MA615 Unit 1 final

Zhaosheng-Xie

Introduction

This project uses row data “berries.csv” to produce a tidy data. Also, I extract subset of tidy data to do EDA and PCA. You will see notation after “##” and “#”. “##” means primary item and “#” means branch item.

read the data

```
berries <- read_csv("berries.csv", col_names = TRUE)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   Year = col_double(),
##   `Week Ending` = col_logical(),
##   `Ag District` = col_logical(),
##   `Ag District Code` = col_logical(),
##   County = col_logical(),
##   `County ANSI` = col_logical(),
##   `Zip Code` = col_logical(),
##   Region = col_logical(),
##   Watershed = col_logical(),
##   `CV (%)` = col_logical()
## )
```

```
## See spec(...) for full column specifications.
```

```
Data = berries
```

Data cleaning

```
##Remove single value
Data %<>% select(-c("Program","Geo Level","Ag District","Week Ending",8:15,21))
#Remove NA,D and process (Z)
Data <- filter(Data,Value!='(D)',Value!='(NA)')
Data$Value[which(Data$Value=='(Z)')] <- 0
```

```
##Split Data Item
new <- Data
# new <- separate(new,col = "Data Item",into = c("species","definition","unit"),sep = ",",remov
e = TRUE)
# new <- separate(new,col = "Data Item",into = c("q","w","e","r"),sep = ",",remove = TRUE)
# new$unit <- tail(strsplit(new$`Data Item`,split=",")[[1]],1)
```

Because Data Item is mess, I split it into several columns and change every wrong item into a right value. Finally, I extract right values from each columns and delete redundant columns.

```
#1.Unit
nr <- nrow(new)
for (i in 1:nr) {
  new$unit[i] <- tail(strsplit(new$`Data Item`,split=",")[[i]],1)

}
```

```
## Warning: Unknown or uninitialised column: `unit`.
```

```
new1 <- new
new2 <- new1
#Replace untidy data
new2$unit[which(new2$unit=="STRAWBERRIES - ACRES HARVESTED"|
                  new2$unit==" WILD - ACRES HARVESTED"|
                  new2$unit==" RED - ACRES HARVESTED"|
                  new2$unit==" TAME - ACRES HARVESTED"|
                  new2$unit=="RASPBERRIES - ACRES HARVESTED"|
                  new2$unit==" BLACK - ACRES HARVESTED")] <- "ACRES HARVESTED"
new2$unit[which(new2$unit=="STRAWBERRIES - ACRES PLANTED")] <- "ACRES PLANTED"
unique(new2$unit)
```

```
##  [1] " MEASURED IN $ / LB"     " MEASURED IN $ / CWT"
##  [3] "ACRES HARVESTED"         " MEASURED IN LB"
##  [5] " MEASURED IN LB / ACRE"  " MEASURED IN $"
##  [7] " AVG"                    "ACRES PLANTED"
##  [9] " MEASURED IN CWT"        " MEASURED IN CWT / ACRE"
## [11] " MEASURED IN $ / TON"    " MEASURED IN TONS"
```

```
#2.Type
new3 <- new2
new3 %<>% separate(`Data Item`, c("B","type", "meas", "what"), sep = ",",remove = FALSE)
```

```
## Warning: Expected 4 pieces. Missing pieces filled with `NA` in 1819 rows [1, 4,
## 5, 6, 7, 8, 11, 14, 17, 20, 21, 26, 27, 28, 29, 30, 33, 34, 35, 36, ...].
```

```
new3 %<>% select(-B)
new3 %<>% separate(type,c("b1", "type", "b2", "lab1", "lab2"), " ")
```

```
## Warning: Expected 5 pieces. Additional pieces discarded in 130 rows [4, 5, 7,
## 26, 28, 33, 46, 129, 533, 1767, 1822, 1823, 1824, 1828, 1829, 1844, 1846, 1847,
## 1853, 1856, ...].
```

```
## Warning: Expected 5 pieces. Missing pieces filled with `NA` in 5594 rows [2, 3,
## 9, 10, 12, 13, 15, 16, 18, 19, 22, 23, 24, 25, 31, 32, 35, 36, 37, 38, ...].
```

```
new3[is.na(new3)] <- " "  ## OK now Data Item has been split into parts
# unique(new3$type)
#I found there is something redundant about new3$type. There will be 3 types in the final data: tame, wild and bearing. So I remove others.
new3$type[which(new3$type=="MEASURED"
                  # new3$type=="FRESH"|
                  # new3$type=="PROCESSING"|
                  # new3$type=="NOT"|
                  # new3$type=="UTILIZED"|
                  # new3$type=="BLACK"|
                  # new3$type=="RED"
                  )] <- " "
new3$type[which(new3$type=="NOT")] <- "NOT SOLD"
new3$type[which(new3$type=="FRESH")] <- "FRESH MARKET"
```

```
#3.Production
#The true values are hided in columns"lab1,lab2,meas,what"
new4 <- new3
new4 %<>% select(-c(`State ANSI`,b1,b2))
#settle these 4 columns and paste into 1 column
# unique(new4$lab1)
new4$lab1[which(new4$lab1=="$"|
                new4$lab1=="-"|
                new4$lab1=="ACRES"|
                new4$lab1=="LB"|
                new4$lab1=="CWT")] <- " "
# unique(new4$lab2)
new4$lab2[which(new4$lab2=="/"|
                new4$lab2=="HARVESTED")] <- " "
# unique(new4$meas)
new4$meas[which(new4$meas==" MEASURED IN $ / LB"|
                new4$meas==" MEASURED IN LB / ACRE"|
                new4$meas==" MEASURED IN LB / ACRE / YEAR"|
                new4$meas==" MEASURED IN $"|
                new4$meas==" MEASURED IN $ / CWT"|
                new4$meas==" MEASURED IN NUMBER"|
                new4$meas==" MEASURED IN CWT"|
                new4$meas==" MEASURED IN LB"|
                new4$meas==" MEASURED IN LB / ACRE / APPLICATION"|
                new4$meas==" MEASURED IN PCT OF AREA BEARING"|
                new4$meas==" MEASURED IN $ / TON")] <- " "
new4$meas[which(new4$meas==" FRESH MARKET - PRICE RECEIVED"|
                new4$meas==" PROCESSING - PRICE RECEIVED"  )] <- "PRICE RECEIVED"
new4$meas[which(new4$meas==" FRESH MARKET - PRODUCTION"|
                new4$meas==" NOT SOLD - PRODUCTION"|
                new4$meas==" PROCESSING - PRODUCTION"|
                new4$meas==" UTILIZED - PRODUCTION")]  <- "PRODUCTION"
new4$meas[which(new4$meas==" UTILIZED - YIELD")] <- "YIELD"
# unique(new4$what)
new4 %<>% select(-what)
```

```
#combine these 3 columns
new4 %<>% mutate(production = str_trim(paste(lab1,lab2,meas)) )
#unique(new4$production)
new4$production[which(new4$production=="PRICE")] <- "PRICE RECEIVED"
new4 %<>% select(-c(lab1,lab2,meas))
#process column production
new4$production[c(4,5,7)] <- "PRICE RECEIVED"
```

```
##onto Domain
new5 <- new4
# new5$Domain %>% unique()

new5 %<>% separate(Domain, c("D_left", "D_right"), sep = ", ")
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1791 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
# new5$D_left %>% unique()
# new5$D_right %>% unique()

new5[is.na(new5)] <- " "

# And now Domain Category


# new5$`Domain Category` %>% unique()

new5 %<>% separate(`Domain Category`, c("DC_left", "DC_right"), sep = ", ")
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 1922
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 1983
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 2042
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 2107
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 2174
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 3995
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4052
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4107
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4169
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4233
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 20 rows [184, 259,
## 332, 414, 498, 1336, 1385, 1431, 1478, 1531, 1932, 1993, 2052, 2117, 2184, 4005,
## 4062, 4117, 4179, 4243].
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1801 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
# looks like DC_left combines labels

head(new5$DC_left %>% unique(),n=20)
```

```
## [1] "NOT SPECIFIED"
## [2] "CHEMICAL"
## [3] "FERTILIZER: (NITROGEN)"
## [4] "FERTILIZER: (PHOSPHATE)"
## [5] "FERTILIZER: (POTASH)"
## [6] "FERTILIZER: (SULFUR)"
## [7] "CHEMICAL, INSECTICIDE: (CYFLUMETOFEN<U+00A0>= 138831)"
```

```
head(new5$DC_right %>% unique(), n=20)
```

```
##  [1] NA
##  [2] "FUNGICIDE: (BOSCALID = 128008)"
##  [3] "FUNGICIDE: (CYPRODINIL = 288202)"
##  [4] "FUNGICIDE: (FLUDIOXONIL = 71503)"
##  [5] "FUNGICIDE: (MYCLOBUTANIL = 128857)"
##  [6] "FUNGICIDE: (PYRACLOSTROBIN = 99100)"
##  [7] "FUNGICIDE: (TOTAL)"
##  [8] "HERBICIDE: (TOTAL)"
##  [9] "INSECTICIDE: (ACEQUINOCYL = 6329)"
## [10] "INSECTICIDE: (BIFENAZATE = 586)"
## [11] "INSECTICIDE: (METHOXYFENOZIDE = 121027)"
## [12] "INSECTICIDE: (PYRETHRINS = 69001)"
## [13] "INSECTICIDE: (SPINETORAM = 110007)"
## [14] "INSECTICIDE: (SPINOSAD = 110003)"
## [15] "INSECTICIDE: (TOTAL)"
## [16] "INSECTICIDE: (ZETA-CYPERMETHRIN = 129064)"
## [17] "OTHER: (TOTAL)"
## [18] "INSECTICIDE: (BT KURSTAKI ABTS-351 = 6522)"
## [19] "FUNGICIDE: (AZOXYSTROBIN = 128810)"
## [20] "FUNGICIDE: (BLAD = 30006)"
```

```
## work on DC_left first

new5 %<>% separate(DC_left, c("DC_left_l", "DC_left_r"), sep = ": ")
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 1922
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 1983
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 2042
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 2107
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 2174
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 3995
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4052
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4107
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4169
```

```
## Warning in gregexpr(pattern, x, perl = TRUE): PCRE error
##  'UTF-8 error: isolated byte with 0x80 bit set'
##  for element 4233
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 5781 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
# new5$DC_left_l %>% unique()
# new5$DC_left_r %>% unique()
## now work on DC_right

head(new5$DC_right %>% unique(), n=20)
```

```
##  [1] NA
##  [2] "FUNGICIDE: (BOSCALID = 128008)"
##  [3] "FUNGICIDE: (CYPRODINIL = 288202)"
##  [4] "FUNGICIDE: (FLUDIOXONIL = 71503)"
##  [5] "FUNGICIDE: (MYCLOBUTANIL = 128857)"
##  [6] "FUNGICIDE: (PYRACLOSTROBIN = 99100)"
##  [7] "FUNGICIDE: (TOTAL)"
##  [8] "HERBICIDE: (TOTAL)"
##  [9] "INSECTICIDE: (ACEQUINOCYL = 6329)"
## [10] "INSECTICIDE: (BIFENAZATE = 586)"
## [11] "INSECTICIDE: (METHOXYFENOZIDE = 121027)"
## [12] "INSECTICIDE: (PYRETHRINS = 69001)"
## [13] "INSECTICIDE: (SPINETORAM = 110007)"
## [14] "INSECTICIDE: (SPINOSAD = 110003)"
## [15] "INSECTICIDE: (TOTAL)"
## [16] "INSECTICIDE: (ZETA-CYPERMETHRIN = 129064)"
## [17] "OTHER: (TOTAL)"
## [18] "INSECTICIDE: (BT KURSTAKI ABTS-351 = 6522)"
## [19] "FUNGICIDE: (AZOXYSTROBIN = 128810)"
## [20] "FUNGICIDE: (BLAD = 30006)"
```

```
new5 %<>% separate(DC_right, c("DC_right_l", "DC_right_r"), sep = ": ")


new5[is.na(new5)] <- " "

#  OK now we need to eliminate the redundancy
# fine and remove redundant columns

# remove column new5$DC_left_l
new5 %<>%  select(-DC_left_l)

# remove column DC_right_l
new5 %<>% select(-DC_right_l)


# remove "Chemical" and joint the columns

new5 %<>% mutate(D_left = "CHEMICAL", D_left = "")

new5 %<>% mutate(Chemical=paste(D_left, D_right))

new5 %<>% select(-c(D_left, D_right))
```

```
# Final tidy data

new5$DC_left_r %>% unique() # rename chemical_family
```

```
## [1] " "           "(NITROGEN)"  "(PHOSPHATE)" "(POTASH)"    "(SULFUR)"
```

```
new5 %<>% rename( Chem_family = DC_left_r, Materials = DC_right_r)
new5 %<>% mutate(Chemical = str_trim(paste(Chem_family, Chemical)))
new5 %<>% select(-c(`Data Item`,Chem_family))
new5 %<>% rename( Type = type, Unit = unit, Production = production)
new5 %<>% select(Year,Period,State,Commodity,Type,Production,Chemical,Materials,Unit,Value)
Tidyberry <- new5
# write.table(Tidyberry,"D:/MSSP/Rdata/615/Berry/Tidyberry.csv",col.names = TRUE,row.name = FAL
SE,sep = ",")
#This Tidyberry contains all berries.
```

Filter some data from Tidyberry

The data is tidy now. The majority of production is for application, so I filter rows of data in some situation.

```
#I choose raspberries and period=YEAR
Rberry <- Tidyberry %>% filter((Commodity=="RASPBERRIES") & (Period=="YEAR"))
Rberry %<>% select(-c(Period, Commodity))
## look at chemicals being applied to food
unfood <- Rberry %>% filter(Production=="APPLICATIONS")
unfood %<>% filter(Value !=  "(NA)")
#unique(unfood$Unit)
#in this case, I choose unit=AVG.
unfood %<>% filter(Unit == " AVG")
unfood$Value <- as.numeric(unfood$Value)

unfood_1 <- unfood %>%  select(Year, State, Chemical, Value)
unfood_1$Value <- as.numeric(unfood_1$Value)
unfood_1 %<>% pivot_wider(names_from = Chemical, values_from = Value)
```

```
## Warning: Values are not uniquely identified; output will contain list-cols.
## * Use `values_fn = list` to suppress this warning.
## * Use `values_fn = length` to identify where the duplicates arise
## * Use `values_fn = {summary_fun}` to summarise duplicates
```

```
## Because of using the pivot_wider, some data was a list. Change to sum.

for (i in 1:6) {
  f <- as.numeric(unlist(unfood_1$INSECTICIDE[i]))
  unfood_1$INSECTICIDE[i] <- sum(f)
}
for (i in 1:6) {

  f <- as.numeric(unlist(unfood_1$FUNGICIDE[i]))
  unfood_1$FUNGICIDE[i] <- sum(f)
}
for (i in 1:6) {
  f <- as.numeric(unlist(unfood_1$HERBICIDE[i]))
  unfood_1$HERBICIDE[i] <- sum(f)
}
for (i in 1:6) {
  f <- as.numeric(unlist(unfood_1$OTHER[i]))
  unfood_1$OTHER[i] <- sum(f)
}
for (i in 1:6) {
  f <- as.numeric(unlist(unfood_1$`(NITROGEN)`[i]))
  unfood_1$`(NITROGEN)`[i] <- sum(f)
}
for (i in 1:6) {
  f <- as.numeric(unlist(unfood_1$`(PHOSPHATE)`[i]))
  unfood_1$`(PHOSPHATE)`[i] <- sum(f)
}
for (i in 1:6) {
  f <- as.numeric(unlist(unfood_1$`(POTASH)`[i]))
  unfood_1$`(POTASH)`[i] <- sum(f)
}

unfood_1$FUNGICIDE <- as.numeric(unfood_1$FUNGICIDE)
unfood_1$INSECTICIDE <- as.numeric(unfood_1$INSECTICIDE)
unfood_1$HERBICIDE <- as.numeric(unfood_1$HERBICIDE)
unfood_1$OTHER <- as.numeric(unfood_1$OTHER)
unfood_1$`(NITROGEN)` <- as.numeric(unfood_1$`(NITROGEN)`)
unfood_1$`(PHOSPHATE)` <- as.numeric(unfood_1$`(PHOSPHATE)`)
unfood_1$`(POTASH)` <- as.numeric(unfood_1$`(POTASH)`)

#kable(head(Rberry,n=10)) %>% kable_styling(fixed_thead = T, font_size = 10)
head(Rberry,n=10)
```

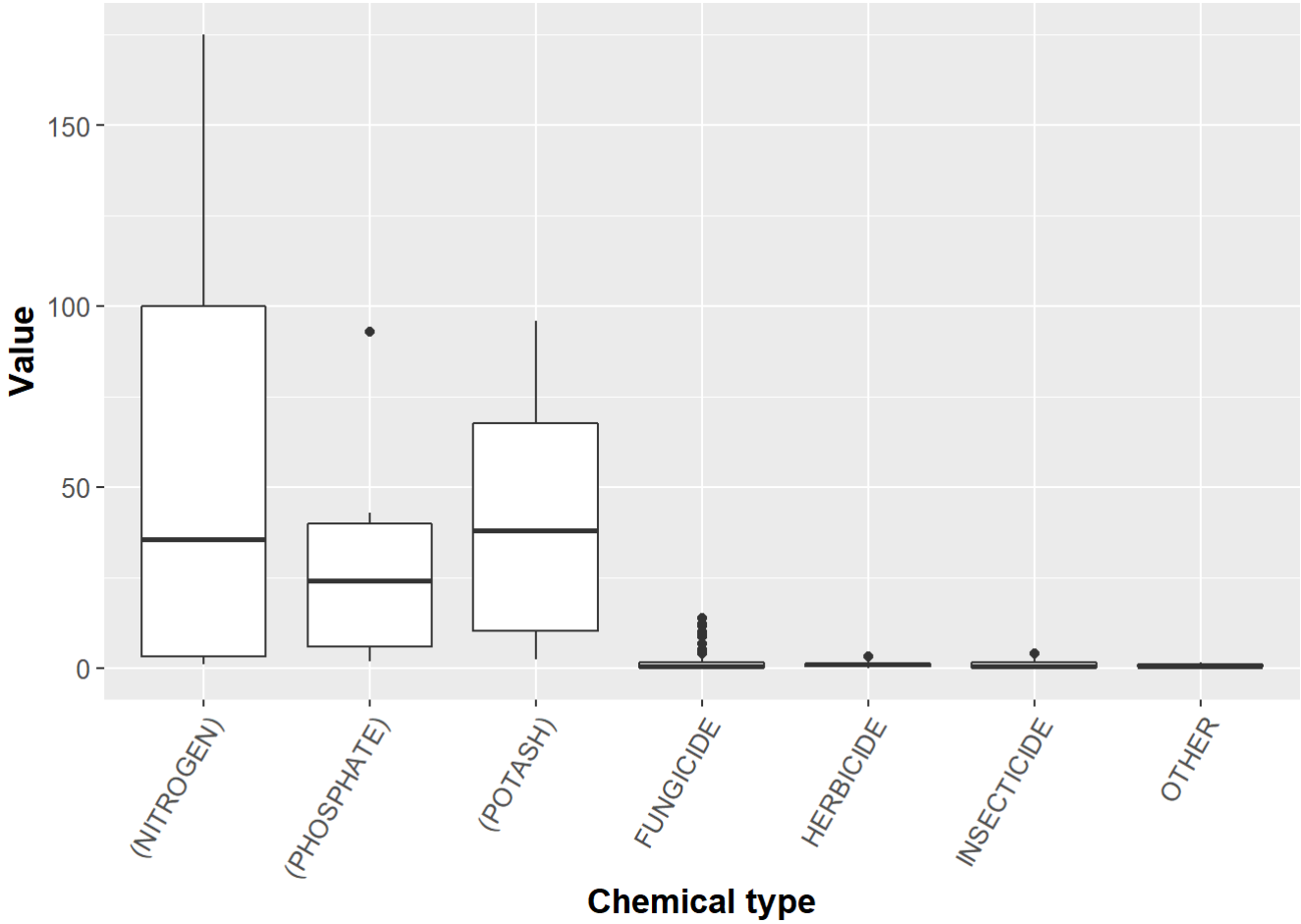
Y…	State	Type	Production	Chemical	Materials	Unit
<dbl>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
2019	CALIFORNIA					ACRES HAR
2019	CALIFORNIA					MEASURED
2019	CALIFORNIA					MEASURED
2019	CALIFORNIA	BEARING	APPLICATIONS	FUNGICIDE	(BOSCALID = 128008)	MEASURED
2019	CALIFORNIA	BEARING	APPLICATIONS	FUNGICIDE	(CYPRODINIL = 288202)	MEASURED
2019	CALIFORNIA	BEARING	APPLICATIONS	FUNGICIDE	(FLUDIOXONIL = 71503)	MEASURED

Y…	State	Type	Production	Chemical	Materials	Unit
<dbl>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
2019	CALIFORNIA	BEARINGAPPLICATIONS		FUNGICIDE	(MYCLOBUTANIL = 128857)	MEASURED
2019	CALIFORNIA	BEARINGAPPLICATIONS		FUNGICIDE	(PYRACLOSTROBIN = 99100)	MEASURED
2019	CALIFORNIA	BEARINGAPPLICATIONS		FUNGICIDE	(TOTAL)	MEASURED
2019	CALIFORNIA	BEARINGAPPLICATIONS		HERBICIDE	(TOTAL)	MEASURED

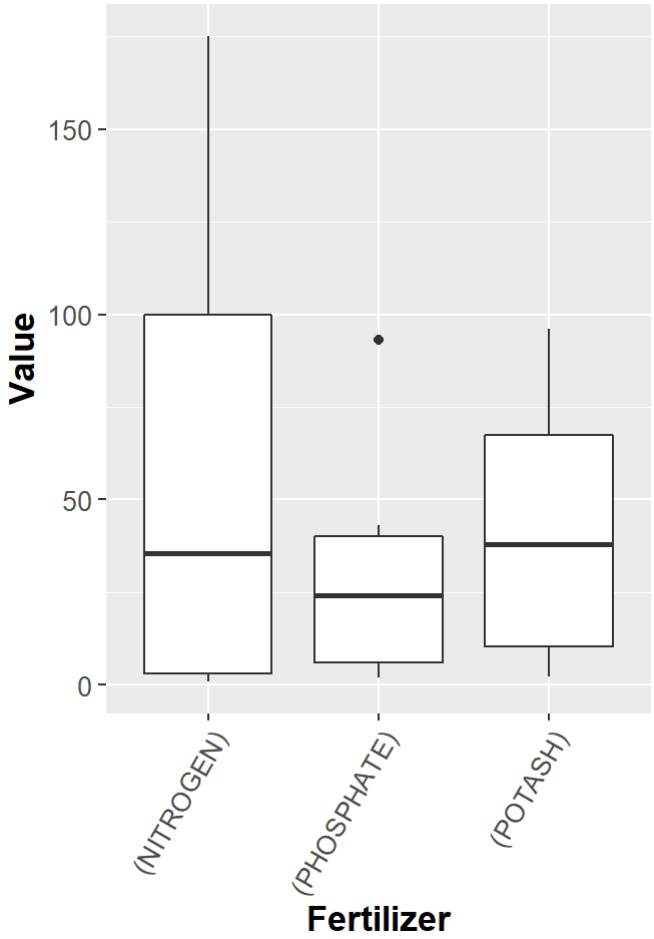
1-10 of 10 rows | 1-7 of 8 columns

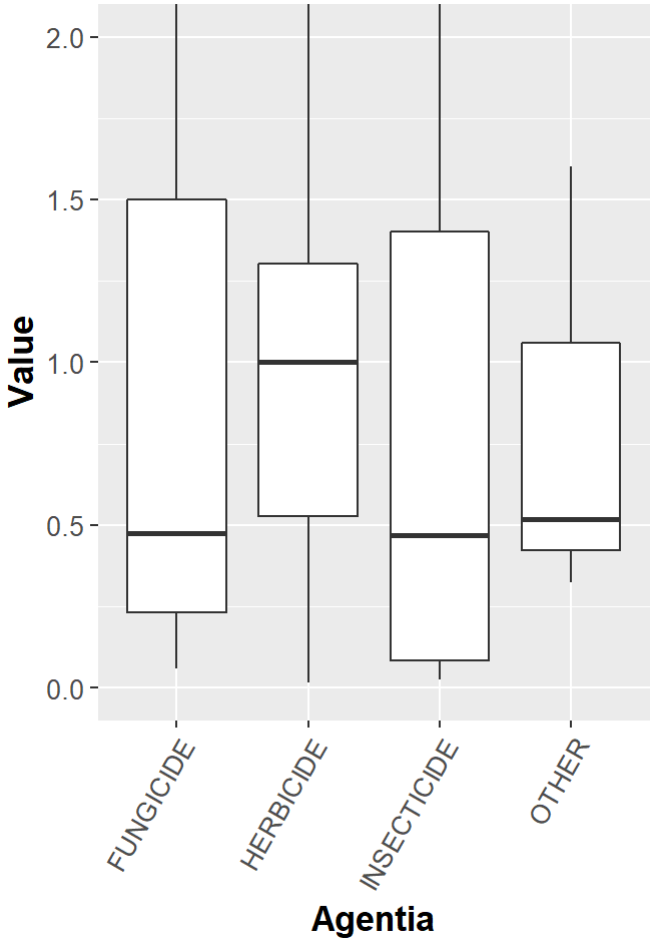
EDA

1.Chemical and Value



In this case, I separate the chemical into to part: agentia and fertilizer. I create an indicate to distinguish them. Two part are drawn separately, but I use grid.arrange function to combine them into same plot. It is clear that the value of raspberry using fertilizer(‘nitrogen’, ‘phosphate’ and ‘potash’) is much higher than those using agentia(‘fungicide’, ‘herbicide’, ‘insecticide’ and ‘other’)(because ‘other’ also have a small range so I put it in the agentia).



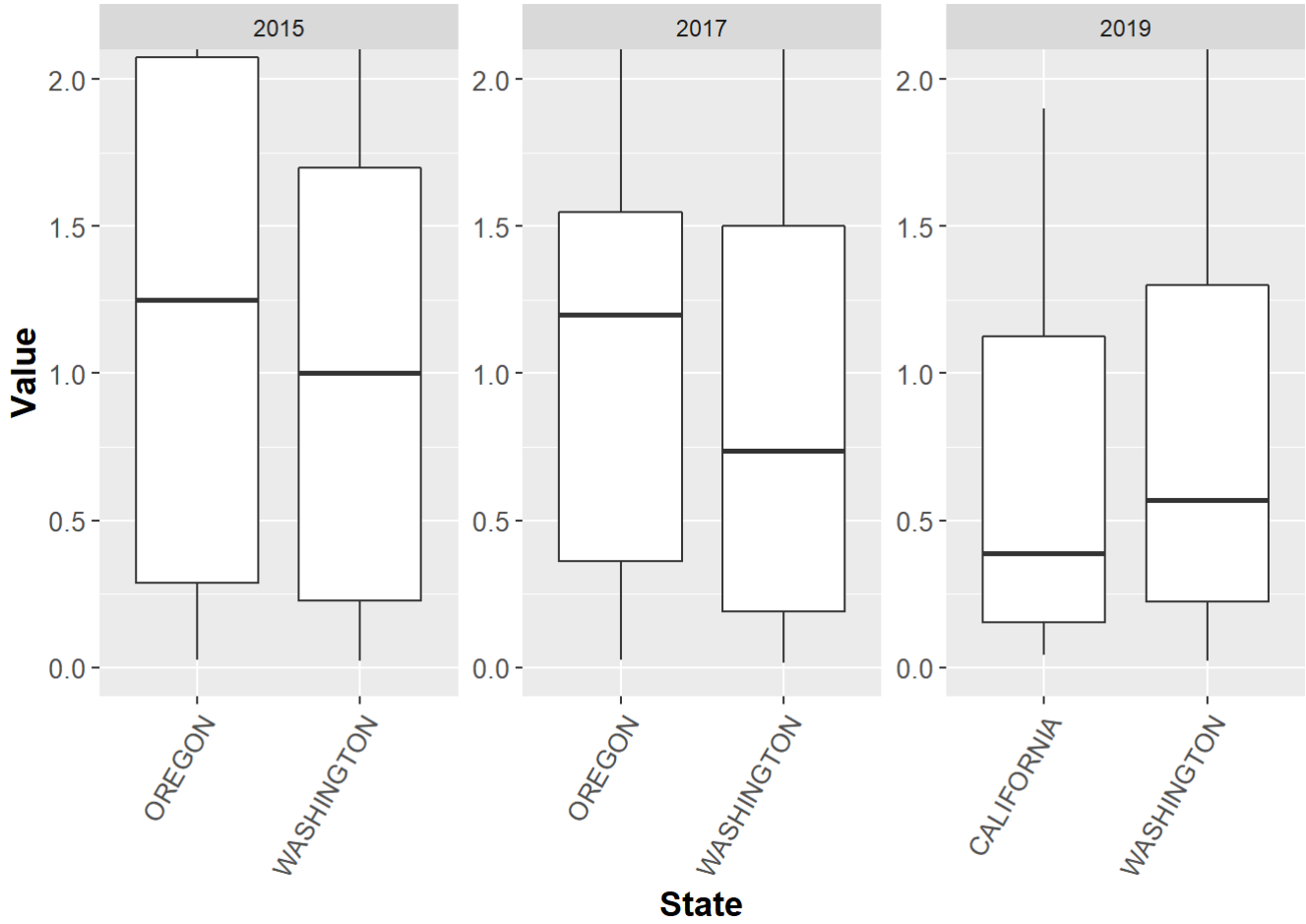


And also can see those outliers here.

Y…	State	Type	Production	Chemical	Materials	U…
<dbl>	<chr>	<chr>	<chr>	<chr>	<chr>	<chr>
2017	OREGON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG
2017	OREGON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG
2017	OREGON	BEARING	APPLICATIONS	FUNGICIDE	(CAPTAN = 81301)	AVG
2017	WASHINGTON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG
2017	WASHINGTON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG
2017	WASHINGTON	BEARING	APPLICATIONS	FUNGICIDE	(CAPTAN = 81301)	AVG
2017	WASHINGTON	BEARING	APPLICATIONS	FUNGICIDE	(CAPTAN = 81301)	AVG
2015	OREGON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG
2015	OREGON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG
2015	WASHINGTON	BEARING	APPLICATIONS	FUNGICIDE	(CALCIUM POLYSULFIDE = 76702)	AVG

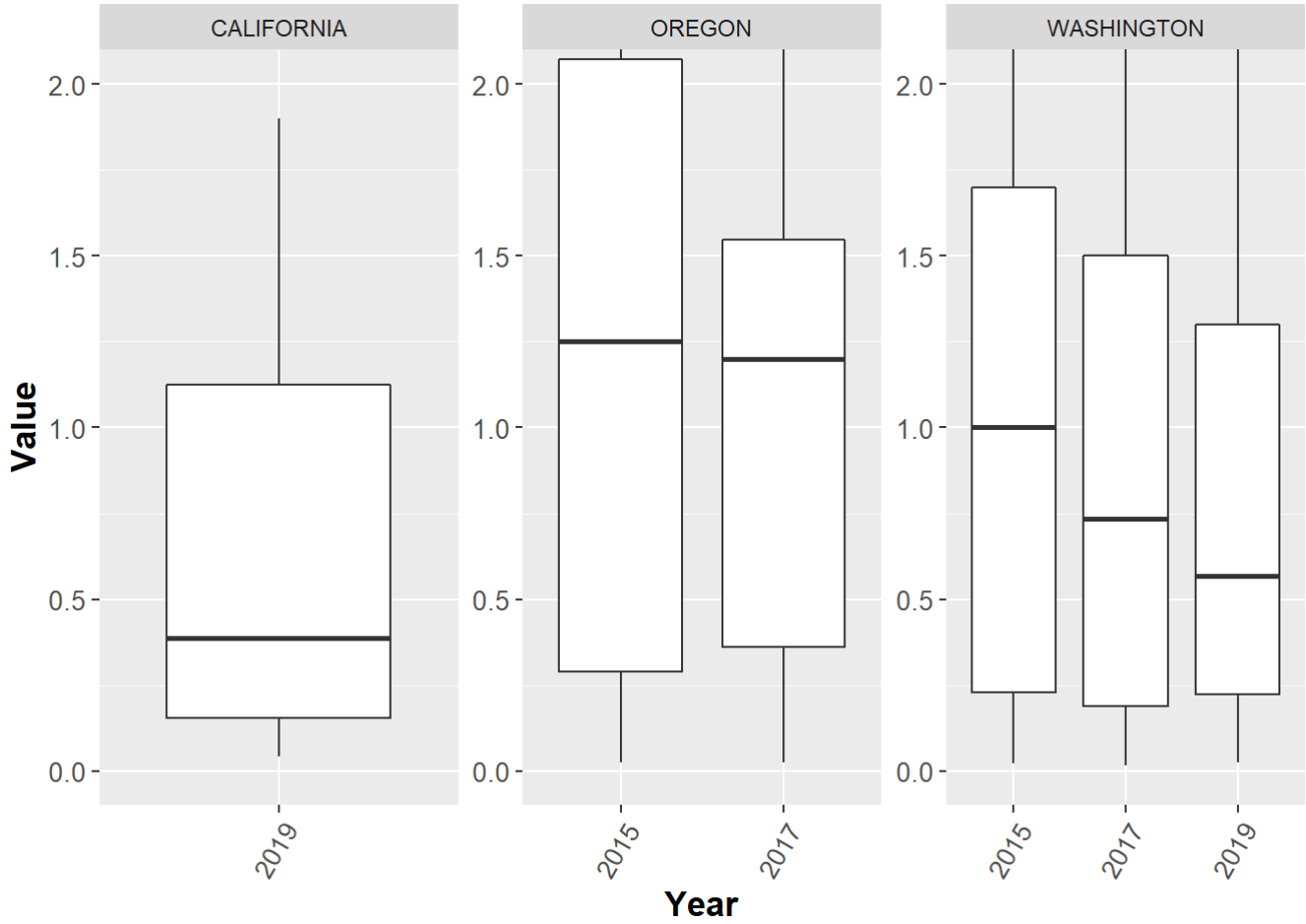
1-10 of 10 rows | 1-8 of 10 columns

2.state and value



3.Year and Value

The dataset only have one year of value in California so there is no much to discuss.

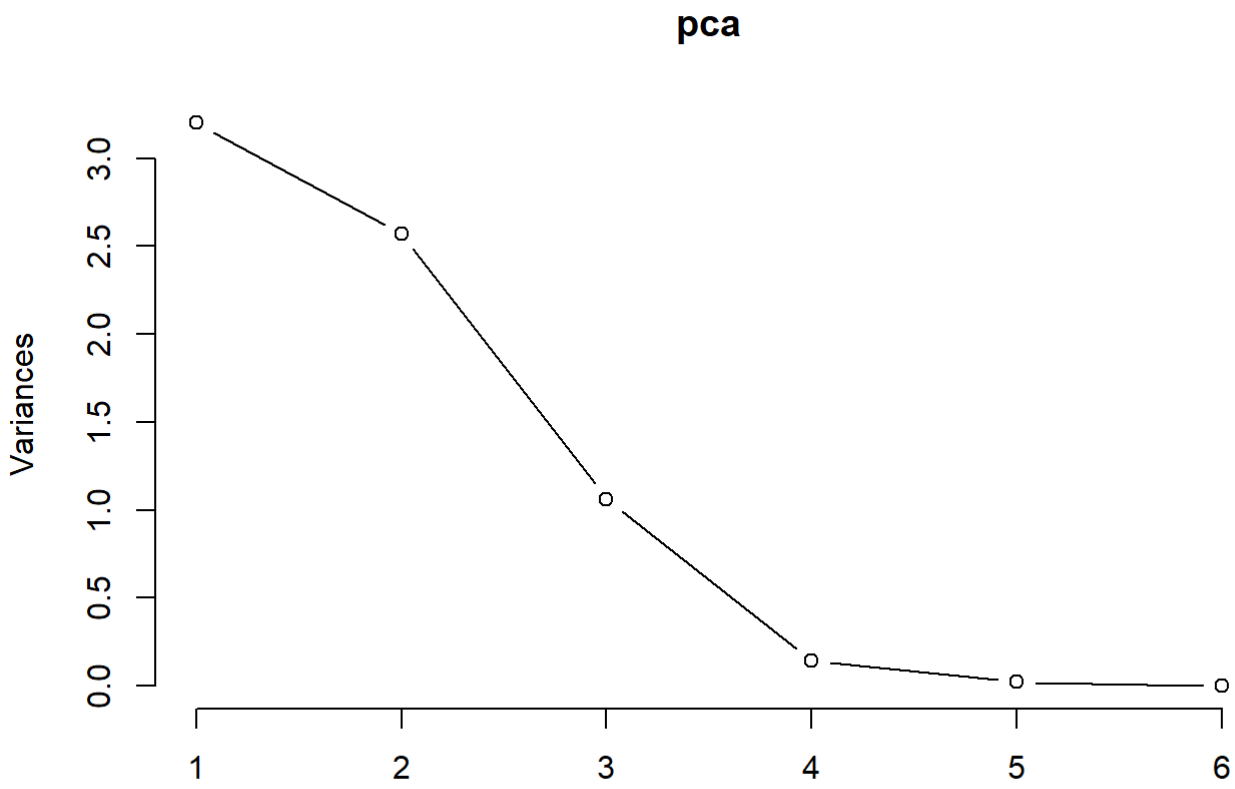


In 2015, the median of value of Oregon and Washington was close, around 1.1. In 2017, the median of value in Oregon is almost two times of that in Washington. In 2019, there was a small difference between the situation of California and Washington. From 2015 to 2017, the median of value of Oregon almost did not change, but the top value decreased. The median of value of Washington decreased by around 0.25. From 2015 to 2019, the median of value of Washington and range were continuously decreased.

PCA

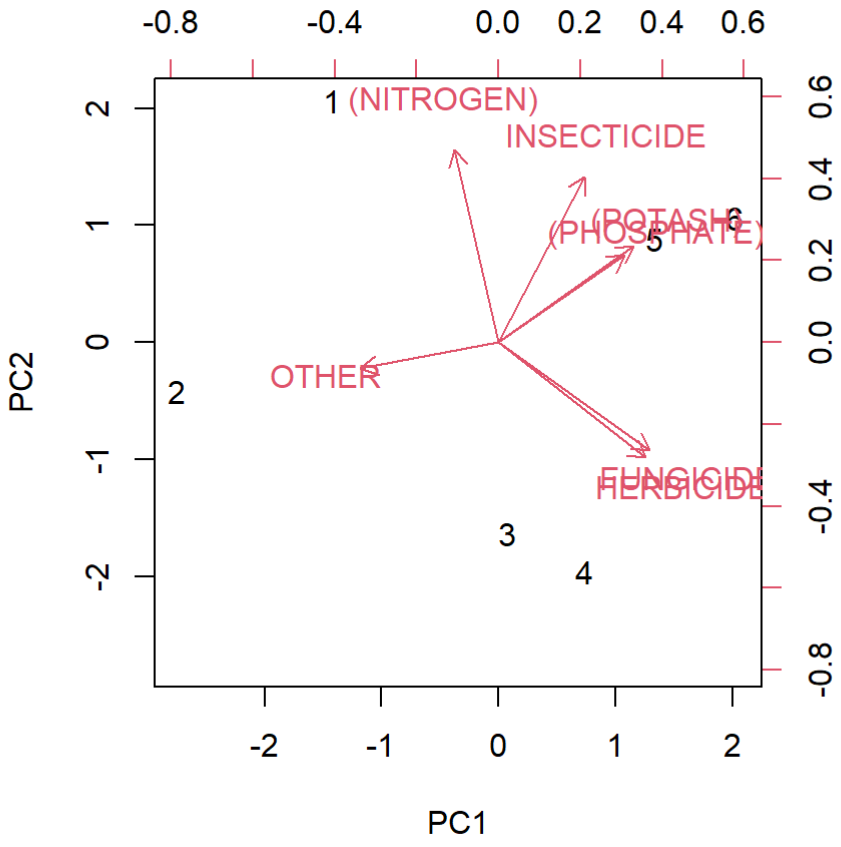
Correlations between chemical type.

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5       PC6
## Standard deviation     1.7892 1.6033 1.0311 0.37724 0.15084 2.362e-16
## Proportion of Variance 0.4573 0.3672 0.1519 0.02033 0.00325 0.000e+00
## Cumulative Proportion  0.4573 0.8245 0.9764 0.99675 1.00000 1.000e+00
```



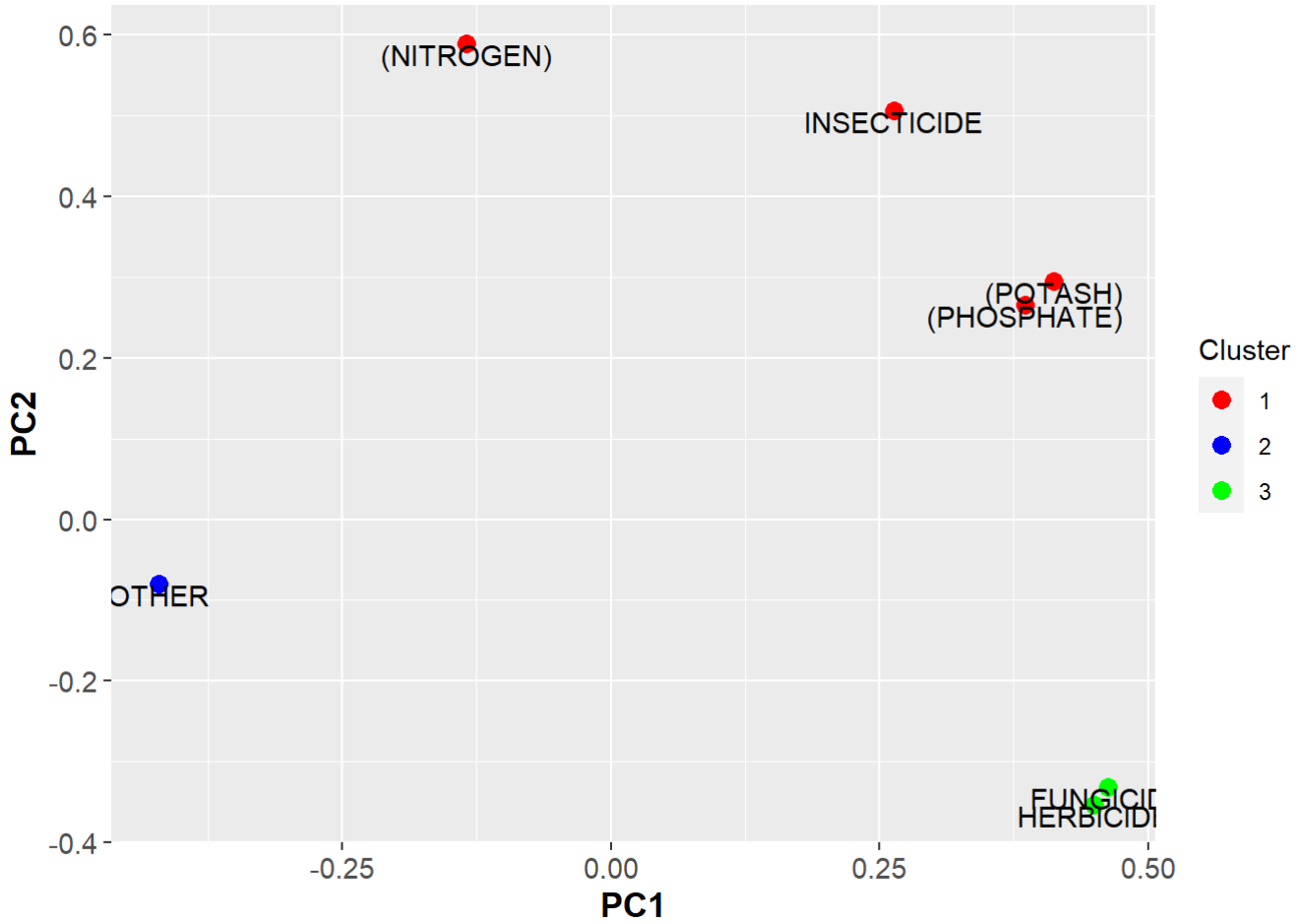
The PCA provides 6 components and 99% of the total variance is attributed to the first 4 components.

```
biplot(pca2.1, scale = 0)
```



> And in the biplot I can see the relationship between each variables. The size of the angle between vectors determines the correlation of the variables, which is the desired indicator to achieve the objective for this analysis. A small angle indicates a strong positive correlation, 90 degrees represents no correlation and 180 degrees represents a negative correlation.

For example, Phosphate and potash is almost coincide; others and herbicide have negative correlation.



In this plot, it is clear that chemical can be separate into to three part, just I mentioned before, apart from 'other', fertilizer('nitrogen', 'phosphate' and 'potash') gathered as cluster1, and agentia('fungicide', 'herbicide', 'insecticide') gathered as cluster3.

Conclusion

Honestly speaking, this project is a big and painful job for me as a rookie. I even did not know how to begin. But after discussing with others and having classes, I went into condition gradually. In this process, I found a big problem: I had some ideas to handle the problems but I can not put my theoretical ideas into practice. In balance, my R skill was too terrible to handle. So I tried to use internet and textbooks to improve my skill. This method was painful because I always made mistake and I even wanted to punch my laptop when R reported errors. But I do learn a lot about no matter in data cleaning, EDA, PCA or shiny.

Citation

[1] Exploratory data analysis into the relationship between different types of crime in London (https://towardsdatascience.com/exploratory-data-analysis-into-the-relationship-between-different-types-of-crime-in-london-20c328e193ff)
[2] R for Data Science (https://r4ds.had.co.nz/)
[3] dmorison/eda-relationships-between-crime-london (https://github.com/dmorison/eda-relationships-between-crime-london)
[4] PCA.rmd (published%20in%20BB) [5] ag_data(2).rmd (published%20in%20BB)
[6] [many of my classmates](I discuss with them a lot. For example, Hao Shen and I discussed problem in data cleaning. He mentioned that we could use # to replace value of “Data Item”. It’s a very good way to separate Data Item by using sep=“#”. By contrast, my way was clumsy. Also, Zhe Yu gave me many advice about EDA and PCA.)