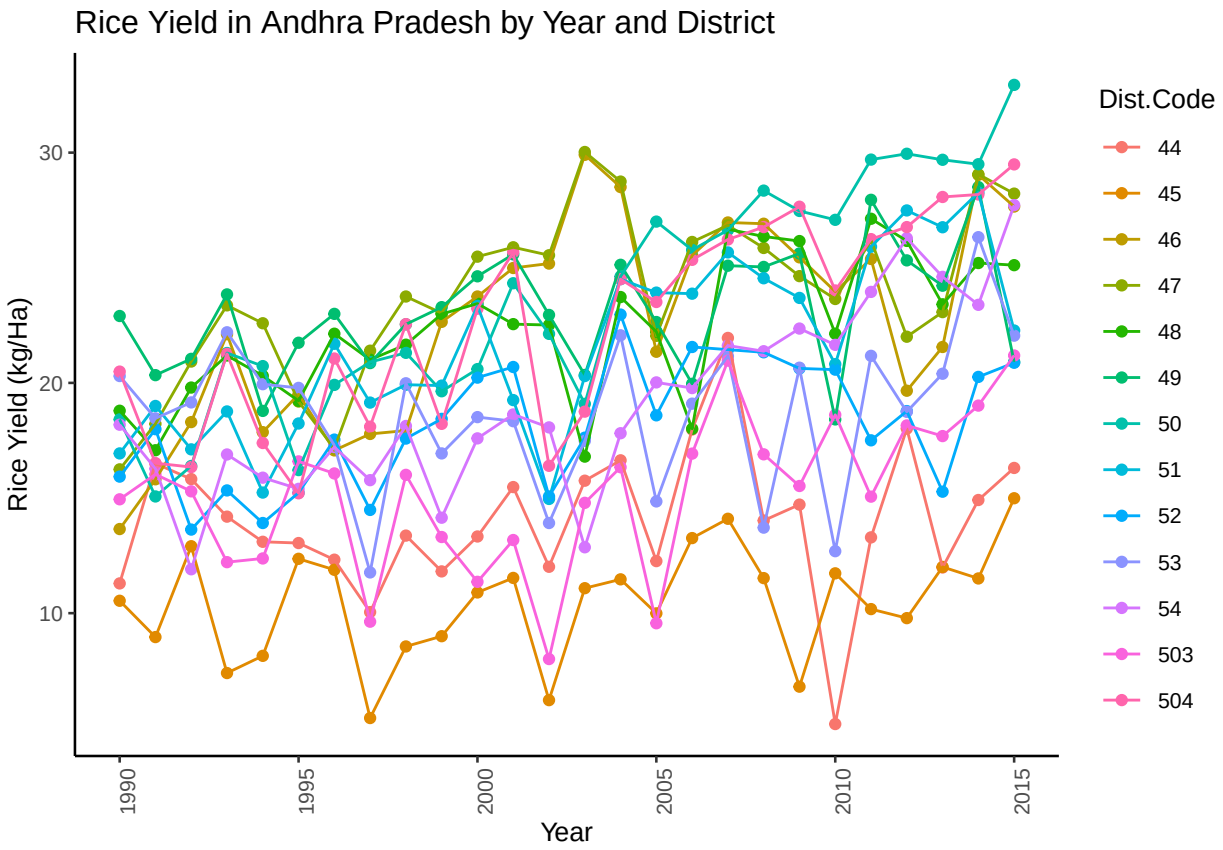
Increase in harvest price can be due to inflation. However major changes in price occurred between 2010 and 2015.

Now lets look at how production of rice has been over the years:

```
ggplot(data = df.all, aes(x = Year.x, y = RICE.PRODUCTION..1000.tons., group = Dist.Code, col = Dist.Co
  geom_point() +
  geom_line() +
  theme_classic(base_size = 10) +
  theme(axis.text.x = element_text(angle = 90))+
  ylab("Rice Production (1000 tons)") +
  xlab("Year") +
  ggtitle("Rice Production in Andhra Pradesh by Year and District")
```



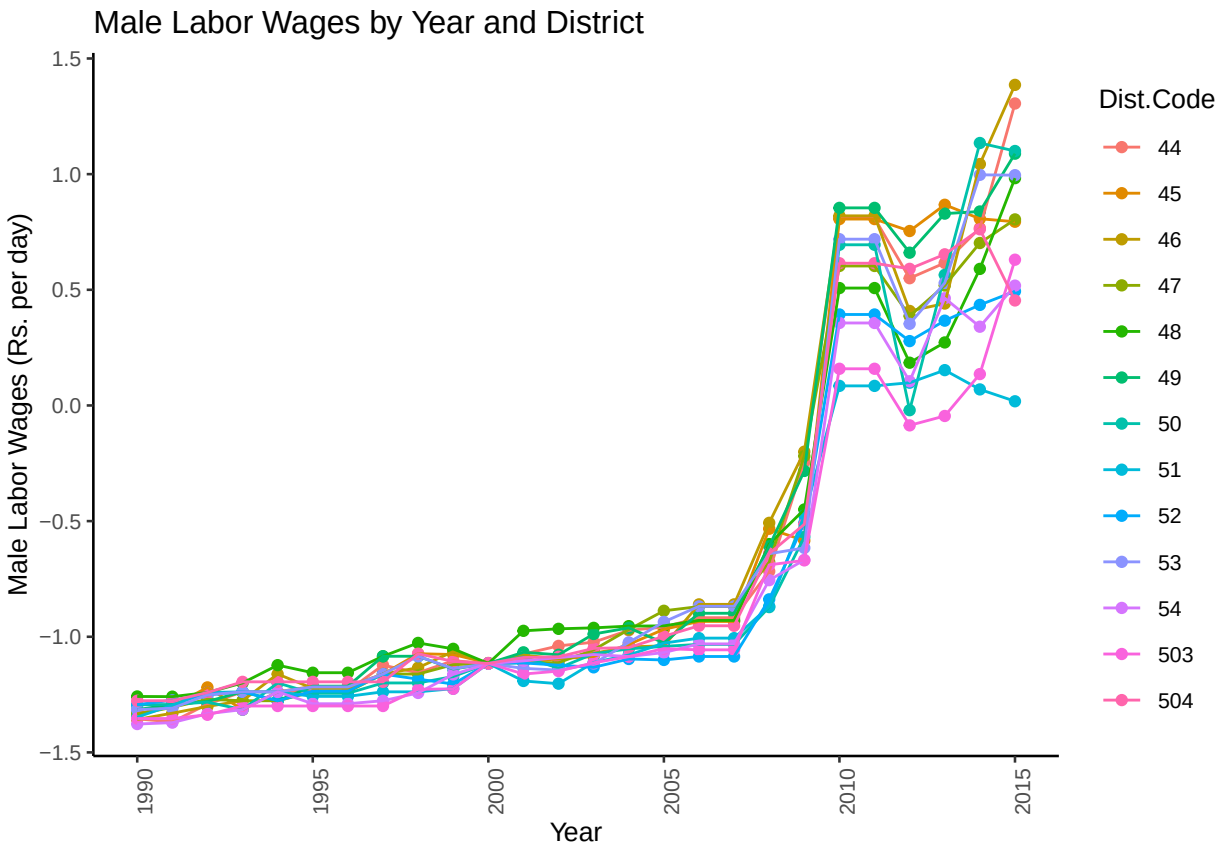

```
ggplot(data = df.all, aes(x = Year.x, y = RICE.YIELD..Kg.per.ha., group = Dist.Code, col = Dist.Code)) +
  geom_point() +
  geom_line() +
  theme_classic(base_size = 10) +
  theme(axis.text.x = element_text(angle = 90))+
  ylab("Rice Yield (kg/Ha)") +
  xlab("Year") +
  ggtitle("Rice Yield in Andhra Pradesh by Year and District")
```



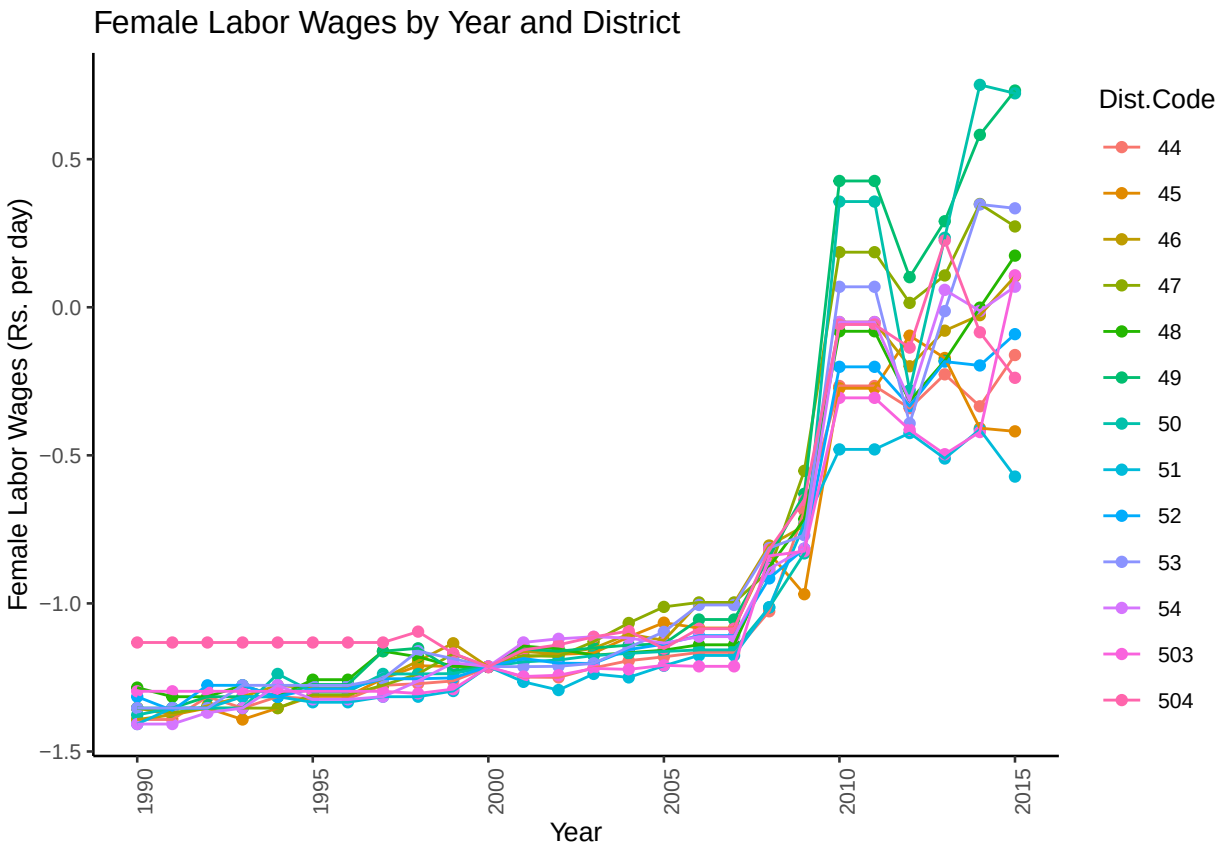
Despite some major dips across the districts in rice production during the early 2000s, the production of rice seems to be mostly the same throughout the years. This could mean two things: the increase in price of the years could be due to inflation as we had originally assumed, or there were other factors that played a more significant role in increasing the prices. Judging by the data we have at hand, most variables play a role into the production of rice. The possible predictors that can influence price greatly outside of rice production would be the wages for each worker.

Lets take a look at how the wages have changed per district across the years:

```
ggplot(data = df.all, aes(x = Year.x, y = DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day., group = Dist.Code, c
  geom_point() +
  geom_line() +
  theme_classic(base_size = 10) +
  theme(axis.text.x = element_text(angle = 90))+
  ylab("Male Labor Wages (Rs. per day)") +
  xlab("Year") +
  ggtitle("Male Labor Wages by Year and District")
```




```
ggplot(data = df.all, aes(x = Year.x, y = DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day., group = Dist.Code,
  geom_point() +
  geom_line() +
  theme_classic(base_size = 10) +
  theme(axis.text.x = element_text(angle = 90))+
  ylab("Female Labor Wages (Rs. per day)") +
  xlab("Year") +
  ggtitle("Female Labor Wages by Year and District")
```



As we can see from the visuals, the wages for the workers have increased steadily until 2007. From 2008-2015 the wages have increased rapidly. This reflects what we saw with the price of harvest as well. This makes intuitive sense because as the cost of workers go up, the prices the farmers charge will reflect that. This further supports our initial assumption of inflation playing a key factor in the increase of rice harvest prices. Now that we have a general idea of the data, we can go into the statistics of the data and see what features play a role in price that we cannot see so easily.

Model building

Train,Test Split

```
#data splitting.

#Here we are splitting the data to train and test where we took data from 1990 to 2010 as training data

#So, technically we are predicting future results.

data.train <- df.all[df.all$Year.x <= 2010,]
data.test <- df.all[df.all$Year.x > 2010,]
```

General Models - Linear Regression

```
library(olsrr)
```

```
##
## Attaching package: 'olsrr'
```

```
## The following object is masked from 'package:datasets':
##
##     rivers
```

```
set.seed(108)

#all predictors model
model.all <- lm(PADDY.HARVEST.PRICE..Rs.per.Quintal.~., data = data.train)
summary(model.all)
```

```
##
## Call:
## lm(formula = PADDY.HARVEST.PRICE..Rs.per.Quintal. ~ ., data = data.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.40458 -0.25136 -0.02099  0.27222  2.02109
##
## Coefficients: (8 not defined because of singularities)
##                                                Estimate Std. Error t value
## (Intercept)                                   2.438e+03  3.390e+03   0.719
## Dist.Code45                                   1.358e+00  3.706e+00   0.366
## Dist.Code46                                  -1.508e+00  3.189e+00  -0.473
## Dist.Code47                                  -1.305e+00  1.623e+00  -0.804
## Dist.Code48                                  -2.674e+00  2.147e+00  -1.246
## Dist.Code49                                  -3.599e+00  3.668e+00  -0.981
## Dist.Code50                                  -4.580e+00  4.402e+00  -1.040
## Dist.Code51                                  -5.174e+00  7.130e+00  -0.726
## Dist.Code52                                  -5.411e+00  8.164e+00  -0.663
## Dist.Code53                                  -2.424e+00  5.611e+00  -0.432
## Dist.Code54                                  -3.327e+00  5.457e+00  -0.610
## Dist.Code503                                  5.857e-01  6.214e-01   0.943
## Dist.Code504                                 -4.035e+00  6.834e+00  -0.590
## Year.x                                        2.705e-01  2.816e-02   9.604
## RICE.AREA..1000.ha.                           4.149e-01  2.025e+00   0.205
## RICE.PRODUCTION..1000.tons.                   4.370e-02  1.132e-01   0.386
## RICE.YIELD..Kg.per.ha.                        5.696e-04  2.585e-02   0.022
## CANALS.AREA..1000.ha.                        -1.713e+01  2.535e+01  -0.676
## TANKS.AREA..1000.ha.                         -1.546e+01  2.522e+01  -0.613
## TUBE.WELLS.AREA..1000.ha.                    -6.875e-01  1.972e+00  -0.349
## OTHER.WELLS.AREA..1000.ha.                   -4.983e-01  1.997e+00  -0.250
## TOTAL.WELLS.AREA..1000.ha.                   -1.623e+01  2.507e+01  -0.648
## OTHER.SOURCES.AREA..1000.ha.                 -1.801e+01  2.578e+01  -0.699
## NET.AREA..1000.ha.                            1.748e+01  2.534e+01   0.690
## GROSS.AREA..1000.ha.                         -6.234e-01  6.132e-01  -1.017
## RICE.IRRIGATED.AREA..1000.ha.                -4.962e-01  1.989e+00  -0.249
## DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day.       7.913e-01  6.812e-01   1.162
## DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day.    -6.639e-01  7.078e-01  -0.938
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.  3.502e-01  5.884e-01   0.595
## STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.        NA         NA      NA
## NITROGEN.KHARIF.CONSUMPTION..tons.            1.384e+00  3.594e+00   0.385
## NITROGEN.RABI.CONSUMPTION..tons.              6.694e-04  2.440e-03   0.274
## PHOSPHATE.KHARIF.CONSUMPTION..tons.           1.392e+00  3.594e+00   0.387
## PHOSPHATE.RABI.CONSUMPTION..tons.            -5.428e-03  3.532e-03  -1.537
## POTASH.KHARIF.CONSUMPTION..tons.              1.392e+00  3.594e+00   0.387
```

```
## POTASH.RABI.CONSUMPTION..tons.                    5.104e-03  4.677e-03   1.091
## TOTAL.KHARIF.CONSUMPTION..tons.                  -1.387e+00  3.594e+00  -0.386
## TOTAL.RABI.CONSUMPTION..tons.                           NA         NA      NA
## NITROGEN.CONSUMPTION..tons.                       9.811e+01  5.669e+01   1.731
## NITROGEN.SHARE.IN.NPK..Percent.                   1.268e+00  1.690e+00   0.750
## NITROGEN.PER.HA.OF.NCA..Kg.per.ha.                5.344e+02  7.719e+02   0.692
## NITROGEN.PER.HA.OF.GCA..Kg.per.ha.                4.028e+02  8.383e+02   0.481
## PHOSPHATE.CONSUMPTION..tons.                      9.811e+01  5.669e+01   1.731
## PHOSPHATE.SHARE.IN.NPK..Percent.                  3.547e+00  4.341e+00   0.817
## PHOSPHATE.PER.HA.OF.NCA..Kg.per.ha.               5.323e+02  7.715e+02   0.690
## PHOSPHATE.PER.HA.OF.GCA..Kg.per.ha.               4.074e+02  8.388e+02   0.486
## POTASH.CONSUMPTION..tons.                         9.813e+01  5.669e+01   1.731
## POTASH.SHARE.IN.NPK..Percent.                    -1.524e+00  4.202e+00  -0.363
## POTASH.PER.HA.OF.NCA..Kg.per.ha.                  5.099e+02  7.721e+02   0.660
## POTASH.PER.HA.OF.GCA..Kg.per.ha.                  4.292e+02  8.386e+02   0.512
## TOTAL.CONSUMPTION..tons.                         -9.811e+01  5.669e+01  -1.731
## TOTAL.PER.HA.OF.NCA..Kg.per.ha.                  -5.305e+02  7.718e+02  -0.687
## TOTAL.PER.HA.OF.GCA..Kg.per.ha.                  -4.076e+02  8.384e+02  -0.486
## Zaid.temp.min.c                                  -9.697e+00  3.009e+01  -0.322
## Kharif.temp.min.c                                 1.381e+01  3.270e+01   0.422
## Rabi.temp.min.c                                   1.097e+00  2.417e+01   0.045
## Zaid.temp.max.c                                  -7.070e+01  2.617e+01  -2.702
## Kharif.temp.max.c                                 2.192e+01  3.212e+01   0.682
## Rabi.temp.max.c                                   3.271e+01  2.612e+01   1.252
## Zaid.percipitation.mm                            -6.953e-01  5.902e-01  -1.178
## Kharif.percipitation.mm                          -5.287e-01  1.946e-01  -2.717
## Rabi.percipitation.mm                            -7.205e-01  5.780e-01  -1.247
## Zaid.evapotranspiration.potential.mm              6.752e+00  2.364e+00   2.856
## Kharif.evapotranspiration.potential.mm           -9.466e-01  3.057e+00  -0.310
## Rabi.evapotranspiration.potential.mm              1.349e+01  4.013e+00   3.362
## Zaid.evapotranspiration.actual.mm                 2.553e-02  7.899e-01   0.032
## Kharif.evapotranspiration.actual.mm              -1.369e+00  7.758e-01  -1.765
## Rabi.evapotranspiration.actual.mm                 1.015e+00  8.945e-01   1.135
## Zaid.rainfall.mm                                 -5.829e-01  2.402e-01  -2.427
## Kharif.rainfall.mm                               -1.505e-01  2.280e-01  -0.660
## Rabi.rainfall.mm                                  1.086e+00  3.686e-01   2.946
## TOTAL.AREA..1000.ha.                              8.481e-02  1.069e+00   0.079
## FOREST.AREA..1000.ha.                             2.060e-01  1.206e+00   0.171
## BARREN.AND.UNCULTIVABLE.LAND.AREA..1000.ha.      -7.939e-01  1.260e+00  -0.630
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha.    1.872e-01  8.888e-01   0.211
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.             -1.068e-01  1.038e+00  -0.103
## PERMANENT.PASTURES.AREA..1000.ha.                 1.459e+00  1.991e+00   0.733
## OTHER.FALLOW.AREA..1000.ha.                       8.348e-01  7.522e-01   1.110
## CURRENT.FALLOW.AREA..1000.ha.                     9.095e-01  6.893e-01   1.319
## NET.CROPPED.AREA..1000.ha.                       -2.355e+00  1.367e+00  -1.722
## GROSS.CROPPED.AREA..1000.ha.                      2.391e+00  1.005e+00   2.378
## CROPING.INTENSITY..Percent.                      -7.424e+00  4.528e+00  -1.640
## LENGTH.OF.GROWING.PERIOD.DAYS..Number.                  NA         NA      NA
## Soil.Type.Orthids                                       NA         NA      NA
## Soil.Type.Sandy.Alfisol                                 NA         NA      NA
## Soil.Type.Loamy.Alfisols                                NA         NA      NA
## Soil.Type.Vertisols                                     NA         NA      NA
## Soil.Type.Vertic.soils                                  NA         NA      NA
##                                                 Pr(>|t|)
```

```
## (Intercept)                                    0.472982
## Dist.Code45                                    0.714456
## Dist.Code46                                    0.636785
## Dist.Code47                                    0.422426
## Dist.Code48                                    0.214451
## Dist.Code49                                    0.327719
## Dist.Code50                                    0.299457
## Dist.Code51                                    0.468890
## Dist.Code52                                    0.508226
## Dist.Code53                                    0.666251
## Dist.Code54                                    0.542795
## Dist.Code503                                   0.347047
## Dist.Code504                                   0.555586
## Year.x                                          < 2e-16 ***
## RICE.AREA..1000.ha.                            0.837853
## RICE.PRODUCTION..1000.tons.                    0.699768
## RICE.YIELD..Kg.per.ha.                         0.982446
## CANALS.AREA..1000.ha.                          0.499933
## TANKS.AREA..1000.ha.                           0.540448
## TUBE.WELLS.AREA..1000.ha.                      0.727724
## OTHER.WELLS.AREA..1000.ha.                     0.803226
## TOTAL.WELLS.AREA..1000.ha.                     0.517996
## OTHER.SOURCES.AREA..1000.ha.                   0.485499
## NET.AREA..1000.ha.                             0.491117
## GROSS.AREA..1000.ha.                           0.310539
## RICE.IRRIGATED.AREA..1000.ha.                  0.803305
## DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day.        0.246853
## DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day.      0.349425
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.   0.552394
## STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.       NA
## NITROGEN.KHARIF.CONSUMPTION..tons.             0.700578
## NITROGEN.RABI.CONSUMPTION..tons.               0.784119
## PHOSPHATE.KHARIF.CONSUMPTION..tons.            0.698940
## PHOSPHATE.RABI.CONSUMPTION..tons.              0.125928
## POTASH.KHARIF.CONSUMPTION..tons.               0.699040
## POTASH.RABI.CONSUMPTION..tons.                 0.276573
## TOTAL.KHARIF.CONSUMPTION..tons.                0.699938
## TOTAL.RABI.CONSUMPTION..tons.                        NA
## NITROGEN.CONSUMPTION..tons.                    0.085106 .
## NITROGEN.SHARE.IN.NPK..Percent.                0.453908
## NITROGEN.PER.HA.OF.NCA..Kg.per.ha.             0.489587
## NITROGEN.PER.HA.OF.GCA..Kg.per.ha.             0.631412
## PHOSPHATE.CONSUMPTION..tons.                   0.085117 .
## PHOSPHATE.SHARE.IN.NPK..Percent.               0.414870
## PHOSPHATE.PER.HA.OF.NCA..Kg.per.ha.            0.491075
## PHOSPHATE.PER.HA.OF.GCA..Kg.per.ha.            0.627764
## POTASH.CONSUMPTION..tons.                      0.085073 .
## POTASH.SHARE.IN.NPK..Percent.                  0.717301
## POTASH.PER.HA.OF.NCA..Kg.per.ha.               0.509744
## POTASH.PER.HA.OF.GCA..Kg.per.ha.               0.609383
## TOTAL.CONSUMPTION..tons.                       0.085108 .
## TOTAL.PER.HA.OF.NCA..Kg.per.ha.                0.492719
## TOTAL.PER.HA.OF.GCA..Kg.per.ha.                0.627444
## Zaid.temp.min.c                                0.747631
```

```
## Kharif.temp.min.c                                0.673245
## Rabi.temp.min.c                                  0.963833
## Zaid.temp.max.c                                  0.007510 **
## Kharif.temp.max.c                                0.495864
## Rabi.temp.max.c                                  0.211944
## Zaid.percipitation.mm                            0.240215
## Kharif.percipitation.mm                          0.007192 **
## Rabi.percipitation.mm                            0.214076
## Zaid.evapotranspiration.potential.mm             0.004753 **
## Kharif.evapotranspiration.potential.mm           0.757167
## Rabi.evapotranspiration.potential.mm             0.000932 ***
## Zaid.evapotranspiration.actual.mm                0.974250
## Kharif.evapotranspiration.actual.mm              0.079108 .
## Rabi.evapotranspiration.actual.mm                0.257851
## Zaid.rainfall.mm                                 0.016136 *
## Kharif.rainfall.mm                               0.510006
## Rabi.rainfall.mm                                 0.003614 **
## TOTAL.AREA..1000.ha.                             0.936835
## FOREST.AREA..1000.ha.                            0.864546
## BARREN.AND.UNCULTIVABLE.LAND.AREA..1000.ha.      0.529572
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha. 0.833423
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.             0.918158
## PERMANENT.PASTURES.AREA..1000.ha.                0.464612
## OTHER.FALLOW.AREA..1000.ha.                      0.268405
## CURRENT.FALLOW.AREA..1000.ha.                    0.188602
## NET.CROPPED.AREA..1000.ha.                       0.086618 .
## GROSS.CROPPED.AREA..1000.ha.                     0.018364 *
## CROPING.INTENSITY..Percent.                      0.102731
## LENGTH.OF.GROWING.PERIOD.DAYS..Number.                NA
## Soil.Type.Orthids                                     NA
## Soil.Type.Sandy.Alfisol                               NA
## Soil.Type.Loamy.Alfisols                              NA
## Soil.Type.Vertisols                                   NA
## Soil.Type.Vertic.soils                                NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4909 on 193 degrees of freedom
## Multiple R-squared:  0.9498, Adjusted R-squared:  0.9293
## F-statistic: 46.23 on 79 and 193 DF,  p-value: < 2.2e-16
```

Our model.all with all predictor variables gave an Adjusted r-squared value of 0.9293.

AIC model

Model selection/variable selection.

We will use AIC and forward prediction because AIC will helps us to prevent overfitting.

Here we are using forward selection process to select the predictors.

```
#forward selection
fwdfit.aic <- ols_step_forward_aic(model.all)
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length

## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
#summary of model
fwdfit.aic
```

```
##
##                                          Selection Summary
## ------------------------------------------------------------------------------------------
## Variable                                        AIC      Sum Sq       RSS      R-Sq      Adj. R-
## ------------------------------------------------------------------------------------------
## Year.x                                      640.037    763.425   163.053   0.82401       0.823
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day. 592.788   790.340   136.139   0.85306       0.851
## Kharif.percipitation.mm                     548.044    811.763   114.715   0.87618       0.874
## PERMANENT.PASTURES.AREA..1000.ha.           534.235    818.218   108.261   0.88315       0.881
## TOTAL.KHARIF.CONSUMPTION..tons.             525.246    822.489   103.990   0.88776       0.885
## CURRENT.FALLOW.AREA..1000.ha.               515.619    826.825    99.654   0.89244       0.890
## NET.CROPPED.AREA..1000.ha.                  506.745    830.716    95.762   0.89664       0.893
## Kharif.evapotranspiration.actual.mm         501.405    833.257    93.222   0.89938       0.896
## Rabi.rainfall.mm                            499.017    834.743    91.735   0.90098       0.897
## OTHER.WELLS.AREA..1000.ha.                  495.324    836.636    89.842   0.90303       0.899
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha. 493.804 837.787    88.691   0.90427       0.900
## Rabi.evapotranspiration.potential.mm        481.879    842.198    84.281   0.90903       0.904
## Dist.Code                                   450.985    857.549    68.929   0.92560       0.918
```

```
## NITROGEN.KHARIF.CONSUMPTION..tons.                444.920    859.556     66.923    0.92777      0.920
## Zaid.rainfall.mm                                  441.380    860.900     65.578    0.92922      0.921
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.              439.266    861.881     64.597    0.93028      0.923
## Zaid.temp.max.c                                   436.999    862.883     63.596    0.93136      0.923
## TANKS.AREA..1000.ha.                              436.029    863.571     62.907    0.93210      0.924
## Rabi.percipitation.mm                             435.113    864.239     62.239    0.93282      0.924
## Rabi.evapotranspiration.actual.mm                 431.318    865.547     60.932    0.93423      0.925
## Rabi.temp.max.c                                   430.942    866.075     60.404    0.93480      0.926
## Zaid.evapotranspiration.potential.mm              430.759    866.556     59.923    0.93532      0.926
## Zaid.percipitation.mm                             430.267    867.100     59.378    0.93591      0.926
## ------------------------------------------------------------------------------------------------------
```

```
#final model
fwdfit.aic$model
```

```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Coefficients:
##                                     (Intercept)
##                                      -5.006e+02
##                                          Year.x
##                                       2.442e-01
##   STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.
##                                       7.399e-01
##                           Kharif.percipitation.mm
##                                      -6.076e-01
##              PERMANENT.PASTURES.AREA..1000.ha.
##                                      -1.503e+00
##                TOTAL.KHARIF.CONSUMPTION..tons.
##                                       4.888e-03
##                  CURRENT.FALLOW.AREA..1000.ha.
##                                       1.006e+00
##                     NET.CROPPED.AREA..1000.ha.
##                                       4.471e-01
##            Kharif.evapotranspiration.actual.mm
##                                      -2.126e+00
##                                Rabi.rainfall.mm
##                                       1.208e+00
##                     OTHER.WELLS.AREA..1000.ha.
##                                      -1.018e+00
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha.
##                                       2.627e-01
##           Rabi.evapotranspiration.potential.mm
##                                       1.500e+01
##                                     Dist.Code45
##                                       1.184e+00
##                                     Dist.Code46
##                                       1.050e-01
##                                     Dist.Code47
##                                      -4.044e-01
##                                     Dist.Code48
##                                      -1.257e+00
```

```
##                                    Dist.Code49
##                                     -1.302e+00
##                                    Dist.Code50
##                                     -5.600e-01
##                                    Dist.Code51
##                                     -3.631e+00
##                                    Dist.Code52
##                                     -4.980e+00
##                                    Dist.Code53
##                                     -7.207e-01
##                                    Dist.Code54
##                                     -8.936e-01
##                                   Dist.Code503
##                                      6.343e-01
##                                   Dist.Code504
##                                     -9.969e-01
##             NITROGEN.KHARIF.CONSUMPTION..tons.
##                                     -5.256e-03
##                                Zaid.rainfall.mm
##                                     -6.408e-01
##           CULTIVABLE.WASTE.LAND.AREA..1000.ha.
##                                     -1.306e+00
##                                  Zaid.temp.max.c
##                                     -5.064e+01
##                             TANKS.AREA..1000.ha.
##                                      1.179e+00
##                            Rabi.percipitation.mm
##                                     -1.213e+00
##               Rabi.evapotranspiration.actual.mm
##                                      1.736e+00
##                                  Rabi.temp.max.c
##                                      3.098e+01
##           Zaid.evapotranspiration.potential.mm
##                                      3.102e+00
##                            Zaid.percipitation.mm
##                                     -4.461e-01
```

```
summary(fwdfit.aic$model)
```

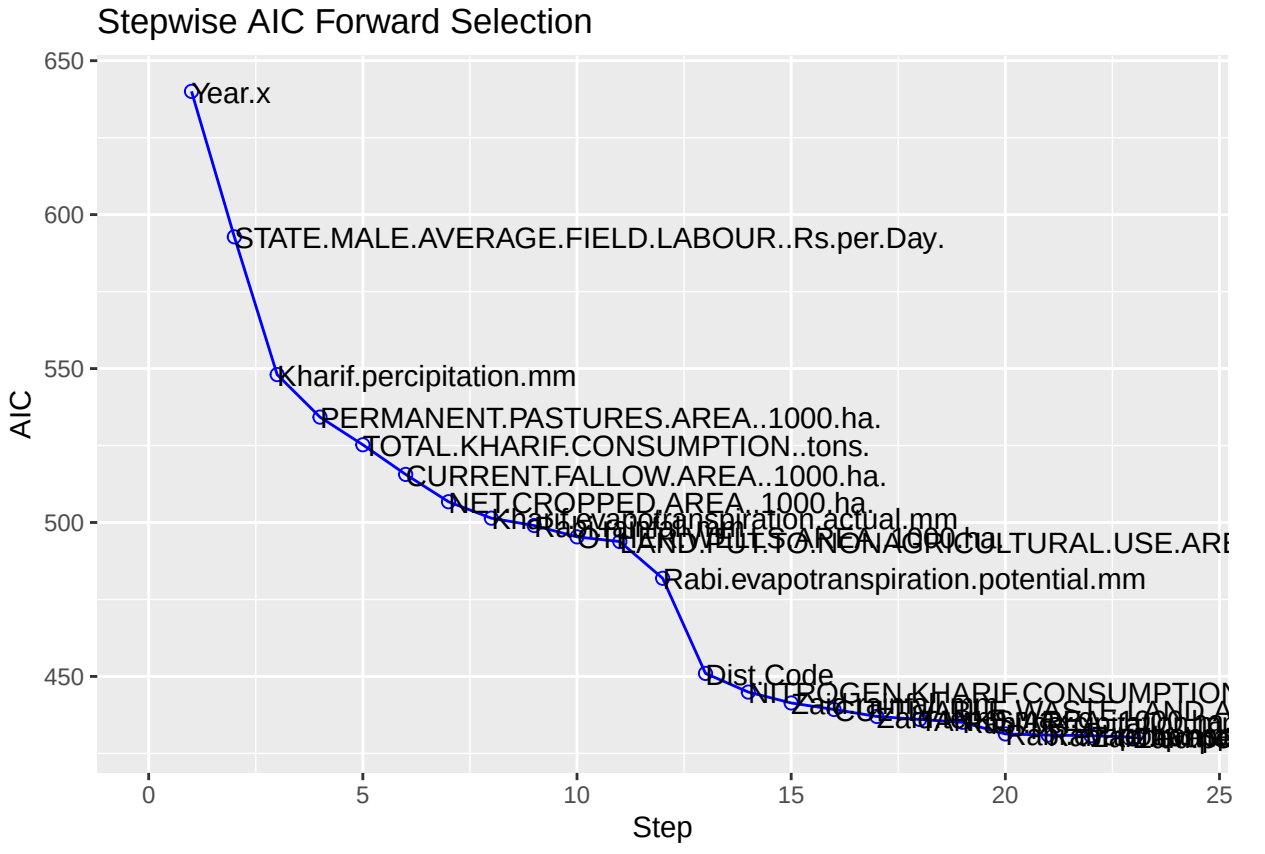
```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47299 -0.26385 -0.00331  0.23742  2.79588
##
## Coefficients:
##                                                Estimate Std. Error t value
## (Intercept)                                  -5.006e+02  4.078e+01 -12.276
## Year.x                                        2.442e-01  1.400e-02  17.446
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.  7.399e-01  1.668e-01   4.435
## Kharif.percipitation.mm                      -6.076e-01  1.270e-01  -4.784
## PERMANENT.PASTURES.AREA..1000.ha.            -1.503e+00  1.335e+00  -1.126
```

```
## TOTAL.KHARIF.CONSUMPTION..tons.                  4.888e-03  1.173e-03   4.167
## CURRENT.FALLOW.AREA..1000.ha.                    1.006e+00  3.085e-01   3.262
## NET.CROPPED.AREA..1000.ha.                       4.471e-01  3.307e-01   1.352
## Kharif.evapotranspiration.actual.mm             -2.126e+00  4.944e-01  -4.301
## Rabi.rainfall.mm                                 1.208e+00  3.200e-01   3.775
## OTHER.WELLS.AREA..1000.ha.                      -1.018e+00  4.321e-01  -2.357
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha.   2.627e-01  5.080e-01   0.517
## Rabi.evapotranspiration.potential.mm             1.500e+01  3.139e+00   4.777
## Dist.Code45                                      1.184e+00  2.945e-01   4.019
## Dist.Code46                                      1.050e-01  6.736e-01   0.156
## Dist.Code47                                     -4.044e-01  6.694e-01  -0.604
## Dist.Code48                                     -1.257e+00  8.604e-01  -1.461
## Dist.Code49                                     -1.302e+00  1.339e+00  -0.973
## Dist.Code50                                     -5.600e-01  1.596e+00  -0.351
## Dist.Code51                                     -3.631e+00  2.070e+00  -1.754
## Dist.Code52                                     -4.980e+00  2.591e+00  -1.922
## Dist.Code53                                     -7.207e-01  9.586e-01  -0.752
## Dist.Code54                                     -8.936e-01  1.005e+00  -0.889
## Dist.Code503                                     6.343e-01  2.078e-01   3.052
## Dist.Code504                                    -9.969e-01  1.739e+00  -0.573
## NITROGEN.KHARIF.CONSUMPTION..tons.              -5.256e-03  2.141e-03  -2.455
## Zaid.rainfall.mm                                -6.408e-01  2.021e-01  -3.170
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.            -1.306e+00  7.069e-01  -1.848
## Zaid.temp.max.c                                 -5.064e+01  1.486e+01  -3.408
## TANKS.AREA..1000.ha.                             1.179e+00  6.472e-01   1.822
## Rabi.percipitation.mm                           -1.213e+00  4.914e-01  -2.468
## Rabi.evapotranspiration.actual.mm                1.736e+00  7.082e-01   2.451
## Rabi.temp.max.c                                  3.098e+01  1.615e+01   1.919
## Zaid.evapotranspiration.potential.mm             3.102e+00  1.953e+00   1.588
## Zaid.percipitation.mm                           -4.461e-01  3.020e-01  -1.477
##                                                 Pr(>|t|)
## (Intercept)                                      < 2e-16 ***
## Year.x                                           < 2e-16 ***
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.    1.41e-05 ***
## Kharif.percipitation.mm                         3.02e-06 ***
## PERMANENT.PASTURES.AREA..1000.ha.               0.261363
## TOTAL.KHARIF.CONSUMPTION..tons.                 4.31e-05 ***
## CURRENT.FALLOW.AREA..1000.ha.                   0.001269 **
## NET.CROPPED.AREA..1000.ha.                      0.177629
## Kharif.evapotranspiration.actual.mm             2.48e-05 ***
## Rabi.rainfall.mm                                0.000202 ***
## OTHER.WELLS.AREA..1000.ha.                      0.019252 *
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha. 0.605483
## Rabi.evapotranspiration.potential.mm            3.11e-06 ***
## Dist.Code45                                     7.85e-05 ***
## Dist.Code46                                     0.876278
## Dist.Code47                                     0.546307
## Dist.Code48                                     0.145228
## Dist.Code49                                     0.331664
## Dist.Code50                                     0.725912
## Dist.Code51                                     0.080634 .
## Dist.Code52                                     0.055833 .
## Dist.Code53                                     0.452867
## Dist.Code54                                     0.374724
```

```
## Dist.Code503                                    0.002529 **
## Dist.Code504                                    0.567025
## NITROGEN.KHARIF.CONSUMPTION..tons.              0.014803 *
## Zaid.rainfall.mm                                0.001723 **
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.            0.065865 .
## Zaid.temp.max.c                                 0.000769 ***
## TANKS.AREA..1000.ha.                            0.069745 .
## Rabi.percipitation.mm                           0.014293 *
## Rabi.evapotranspiration.actual.mm               0.014967 *
## Rabi.temp.max.c                                 0.056231 .
## Zaid.evapotranspiration.potential.mm            0.113614
## Zaid.percipitation.mm                           0.140936
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4995 on 238 degrees of freedom
## Multiple R-squared:  0.9359, Adjusted R-squared:  0.9268
## F-statistic: 102.2 on 34 and 238 DF,  p-value: < 2.2e-16
```

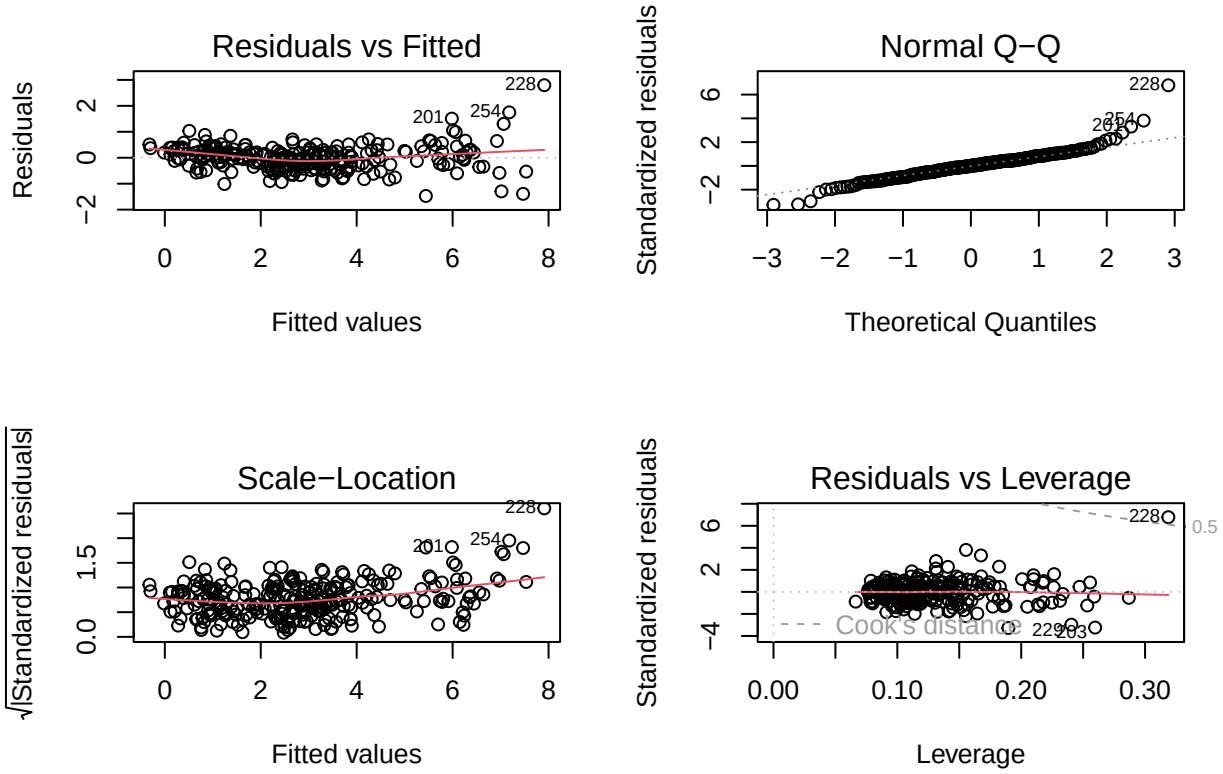
Our fwdfit.aic model gave Adjusted R-squared values of 0.9765 with only 23 predictors which indicates a very good model compared to model.all

```
#AIC model plot

plot(fwdfit.aic)
```

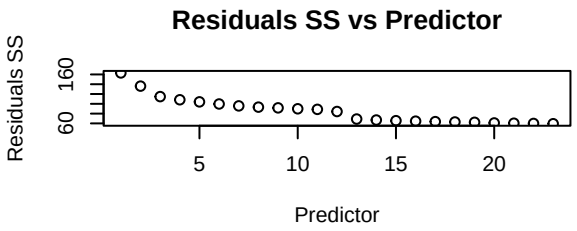


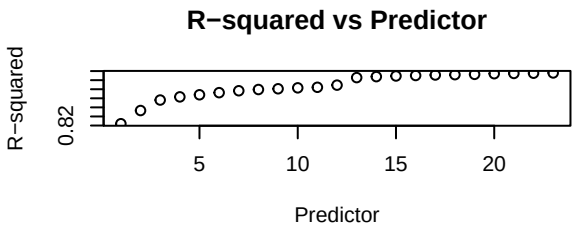

```
#plotting final model characteristics
par(mfrow=c(2,2))
```

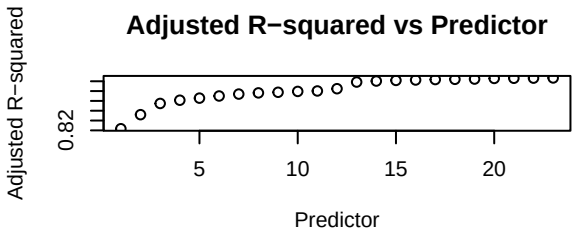
```
plot(fwdfit.aic$model)
```

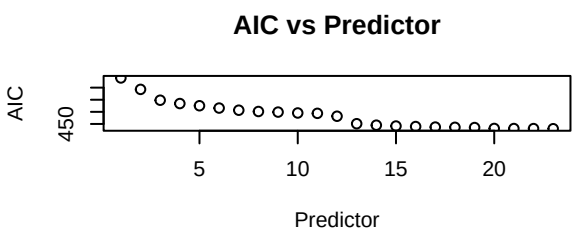


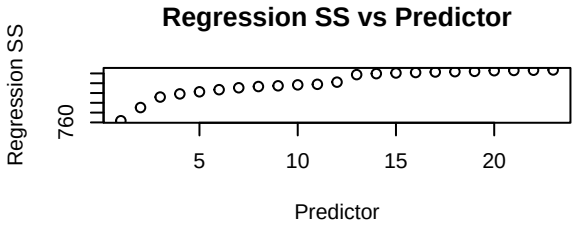

```
#plotting final model residuals.
par(mfrow=c(3,2))
plot(fwdfit.aic$ess,main = "Residuals SS vs Predictor",xlab = "Predictor",ylab = "Residuals SS")
plot(fwdfit.aic$rsq,main = "R-squared vs Predictor",xlab = "Predictor",ylab = "R-squared")
plot(fwdfit.aic$arsq,main = "Adjusted R-squared vs Predictor",xlab = "Predictor",ylab = "Adjusted R-squa
plot(fwdfit.aic$aics,main = "AIC vs Predictor",xlab = "Predictor",ylab = "AIC")
plot(fwdfit.aic$rss,main = "Regression SS vs Predictor",xlab = "Predictor",ylab = "Regression SS")
```











Lasso regression

```
library(glmnet)
```

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

```
## Loaded glmnet 4.1-4
```

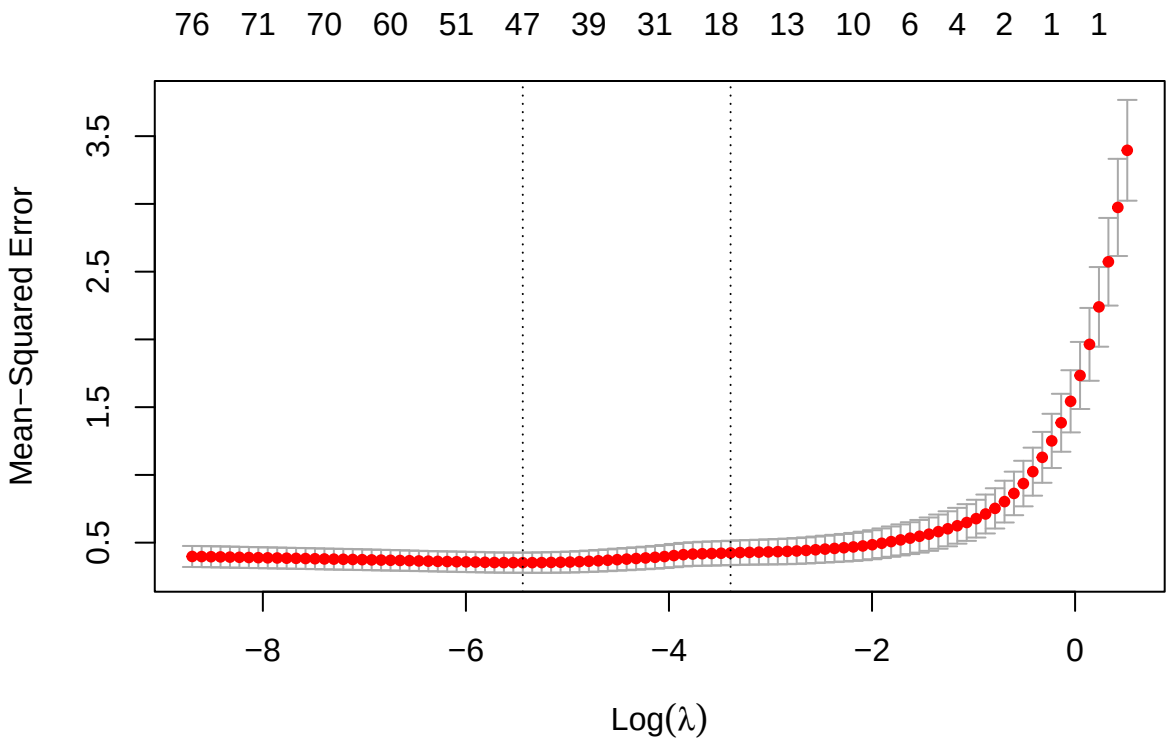
```
#perform k-fold cross-validation to find optimal lambda value
cv_model <- cv.glmnet(model.matrix(PADDY.HARVEST.PRICE..Rs.per.Quintal.~., data = data.train)[,-6],data

summary(cv_model)
```

```
##            Length Class  Mode
## lambda     100    -none- numeric
## cvm        100    -none- numeric
## cvsd       100    -none- numeric
## cvup       100    -none- numeric
## cvlo       100    -none- numeric
## nzero      100    -none- numeric
## call         4    -none- call
## name         1    -none- character
## glmnet.fit  12    elnet  list
## lambda.min   1    -none- numeric
## lambda.1se   1    -none- numeric
## index        2    -none- numeric
```

```
#find optimal lambda value that minimizes test MSE
best_lambda <- cv_model$lambda.min
best_lambda
```

```
## [1] 0.004339537
```

```
#produce plot of test MSE by lambda value
plot(cv_model)
```




```
#find coefficients of best model
best_model <- glmnet(model.matrix(PADDY.HARVEST.PRICE..Rs.per.Quintal.~., data = data.train)[,-6],data.

#coeffiecients of lasso best model
coef(best_model)
```

```
## 88 x 1 sparse Matrix of class "dgCMatrix"
##                                                       s0
## (Intercept)                                -4.734476e+02
## (Intercept)                                 .
## Dist.Code45                                 6.639608e-01
## Dist.Code46                                 .
## Dist.Code47                                -8.271747e-02
## Dist.Code48                                -2.325481e-01
## Dist.Code50                                 .
## Dist.Code51                                -3.401945e-01
## Dist.Code52                                -5.906297e-01
## Dist.Code53                                 7.538067e-02
## Dist.Code54                                 .
## Dist.Code503                                4.503646e-01
## Dist.Code504                                1.388253e-03
## Year.x                                      2.336135e-01
## RICE.AREA..1000.ha.                         .
## RICE.PRODUCTION..1000.tons.                 .
## RICE.YIELD..Kg.per.ha.                      .
```

```
## CANALS.AREA..1000.ha.                              .
## TANKS.AREA..1000.ha.                               9.734521e-01
## TUBE.WELLS.AREA..1000.ha.                          .
## OTHER.WELLS.AREA..1000.ha.                        -6.805813e-01
## TOTAL.WELLS.AREA..1000.ha.                         .
## OTHER.SOURCES.AREA..1000.ha.                       .
## NET.AREA..1000.ha.                                 .
## GROSS.AREA..1000.ha.                               .
## RICE.IRRIGATED.AREA..1000.ha.                      .
## DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day.            3.371044e-01
## DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day.          .
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.       5.087735e-01
## STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.     .
## NITROGEN.KHARIF.CONSUMPTION..tons.                 .
## NITROGEN.RABI.CONSUMPTION..tons.                   .
## PHOSPHATE.KHARIF.CONSUMPTION..tons.                3.584018e-03
## PHOSPHATE.RABI.CONSUMPTION..tons.                  .
## POTASH.KHARIF.CONSUMPTION..tons.                   1.456769e-03
## POTASH.RABI.CONSUMPTION..tons.                     3.053258e-04
## TOTAL.KHARIF.CONSUMPTION..tons.                    .
## TOTAL.RABI.CONSUMPTION..tons.                      .
## NITROGEN.CONSUMPTION..tons.                        1.570547e-04
## NITROGEN.SHARE.IN.NPK..Percent.                    4.157078e-01
## NITROGEN.PER.HA.OF.NCA..Kg.per.ha.                 .
## NITROGEN.PER.HA.OF.GCA..Kg.per.ha.                 .
## PHOSPHATE.CONSUMPTION..tons.                       .
## PHOSPHATE.SHARE.IN.NPK..Percent.                   .
## PHOSPHATE.PER.HA.OF.NCA..Kg.per.ha.                .
## PHOSPHATE.PER.HA.OF.GCA..Kg.per.ha.                3.196759e-01
## POTASH.CONSUMPTION..tons.                          .
## POTASH.SHARE.IN.NPK..Percent.                      1.160564e+00
## POTASH.PER.HA.OF.NCA..Kg.per.ha.                  -1.961630e-01
## POTASH.PER.HA.OF.GCA..Kg.per.ha.                   .
## TOTAL.CONSUMPTION..tons.                           1.794810e-04
## TOTAL.PER.HA.OF.NCA..Kg.per.ha.                    .
## TOTAL.PER.HA.OF.GCA..Kg.per.ha.                    .
## Zaid.temp.min.c                                    .
## Kharif.temp.min.c                                  2.572663e+01
## Rabi.temp.min.c                                    1.382631e+00
## Zaid.temp.max.c                                   -3.477152e+01
## Kharif.temp.max.c                                  .
## Rabi.temp.max.c                                    1.034213e+00
## Zaid.percipitation.mm                             -3.432855e-01
## Kharif.percipitation.mm                           -5.267218e-01
## Rabi.percipitation.mm                             -7.679079e-01
## Zaid.evapotranspiration.potential.mm               2.488405e+00
## Kharif.evapotranspiration.potential.mm            -1.838875e+00
## Rabi.evapotranspiration.potential.mm               1.269889e+01
## Zaid.evapotranspiration.actual.mm                  .
## Kharif.evapotranspiration.actual.mm               -1.590367e+00
## Rabi.evapotranspiration.actual.mm                  9.578629e-01
## Zaid.rainfall.mm                                  -3.730330e-01
## Kharif.rainfall.mm                                -9.916964e-02
## Rabi.rainfall.mm                                   9.894342e-01
```

```
## TOTAL.AREA..1000.ha.                              .
## FOREST.AREA..1000.ha.                             1.910167e-01
## BARREN.AND.UNCULTIVABLE.LAND.AREA..1000.ha.       .
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha. -2.677306e-01
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.           -1.165558e+00
## PERMANENT.PASTURES.AREA..1000.ha.              -8.547726e-01
## OTHER.FALLOW.AREA..1000.ha.                       .
## CURRENT.FALLOW.AREA..1000.ha.                     5.877895e-01
## NET.CROPPED.AREA..1000.ha.                        .
## GROSS.CROPPED.AREA..1000.ha.                      .
## CROPING.INTENSITY..Percent.                       4.843181e-01
## LENGTH.OF.GROWING.PERIOD.DAYS..Number.            1.524766e-01
## Soil.Type.Orthids                                 .
## Soil.Type.Sandy.Alfisol                           .
## Soil.Type.Loamy.Alfisols                       -7.761737e+00
## Soil.Type.Vertisols                               .
## Soil.Type.Vertic.soils                            .
```

```
#summary of lasso best model.
summary(best_model)
```

```
##           Length Class     Mode
## a0         1     -none-    numeric
## beta      87     dgCMatrix S4
## df         1     -none-    numeric
## dim        2     -none-    numeric
## lambda     1     -none-    numeric
## dev.ratio  1     -none-    numeric
## nulldev    1     -none-    numeric
## npasses    1     -none-    numeric
## jerr       1     -none-    numeric
## offset     1     -none-    logical
## call       5     -none-    call
## nobs       1     -none-    numeric
```

coefficeints of lasso

```
coef.lasso<- predict(best_model,type = "coefficients",s= best_lambda)
coef.lasso
```

```
## 88 x 1 sparse Matrix of class "dgCMatrix"
##                                                       s1
## (Intercept)                                    -4.734476e+02
## (Intercept)                                       .
## Dist.Code45                                       6.639608e-01
## Dist.Code46                                       .
## Dist.Code47                                    -8.271747e-02
## Dist.Code48                                    -2.325481e-01
## Dist.Code50                                       .
## Dist.Code51                                    -3.401945e-01
## Dist.Code52                                    -5.906297e-01
## Dist.Code53                                       7.538067e-02
## Dist.Code54                                       .
## Dist.Code503                                      4.503646e-01
## Dist.Code504                                      1.388253e-03
```

```
## Year.x                                          2.336135e-01
## RICE.AREA..1000.ha.                              .
## RICE.PRODUCTION..1000.tons.                      .
## RICE.YIELD..Kg.per.ha.                           .
## CANALS.AREA..1000.ha.                            .
## TANKS.AREA..1000.ha.                             9.734521e-01
## TUBE.WELLS.AREA..1000.ha.                        .
## OTHER.WELLS.AREA..1000.ha.                      -6.805813e-01
## TOTAL.WELLS.AREA..1000.ha.                       .
## OTHER.SOURCES.AREA..1000.ha.                     .
## NET.AREA..1000.ha.                               .
## GROSS.AREA..1000.ha.                             .
## RICE.IRRIGATED.AREA..1000.ha.                    .
## DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day.          3.371044e-01
## DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day.        .
## STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.     5.087735e-01
## STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.   .
## NITROGEN.KHARIF.CONSUMPTION..tons.               .
## NITROGEN.RABI.CONSUMPTION..tons.                 .
## PHOSPHATE.KHARIF.CONSUMPTION..tons.              3.584018e-03
## PHOSPHATE.RABI.CONSUMPTION..tons.                .
## POTASH.KHARIF.CONSUMPTION..tons.                 1.456769e-03
## POTASH.RABI.CONSUMPTION..tons.                   3.053258e-04
## TOTAL.KHARIF.CONSUMPTION..tons.                  .
## TOTAL.RABI.CONSUMPTION..tons.                    .
## NITROGEN.CONSUMPTION..tons.                      1.570547e-04
## NITROGEN.SHARE.IN.NPK..Percent.                  4.157078e-01
## NITROGEN.PER.HA.OF.NCA..Kg.per.ha.               .
## NITROGEN.PER.HA.OF.GCA..Kg.per.ha.               .
## PHOSPHATE.CONSUMPTION..tons.                     .
## PHOSPHATE.SHARE.IN.NPK..Percent.                 .
## PHOSPHATE.PER.HA.OF.NCA..Kg.per.ha.              .
## PHOSPHATE.PER.HA.OF.GCA..Kg.per.ha.              3.196759e-01
## POTASH.CONSUMPTION..tons.                        .
## POTASH.SHARE.IN.NPK..Percent.                    1.160564e+00
## POTASH.PER.HA.OF.NCA..Kg.per.ha.                -1.961630e-01
## POTASH.PER.HA.OF.GCA..Kg.per.ha.                 .
## TOTAL.CONSUMPTION..tons.                         1.794810e-04
## TOTAL.PER.HA.OF.NCA..Kg.per.ha.                  .
## TOTAL.PER.HA.OF.GCA..Kg.per.ha.                  .
## Zaid.temp.min.c                                  .
## Kharif.temp.min.c                                2.572663e+01
## Rabi.temp.min.c                                  1.382631e+00
## Zaid.temp.max.c                                 -3.477152e+01
## Kharif.temp.max.c                                .
## Rabi.temp.max.c                                  1.034213e+00
## Zaid.percipitation.mm                           -3.432855e-01
## Kharif.percipitation.mm                         -5.267218e-01
## Rabi.percipitation.mm                           -7.679079e-01
## Zaid.evapotranspiration.potential.mm             2.488405e+00
## Kharif.evapotranspiration.potential.mm          -1.838875e+00
## Rabi.evapotranspiration.potential.mm             1.269889e+01
## Zaid.evapotranspiration.actual.mm                .
## Kharif.evapotranspiration.actual.mm             -1.590367e+00
```

```
## Rabi.evapotranspiration.actual.mm                9.578629e-01
## Zaid.rainfall.mm                                -3.730330e-01
## Kharif.rainfall.mm                              -9.916964e-02
## Rabi.rainfall.mm                                 9.894342e-01
## TOTAL.AREA..1000.ha.                             .
## FOREST.AREA..1000.ha.                            1.910167e-01
## BARREN.AND.UNCULTIVABLE.LAND.AREA..1000.ha.      .
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha. -2.677306e-01
## CULTIVABLE.WASTE.LAND.AREA..1000.ha.           -1.165558e+00
## PERMANENT.PASTURES.AREA..1000.ha.              -8.547726e-01
## OTHER.FALLOW.AREA..1000.ha.                      .
## CURRENT.FALLOW.AREA..1000.ha.                    5.877895e-01
## NET.CROPPED.AREA..1000.ha.                       .
## GROSS.CROPPED.AREA..1000.ha.                     .
## CROPING.INTENSITY..Percent.                      4.843181e-01
## LENGTH.OF.GROWING.PERIOD.DAYS..Number.           1.524766e-01
## Soil.Type.Orthids                                .
## Soil.Type.Sandy.Alfisol                          .
## Soil.Type.Loamy.Alfisols                        -7.761737e+00
## Soil.Type.Vertisols                              .
## Soil.Type.Vertic.soils                           .
```

```
#coeffs of full model  \beta
coef(model.all)
```

```
##                                    (Intercept)
##                                   2.437591e+03
##                                    Dist.Code45
##                                   1.357950e+00
##                                    Dist.Code46
##                                  -1.508261e+00
##                                    Dist.Code47
##                                  -1.304627e+00
##                                    Dist.Code48
##                                  -2.673835e+00
##                                    Dist.Code49
##                                  -3.599301e+00
##                                    Dist.Code50
##                                  -4.579574e+00
##                                    Dist.Code51
##                                  -5.174468e+00
##                                    Dist.Code52
##                                  -5.411478e+00
##                                    Dist.Code53
##                                  -2.423772e+00
##                                    Dist.Code54
##                                  -3.327058e+00
##                                   Dist.Code503
##                                   5.857345e-01
##                                   Dist.Code504
##                                  -4.035222e+00
##                                         Year.x
##                                   2.704590e-01
##                             RICE.AREA..1000.ha.
##                                   4.149004e-01
```

```
##                     RICE.PRODUCTION..1000.tons.
##                                    4.370201e-02
##                          RICE.YIELD..Kg.per.ha.
##                                    5.695575e-04
##                           CANALS.AREA..1000.ha.
##                                   -1.713381e+01
##                            TANKS.AREA..1000.ha.
##                                   -1.546361e+01
##                       TUBE.WELLS.AREA..1000.ha.
##                                   -6.874755e-01
##                      OTHER.WELLS.AREA..1000.ha.
##                                   -4.982609e-01
##                      TOTAL.WELLS.AREA..1000.ha.
##                                   -1.623355e+01
##                   OTHER.SOURCES.AREA..1000.ha.
##                                   -1.801380e+01
##                               NET.AREA..1000.ha.
##                                    1.747914e+01
##                             GROSS.AREA..1000.ha.
##                                   -6.234314e-01
##                   RICE.IRRIGATED.AREA..1000.ha.
##                                   -4.961770e-01
##        DISTRICT.MALE.FIELD.LABOUR..Rs.per.Day.
##                                    7.912864e-01
##      DISTRICT.FEMALE.FIELD.LABOUR..Rs.per.Day.
##                                   -6.638766e-01
##   STATE.MALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.
##                                    3.502104e-01
## STATE.FEMALE.AVERAGE.FIELD.LABOUR..Rs.per.Day.
##                                              NA
##              NITROGEN.KHARIF.CONSUMPTION..tons.
##                                    1.384281e+00
##                NITROGEN.RABI.CONSUMPTION..tons.
##                                    6.693772e-04
##             PHOSPHATE.KHARIF.CONSUMPTION..tons.
##                                    1.392206e+00
##               PHOSPHATE.RABI.CONSUMPTION..tons.
##                                   -5.428280e-03
##                POTASH.KHARIF.CONSUMPTION..tons.
##                                    1.391683e+00
##                  POTASH.RABI.CONSUMPTION..tons.
##                                    5.103688e-03
##                 TOTAL.KHARIF.CONSUMPTION..tons.
##                                   -1.387349e+00
##                   TOTAL.RABI.CONSUMPTION..tons.
##                                              NA
##                     NITROGEN.CONSUMPTION..tons.
##                                    9.811428e+01
##                  NITROGEN.SHARE.IN.NPK..Percent.
##                                    1.267984e+00
##              NITROGEN.PER.HA.OF.NCA..Kg.per.ha.
##                                    5.343926e+02
##              NITROGEN.PER.HA.OF.GCA..Kg.per.ha.
##                                    4.028258e+02
```

```
##                   PHOSPHATE.CONSUMPTION..tons.
##                                     9.811273e+01
##                PHOSPHATE.SHARE.IN.NPK..Percent.
##                                     3.546885e+00
##             PHOSPHATE.PER.HA.OF.NCA..Kg.per.ha.
##                                     5.322678e+02
##             PHOSPHATE.PER.HA.OF.GCA..Kg.per.ha.
##                                     4.073663e+02
##                      POTASH.CONSUMPTION..tons.
##                                     9.812658e+01
##                  POTASH.SHARE.IN.NPK..Percent.
##                                    -1.523691e+00
##                POTASH.PER.HA.OF.NCA..Kg.per.ha.
##                                     5.099232e+02
##                POTASH.PER.HA.OF.GCA..Kg.per.ha.
##                                     4.291626e+02
##                       TOTAL.CONSUMPTION..tons.
##                                    -9.811446e+01
##                 TOTAL.PER.HA.OF.NCA..Kg.per.ha.
##                                    -5.304725e+02
##                 TOTAL.PER.HA.OF.GCA..Kg.per.ha.
##                                    -4.075732e+02
##                                 Zaid.temp.min.c
##                                    -9.696911e+00
##                               Kharif.temp.min.c
##                                     1.381122e+01
##                                 Rabi.temp.min.c
##                                     1.097482e+00
##                                 Zaid.temp.max.c
##                                    -7.070264e+01
##                               Kharif.temp.max.c
##                                     2.191571e+01
##                                 Rabi.temp.max.c
##                                     3.270980e+01
##                            Zaid.percipitation.mm
##                                    -6.952600e-01
##                          Kharif.percipitation.mm
##                                    -5.286743e-01
##                            Rabi.percipitation.mm
##                                    -7.204925e-01
##           Zaid.evapotranspiration.potential.mm
##                                     6.751706e+00
##         Kharif.evapotranspiration.potential.mm
##                                    -9.465617e-01
##           Rabi.evapotranspiration.potential.mm
##                                     1.349305e+01
##              Zaid.evapotranspiration.actual.mm
##                                     2.553029e-02
##            Kharif.evapotranspiration.actual.mm
##                                    -1.369456e+00
##              Rabi.evapotranspiration.actual.mm
##                                     1.015059e+00
##                                Zaid.rainfall.mm
##                                    -5.829220e-01
```

```
##                                Kharif.rainfall.mm
##                                     -1.504864e-01
##                                  Rabi.rainfall.mm
##                                      1.085945e+00
##                               TOTAL.AREA..1000.ha.
##                                      8.481028e-02
##                              FOREST.AREA..1000.ha.
##                                      2.060241e-01
##    BARREN.AND.UNCULTIVABLE.LAND.AREA..1000.ha.
##                                     -7.938553e-01
## LAND.PUT.TO.NONAGRICULTURAL.USE.AREA..1000.ha.
##                                      1.871706e-01
##           CULTIVABLE.WASTE.LAND.AREA..1000.ha.
##                                     -1.067573e-01
##              PERMANENT.PASTURES.AREA..1000.ha.
##                                      1.458712e+00
##                     OTHER.FALLOW.AREA..1000.ha.
##                                      8.348447e-01
##                   CURRENT.FALLOW.AREA..1000.ha.
##                                      9.095095e-01
##                      NET.CROPPED.AREA..1000.ha.
##                                     -2.354998e+00
##                    GROSS.CROPPED.AREA..1000.ha.
##                                      2.391081e+00
##                     CROPING.INTENSITY..Percent.
##                                     -7.423637e+00
##         LENGTH.OF.GROWING.PERIOD.DAYS..Number.
##                                                NA
##                               Soil.Type.Orthids
##                                                NA
##                          Soil.Type.Sandy.Alfisol
##                                                NA
##                         Soil.Type.Loamy.Alfisols
##                                                NA
##                              Soil.Type.Vertisols
##                                                NA
##                            Soil.Type.Vertic.soils
##                                                NA
```

We an clearly see that compared to the linear fit, our Lasso Regularization has shrinked the coefficients and we can also notice that most of them are shrinked to almost zero.

Results

Data Rescaling

```
#rescaling function
#DE-standardizing
for (i in names.list){
  x = 1
  while(x <= nrow(df.stats)){
    if(i == df.stats[x,'names']){
      #print(i)
      m = df.stats[1,'m']
```

```
      s = df.stats[1,'s']
      k = unlist(df.all[i])
      z <- ((k * as.numeric(s)) + as.numeric(m))
      df.all[i] <- z
    }
    x = x+1
  }
}
```

Train,test,split - rescaled data

Since, here we are only splitting data based on year and its not a random shuffling, so now we are splitting data again for rescaled data and we will run our model on re-scaled observations.

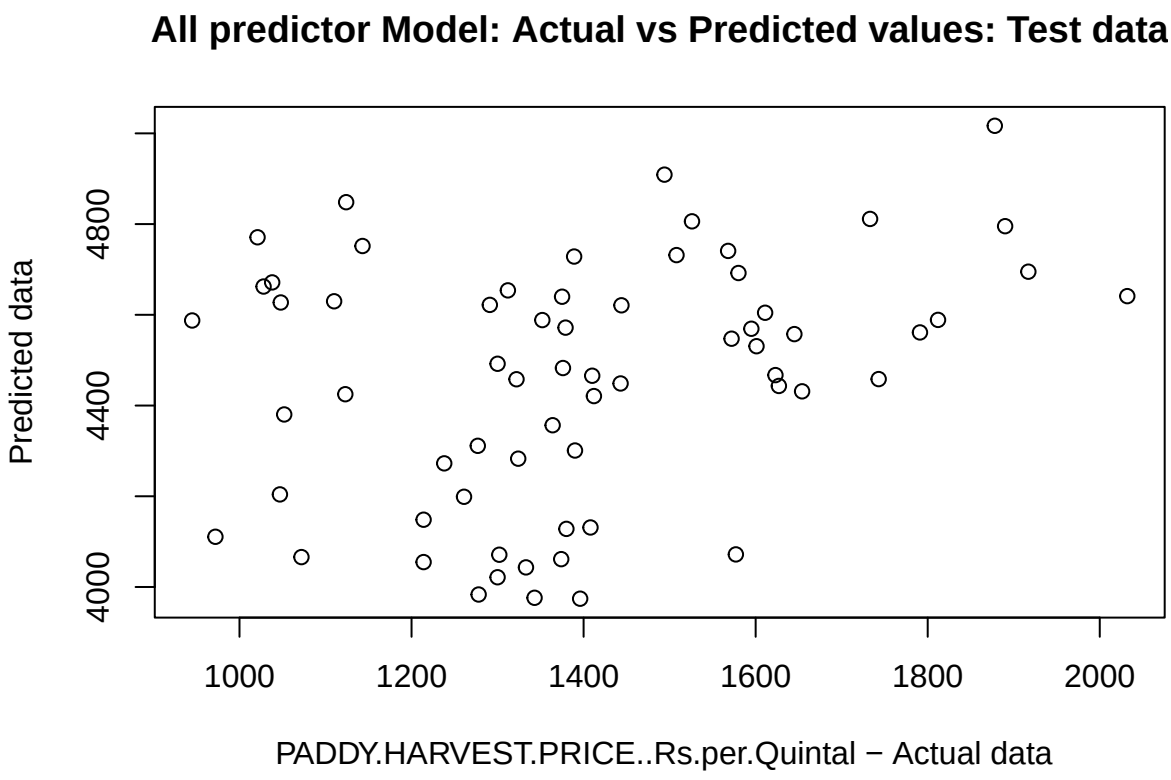
```
#data split
data.train <- df.all[df.all$Year.x <= 2010,]
data.test <- df.all[df.all$Year.x > 2010,]
```

Testing Model with all predictors

```
#testing predictions
prediction.all.test <- predict(model.all,newdata = data.test)
```

```
## Warning in predict.lm(model.all, newdata = data.test): prediction from a rank-
## deficient fit may be misleading
```

```
testing.df1 <- data.test%>% mutate(pred.all.model.test = prediction.all.test)
plot(testing.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal.,testing.df1$pred.all.model.test,main = "All predi
```



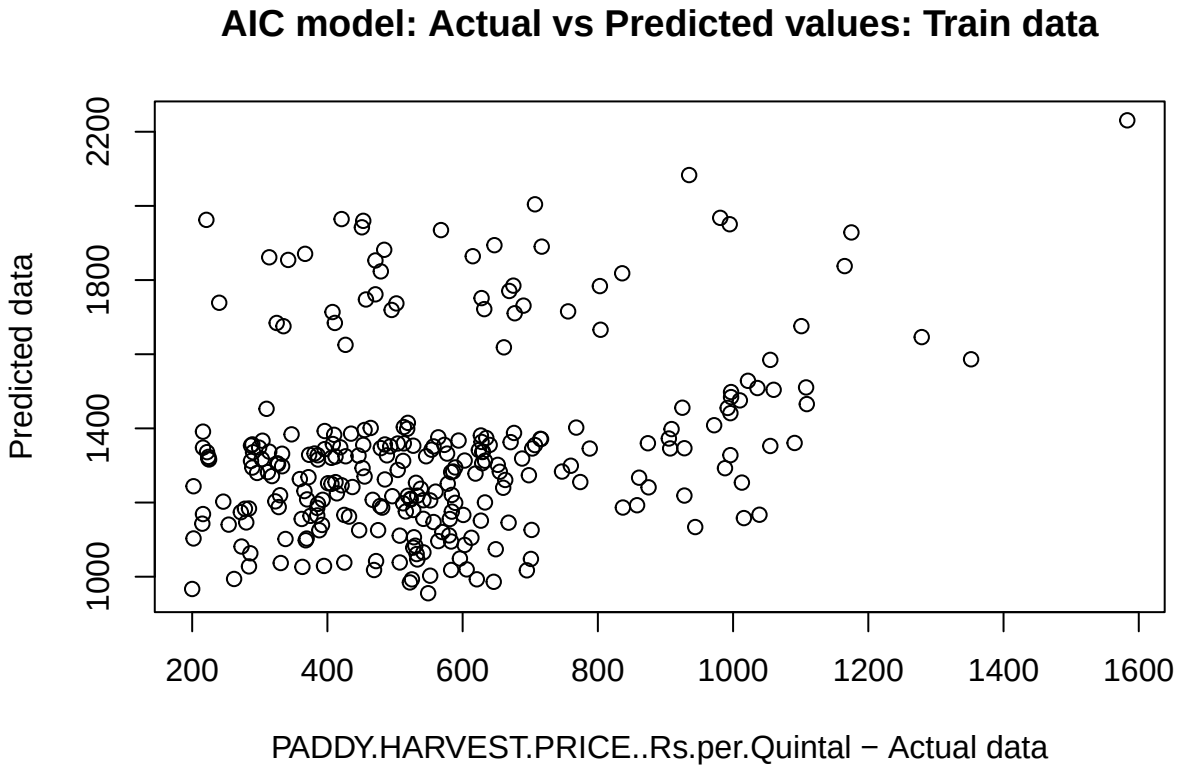
```
#RMSE for all predictor model
#sqrt(mean((data$actual - data$predicted)^2))

RMSE.test <- sqrt(mean((testing.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal. - testing.df1$pred.all.model.te
RMSE.test
```

```
## [1] 3076.591
```

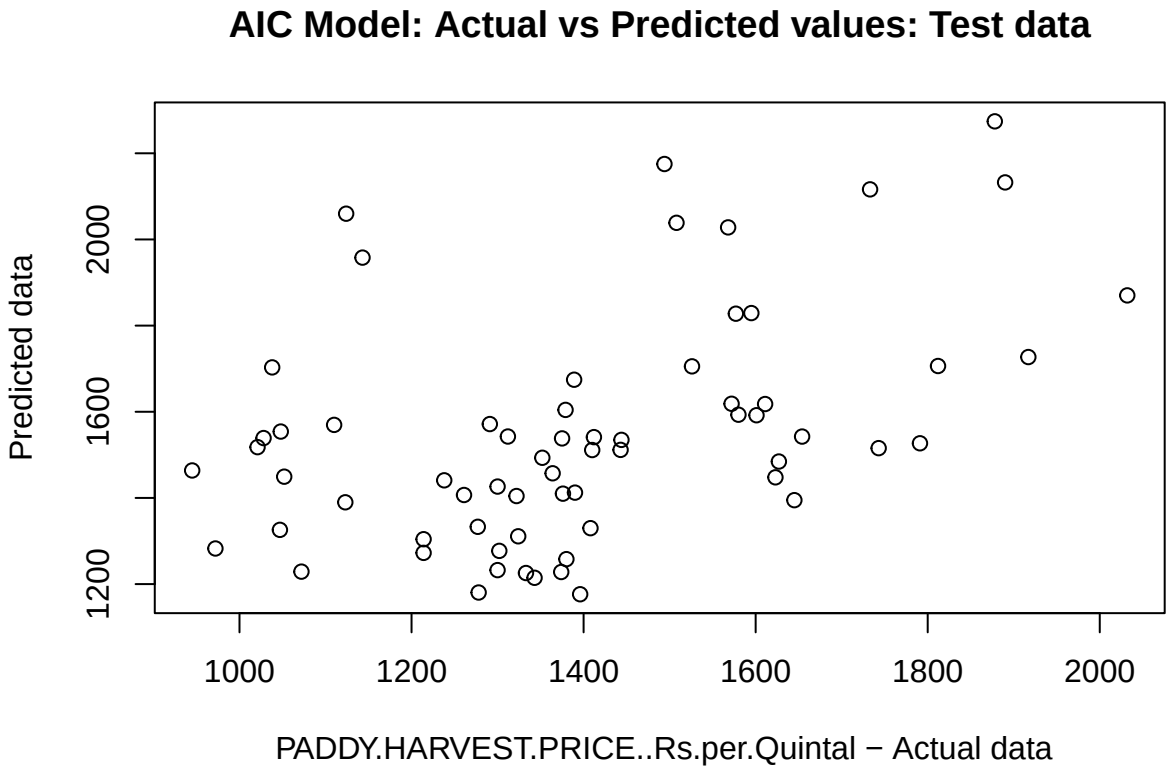
We can see that our model with all predictors did not correctly predict the results, the predictions were too bad. And RMSE is around 3076.591

Testing AIC model

```
#training predictions
prediction.train <- predict(fwdfit.aic$model,newdata = data.train)
training.df1 <- data.train%>% mutate(pred.reg.train = prediction.train)
plot(training.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal.,training.df1$pred.reg.train,main = "AIC model: Ac
```




```
#testing predictions
prediction.test <- predict(fwdfit.aic$model,newdata = data.test)
testing.df1 <- data.test%>% mutate(pred.reg.test = prediction.test)
plot(testing.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal.,testing.df1$pred.reg.test,main = "AIC Model: Actua
```




```
#Calculating RMSE
RMSE.train <- sqrt(mean((training.df1$pred.reg.train-training.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal.)
RMSE.train
```

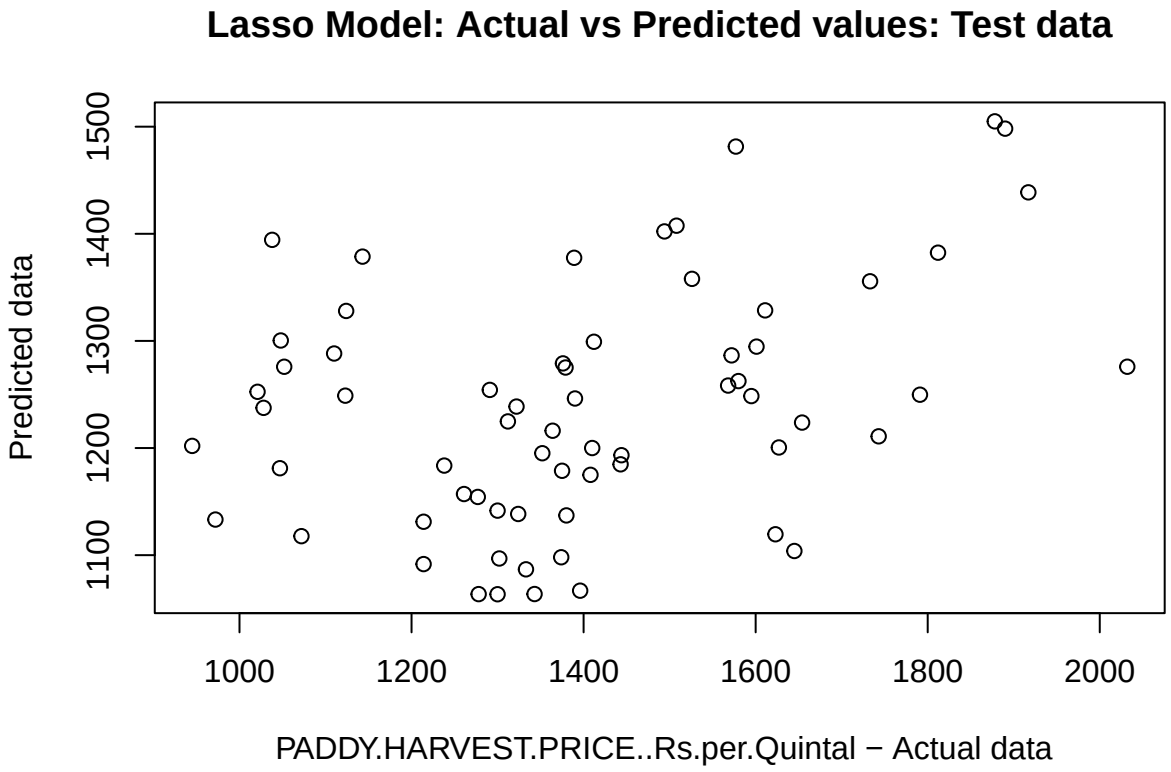
```
## [1] 830.8445
```

```
#sqrt(mean((data$actual - data$predicted)^2))

RMSE.test <- sqrt(mean((testing.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal. - testing.df1$pred.reg.test)^2
RMSE.test
```

```
## [1] 304.8638
```

Testing lasso regression model

```
#use lasso regression model to predict response value

prediction.test.lasso <- predict(best_model,s=best_lambda,newx = model.matrix(PADDY.HARVEST.PRICE..Rs.p

#plotting results

testing.df1 <- data.test%>% mutate(pred.lasso.test = prediction.test.lasso)
plot(testing.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal.,testing.df1$pred.lasso.test,main = "Lasso Model:
```




```
#RMSE
#sqrt(mean((data$actual - data$predicted)^2))

RMSE.test.lasso <- sqrt(mean((testing.df1$PADDY.HARVEST.PRICE..Rs.per.Quintal. - testing.df1$pred.lasso

RMSE.test.lasso
```

```
## [1] 281.5105
```

Conclusion:

After carefully cleaning, processing and transforming our data, we could reduce the number of parameters from 137 to 77. Feature engineering also helped us in reducing reductant rows that were created during joining the dataframes.

Even 77 features is a very large for observations of size 338. We tested this by creating Linear Regression model with all 77 parameters and our test RMSE is 3076.591.

So, we decided to further reduce the features by choosing any of the available feature selection models, so we decided to choose AIC and Lasso to compare the results.

Our test RMSE for AIC model is 304.8638, our test RMSE for Lasso model is 281.5105

We can see both of our models predicted good results compared to Model with all predictors.