Causal Inference on Crop Yield

Md Mominul Islam(101009250)

8/10/2021

Introduction

In this project, we have harvest and seeding data of a particular field for two different crops (Soybean and Corn) from year 2017 to 2020. We have seen that there are 2 years data for Soybeans Harvest (2017 and 2019), 2 years data for Corn Harvest and 2 years data for Corn Seeding Rates which were harvested over the years. Our goal is to to make causal inferences about the relationship among grain yield and seeding rate,and historical grain yields. With using pairs plot we would be able to find some meaningful patterns about higher seeding rates and yield rates or usefulness about alternative planting of corn and soybeans. In order to complete our inferences, we have to combine individual yield and seeding maps for 6 years followed by mergeing the combined data set by grid cell into a single data set for further analysis. Then we will normalize the data using ranks and then we will make pairs plot to assert a causal relationship.

Data

In our project we will use two kinds of data set for a particlar field which are harvesting and seeding. In each data set, there are 'Longitude', 'Latitude', 'IsoTime', 'Distance' , 'Heading' columns. We will use 'Longitude' and 'Longitude' data to plot our desired field plot. In our data set, we have different units of measurement as well. For the harvest data, we have 'Yield' column which is measured in bushels per acre while 'ControlRate' and 'AppliedRate' are measured in bushels per acre. As ControlRate and AppliedRate are mostly correlated, we will use AppliedRate and Yield data to assert causal inferences about the relationship among grain yield and seeding rate.

Algorithm of the Project

- Our first step is to loading necessary library and defining functions useful for our project. Next we will Load the data files provided for our project and plot them by Latitude and Longitude to check the data. Creating Grid and assigning individual Grid Cells by defining 'Row' and 'Column' and plotting the Data with Grid lines would be our final step of loading the data files.

- After aggregating the data by grid cell, we will see QQ plotting to check the normality of our aggregated data set and detecting outliers. Using the 'merge()' function, we will merge individual aggregated data sets into a single data table.

- Using Histogram and Pairs Plotting for the aggregated data set, we will find the relationship among the data columns in the combined data set.

- For an appropriate causal inference, we will create a directed acyclic graph. Then with using ranks to normalize the data, we will do plotting the average aggregated variables based on ranks for our allocated grid cells.
- After normalization, we will do histogram and QQ plotting again to observe the change in the data set. Finally we will apply pairs plot to show the relationship among the data columns in the combined data set using ranks. For an appropriate causal inference to the normalized data, we will create a directed acyclic graph.
- Another way of modification is using ranks to normalize the data first and then aggragating the data based on cell numbers. Necessary QQ plotting to observe the change in the data and histogram for every aggregated variable using ranks will be done. Final step would be applying pairs plot to show the relationship among the data columns in the combined data set using ranks. For an appropriate causal inference to the data using ranks, we will create a directed acyclic graph.
- Our last step would be comparing different directed acyclic graphs based on the original data, normalization using ranks before aggrating the data and after aggregating the data.

Math

Formation of Grid Cell

In our project, we will create a Row variable which will be calculated from Latitude in such a way that grid cell 1 is associated with data rows with 0<Latitude<=50, grid cell 2 is associated with 50<Latitude<=100. The simplest way to do this is to divide Latitude by 50 and use the ceiling function. Similarly, create a Column variable from Longitude.

$ Row = $ $ Column = $

$ Cell = 1000 * Row + Column $

Normalization Method

- We need to convert the data to a common scale because we have three distinct crops with potentially very different means.
- We will denote the i_{th} yield observation for Year j as y_{ij} where we normalize yield by one of the following methods, in each case holding j constant and iterating over i only within years. If we assume 20 rows and 6 columns, then y_{ij} = y_{1j}, y_{2j},..., y_{Ij} where I=120. Similarly, we would denote the successive yield estimates for grid cell i as y_{ij} = y_{i1}, y_{i2}, … , y_{iJ} where J=5.
- Three possible normalization formulas were given among them Rank and Z score method were chosen to convert yield values across different crops to a common scale.

Rank

We have replaced y_{ij} with $r_{ij} = rank(y_{ij})$ and determined ranks independently for j = 1,2,3, … ,J for years {2017, 2018, … , 2020}

Z-score

$$\overline{y_{.i}} = \frac{\sum_{i=1}^{I} y_{ij}}{I}$$

and

$$s_{.j}^2 = \frac{\sum_{i=1}^{I} (y_{ij} - \overline{y_{.j}})^2}{I - 1}$$

where I are the number of Yield values for year j. Replace y_{ij} with

$$z_{ij} = \frac{(y_{ij} - \overline{y_{.j}})}{s_{.j}}$$

Independently for j = 1,2, … ,J for years {2017, 2018, … , 2020}. We have to keep in mind that this method makes use of the first (mean) and second moments (variance).

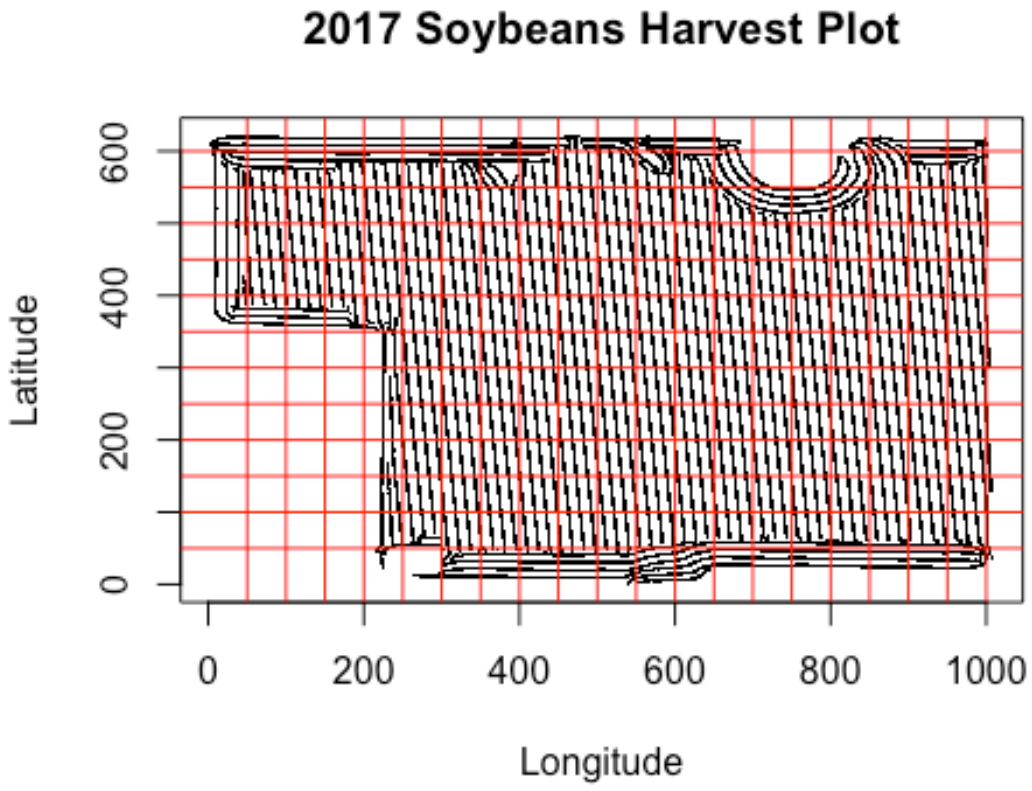
Project Code

```
#Loading necessary Libraries for the project

library(bnlearn)
library(Hmisc)

library(car)

library(ggplot2)
#install.packages("BiocManager")
#BiocManager::install("Rgraphviz")
library(Rgraphviz)

#Defining Necessary Function

AggregateField <-
function(harvest.dat,response='Yield',grid.width=c(50,50),FUN=mean) {
  harvest.dat$Row <- ceiling(harvest.dat$Latitude/grid.width[1])
  harvest.dat$Column <- ceiling(harvest.dat$Longitude/grid.width[2])
  harvest.dat$Cell <- harvest.dat$Row*1000 + harvest.dat$Column
  fmla <- as.formula(paste(response,'~ Row + Column'))
  tmp <- aggregate(fmla,data=harvest.dat,FUN=FUN,na.rm=TRUE)
  count <- aggregate(fmla,data=harvest.dat,FUN=length)
  row.names(tmp) <- paste(tmp$Row,tmp$Column,sep=":")
  tmp$Samples <- count[,3]
```

```
  return(tmp)
}
```

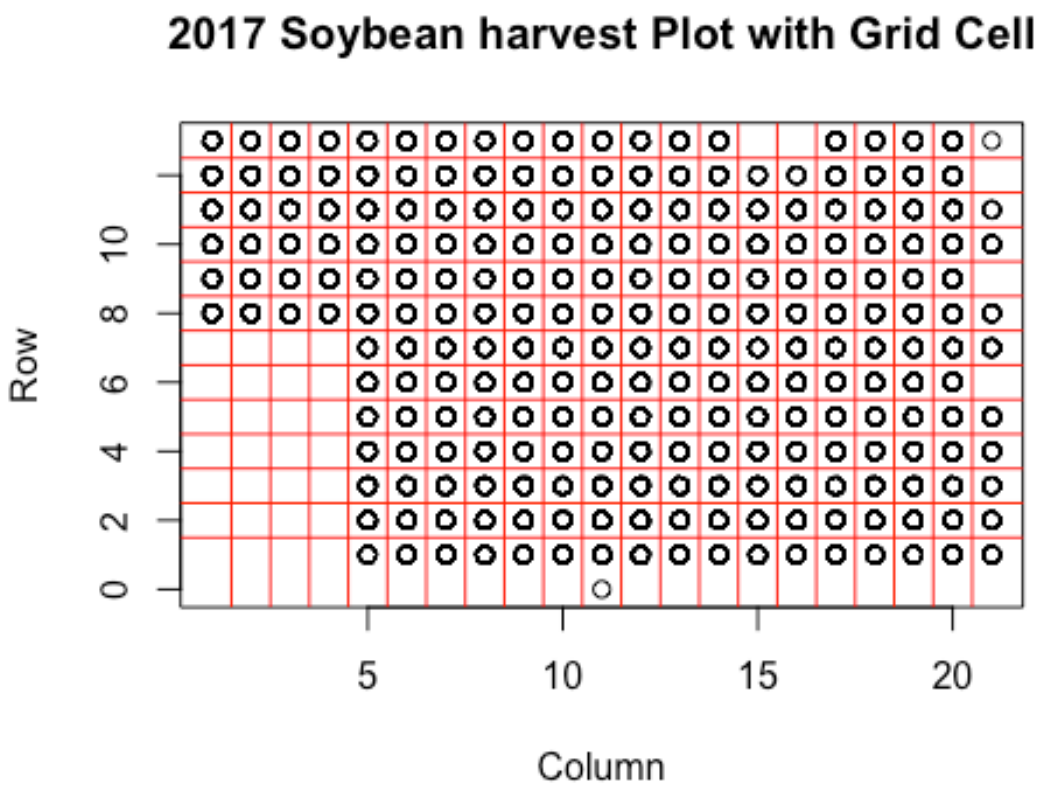
Soybeans Harvest Data: 2017

```
#Loading Data set

SoybeansHarvest.2017 <- read.csv("~/OneDrive - South Dakota State University
- SDSU/STAT 600/Final Project/A 2017 Soybeans Harvest.csv")

#Plotting Data for Visualization

plot(Latitude ~ Longitude,data=SoybeansHarvest.2017,pch = ".", main = "2017
Soybeans Harvest Plot")
abline(h=1:12*50,v=1:20*50,col='red')
```

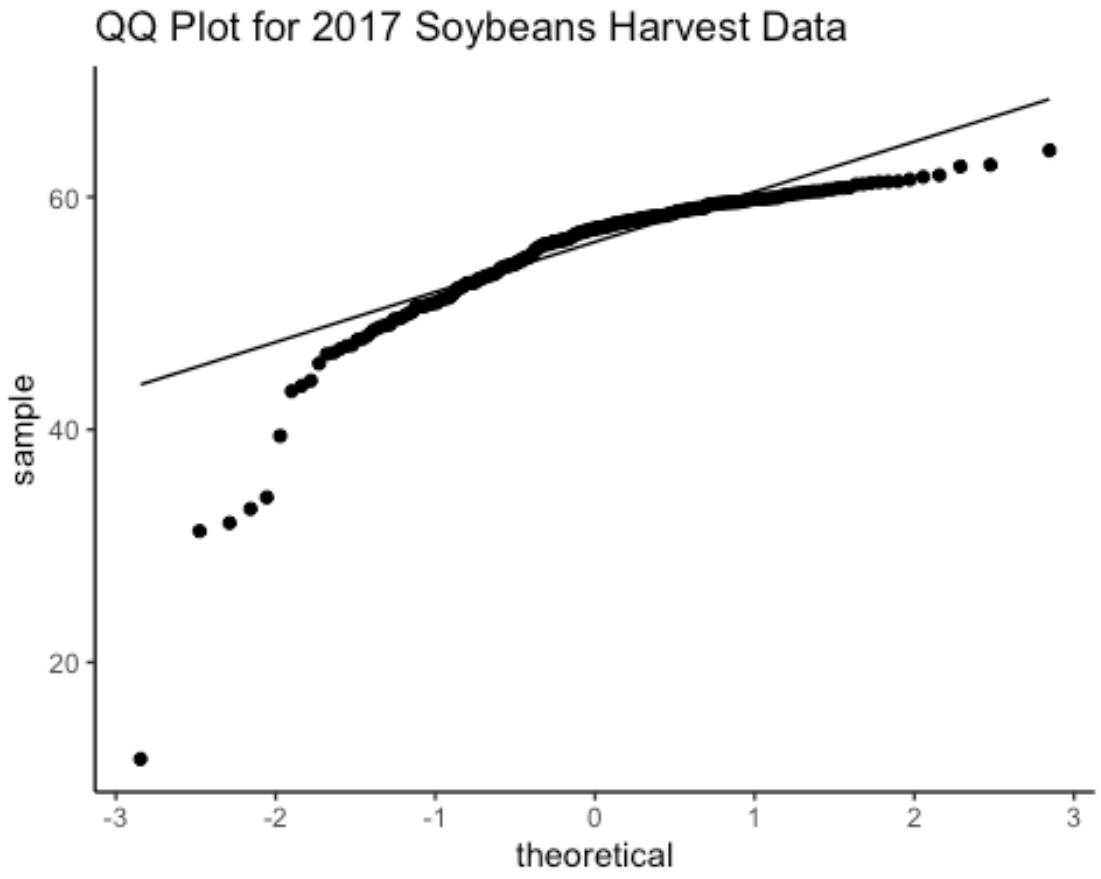



```
# Creating Grid based on video instructions

Row <- ceiling(SoybeansHarvest.2017[,3] / 50)
Column <- ceiling(SoybeansHarvest.2017[,2] / 50)
Cell <- Row*1000 + Column
SoybeansHarvestCombined.2017 <- cbind(Cell,SoybeansHarvest.2017, Row,Column )
plot(Row ~ Column,data=SoybeansHarvestCombined.2017, main = "2017 Soybean
```

```
harvest Plot with Grid Cell" )
abline(h=1:12+0.5,v=1:20+0.5,col='red')
```



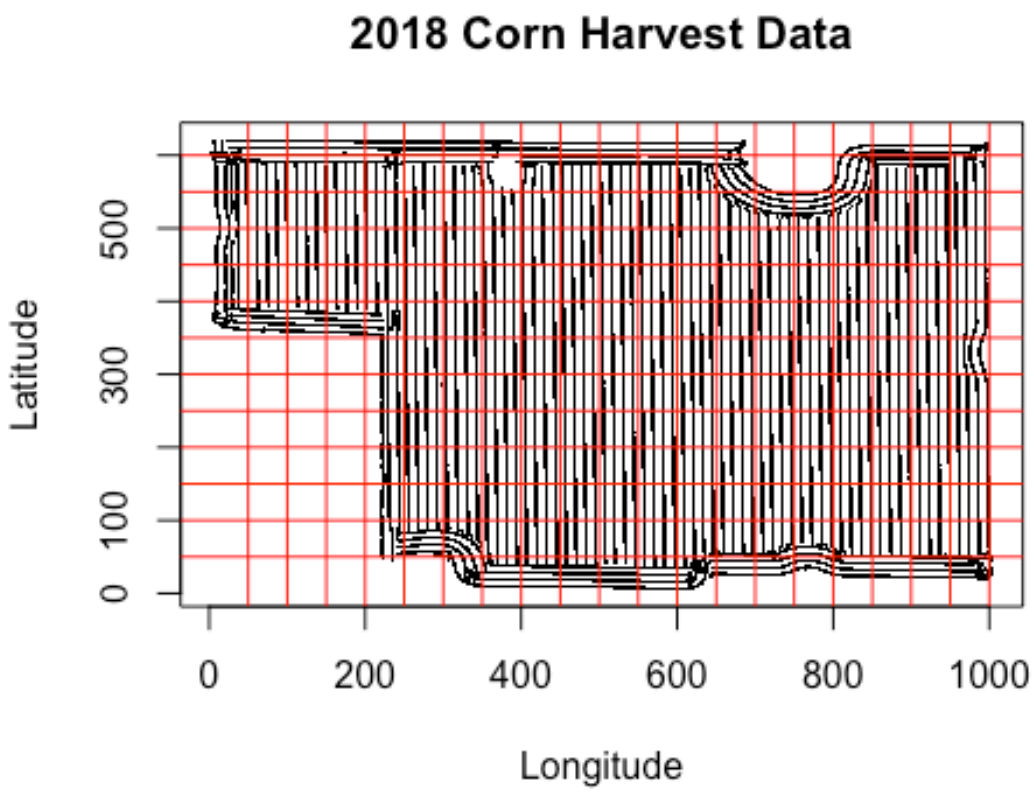

```
#Aggregating 2017 Soybeans Harvest Data
Soybeans.Aggregate.2017 <-
AggregateField(SoybeansHarvestCombined.2017,response='Yield')
Soybeans.Aggregate.2017 <-
Soybeans.Aggregate.2017[Soybeans.Aggregate.2017$Samples>30,]
Soybeans.Aggregate.2017$Cell <- Soybeans.Aggregate.2017$Row*1000 +
Soybeans.Aggregate.2017$Column
names(Soybeans.Aggregate.2017)[3] <- 'Y17'

#Plotting QQ plot
ggplot(Soybeans.Aggregate.2017, aes(sample = Y17)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2017 Soybeans Harvest Data") +
theme_classic()
```



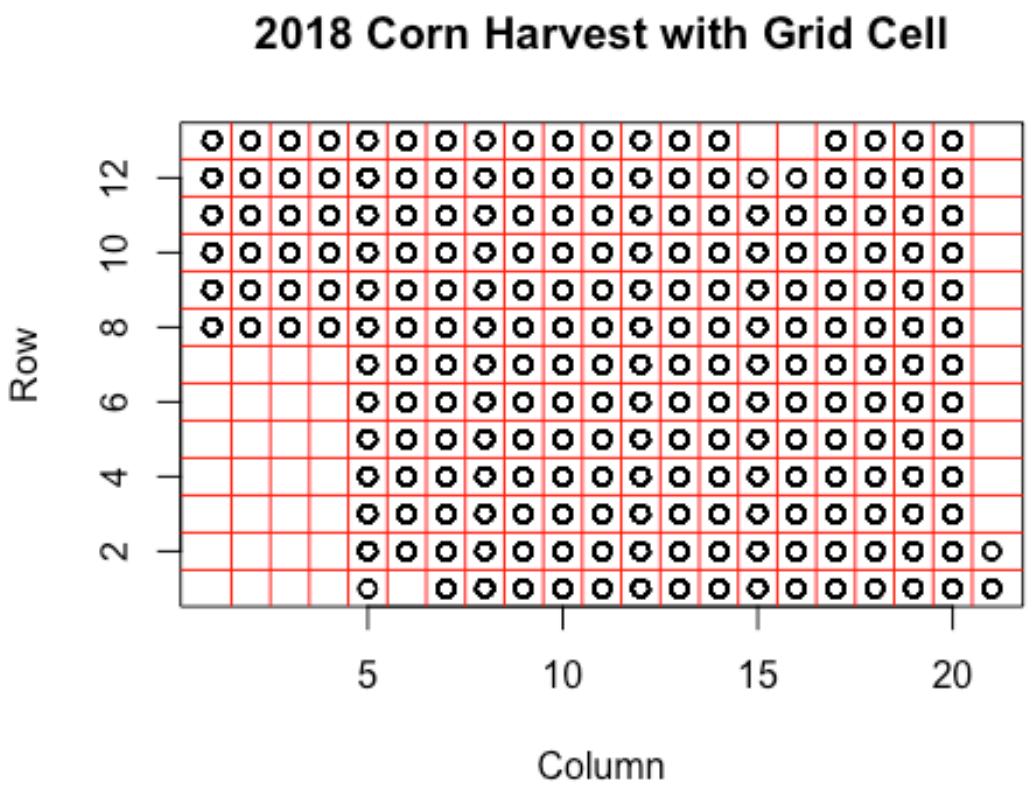
With data aggregation, we can see that the points follow straight line along with some outliers for 2017 Soybeans harvest data set.

##Corn Harvest Data: 2018

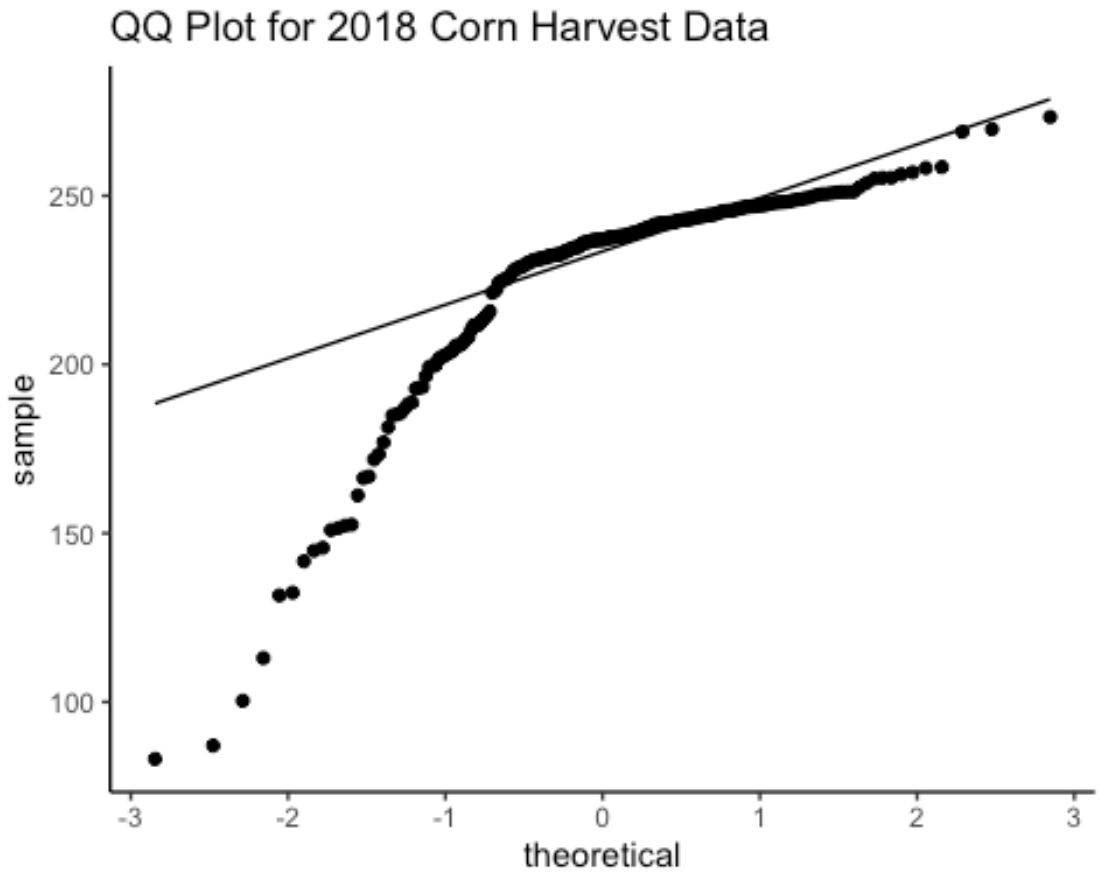
```
#Loading 2018 Corn Harvest Data
CornHarvest.2018<- read.csv("~/OneDrive - South Dakota State University - SDSU/STAT 600/Final Project/A 2018 Corn Harvest.csv")

#Plotting Data set
plot(Latitude ~ Longitude,data=CornHarvest.2018,pch = ".", main = '2018 Corn Harvest Data')
abline(h=1:12*50,v=1:20*50,col='red')
```




```r
#Creating Grid for 2018 Corn Harvest Data
Row <- ceiling(CornHarvest.2018[,3] / 50)
Column <- ceiling(CornHarvest.2018[,2] / 50)
Cell <- Row*1000 + Column
CornHarvestCombined.2018 <- cbind(Cell,CornHarvest.2018, Row,Column )
plot(Row ~ Column,data=CornHarvestCombined.2018, main = '2018 Corn Harvest
with Grid Cell')
abline(h=1:12+0.5,v=1:20+0.5,col='red')
```



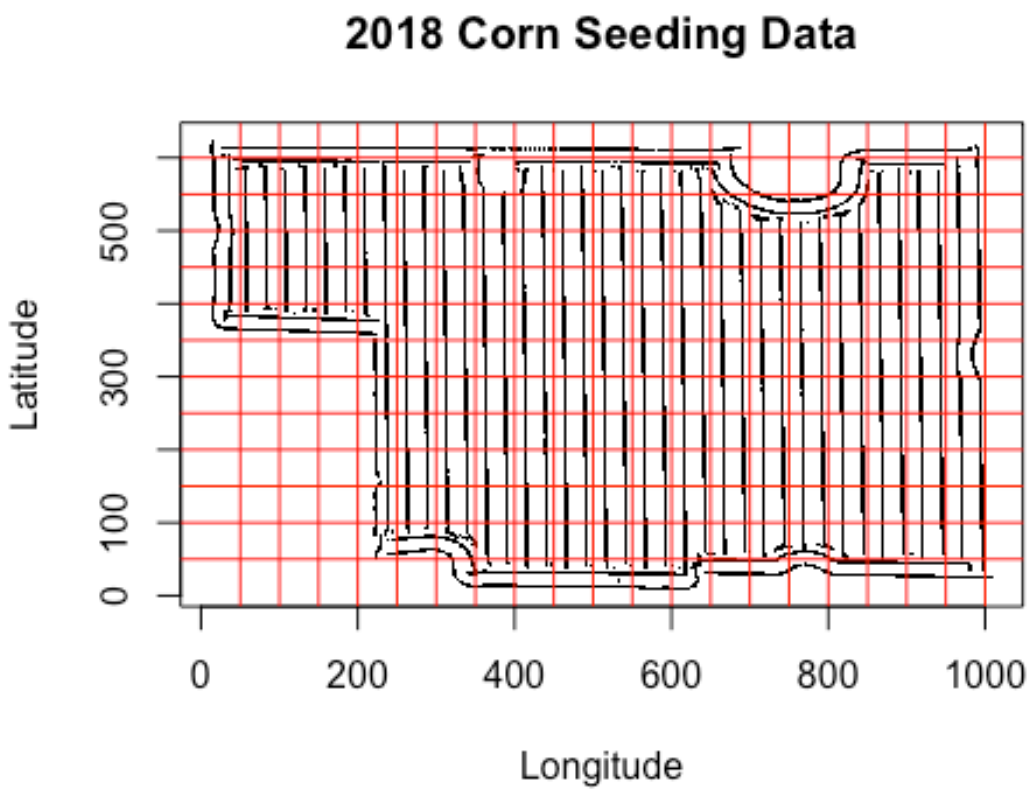

```
#Aggregating 2018 Corn Harvest Data
Corn.Aggregate.2018 <-
AggregateField(CornHarvestCombined.2018,response='Yield')
Corn.Aggregate.2018 <- Corn.Aggregate.2018[Corn.Aggregate.2018$Samples>30,]
Corn.Aggregate.2018$Cell <- Corn.Aggregate.2018$Row*1000 +
Corn.Aggregate.2018$Column
names(Corn.Aggregate.2018)[3] <- 'Y18'

#Plotting QQ plot
ggplot(Corn.Aggregate.2018, aes(sample = Y18)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2018 Corn Harvest Data") +
theme_classic()
```



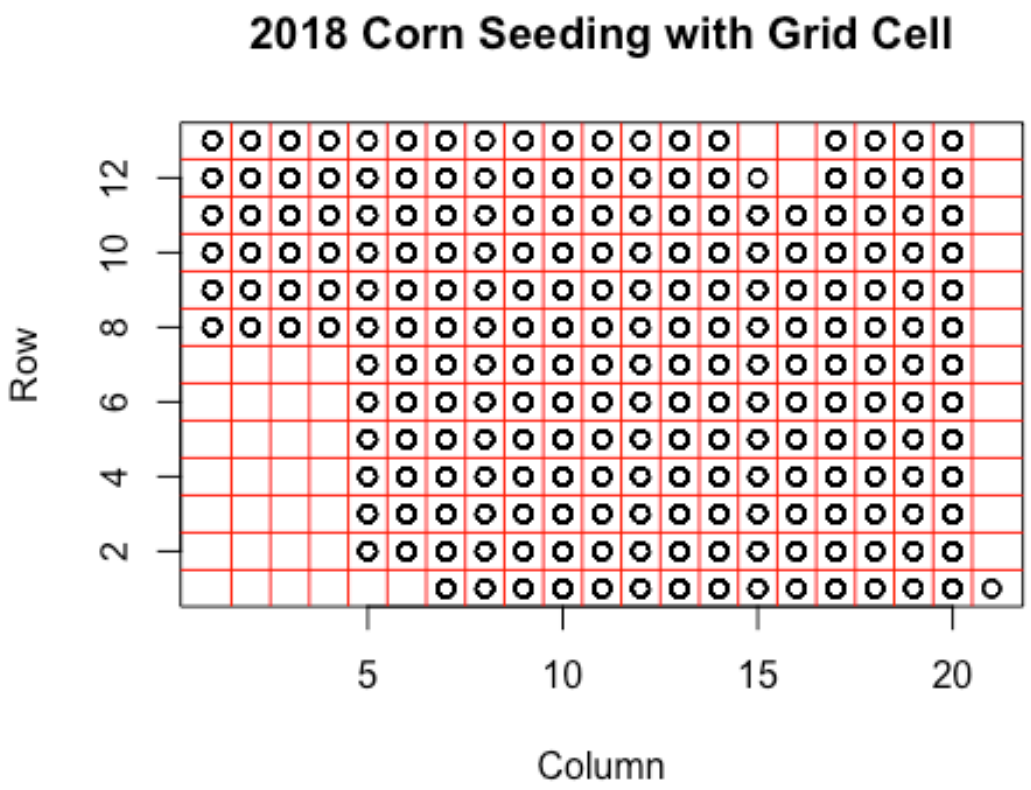
For 2018 Corn harvest data, we have found that the points have formed a curve rather than a straight line. As we know that normal Q-Q plots which looks like the depicted graph above usually means that sample data are skewed.

##Corn Seeding Data: 2018

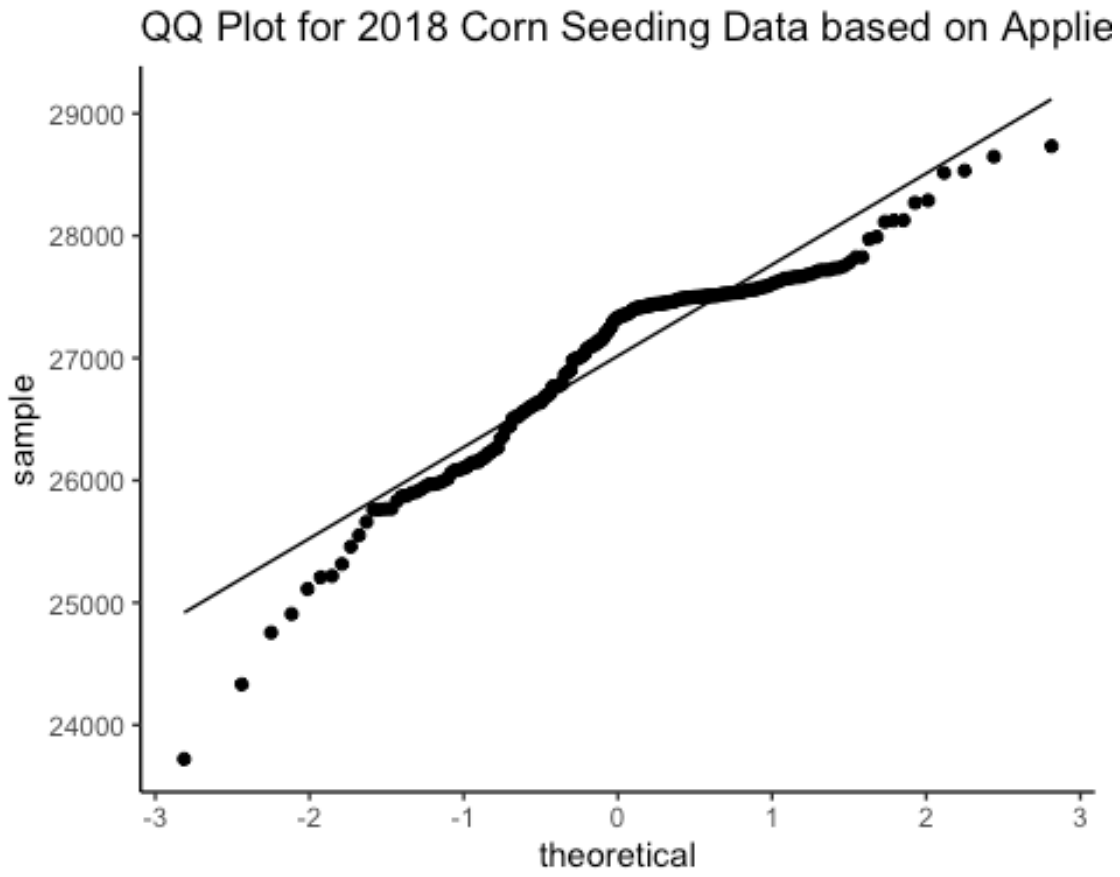
```
#Loading 2018 Corn Seeding Data
CornSeeding.2018<- read.csv("~/OneDrive - South Dakota State University - SDSU/STAT 600/Final Project/A 2018 Corn Seeding.csv")

#Plotting Data
plot(Latitude ~ Longitude,data=CornSeeding.2018,pch = ".", main = '2018 Corn Seeding Data')
abline(h=1:12*50,v=1:20*50,col='red')
```




```r
#Creating Grid for 2018 Corn Seeding Data
Row <- ceiling(CornSeeding.2018[,3] / 50)
Column <- ceiling(CornSeeding.2018[,2] / 50)
Cell <- Row*1000 + Column
CornSeedingCombined.2018 <- cbind(Cell,CornSeeding.2018, Row,Column )
plot(Row ~ Column,data=CornSeedingCombined.2018, main= '2018 Corn Seeding
with Grid Cell')
abline(h=1:12+0.5,v=1:20+0.5,col='red')
```



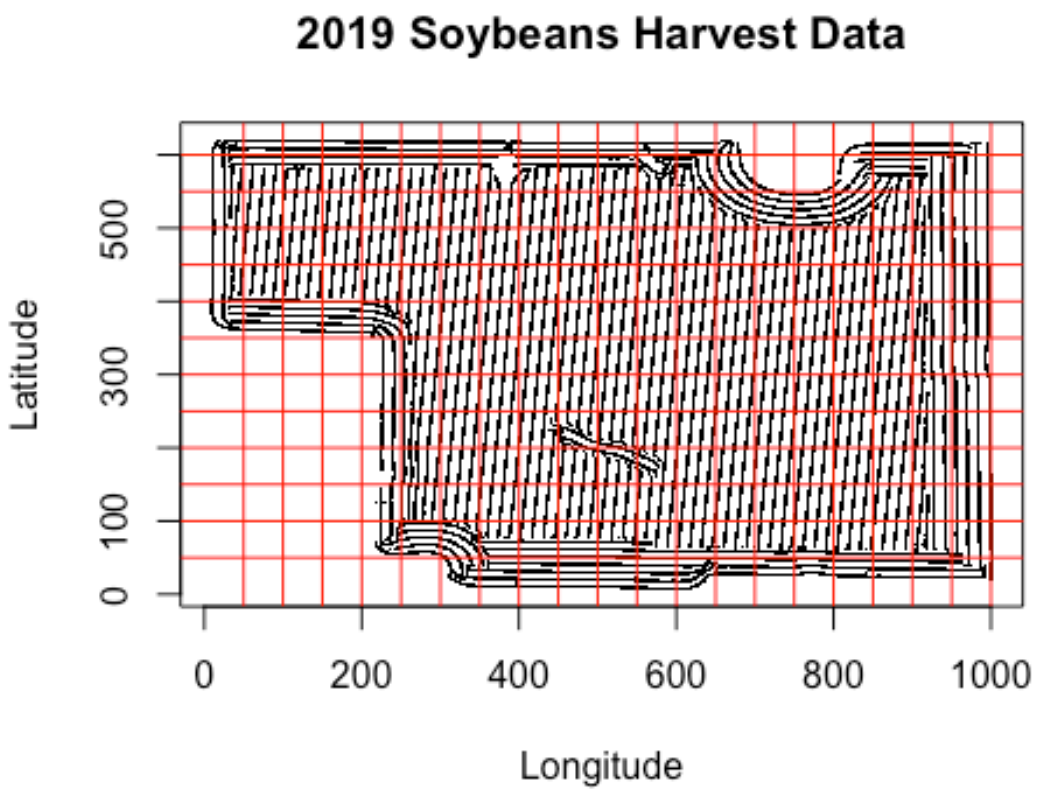

```
#Aggregating 2018 Corn Seeding Data based on AppliedRate
CornSeeding.Aggregate.2018 <-
AggregateField(CornSeedingCombined.2018,response='AppliedRate')
CornSeeding.Aggregate.2018<-
CornSeeding.Aggregate.2018[CornSeeding.Aggregate.2018$Samples>30,]
CornSeeding.Aggregate.2018$Cell <- CornSeeding.Aggregate.2018$Row*1000 +
CornSeeding.Aggregate.2018$Column
names(CornSeeding.Aggregate.2018)[3] <- 'AR18'
#Plotting QQ plot
ggplot(CornSeeding.Aggregate.2018, aes(sample = AR18)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2018 Corn Seeding Data based on
AppliedRate") + theme_classic()
```



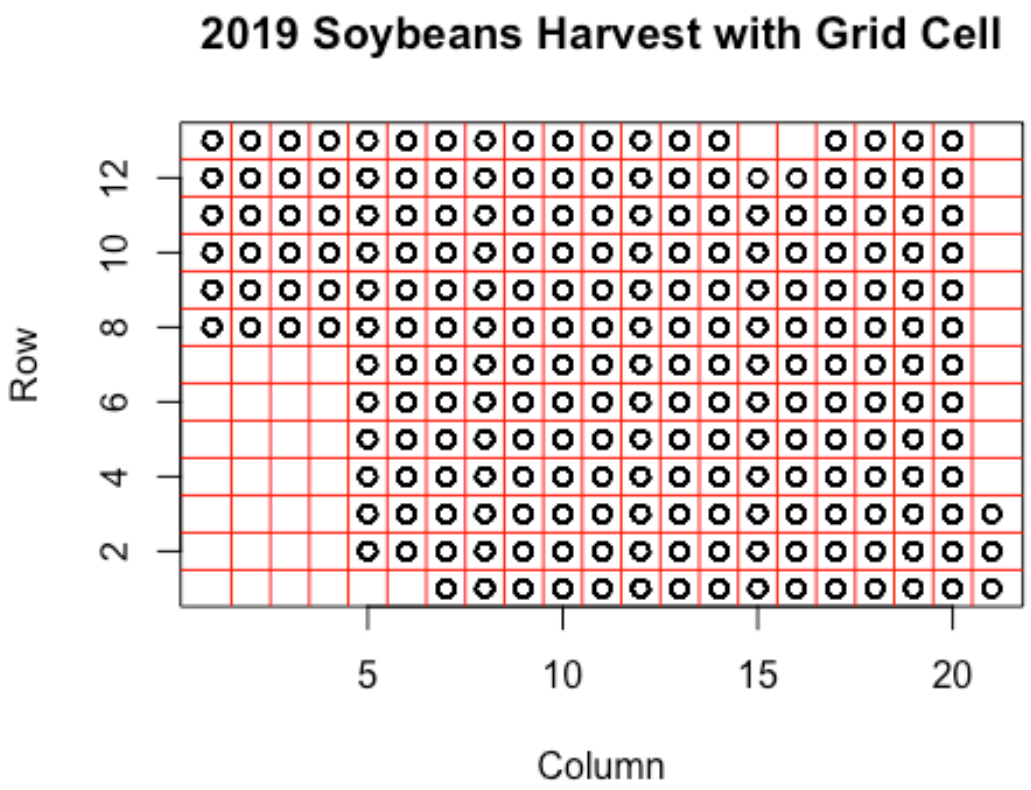
For 2018 Corn Seeding Data based on AppliedRate, the QQ plot doesn't follow the straight line which clearly indicates that data is skewed.

##Soybeans Harvest Data: 2019

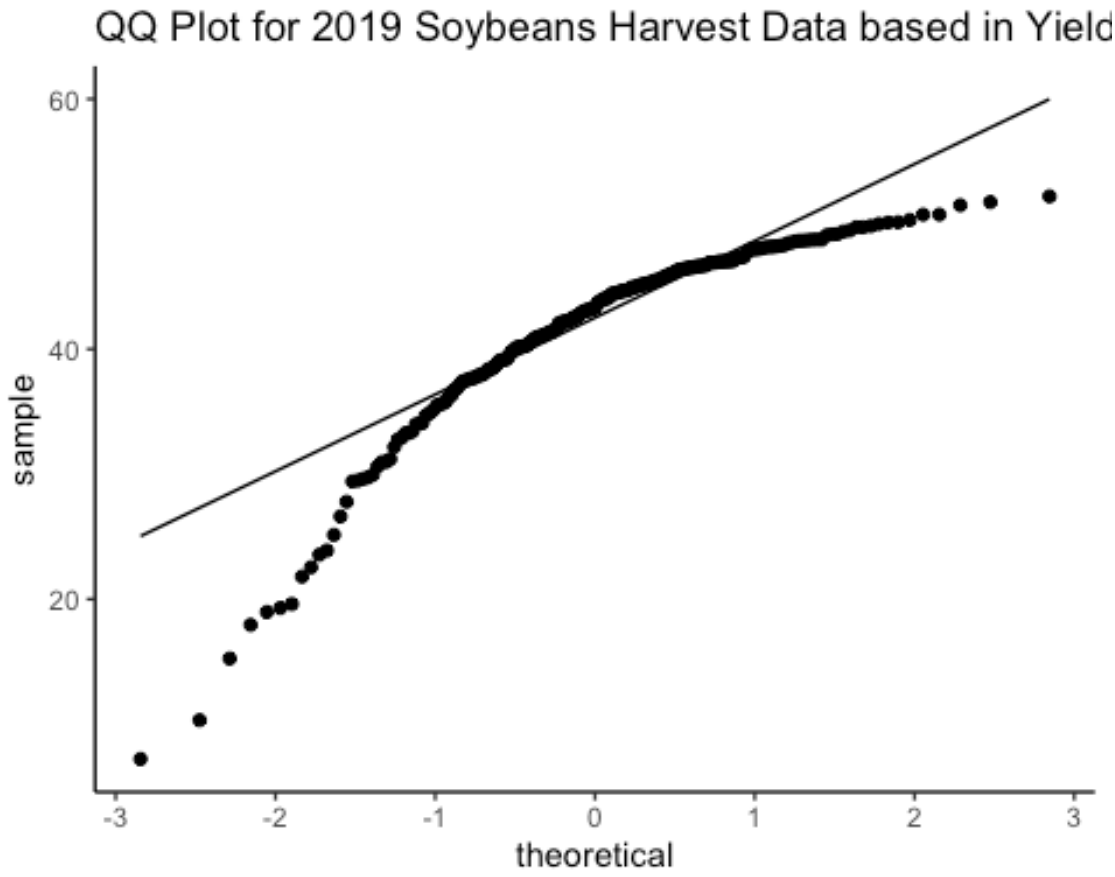
```
#Loading 2019 Soybeans Harvest Data
SoybeansHarvest.2019<- read.csv("~/OneDrive - South Dakota State University - SDSU/STAT 600/Final Project/A 2019 Soybeans Harvest.csv")
#Plotting Data
plot(Latitude ~ Longitude,data=SoybeansHarvest.2019,pch = ".", main = '2019 Soybeans Harvest Data')
abline(h=1:12*50,v=1:20*50,col='red')
```




```r
#Creating Grid 2019 Soybeans Harvest Data
Row <- ceiling(SoybeansHarvest.2019[,3] / 50)
Column <- ceiling(SoybeansHarvest.2019[,2] / 50)
Cell <- Row*1000 + Column
SoybeansHarvestCombined.2019 <- cbind(Cell,SoybeansHarvest.2019, Row,Column )
plot(Row ~ Column,data=SoybeansHarvestCombined.2019, main = '2019 Soybeans
Harvest with Grid Cell')
abline(h=1:12+0.5,v=1:20+0.5,col='red')
```



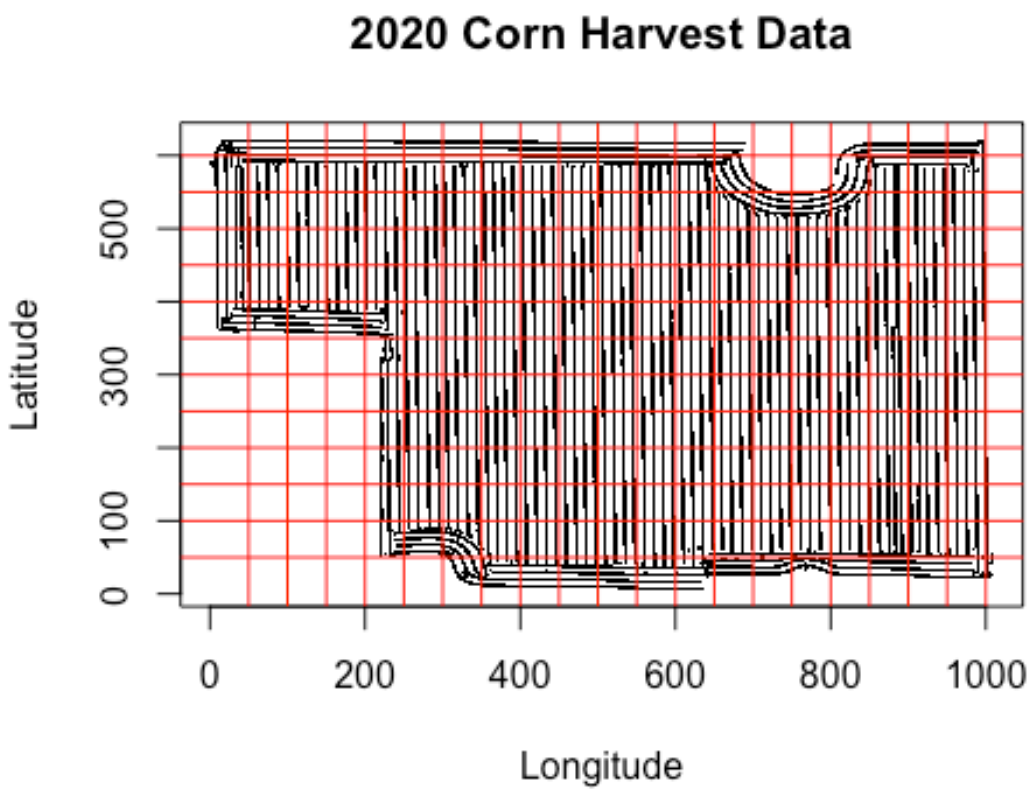

```
#Aggregating 2019 Soybeans Harvest Data based in Yield
SoybeansHarvest.Aggregate.2019 <-
AggregateField(SoybeansHarvestCombined.2019,response='Yield')
SoybeansHarvest.Aggregate.2019 <-
SoybeansHarvest.Aggregate.2019[SoybeansHarvest.Aggregate.2019$Samples>30,]
SoybeansHarvest.Aggregate.2019$Cell <-
SoybeansHarvest.Aggregate.2019$Row*1000 +
SoybeansHarvest.Aggregate.2019$Column
names(SoybeansHarvest.Aggregate.2019)[3] <- 'Y19'
#Plotting QQ plot
ggplot(SoybeansHarvest.Aggregate.2019, aes(sample = Y19)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2019 Soybeans Harvest Data based
in Yield") + theme_classic()
```



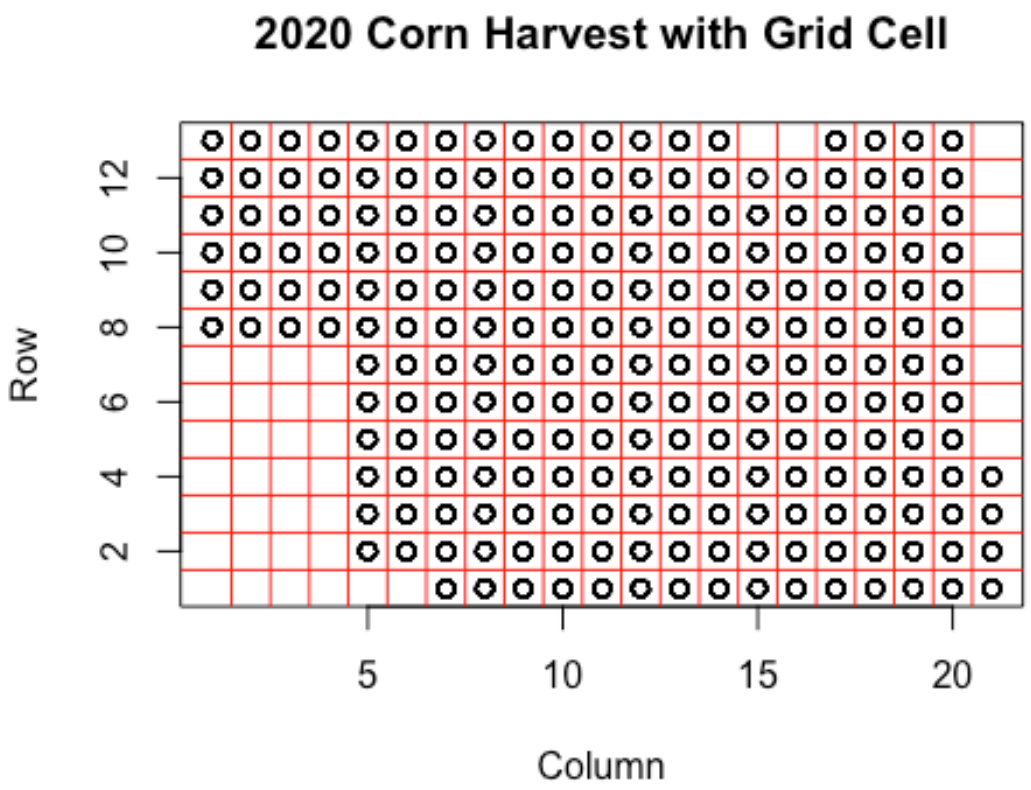
For 2019 Soybeans Harvest Data based in Yield, the QQ plot shows the points follow straight line with some outliers.

Corn Harvest Data: 2020

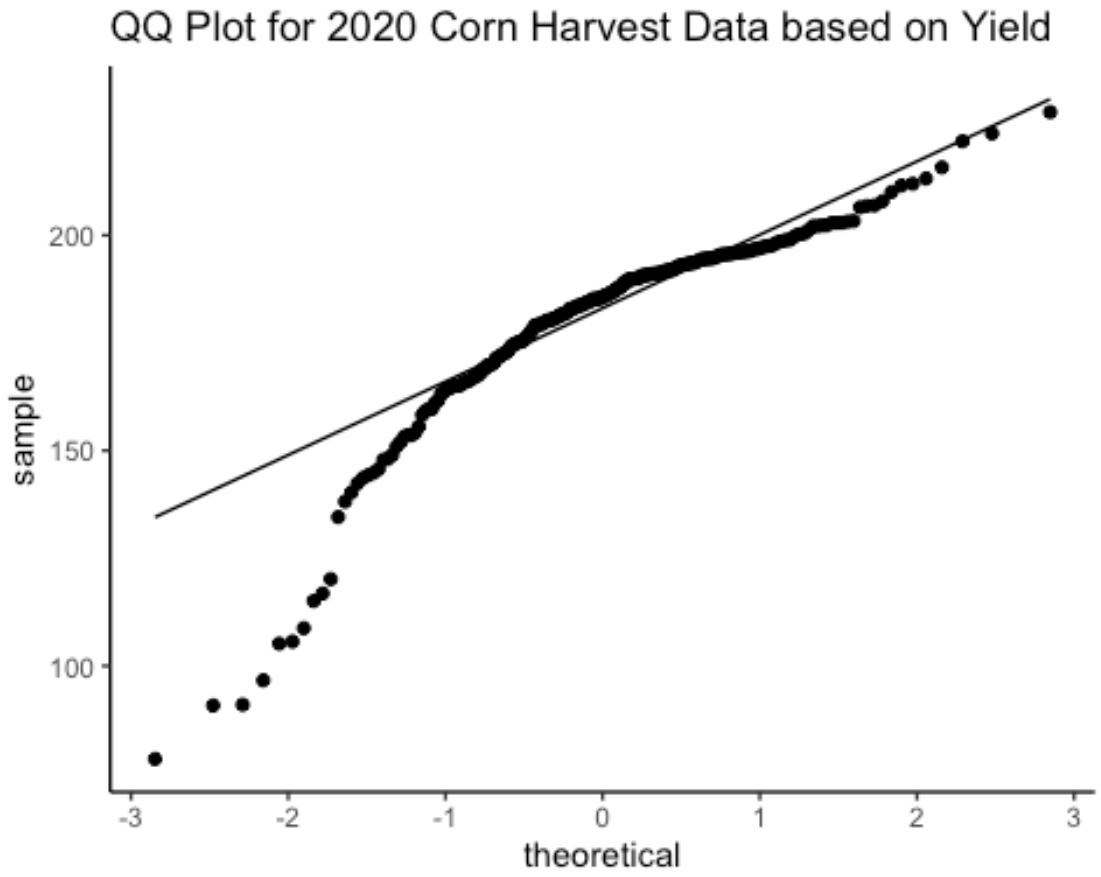
```
#Loading 2020 Corn Harvest Data
CornHarvest.2020<- read.csv("~/OneDrive - South Dakota State University - SDSU/STAT 600/Final Project/A 2020 Corn Harvest.csv")
#Plotting Data
plot(Latitude ~ Longitude,data=CornHarvest.2020,pch = ".", main = '2020 Corn Harvest Data')
abline(h=1:12*50,v=1:20*50,col='red')
```




```r
#Creating Grid for 2020 Corn Harvest Data
Row <- ceiling(CornHarvest.2020[,3] / 50)
Column <- ceiling(CornHarvest.2020[,2] / 50)
Cell <- Row*1000 + Column
CornHarvestCombined.2020 <- cbind(Cell,CornHarvest.2020, Row,Column )
plot(Row ~ Column,data=CornHarvestCombined.2020, main = '2020 Corn Harvest
with Grid Cell')
abline(h=1:12+0.5,v=1:20+0.5,col='red')
```



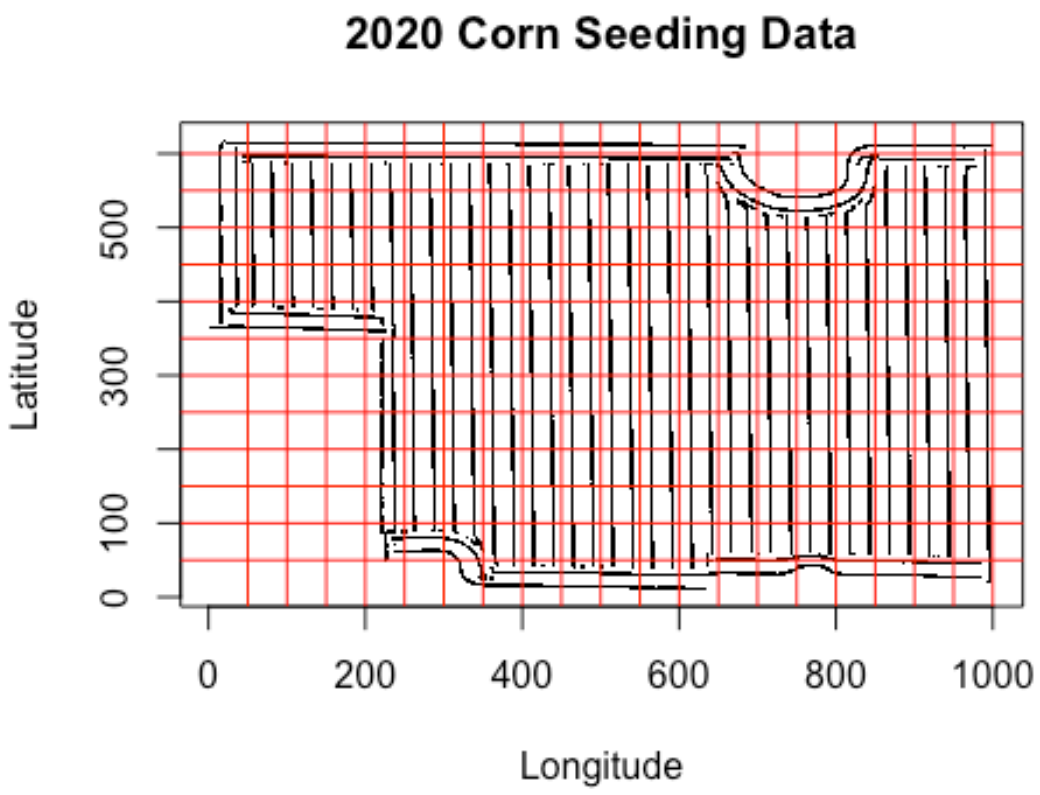

```
#Aggregating 2020 Corn Harvest Data based on Yield
CornHarvestAggregate.2020 <-
AggregateField(CornHarvestCombined.2020,response='Yield')
CornHarvestAggregate.2020 <-
CornHarvestAggregate.2020[CornHarvestAggregate.2020$Samples>30,]
CornHarvestAggregate.2020$Cell <- CornHarvestAggregate.2020$Row*1000 +
CornHarvestAggregate.2020$Column
names(CornHarvestAggregate.2020)[3] <- 'Y20'

#Plotting QQ plot
ggplot(CornHarvestAggregate.2020, aes(sample = Y20)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2020 Corn Harvest Data based on
Yield") + theme_classic()
```



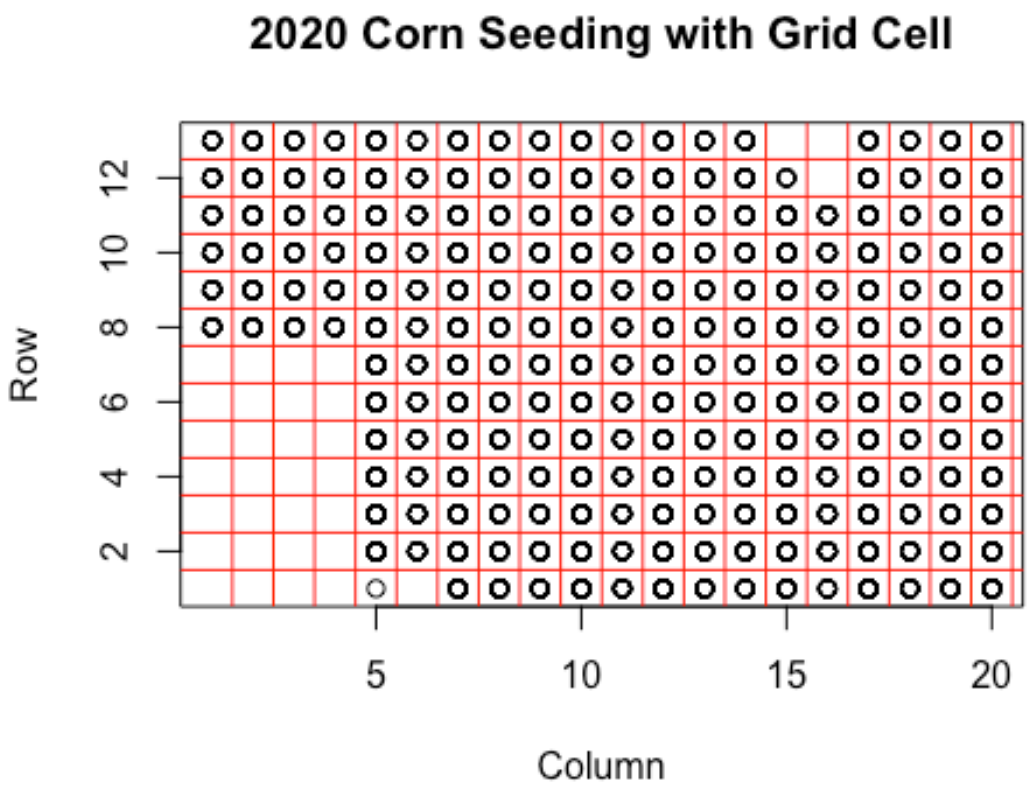
The points in the centre of the graph fall along a line, but they curve off towards the extremes. When we see this behavior in a Normal Q-Q plot, it typically means that the data has more extreme values than would be anticipated if it came from a true Normal distribution.

Corn Seeding Data: 2020

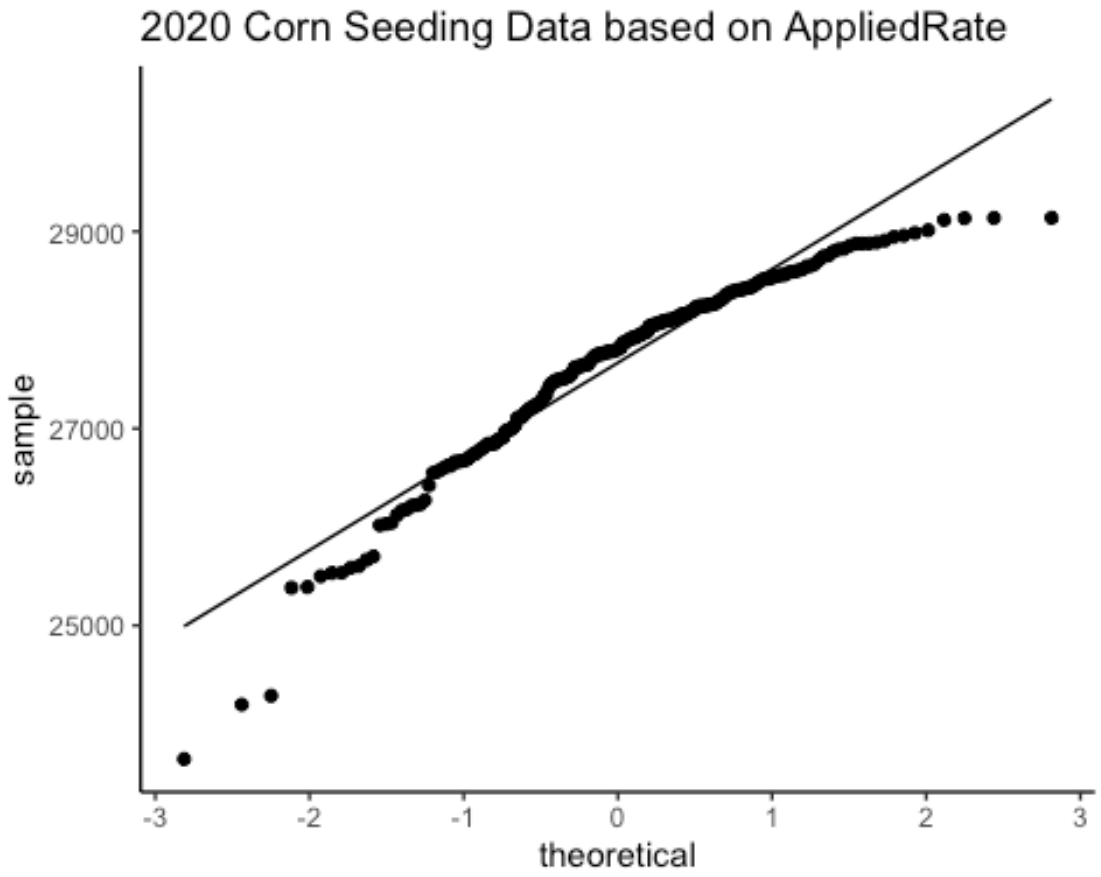
```
#Loading 2020 Corn Seeding Data
CornSeeding.2020<- read.csv("~/OneDrive - South Dakota State University - SDSU/STAT 600/Final Project/A 2020 Corn Seeding.csv")

#Plotting Data
plot(Latitude ~ Longitude,data=CornSeeding.2020,pch = ".", main = '2020 Corn Seeding Data')
abline(h=1:12*50,v=1:20*50,col='red')
```




```r
#Creating Grid for 2020 Corn Seeding Data
Row <- ceiling(CornSeeding.2020[,3] / 50)
Column <- ceiling(CornSeeding.2020[,2] / 50)
Cell <- Row*1000 + Column
CornSeedingCombined.2020 <- cbind(Cell,CornSeeding.2020, Row,Column )
plot(Row ~ Column,data=CornSeedingCombined.2020, main= '2020 Corn Seeding
with Grid Cell')
abline(h=1:12+0.5,v=1:20+0.5,col='red')
```




```
#Aggregating 2020 Corn Seeding Data based on AppliedRate
CornSeedingAggregate.2020 <-
AggregateField(CornSeedingCombined.2020,response='AppliedRate')
CornSeedingAggregate.2020 <-
CornSeedingAggregate.2020[CornSeedingAggregate.2020$Samples>30,]
CornSeedingAggregate.2020$Cell <- CornSeedingAggregate.2020$Row*1000 +
CornSeedingAggregate.2020$Column
names(CornSeedingAggregate.2020)[3] <- 'AR20'

#Plotting QQ plot
ggplot(CornSeedingAggregate.2020, aes(sample = AR20)) +
  stat_qq()  +
  stat_qq_line() +
  labs(title="2020 Corn Seeding Data based on AppliedRate") + theme_classic()
```

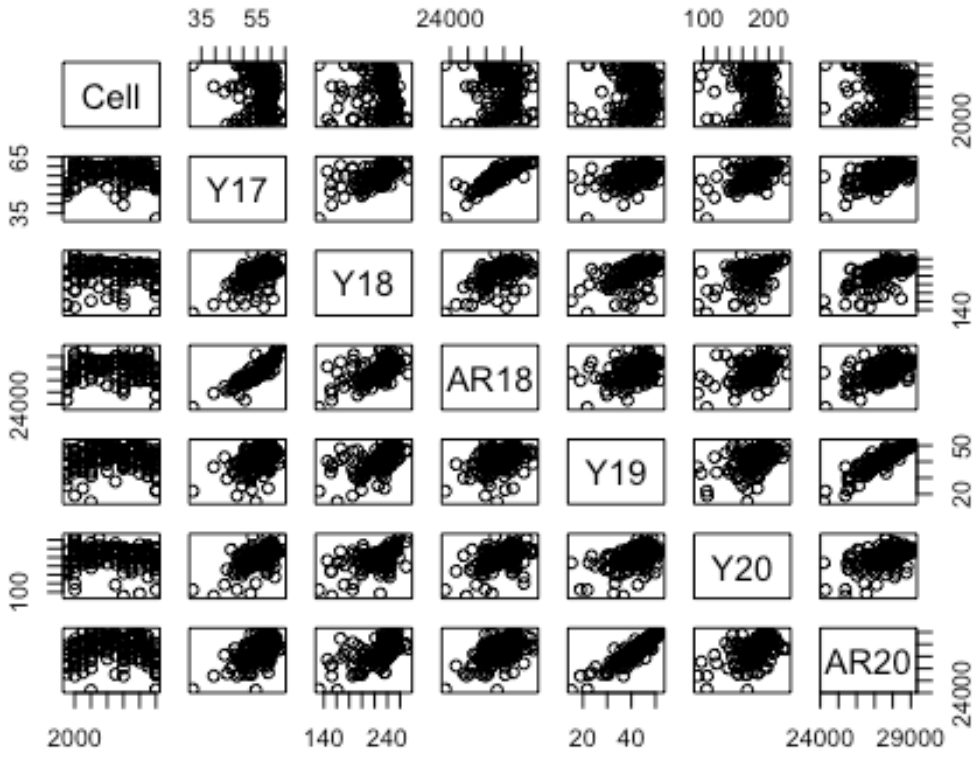


The points in the centre of the graph fall along a line, but they curve off towards the extremes. When we see this behavior in a Normal Q-Q plot, it typically means that the data has more extreme values than would be anticipated if it came from a true Normal distribution.

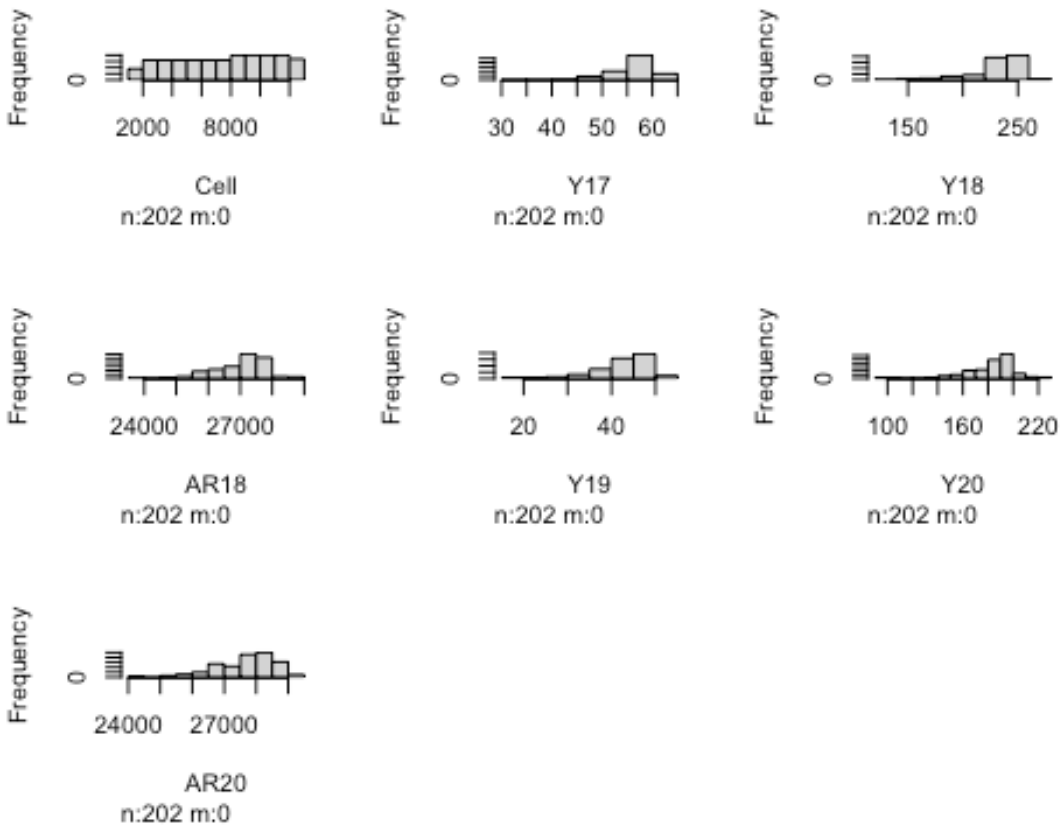
##Data Merging

```
#Merging Soybeans and Corn Data between 2017 and 2018
Combined_SoybeansCorn1 <- merge(Soybeans.Aggregate.2017,Corn.Aggregate.2018,
by="Cell")

#Merging Soybeans and Corn Data between 2018 (AppliedRate) and 2019
Combined_SoybeansCorn2 <-
merge(CornSeeding.Aggregate.2018,SoybeansHarvest.Aggregate.2019, by="Cell")

#Merging Data between 2017 and 2018 and 2018 (AppliedRate) and 2019
Combined_SoybeansCorn12 <-
merge(Combined_SoybeansCorn1,Combined_SoybeansCorn2, by="Cell")

#Merging Data between 2020
Combined_Corn<- merge(CornHarvestAggregate.2020,CornSeedingAggregate.2020,
by="Cell")

#Merging Data all Data
```

```r
Combined.all <- merge(Combined_SoybeansCorn12,Combined_Corn, by="Cell")

#Removing Unneccesary Coloumns
column_needs <- c("Cell","Y17","Y18","AR18","Y19","Y20","AR20")
Combined.dat <- Combined.all[column_needs]
plot(Combined.dat)
```




```r
#Plotting Histrogram
hist.data.frame(Combined.dat)
```

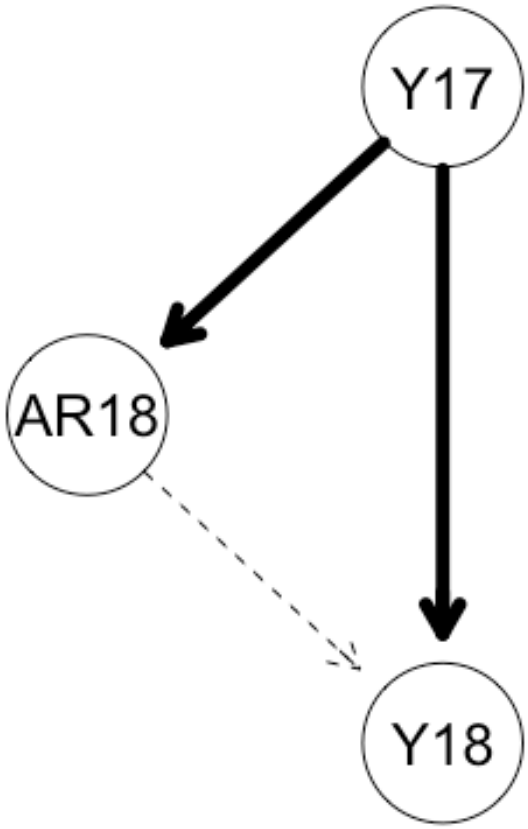


Description:

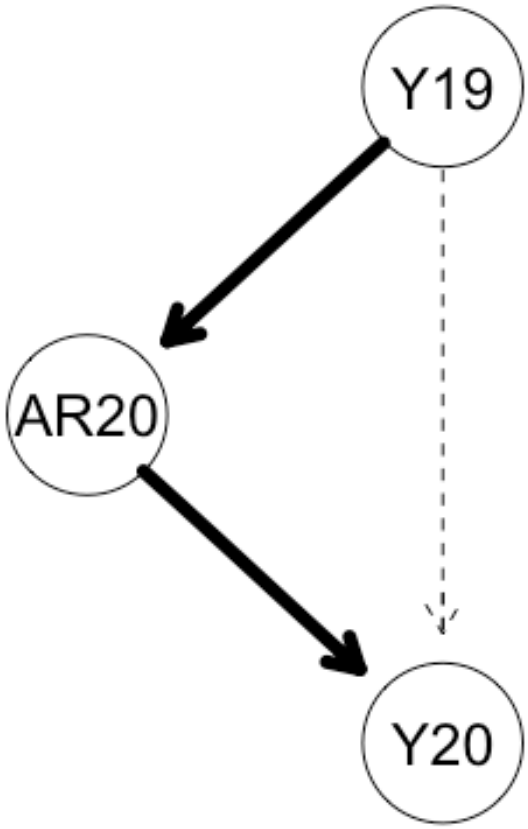
- From the histrogram plots depicted above, we can celarly see that almost every aggregated variables show a Left skewed histogram. This shape implies that any outlines have a lower predominance than the mean.
- From the pair plot mentioned above, we can see a strong linear relationship between yield rate of 2017 and applied rate of 2018. Also similar kind of strong linear relationship can be found between yield rate of 2019 and applied rate of 2020.

Causal Inference

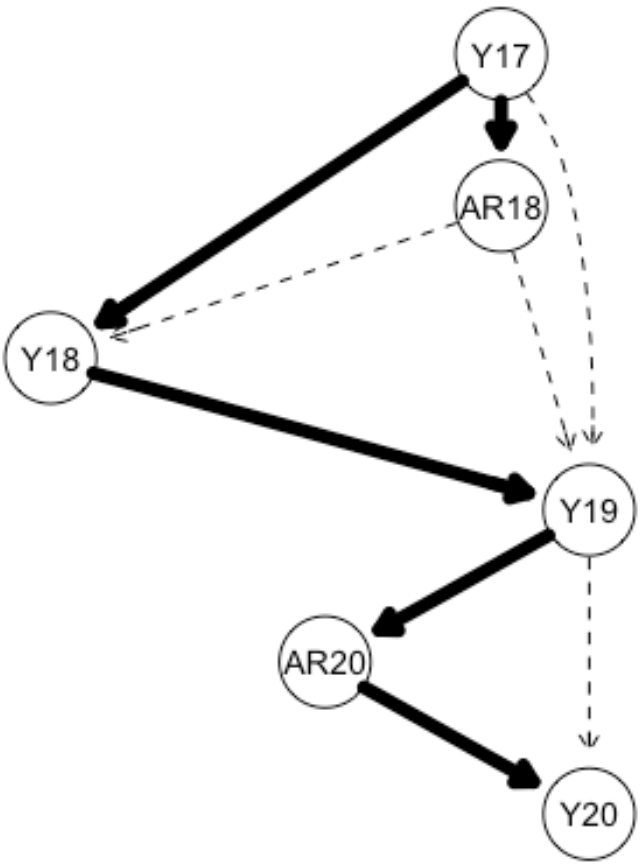
```
#Directed Acyclic Graphs

modela.dag <- model2network("[Y17][AR18|Y17][Y18|AR18:Y17]")
fit1 = bn.fit(modela.dag, Combined.dat[,c('Y17','AR18','Y18')])
#fit1

strengtha <- arc.strength(modela.dag, Combined.dat[,c('Y17','AR18','Y18')])
strength.plot(modela.dag, strengtha)
```




```
modelb.dag <- model2network("[Y19][AR20|Y19][Y20|AR20:Y19]")
fit2 = bn.fit(modelb.dag, Combined.dat[,c('Y19','AR20','Y20')])
#fit2
strengthb <- arc.strength(modelb.dag, Combined.dat[,c('Y19','AR20','Y20')])
strength.plot(modelb.dag, strengthb)
```




```
model1.dag <-
model2network("[Y17][AR18|Y17][Y18|AR18:Y17][Y19|Y17:AR18:Y18][AR20|Y19][Y20|
AR20:Y19]")
fit3 = bn.fit(model1.dag,
Combined.dat[,c('Y17','AR18','Y18','Y19','AR20','Y20')])
#fit3
strength1 <- arc.strength(model1.dag,
Combined.dat[,c('Y17','AR18','Y18','Y19','AR20','Y20')])
strength.plot(model1.dag, strength1)
```



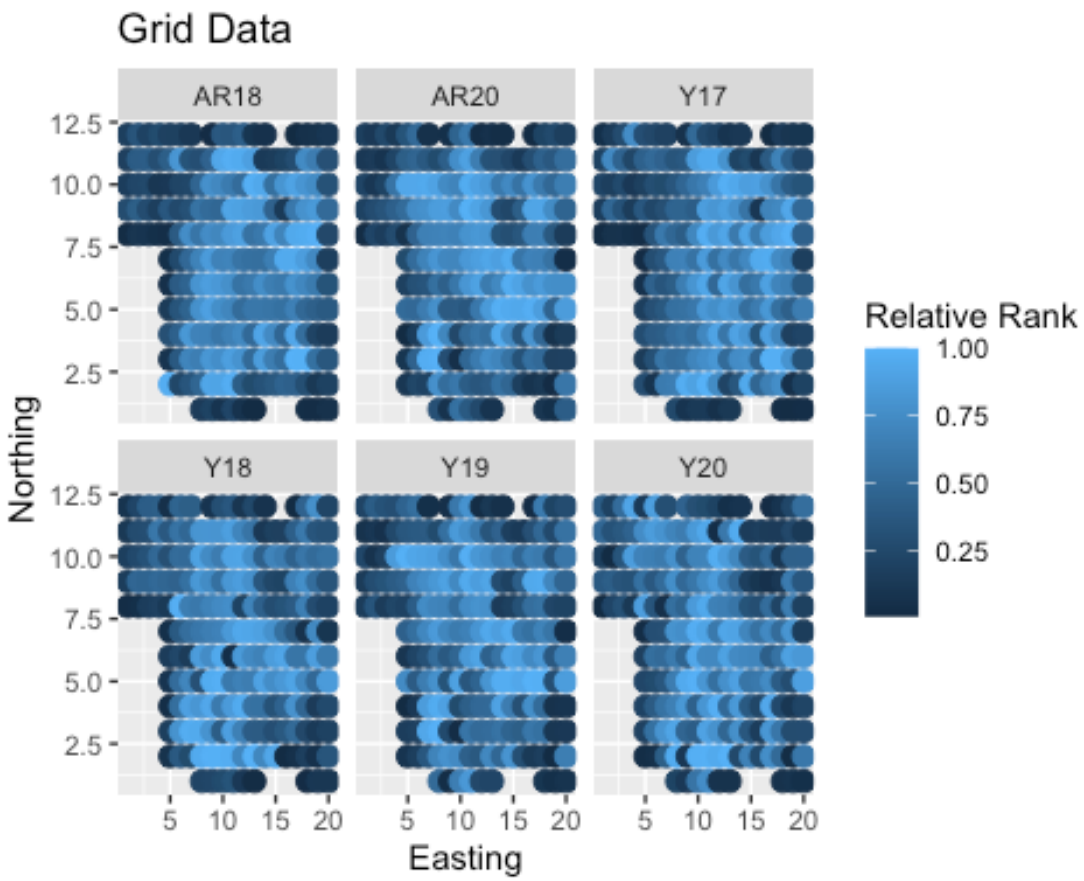
Description:

From the Acyclic Graphs depicted above we have seen that there is strong relation between yield rate of 2017 and applied rate of 2018. But we can not assert any relationship between Applied Rate of 2018 and Yiled rate of 2018 because they are lightly connected. In another way, We have also seen that there is also a relation between yield rate of 2017 and yield rate of 2018. We can see there is strong relationship between yield rate of 2019 and applied rate of 2020. Additionally, it has also strongly related between applied rate of 2020 and yield rate of 2020. But, yield rate of 2019 and yield rate of 2020 is lightly connected. Also, yield rate of 2019 and yield rate of 2020 is lightly connected.

Rank after Aggregrating data

```
Combined.all$Y17r <- rank(Combined.all$Y17)/max(rank(Combined.all$Y17))
Combined.all$Y18r <- rank(Combined.all$Y18)/max(rank(Combined.all$Y18))
Combined.all$AR18r <- rank(Combined.all$AR18)/max(rank(Combined.all$AR18))
Combined.all$Y19r <- rank(Combined.all$Y19)/max(rank(Combined.all$Y19))
Combined.all$Y20r <- rank(Combined.all$Y20)/max(rank(Combined.all$Y20))
Combined.all$AR20r <- rank(Combined.all$AR20)/max(rank(Combined.all$AR20))



GridMaps <- data.frame(Row=c(Combined.all$Row.x.x),
```

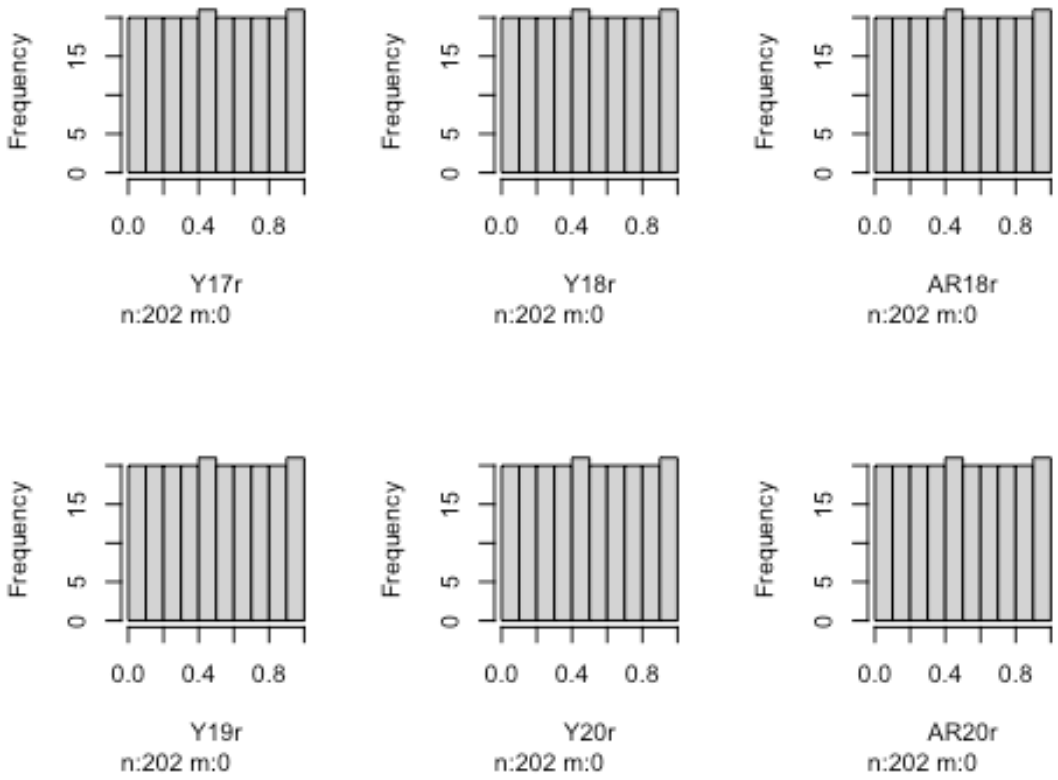
```
                    Column=c(Combined.all$Column.x.x),

Value=c(Combined.all$Y17r,Combined.all$Y18r,Combined.all$AR18r,Combined.all$Y
19r,Combined.all$Y20r,Combined.all$AR20r),
                    Map=c(rep('Y17',length(Combined.all$Y17r)),
                          rep('Y18',length(Combined.all$Y18r)),
                          rep('AR18',length(Combined.all$AR18r)),
                          rep('Y19',length(Combined.all$Y19r)),
                          rep('Y20',length(Combined.all$Y20r)),
                          rep('AR20',length(Combined.all$AR20r))))
ggplot(GridMaps, aes(Column,Row)) +
geom_point(aes(colour = Value),size=3) +
labs(colour = "Relative Rank", x="Easting", y="Northing", title = "Grid
Data") + facet_wrap(~ Map)
```



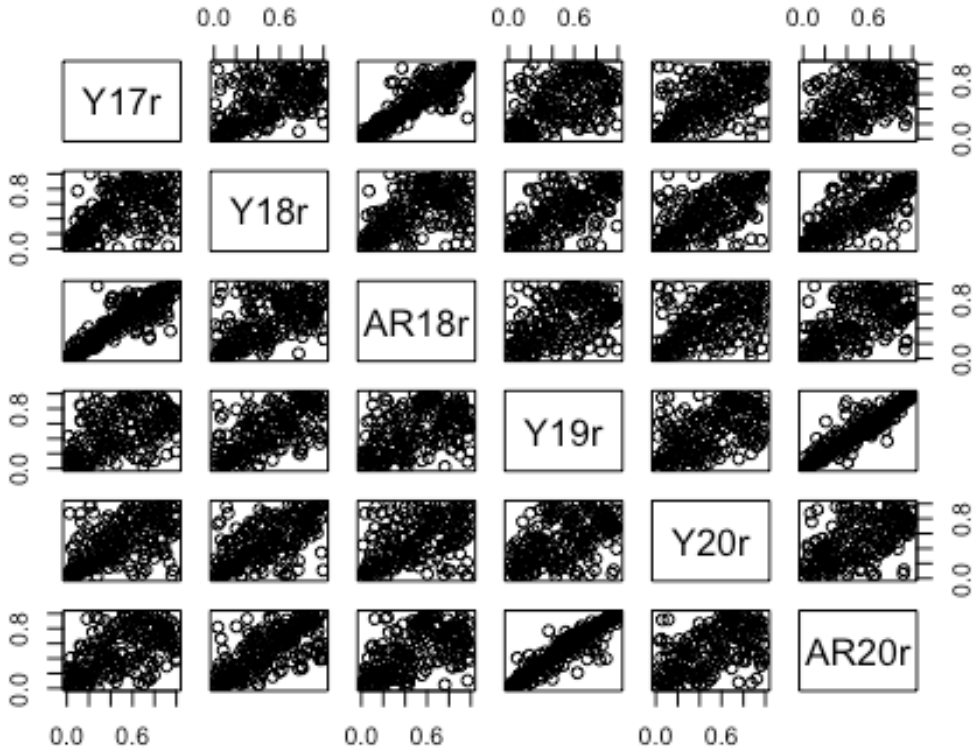
Here the above plots clearly shows the mean for each grid cell which are based on ranks.

Histogram and Pairs Plot after Rank

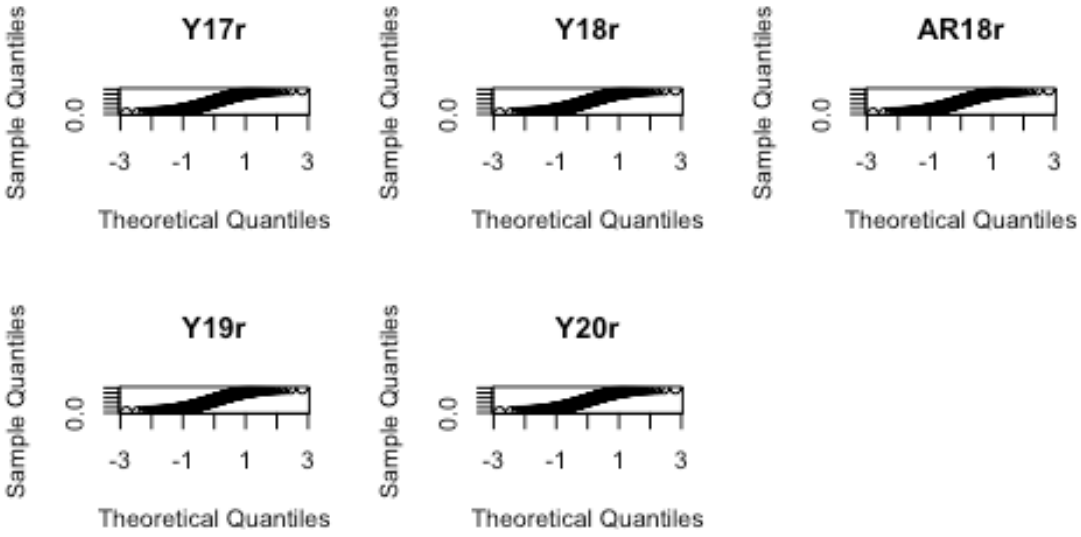
```
columns_rank <- c("Y17r","Y18r","AR18r","Y19r","Y20r","AR20r")
Combined.dat.rank <- Combined.all[columns_rank]

# Histogram
hist.data.frame(Combined.dat.rank)
```




```r
#pairs plot
plot(Combined.dat.rank)
```




```r
#QQ plot
par(mfrow=c(3,3))
for (i in 1:ncol(Combined.dat.rank[,1: ncol(Combined.dat.rank) - 1 ])){
qqnorm(Combined.dat.rank[, i], main = names(Combined.dat.rank[i]))
   qqline(Combined.dat.rank[, i])
}
```

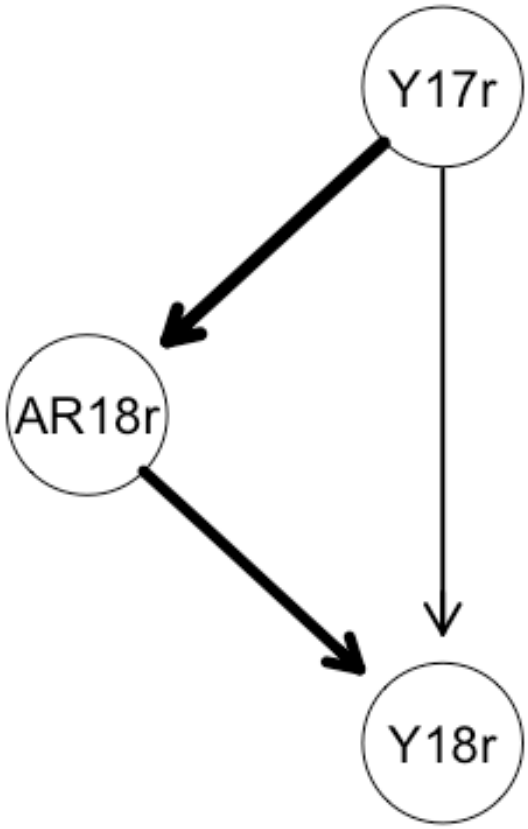


Description:

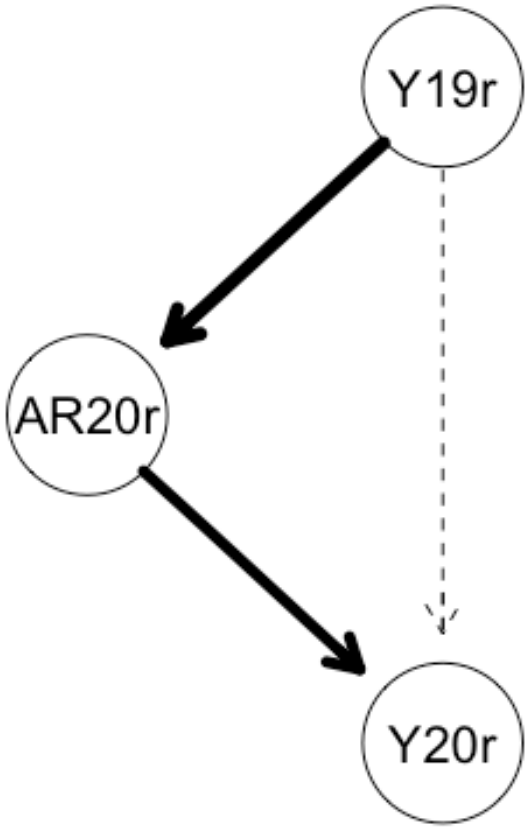
- Previously we have seen that almost every aggregated variables show a Left skewed histogram. But after ranking, we have seen that almost every aggregated variables show a uniform histogram indicating normal distribution.
- From the pair plot mentioned above, we can see strong linear relationship between yield rate of 2017 and applied rate of 2018. Also similar kind of strong linear relationship can found between yield rate of 2019 and applied rate of 2020.

Causal Inference After Rank

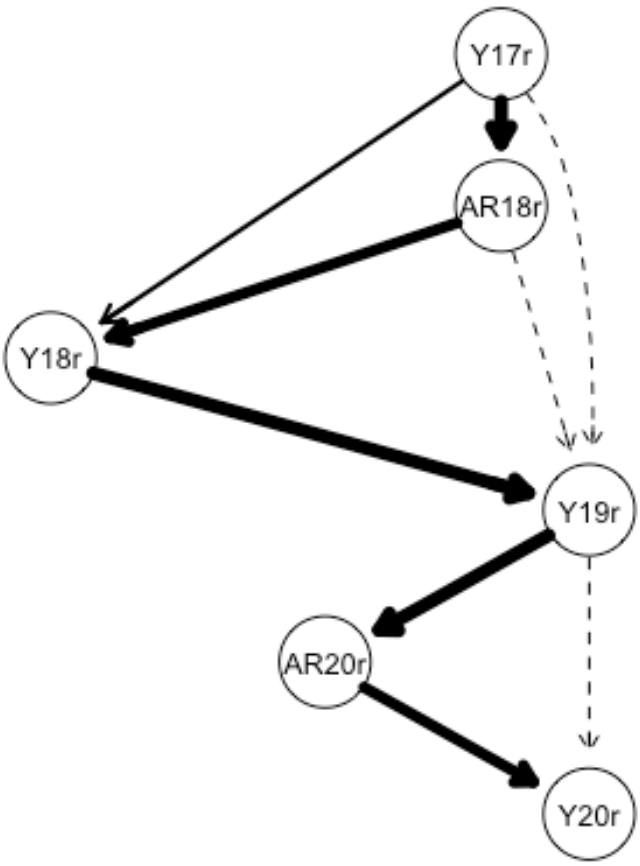
```
#Plotting Directed Acyclic Graphs

modela.dag <- model2network("[Y17r][AR18r|Y17r][Y18r|AR18r:Y17r]")
fit1 = bn.fit(modela.dag, Combined.dat.rank[,c('Y17r','AR18r','Y18r')])
#fit1

strengtha <- arc.strength(modela.dag,
Combined.dat.rank[,c('Y17r','AR18r','Y18r')])
strength.plot(modela.dag, strengtha)
```




```
modelb.dag <- model2network("[Y19r][AR20r|Y19r][Y20r|AR20r:Y19r]")
fit2 = bn.fit(modelb.dag, Combined.dat.rank[,c('Y19r','AR20r','Y20r')])
#fit2
strengthb <- arc.strength(modelb.dag,
Combined.dat.rank[,c('Y19r','AR20r','Y20r')])
strength.plot(modelb.dag, strengthb)
```




```
model1.dagRa <-
model2network("[Y17r][AR18r|Y17r][Y18r|AR18r:Y17r][Y19r|Y17r:AR18r:Y18r][AR20
r|Y19r][Y20r|AR20r:Y19r]")
fit3 = bn.fit(model1.dagRa,
Combined.dat.rank[,c('Y17r','AR18r','Y18r','Y19r','AR20r','Y20r')])
#fit3
strength1Ra <- arc.strength(model1.dagRa,
Combined.dat.rank[,c('Y17r','AR18r','Y18r','Y19r','AR20r','Y20r')])
strength.plot(model1.dagRa, strength1Ra)
```



Description:

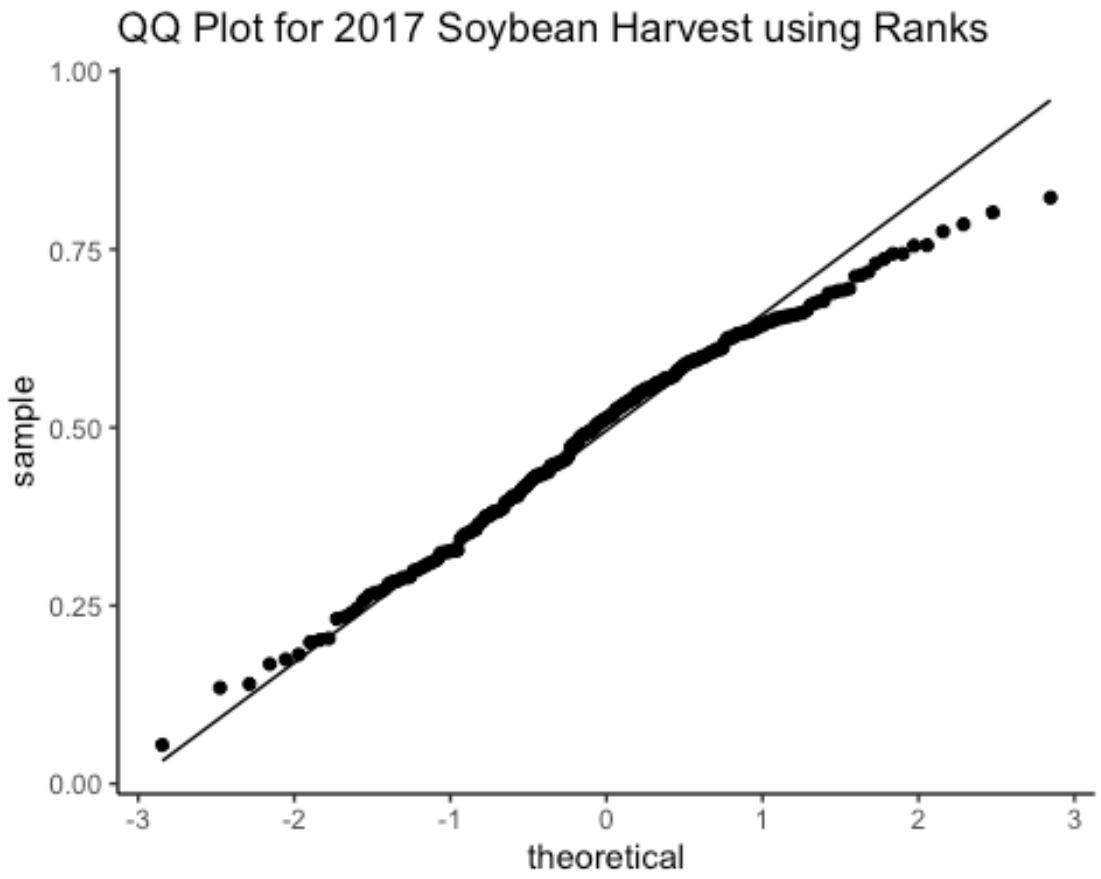
From the Acyclic Graphs depicted above we have seen that there is a strong relationship between yield rate of 2017 and applied rate of 2018 after applying rank. Although, yield rate of 2017 and yield rate of 2018 are lightly related. Yiled rate of 2019 and Applied Rate of 2020 have also shown strong relationship.Whereas, yield rate of 2019 and yield rate of 2020 is lightly connected.Again, Applied rate of 2018 and yield rate of 2018 is strongly connected. We can see there is a strong relationship between yield rate of 2019 and applied rate of 2020. It has also a relation between applied rate of 2020 and yield rate of 2020.

Rank before Aggreating Data

Normalization of 2017 Soybean Harvest using Ranks

```
#Normalization of 2017 Soybean Harvest using Ranks
SoybeansHarvestCombined.2017$Y17.r<-
rank(SoybeansHarvestCombined.2017$Yield)/max(rank(SoybeansHarvestCombined.201
7$Yield))
Soybeans.Aggregate.2017.rank <-
AggregateField(SoybeansHarvestCombined.2017,response='Y17.r')
Soybeans.Aggregate.2017.rank <-
Soybeans.Aggregate.2017.rank[Soybeans.Aggregate.2017.rank$Samples>30,]
Soybeans.Aggregate.2017.rank$Cell <- Soybeans.Aggregate.2017.rank$Row*1000 +
```

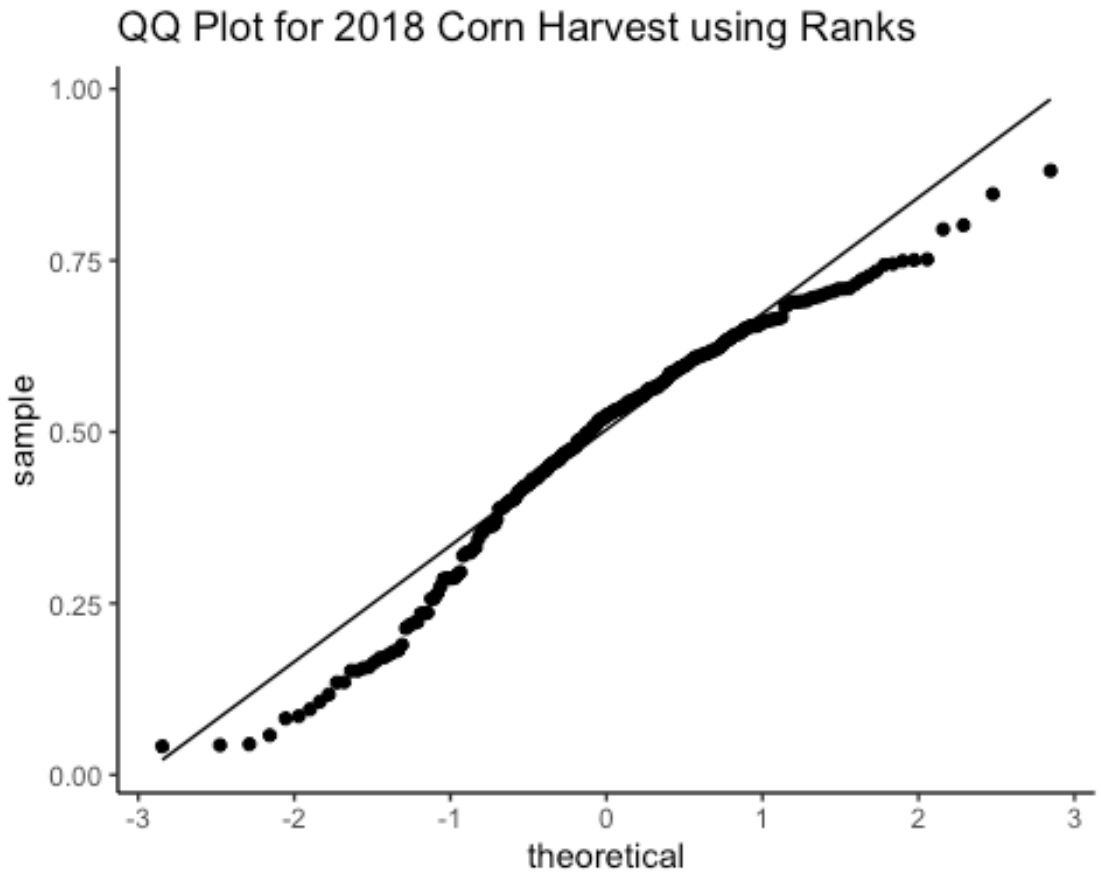
```
Soybeans.Aggregate.2017.rank$Column
names(Soybeans.Aggregate.2017.rank)[3] <- 'Y17.r'
#Plotting QQ plot
ggplot(Soybeans.Aggregate.2017.rank, aes(sample = Y17.r)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2017 Soybean Harvest using Ranks")
+ theme_classic()
```



Normalization of 2018 Corn Harvest using Ranks

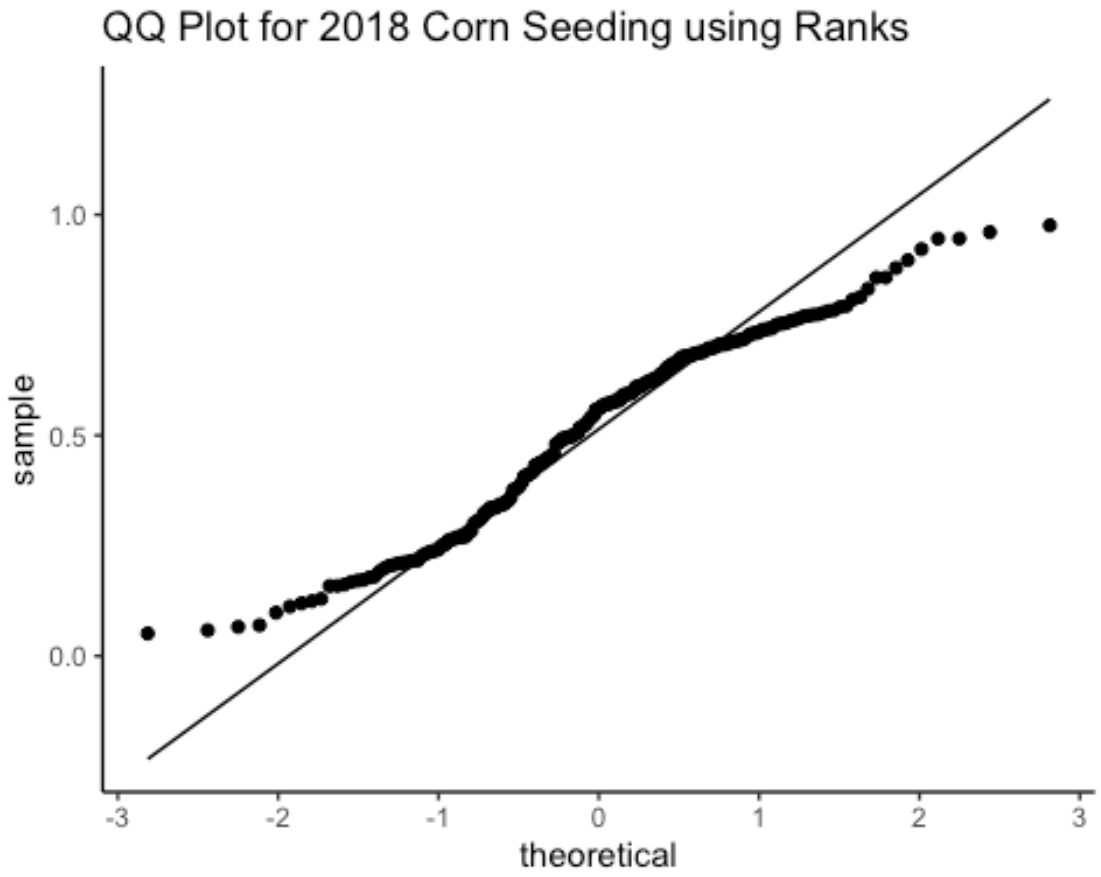
```
#Normalization of 2018 Corn Harvest using Ranks
CornHarvestCombined.2018$Y18.r<-
rank(CornHarvestCombined.2018$Yield)/max(rank(CornHarvestCombined.2018$Yield)
)
CornHarvestCombined.2018.rank <-
AggregateField(CornHarvestCombined.2018,response='Y18.r')
CornHarvestCombined.2018.rank <-
CornHarvestCombined.2018.rank[CornHarvestCombined.2018.rank$Samples>30,]
CornHarvestCombined.2018.rank$Cell <- CornHarvestCombined.2018.rank$Row*1000
+ CornHarvestCombined.2018.rank$Column
names(CornHarvestCombined.2018.rank)[3] <- 'Y18.r'
#Plotting QQ plot
ggplot(CornHarvestCombined.2018.rank, aes(sample = Y18.r)) +
  stat_qq()  +
```

```
  stat_qq_line() + labs(title="QQ Plot for 2018 Corn Harvest using Ranks") +
theme_classic()
```



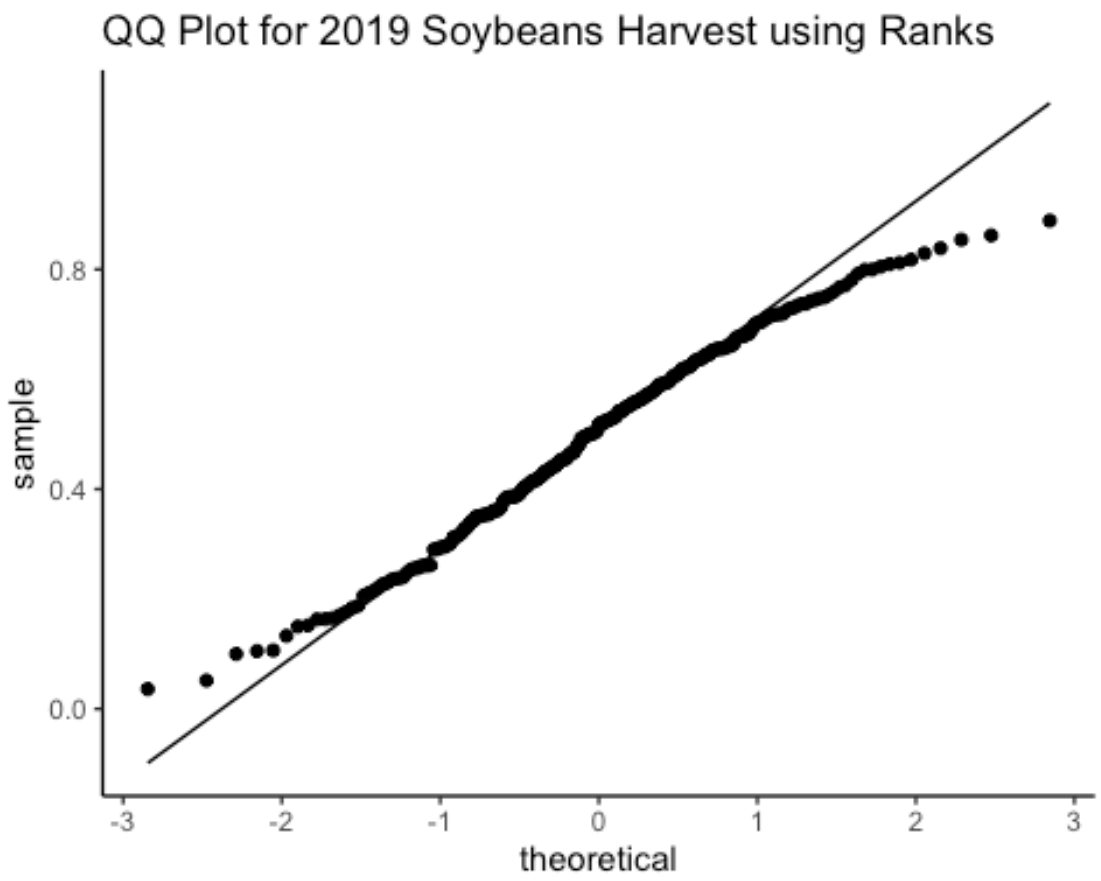
Normalization of 2018 Corn Seeding using Ranks

```
#Normalization of 2018 Corn Seeding using Ranks
CornSeedingCombined.2018$AR18.r<-
rank(CornSeedingCombined.2018$AppliedRate)/max(rank(CornSeedingCombined.2018$
AppliedRate))
CornSeedingCombined.2018.rank <-
AggregateField(CornSeedingCombined.2018,response='AR18.r')
CornSeedingCombined.2018.rank <-
CornSeedingCombined.2018.rank[CornSeedingCombined.2018.rank$Samples>30,]
CornSeedingCombined.2018.rank$Cell <- CornSeedingCombined.2018.rank$Row*1000
+ CornSeedingCombined.2018.rank$Column
names(CornSeedingCombined.2018.rank)[3] <- 'AR18.r'
#Plotting QQ plot
ggplot(CornSeedingCombined.2018.rank, aes(sample = AR18.r)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2018 Corn Seeding using Ranks") +
theme_classic()
```



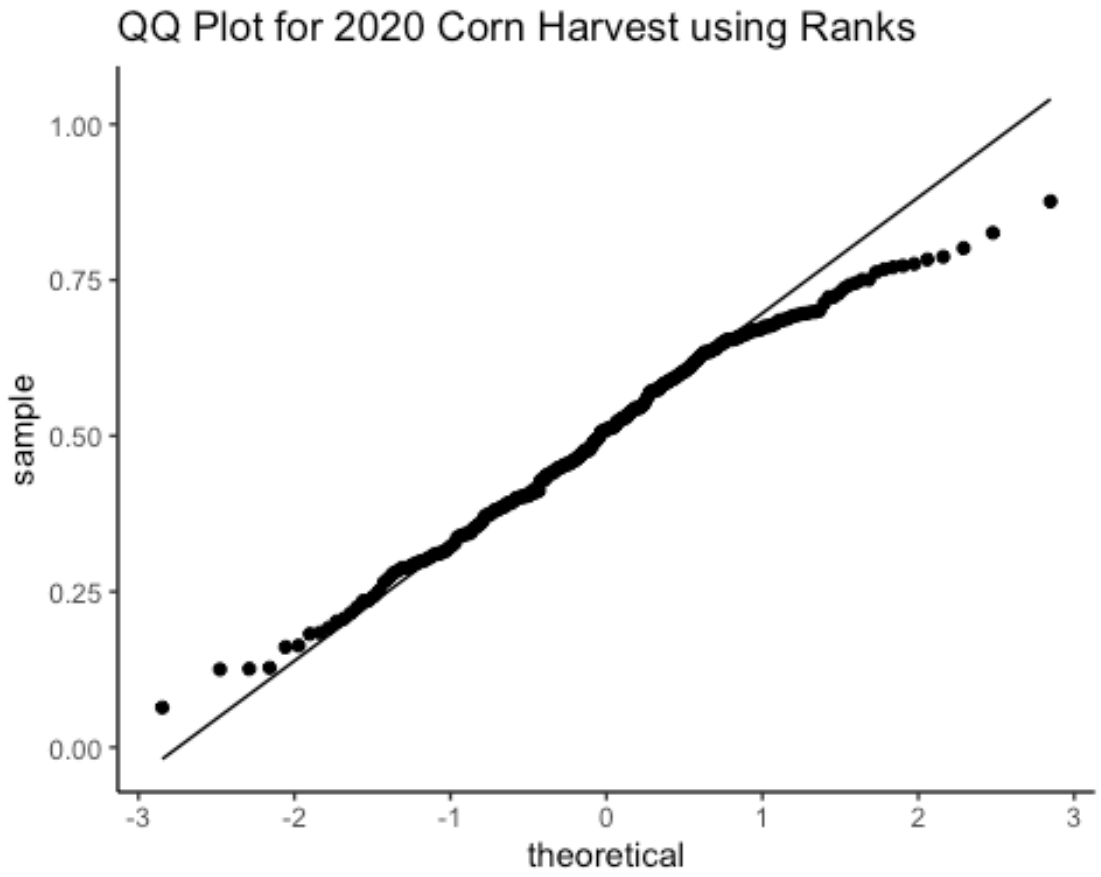
Normalization of 2019 Soybeans Harvest using Ranks

```
#Normalization of 2019 Soybeans Harvest using Ranks
SoybeansHarvestCombined.2019$Y19.r<-
rank(SoybeansHarvestCombined.2019$Yield)/max(rank(SoybeansHarvestCombined.201
9$Yield))
SoybeansHarvestCombined.2019.rank <-
AggregateField(SoybeansHarvestCombined.2019,response='Y19.r')
SoybeansHarvestCombined.2019.rank <-
SoybeansHarvestCombined.2019.rank[SoybeansHarvestCombined.2019.rank$Samples>3
0,]
SoybeansHarvestCombined.2019.rank$Cell <-
SoybeansHarvestCombined.2019.rank$Row*1000 +
SoybeansHarvestCombined.2019.rank$Column
names(SoybeansHarvestCombined.2019.rank)[3] <- 'Y19.r'

#Plotting QQ plot
ggplot(SoybeansHarvestCombined.2019.rank, aes(sample = Y19.r)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2019 Soybeans Harvest using
Ranks") + theme_classic()
```



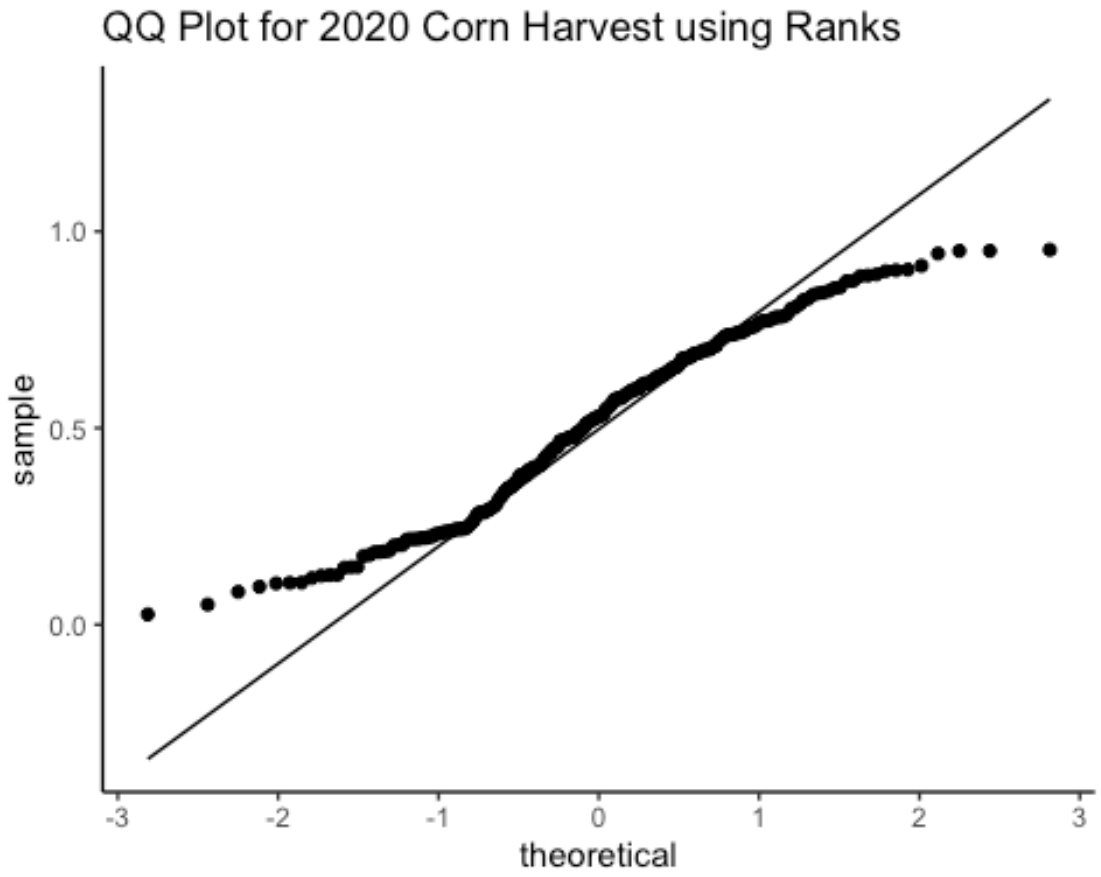
Normalization of 2020 Corn Harvest using Ranks

```
#Normalization of 2020 Corn Harvest using Ranks
CornHarvestCombined.2020$Y20.r<-
rank(CornHarvestCombined.2020$Yield)/max(rank(CornHarvestCombined.2020$Yield)
)
CornHarvestCombined.2020.rank <-
AggregateField(CornHarvestCombined.2020,response='Y20.r')
CornHarvestCombined.2020.rank <-
CornHarvestCombined.2020.rank[CornHarvestCombined.2020.rank$Samples>30,]
CornHarvestCombined.2020.rank$Cell <- CornHarvestCombined.2020.rank$Row*1000
+ CornHarvestCombined.2020.rank$Column
names(CornHarvestCombined.2020.rank)[3] <- 'Y20.r'
#Plotting QQ plot
ggplot(CornHarvestCombined.2020.rank, aes(sample = Y20.r)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2020 Corn Harvest using Ranks") +
theme_classic()
```



Normalization of 2020 Corn Seeding using Ranks

```
#Normalization of 2020 Corn Seeding using Ranks
CornSeedingCombined.2020$AR20.r<-
rank(CornSeedingCombined.2020$AppliedRate)/max(rank(CornSeedingCombined.2020$
AppliedRate))
CornSeedingCombined.2020.rank <-
AggregateField(CornSeedingCombined.2020,response='AR20.r')
CornSeedingCombined.2020.rank <-
CornSeedingCombined.2020.rank[CornSeedingCombined.2020.rank$Samples>30,]
CornSeedingCombined.2020.rank$Cell <- CornSeedingCombined.2020.rank$Row*1000
+ CornSeedingCombined.2020.rank$Column
names(CornSeedingCombined.2020.rank)[3] <- 'AR20.r'
#Plotting QQ plot
ggplot(CornSeedingCombined.2020.rank, aes(sample = AR20.r)) +
  stat_qq()  +
  stat_qq_line() + labs(title="QQ Plot for 2020 Corn Harvest using Ranks") +
theme_classic()
```

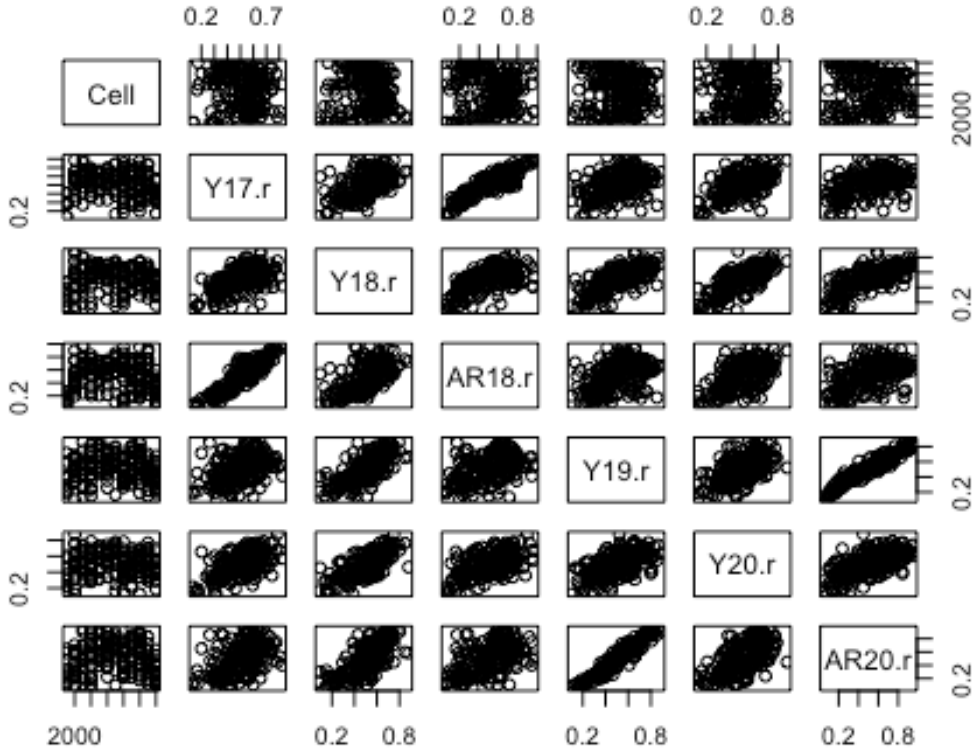


After applying ranks, we have seen that the no of outliers have been decreased on QQ plot and also the points follow the straight line reletively better then before applying ranks.

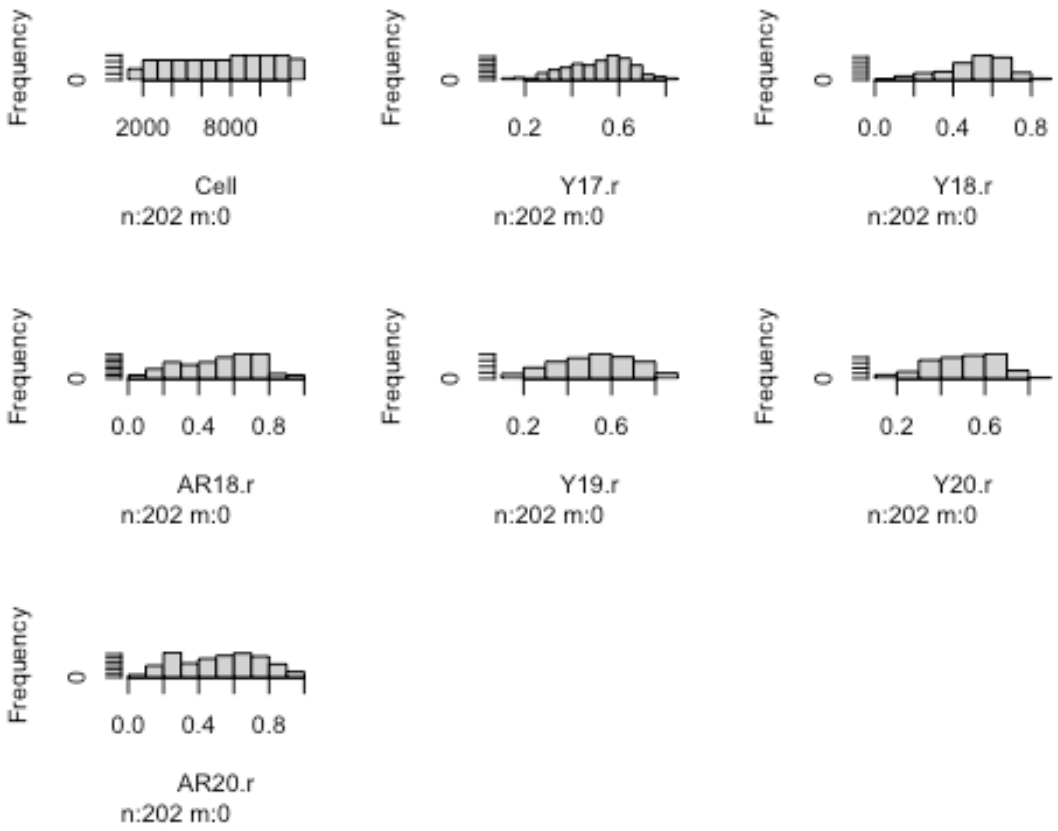
Data Merging after applying Ranks

```
#Merging Soybeans and Corn Data between 2017 and 2018 after applying ranks
Combined_Rank_SoybeansCorn1 <-
merge(Soybeans.Aggregate.2017.rank,CornHarvestCombined.2018.rank, by="Cell")

#Merging Soybeans and Corn Data between 2018 (AppliedRate) and 2019 after
applying ranks
Combined_Rank_SoybeansCorn2 <-
merge(CornSeedingCombined.2018.rank,SoybeansHarvestCombined.2019.rank,
by="Cell")

#Merging Data between 2017 and 2018 and 2018 (AppliedRate) and 2019
Combined_Rank_SoybeansCorn12 <-
merge(Combined_Rank_SoybeansCorn1,Combined_Rank_SoybeansCorn2, by="Cell")

#Merging Data between 2020 after ranks
CombinedRank.2020<-
merge(CornHarvestCombined.2020.rank,CornSeedingCombined.2020.rank, by="Cell")

#Merging Data all Data
```

```r
Combined.all.Rank <- merge(Combined_Rank_SoybeansCorn12,CombinedRank.2020,
by="Cell")

#Removing Unneccesary Coloumns
column_needsRank <-
c("Cell","Y17.r","Y18.r","AR18.r","Y19.r","Y20.r","AR20.r")
Combined.dat.Ra <- Combined.all.Rank[column_needsRank]


plot(Combined.dat.Ra)
```




```r
#Plotting Histrogram
hist.data.frame(Combined.dat.Ra)
```

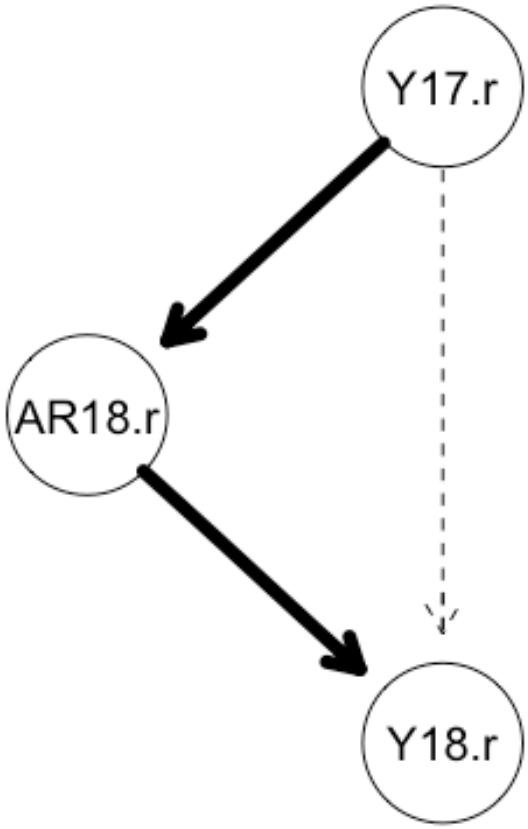


Description:

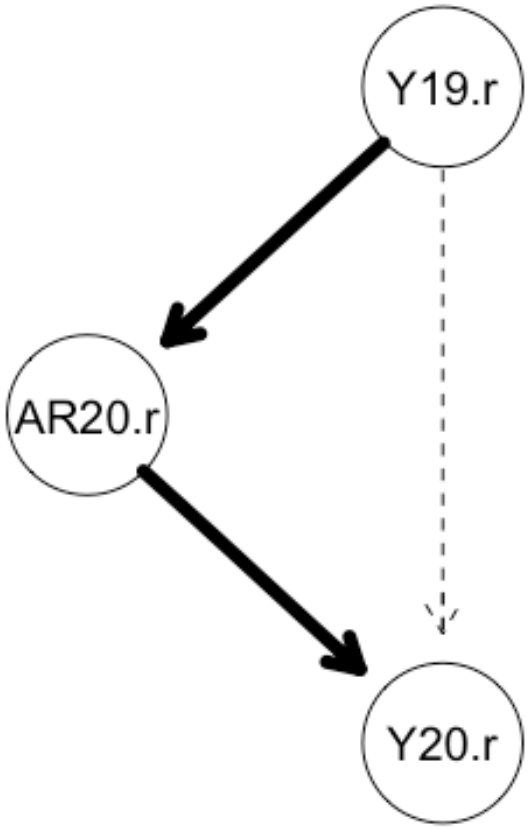
- From the histrogram plots depicted above, we can clearly see that Almost all aggregated variables have a Bell-Shaped histogram. Normally distributed data set creates a symmetric histogram that looks like a bell, leading to the common term for a normal distribution.
- From the pairs plot above, we can see a strong linear relationship between yield rate of 2017 and applied rate of 2018. Also similar kind of strong linear relationship can found between yield rate of 2019 and applied rate of 2020.

Causal Inference After Ranks

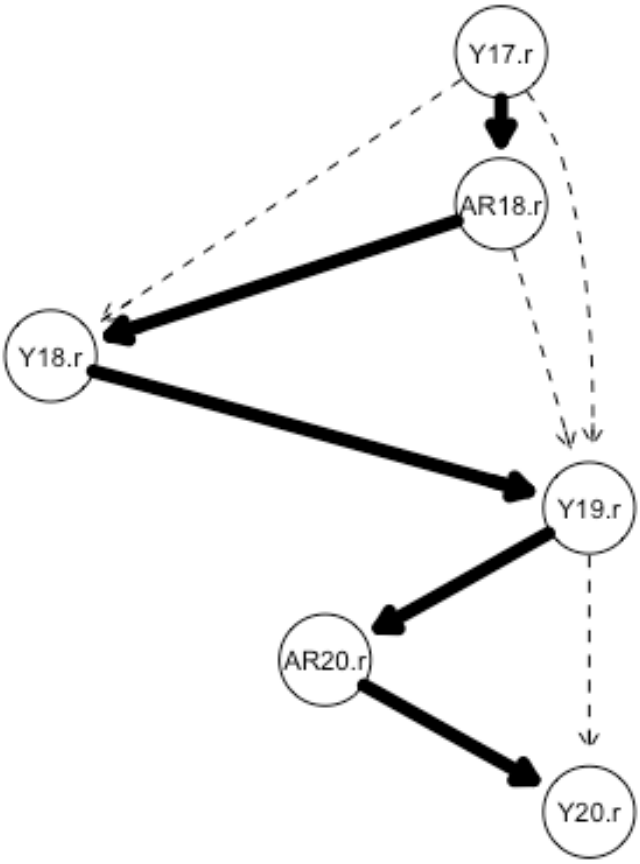
```
#Plotting Directed Acyclic Graphs

modela.dag <- model2network("[Y17.r][AR18.r|Y17.r][Y18.r|AR18.r:Y17.r]")
fit1 = bn.fit(modela.dag, Combined.dat.Ra[,c('Y17.r','AR18.r','Y18.r')])
#fit1

strengtha <- arc.strength(modela.dag,
Combined.dat.Ra[,c('Y17.r','AR18.r','Y18.r')])
strength.plot(modela.dag, strengtha)
```




```
modelb.dag <- model2network("[Y19.r][AR20.r|Y19.r][Y20.r|AR20.r:Y19.r]")
fit2 = bn.fit(modelb.dag, Combined.dat.Ra[,c('Y19.r','AR20.r','Y20.r')])
#fit2
strengthb <- arc.strength(modelb.dag,
Combined.dat.Ra[,c('Y19.r','AR20.r','Y20.r')])
strength.plot(modelb.dag, strengthb)
```




```
model1.dagRb <-
model2network("[Y17.r][AR18.r|Y17.r][Y18.r|AR18.r:Y17.r][Y19.r|Y17.r:AR18.r:Y
18.r][AR20.r|Y19.r][Y20.r|AR20.r:Y19.r]")
fit3 = bn.fit(model1.dagRb,
Combined.dat.Ra[,c('Y17.r','AR18.r','Y18.r','Y19.r','AR20.r','Y20.r')])
#fit3
strength1Rb <- arc.strength(model1.dagRb,
Combined.dat.Ra[,c('Y17.r','AR18.r','Y18.r','Y19.r','AR20.r','Y20.r')])
strength.plot(model1.dagRb, strength1Rb)
```

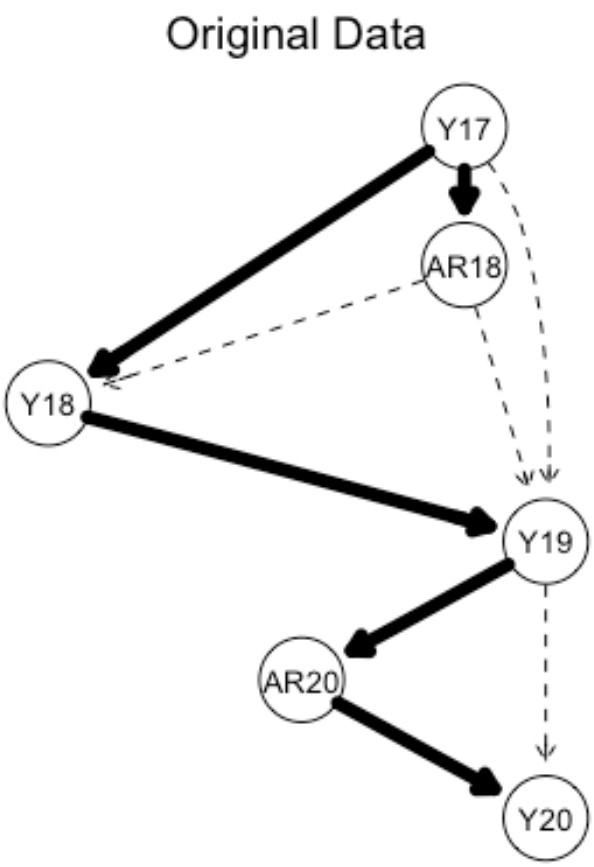


Description:

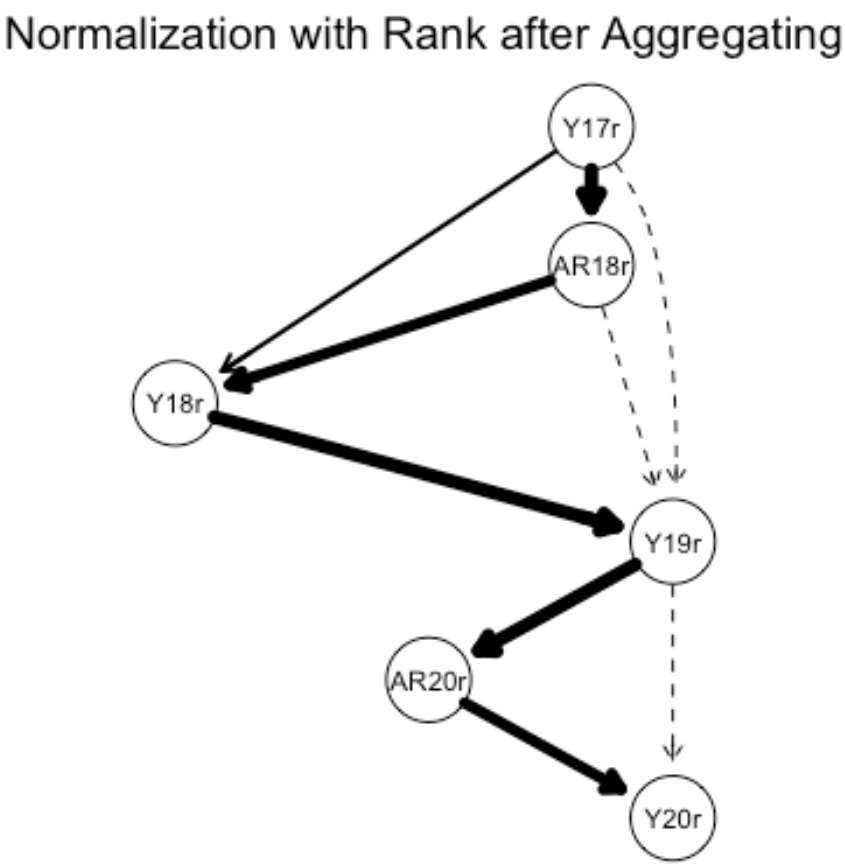
From the Acyclic Graphs depicted above, we can see there is strong relation between yield rate of 2017 and applied rate of 2018 after applying ranks.Although,it shows a light connection between yield rate of 2017 and yield rate of 2018. There present a strong relationship among the yield rate of 2019, applied rate of 2020 and Yield rate of 2020 after using ranks. Additionally, Applied rate of 2018 has strong relation with Yield rate of 2018. It has also strong relation with yield rate of 2019.

Final Comparison

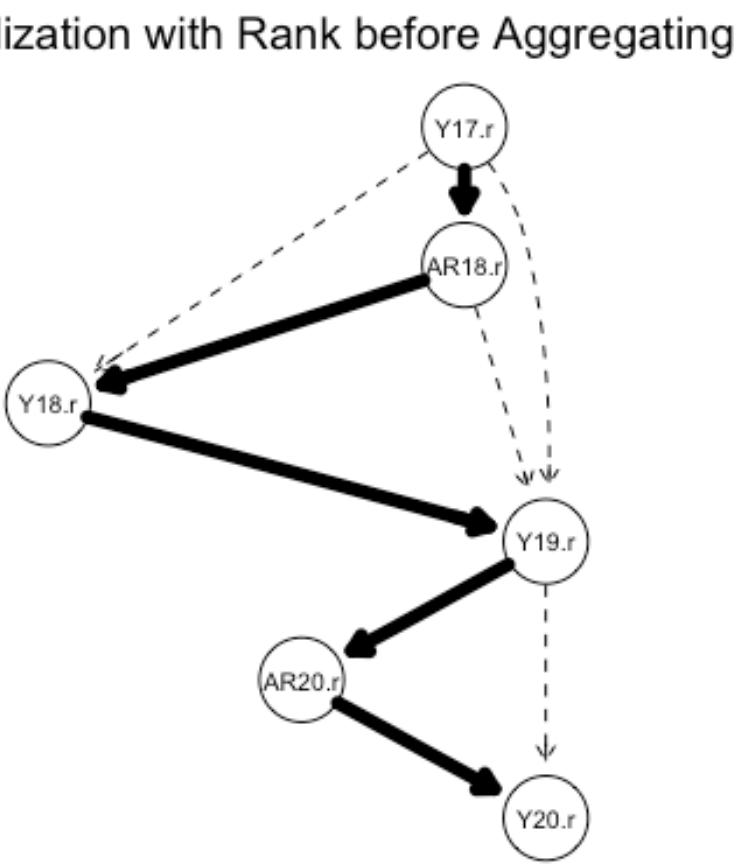
```
strength.plot(model1.dag, strength1,main='Original Data')
```




```
strength.plot(model1.dagRa, strength1Ra,main='Normalization with Rank after
Aggregating')
```




```
strength.plot(model1.dagRb, strength1Rb,main='Normalization with Rank before
Aggregating')
```



Description:

On the basis of normalization, we may compare the three Acyclic Graphs. For the original data we have seen a strong relation between yield rate of 2017 soybean harvest and yield rate of 2018 corn harvest. But when we compare it with 'Normalization with Rank after Aggregating' we can see there is a lightly medium connection between yield rate of 2017 soybean harvest and yield rate of 2018 soybean harvest. Again, when we apply Normalization with Rank before aggregating the data, we have seen a very light connection between yield rate of 2017 soybean harvest and yield rate of 2018 soybean harvest.We can also say that, all the other relation stays same for every yield rate and applied rate among different years.

Conclusion

Based on the our analysis, we can say in conclusion that there are some certain relations between yield rate and applied rate over the years. Furthermore, we have shown that normalization based on rankings may reduce skewness and drive data towards a normal distribution, ensuring the analyses' reliability.