Regression Model for Corn Yield prediction

Members: Yaopeng Ma, Changhe Ji, John Zhang

**Abstract:**

Corn yield prediction is essential for decision-makers, and it is complex because so many potential explanatory variables may affect the growth of crops. Regressors represent four different aspects: weather (varies weekly), soil, scale, and growth performance (varies weekly), and the response is corn yield in BU/Acre. Data was collected from 1990 to 2018, with a total of 8,352 observations and 688 features. The goal is to forecast corn yield performance for the Corn Belt states Illinois, Iowa, and Indiana.

After pre-processing the data, feature selection method Lasso and Recursive Feature Elimination was applied to the model and then build up linear regression model to predict. The set up for experimental setting which prediction is that using exist data from this year and combine with all available historical data as potential unobserved data to construct the prediction interval. From doing that, we achieved an accurate RRMSE( relative root mean square) of 10%, and model fitting of about 75% for both R squared and adjusted R squared for the optimal model.

**Introduction:**

The agriculture industry is one of the largest industries in the world and plays a significant role in economic growth and development. With the recent acceleration of global warming, we have encountered so many severe weather conditions, including extreme participation, flood, wildfire, etc., that would affect the system's stability and food security. Having a stable agriculture system would be the prerequisite and assure any industry. The main factor that reflects the degree of food security would be yield. Therefore, Crop yield prediction is essential for global food security and is complex based on multidimensional data, including environment, scale, growth, etc. An accurate crop yield prediction model can help farmers decide when and what to grow to maximize the yield and maximize possible profit. More importantly, an accurate prediction would inform USDA's economic and management decisions while cooperating with other departments.

Underlying prediction factors include environment, management, the scale of pant, and their protentional interactions. Other factors that may affect the prediction results that are not covered by this Project but still need attention, such as genotype. The above factors make the data high-dimensional and complex, increasing the prediction difficulty. Three types of models are usually used for Corp yield Prediction: linear regression, machine learning, and crop models. All of them have their advantages and disadvantages. We decide to focus on linear regression models, which will indicate the feature importance of each regressor and have a better interpretation of the response. Still, it needs to improve in capturing the non-linear relationship.

We used linear regression to build our base model and the variable selection method lasso to eliminate redundant variables to avoid potential overfitting. From the base model, we develop two more models using different approaches to approve the limitation of our base model.

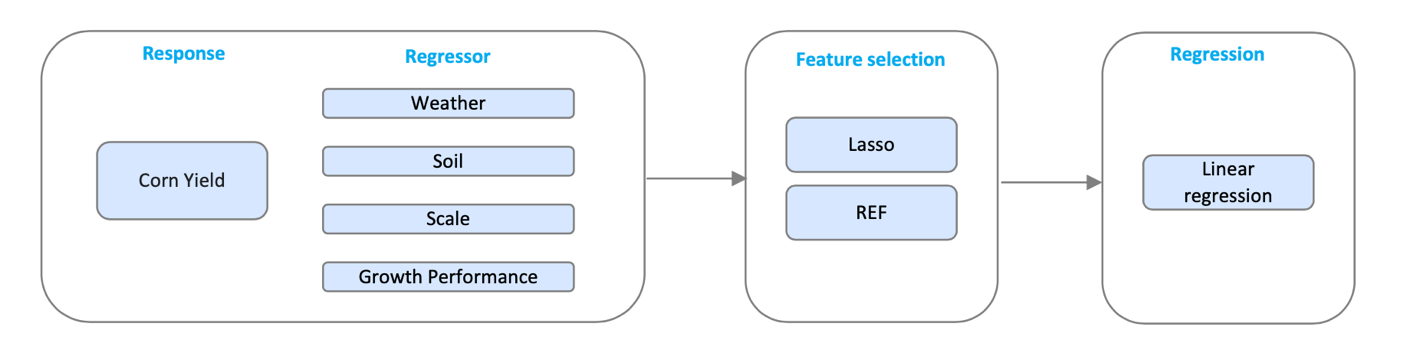
Corn has become the staple food, a standard diet for energy needs, for many countries and the largest crop in the U.S. So, we decide to focus on Corn Yield as the primary response to our Project. With the Corn Yield as the primary response, we divide the potential predictor into four sectors: Weather, Soil, Scale, and Growth performance.

**Method:**

***Model:***

X variables are denoted as Weather, Soil, Scale, and Growth performance and Y is denoted as the response, corn yield for each state. To solve the equation and get a reliable result. We train our model using 8,352 total observations from 3 states which include 293 counties and used total 688 variables as the initial input.

And then, we tried Lasso, rigid, etc., to get the initial variables and then tried to use stepwise forward and backward selection, REF, and the best subset method to form a better model. We tried several steps and selected the three most significant models shown in this project.



***Data pre-processing:***

To simplify data collection, we decided to focus the Project on the following U.S. states Illinois, Indiana, and Iowa, which are central corn-growing states. We collected weather, soil, scale, growth performance, and yield result from publicly available. Weather data was collected from the Iowa Environmental Mesonet (*IOWA State Library*), Soil data was collected from the Gridded Soil Survey Geographic Database (*USDA*), and Scale, Growth performance, and Yield data was collected from the National Agricultural Statistics Service (*USDA*) for all 293 counties of the states of Illinois, Indiana, and Iowa from 1990 to 2018.

Weather data include the following four weather indicators for each county, which are precipitation (Prcp, mm), solar radiation (Srad, MJ/m^2), maximum temperature (Tmax, Celsius), and minimum temperature (Tmin, Celsius). To better illustrate the effect of weather on corn yield, we developed several common additional weather factors from an agronomic and literature review perspective, which are calculated from raw weather data. Growing degree days (Gdd, Celsius), calculated by max{0,mean(Tmax,Tmin)-10}. Number of rainy days (Rdays), which is the number of days with rain above 5 mm and below 24 mm weekly. Number of extreme rainy days (Exrain), the number of days with rain above 24 mm weekly. Number of heat days (Hdays), the number of days with Tmax above 34 Celsius weekly. Number of cold days (Codays), the number of days with Tmin below 5 Celsius weekly. Number of cloudy days (Cldays), the number of days with solar radiation below 10 MJ/m^2 weekly. Heat units (Hunits) are the summation of max{0, Tmax-34} weekly. Weather data includes weeks 13 (April) to 52 (December) data in the model, and we use panted week 1 and week 40 in this model, which correspond with the corn growing season, and data from January to March were excluded.

Soil data included ten parameters that described soil conditions, which are dry bulk density (BDdry, g/cm^3), clay percentage (clay, %), soil pH (pH), drained upper limit (dul, mm/mm), soil saturated hydraulic conductivity (ksat, mm/day), drained lower limit (ll, mm/mm), organic matter (om, %), sand percentage (sand, %), and saturated volumetric water content (sat, mm/mm). All the factors are available at soil depths: 0-5, 5-10, 10-15, 15-30, 30-45, 45-60, 60-80, 80-100, and 100-120 cm. Scale and growth performance data included county-level acres planted and state-level planting process (harvest, progress, flower) and corn population density data. Yield data included observed average yield performance between 1980 and 2018 for corn for all 293 counties of three states of the Corn Belt: Illinois, Indiana, and Iowa.

***Experimental Setting:***

Our model is designed to predict the Corn yield for Illinois, Indiana, and Iowa for the years 2017 and 2018. For predicting the 2017 corn yield, we used data from 1990 to 2016 to train our model. For 2018, we used data from 1990 to 2017. An accurate prediction would be beneficial for economic and management decisions but based on the unpredicted feature of the weather.As crops grow, we can collect more detailed information throughout the year, so we can improve our forecasts on a weekly basis, which could be beneficial to policy makers. It would be quite challenging to have a reliable and credible interval for prediction. Therefore, we developed a method that helps us build reliable confidence intervals when we encounter uncertainty in weather data.

Our model can predict the result for the whole year based on weekly information, which is we can predict the corn yield for 2017 based only on week 1 of 2017. With more data being collected throughout the year, our models are expected to have a more accurate prediction interval which is a narrower confidence interval. To obtain more accurate full-year results, we first train our model on historical data and reuse the recorded information as potential future weather data to predict the full-year results. Over time, we replace historical weather data from previous years with new weather data for forecasting. We run all weather data from previous years as the underlying weather data to construct confidence intervals for the forecasts. And from the results summarize the maximum and minimum values, which build the upper and lower bounds of the interval, and the median as the underlying expected value. To better understand the procedure, we construct the plot below and use 2018 as an example.

**Graphical user interface, application

Description automatically generated**

**Procedure:**

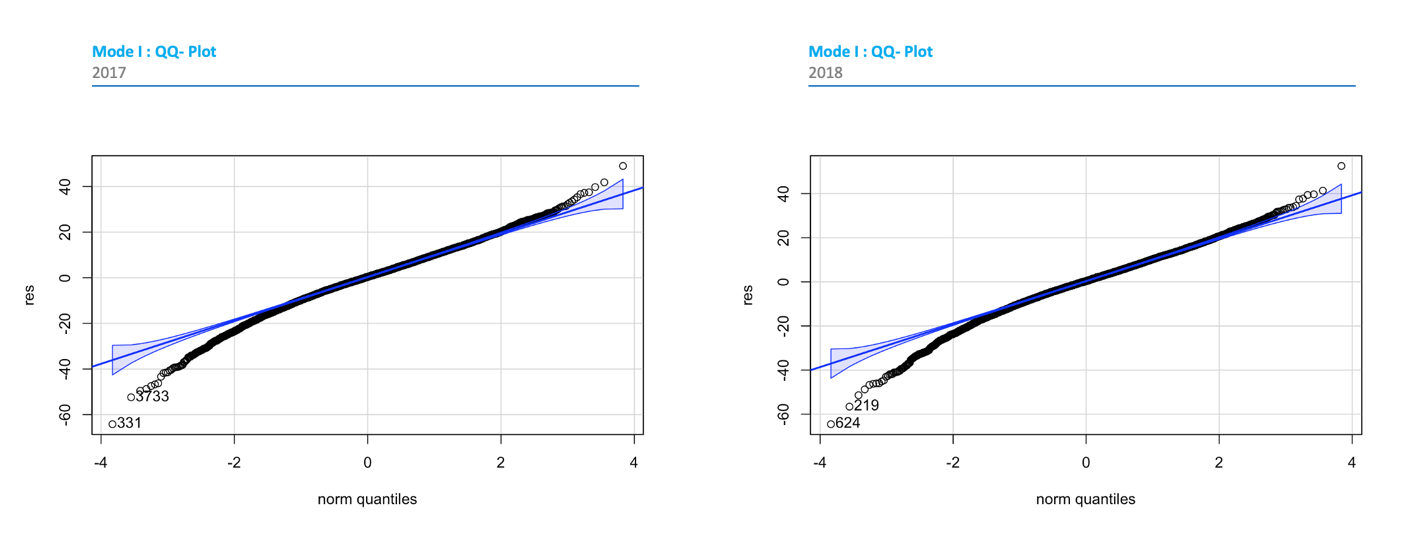
***Model I***

There are many difficulties tend to arise if there are too many variables in our model, one of the most common problems is that the variables are highly correlated to one another which is called multicollinearity, Also, some variables do not contribute very much to the process of predicting the response would cause overfitting of the model. Therefore, the first task to build the model is to eliminate some variables. In this case, we decide to apply LASSO to our dataset to select the best subset to train our model. LASSO (Least Absolute Shrinkage and Selection Operator) is a regression analysis method that can enhance prediction accuracy by utilizing variable selection and regularization. The idea of the lasso is simply adding a penalty term. This is done by adding a constraint in the equation. The consequence of imposing such a penalty is to shrink the coefficient values towards zero, this would set the less contributive regressor to have a zero coefficient which achieves the purpose of variable elimination. We used the R-package ‘glmnet’ to perform lasso on the dataset. After shrinking, there are 519 variables selected.

Next step is to divide the data set into training set and test set. We select the data from 1990-2016 to train the model and apply the model to predict the corn yield in 2017. And we also train the model based on the data from 1990-2017 to predict the corn yield in 2018.

***Residual Analysis***

The following two plots are normal probability plot to test whether fits normal distribution.

****

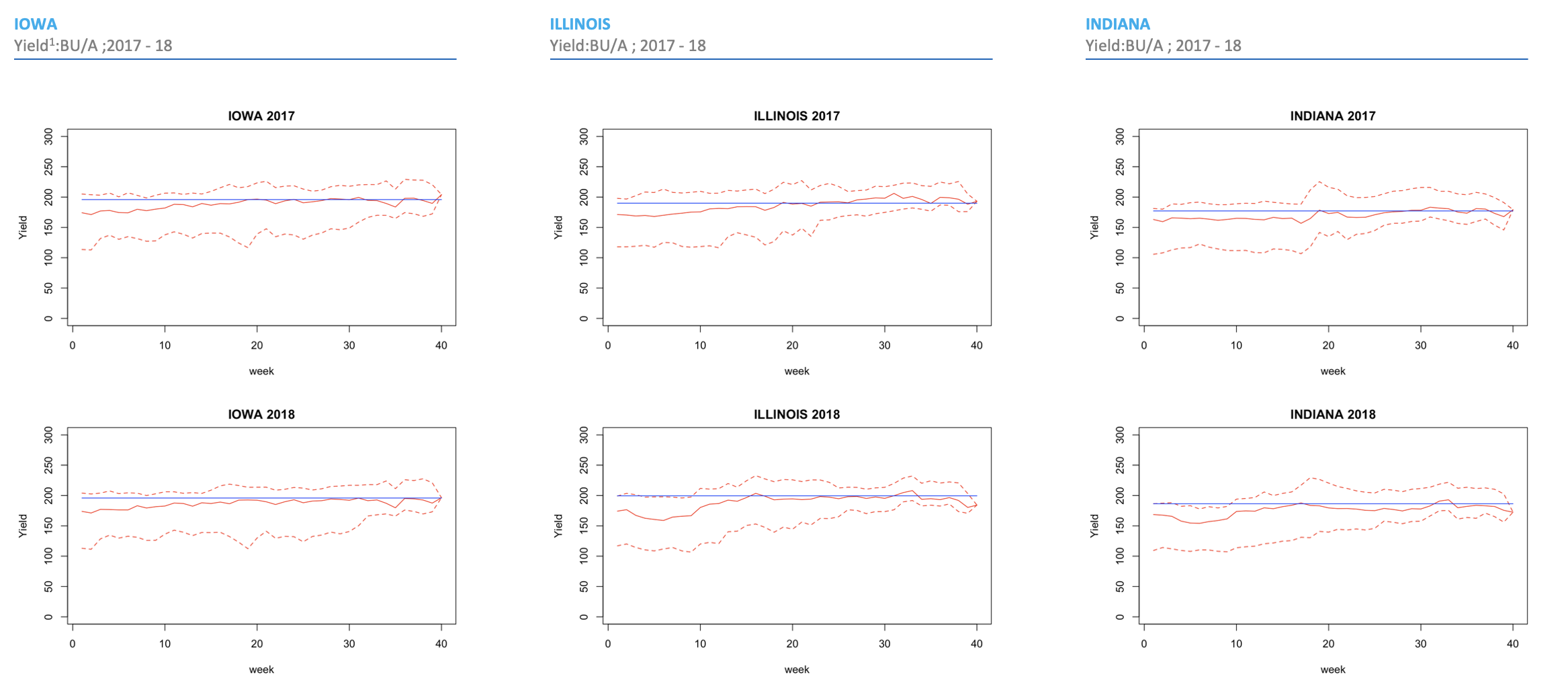
By observing the QQ-plot, we see that the distribution is skewed left, which means that there are more values are concentrated on the right side of the distribution. This is different from our expectation that it should be a perfectly normal distributed. After some research and reading other related articles, we believe this anomaly is caused by interactions and the effects of the advancement of the technology on the crop yield. In order to resolve these issues, first we applied interaction term, but all these interactions have minimum effects on our model.

Chart, line chart

Description automatically generated

Then we plot the corn yield per acer for each year, we see there is an increase trend in the corn yield per acer over the years. Which is caused by the increasing popularity of genetic modified seed and traditional plant breeding. Also, the other advances in science technology leads to drastic improvement on seed irrigation and fertilizers, helping farmers increase yields. We try to find the relevant data regarding the effects of the genotype and the technology on crops yield, but we cannot find useful data for the model. So far, we have no practical solution to resolve these issues.

***Prediction***

****

The prediction interval is relatively large in first 18-20 weeks after planting. But the expected yield is very close to the upper interval of our model in the beginning. The prediction graph shows that the model is still very unstable.

***RRMSE plot***

Chart

Description automatically generated

From RRMSE plot, we see that most outcomes are accurate, but still need improvement. We believe that 519 independent variables still too much for the model, we try to implement other method to eliminate more variables.

***Modell II***

In order to approve our first model, we decide to add recursive feature elimination together with Lasso. Recursive feature elimination or RFE, is a very popular feature selection algorithm, because it is very easy to implement and it is effective to selecting desired number of variables. In model II, we applied RFE on the lasso model, set the number of variables to 500, 450, 400, …, 50. Then we compare each model’s performance. In conclusion, the model with 250 variables has the best performance.

***Residual analysis, QQ-plot***

**Chart, line chart

Description automatically generated**

We can see that the QQ-plot has limited improvement from model I.

***Prediction plot***

**Graphical user interface, chart, line chart

Description automatically generated**

Smaller effects after 35 weeks compared to the model I, the predictions seems to stay very stable for the first 20 weeks, and then slowly converge to the expected yield. Which provide very little information for us in the first 20 weeks.

***RRMSE***

**Chart

Description automatically generated**

From RRMSE perspective, the model II has slightly worse performance compared to the model 1.

***Model III***

The second approach of resolving issues from model I is reduce the time frame of the data observations in addition to lasso and RFE model. We believe reducing the time frame of our training set would eliminate the effect of advancement of new technologies on the crop yield. Since the development of new technologies in agriculture would take years to be accepted by farmers, so we would like to set the time frame as 28 years to 5 years, and train each model to compare their performance. Lasso and REF are also used to determine the best independent variables to fit the model. After comparison, we decided to use 109 variables with 17 years of time frame (2010-2016) to train the model.

***Residual analysis***

**Chart, line chart

Description automatically generated**

There is still very little improvement on QQ-plot.

***Predictions***

**Graphical user interface, chart, line chart

Description automatically generated**

We see that the prediction intervals decreased after 20th weeks, which signify the model are more stable after 20th weeks. And there are less effects on the prediction in the last 5 weeks.

***RRMSE***

**Chart

Description automatically generated**

The RRMSE calculated from model III suggests that the model III has better performance than both model I and model II.

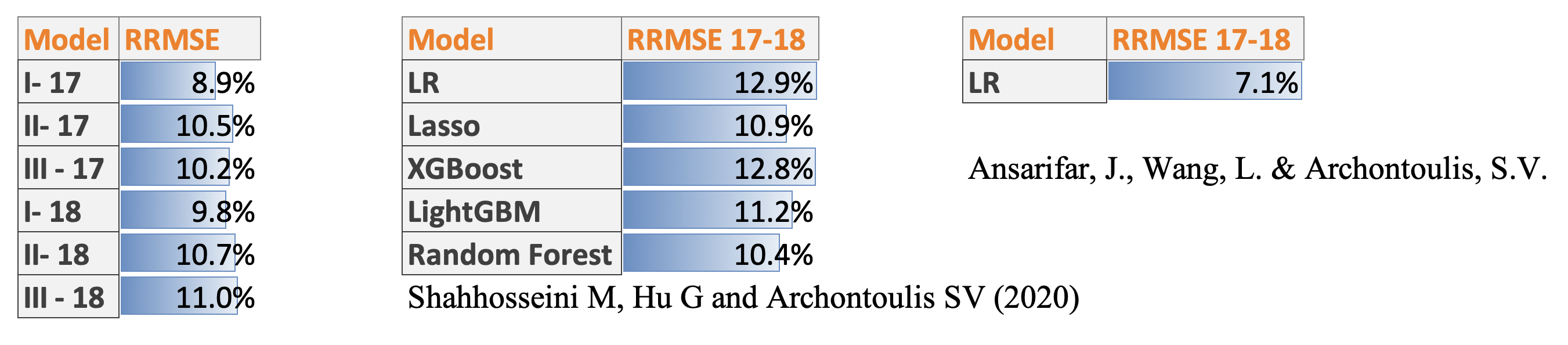
**Conclusion:**

Table

Description automatically generated

For three different models, model I have the best performance in both model fitting and prediction accuracy, but the model disadvantage complexity needs to be improved; Model II has less complexity but an obverse prediction accuracy drop based on the RRMSE. Model III has the best performance in model complexity and mostly better prediction than model II and is slightly weaker than model I.

After comparison, all models have their own biases and advantages. We chose model III as the optimal solution for our project. From all models, we chose the best solution from the trade-off of model complexity and prediction accuracy. But the slight has the potential to be improved.



We compare our result with the article Forecasting Corn Yield With Machine Learning Ensembles (Shahhosseini M, Hu G and Archontoulis SV (2020)), and the article an interaction regression model for crop yield prediction (Ansarifar, J., Wang, L. & Archontoulis, S.V.), we compared average state-level RRMSE and received a decent result similar to what academia accepted.

**REFERENCE**

1. *IOWA State Library*, 1990, https://mesonet.agron.iastate.edu. Accessed 20 Nov. 2022.
2. *USDA*, 1990, https://gdg.sc.egov.usda.gov. Accessed 20 Nov. 2022.
3. *USDA*, 1990, https://quickstats.nass.usda.gov. Accessed 20 Nov. 2022.
4. Shahhosseini M, Hu G and Archontoulis SV (2020) Forecasting Corn Yield With Machine Learning Ensembles. Front. Plant Sci. 11:1120. doi: 10.3389/fpls.2020.01120.
5. Ansarifar, J., Wang, L. & Archontoulis, S.V. An interaction regression model for crop yield prediction. Sci Rep 11, 17754 (2021). https://doi.org/10.1038/s41598-021-97221-7.

**Appendix**

```{r data\_preprocessing}

#load\_data

data\_origin = read.csv("/Users/akaike/Documents/fall22/STAT615/final pj/corn\_data.csv", header = TRUE)

data\_normalized = read.csv("/Users/akaike/Documents/fall22/STAT615/final pj/corn\_data\_normalized.csv", header = TRUE)

data\_normalized = data\_normalized[,c(2:690)]

Y = data\_normalized$Value

X = as.matrix(data\_normalized[,c(2:689)])

Y\_train = Y[587:8352]

X\_train = X[587:8352,]

```

```{r}

#load variables

rbf\_variables = read.table("/Users/akaike/Documents/fall22/STAT615/final pj/rbf\_variable\_150.txt",header = FALSE)

length(rbf\_variables$V1)

model\_REF150\_2017 = lm(Value ~. , data = data\_normalized[587:5436,c("Value",rbf\_variables$V1)])

```

```{r}

# return the regression result and draw qqplot

summary(model\_REF150\_2017)

library("car")

qqPlot(resid(model\_REF150\_2017))

qqline(resid(model\_REF150\_2017), col = "steelblue", lwd = 2)

```

\section{2017 prediction 150 variables}

```{r}

data\_normalized2 = cbind(data\_origin[,c(1:8)],data\_normalized)

data\_2017 = data\_normalized2[data\_normalized2$Year==2017,]

```

\subsection{2017 Illinois 150 variables}

```{r warning=FALSE}

#2017\_predictions\_Illinois\_150 variables

predictions\_ILLINOIS\_2017 = matrix(0,27,40)

Illinois\_2017 = data\_2017[data\_2017$State=="ILLINOIS",]

data\_Illinois = data\_normalized2[data\_normalized2$State == "ILLINOIS",]

week = 0

for (i in 13:52) {

year = 1

for (j in 1990:2016) {

data\_county = data\_Illinois[data\_Illinois$Year == 2019,]

for (county in Illinois\_2017$County) {

temp\_data = data\_Illinois[data\_Illinois$Year == j,]

if (county %in% temp\_data$County) {

if (week == 0) {

a = cbind(Illinois\_2017[Illinois\_2017$County == county,c(1:99,100)],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

Illinois\_2017[Illinois\_2017$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):203)],

Illinois\_2017[Illinois\_2017$County == county,c(204:205+week)],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

Illinois\_2017[Illinois\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

} else if (week == 39) {

a = Illinois\_2017[Illinois\_2017$County == county,]

} else {

a = cbind(Illinois\_2017[Illinois\_2017$County == county,c(1:99,100:(100+week))],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

Illinois\_2017[Illinois\_2017$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):204)],

Illinois\_2017[Illinois\_2017$County == county,c(205:(205+week))],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

Illinois\_2017[Illinois\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

}

} else {

a = Illinois\_2017[Illinois\_2017$County == county,]

}

data\_county = rbind(data\_county,a)

}

yhat0 <- predict(model\_REF150\_2017, newdata= data\_county[,c(rbf\_variables$V1)])

predictions\_ILLINOIS\_2017[year,week + 1] = sum(yhat0)/length(yhat0)

year = year + 1

}

week = week + 1

}

acre\_value = sum(Illinois\_2017$Acre.Value)

real\_Yield = matrix(0,40,2)

max\_predicts = matrix(0,40,2)

for (i in 1:40) {

max\_value = max(predictions\_ILLINOIS\_2017[,i])

max\_predicts[i,1] = max\_value

max\_predicts[i,2] = i

real\_Yield[i,1] = sum(Illinois\_2017$Value)/length(yhat0)

real\_Yield[i,2] = i

}

min\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = min(predictions\_ILLINOIS\_2017[,i])

min\_predicts[i,1] = min\_value

min\_predicts[i,2] = i

}

median\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = median(predictions\_ILLINOIS\_2017[,i])

median\_predicts[i,1] = min\_value

median\_predicts[i,2] = i

}

plot(x = max\_predicts[,2], y = max\_predicts[,1],ylim = c(0,300),lty =2,col='red',type= 'l', ylab = "Yield",xlab = "week", main="ILLINOIS 2017")

lines(x = min\_predicts[,2],y=min\_predicts[,1],lty =2,col='red')

lines(x = median\_predicts[,2], y = median\_predicts[,1], lty =1, col='red')

lines(x = real\_Yield[,2], y = real\_Yield[,1], lty =1, col='blue')

```

```{r warning=FALSE}

yhat\_Illinois\_2017\_RFE150 = predict(model\_REF150\_2017, newdata = data\_county[,c(rbf\_variables$V1)])

res = Illinois\_2017$Value - yhat\_Illinois\_2017\_RFE150

Illinois\_2017\_RFE150\_RMSE = sqrt(sum(res^2)/length(res))

Illinois\_2017\_RFE150\_RRMSE = Illinois\_2017\_RFE150\_RMSE/mean(Illinois\_2017$Value)

print(c(Illinois\_2017\_RFE150\_RMSE,Illinois\_2017\_RFE150\_RRMSE))

RRMSE\_for\_county\_Illinois\_2017\_RFE150 = sqrt(res^2)/Illinois\_2017$Value

write.table(RRMSE\_for\_county\_Illinois\_2017\_RFE150, "/Users/akaike/Documents/fall22/STAT615/final pj/RRMSE\_RBF150\_5437/RRMSE\_for\_county\_Illinois\_2017\_RFE150.txt", append = FALSE, sep = " ", dec = ".",

row.names = TRUE, col.names = TRUE)

```

\subsection{2017 IOWA 150 variables}

```{r warning=FALSE}

#2017\_predictions\_IOWA\_150 variables

predictions\_IOWA\_2017 = matrix(0,27,40)

IOWA\_2017 = data\_2017[data\_2017$State=="IOWA",]

data\_IOWA = data\_normalized2[data\_normalized2$State == "IOWA",]

week = 0

for (i in 13:52) {

year = 1

for (j in 1990:2016) {

data\_county = data\_IOWA[data\_IOWA$Year == 2019,]

for (county in IOWA\_2017$County) {

temp\_data = data\_IOWA[data\_IOWA$Year == j,]

if (county %in% temp\_data$County) {

if (week == 0) {

a = cbind(IOWA\_2017[IOWA\_2017$County == county,c(1:99,100)],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

IOWA\_2017[IOWA\_2017$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):203)],

IOWA\_2017[IOWA\_2017$County == county,c(204:205+week)],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

IOWA\_2017[IOWA\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

} else if (week == 39) {

a = IOWA\_2017[IOWA\_2017$County == county,]

} else {

a = cbind(IOWA\_2017[IOWA\_2017$County == county,c(1:99,100:(100+week))],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

IOWA\_2017[IOWA\_2017$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):204)],

IOWA\_2017[IOWA\_2017$County == county,c(205:(205+week))],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

IOWA\_2017[IOWA\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

}

# data\_county[c, c(1:99,100:(100+week))] = Illinois\_2017[Illinois\_2017$County == county,c(1:99,100:(100+week))]

# data\_county[c, c((100+week+1):151)] = temp\_data[temp\_data$County == county,c((100+week+1):151)]

# data\_county[c, c(152,153:(153+week))] = Illinois\_2017[Illinois\_2017$County == county,c(152,153:(153+week))]

# data\_county[c, c((153+week+1):204)] = temp\_data[temp\_data$County == county,c((153+week+1):204)]

# data\_county[c, c(205:(205+week))] = Illinois\_2017[Illinois\_2017$County == county,c(205:(205+week))]

# data\_county[c, c((205+week+1):256)] = temp\_data[temp\_data$County == county,c((205+week+1):256)]

# data\_county[c, c(257,258:(258+(week+1)\*11 - 1))] = Illinois\_2017[Illinois\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))]

# data\_county[c, c((258+(week+1)\*11):697)] = temp\_data[Illinois\_2017$County == county,c((258+(week+1)\*11):697)]

} else {

a = IOWA\_2017[IOWA\_2017$County == county,]

}

data\_county = rbind(data\_county,a)

}

yhat0 <- predict(model\_REF150\_2017, newdata= data\_county[,c(rbf\_variables$V1)])

predictions\_IOWA\_2017[year,week + 1] = sum(yhat0)/length(yhat0)

year = year + 1

}

week = week + 1

}

```

```{r}

# calculate the real yield and max,min and median yield of prediction

acre\_value = sum(IOWA\_2017$Acre.Value)

real\_Yield = matrix(0,40,2)

max\_predicts = matrix(0,40,2)

for (i in 1:40) {

max\_value = max(predictions\_IOWA\_2017[,i])

max\_predicts[i,1] = max\_value

max\_predicts[i,2] = i

real\_Yield[i,1] = sum(IOWA\_2017$Value)/length(yhat0)

real\_Yield[i,2] = i

}

min\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = min(predictions\_IOWA\_2017[,i])

min\_predicts[i,1] = min\_value

min\_predicts[i,2] = i

}

median\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = median(predictions\_IOWA\_2017[,i])

median\_predicts[i,1] = min\_value

median\_predicts[i,2] = i

}

```

```{r}

#draw the prediction plot

plot(x = max\_predicts[,2], y = max\_predicts[,1],ylim = c(0,300),lty =2,col='red',type= 'l', ylab = "Yield",xlab = "week", main="IOWA 2017")

lines(x = min\_predicts[,2],y=min\_predicts[,1],lty =2,col='red')

lines(x = median\_predicts[,2], y = median\_predicts[,1], lty =1, col='red')

lines(x = real\_Yield[,2], y = real\_Yield[,1], lty =1, col='blue')

```

```{r warning=FALSE}

#calculate the RRMSE

yhat\_IOWA\_2017\_RFE150 = predict(model\_REF150\_2017, newdata= data\_county[,c(rbf\_variables$V1)])

res = IOWA\_2017$Value - yhat\_IOWA\_2017\_RFE150

IOWA\_2017\_RFE150\_RMSE = sqrt(sum(res^2)/length(res))

IOWA\_2017\_RFE150\_RRMSE = IOWA\_2017\_RFE150\_RMSE/mean(IOWA\_2017$Value)

print(c(IOWA\_2017\_RFE150\_RMSE,IOWA\_2017\_RFE150\_RRMSE))

RRMSE\_for\_county\_IOWA\_2017\_RFE150 = sqrt(res^2)/IOWA\_2017$Value

write.table(RRMSE\_for\_county\_IOWA\_2017\_RFE150, "/Users/akaike/Documents/fall22/STAT615/final pj/RRMSE\_RBF150\_5437/RRMSE\_for\_county\_IOWA\_2017\_RFE150.txt", append = FALSE, sep = " ", dec = ".",

row.names = TRUE, col.names = TRUE)

```

\subsection{2017 INDIANA 150 variables}

```{r warning=FALSE}

#2017\_predictions\_INDIANA\_150 variables

predictions\_INDIANA\_2017 = matrix(0,27,40)

INDIANA\_2017 = data\_2017[data\_2017$State=="INDIANA",]

data\_INDIANA = data\_normalized2[data\_normalized2$State == "INDIANA",]

week = 0

for (i in 13:52) {

year = 1

for (j in 1990:2016) {

data\_county = data\_INDIANA[data\_INDIANA$Year == 2019,]

for (county in INDIANA\_2017$County) {

temp\_data = data\_INDIANA[data\_INDIANA$Year == j,]

if (county %in% temp\_data$County) {

if (week == 0) {

a = cbind(INDIANA\_2017[INDIANA\_2017$County == county,c(1:99,100)],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

INDIANA\_2017[INDIANA\_2017$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):203)],

INDIANA\_2017[INDIANA\_2017$County == county,c(204:205+week)],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

INDIANA\_2017[INDIANA\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

} else if (week == 39) {

a = INDIANA\_2017[INDIANA\_2017$County == county,]

} else {

a = cbind(INDIANA\_2017[INDIANA\_2017$County == county,c(1:99,100:(100+week))],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

INDIANA\_2017[INDIANA\_2017$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):204)],

INDIANA\_2017[INDIANA\_2017$County == county,c(205:(205+week))],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

INDIANA\_2017[INDIANA\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

}

# data\_county[c, c(1:99,100:(100+week))] = Illinois\_2017[Illinois\_2017$County == county,c(1:99,100:(100+week))]

# data\_county[c, c((100+week+1):151)] = temp\_data[temp\_data$County == county,c((100+week+1):151)]

# data\_county[c, c(152,153:(153+week))] = Illinois\_2017[Illinois\_2017$County == county,c(152,153:(153+week))]

# data\_county[c, c((153+week+1):204)] = temp\_data[temp\_data$County == county,c((153+week+1):204)]

# data\_county[c, c(205:(205+week))] = Illinois\_2017[Illinois\_2017$County == county,c(205:(205+week))]

# data\_county[c, c((205+week+1):256)] = temp\_data[temp\_data$County == county,c((205+week+1):256)]

# data\_county[c, c(257,258:(258+(week+1)\*11 - 1))] = Illinois\_2017[Illinois\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))]

# data\_county[c, c((258+(week+1)\*11):697)] = temp\_data[Illinois\_2017$County == county,c((258+(week+1)\*11):697)]

} else {

a = INDIANA\_2017[INDIANA\_2017$County == county,]

}

data\_county = rbind(data\_county,a)

}

yhat0 <- predict(model\_REF150\_2017, newdata= data\_county[,c(rbf\_variables$V1)])

predictions\_INDIANA\_2017[year,week + 1] = sum(yhat0)/length(yhat0)

year = year + 1

}

week = week + 1

}

```

```{r}

# calculate the real yield and max,min and median of predictions

acre\_value = sum(INDIANA\_2017$Acre.Value)

real\_Yield = matrix(0,40,2)

max\_predicts = matrix(0,40,2)

for (i in 1:40) {

max\_value = max(predictions\_INDIANA\_2017[,i])

max\_predicts[i,1] = max\_value

max\_predicts[i,2] = i

real\_Yield[i,1] = sum(INDIANA\_2017$Value)/length(yhat0)

real\_Yield[i,2] = i

}

min\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = min(predictions\_INDIANA\_2017[,i])

min\_predicts[i,1] = min\_value

min\_predicts[i,2] = i

}

median\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = median(predictions\_INDIANA\_2017[,i])

median\_predicts[i,1] = min\_value

median\_predicts[i,2] = i

}

```

```{r}

# draw the prediction plot

plot(x = max\_predicts[,2], y = max\_predicts[,1],ylim = c(0,300),lty =2,col='red',type= 'l', ylab = "Yield",xlab = "week", main="INDIANA 2017")

lines(x = min\_predicts[,2],y=min\_predicts[,1],lty =2,col='red')

lines(x = median\_predicts[,2], y = median\_predicts[,1], lty =1, col='red')

lines(x = real\_Yield[,2], y = real\_Yield[,1], lty =1, col='blue')

```

```{r warning=FALSE}

# calculate the RRMSE

yhat\_INDIANA\_2017\_RFE150 = predict(model\_REF150\_2017, newdata= data\_county[,c(rbf\_variables$V1)])

res = INDIANA\_2017$Value - yhat\_INDIANA\_2017\_RFE150

INDIANA\_2017\_RFE150\_RMSE = sqrt(sum(res^2)/length(res))

INDIANA\_2017\_RFE150\_RRMSE = INDIANA\_2017\_RFE150\_RMSE/mean(INDIANA\_2017$Value)

print(c(INDIANA\_2017\_RFE150\_RMSE,INDIANA\_2017\_RFE150\_RRMSE))

RRMSE\_for\_county\_INDIANA\_2017\_RFE150 = sqrt(res^2)/INDIANA\_2017$Value

write.table(RRMSE\_for\_county\_INDIANA\_2017\_RFE150, "/Users/akaike/Documents/fall22/STAT615/final pj/RRMSE\_RBF150\_5437/RRMSE\_for\_county\_INDIANA\_2017\_RFE150.txt", append = FALSE, sep = " ", dec = ".",

row.names = TRUE, col.names = TRUE)

```

\section{2018 predictions}

```{r}

# load data of 2018

data\_2018 = data\_normalized2[data\_normalized2$Year==2018,]

```

```{r}

#fit the model

model\_RFE150\_2018 = lm(Value ~. , data = data\_normalized[294:5149,c("Value",rbf\_variables$V1)])

```

```{r}

#return the regression result and draw the qqplot

summary(model\_RFE150\_2018)

qqPlot(resid(model\_RFE150\_2018))

rbf\_RMSE = sqrt(sum(resid(model\_RFE150\_2018)^2)/length(resid(model\_RFE150\_2018)))

rbf\_RRMSE = rbf\_RMSE/mean(Y\_train)

```

\subsection{2018 Illinois RFE 150 variables}

```{r warning=FALSE}

#2018\_predictions\_Illinois\_RFE 150 variables and draw the prediction plot

predictions\_ILLINOIS\_2018 = matrix(0,28,40)

Illinois\_2018 = data\_2018[data\_2018$State=="ILLINOIS",]

data\_Illinois = data\_normalized2[data\_normalized2$State == "ILLINOIS",]

week = 0

for (i in 13:52) {

year = 1

for (j in 1990:2017) {

data\_county = data\_Illinois[data\_Illinois$Year == 2019,]

for (county in Illinois\_2018$County) {

temp\_data = data\_Illinois[data\_Illinois$Year == j,]

if (county %in% temp\_data$County) {

if (week == 0) {

a = cbind(Illinois\_2018[Illinois\_2018$County == county,c(1:99,100)],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

Illinois\_2018[Illinois\_2018$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):203)],

Illinois\_2018[Illinois\_2018$County == county,c(204:205+week)],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

Illinois\_2018[Illinois\_2018$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

} else if (week == 39) {

a = Illinois\_2018[Illinois\_2018$County == county,]

} else {

a = cbind(Illinois\_2018[Illinois\_2018$County == county,c(1:99,100:(100+week))],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

Illinois\_2018[Illinois\_2018$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):204)],

Illinois\_2018[Illinois\_2018$County == county,c(205:(205+week))],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

Illinois\_2018[Illinois\_2018$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

}

# data\_county[c, c(1:99,100:(100+week))] = Illinois\_2017[Illinois\_2017$County == county,c(1:99,100:(100+week))]

# data\_county[c, c((100+week+1):151)] = temp\_data[temp\_data$County == county,c((100+week+1):151)]

# data\_county[c, c(152,153:(153+week))] = Illinois\_2017[Illinois\_2017$County == county,c(152,153:(153+week))]

# data\_county[c, c((153+week+1):204)] = temp\_data[temp\_data$County == county,c((153+week+1):204)]

# data\_county[c, c(205:(205+week))] = Illinois\_2017[Illinois\_2017$County == county,c(205:(205+week))]

# data\_county[c, c((205+week+1):256)] = temp\_data[temp\_data$County == county,c((205+week+1):256)]

# data\_county[c, c(257,258:(258+(week+1)\*11 - 1))] = Illinois\_2017[Illinois\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))]

# data\_county[c, c((258+(week+1)\*11):697)] = temp\_data[Illinois\_2017$County == county,c((258+(week+1)\*11):697)]

} else {

a = Illinois\_2018[Illinois\_2018$County == county,]

}

data\_county = rbind(data\_county,a)

}

yhat0 <- predict(model\_RFE150\_2018, newdata= data\_county[,c(rbf\_variables$V1)])

predictions\_ILLINOIS\_2018[year,week + 1] = sum(yhat0)/length(yhat0)

year = year + 1

}

week = week + 1

}

acre\_value = sum(Illinois\_2018$Acre.Value)

real\_Yield = matrix(0,40,2)

max\_predicts = matrix(0,40,2)

for (i in 1:40) {

max\_value = max(predictions\_ILLINOIS\_2018[,i])

max\_predicts[i,1] = max\_value

max\_predicts[i,2] = i

real\_Yield[i,1] = sum(Illinois\_2018$Value)/length(yhat0)

real\_Yield[i,2] = i

}

min\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = min(predictions\_ILLINOIS\_2018[,i])

min\_predicts[i,1] = min\_value

min\_predicts[i,2] = i

}

median\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = median(predictions\_ILLINOIS\_2018[,i])

median\_predicts[i,1] = min\_value

median\_predicts[i,2] = i

}

plot(x = max\_predicts[,2], y = max\_predicts[,1],ylim = c(0,300),lty =2,col='red',type= 'l', ylab = "Yield",xlab = "week", main="ILLINOIS 2018")

lines(x = min\_predicts[,2],y=min\_predicts[,1],lty =2,col='red')

lines(x = median\_predicts[,2], y = median\_predicts[,1], lty =1, col='red')

lines(x = real\_Yield[,2], y = real\_Yield[,1], lty =1, col='blue')

```

```{r warning=FALSE}

#calculate the RRMSE

yhat\_Illinois\_2018\_RFE150 = predict(model\_RFE150\_2018, newdata= Illinois\_2018[,c(rbf\_variables$V1)])

res = Illinois\_2018$Value - yhat\_Illinois\_2018\_RFE150

Illinois\_2018\_RFE150\_RMSE = sqrt(sum(res^2)/length(res))

Illinois\_2018\_RFE150\_RRMSE = Illinois\_2018\_RFE150\_RMSE/mean(Illinois\_2018$Value)

print(c(Illinois\_2018\_RFE150\_RMSE, Illinois\_2018\_RFE150\_RRMSE))

RRMSE\_for\_county\_Illinois\_2018\_RFE150 = sqrt(res^2)/Illinois\_2018$Value

write.table(RRMSE\_for\_county\_Illinois\_2018\_RFE150, "/Users/akaike/Documents/fall22/STAT615/final pj/RRMSE\_RBF150\_5437/RRMSE\_for\_county\_Illinois\_2018\_RFE150.txt", append = FALSE, sep = " ", dec = ".",

row.names = TRUE, col.names = TRUE)

```

\subsection{2018 IOWA RFE 150 variables}

```{r warning=FALSE}

#2018\_predictions\_IOWA\_RFE 150 variables and draw the prediction plot

predictions\_IOWA\_2018 = matrix(0,28,40)

IOWA\_2018 = data\_2018[data\_2018$State=="IOWA",]

data\_IOWA = data\_normalized2[data\_normalized2$State == "IOWA",]

week = 0

for (i in 13:52) {

year = 1

for (j in 1990:2017) {

data\_county = data\_IOWA[data\_IOWA$Year == 2019,]

for (county in IOWA\_2018$County) {

temp\_data = data\_IOWA[data\_IOWA$Year == j,]

if (county %in% temp\_data$County) {

if (week == 0) {

a = cbind(IOWA\_2018[IOWA\_2018$County == county,c(1:99,100)],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

IOWA\_2018[IOWA\_2018$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):203)],

IOWA\_2018[IOWA\_2018$County == county,c(204:205+week)],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

IOWA\_2018[IOWA\_2018$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

} else if (week == 39) {

a = IOWA\_2018[IOWA\_2018$County == county,]

} else {

a = cbind(IOWA\_2018[IOWA\_2018$County == county,c(1:99,100:(100+week))],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

IOWA\_2018[IOWA\_2018$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):204)],

IOWA\_2018[IOWA\_2018$County == county,c(205:(205+week))],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

IOWA\_2018[IOWA\_2018$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

}

} else {

a = IOWA\_2018[IOWA\_2018$County == county,]

}

data\_county = rbind(data\_county,a)

}

yhat0 <- predict(model\_RFE150\_2018, newdata= data\_county[,c(rbf\_variables$V1)])

predictions\_IOWA\_2018[year,week + 1] = sum(yhat0)/length(yhat0)

year = year + 1

}

week = week + 1

}

acre\_value = sum(IOWA\_2018$Acre.Value)

real\_Yield = matrix(0,40,2)

max\_predicts = matrix(0,40,2)

for (i in 1:40) {

max\_value = max(predictions\_IOWA\_2018[,i])

max\_predicts[i,1] = max\_value

max\_predicts[i,2] = i

real\_Yield[i,1] = sum(IOWA\_2018$Value)/length(yhat0)

real\_Yield[i,2] = i

}

min\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = min(predictions\_IOWA\_2018[,i])

min\_predicts[i,1] = min\_value

min\_predicts[i,2] = i

}

median\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = median(predictions\_IOWA\_2018[,i])

median\_predicts[i,1] = min\_value

median\_predicts[i,2] = i

}

plot(x = max\_predicts[,2], y = max\_predicts[,1],ylim = c(0,300),lty =2,col='red',type= 'l', ylab = "Yield",xlab = "week", main="IOWA 2018")

lines(x = min\_predicts[,2],y=min\_predicts[,1],lty =2,col='red')

lines(x = median\_predicts[,2], y = median\_predicts[,1], lty =1, col='red')

lines(x = real\_Yield[,2], y = real\_Yield[,1], lty =1, col='blue')

```

```{r warning=FALSE}

#calculate the RRMSE

yhat\_IOWA\_2018\_RFE150 = predict(model\_RFE150\_2018, newdata= data\_county[,c(rbf\_variables$V1)])

res = IOWA\_2018$Value - yhat\_IOWA\_2018\_RFE150

IOWA\_2018\_RFE150\_RMSE = sqrt(sum(res^2)/length(res))

IOWA\_2018\_RFE150\_RRMSE = IOWA\_2018\_RFE150\_RMSE/mean(IOWA\_2018$Value)

print(c(IOWA\_2018\_RFE150\_RMSE, IOWA\_2018\_RFE150\_RRMSE))

RRMSE\_for\_county\_IOWA\_2018\_RFE150 = sqrt(res^2)/IOWA\_2018$Value

write.table(RRMSE\_for\_county\_IOWA\_2018\_RFE150, "/Users/akaike/Documents/fall22/STAT615/final pj/RRMSE\_RBF150\_5437/RRMSE\_for\_county\_IOWA\_2018\_RFE150.txt", append = FALSE, sep = " ", dec = ".",

row.names = TRUE, col.names = TRUE)

```

\subsection{2018 INDIANA RFE 250 variables}

```{r warning=FALSE}

#2018\_predictions\_INDIANA\_RFE 250 variables and draw the prediction plot

predictions\_INDIANA\_2018 = matrix(0,28,40)

INDIANA\_2018 = data\_2018[data\_2018$State=="INDIANA",]

data\_INDIANA = data\_normalized2[data\_normalized2$State == "INDIANA",]

week = 0

for (i in 13:52) {

year = 1

for (j in 1990:2017) {

data\_county = data\_INDIANA[data\_INDIANA$Year == 2019,]

for (county in INDIANA\_2018$County) {

temp\_data = data\_INDIANA[data\_INDIANA$Year == j,]

if (county %in% temp\_data$County) {

if (week == 0) {

a = cbind(INDIANA\_2018[INDIANA\_2018$County == county,c(1:99,100)],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

INDIANA\_2018[INDIANA\_2018$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):203)],

INDIANA\_2018[INDIANA\_2018$County == county,c(204:205+week)],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

INDIANA\_2018[INDIANA\_2018$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

} else if (week == 39) {

a = INDIANA\_2018[INDIANA\_2018$County == county,]

} else {

a = cbind(INDIANA\_2018[INDIANA\_2018$County == county,c(1:99,100:(100+week))],

temp\_data[temp\_data$County == county,c((100+week+1):151)],

INDIANA\_2018[INDIANA\_2018$County == county,c(152,153:(153+week))],

temp\_data[temp\_data$County == county,c((153+week+1):204)],

INDIANA\_2018[INDIANA\_2018$County == county,c(205:(205+week))],

temp\_data[temp\_data$County == county,c((205+week+1):256)],

INDIANA\_2018[INDIANA\_2018$County == county,c(257,258:(258+(week+1)\*11 - 1))],

temp\_data[temp\_data$County == county,c((258+(week+1)\*11):697)])

}

# data\_county[c, c(1:99,100:(100+week))] = Illinois\_2017[Illinois\_2017$County == county,c(1:99,100:(100+week))]

# data\_county[c, c((100+week+1):151)] = temp\_data[temp\_data$County == county,c((100+week+1):151)]

# data\_county[c, c(152,153:(153+week))] = Illinois\_2017[Illinois\_2017$County == county,c(152,153:(153+week))]

# data\_county[c, c((153+week+1):204)] = temp\_data[temp\_data$County == county,c((153+week+1):204)]

# data\_county[c, c(205:(205+week))] = Illinois\_2017[Illinois\_2017$County == county,c(205:(205+week))]

# data\_county[c, c((205+week+1):256)] = temp\_data[temp\_data$County == county,c((205+week+1):256)]

# data\_county[c, c(257,258:(258+(week+1)\*11 - 1))] = Illinois\_2017[Illinois\_2017$County == county,c(257,258:(258+(week+1)\*11 - 1))]

# data\_county[c, c((258+(week+1)\*11):697)] = temp\_data[Illinois\_2017$County == county,c((258+(week+1)\*11):697)]

} else {

a = INDIANA\_2018[INDIANA\_2018$County == county,]

}

data\_county = rbind(data\_county,a)

}

yhat0 <- predict(model\_RFE150\_2018, newdata= data\_county[,c(rbf\_variables$V1)])

predictions\_INDIANA\_2018[year,week + 1] = sum(yhat0)/length(yhat0)

year = year + 1

}

week = week + 1

}

acre\_value = sum(INDIANA\_2018$Acre.Value)

real\_Yield = matrix(0,40,2)

max\_predicts = matrix(0,40,2)

for (i in 1:40) {

max\_value = max(predictions\_INDIANA\_2018[,i])

max\_predicts[i,1] = max\_value

max\_predicts[i,2] = i

real\_Yield[i,1] = sum(INDIANA\_2018$Value)/length(yhat0)

real\_Yield[i,2] = i

}

min\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = min(predictions\_INDIANA\_2018[,i])

min\_predicts[i,1] = min\_value

min\_predicts[i,2] = i

}

median\_predicts = matrix(0,40,2)

for (i in 1:40) {

min\_value = median(predictions\_INDIANA\_2018[,i])

median\_predicts[i,1] = min\_value

median\_predicts[i,2] = i

}

plot(x = max\_predicts[,2], y = max\_predicts[,1],ylim = c(0,300),lty =2,col='red',type= 'l', ylab = "Yield",xlab = "week", main="INDIANA 2018")

lines(x = min\_predicts[,2],y=min\_predicts[,1],lty =2,col='red')

lines(x = median\_predicts[,2], y = median\_predicts[,1], lty =1, col='red')

lines(x = real\_Yield[,2], y = real\_Yield[,1], lty =1, col='blue')

```

```{r warning=FALSE}

#calculate the RRMSE

yhat\_INDIANA\_2018\_RFE150 = predict(model\_RFE150\_2018, newdata= INDIANA\_2018[,c(rbf\_variables$V1)])

res = INDIANA\_2018$Value - yhat\_INDIANA\_2018\_RFE150

INDIANA\_2018\_RFE150\_RMSE = sqrt(sum(res^2)/length(res))

INDIANA\_2018\_RFE150\_RRMSE = INDIANA\_2018\_RFE150\_RMSE/mean(INDIANA\_2018$Value)

print(c(INDIANA\_2018\_RFE150\_RMSE, INDIANA\_2018\_RFE150\_RRMSE))

RRMSE\_for\_county\_INDIANA\_2018\_RFE150 = sqrt(res^2)/INDIANA\_2018$Value

write.table(RRMSE\_for\_county\_INDIANA\_2018\_RFE150, "/Users/akaike/Documents/fall22/STAT615/final pj/RRMSE\_RBF150\_5437/RRMSE\_for\_county\_INDIANA\_2018\_RFE150.txt", append = FALSE, sep = " ", dec = ".",

row.names = TRUE, col.names = TRUE)

```