Homework #2

Xiaowei Zhao

1. Flowers

Data: `flowers` dataset in **cluster** package

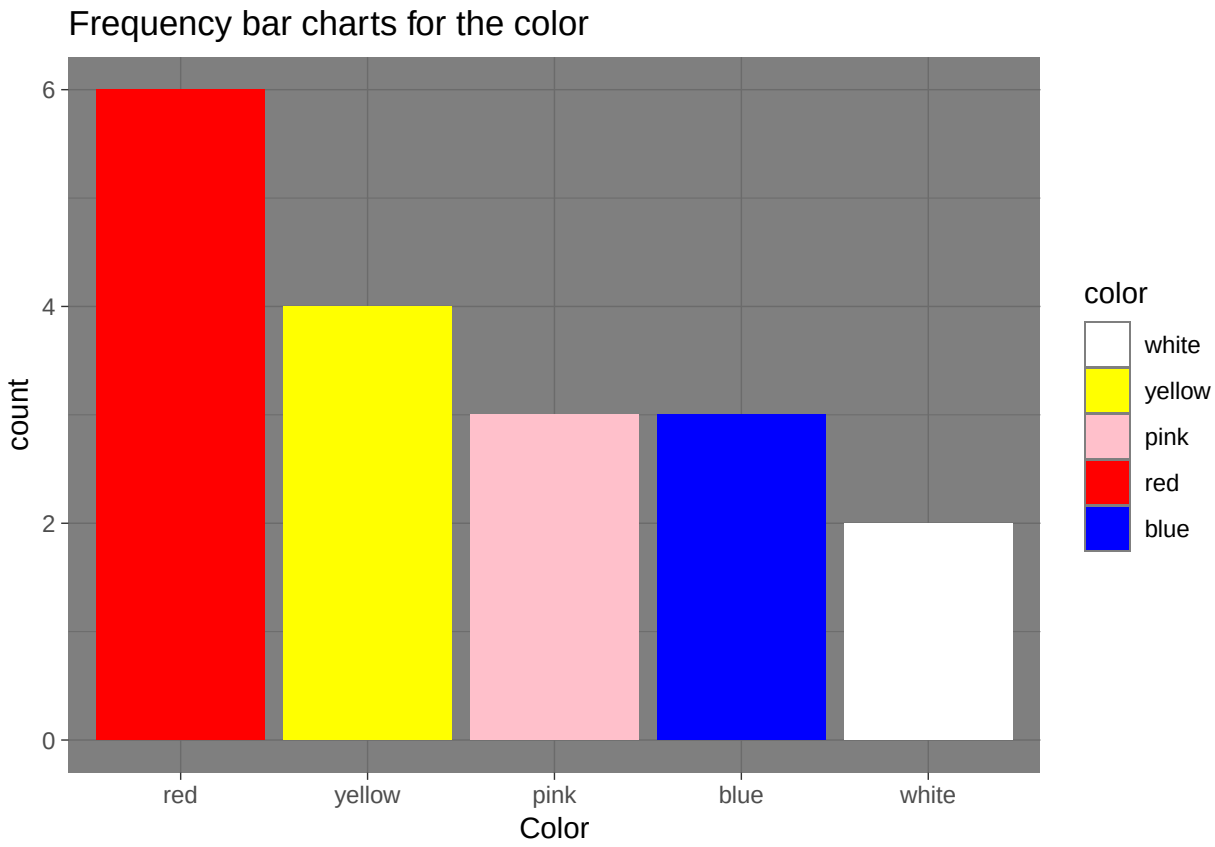
(a) Rename the column names and recode the levels of categorical variables to descriptive names. For example, "V1" should be renamed "winters" and the levels to "no" or "yes". Display the full dataset.

```
flowers=tbl_df(flower)
names(flowers)=c('winters','shadow','tubers','color','soil','preference','height','distance')
levels(flowers$winters)=c('no','yes')
levels(flowers$shadow)=c('no','yes')
levels(flowers$tubers)=c('no','yes')
levels(flowers$color)=c('white','yellow','pink','red','blue')
levels(flowers$soil)=c('dry','normal','wet')
flowers
```

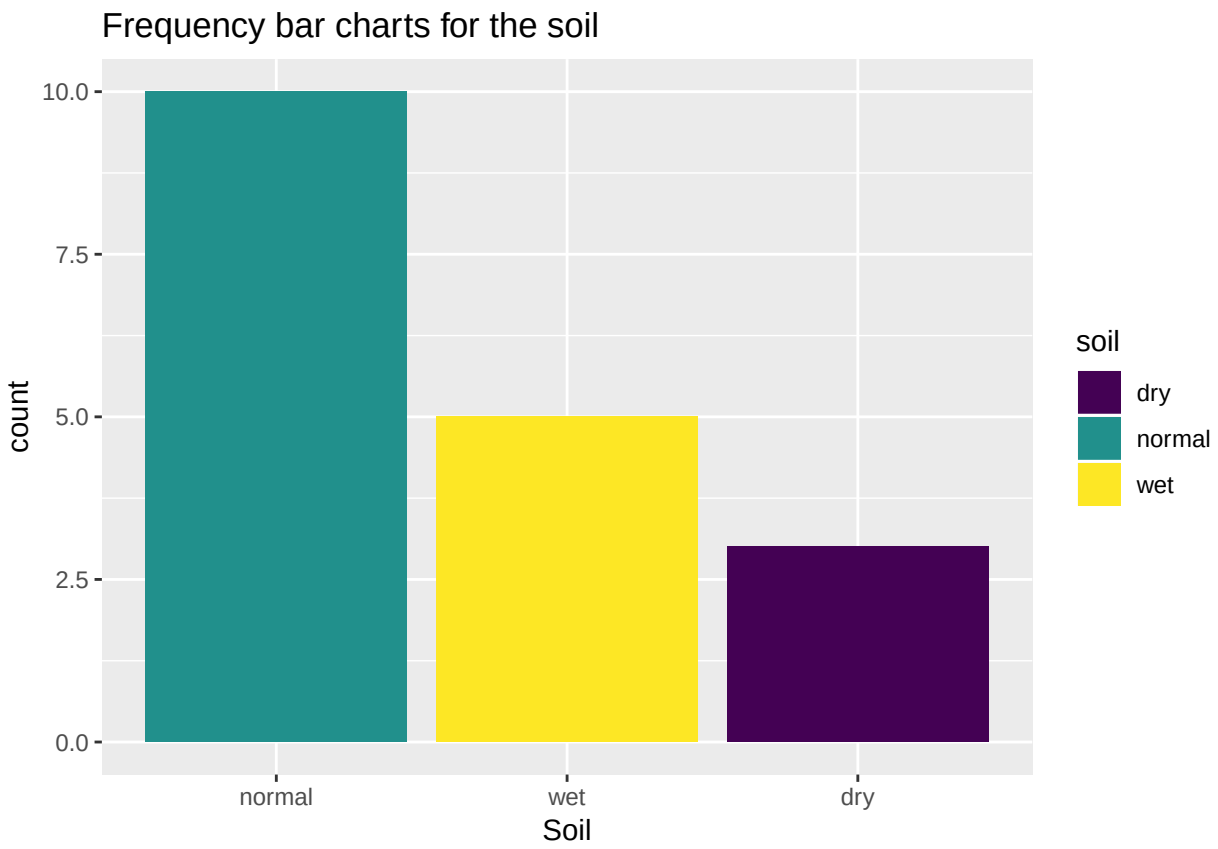
```
## # A tibble: 18 x 8
##    winters shadow tubers color  soil   preference height distance
##    <fct>   <fct>  <fct>  <fct>  <ord>  <ord>       <dbl>    <dbl>
##  1 no      yes    yes    red    wet    15             25       15
##  2 yes     no     no     yellow dry    3             150       50
##  3 no      yes    no     pink   wet    1             150       50
##  4 no      no     yes    red    normal 16            125       50
##  5 no      yes    no     blue   normal 2              20       15
##  6 no      yes    no     red    wet    12             50       40
##  7 no      no     no     red    wet    13             40       20
##  8 no      no     yes    yellow normal 7             100       15
##  9 yes     yes    no     pink   dry    4              25       15
## 10 yes     yes    no     blue   normal 14            100       60
## 11 yes     yes    yes    blue   wet    8              45       10
## 12 yes     yes    yes    white  normal 9              90       25
## 13 yes     yes    no     white  normal 6              20       10
## 14 yes     yes    yes    red    normal 11             80       30
## 15 yes     no     no     pink   normal 10             40       20
## 16 yes     no     no     red    normal 18            200       60
## 17 yes     no     no     yellow normal 17            150       60
## 18 no      no     yes    yellow dry    5              25       10
```

(b) Create frequency bar charts for the `color` and `soil` variables, using best practices for the order of the bars.

```
p1a=ggplot(flowers,aes(x=fct_infreq(color),fill=color)) +
  geom_bar() +
  scale_fill_manual(values = c('white','yellow','pink','red','blue')) +
  xlab('Color') +
  ggtitle('Frequency bar charts for the color') +
  theme_dark()
p1a
```




```
p1b=ggplot(flowers,aes(x=fct_infreq(soil),fill=soil)) +
  geom_bar() +
  xlab('Soil') +
  ggtitle('Frequency bar charts for the soil')
p1b
```



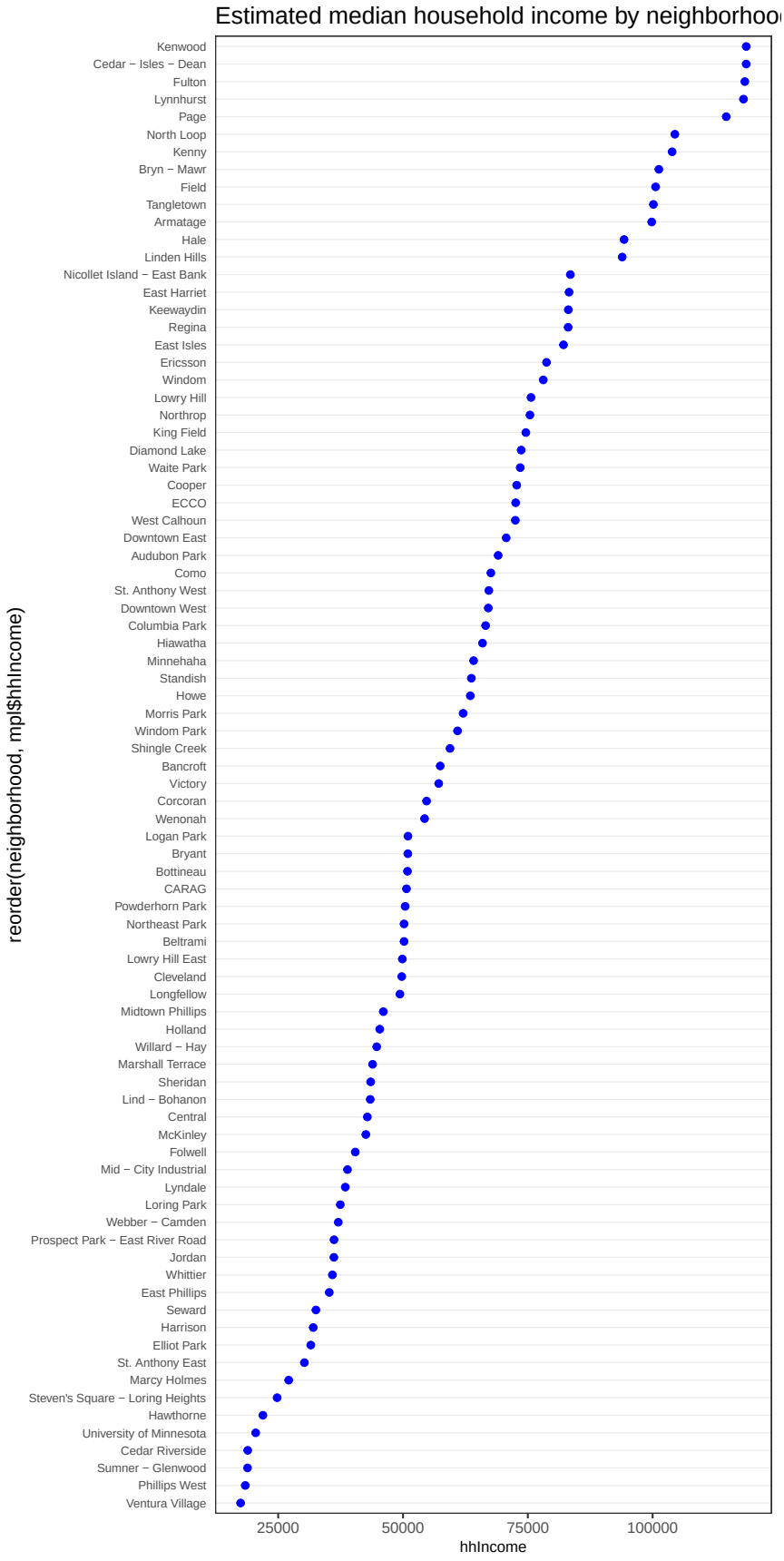
2. Minneapolis

Data: `MplsDemo` dataset in **carData** package

(a) Create a Cleveland dot plot showing estimated median household income by neighborhood.

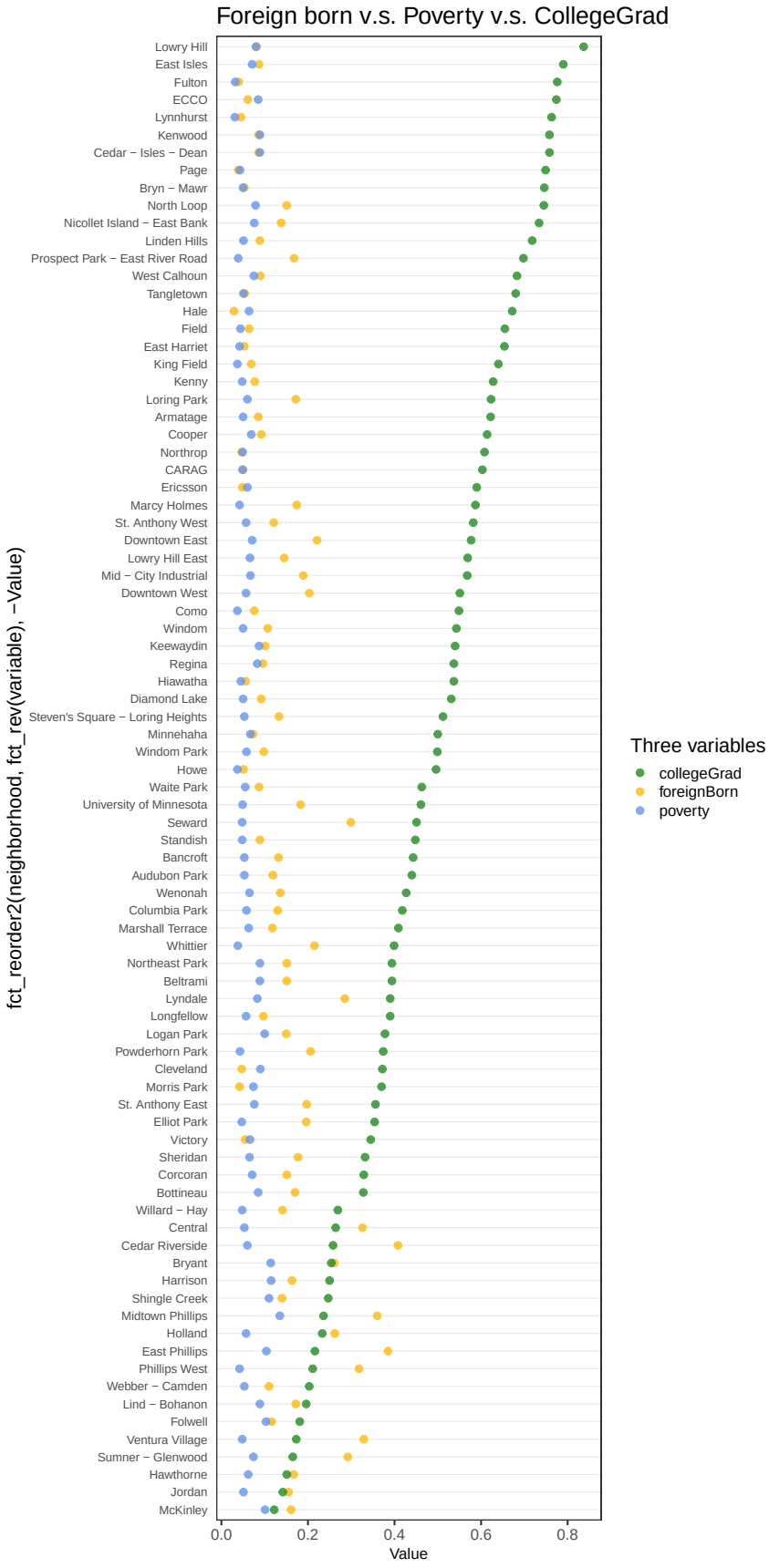
```
theme_dotplot=theme_bw(18) +
    theme(axis.text.y = element_text(size = rel(.75)),
          axis.ticks.y = element_blank(),
          axis.title.x = element_text(size = rel(.75)),
          panel.grid.major.x = element_blank(),
          panel.grid.major.y = element_line(size = 0.5),
          panel.grid.minor.x = element_blank())
mpl=tbl_df(MplsDemo)
p2a=ggplot(mpl,aes(x=hhIncome,y=reorder(neighborhood,mpl$hhIncome))) +
         geom_point(col='blue',size=3) +
         theme_dotplot +
         ggtitle('Estimated median household income by neighborhood')

p2a
```


Estimated median household income by neighborhoo
reorder(neighborhood, mpl$hhIncome)
Kenwood
Cedar − Isles − Dean
Fulton
Lynnhurst
Page
North Loop
Kenny
Bryn − Mawr
Field
Tangletown
Armatage
Hale
Linden Hills
Nicollet Island − East Bank
East Harriet
Keewaydin
Regina
East Isles
Ericsson
Windom
Lowry Hill
Northrop
King Field
Diamond Lake
Waite Park
Cooper
ECCO
West Calhoun
Downtown East
Audubon Park
Como
St. Anthony West
Downtown West
Columbia Park
Hiawatha
Minnehaha
Standish
Howe
Morris Park
Windom Park
Shingle Creek
Bancroft
Victory
Corcoran
Wenonah
Logan Park
Bryant
Bottineau
CARAG
Powderhorn Park
Northeast Park
Beltrami
Lowry Hill East
Cleveland
Longfellow
Midtown Phillips
Holland
Willard − Hay
Marshall Terrace
Sheridan
Lind − Bohanon
Central
McKinley
Folwell
Mid − City Industrial
Lyndale
Loring Park
Webber − Camden
Prospect Park − East River Road
Jordan
Whittier
East Phillips
Seward
Harrison
Elliot Park
St. Anthony East
Marcy Holmes
Steven's Square − Loring Heights
Hawthorne
University of Minnesota
Cedar Riverside
Sumner − Glenwood
Phillips West
Ventura Village
25000
50000
75000
100000
hhIncome

(b) Create a Cleveland dot plot *with multiple dots* to show percentage of 1) foreign born, 2) earning less than twice the poverty level, and 3) with a college degree *by neighborhood.* Each of these three continuous variables should appear in a different color. Data should be sorted by college degree.

```
mpl2<-gather(mpl,key=variable,value=Value,foreignBorn,poverty,collegeGrad)
p2b=ggplot(mpl2,aes(x=Value,y=fct_reorder2(neighborhood, fct_rev(variable), -Value))) +
  geom_point(aes(col=variable),alpha=0.8,size=3) +
  scale_color_manual('Three variables',values=c('forestgreen', 'darkgoldenrod1', 'cornflowerblue')) +
  theme_dotplot +
  ggtitle('Foreign born v.s. Poverty v.s. CollegeGrad')
p2b
```


Foreign born v.s. Poverty v.s. CollegeGrad
fct_reorder2(neighborhood, fct_rev(variable), −Value)
Lowry Hill
East Isles
Fulton
ECCO
Lynnhurst
Kenwood
Cedar − Isles − Dean
Page
Bryn − Mawr
North Loop
Nicollet Island − East Bank
Linden Hills
Prospect Park − East River Road
West Calhoun
Tangletown
Hale
Field
East Harriet
King Field
Kenny
Loring Park
Armatage
Cooper
Northrop
CARAG
Ericsson
Marcy Holmes
St. Anthony West
Downtown East
Lowry Hill East
Mid − City Industrial
Downtown West
Como
Windom
Keewaydin
Regina
Hiawatha
Diamond Lake
Steven's Square − Loring Heights
Minnehaha
Windom Park
Howe
Waite Park
University of Minnesota
Seward
Standish
Bancroft
Audubon Park
Wenonah
Columbia Park
Marshall Terrace
Whittier
Northeast Park
Beltrami
Lyndale
Longfellow
Logan Park
Powderhorn Park
Cleveland
Morris Park
St. Anthony East
Elliot Park
Victory
Sheridan
Corcoran
Bottineau
Willard − Hay
Central
Cedar Riverside
Bryant
Harrison
Shingle Creek
Midtown Phillips
Holland
East Phillips
Phillips West
Webber − Camden
Lind − Bohanon
Folwell
Ventura Village
Sumner − Glenwood
Hawthorne
Jordan
McKinley
0.0
0.2
0.4
0.6
0.8
Value
Three variables
collegeGrad
foreignBorn
poverty

(c) What patterns do you observe? What neighborhoods do not appear to follow these patterns?

I observe that the percentage of people with college degree is higher than both foreign born and poverty rate. However, Central, Cedar Riverside, Midtown Phillips, Holland, East Phillips, Phipplips West, Ventura Village and Sumner-Glenwood are those neighbours which don't follow this pattern. This maybe related to the higher foreign born rate in those neighbours and they also have a relatively higher poverty rate. And also, those neighbours with higher college degree rate have relatively lower poverty and foreign born rate.

3. Taxis

Data: NYC yellow cab rides in June 2018, available here:

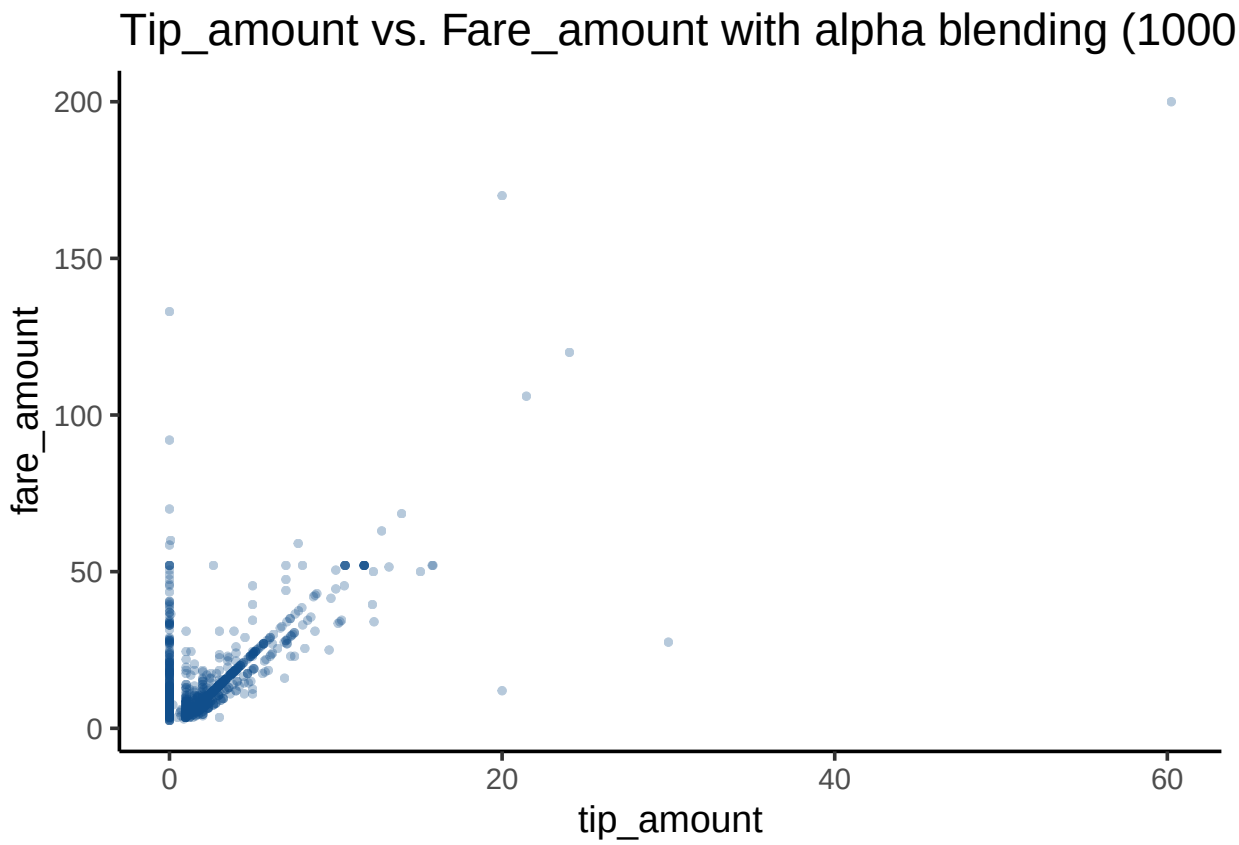
http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

It's a large file so work with a reasonably-sized random subset of the data.

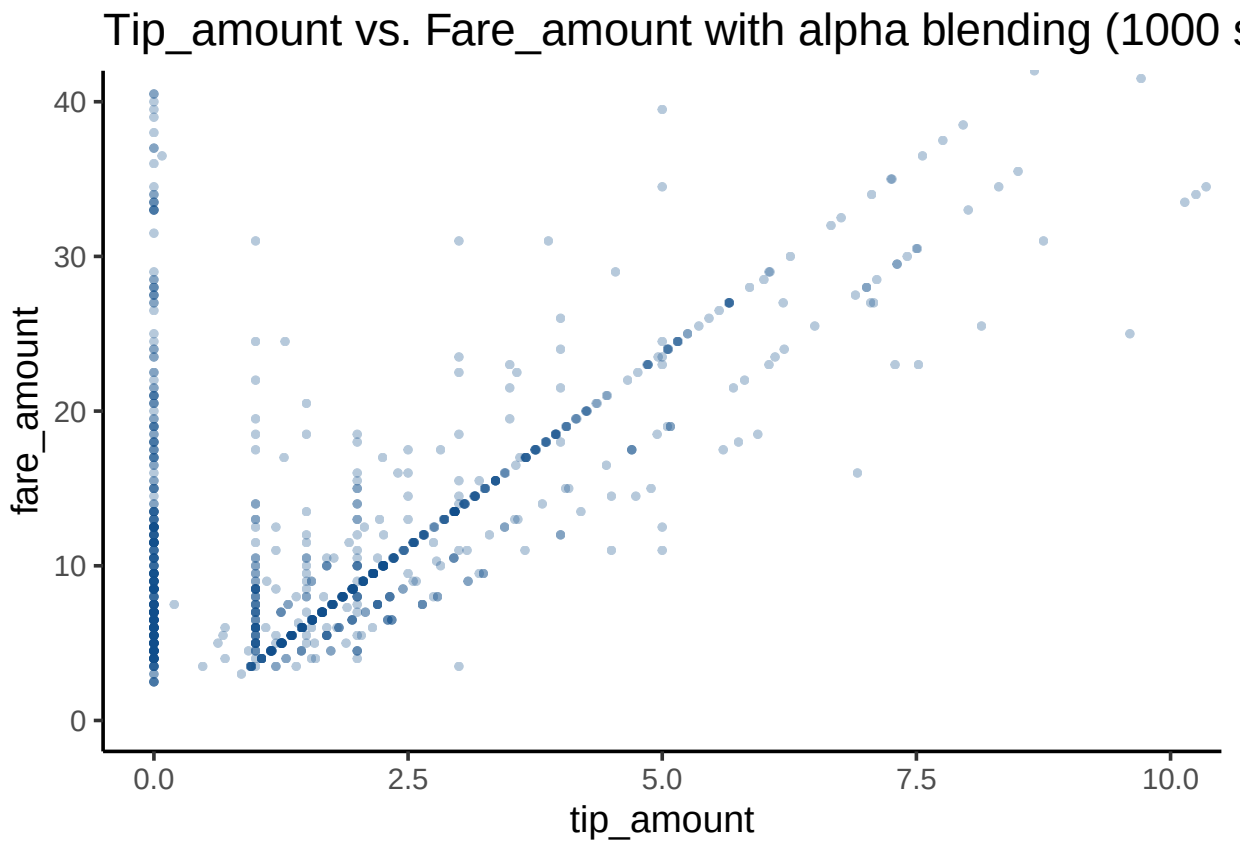
Draw four scatterplots of `tip_amount` vs. `far_amount` with the following variations:

(a) Points with alpha blending

```
taxi=read_csv('yellow_tripdata_2018-06.csv',n_max=10000)
set.seed(123)
taxi_sample=tbl_df(sample_n(taxi,size=1000))
p3a=ggplot(taxi_sample, aes(x=tip_amount, y=fare_amount)) +
  geom_point(alpha = 0.3, color = "dodgerblue4", stroke = 0) +
  #coord_cartesian(xlim=c(0,10),ylim=c(0,40)) +
  theme_classic(14) +
  ggtitle('Tip_amount vs. Fare_amount with alpha blending (1000 sample)')
p3a
```



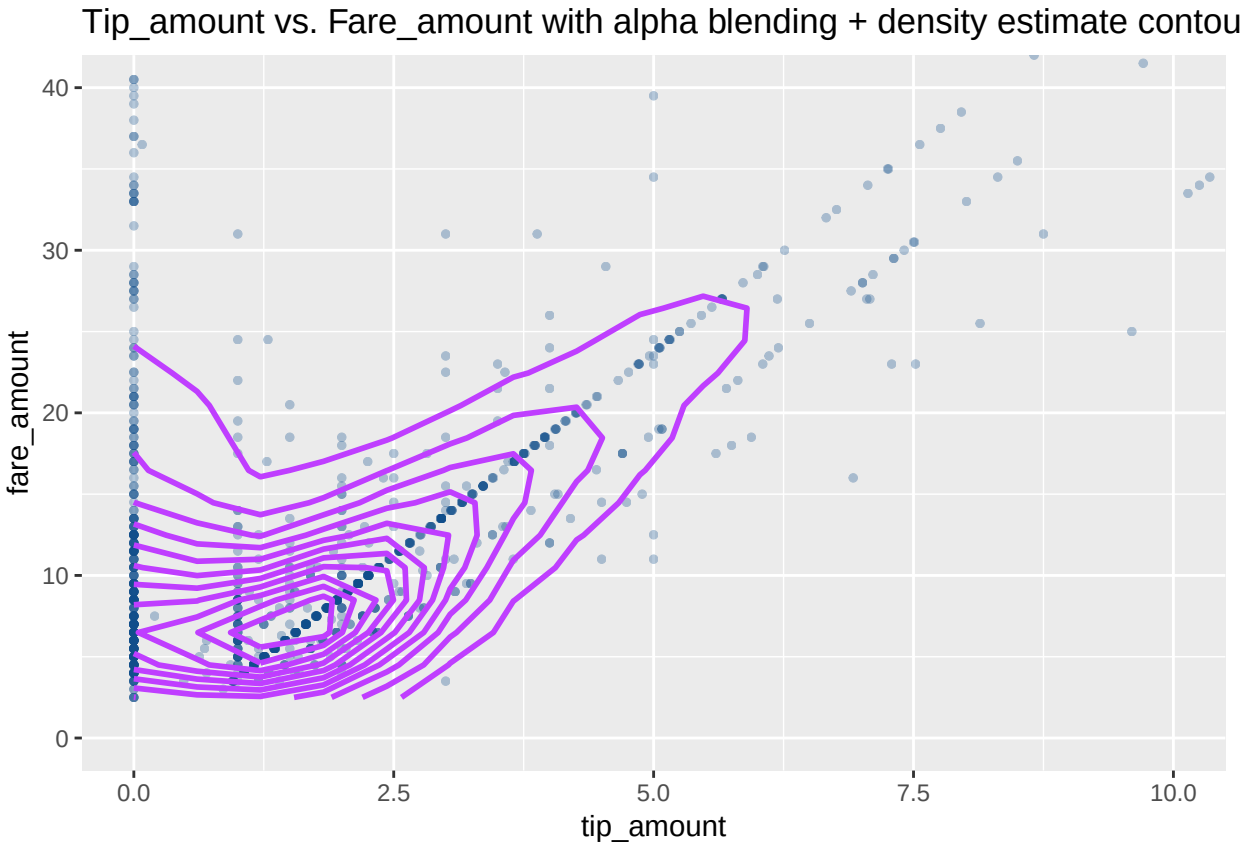

```
p3a + coord_cartesian(xlim=c(0,10),ylim=c(0,40))
```



I zoom in the graph here to obeserve the graph more comfortably and clearly.There seems to be some outliers

(b) Points with alpha blending + density estimate contour lines

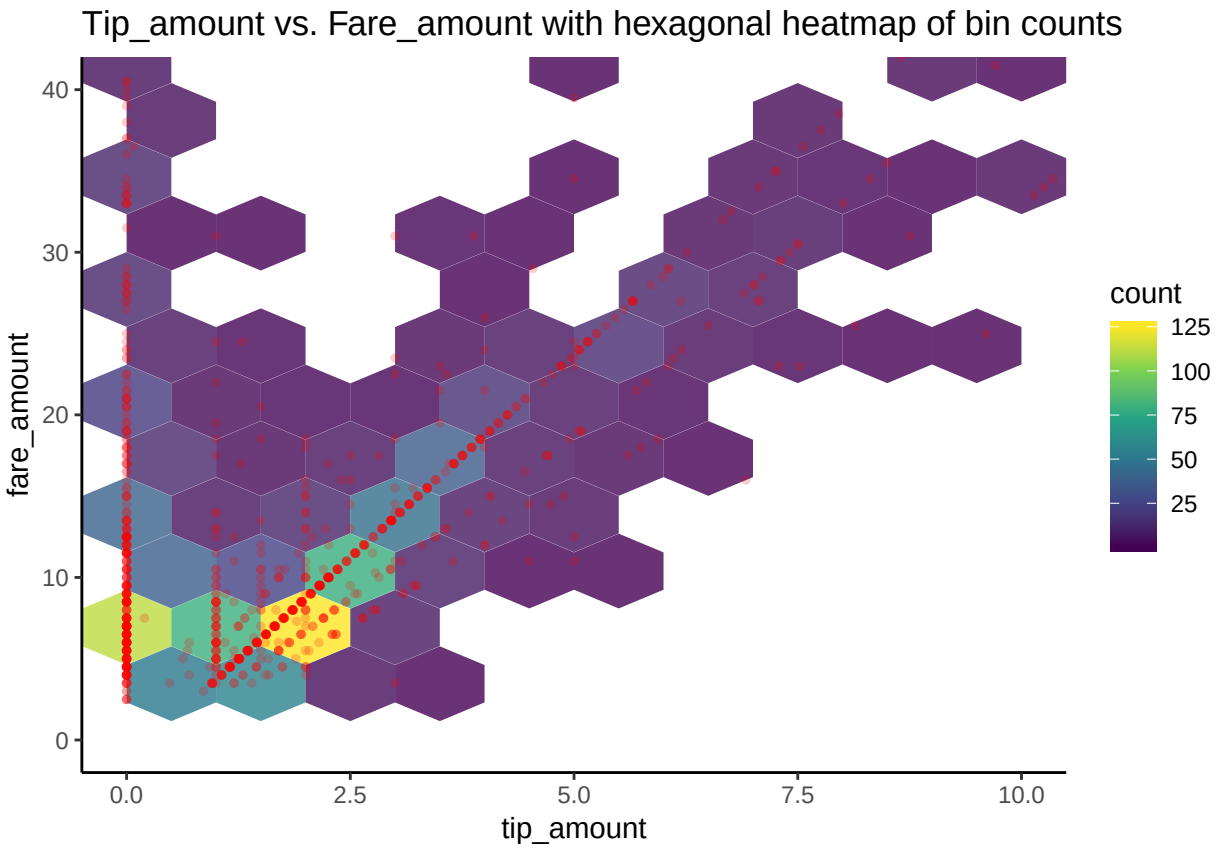
```
p3b=ggplot(taxi_sample, aes(x=tip_amount, y=fare_amount)) +
  geom_point(alpha = 0.3, color = "dodgerblue4", stroke = 0) +
  geom_density_2d(col='darkorchid1',size=1) +
  coord_cartesian(xlim=c(0,10),ylim=c(0,40)) +
  theme_grey() +
  ggtitle('Tip_amount vs. Fare_amount with alpha blending + density estimate contour lines')
p3b
```



** Also, I zoom in the graph here to see the density estimate contour lines more clearly. Also, I zoom in the two graphs below**

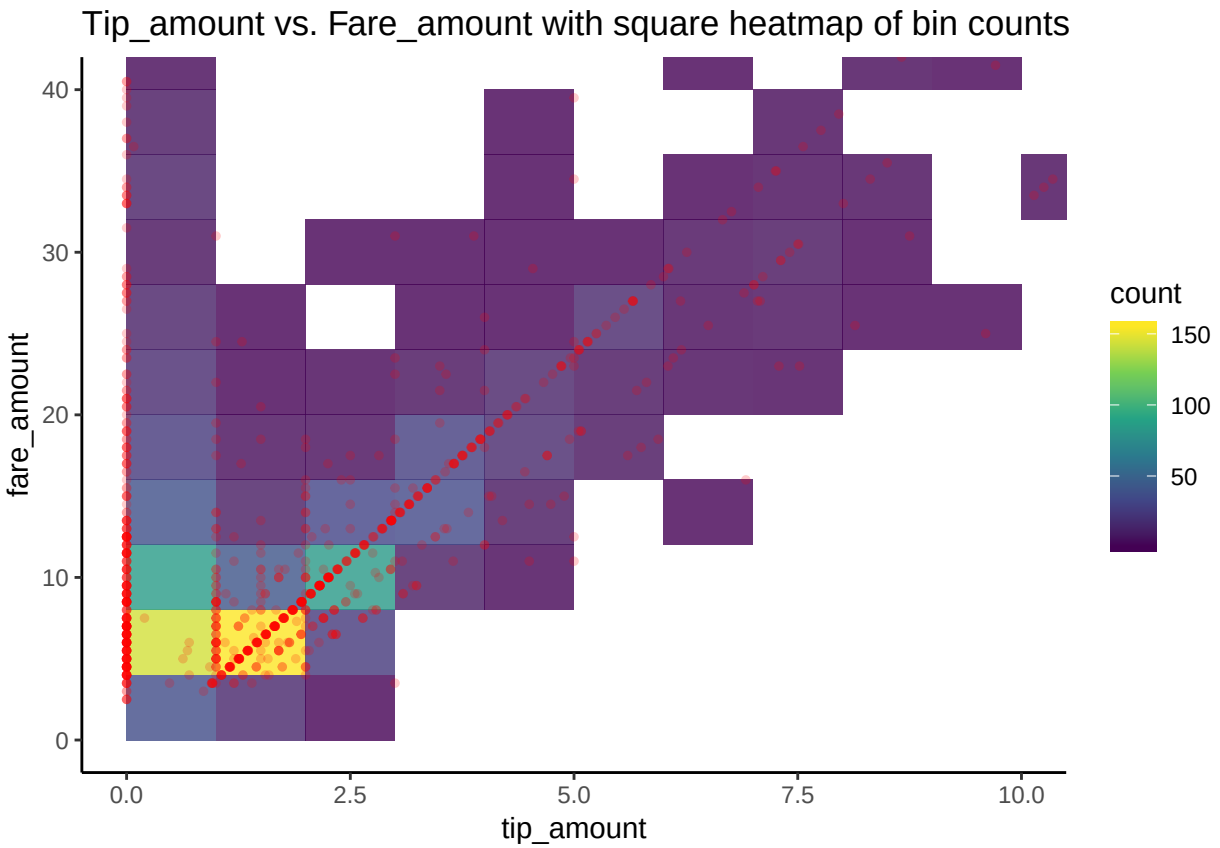
(c) Hexagonal heatmap of bin counts

```
p3c=ggplot(taxi_sample, aes(x=tip_amount, y=fare_amount)) +
  geom_hex(binwidth = c(1, 4),alpha=0.8) +
  geom_point(size = 1,alpha=0.2,col='red') +
  coord_cartesian(xlim=c(0,10),ylim=c(0,40)) +
  scale_fill_viridis_c() +
  theme_classic() +
  ggtitle('Tip_amount vs. Fare_amount with hexagonal heatmap of bin counts')
p3c
```



(d) Square heatmap of bin counts

```
p3d=ggplot(taxi_sample, aes(x=tip_amount, y=fare_amount)) +
  geom_bin2d(binwidth = c(1, 4),alpha=0.8) +
  geom_point(size = 1,alpha=0.2,col='red') +
  coord_cartesian(xlim=c(0,10),ylim=c(0,40)) +
  scale_fill_viridis_c() +
  theme_classic() +
  ggtitle('Tip_amount vs. Fare_amount with square heatmap of bin counts')
p3d
```



For all, adjust parameters to the levels that provide the best views of the data.

(e) Describe noteworthy features of the data, using the "Movie ratings" example on page 82 (last page of Section 5.3) as a guide.

1. There are no trips with a low fare amount and a high tip amount.

2. Despite those 0 tips, tip amount increases with the amount of fare.

3. There seems to be a barrier around 40.

4. A few trip with a high amount of tips, over 7.5, look like outliers. They have a realtively lower fare amount than other trips with similar tips amount.

5. Trips with 0 tip may have any fare amount from the lowest to the highest.

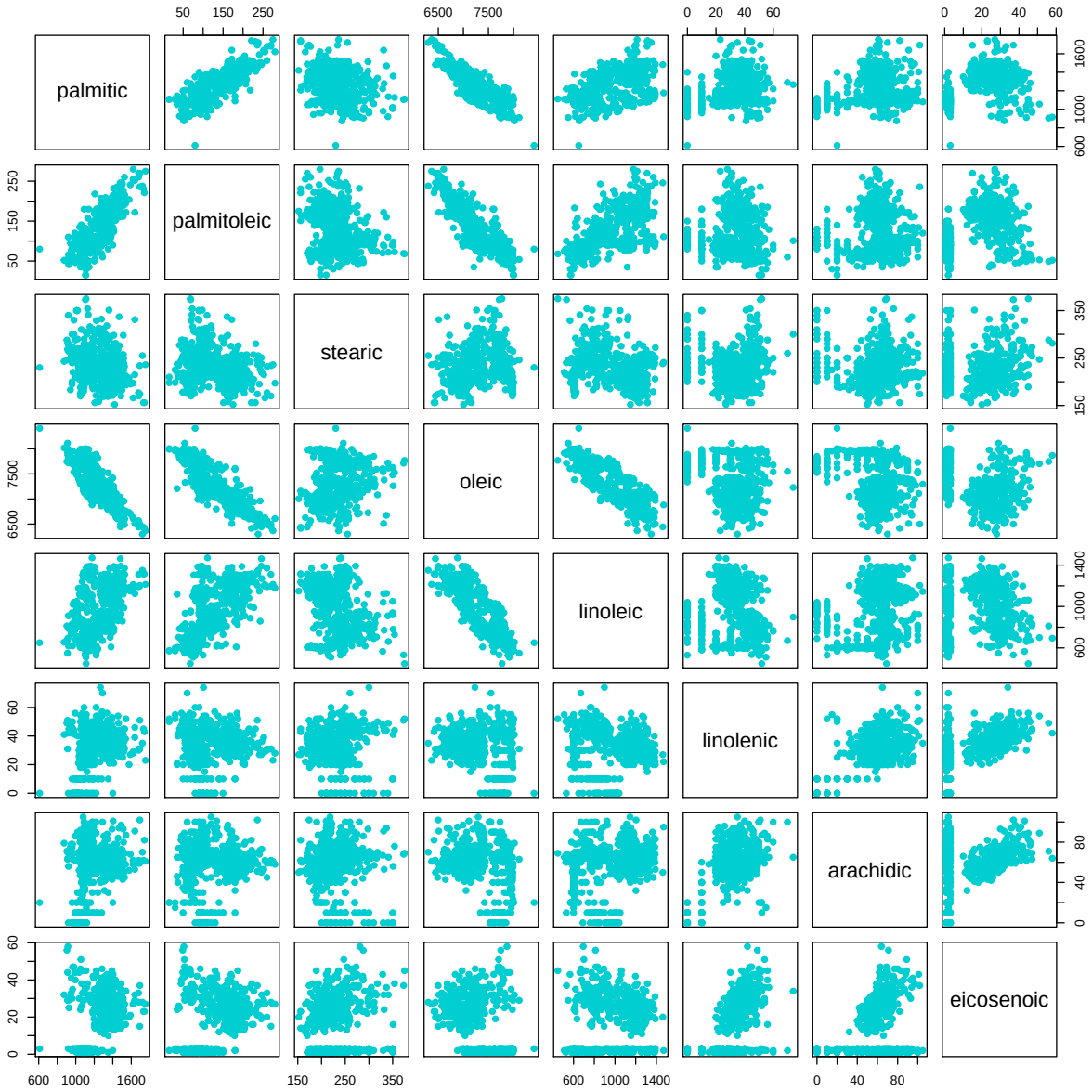
6. It seems like the only trip with very high tip amount are trips with relatively high fare amount.

4. Olive Oil

Data: `olives` dataset in **extracat** package

(a) Draw a scatterplot matrix of the eight continuous variables. Which pairs of variables are strongly positively associated and which are strongly negatively associated?

```
olives_df=tbl_df(select(olives,palmitic,palmitoleic,stearic,oleic,linoleic,linolenic,arachidic,eicoseno
plot(olives_df,col = 'darkturquoise', pch = 19)
```



Palmitic and palmitoleic are strongly positvely correlated.

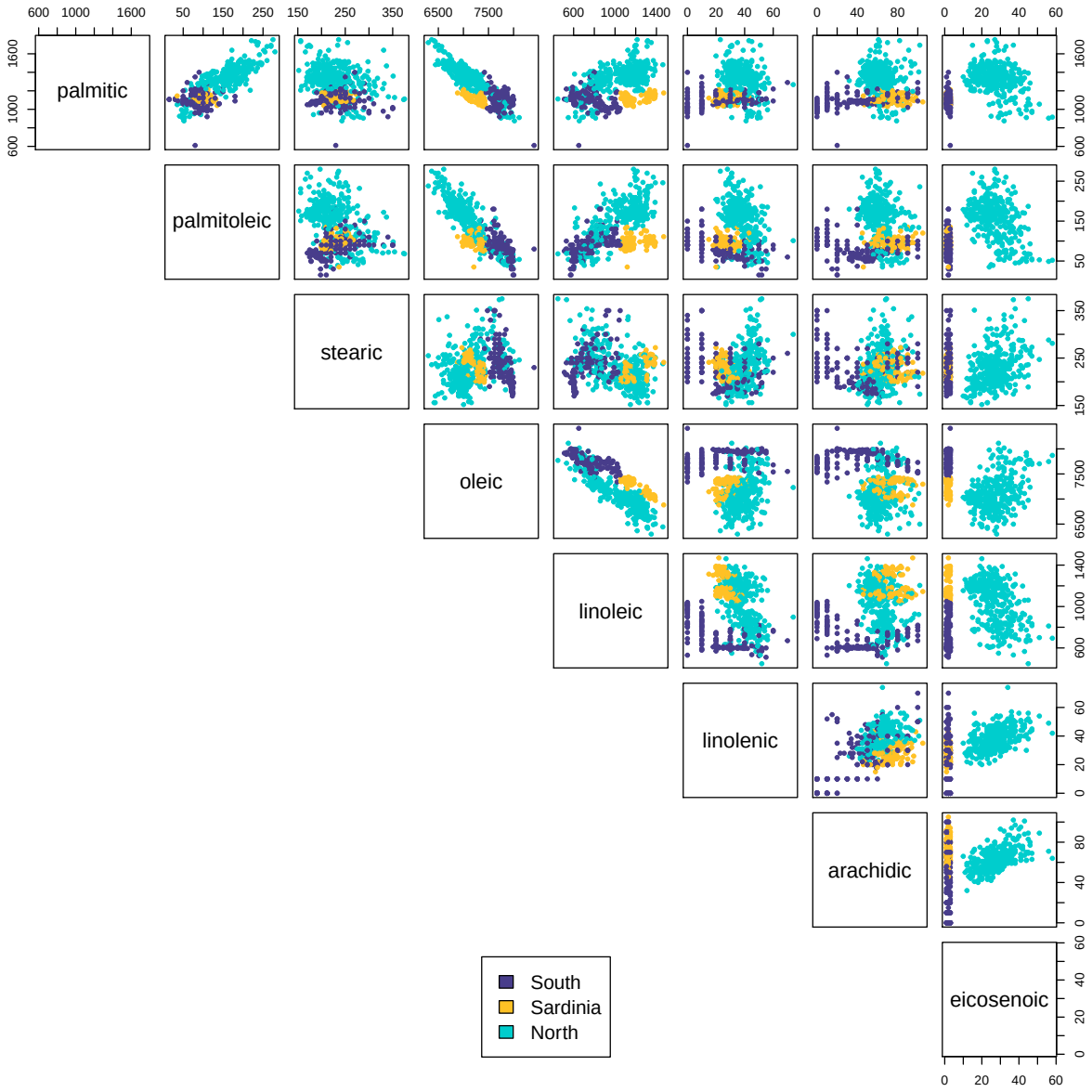
Palmitic and Oleic are strongly negatively correlated.

Palmitoleic and Oleic are strongly negatively correlated.

Oleic and linoleic are strongly negatively correlated.

(b) Color the points by region. What do you observe?

```
mycol=c('darkslateblue','goldenrod1','cyan3')
plot(olives_df,col=mycol[olives$Region],pch=19,cex=0.7,lower.panel=NULL)
par(xpd=TRUE)
legend('bottom',legend=as.vector(unique(olives$Region)),fill=mycol)
```



Palmitic and palmitoleic are strongly positvely correlated in North region but weak correlated in South and Sardinia

Palmitic and Oleic are strongly negatively correlated in North and relatively weaker negatively correlated in Sardinia. It seems like that they are uncorrelated in South

Palmitoleic and Oleic are strongly negatively correlated in North and relatively weaker negatively correlated in South and Sardinia.

Oleic and linoleic are strongly negatively correlated in all three regions.

Overall, the data in the North dominated the ones in other two regions

In the most situations, the regions can be clearly seperated by color in this scatterplot. The olives from Sardinia and South have a more compact scatter range. The ones from the North have a more spread range. We can easily classify the olives from Sardinia and South becasue

they almost don't overlap. And Of all eight variables, Eicosenoic is the one to best seperate olives from the North.

5. Wine

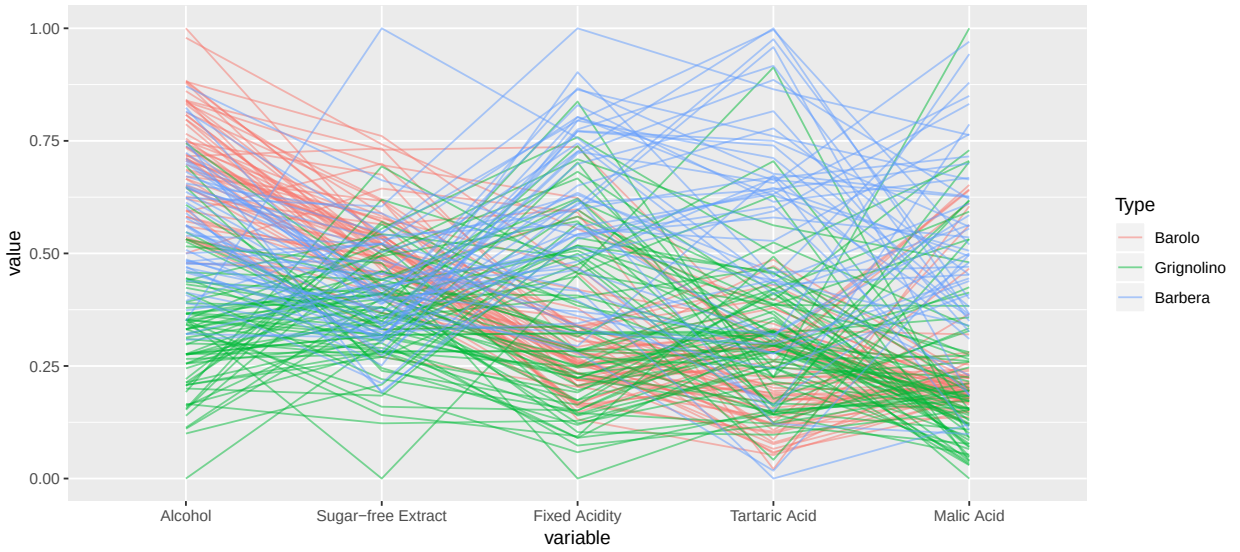
Data: `wine` dataset in **pgmm** package

(Recode the `Type` variable to descriptive names.)

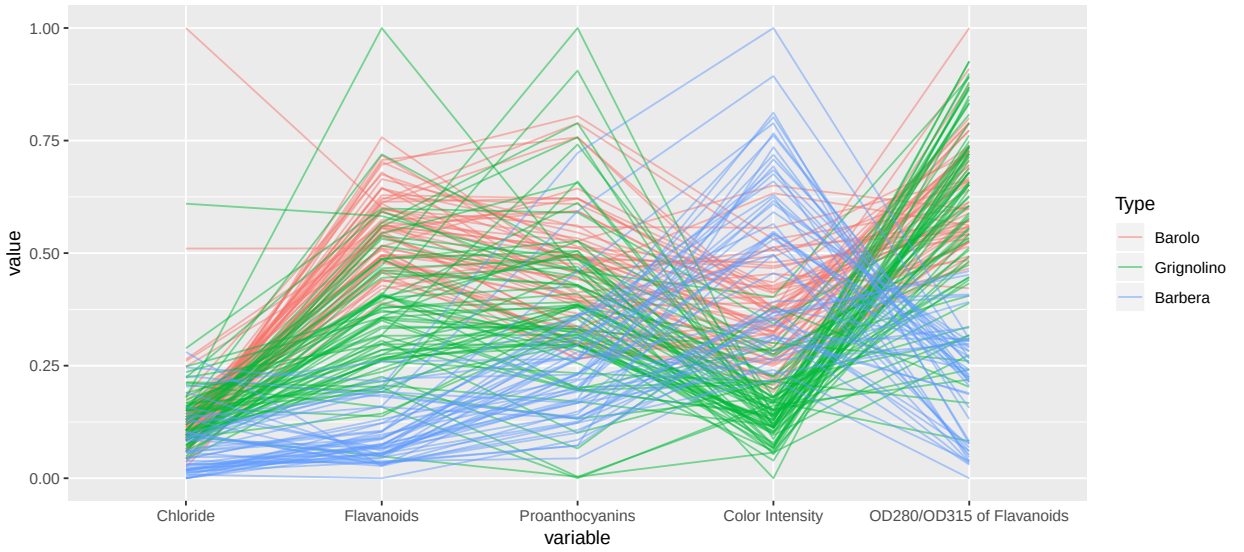
(a) Use parallel coordinate plots to explore how the variables separate the wines by `Type`. Present the version that you find to be most informative. You do not need to include all of the variables.

```
data(wine)
wine_df=wine
wine_df$Type=as.factor(wine_df$Type)
levels(wine_df$Type)=c('Barolo','Grignolino','Barbera')

ggparcoord(wine_df,column=2:6,groupColumn =  'Type' ,scale = "uniminmax",alphaLines = 0.5)
```




```
ggparcoord(wine_df,column=c(15,17,19,20,23),groupColumn =  'Type' ,scale = "uniminmax",alphaLines = 0.5)
```



(b) Explain what you discovered.

As can be seen in the first graph. I select the variables: Alcohol, Sugar-free Extract, Fixed Acidity, Tartaric Acid, Malic Acid. In the graph, we can observe Barolo is higher than Grignolino and Barbera on average in alcohol but it is relatively lower than other two type wine in Tartaric Acid and Malic Acid. Grignolino varies a lot in fixed Acidity and it is in the middle place about sugar-free extract and tartaric acid. It also has the relatively lowest Alcohol and Malic Acid among all three types. Barbera has higher Fixed Acidity and Tartaric Acid on average. It is in the middle place of alcohol.

As can be seen in the second graph. I select the variables: Chloride, Flavanoids, Proanthocyanins, Color Intensity and OD280/315. Barolo has similar Chloride as Grignolino and relatively higher flavanoids. It is in the middle place of color intensity and OD280/315. Grignolino is in the middle place of flavanoids. It is the lowest in color intensity but highest in OD280/315. We can also observe that some extremely large value in flavanoids and Proanthocyanins. Barbera has lowest level in chloride, flavanoids, Proanthocyanins and OD280/315 but it is higher than other two type wine in color intensity.

Overally speaking, I find that Alcohol can clearly seperate Barolo and Grignolino. Fixed Acidity can clearly seperate Grignolino and Barbera. Flavanoids can clearly seperate all three types wine. Color Intensity and OD280/315 can seperate Grignolino and Barbera.