

Average User CA

Ariella Fuzaylov

3/28/2021

Loading Data From CSV and Update Header

```
my.data=read.csv("Online Recipe Sharing.csv", header=TRUE)
colnames(my.data)<-c("Timestamp", "Age", "Primary.Meal.Prepper", "Household.Dietary.Restriction",
"Home.Cooking.Rate",
"Primary.Recipe.Format",
"Primary.Search.Website",
"Enjoyed.Website.Searching", "Comments.Enjoyed.Website.Searching", "NOT.Enjoyed.Website.Searching", "Comments.Enjoyed.Website.Browsing", "Comments.NOT.Enjoyed.Website.Browsing",
"Previous.Recipe.Search.Frequency",
"Browsing.While.Searching.Frequency",
"Click.Rate",
"Search.Browse.Same.Websites",
"Primary.Browsing.Website",
"Enjoyed.Website.Browsing",
"Comments.Enjoyed.Website.Browsing", "NOT.Enjoyed.Website.Browsing", "Comments.NOT.Enjoyed.Website.Browsing",
"Source.of.Influential.Reviews", "Frequency.Reviews.Effect.Behavior",
"Frequency.Seek.Out.Review",
"Frequency.of.Review",
"Frequency.of.Recipe.Saving",
"Method.of.Recipe.Saving",
"Modification.Frequency",
"Modification.Influence.Factors",
"Modification.Record.Frequency",
"Modification.Record.Method",
"Satisfaction.with.Available.Record.Methods",
"Interest.in.Improved.Record.Method",
"Frequency.of.Recipe.Discussion", "Frequency.of.Reading.Discussion",
"Primary.Discussion.Medium", "Enjoyed.Features.of.Discussion.Mediums", "Ingredients.L.V.Above",
"Ingredients.L.Comments.Inline.V.Below", "Ingredients.Above.Comments.Below.V.Inline", "Ingredients.By.Step.V.Scroll.L",
"Ingredients.By.Step.V.Scroll.L",
"Ingredients.Above.V.Scroll.L")
```

Re-Factor Data

If Respondent indicated that they search and browse on the same websites, populate the empty cells with the same data. This assumes that the user's searching behavior is exactly the same as the browsing behavior if the user selected yes for searching and browsing on the same websites.

```

for (i in 1:nrow(my.data)){
  if (my.data$Search.Browse.Same.Websites[i]=="No"){
    my.data$Primary.Browsing.Website[i]<-my.data$Primary.Search.Website[i]
    my.data$Enjoyed.Website.Browsing[i]<-my.data$Enjoyed.Website.Searching[i]
    my.data$NOT.Enjoyed.Website.Browsing[i]<-my.data$NOT.Enjoyed.Website.Searching[i]
  }
}

```

Since the data set is small, I am consolidating some of the categories.

- Primary Meal Prepper will be Respondent if the individual taking the survey indicated that they are the primary meal prepper in their household or if they cook for themselves, and other in all other cases.
- Dietary restriction will become a yes or no question
- Home Cooking Rate will become Daily if the respondents cooks at home most days, weekly if the respondent cooks several times a week, and monthly is the respondent cooks a couple times a month.

```

my.data.factored<-my.data
my.data.factored$Age<-as.factor(my.data$Age)

my.data.factored$Primary.Meal.Prepper<-as.factor(my.data.factored$Primary.Meal.Prepper)

my.data.factored$Household.Dietary.Restriction<-as.factor(my.data.factored$Household.Dietary.Restriction)

my.data.factored$Home.Cooking.Rate<-as.factor(my.data.factored$Home.Cooking.Rate)

my.data.factored$Ingredients.L.V.Above<-as.factor(my.data.factored$Ingredients.L.V.Above)
my.data.factored$Ingredients.By.Step.V.Above<-as.factor(my.data.factored$Ingredients.By.Step.V.Above)
my.data.factored$Ingredients.Above.V.Scroll.L<-as.factor(my.data.factored$Ingredients.Above.V.Scroll.L)
my.data.factored$Ingredients.L.Comments.Inline.V.Below<-as.factor(my.data.factored$Ingredients.L.Comments.Inline.V.Below)
my.data.factored$Ingredients.By.Step.V.Scroll.L<-
  as.factor(my.data.factored$Ingredients.By.Step.V.Scroll.L)
my.data.factored$Ingredients.Above.Comments.Below.V.Inline<-
  as.factor(my.data.factored$Ingredients.Above.Comments.Below.V.Inline)
my.data.factored$Click.Rate<-as.factor(my.data.factored$Click.Rate)
my.data.factored$Search.Browse.Same.Websites<-as.factor(my.data.factored$Search.Browse.Same.Websites)
my.data.factored$Interest.in.Improved.Record.Method<-as.factor(my.data.factored$Interest.in.Improved.Record.Method)

my.data.factored<- mutate(my.data.factored,
  Age = fct_collapse(Age,
    YA = c("18 - 24 years old", "25 - 34 years old"),
    Adult = c("35 - 44 years old", "45 - 54 years old", "55 - 64 years old"),
    Primary.Meal.Prepper = fct_collapse(Primary.Meal.Prepper,
      Respondent = c("You", "I cook for myself"),
      other_level = "Other"),
    Household.Dietary.Restriction=fct_collapse(Household.Dietary.Restriction,
      No="None",
      other_level = "Yes"),
    Home.Cooking.Rate=fct_collapse(Home.Cooking.Rate,
      Daily=c("Almost every meal", "Daily", "Every meal"),
      Weekly.or.Less=c("Several times a week", "Once or twice a week", "Monthly"),
      Ingredients.L.V.Above=fct_collapse(Ingredients.L.V.Above,
        Ing.L =c("A"),

```

```

        Ing.Above=("B"),
        other_level = "EMPTY"),
Ingredients.By.Step.V.Above=fct_collapse(Ingredients.By.Step.V.Above,
        Ing.By.Step=c("A"),
        Ing.Above=c("B"),
        other_level = "EMPTY"),
Ingredients.Above.V.Scroll.L=fct_collapse(Ingredients.Above.V.Scroll.L,
        Ing.Above=c("A"),
        Scroll.L=c("B"),
        other_level = "EMPTY"),
Ingredients.L.Comments.Inline.V.Below=fct_collapse(Ingredients.L.Comments.Inline.V.Below,
        Ing.L.Com.Inline=c("A"),
        Ing.L.Com.Below=c("B"),
        other_level = "EMPTY"),
Ingredients.By.Step.V.Scroll.L=fct_collapse(Ingredients.By.Step.V.Scroll.L,
        Ing.By.Step=c("A"),
        Ing.Scroll=c("B"),
        other_level = "EMPTY"),
Ingredients.Above.Comments.Below.V.Inline=fct_collapse(Ingredients.Above.Comments.Below.V.Inline,
        Ing.Above.C.Below=c("A"),
        Ing.Above.C.Inline=c("B"),
        other_level = "EMPTY"),
Click.Rate=fct_collapse(Click.Rate, "0 - 3" =c("0 - 3"), "4 - 7"=c("4 - 7") ,
        "8 - 11"=c("8 - 11"), "16 - 19"=c("16 - 19")) ,
Search.Browse.Same.Websites=fct_collapse(Search.Browse.Same.Websites,Yes=c("Yes"), No=c("No"),
        other_level = "EMPTY"),
Interest.in.Improved.Record.Method=fct_collapse(Interest.in.Improved.Record.Method,Yes=c("Yes"), No=c("No"),
        other_level = "EMPTY"),
)

```

Website Recoding:

For the sake of this analysis any website that has a test kitchen that creates editorial content or is able to curate content from professional sources is a magazine, a website with one or two people testing recipes is a blog, and a website that allows users to contribute their own recipes is community based. The information for this classification is found on the website's about page. Additionally, media such as cookbooks and podcasts are classified under Influencers due to their personality driven nature.

Discussion Method Recoding:

Any type of online chatting be it texting, discord, etc. has been grouped together into Digital Chat. Any type of interpersonal communication where a chat method was not specified is grouped into verbal.

Note saving methods that mention remembering or memory are grouped into memory, while respondents that indicate that they do not take any type of notes and do not try to remember are grouped into None.

Modification Recoding:

Modification influence factors pertaining to diet, or nutrition are grouped together under the umbrella of "Diet".

Modification influence factors pertaining to personal preference for food, flavor, or preparation method are grouped together under the category of "Personal Preference.

Modification influence factors pertaining ingredients availability are grouped together under the category of "Ing. Availability"

```
unique(separate_rows(my.data.factored[32],1, sep = ";"))
```

```
## # A tibble: 18 x 1
##   Modification.Record.Method
##   <chr>
## 1 ""
## 2 "None"
## 3 "Mentally?"
## 4 "Digital notes"
## 5 "Physical notes"
## 6 "Mental note"
## 7 "I don't :o"
## 8 "Comments section provided for recipe"
## 9 "Memory"
## 10 "N/A"
## 11 "I mostly just remember it for next time "
## 12 "brainpower"
## 13 "I dont"
## 14 "I don't"
## 15 "I store it in my noggin"
## 16 "I don't.."
## 17 "i dont"
## 18 "i don't"
```

```
my.data.selected<-my.data.factored[c(6,7,8,10,17,18,20,22,23,28,37,30,32,38)]
variables<-c()
```

```
##This creates a vector that will recode the variables with the proper names
for (i in 1:ncol(my.data.selected)){
  temp<- my.data.selected[i]
  temp<-separate_rows(temp,1, sep = ";")
  variables<-append(variables,temp[[1]])
  variables<-unique(variables)
  data.frame(variables)
}
```

```
variables
```

```
## [1] "Mobile Website"
## [2] "Desktop Website"
## [3] "Digital photos of cookbook recipes"
## [4] "Cook Book"
## [5] "Printed from Internet"
## [6] "Video recipe"
## [7] "handwritten"
## [8] "Recipe cards"
## [9] "mom's recipes"
```

```

## [10] "Some old family recipes on 3x5 cards etc."
## [11] "Online Cooking Magazines (New York Times, Bon Appetit, etc.)"
## [12] "Blogs (Budget Bytes, Smitten Kitchen, etc.)"
## [13] "Google"
## [14] "YouTube"
## [15] "Community Based Cooking Websites (AllRecipes, etc.)"
## [16] "Edited recipe websites (e.g. Serious Eats)"
## [17] "Allrecipes "
## [18] "Pinterest"
## [19] "Cooks I follow their websites , ie againinstallgrain"
## [20] "TikTok"
## [21] "King Arthur Flour"
## [22] "Facebook"
## [23] "Reddit"
## [24] "epicurious"
## [25] "betty crocker's website"
## [26] "Serious Eats, Americaâ\200\231s Test Kitchen"
## [27] "Serious Eats!"
## [28] "Instagram"
## [29] "King Arthur Flour, NYTimes, NPR"
## [30] "My family and friends directly"
## [31] "Betty Crocker's website"
## [32] ""
## [33] "Any website that buries the recipe under tons of useless text"
## [34] "Online Cooking Magazines (New York Times, Bon Appetit, etc)"
## [35] "Instagram "
## [36] "instagram"
## [37] "I do not dislike"
## [38] "None"
## [39] "Immediate family / Friends"
## [40] "Groups on social media"
## [41] "Recipe Comments/ Other user's reviews"
## [42] "Influencers (Instagram, YouTube, Tiktok, etc.)"
## [43] "Cookbooks, podcasts"
## [44] "Flavcity on facebook"
## [45] "Browser Bookmarks"
## [46] "Digital filing system"
## [47] "Memory"
## [48] "search history"
## [49] "Save function built into your website of choice"
## [50] "Physical filing system"
## [51] "I donâ\200\231t "
## [52] "brain"
## [53] "memory"
## [54] "I tell myself I won't forget how to make this recipe and then I do :("
## [55] "tiktok favorites"
## [56] "In person conversation with others"
## [57] "Verbal"
## [58] "Word of mouth"
## [59] "Discord"
## [60] "With friends"
## [61] "Friends"
## [62] "Text with friends"
## [63] "Google Docs"

```

```

## [64] "Messages with friends and family "
## [65] "talking to people"
## [66] "Actual conversation with a human in person or on the phone"
## [67] "talking"
## [68] "discussing them with friends"
## [69] "Talking to friends and family"
## [70] "Chatting with pals"
## [71] "Privately with family/friends"
## [72] "I don't really. I read comments and will directly give recs to friends"
## [73] "Various channels of communication (i.e. personal text, group chats, etc.)"
## [74] "i don't"
## [75] "I text people, or I check reviews on google"
## [76] "discuss with family and friends "
## [77] "conversations/texts"
## [78] "Messaging platforms"
## [79] "don't really do this"
## [80] "Dietary restriction"
## [81] "Allergies"
## [82] "Flavor or food preference"
## [83] "Nutritional or dietary need"
## [84] "Necessary ingredient(s) unavailable"
## [85] "Use ingredients in your fridge"
## [86] "Someone's recommendation"
## [87] "laziness, sometimes recipes are too complicated for no reason"
## [88] "Mentally?"
## [89] "Digital notes"
## [90] "Physical notes"
## [91] "Mental note"
## [92] "I donâ\200\231t :o"
## [93] "Comments section provided for recipe"
## [94] "N/A"
## [95] "I mostly just remember it for next time "
## [96] "brainpower"
## [97] "I dont"
## [98] "I don't"
## [99] "I store it in my noggin"
## [100] "I don't.."
## [101] "i dont"
## [102] "5 star review system"
## [103] "Dedicated groups for different interests"
## [104] "Up/down voting posts"
## [105] "Up/down voting comments"
## [106] "Collapsible comment threads"
## [107] "Comment replies"
## [108] "Comment threads"
## [109] "Inline comments"

```

```

cleaned.variables<-c(
  "Mobile",
  "Desktop",
  "Digital",
  "Physical Print",
  "Physical Print",
  "Digital",

```

```

"Physical Family",
"Physical Family",
"Physical Family",
"Physical Family",
"Mags",
"Blogs",
"Google",
#"Youtube",
"Video",
"Community Based" ,
"Mags",
"Community Based" ,
"Pinterest",
"Blogs",
# "TikTok",
"Video",
"Mags",
"Facebook",
"Reddit",
"Mags",
"Mags",
"Mags",
"Mags",
"Instagram",
"Mags",
"Friends/Family",
"Blogs",
"NA",
"Blogs",
"Mags",
"Instagram",
"Instagram",
"None",
"None",
"Friends/Family",
"Online Groups",
"Other Users",
"Influencers",
"Influencers",
"Facebook",
"Browser Bookmarks",
"Digital Filing",
"Memory",
"Search History",
"Save Function",
"Physical Filing",
"None",
"Memory",
"Memory",
"Memory",
"Save Function",
"Verbal",
"Verbal",

```

```
"Verbal",
"Digital Chat",
"Verbal",
"Verbal",
"Digital Chat",
"Google Docs",
"Digital Chat",
"Verbal",
"Verbal",
"Verbal",
"Verbal",
"Verbal",
"Verbal",
"Verbal",
"Verbal",
"Verbal",
"Digital Chat",
"None",
"Digital Chat",
"Verbal",
"Digital Chat",
"Digital Chat",
"None",
"Diet",
"Diet",
"Preference",
"Diet",
"Ing. Availability",
"Ing. Availability",
"Recommendation",
"Preference",
"Memory",
"Digital",
"Physical",
"Memory",
"None",
"Comments",
"None",
"Memory",
"Memory",
"None",
"None",
"Memory",
"None",
"None",
"5 Star Review",
"Groups",
"Up/Down Vote Posts",
"Up/Down Vote Com.",
"Collapse Comment",
"Comment Reply",
"Comment Thread",
"Inline Comment")
```



```
names(cleaned.variables)<-variables
```

Functions for Cleaning Data

```
dummies<-function(search.data, to.clean){
  col.names<-c(names(search.data))
  col.names<-col.names[col.names!=to.clean]
  search.data.clean<- search.data%>% separate_rows(all_of(to.clean), sep = ";")

  search.data.clean[to.clean]<-
    as.character(cleaned.variables[search.data.clean[[to.clean]])]
  search.data.clean[to.clean]<-lapply(search.data.clean[to.clean],function(x) replace(x,is.na(x),"Empty"))

  search.data.dummies<-search.data.clean%>%
    select((to.clean))%>%
    dummy()%>%
    bind_cols(search.data.clean)%>%
    select(-(to.clean))%>%
    pivot_longer(cols=-col.names, names_to = "key", values_to = "value"%>%
    filter(value!=0)

  search.data.dummies<-search.data.dummies%>%
    unique()

  search.data.dummies<-search.data.dummies%>%
    spread(key, value, fill = 0)
}
```

Load Factored Data

```
search.data<-my.data.factored[-c(1,9,11,19,21)]

search.data<-data.frame(search.data)
new.names=c("Age", "Meal.Prepper","Dietary.Restriction","Home.Cook.Rate","Primary.Format.C","Primary.S.
  "Enjoyed.S.C","NOT.Enjoyed.S.C","Recipe.Search.F","Repeat.S.F","Browse.Search.F","Click.Rat
  "Search.Browse.Same","Primary.B.C","Enjoyed.B.C","NOT.Enjoyed.B.C", "Primary.R.C", "Influen
  "Use.R.F","Seek.R.F", "R.F","Save.F","Save.C","Mod.F","Why.Mod.C", "Mod.Note.F", "Mod.Note.
  "Note.Method.S","Potential.Note.Taker","Disc.F","Read.Disc.F","Disc.C","Enjoy.Disc.C", "Ing
  "Ing.L.Com.Inline.V.Below","Ing.Above.Com.Below.V.Inline", "Ing.By.Step.V.Above", "Ing.By
  "Ing.Above.V.Scroll.L"
  )
colnames(search.data)<-new.names

cleaned<-search.data
to.dummy<-select(cleaned, ends_with(".C"))
to.dummy.cols<-c(colnames(to.dummy))
for (col in to.dummy.cols){
  cleaned<-dummies(cleaned,c(col))
}
```

```
}
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use 'all_of(to.clean)' instead of 'to.clean' to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use 'all_of(col.names)' instead of 'col.names' to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
cleaned<-data.frame(cleaned)
to.num<-grepl(".C_",colnames(cleaned))
cleaned[to.num]<-as.integer(unlist(cleaned[to.num]))
```

```
cleaned[names(select(cleaned, ends_with(".F")))] [is.na(cleaned[names(select(cleaned, ends_with(".F"))])]
```

```
# aggregate(select(cleaned, where(is.numeric)), by=list(cleaned$Age,cleaned$Meal.Prepper,cleaned$Dietary
```

```
aggregate(select(cleaned, ends_with(".F")), by=list(cleaned$Meal.Prepper,cleaned$Dietary.Restriction,cl
```

```
##      Group.1 Group.2      Group.3 Recipe.Search.F Repeat.S.F Browse.Search.F
## 1 Respondent      No      Daily      2.909091      3.636364      3.090909
## 2      Other      No      Daily      2.800000      3.600000      2.900000
## 3 Respondent     Yes      Daily      3.157895      2.894737      3.263158
## 4      Other     Yes      Daily      2.333333      3.166667      2.000000
## 5 Respondent      No Weekly.or.Less      3.000000      3.250000      2.500000
## 6      Other      No Weekly.or.Less      1.500000      3.500000      3.000000
## 7 Respondent     Yes Weekly.or.Less      2.750000      3.500000      2.500000
## 8      Other     Yes Weekly.or.Less      4.500000      4.500000      1.500000
##      Use.R.F Seek.R.F      R.F      Save.F      Mod.F Mod.Note.F      Disc.F Read.Disc.F
## 1 3.272727 2.818182 1.727273 3.181818 3.181818      2.090909 2.818182      2.363636
## 2 3.800000 2.900000 2.700000 4.100000 3.100000      1.900000 3.600000      2.700000
## 3 3.631579 3.473684 2.736842 3.210526 2.631579      2.263158 3.421053      2.894737
## 4 3.666667 3.333333 3.500000 3.000000 2.500000      1.666667 2.333333      1.666667
## 5 3.750000 3.000000 4.000000 2.250000 3.250000      1.250000 3.250000      2.750000
## 6 2.000000 2.500000 1.500000 2.500000 2.000000      0.500000 0.500000      0.500000
## 7 2.000000 2.750000 1.750000 3.250000 2.750000      2.250000 2.000000      1.000000
## 8 1.500000 1.500000 2.000000 2.000000 3.500000      2.500000 2.000000      1.500000
```

```
#isolate frequent users
```

```
cleaned<- filter(cleaned, Home.Cook.Rate!="Monthly")
```

```
#REMOVE USERS WHO DID NOT COMPLETE ab TEST
```

```
cleaned<-filter(cleaned, Ing.L.Com.Inline.V.Below!="EMPTY")
```

```
cleaned.f<-cleaned
```

```
## construct dendrograms
```

```
hc.complete =hclust (dist(cleaned.f), method ="complete")
```

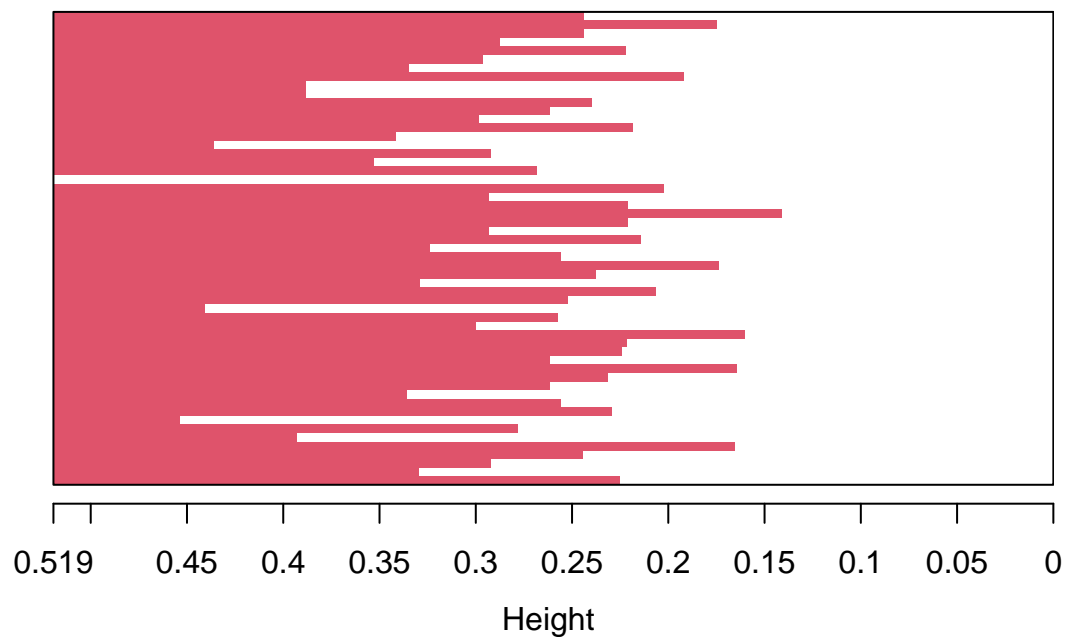
```
## Warning in dist(cleaned.f): NAs introduced by coercion
```

```
gower.dist <- daisy(cleaned.f, metric = c("gower"))
```

```
## Warning in daisy(cleaned.f, metric = c("gower")): binary variable(s) 26, 27, 28,  
## 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48,  
## 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68,  
## 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88,  
## 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106,  
## 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122,  
## 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138,  
## 139, 140, 141, 142 treated as interval scaled
```

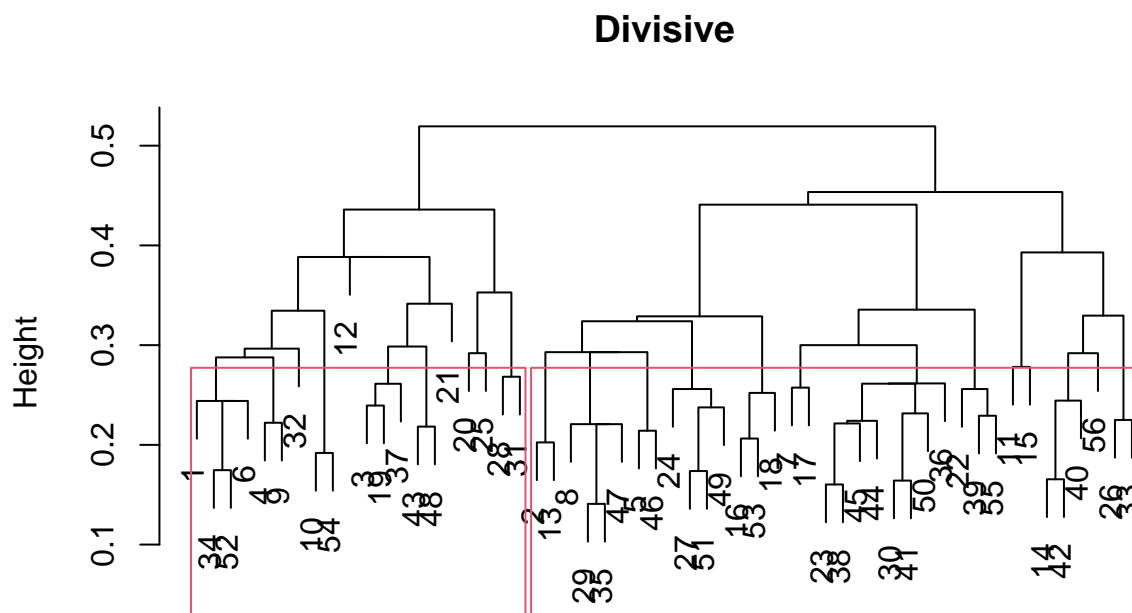
```
##plot dendrograms and groups  
divisive.clust <- diana(as.matrix(gower.dist),  
  diss = TRUE, keep.diss = TRUE)  
plot(divisive.clust, main = "Divisive")
```

Divisive



Divisive Coefficient = 0.56

```
rect.hclust(divisive.clust,k=2)
```

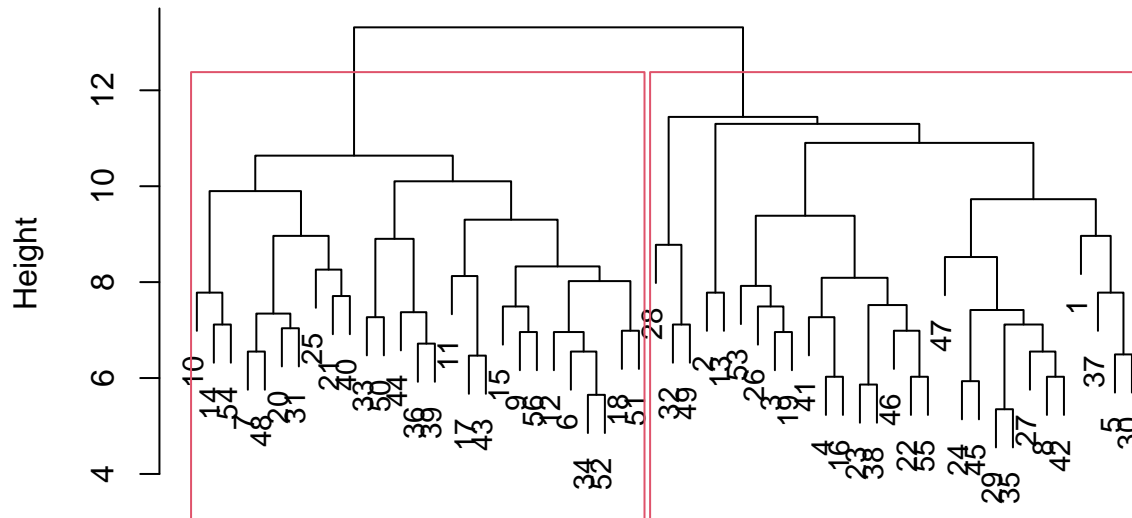


as.matrix(gower.dist)
Divisive Coefficient = 0.56

```
all.gower.g<-cutree(divisive.clust,k=2)
all.gower<-mutate(cleaned.f, cluster=as.factor(all.gower.g))

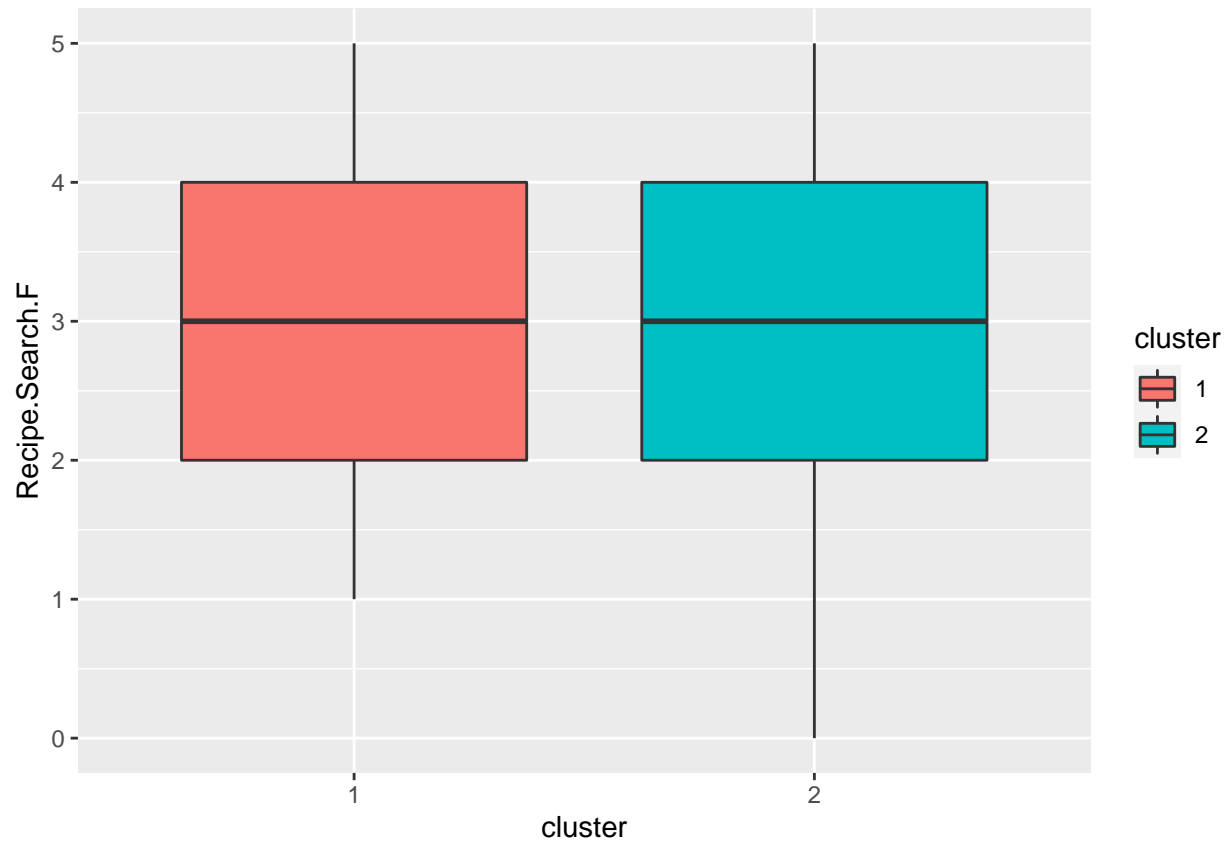
plot(hc.complete ,main = " Complete Linkage ", xlab="", sub = "",
cex =.9)
rect.hclust(hc.complete,k=2)
```

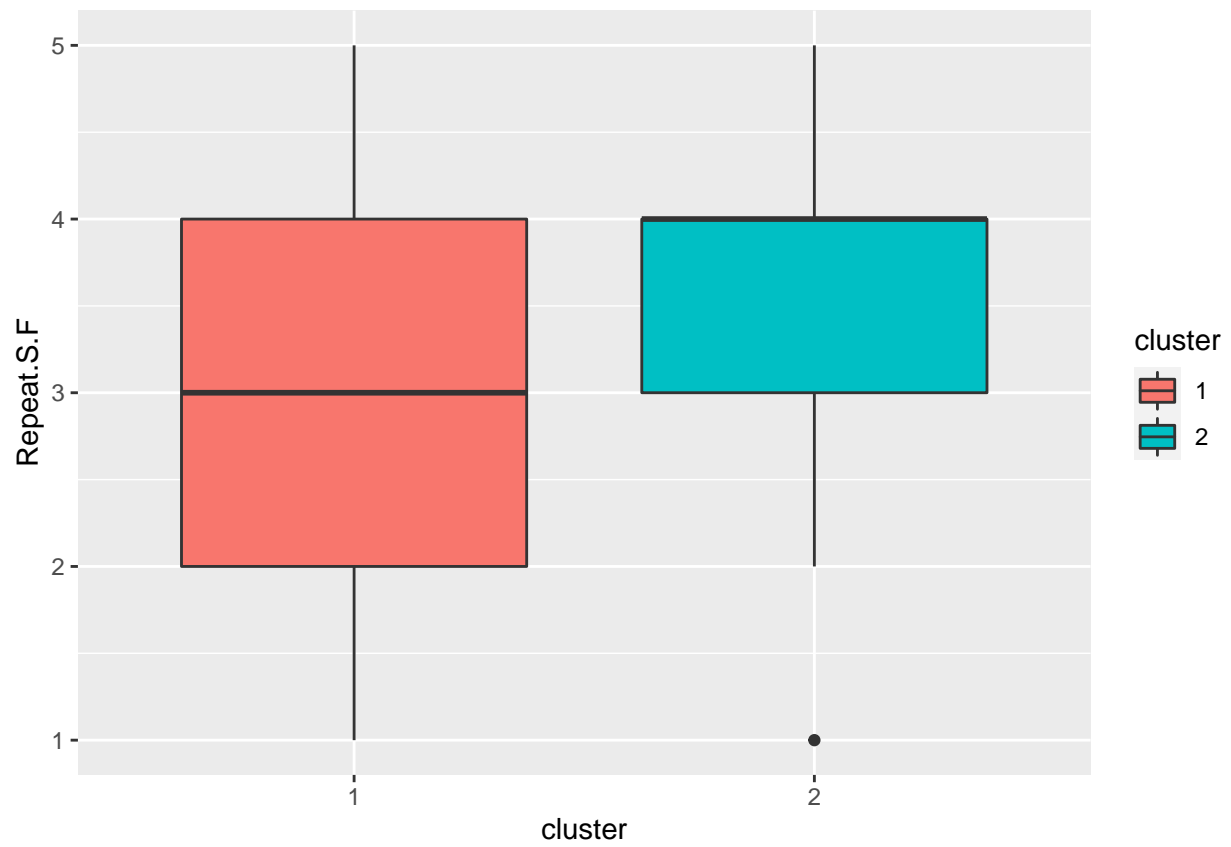
Complete Linkage

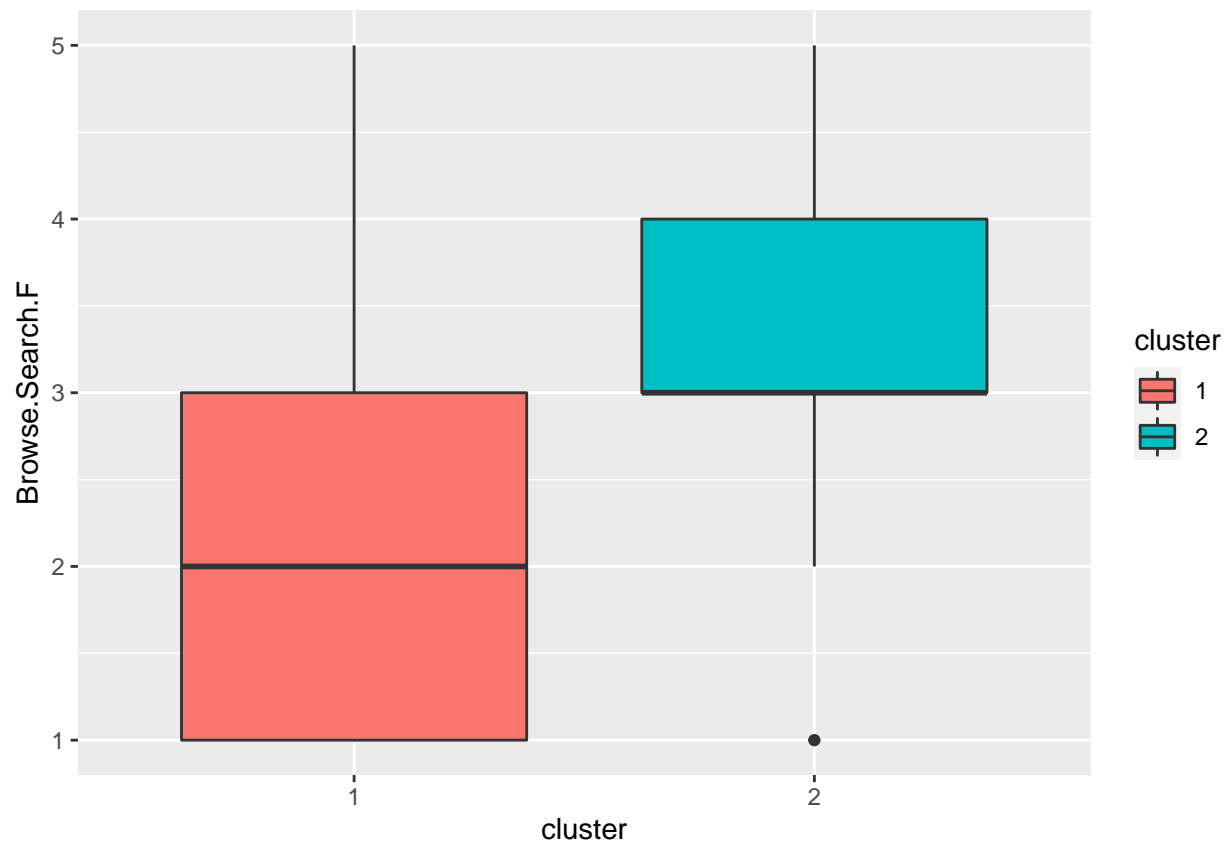


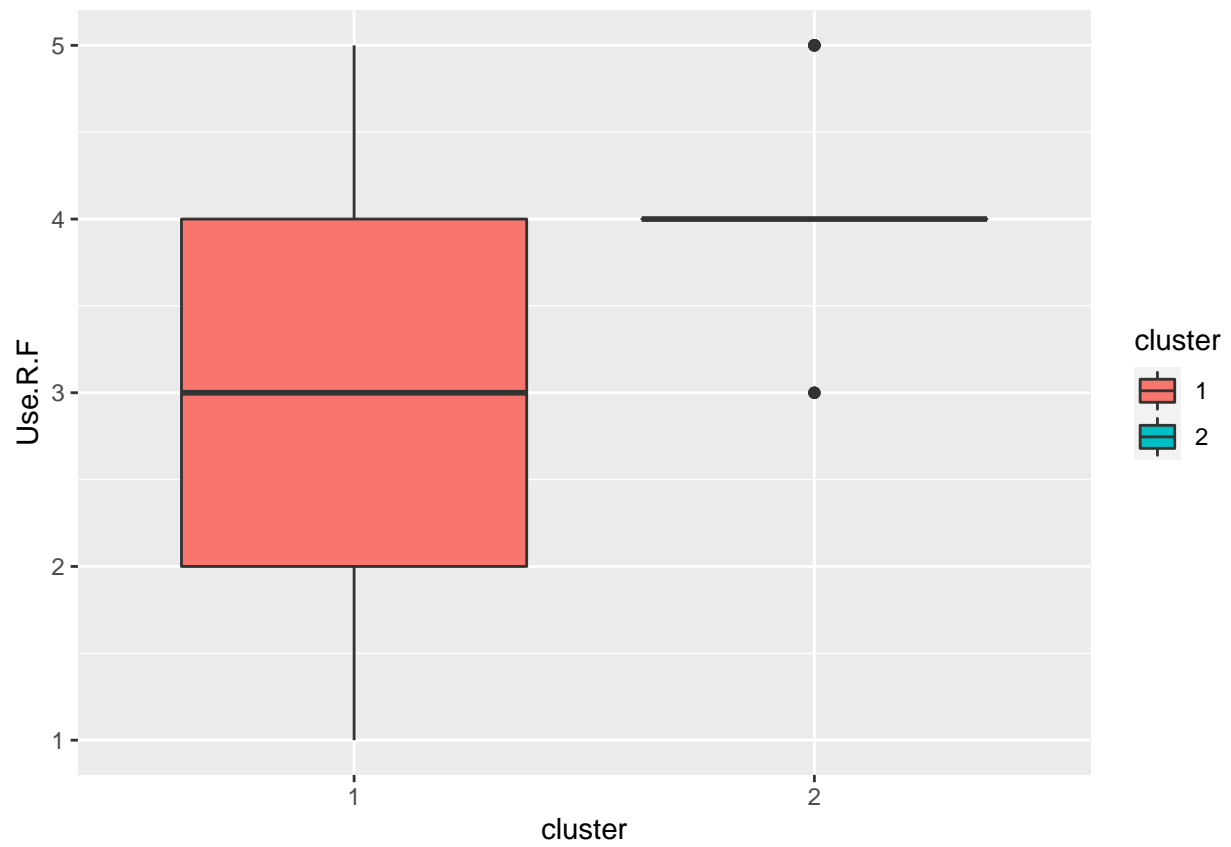
```
all.complete.g<-cutree(hc.complete, k=2)
all.complete<-mutate(cleaned.f, cluster=as.factor(all.complete.g))
```

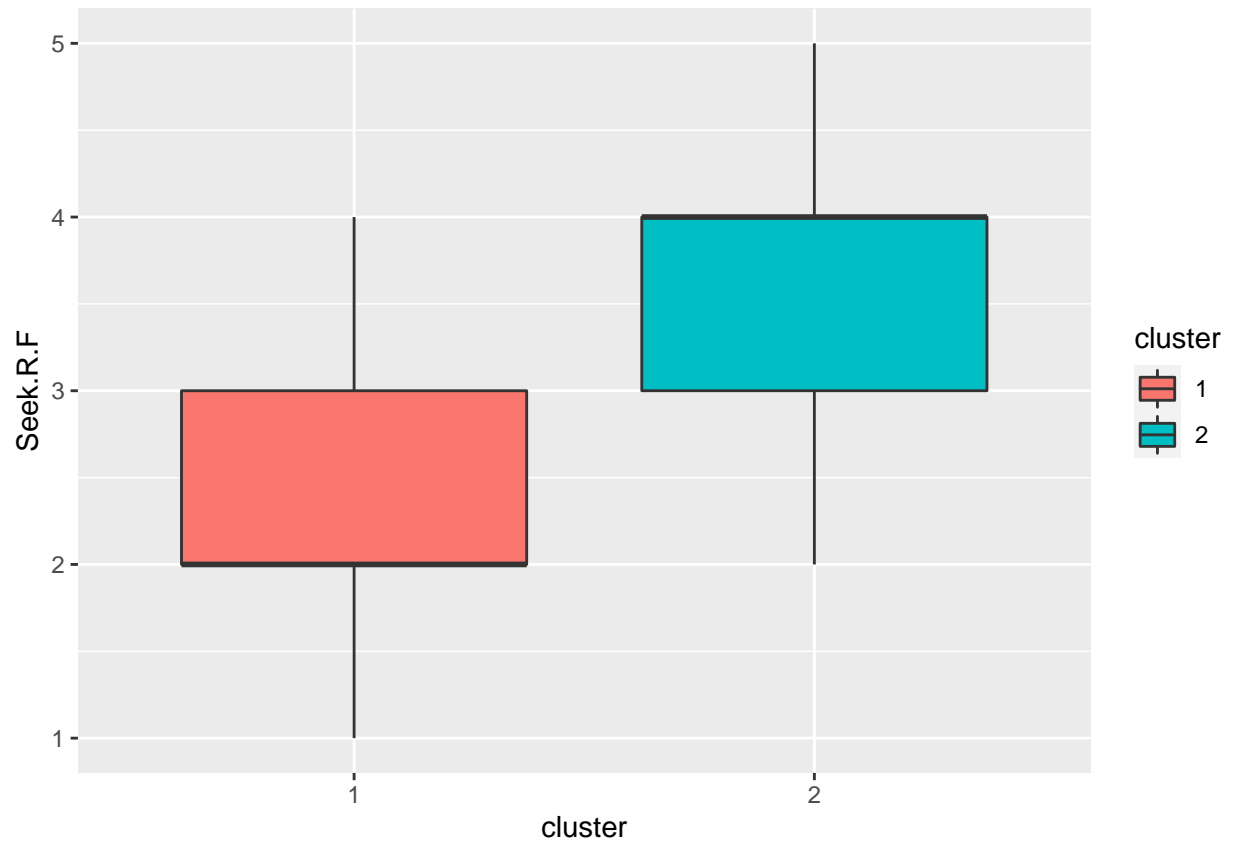
```
for (f in names(select(all.complete, ends_with(".F")))){
  p<-ggplot(all.complete, aes(x=cluster, fill=cluster))+
    geom_boxplot(aes_string(y=f))#+
    # facet_grid(~Dietary.Restriction)
  print(p)
}
```

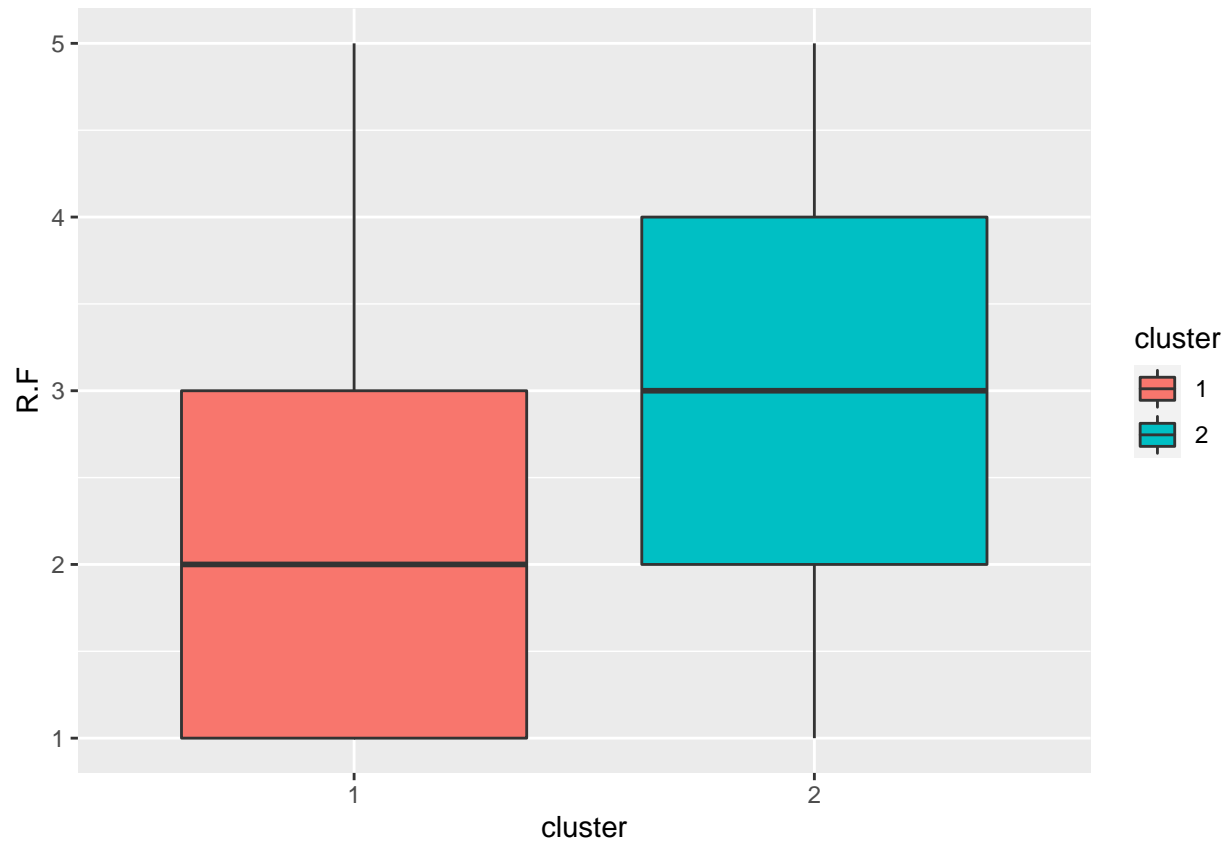


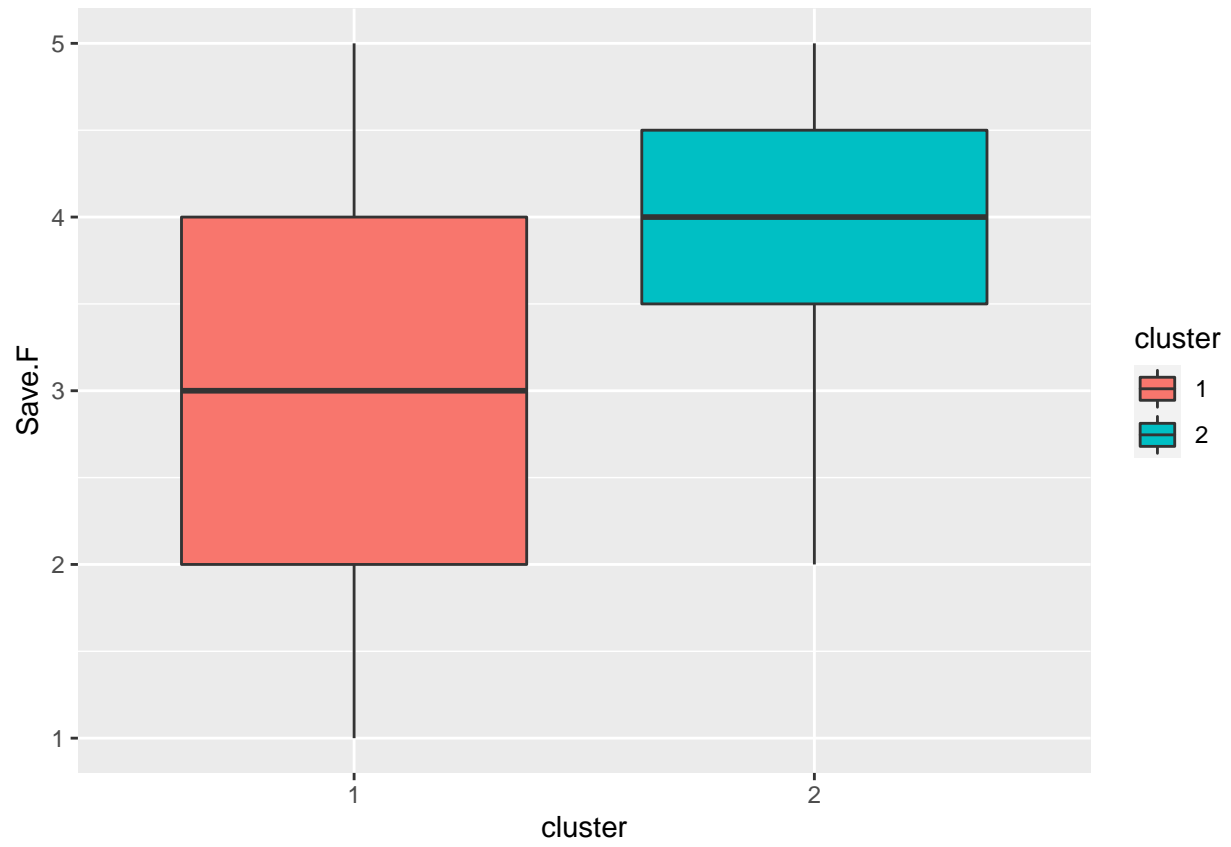


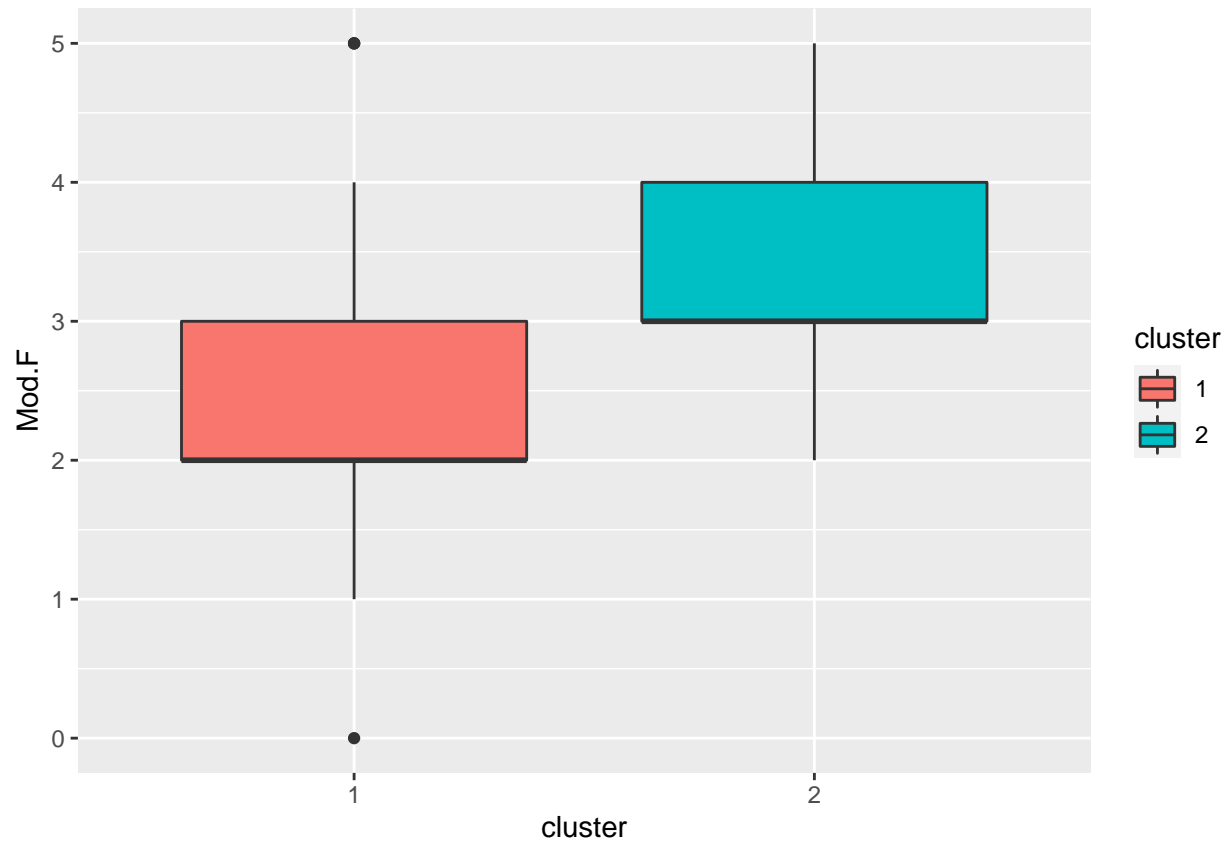


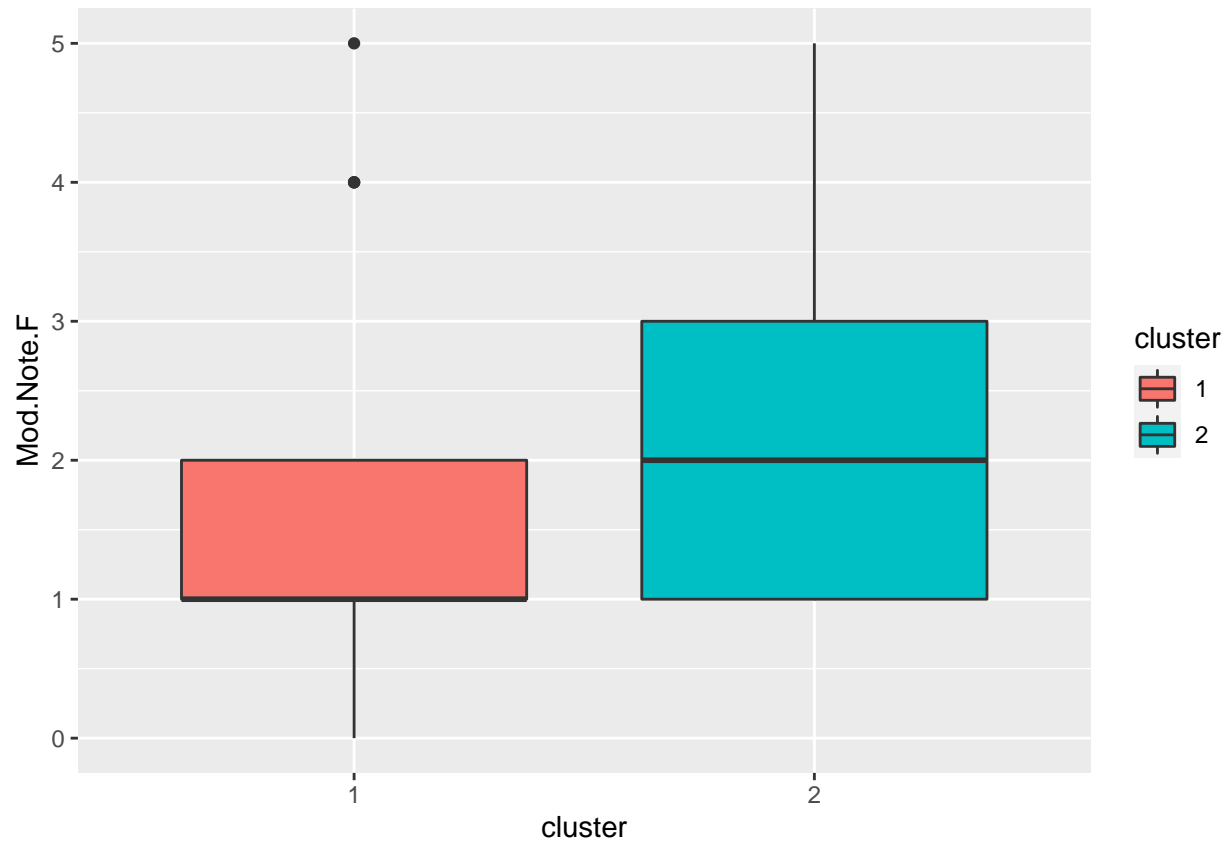


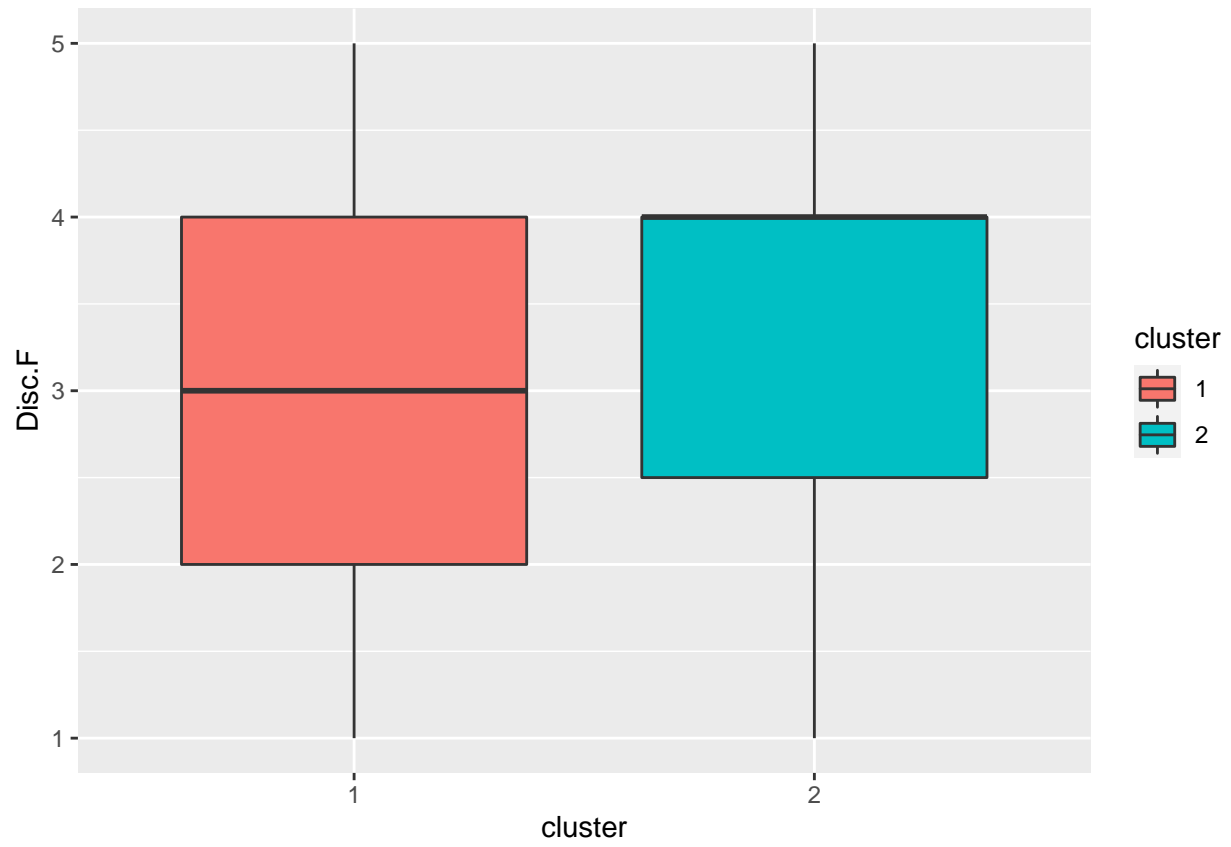


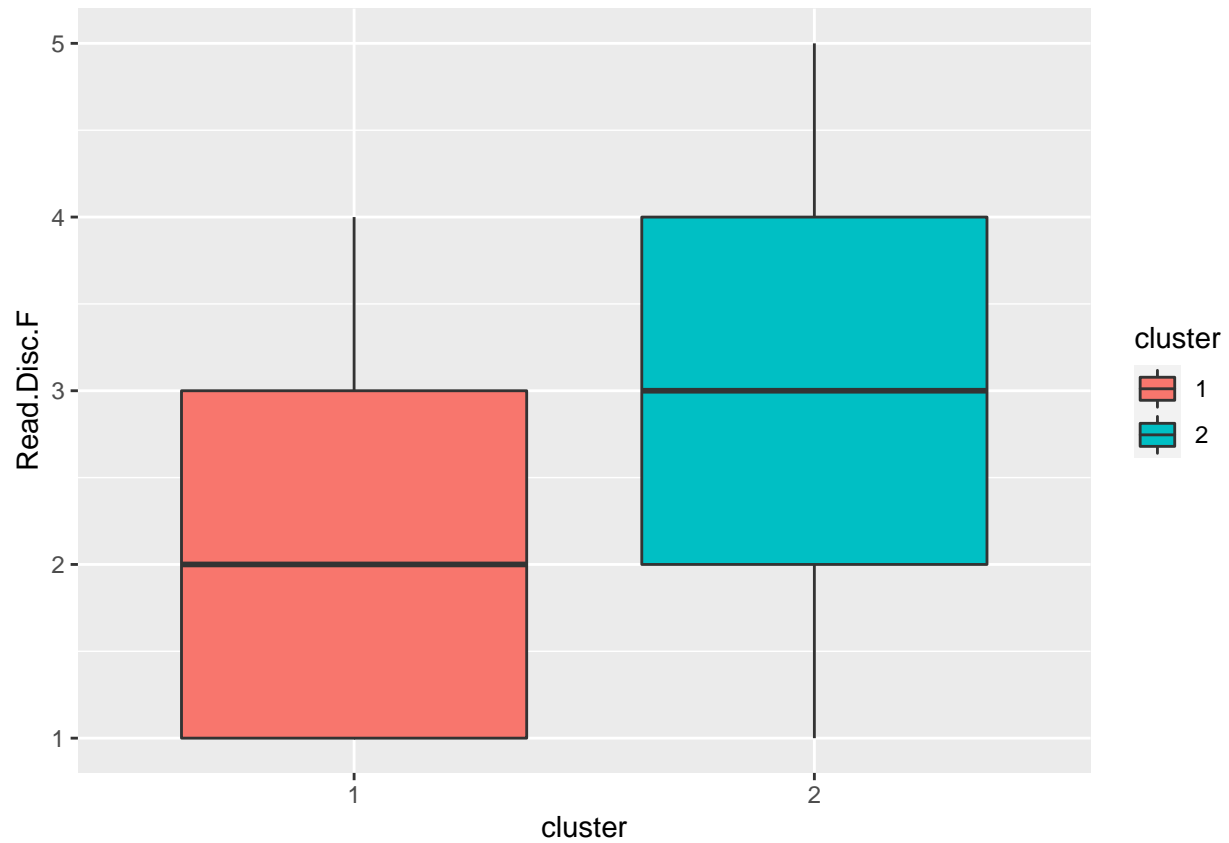






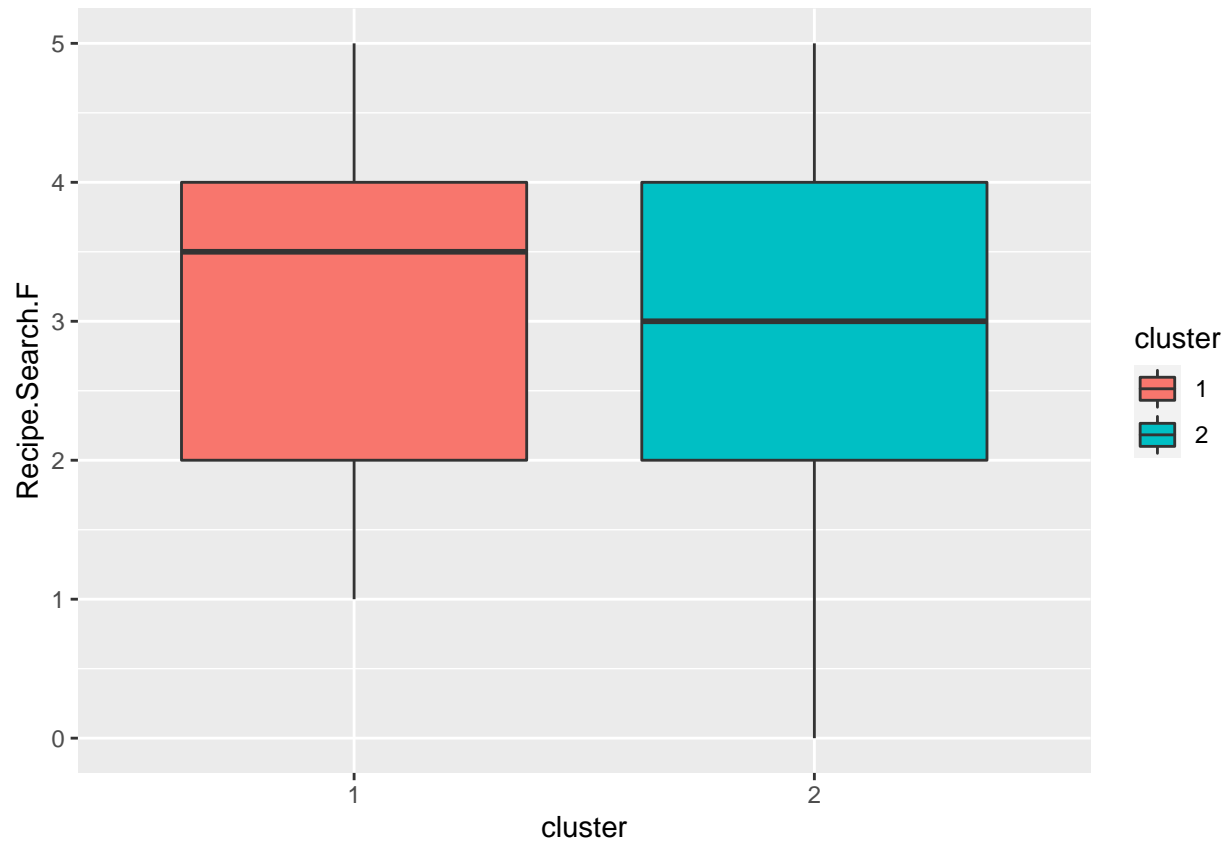


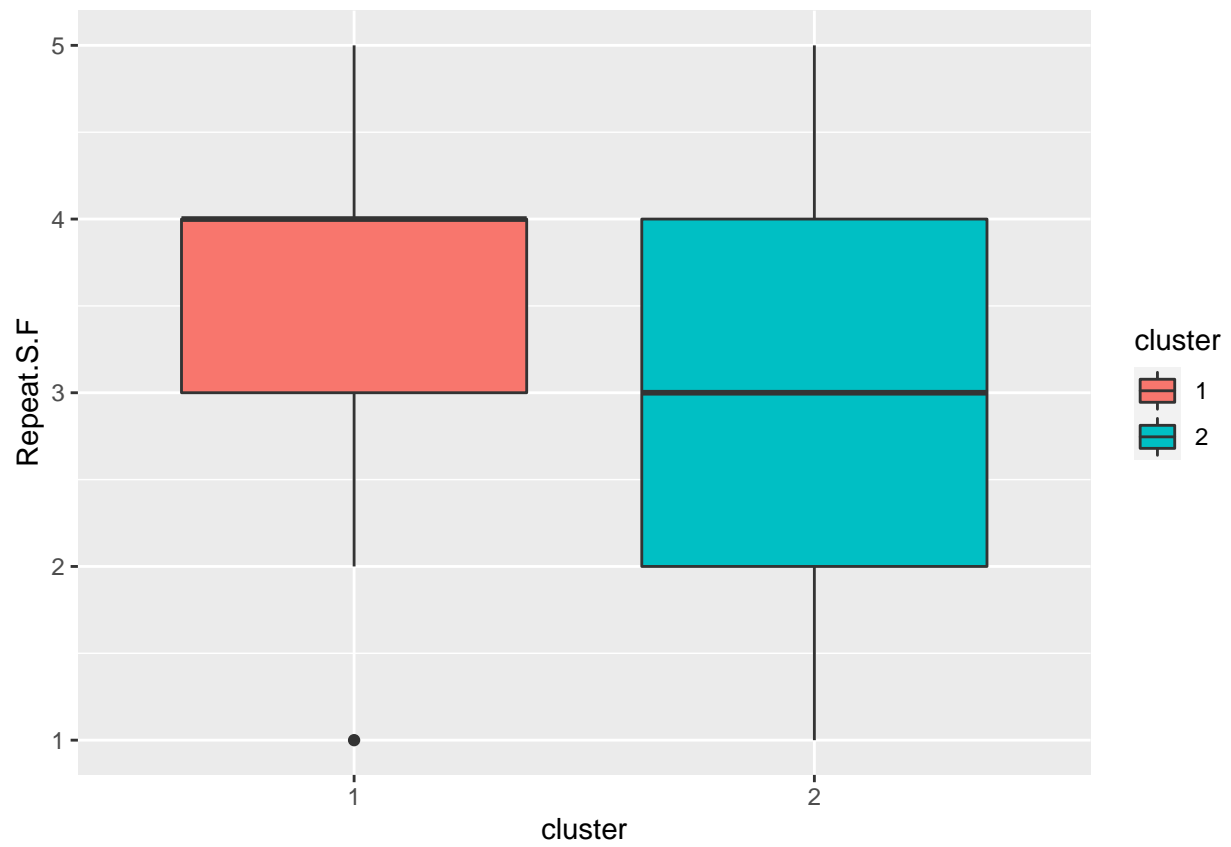


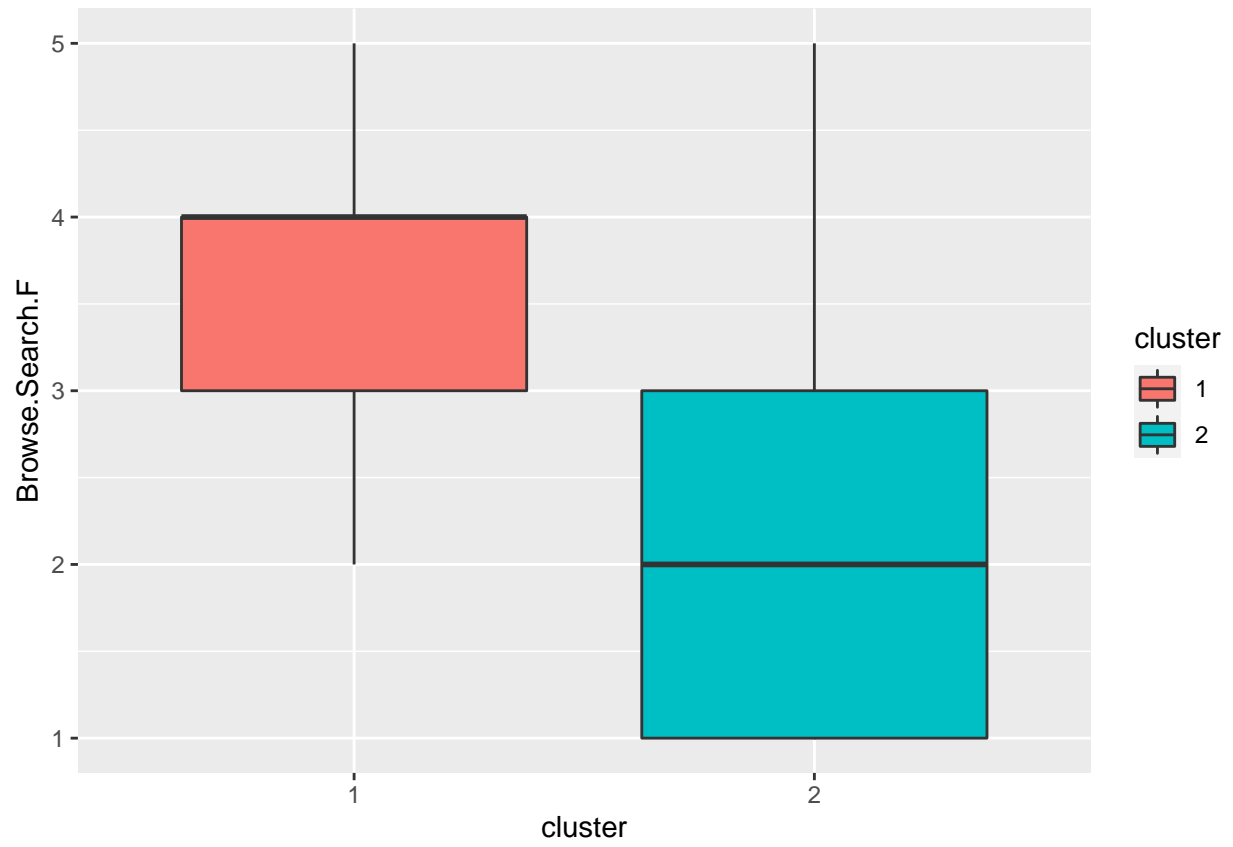


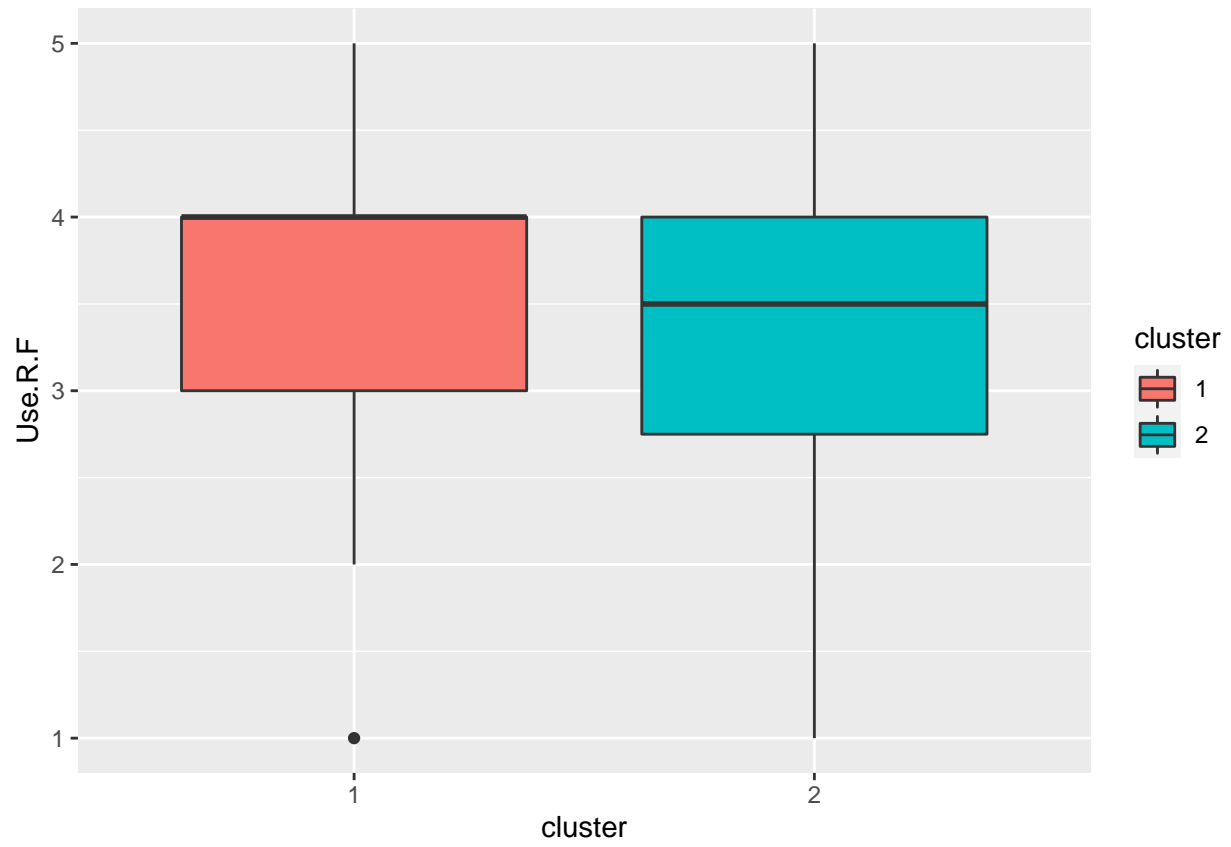
Cluster 2 contains respondents who on average more frequently interact with recipes.

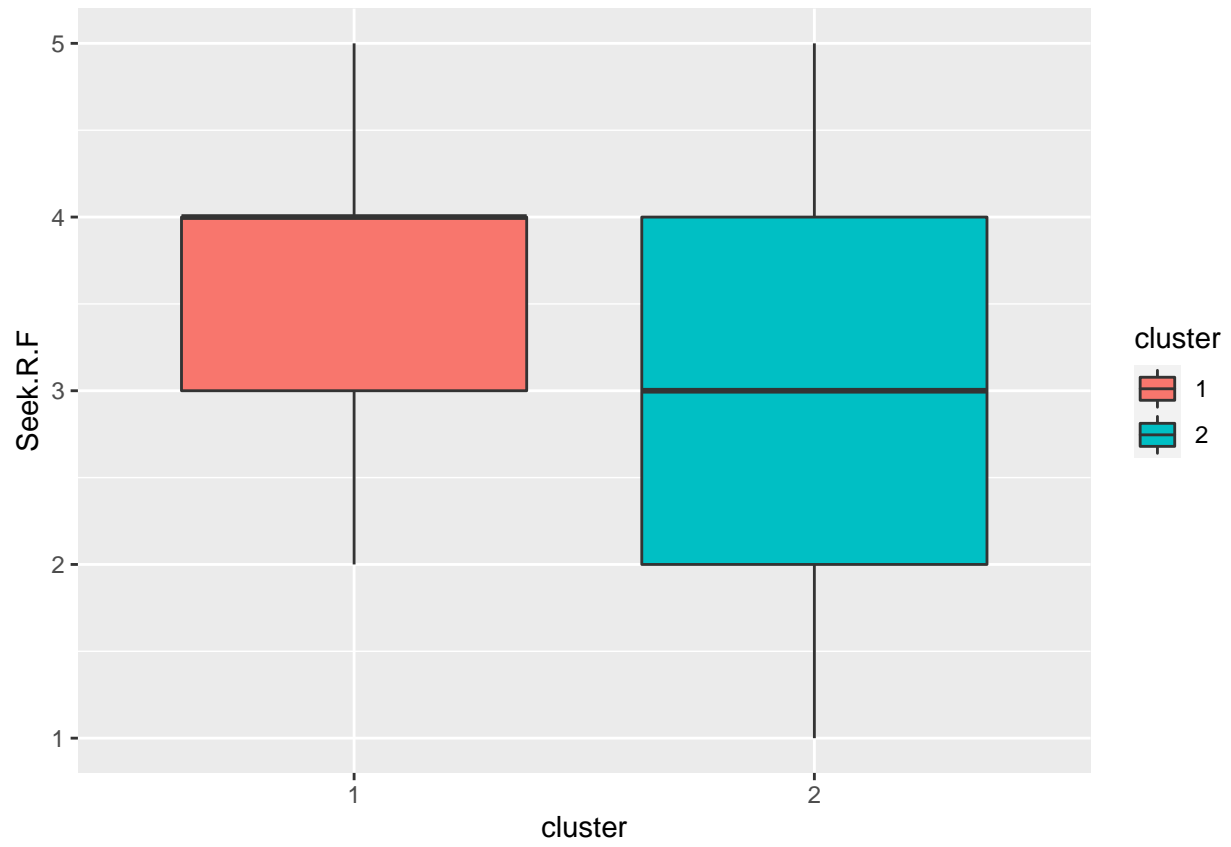
```
for (f in names(select(all.gower, ends_with(".F")))){  
  p<-ggplot(all.gower, aes(x=cluster, fill=cluster))+  
    geom_boxplot(aes_string(y=f))#+  
    # facet_grid(~Dietary.Restriction)  
  print(p)  
}
```

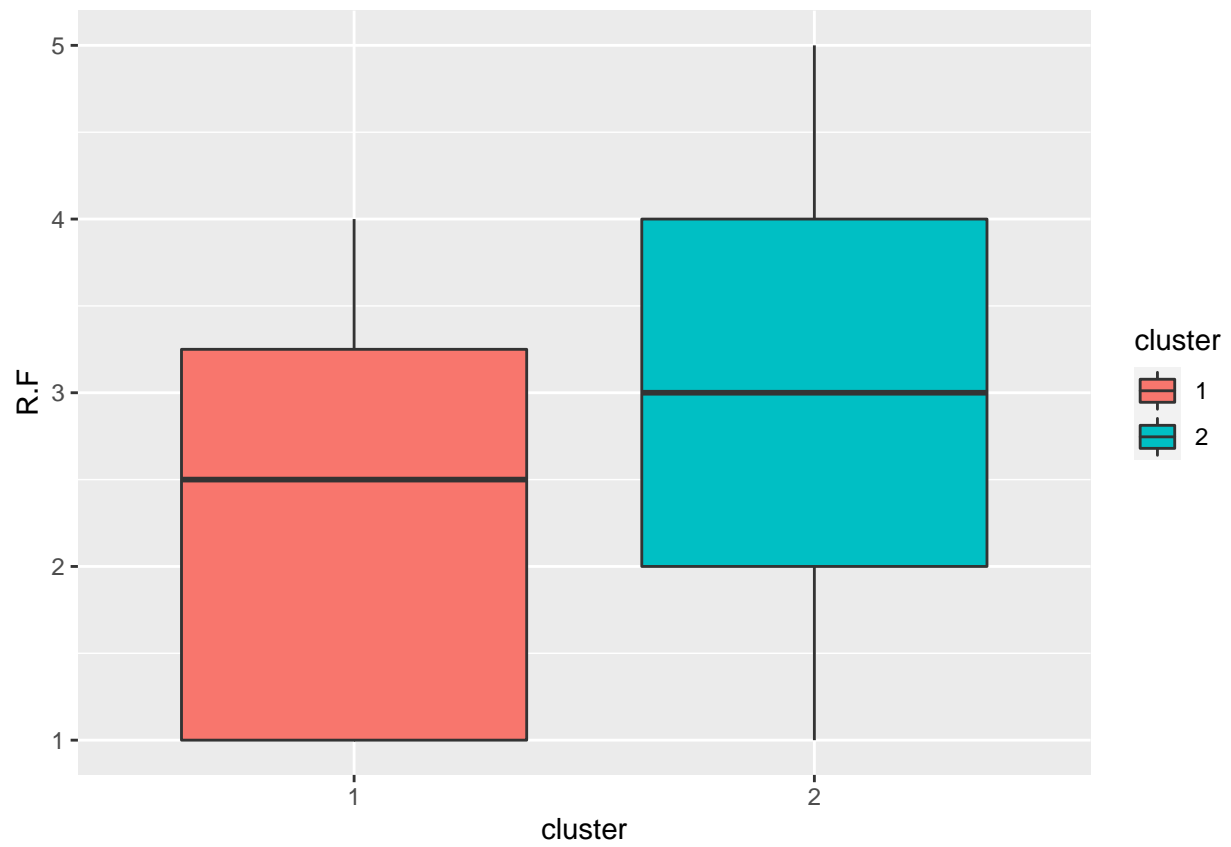



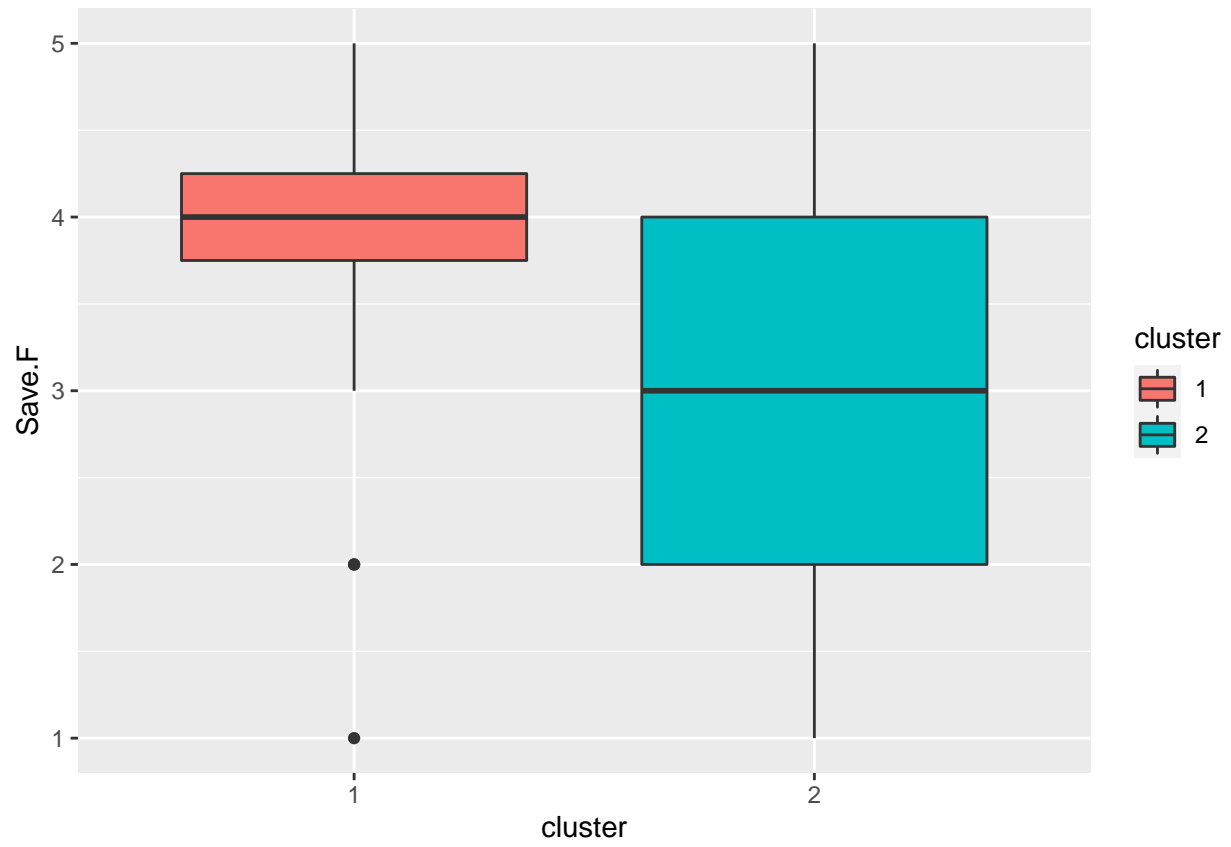


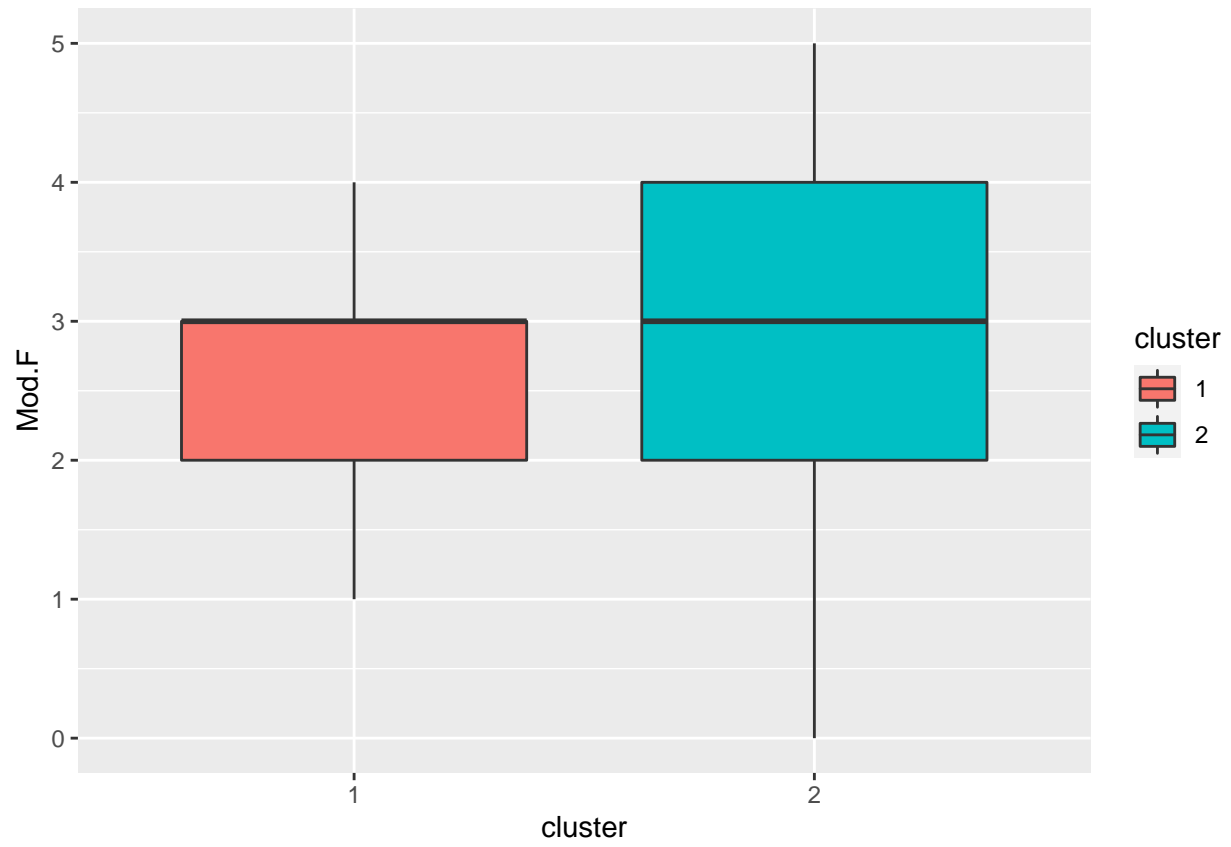


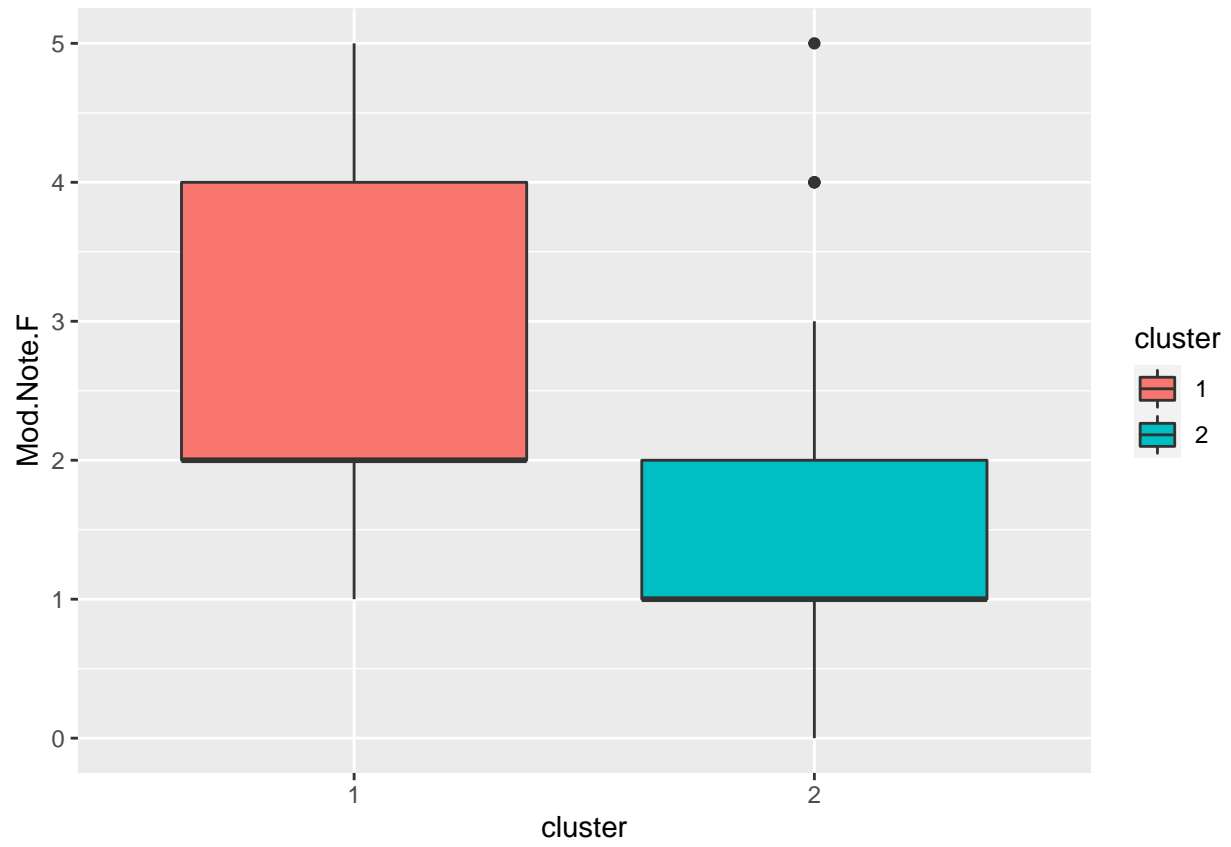


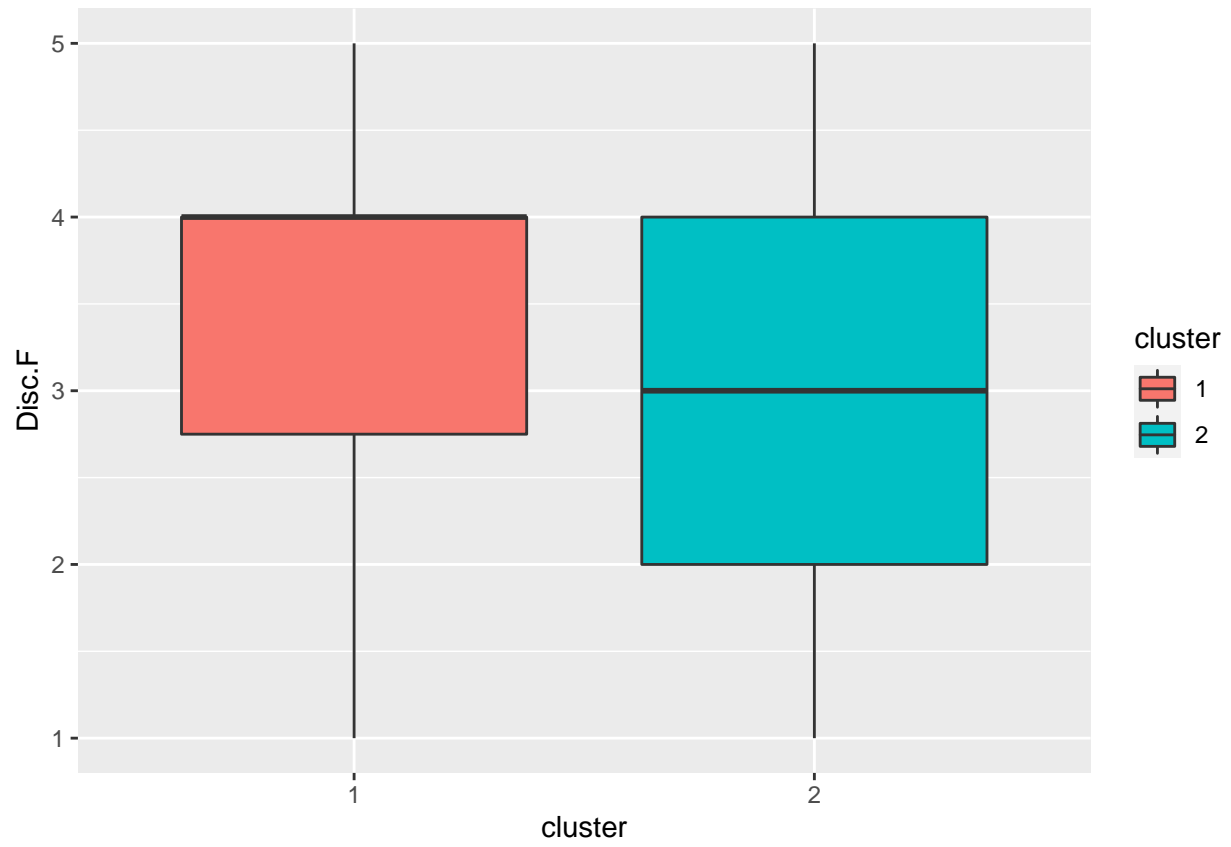


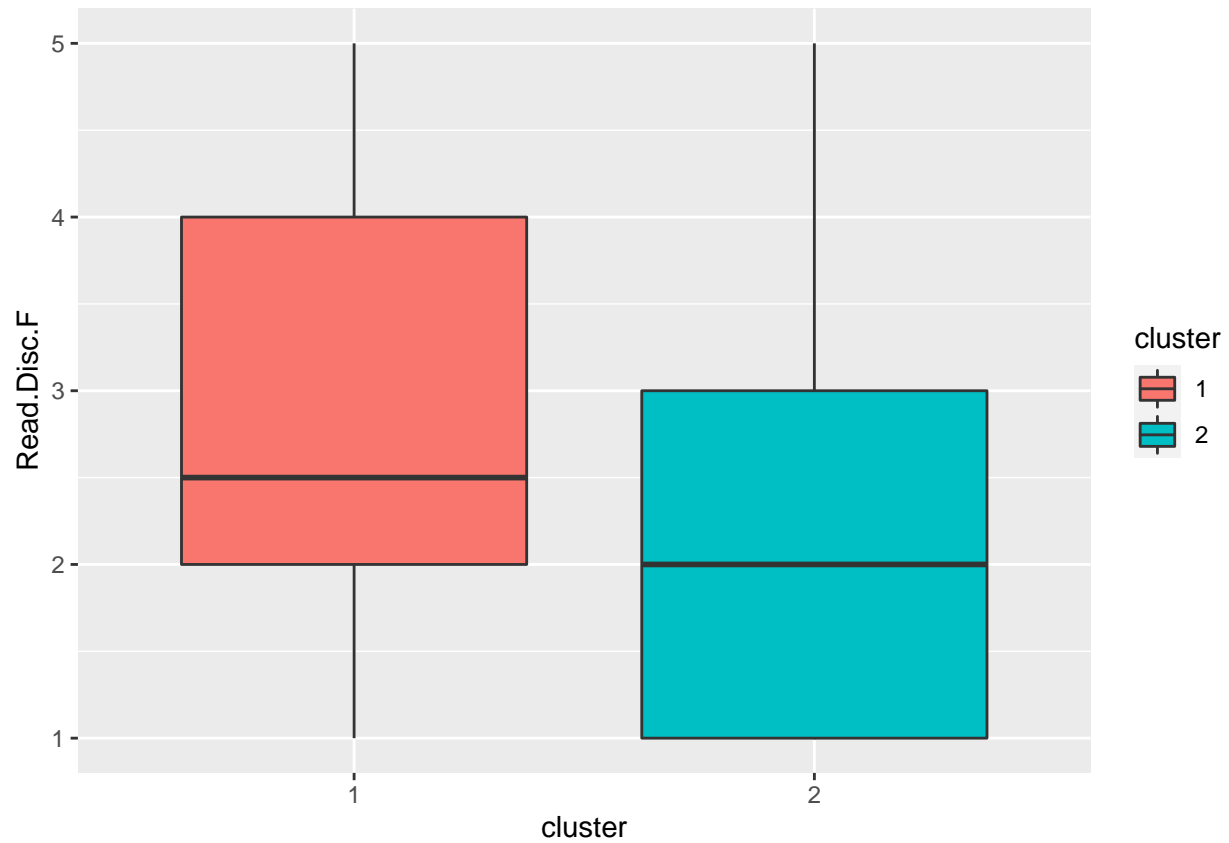






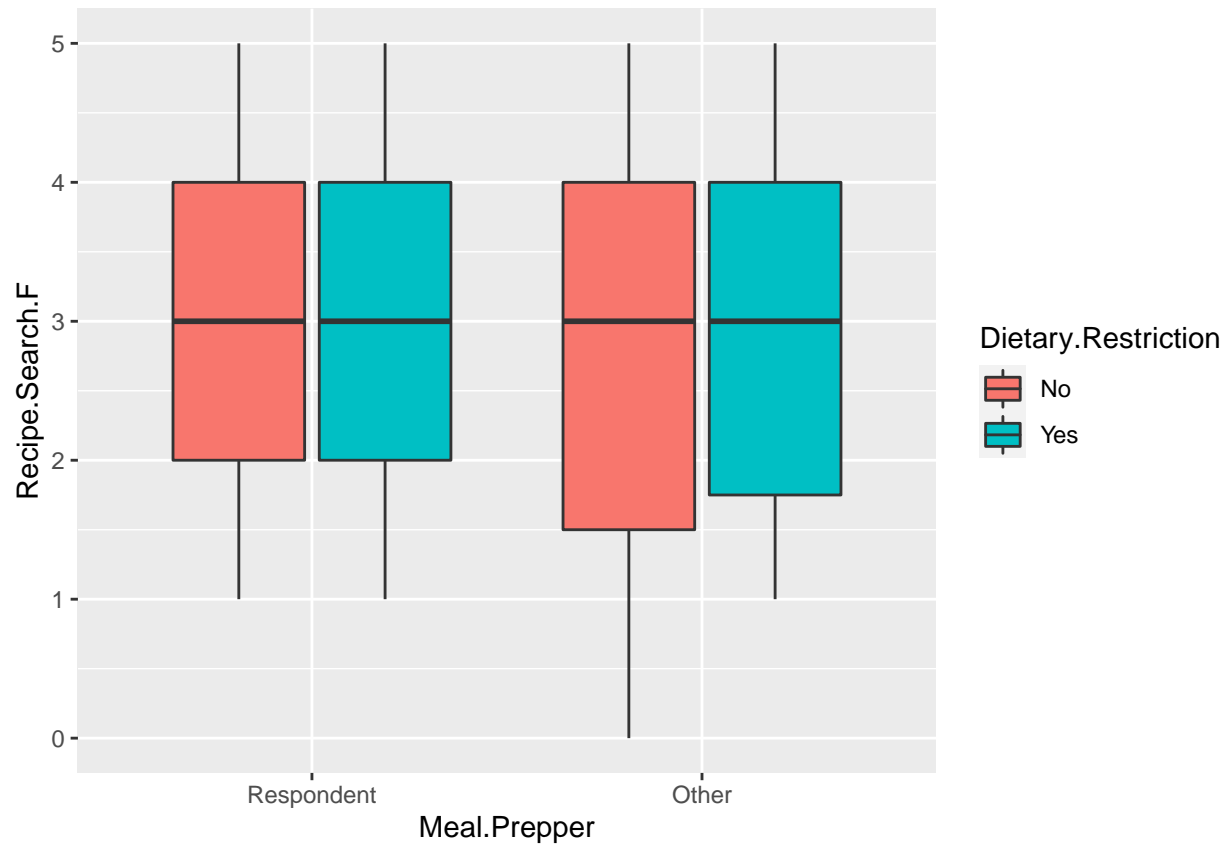


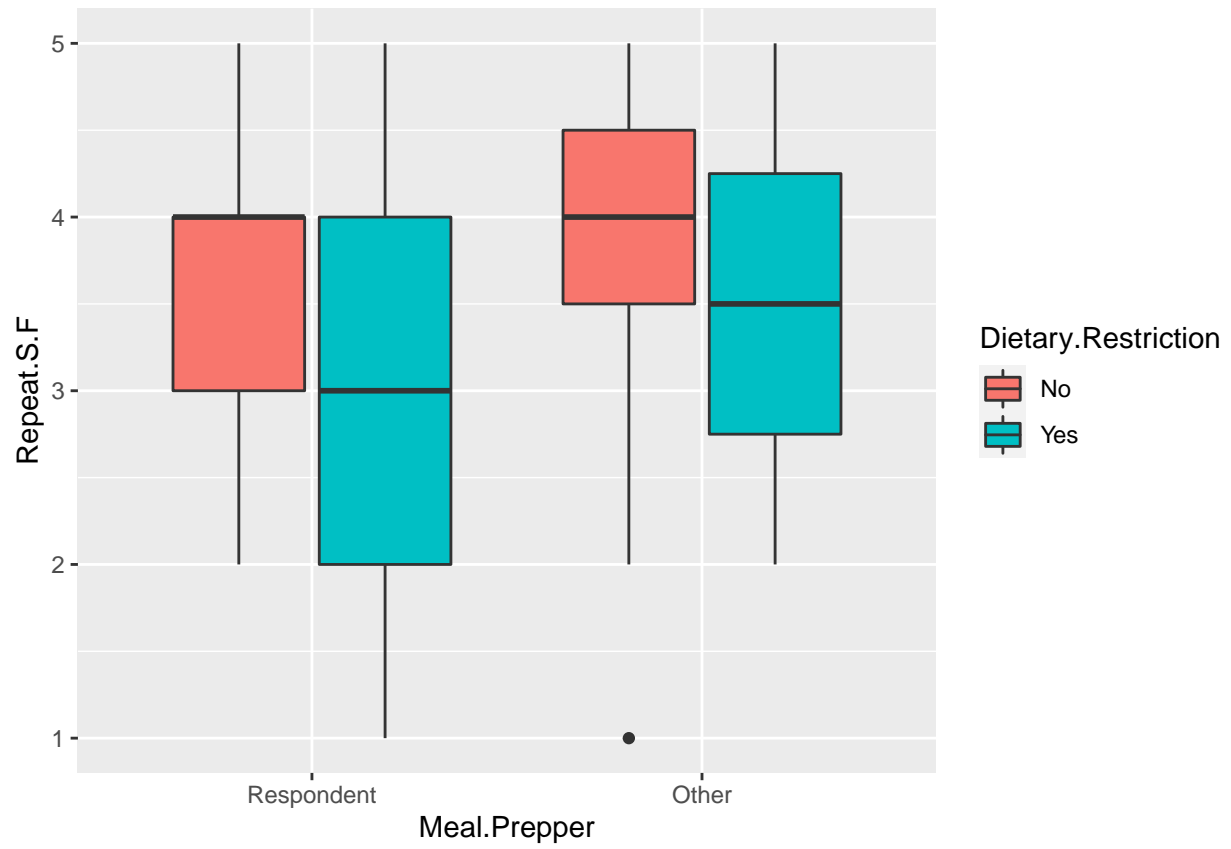


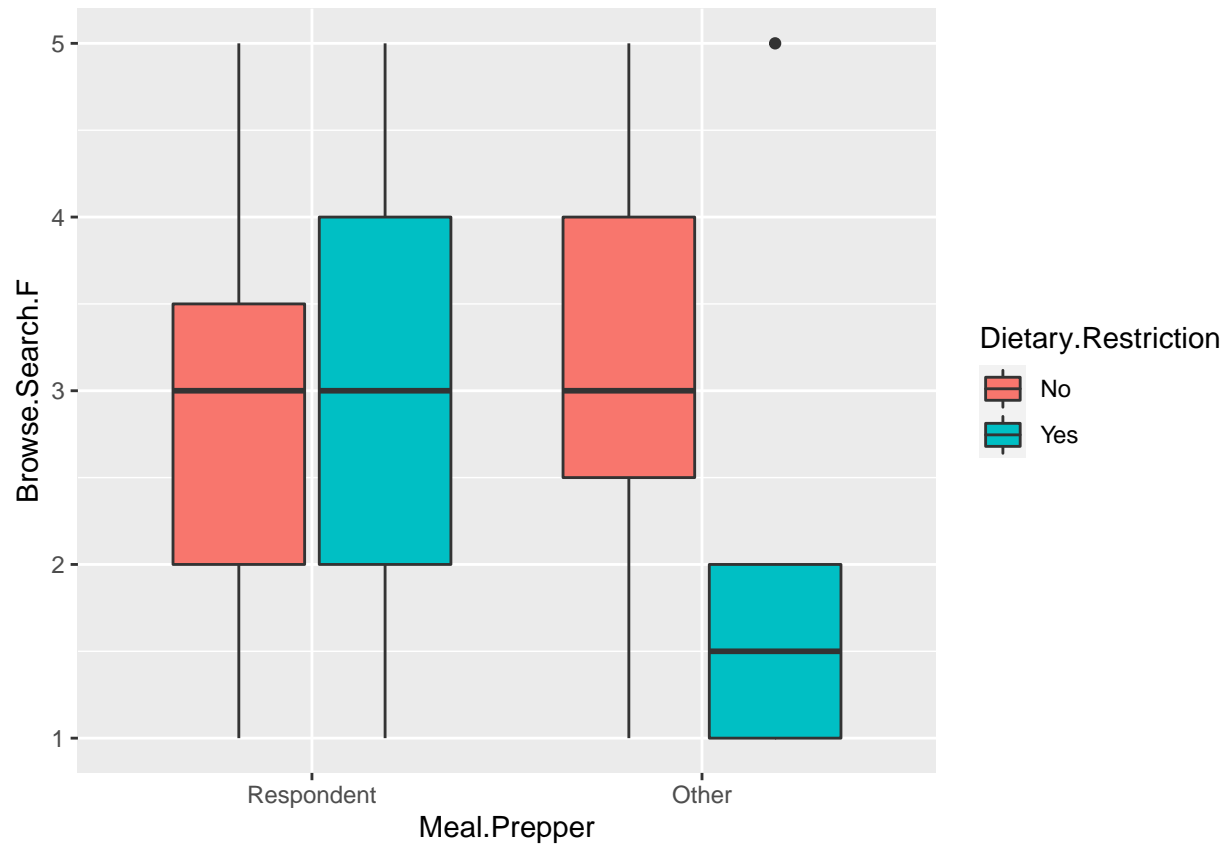


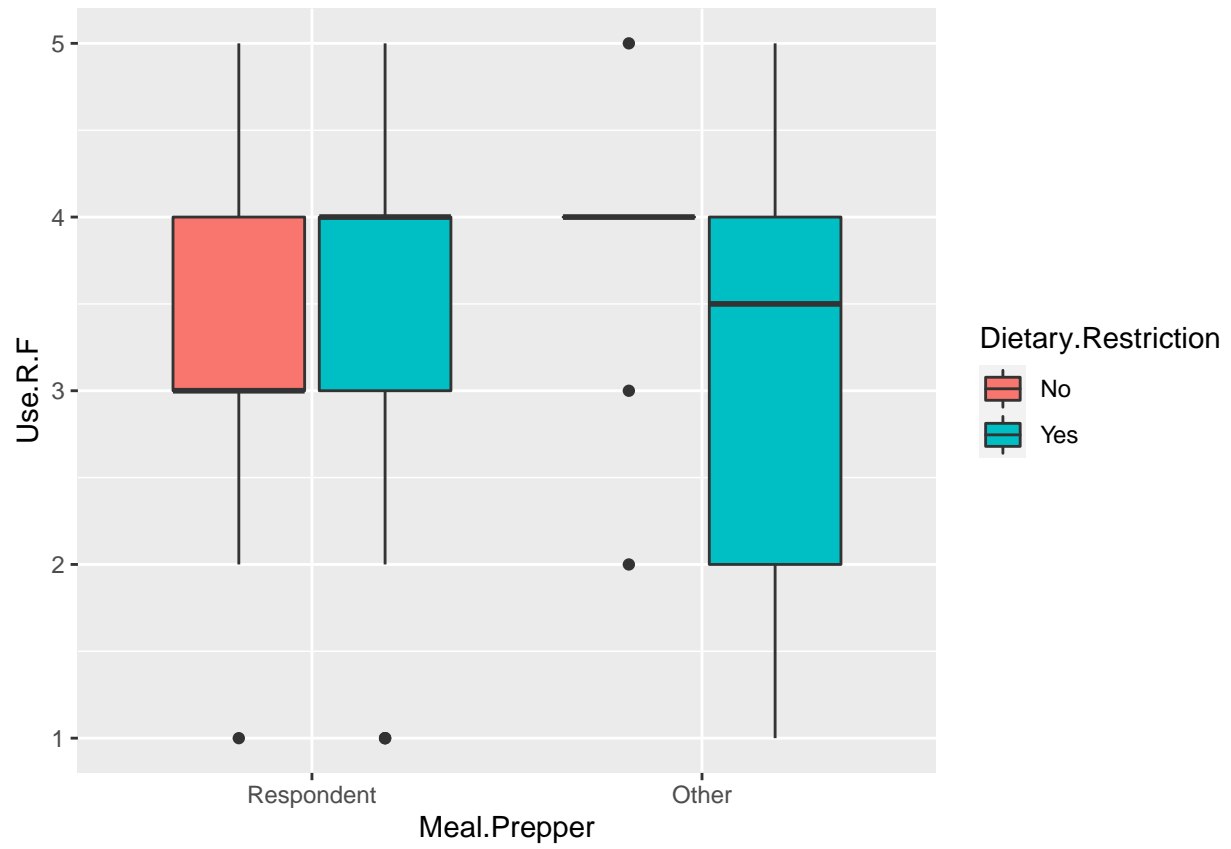
The Gower distance does not divide the users by recipe interaction frequency.

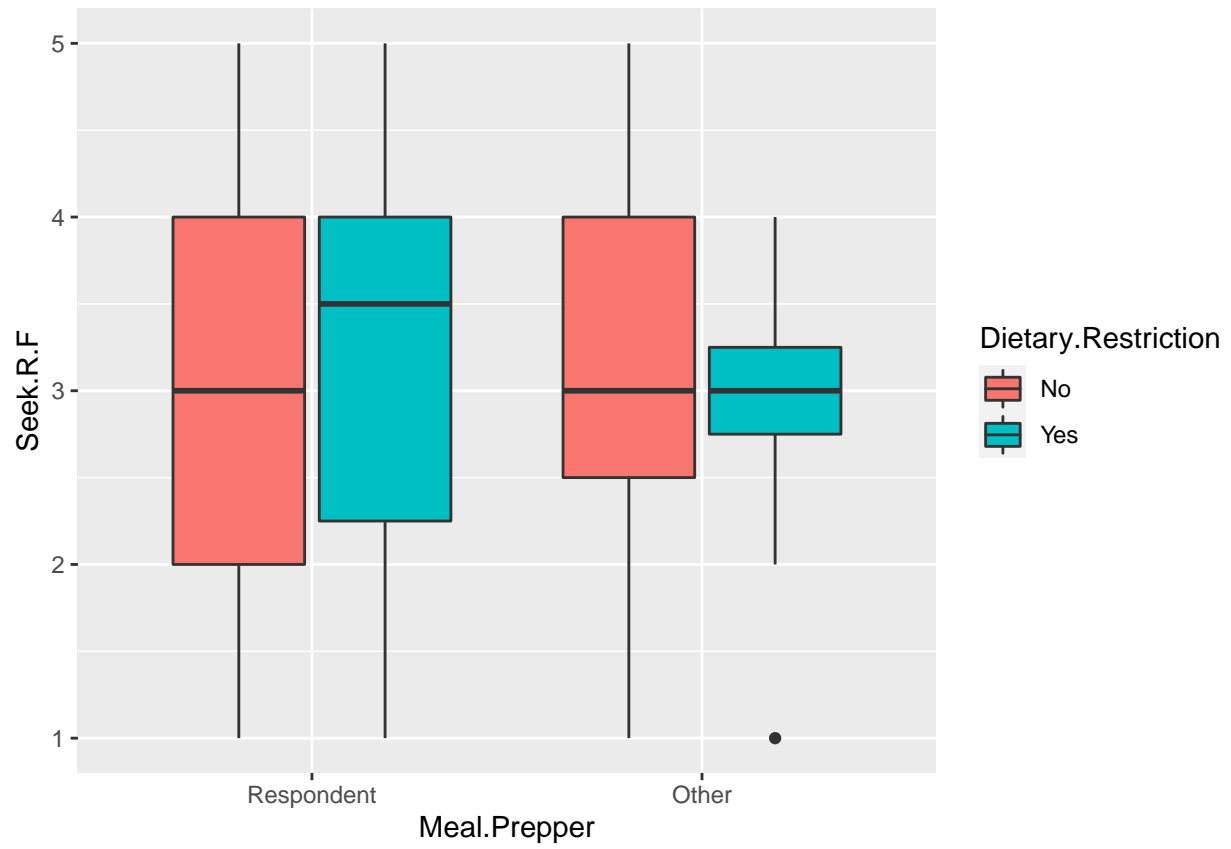
```
for (f in names(select(cleaned, ends_with(".F")))){  
  p<-ggplot(cleaned, aes(x=Meal.Prepper, fill=Dietary.Restriction))+  
    geom_boxplot(aes_string(y=f))#+  
    # facet_grid(~Dietary.Restriction)  
  print(p)  
}
```

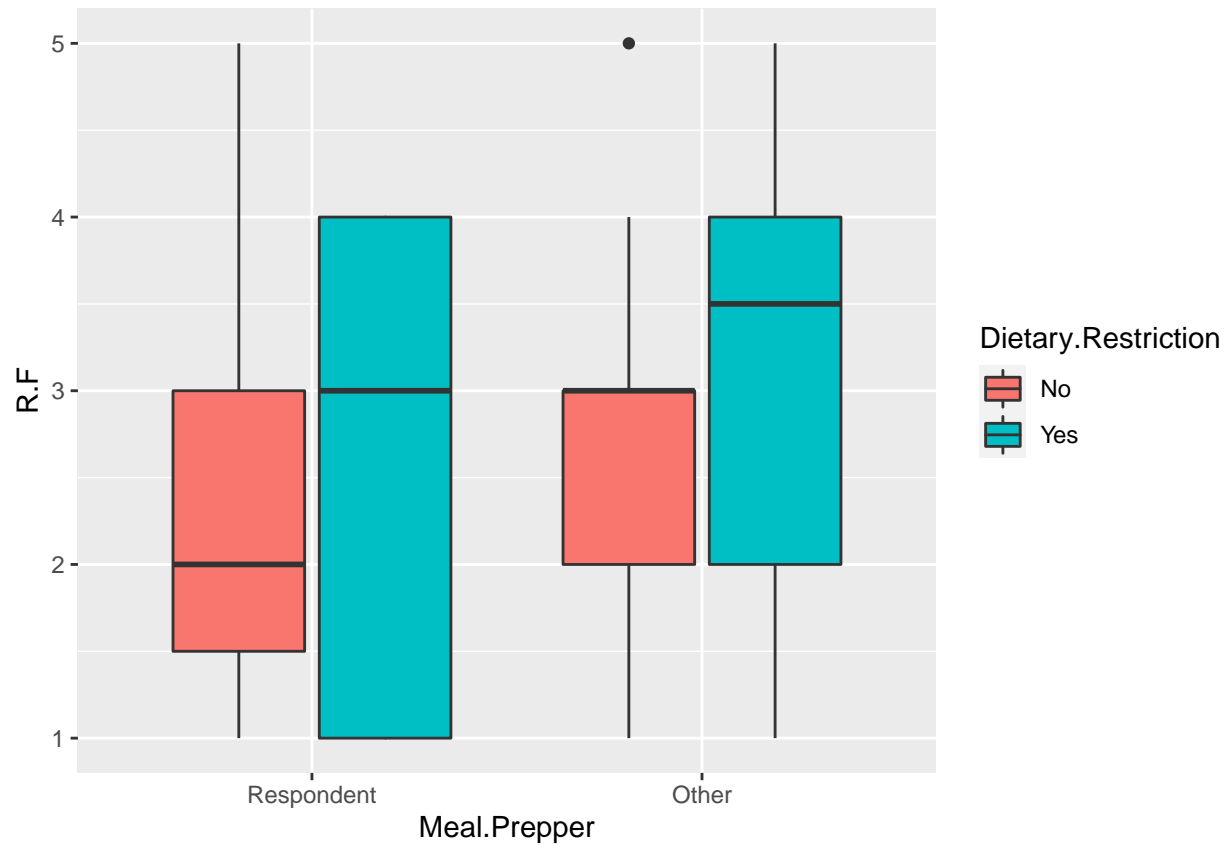


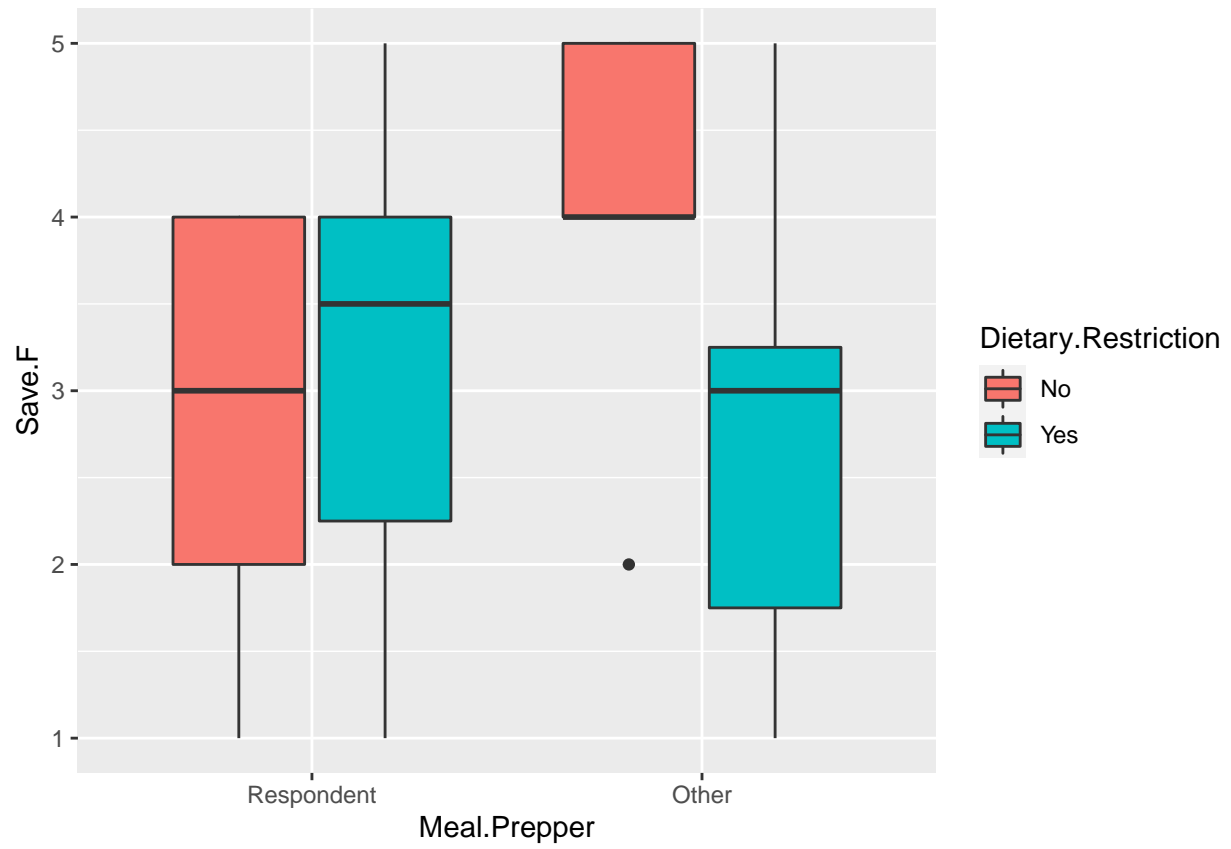


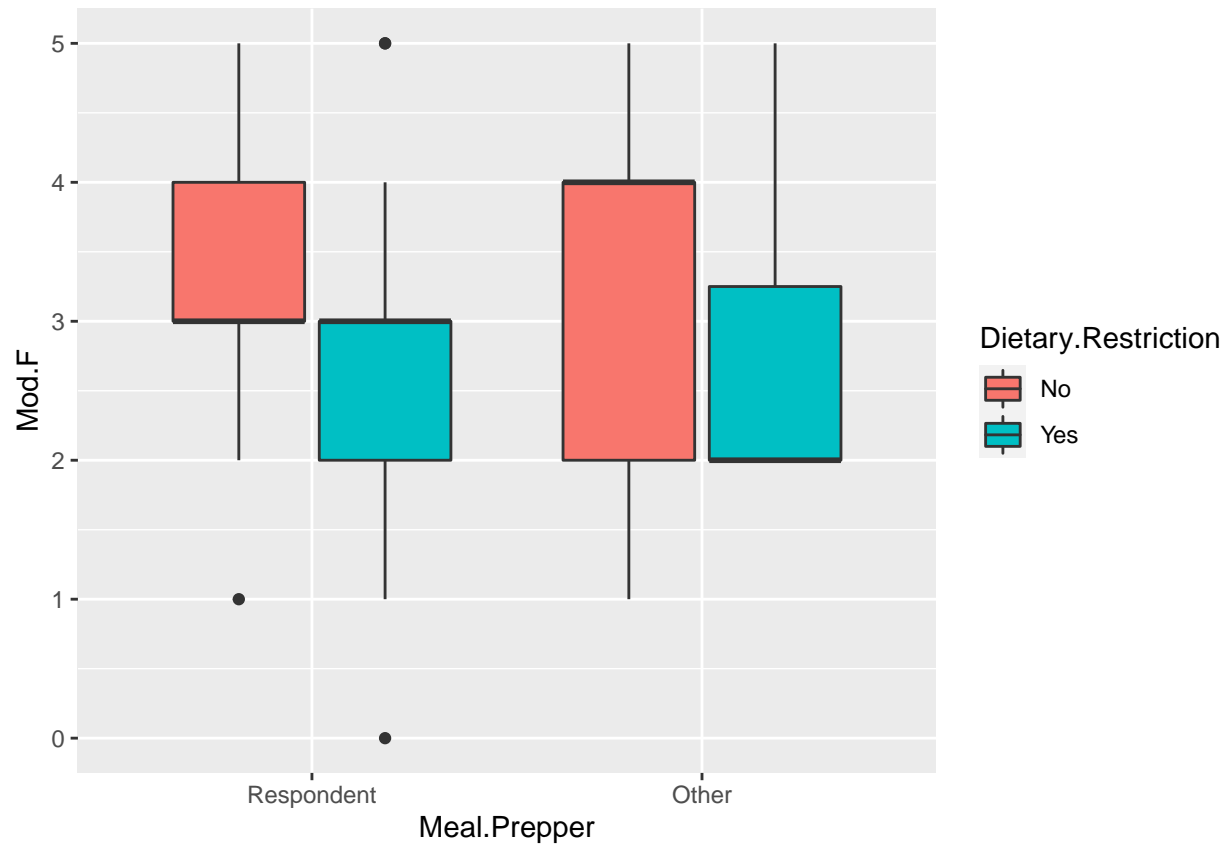


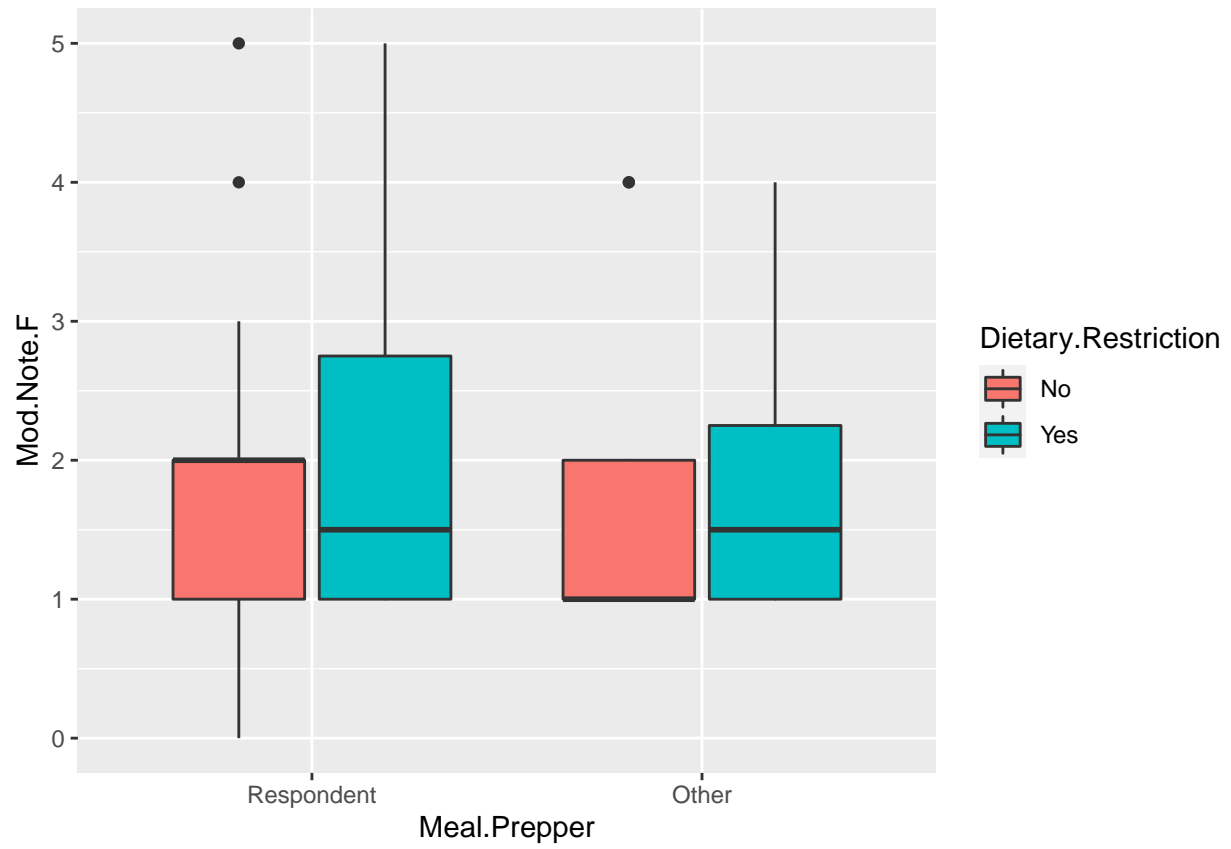


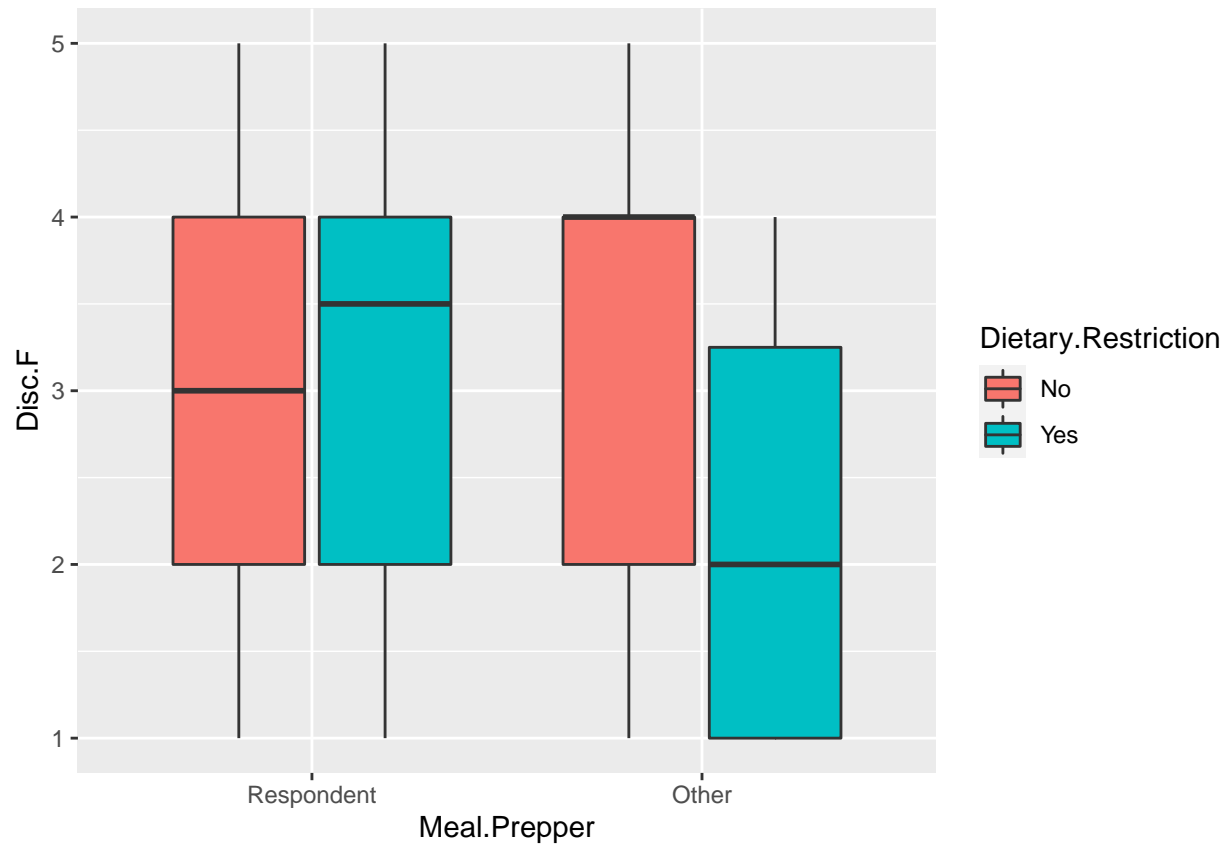


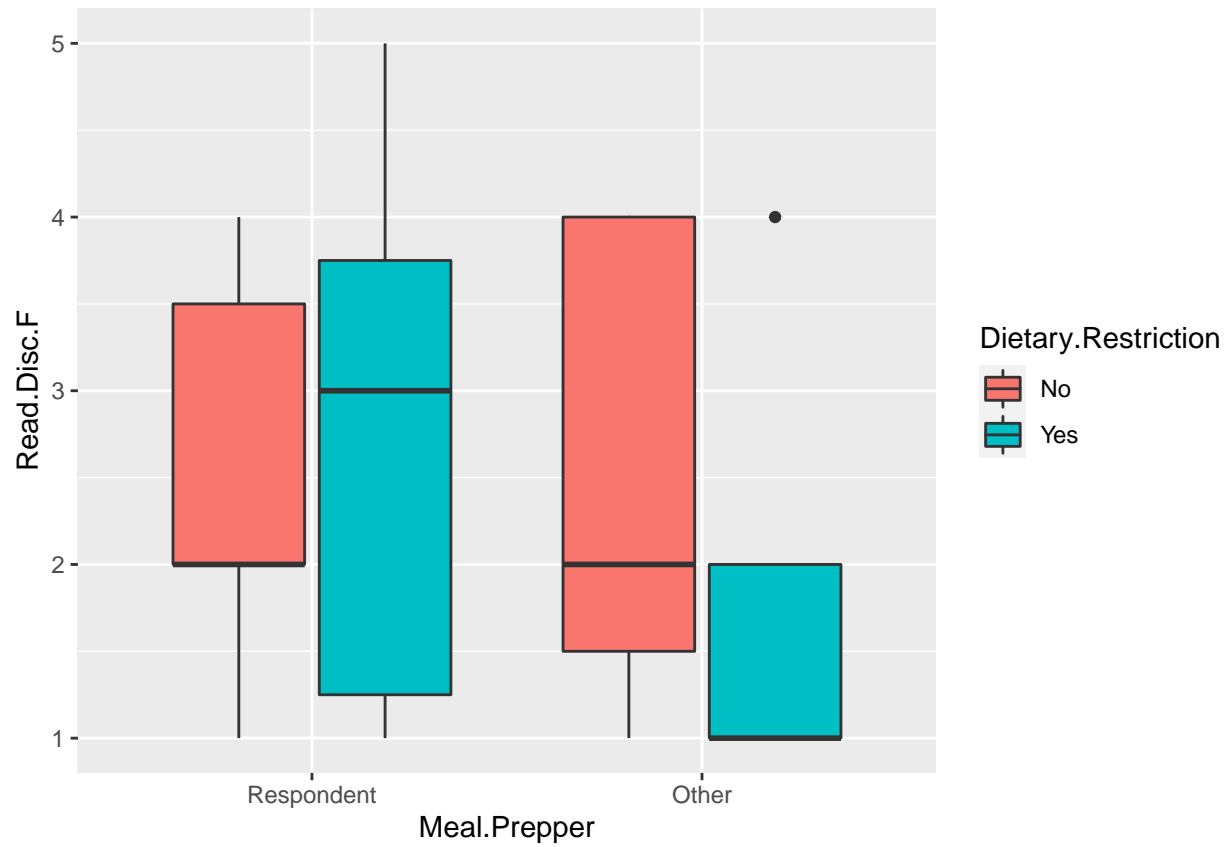




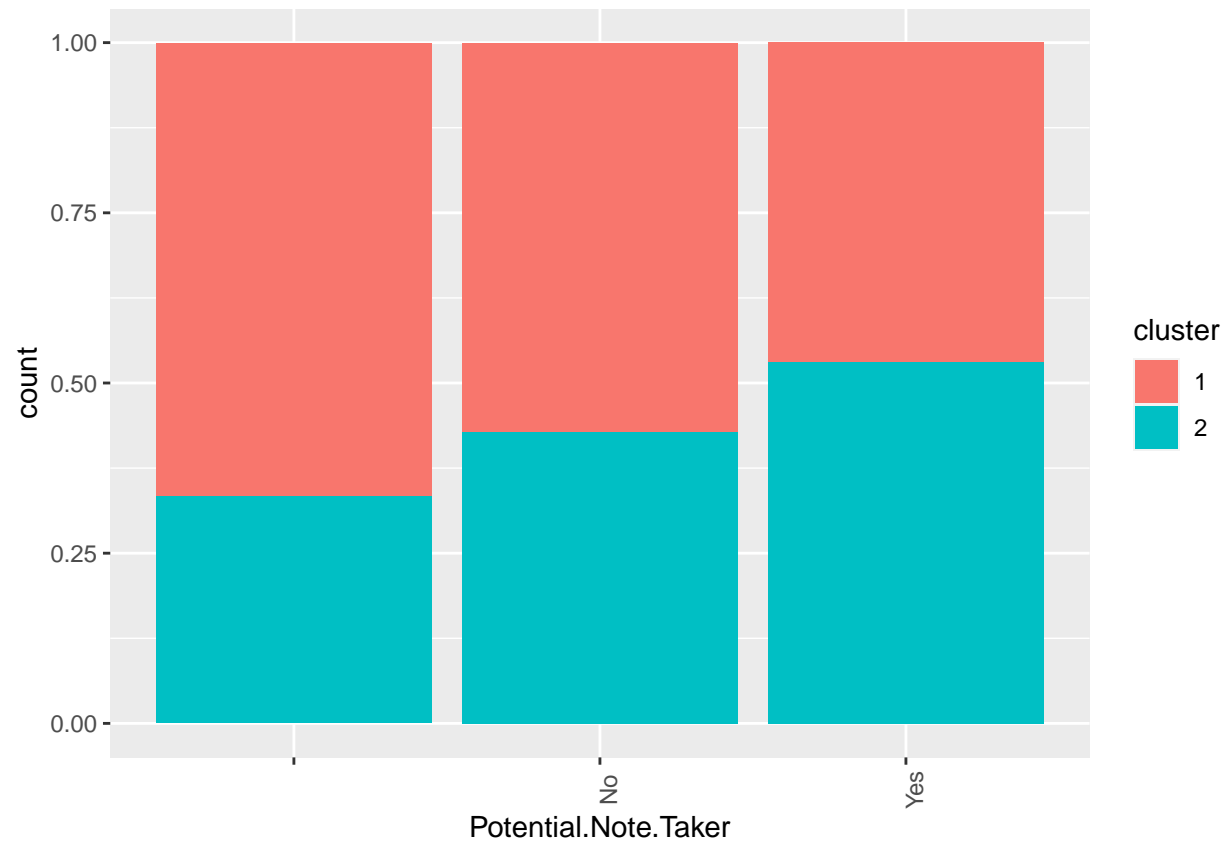






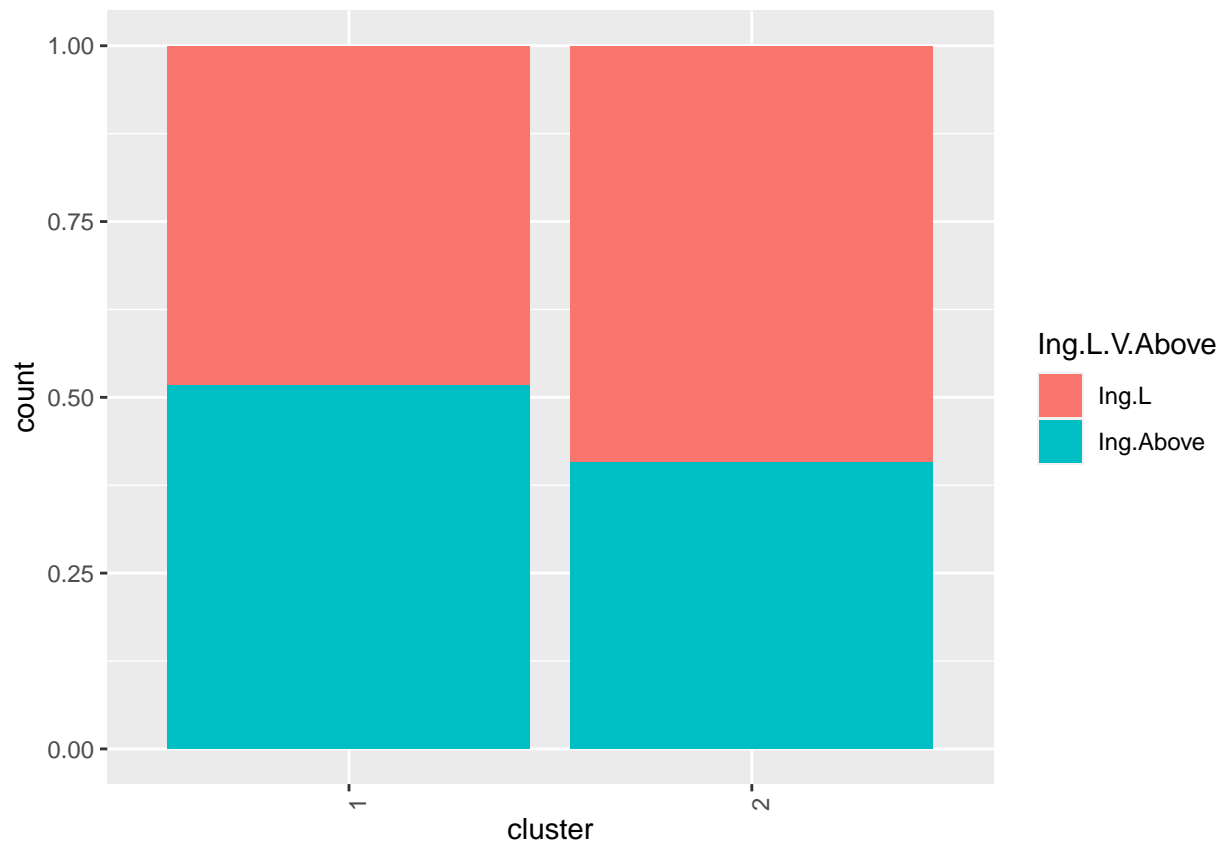


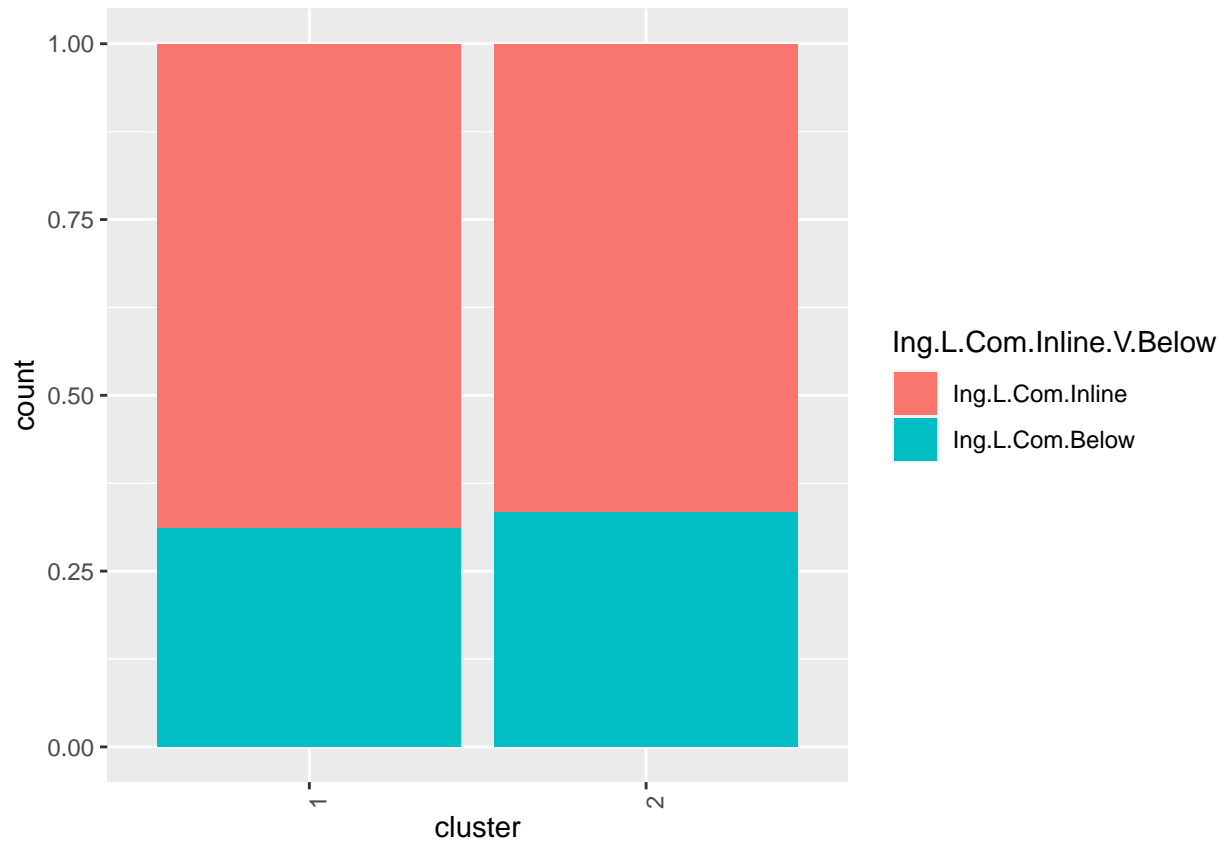
```
ggplot(all.complete,aes( x=Potential.Note.Taker))+
  geom_bar(aes(fill=cluster), position = "fill")+
  theme(axis.text.x = element_text(angle = 90))
```

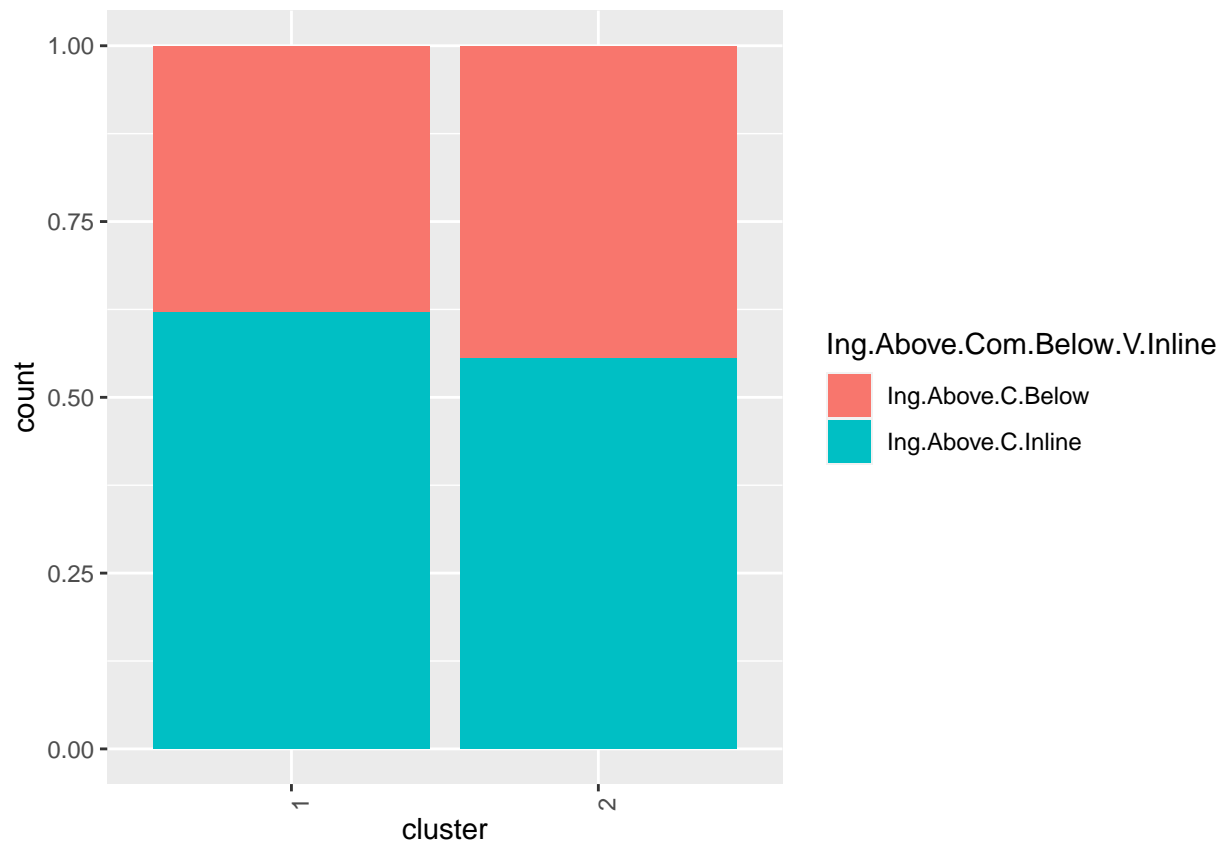


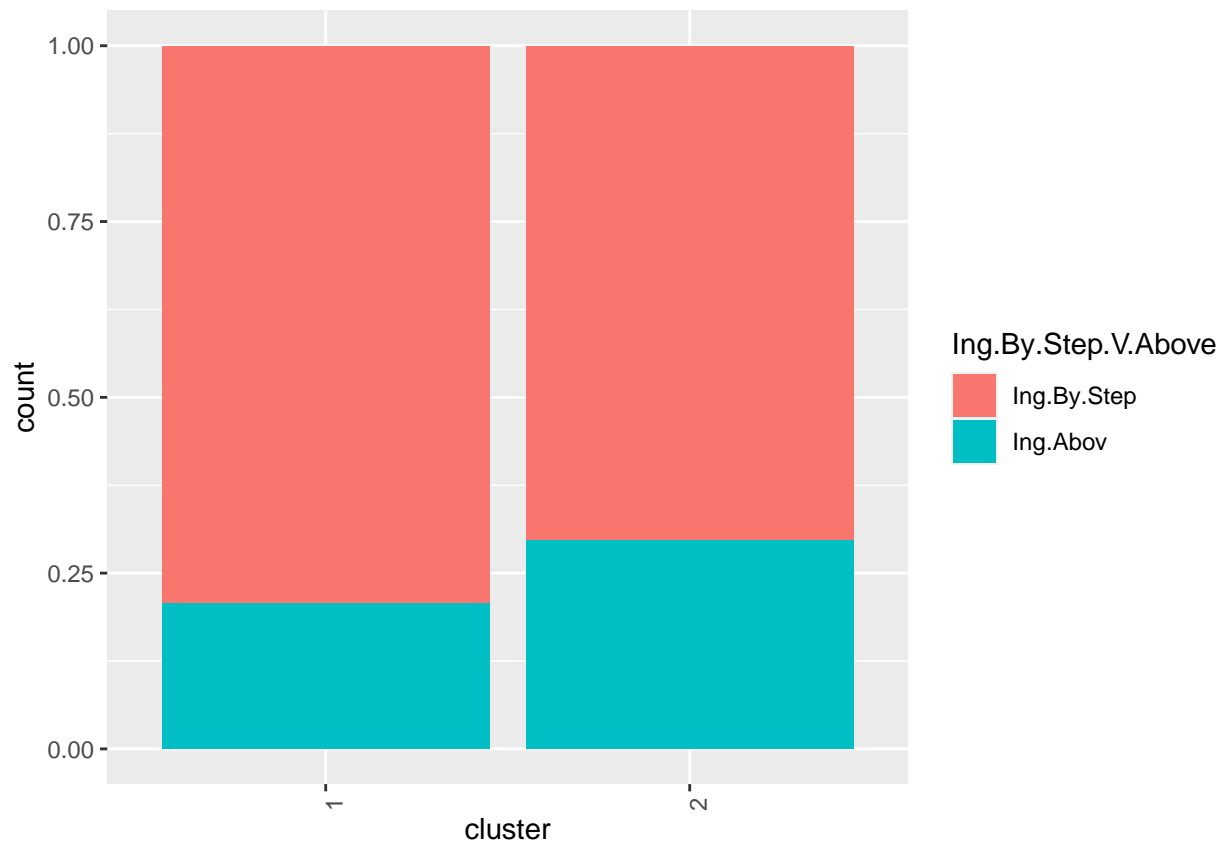
```
for (i in names(select(all.complete, starts_with("Ing.")))){
  p<-ggplot(all.complete,aes( x=cluster))+
    geom_bar(aes_string(fill=i), position = "fill")+
    # facet_grid(~Meal.Prepper)+
    theme(axis.text.x = element_text(angle = 90))

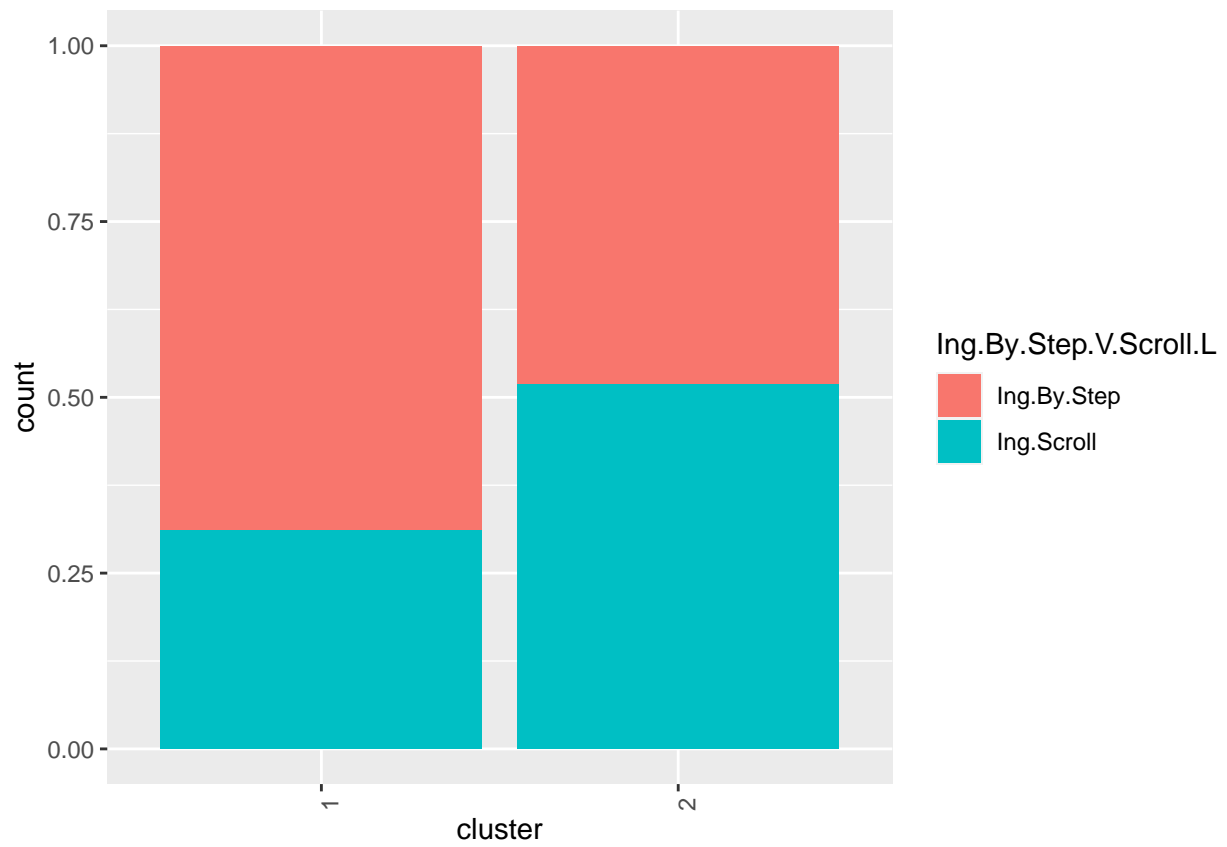
  print(p)
}
```

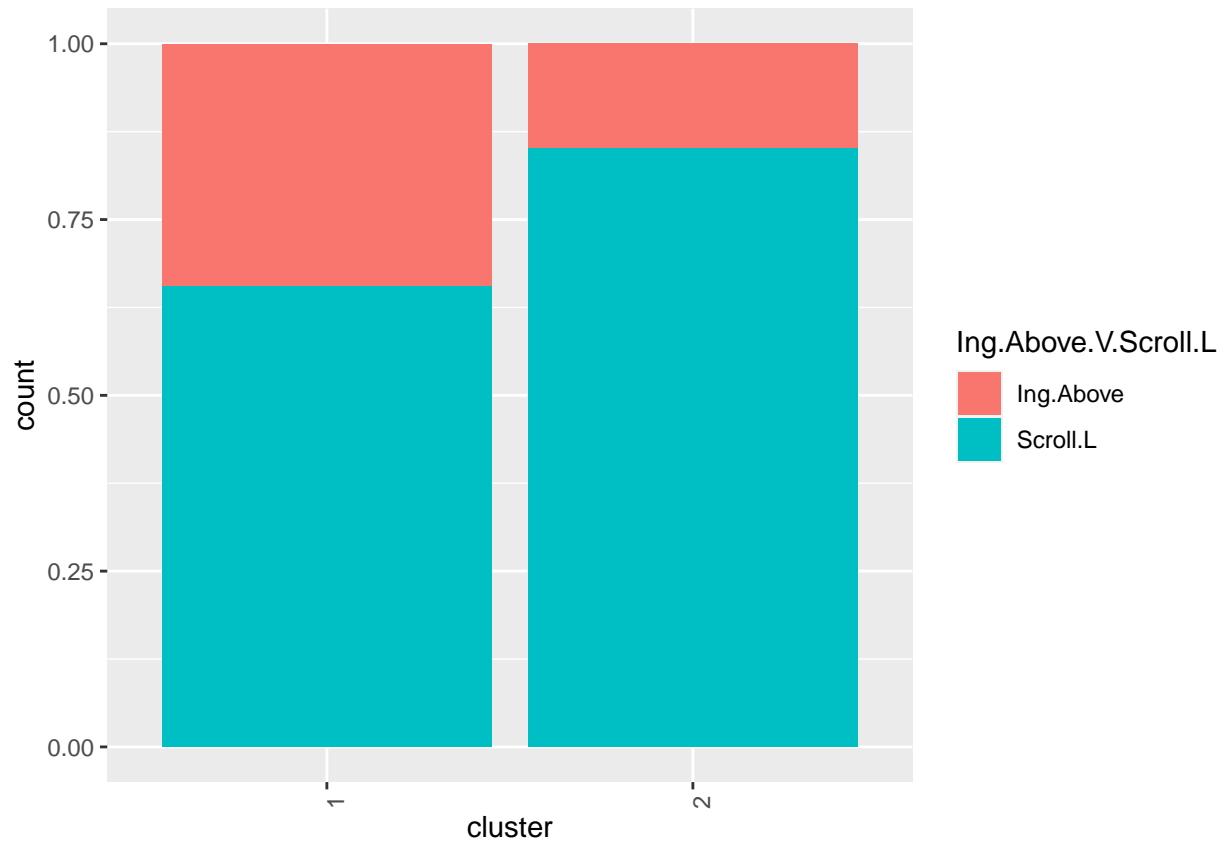












Since cluster two represents a group of users who interacts with recipes more frequently than group one, I will weight their responses higher than cluster one. # Results:

Ingredients Listed Left vs. Above:

Ingredients left

(Given Ingredients Left) Comments Inline vs. Below:

Comments inline

(Given Ingredients Above) Comments Below vs. Inline:

Comments inline

Ingredients Listed Step by Step vs. Above:

Ingredients listed step by step

Ingredients Listed Step by Step vs. Pinned Left While Scrolling:

Ingredients listed step by step

- Ingredients listed step by step is overall the most liked

Ingredients Listed Above vs. Pinned Left While Scrolling:

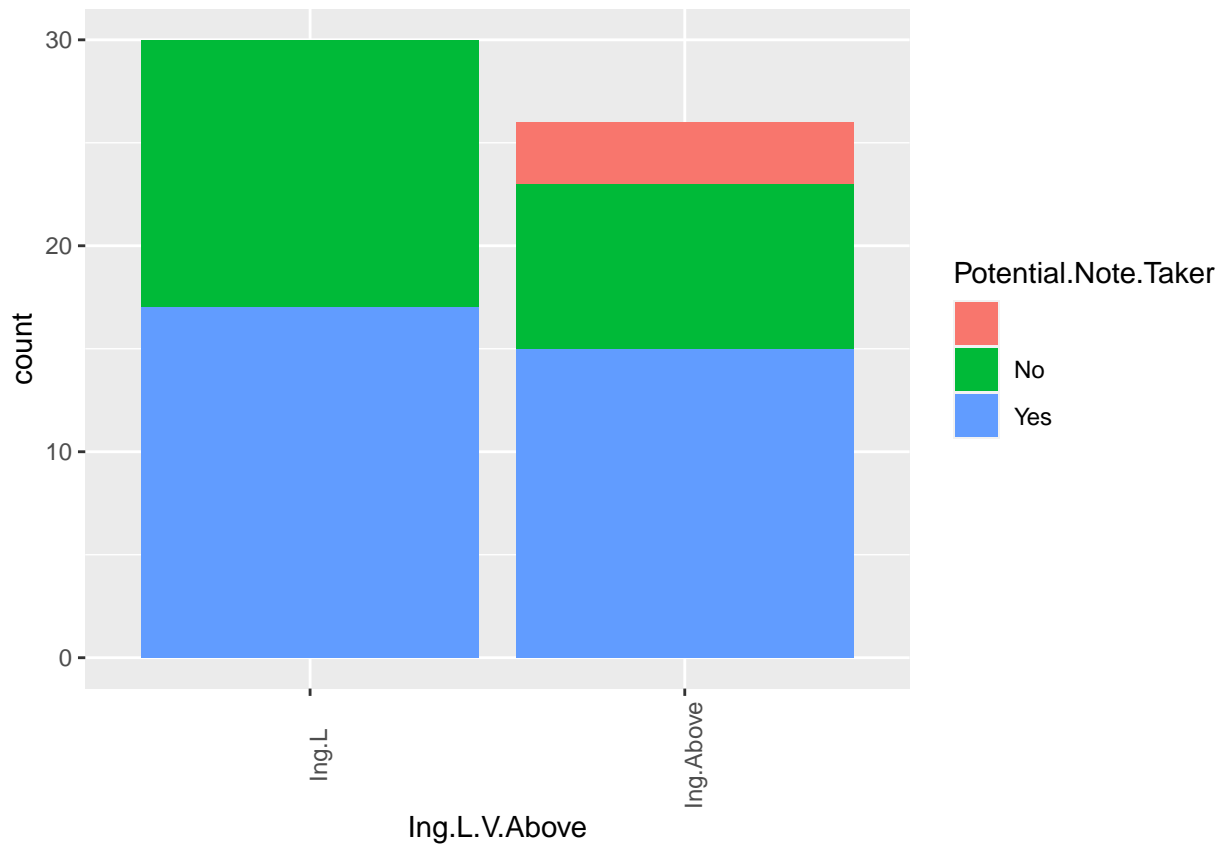
Ingredients pinned left while scrolling

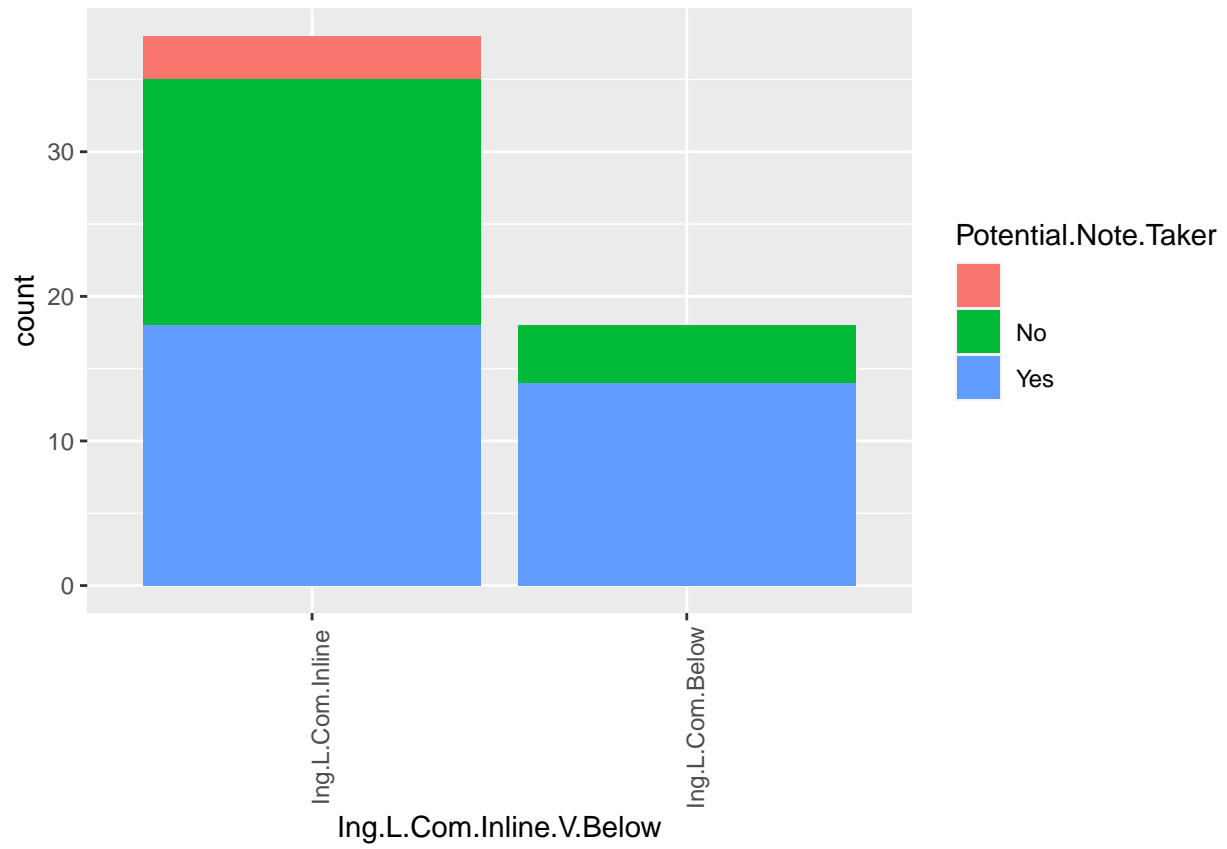
The overall recipe view most enjoyed by users is ingredients pinned left while scrolling with comments inline.

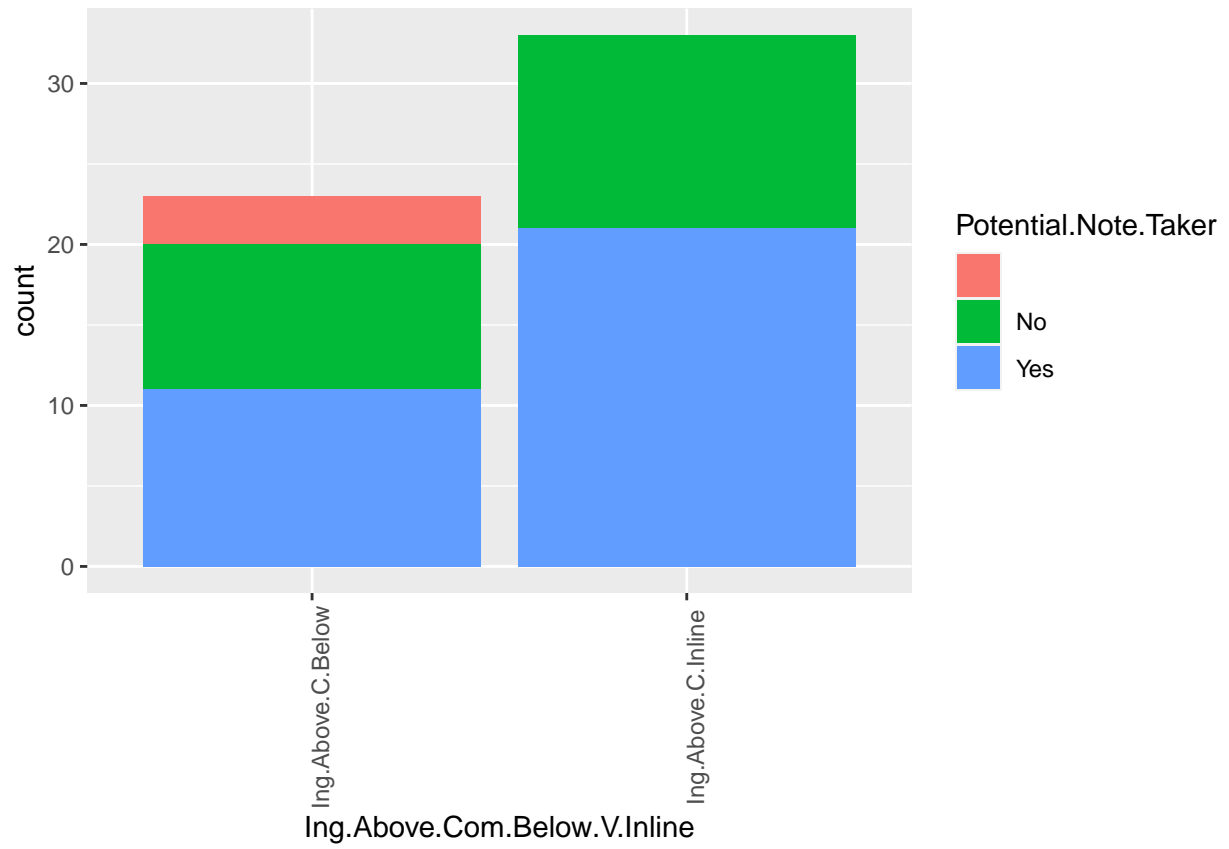
The in use recipe view most enjoyed by users is ingredients listed step by step.

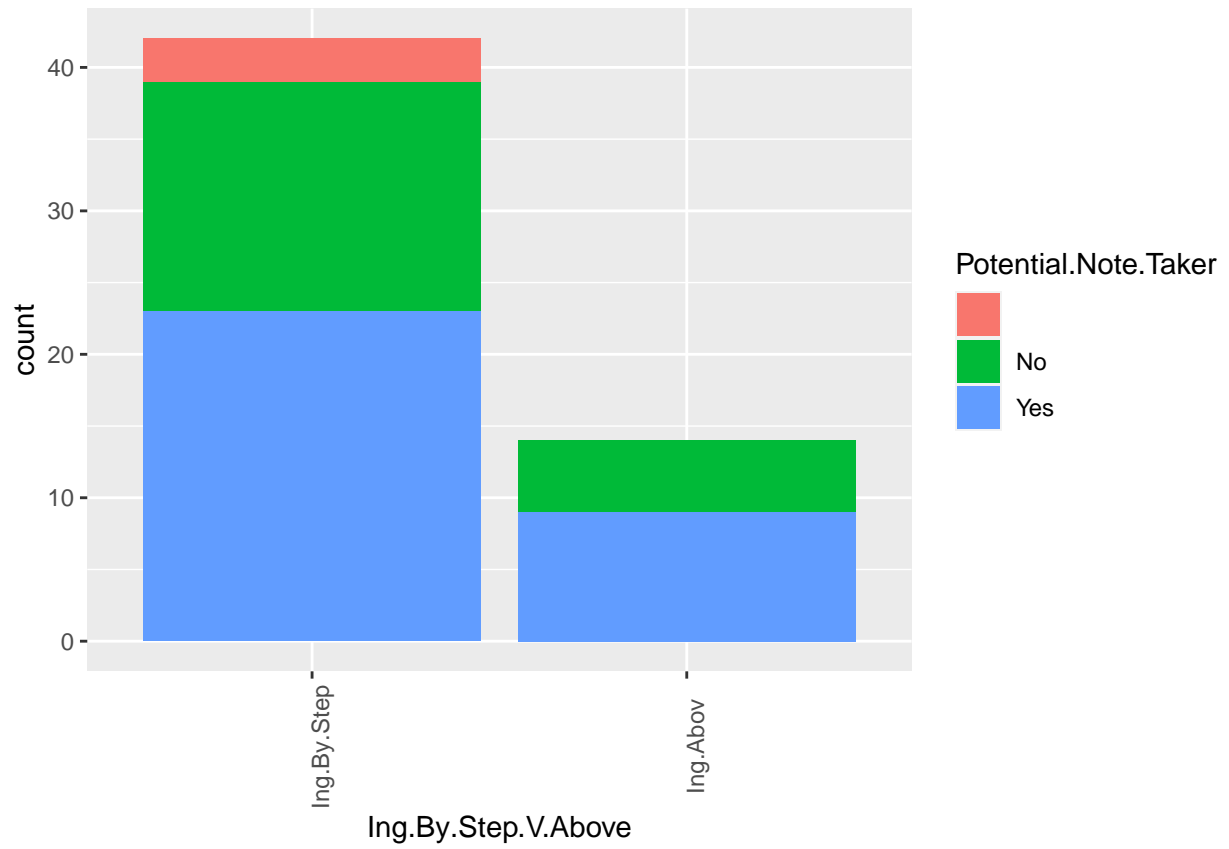
```
for (i in names(select(cleaned, starts_with("Ing.")))){
  p<-ggplot(cleaned,aes( fill=Potential.Note.Taker))+
    geom_bar(aes_string(x=i))+
    # facet_grid(~Meal.Prepper)+
    theme(axis.text.x = element_text(angle = 90))

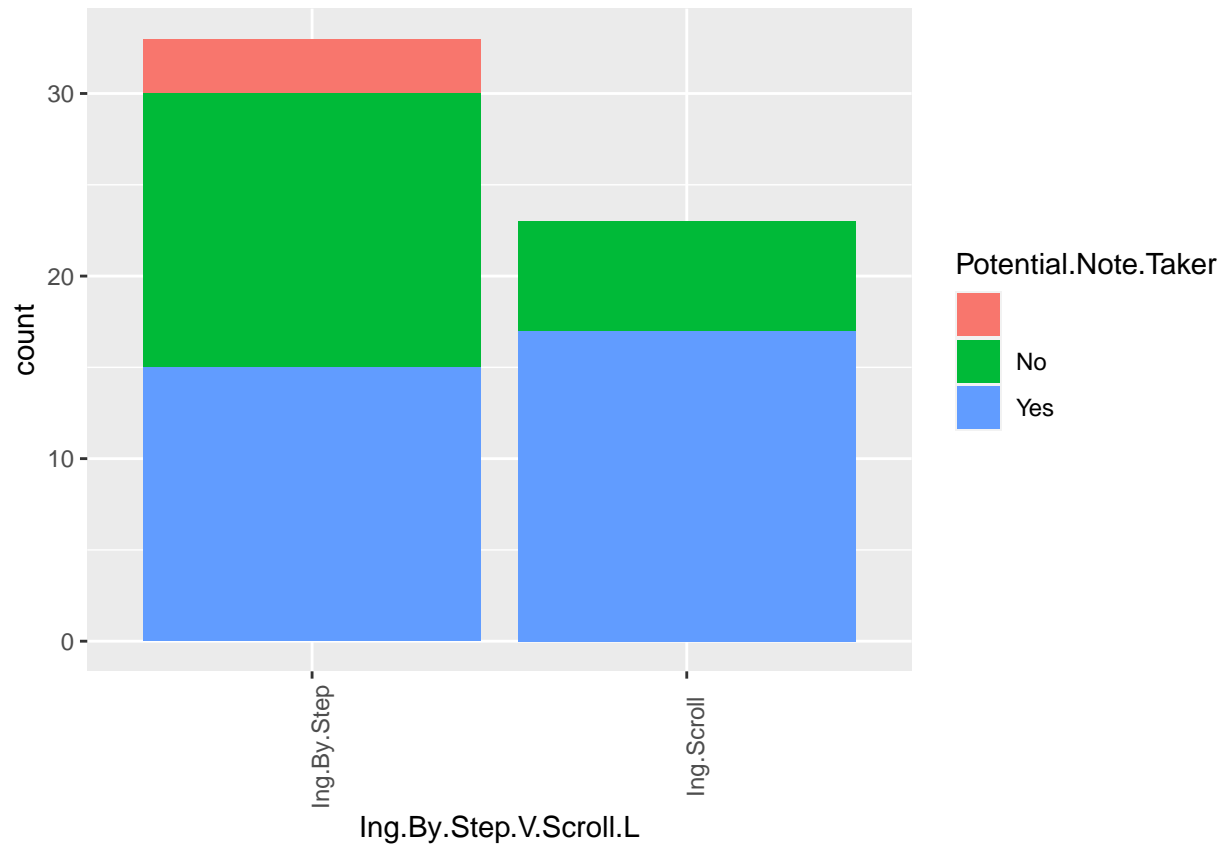
  print(p)
}
```

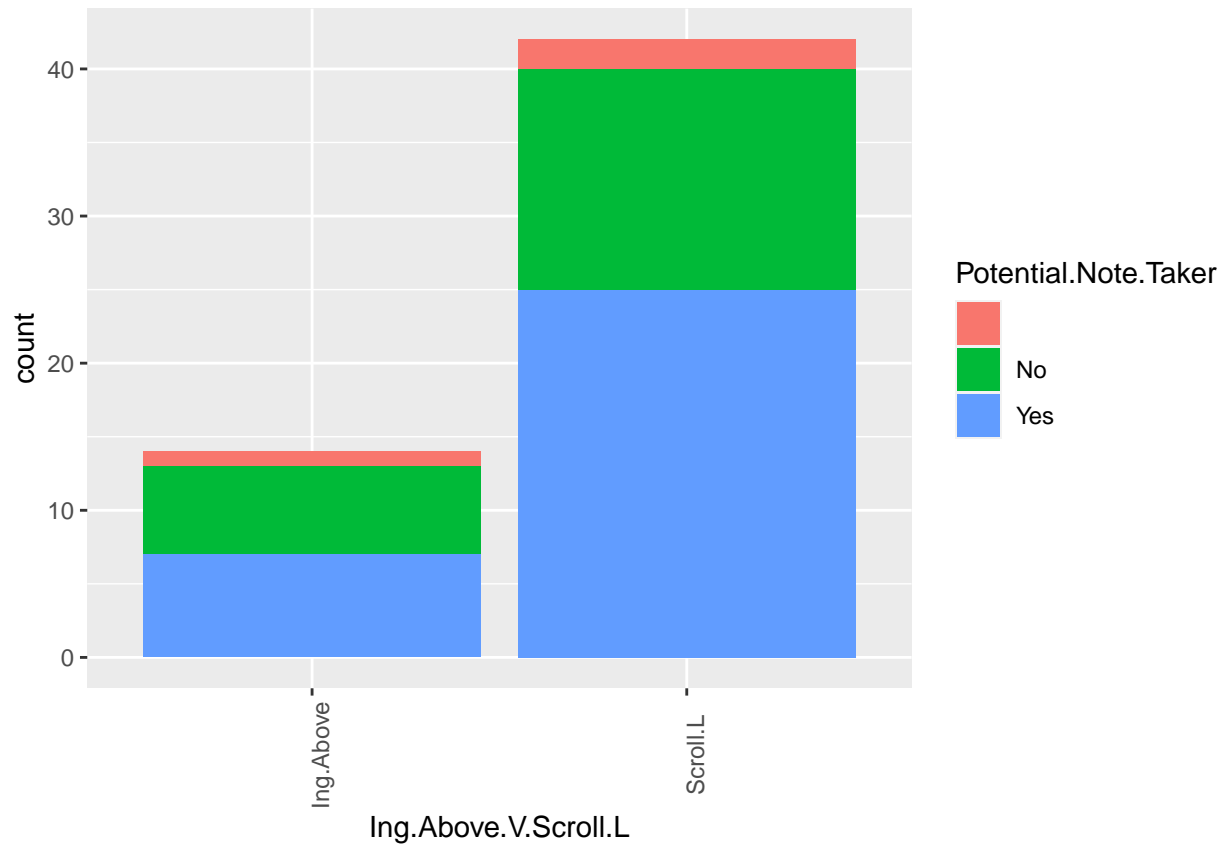






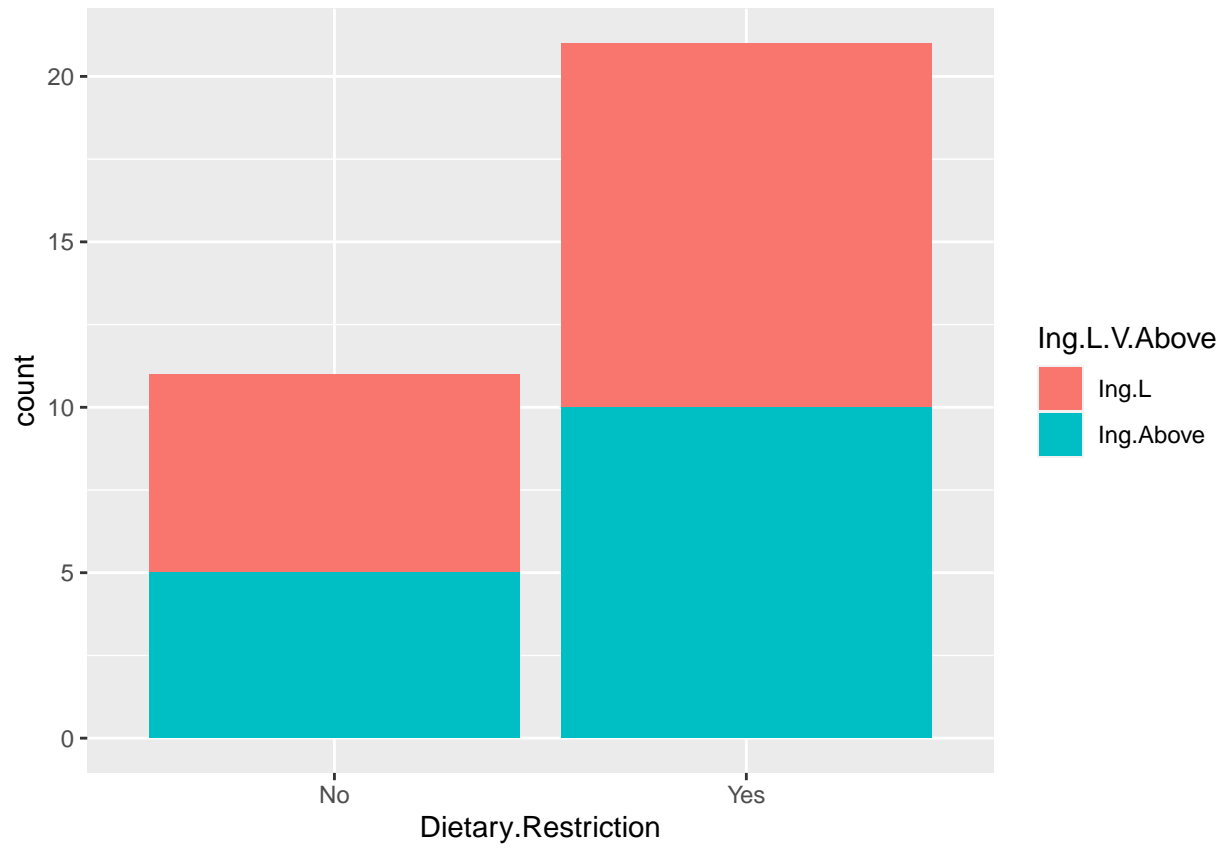


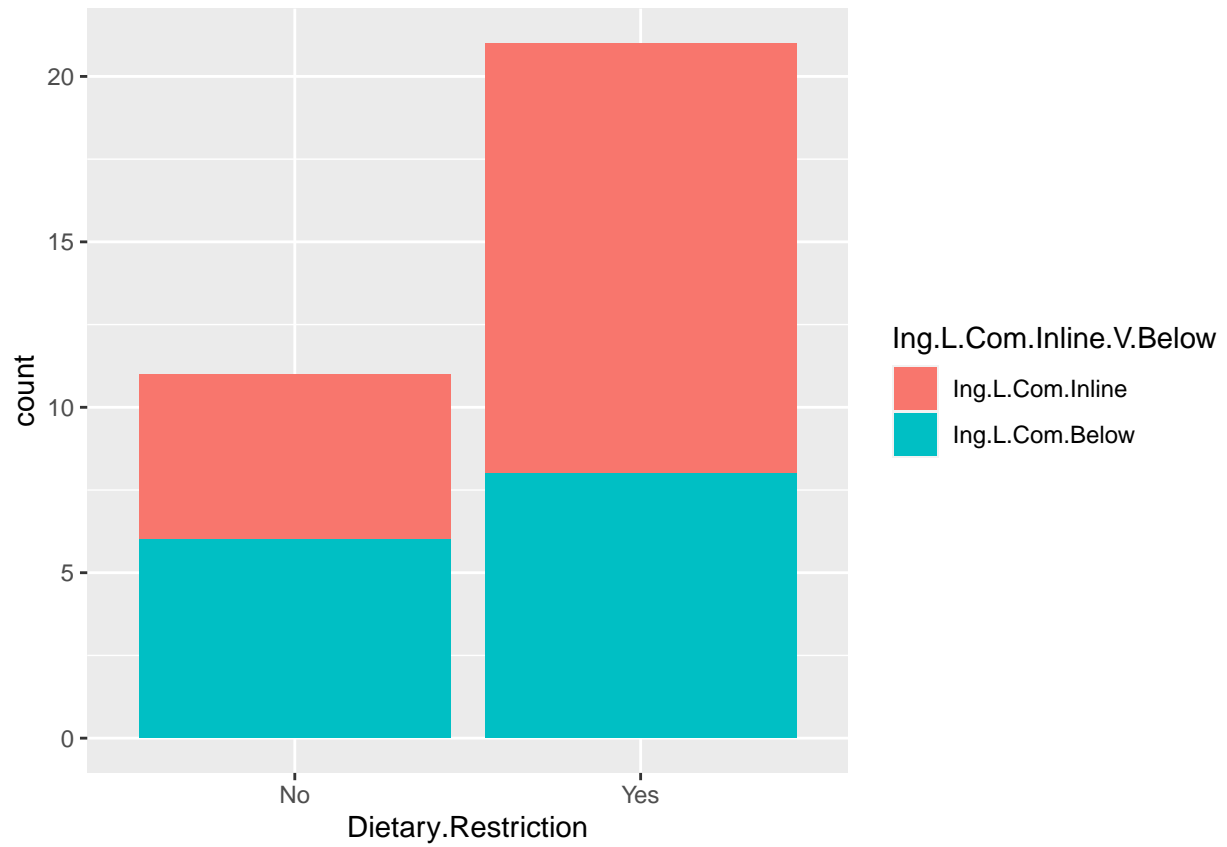


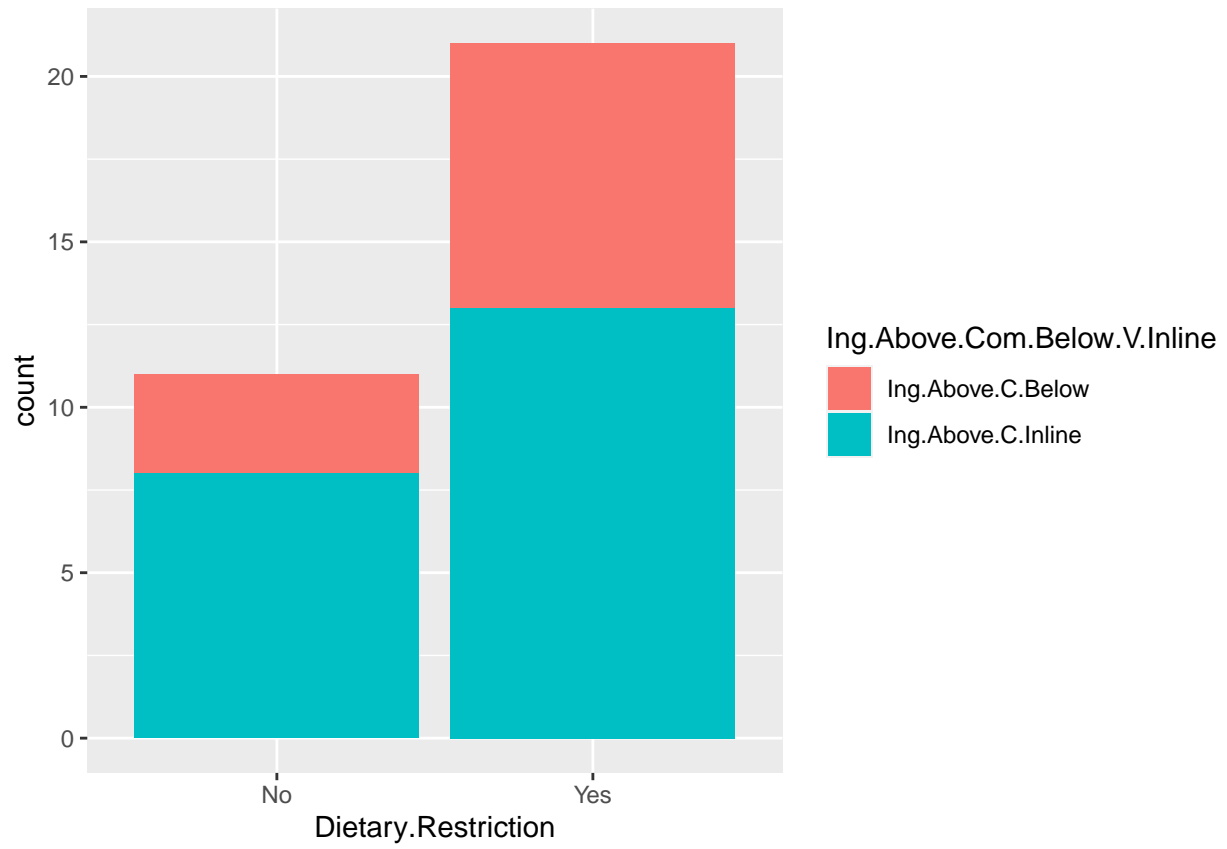


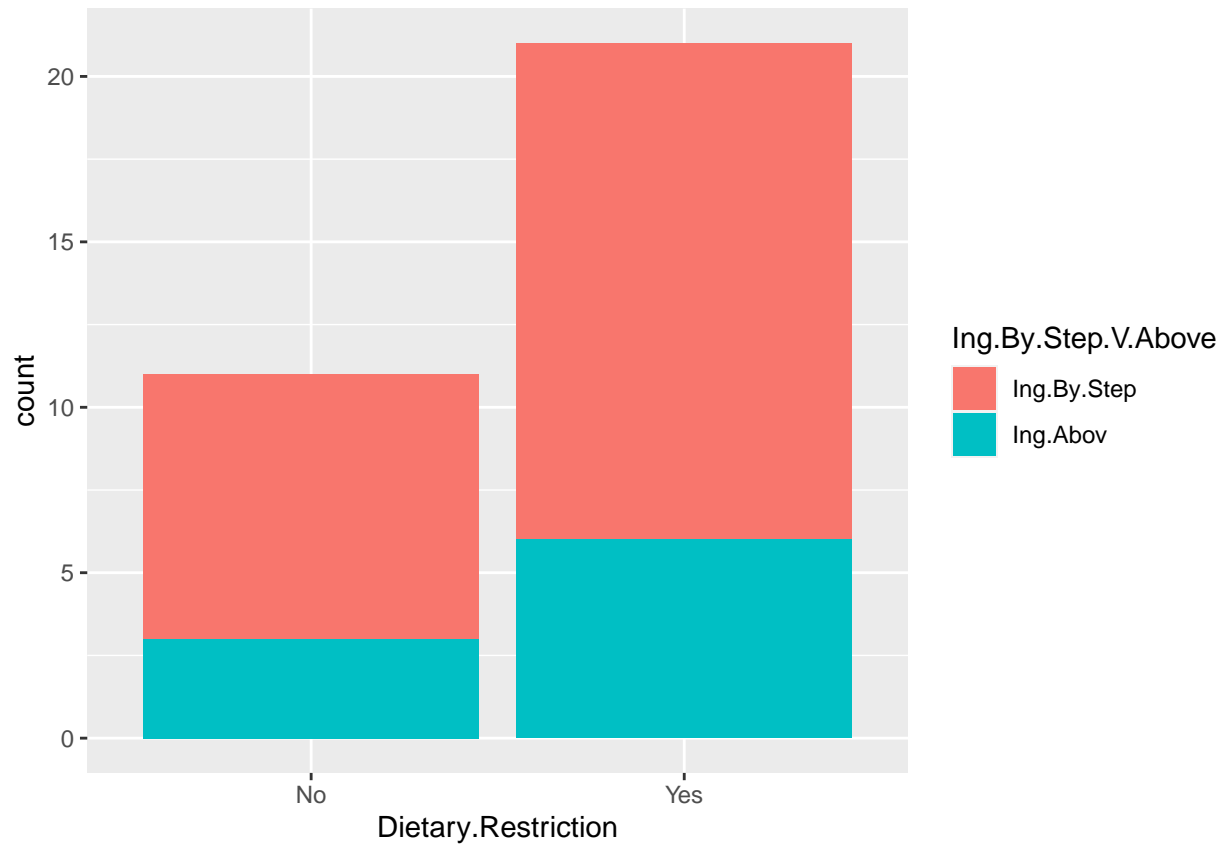
```
note.takers<-filter(cleaned, Potential.Note.Taker == "Yes")
for (i in names(select(note.takers, starts_with("Ing.")))){
  p<-ggplot(note.takers,aes_string( fill=i))+
    geom_bar(aes(x=Dietary.Restriction))#+
    # facet_grid(~Dietary.Restriction+Meal.Prepper)+
    theme(axis.text.x = element_text(angle = 90))

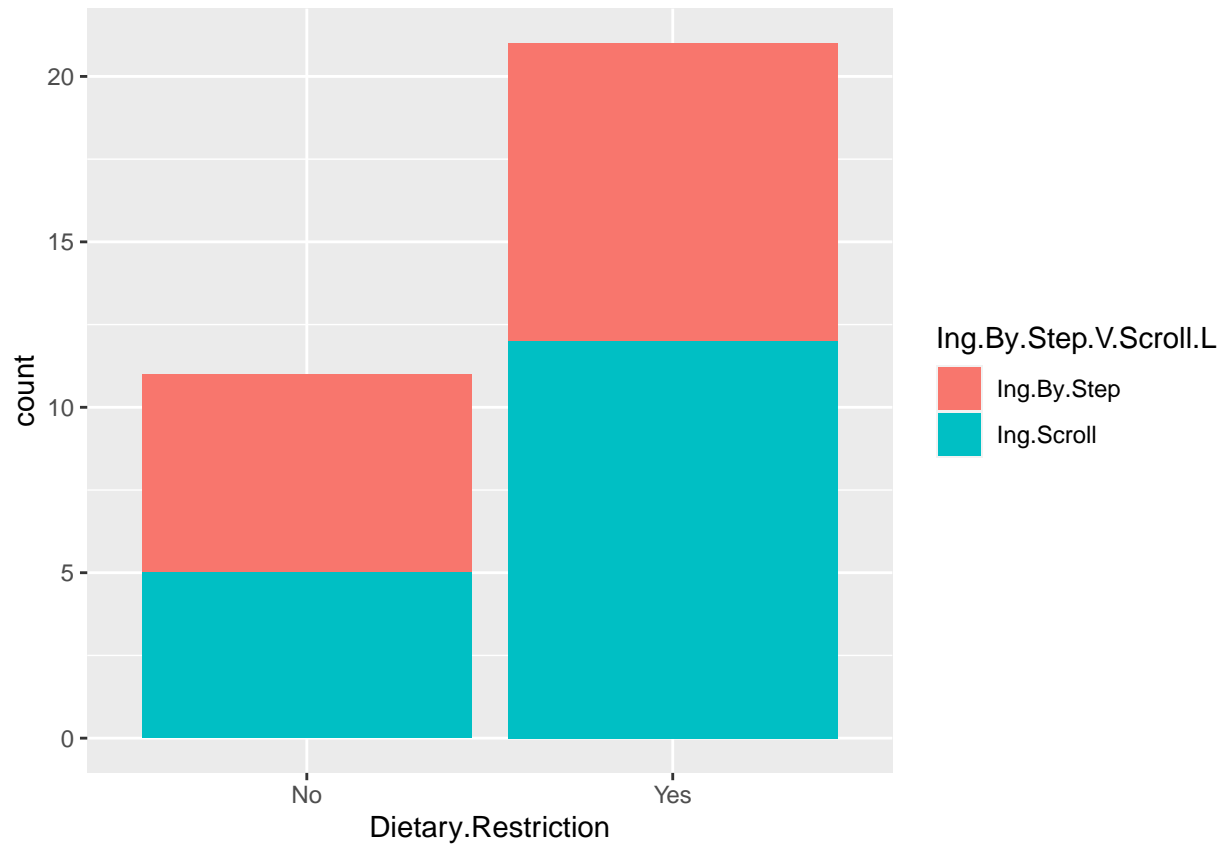
  print(p)
}
```

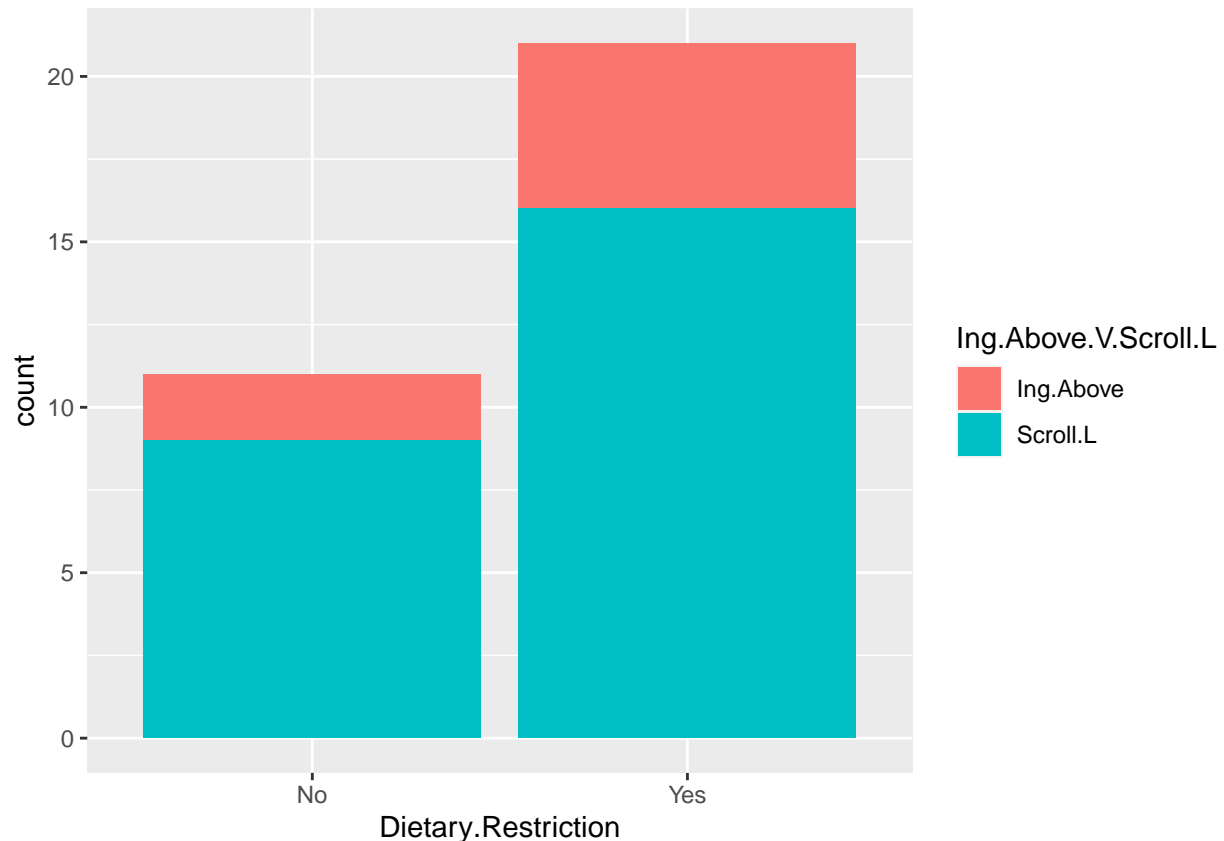












```
hc.complete =hclust (dist(note.takers), method ="complete")
```

```
## Warning in dist(note.takers): NAs introduced by coercion
```

```
hc.average =hclust (dist(note.takers), method ="average")
```

```
## Warning in dist(note.takers): NAs introduced by coercion
```

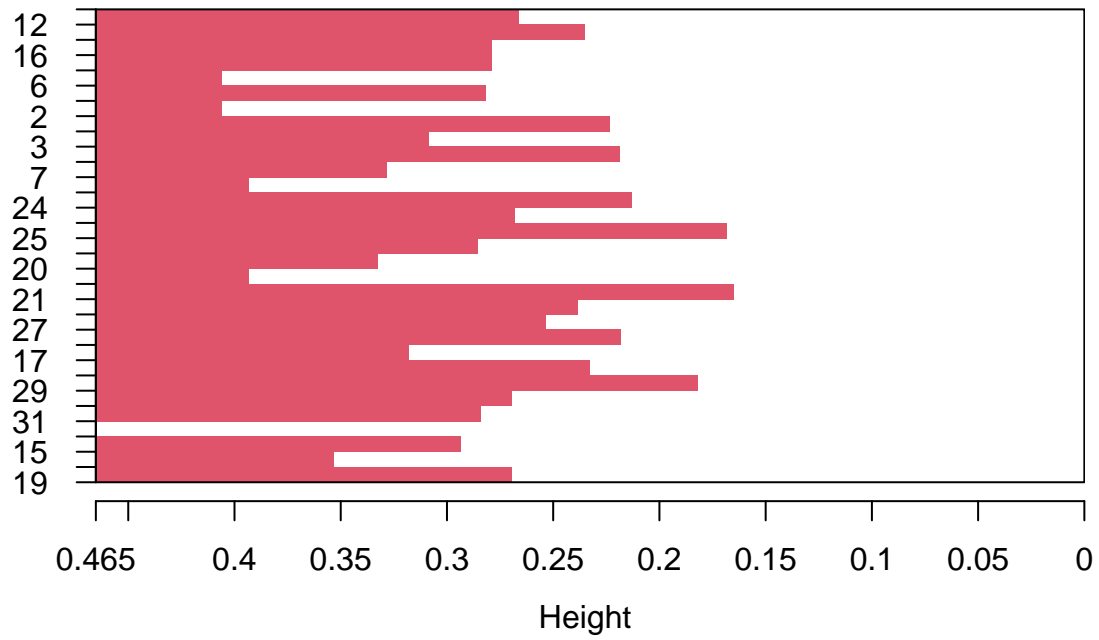
```
gower.dist <- daisy(note.takers, metric = c("gower"))
```

```
## Warning in daisy(note.takers, metric = c("gower")): binary variable(s) 26, 27,
## 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 44, 45, 46, 47,
## 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
## 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87,
## 88, 89, 91, 92, 93, 94, 95, 96, 97, 99, 100, 101, 102, 104, 106, 107, 108, 109,
## 110, 112, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127,
## 128, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142 treated as
## interval scaled
```

```
divisive.clust <- diana(as.matrix(gower.dist),
                        diss = TRUE, keep.diss = TRUE)
```

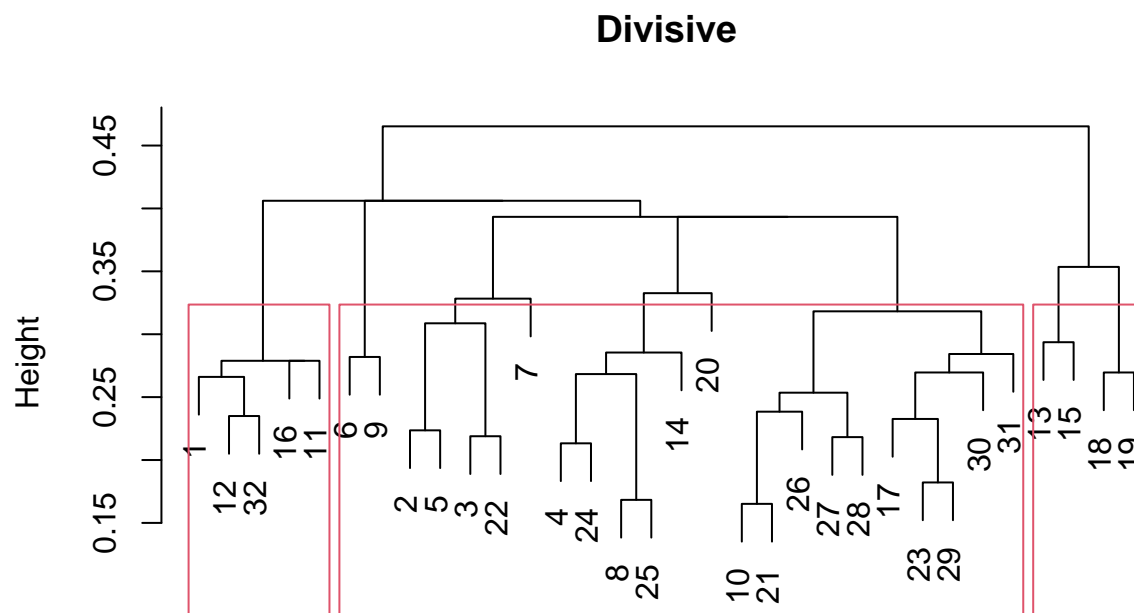
```
plot(divisive.clust, main = "Divisive")
```

Divisive



Divisive Coefficient = 0.48

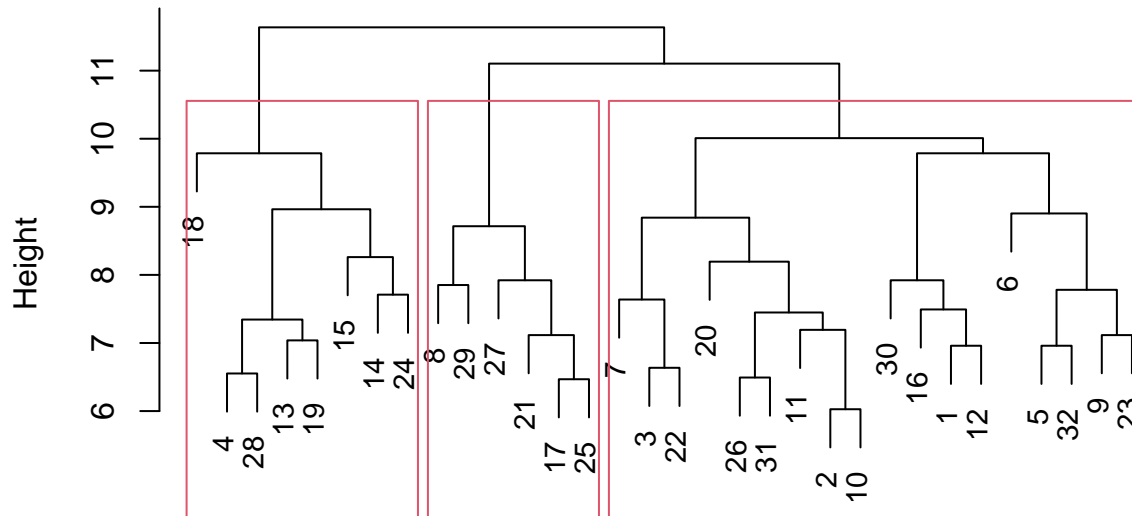
```
rect.hclust(divisive.clust,k=3)
```



as.matrix(gower.dist)
Divisive Coefficient = 0.48

```
plot(hc.complete ,main = " Complete Linkage ", xlab="", sub = "",
cex =.9)
rect.hclust(hc.complete,k=3)
```

Complete Linkage



```
plot(hc.average , main = " Average Linkage ", xlab="", sub = "",
cex = .9)
rect.hclust(hc.average,k=3)
```

Average Linkage

