Assignment 4

Student Name: Koki Sasagawa
Student ID: 999054646

Assignment 4: Rice SNPs and GWAS

This should be a knittable .Rmd file. Include this file, a .pdf and a .html when you turn in the assignment.

Practice New functions:

```
#Import data into object of current environment
fruit.size <- read.csv("./RiceSNPData/fruit_size.csv")
fruit.color <- read.csv("./RiceSNPData/fruit_color.csv")
fruit.shape <- read.csv("./RiceSNPData/fruit_shape.csv")

#Display Contents
fruit.size
```

```
##         name        size
## 1 watermelon extra large
## 2       pear      medium
## 3      peach      medium
## 4  nectarine      medium
## 5      grape       small
## 6   eggplant       large
## 7  blueberry  very small
## 8     banana      medium
## 9      apple      medium
```

```
fruit.color
```

```
##       fruit  color
## 1     apple    red
## 2    banana yellow
## 3 blueberry   blue
## 4  eggplant purple
## 5     grape  green
## 6     peach yellow
## 7    tomato    red
```

```
fruit.shape
```

```
##       fruit  shape
## 1    banana oblong
## 2     peach  round
## 3     apple  round
## 4     grape oblong
## 5 blueberry  round
## 6    tomato  round
## 7 nectarine  round
## 8  eggplant oblong
## 9      pear  other
```

```r
#Use the merge function to combine the different data frames into a single one
fruit.color.shape <- merge(fruit.color, fruit.shape, by="fruit")

#Display Contents
fruit.color.shape
```

```
##       fruit  color  shape
## 1     apple    red  round
## 2    banana yellow oblong
## 3 blueberry   blue  round
## 4  eggplant purple oblong
## 5     grape  green oblong
## 6     peach yellow  round
## 7    tomato    red  round
```

PRACTICE 1:

a. What does "by" do? In the command above (hint: look at the help page for merge())

> by specifies the column used for merging.

b. Why are there only seven rows in the merged data set even though fruit.shape had nine? Read the help page for merge() to figure out how to keep all of the data in the original fruit.shape data sheet.

> There are only seven rows instead of 9 because fruit.color only had 7 fruit entries. Since we are merging the two data frames by the column fruit, the 2 fruits in fruit.shapes are left out because it is not included in fruit.color.

```r
#Use the merge function to combine and keep all the rows.
fruit.color.shape <- merge(fruit.color, fruit.shape, by = "fruit", all = T)

#Display Contents
fruit.color.shape
```

```
##       fruit  color  shape
## 1     apple    red  round
## 2    banana yellow oblong
## 3 blueberry   blue  round
## 4  eggplant purple oblong
## 5     grape  green oblong
## 6     peach yellow  round
## 7    tomato    red  round
## 8 nectarine   <NA>  round
## 9      pear   <NA>  other
```

c. Merge fruit.size with fruit.color.shape, keeping all of the rows from each original sheet. Place the merged dataframe in fruit.all. Note that the column that you want to merge on for fruit size has a different name. Read help on merge() to figure out how to deal with this.

```r
#Rename the column from name to fruit using colnames
colnames(fruit.size) <- c("fruit", "size")
fruit.all <- merge(fruit.size, fruit.color.shape, by = "fruit", all = T)

#Display Content
fruit.all
```

```
##         fruit        size  color  shape
## 1       apple      medium    red  round
## 2      banana      medium yellow oblong
```

```
## 3   blueberry  very small   blue  round
## 4    eggplant       large purple oblong
## 5       grape       small  green oblong
## 6   nectarine      medium   <NA>  round
## 7       peach      medium yellow  round
## 8        pear      medium   <NA>  other
## 9  watermelon extra large   <NA>   <NA>
## 10     tomato        <NA>    red  round
```

Sorting and ordering data:

We can use the `sort()` function to sort any single vector of data.

```
sort(fruit.shape$fruit)
```

```
## [1] apple     banana    blueberry eggplant  grape     nectarine peach
## [8] pear      tomato
## 9 Levels: apple banana blueberry eggplant grape nectarine peach ... tomato
```

```
#Sort reverse alphabetical order
sort(fruit.shape$fruit, decreasing = T)
```

```
## [1] tomato    pear      peach     nectarine grape     eggplant  blueberry
## [8] banana    apple
## 9 Levels: apple banana blueberry eggplant grape nectarine peach ... tomato
```

We can use the `order()` to tell us how we could reorder the items to obtain a sorted list.

```
order(fruit.shape$fruit)
```

```
## [1] 3 1 5 8 4 7 2 9 6
```

This tells us that the thrid item "apple" should be first, "banana" second, and so on.

PRACTICE 2:
reorder fruit.all so that the whole data.frame is sorted by fruit shape. Include the code:

```
order(fruit.all$shape)
```

```
##  [1]  2  4  5  8  1  3  6  7 10  9
```

```
sort(fruit.all$shape)
```

```
## [1] oblong oblong oblong other  round  round  round  round  round
## Levels: oblong other round
```

```
#Using [], order the fruit.all by shape
fruit.all[order(fruit.all$shape),]
```

```
##         fruit        size  color  shape
## 2      banana      medium yellow oblong
## 4    eggplant       large purple oblong
## 5       grape       small  green oblong
## 8        pear      medium   <NA>  other
## 1       apple      medium    red  round
## 3   blueberry  very small   blue  round
## 6   nectarine      medium   <NA>  round
## 7       peach      medium yellow  round
## 10     tomato        <NA>    red  round
## 9  watermelon extra large   <NA>   <NA>
```

PRACTICE 3:
Re-order fruit.all so that the whole data.frame is sorted by fruit size, then by fruit shape. Include the code. (hint: look at help for order) Your output should look like:

```
fruit.all[order(fruit.all$size, fruit.all$shape),]
```

```
##         fruit        size  color  shape
## 9  watermelon extra large   <NA>   <NA>
## 4    eggplant       large purple oblong
## 2      banana      medium yellow oblong
## 8        pear      medium   <NA>  other
## 1       apple      medium    red  round
## 6   nectarine      medium   <NA>  round
## 7       peach      medium yellow  round
## 5       grape       small  green oblong
## 3   blueberry  very small   blue  round
## 10     tomato        <NA>    red  round
```

Reshaping data:
In a long format, each row represents a single observation. The reshape library has the `melt()` function to covert wide to long.

```
library(reshape2)
fruit.all
```

```
##         fruit        size  color  shape
## 1       apple      medium    red  round
## 2      banana      medium yellow oblong
## 3   blueberry  very small   blue  round
## 4    eggplant       large purple oblong
## 5       grape       small  green oblong
## 6   nectarine      medium   <NA>  round
## 7       peach      medium yellow  round
## 8        pear      medium   <NA>  other
## 9  watermelon extra large   <NA>   <NA>
## 10     tomato        <NA>    red  round
```

```
#id.var allows to specify which column holds the identification information. meas.far can allow specifi
fruit.all.melt <- melt(fruit.all, id.var="fruit")
```

```
## Warning: attributes are not identical across measure variables; they will
## be dropped
```

```
fruit.all.melt
```

```
##         fruit variable       value
## 1       apple     size      medium
## 2      banana     size      medium
## 3   blueberry     size  very small
## 4    eggplant     size       large
## 5       grape     size       small
## 6   nectarine     size      medium
## 7       peach     size      medium
## 8        pear     size      medium
## 9  watermelon     size extra large
## 10     tomato     size        <NA>
## 11      apple    color         red
## 12     banana    color      yellow
```

```
## 13  blueberry    color        blue
## 14   eggplant    color      purple
## 15      grape    color       green
## 16  nectarine    color        <NA>
## 17      peach    color      yellow
## 18       pear    color        <NA>
## 19 watermelon    color        <NA>
## 20     tomato    color         red
## 21      apple    shape       round
## 22     banana    shape      oblong
## 23  blueberry    shape       round
## 24   eggplant    shape      oblong
## 25      grape    shape      oblong
## 26  nectarine    shape       round
## 27      peach    shape       round
## 28       pear    shape       other
## 29 watermelon    shape        <NA>
## 30     tomato    shape       round
```

Applying funcions across rows or columns:
`apply()` takes at least 3 arguments where X is a data frame or matrix, MARGIN is whether to apply a function to each row (1) or each column (2), FUN is the function that you want to use.

```
m <- matrix(rnorm(24),ncol=6)
m
```

```
##             [,1]        [,2]        [,3]        [,4]       [,5]
## [1,]  0.08795168  0.07025305  0.06558948  0.24108642 -1.0631634
## [2,]  1.01810841  0.40553789 -1.41189656  1.28105094  1.8335855
## [3,] -0.78299942 -0.90526100  0.51730461  2.23442712  0.6462911
## [4,] -1.51966734 -1.30544064  1.60263877 -0.06472999 -0.6034798
##             [,6]
## [1,] -0.29775644
## [2,] -1.16351468
## [3,] -2.13871922
## [4,]  0.06807443
```

```
apply(m,1,min)
```

```
## [1] -1.063163 -1.411897 -2.138719 -1.519667
```

PRACTICE 4
Find the mean of each column of m

```
apply(m,1,mean)
```

```
## [1] -0.1493399  0.3271453 -0.0714928 -0.3037674
```

Lets get started with the real data

```
data.geno <- read.csv("./RiceSNPData/Rice_44K_genotypes.csv.gz",
                      row.names=1, #this tells R to use the first column as row names
                      na.strings=c("NA","00")) #this tells R that missing data is denoted as "NA" or "0
```

```
head(data.geno[,1:20])
```

```
##        X1_13147 X1_73192 X1_74969 X1_75852 X1_75953 X1_91016 X1_146625
```

```
## NSFTV1       TT       TT       CC       GG       TT       AA       CC
## NSFTV3       CC       CC       CC       AA       GG     <NA>       CC
## NSFTV4       CC       CC       CC       AA       GG       GG       CC
## NSFTV5       CC       CC       TT       GG       GG       AA       TT
## NSFTV6       CC       CC       CC       AA       GG       GG       CC
## NSFTV7       TT       TT       CC       GG       TT       AA       CC
##        X1_149005 X1_149754 X1_151492 X1_152899 X1_172755 X1_172923
## NSFTV1        TT        AA        AA        GG        CC        CC
## NSFTV3        GG        TT        GG        GG        CC        CC
## NSFTV4        GG        TT        GG        GG        CC        CC
## NSFTV5        GG        TT        AA      <NA>        CC        CC
## NSFTV6        GG        TT        GG        GG        CC        CC
## NSFTV7        TT        AA        AA        GG        CC        TT
##        X1_173692 X1_195327 X1_199011 X1_202999 X1_203126 X1_205867
## NSFTV1        AA        TT        TT        TT        AA        AA
## NSFTV3        TT        CC        TT        TT        CC        GG
## NSFTV4        TT        CC        TT        TT        CC        GG
## NSFTV5        AA        CC        TT        AA        CC        GG
## NSFTV6        TT        CC        TT        TT        CC        GG
## NSFTV7        AA        TT        TT        TT        AA        AA
##        X1_212693
## NSFTV1        GG
## NSFTV3        TT
## NSFTV4        TT
## NSFTV5        GG
## NSFTV6        TT
## NSFTV7        GG
```

```
summary(data.geno[,1:20])
```

```
##  X1_13147   X1_73192   X1_74969   X1_75852 X1_75953   X1_91016
##  CC  :124   CC  :125   CC  :349   AA: 58   GG  :123   AA  :312
##  TT  :288   TT  :287   TT  : 63   GG:355   TT  :288   GG  : 34
##  NA's:  1   NA's:  1   NA's:  1            NA's:  2   NA's: 67
##  X1_146625  X1_149005  X1_149754  X1_151492  X1_152899  X1_172755
##  CC  :349   GG  :123   AA  :287   AA  :352   AA  : 38   CC  :396
##  TT  : 63   TT  :288   TT  :118   GG  : 59   GG  :344   TT  : 16
##  NA's:  1   NA's:  2   NA's:  8   NA's:  2   NA's: 31   NA's:  1
##  X1_172923  X1_173692  X1_195327  X1_199011 X1_202999  X1_203126
##  CC  :313   AA  :347   CC  :144   CC: 45    AA  : 61   AA  :256
##  TT  : 94   TT  : 54   TT  :254   TT:368    TT  :347   CC  :149
##  NA's:  6   NA's: 12   NA's: 15             NA's:  5   NA's:  8
##  X1_205867  X1_212693
##  AA  :259   GG  :322
##  GG  :152   TT  : 89
##  NA's:  2   NA's:  2
```

Create a data subset that contains a random sample of 2500 SNPs

```
data.geno.2500 <- sample(data.geno, 2500)
dim(data.geno.2500)
```

```
## [1]  413 2500
```

Create a MDS plot with our smaller subset

```
#convert the data matrix to numbers
geno.numeric <- data.matrix(data.geno.2500)
head(geno.numeric[,1:20])
```

```
##        X12_23632939 X3_6724521 X3_34531210 X6_7841654 X3_25847028
## NSFTV1            1          2           2          2           1
## NSFTV3           NA          1           2          2           1
## NSFTV4            1          1           2          2           1
## NSFTV5            1          1           2          2           2
## NSFTV6            1          1           2          2           1
## NSFTV7            2          2           2          2          NA
##        X11_9071157 X6_24875226 X6_17195755 X1_18047642 X10_13087617
## NSFTV1           1           1           2           1            1
## NSFTV3           1           1           1           1            1
## NSFTV4           2          NA          NA           1            1
## NSFTV5          NA           1           1           1            1
## NSFTV6           2          NA           1           1            1
## NSFTV7           2           1           1           1            1
##        X12_23124865 X2_34834496 X1_32992641 X7_2070838 X1_1848812
## NSFTV1            1           2           2          1          1
## NSFTV3            1           1           2          1          2
## NSFTV4            1           1           1          2          2
## NSFTV5            1           1           2          1          2
## NSFTV6            1           1           1          2          2
## NSFTV7            1           1           2          1          1
##        X7_21757895 X1_9119444 X12_18283876 X1_38733382 X4_20481570
## NSFTV1           2          2            1           1           2
## NSFTV3           1          1            2           1           1
## NSFTV4           1          1            2           1           1
## NSFTV5           2          2            1           1          NA
## NSFTV6          NA          1            2           1           1
## NSFTV7           2          2            1           1           2
```

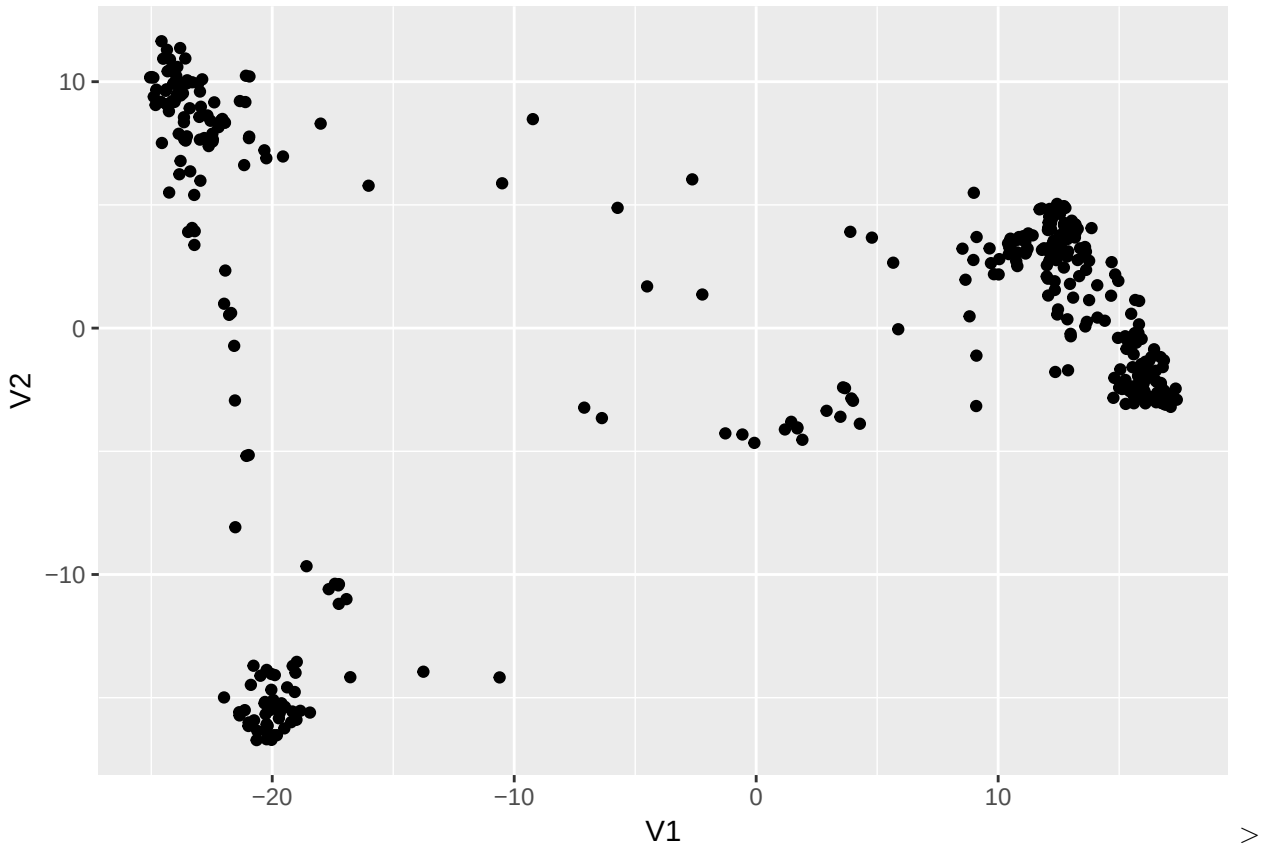
```
#calculate the Euclidian distance between each rice variety
genDist <- as.matrix(dist(geno.numeric))
```

```
#perform the multi-dimensional scaling
geno.mds <- as.data.frame(cmdscale(genDist))
head(geno.mds) #now we have 2 dimensions
```

```
##                V1         V2
## NSFTV1  16.724400  -2.800265
## NSFTV3 -24.367765   9.690134
## NSFTV4 -17.243869 -10.390602
## NSFTV5   2.909696  -3.357328
## NSFTV6 -16.932450 -10.999276
## NSFTV7  10.620649   3.513382
```

EXERCISE 1: Is there any evidence for populations structure (different sub populations)? If so, how many sub populations do you think the MDS plot reveals? What do you make of the individuals that are between the major groups? (Include your plot and code)

```
library(ggplot2)

riceMDS <- ggplot(geno.mds, aes(x = V1, y = V2))
riceMDS + geom_point()
```



The MDS graph shows 3 groups of SNPs, thus suggesting there are 3 sub populations. Each of these dots represents a rice strain and we are comparing 2500 random SNPs (reduction from the original 44000). The rice strains that are grouped together thus share similar genotypes of SNPs. The individuals that are in between the major groups are strains of rice are variants that do not share the same genotype of SNPs as the 3 sub populations. These rice strains are perhaps mutant strains that have gained a change in SNP genotype.

EXERCISE 2:

- Use the `read.csv()` `head()` and `summary()` functions that you learned earlier to import and look at this file. Import the file into an object called "data.pheno".
- Use merge() to merge the MDS scaled genotype data with the phenotype data. Here the column that we are merging on is the "row.name" column. So you can use `by="row.names"` or `by=1` in your call to merge. Use summary and head to look at the new object and make sure that it is as you expect.
- Include your code in the .Rmd

```
data.pheno <- read.csv("./RiceSNPData/RiceDiversity.44K.MSU6.Phenotypes.csv",row.names=1,na.strings=c("

data.geno.pheno <- merge(geno.mds, data.pheno, by=0, All = T)

head(data.geno.pheno)
```

```
##   Row.names        V1         V2     Accession_Name Country_of_Origin
## 1    NSFTV1  16.72440 -2.8002646           Agostano             Italy
## 2   NSFTV10  16.57548 -2.6215418 Baghlani Nangarhar       Afghanistan
## 3  NSFTV100  15.49797  0.5808688           Lacrosse     United States
## 4  NSFTV101  10.86102  3.6953413             Lemont     United States
## 5  NSFTV102 -23.53711  9.9287307               <NA>              <NA>
## 6  NSFTV103  15.64048 -2.3548656         Luk Takhar       Afghanistan
##     Region Alu.Tol Flowering.time.at.Arkansas Flowering.time.at.Faridpur
```

```
## 1   Europe   0.730                  75.08333                        64
## 2 Mid East   0.902                  89.00000                        55
## 3  America   0.800                  84.11111                        78
## 4  America   0.630                  86.16667                        79
## 5     <NA>   0.440                        NA                        NA
## 6 Mid East   0.550                  84.00000                        78
##   Flowering.time.at.Aberdeen FT.ratio.of.Arkansas.Aberdeen
## 1                         81                     0.9269547
## 2                         74                     1.2027027
## 3                        122                     0.6894353
## 4                         88                     0.9791667
## 5                        165                            NA
## 6                        108                     0.7777778
##   FT.ratio.of.Faridpur.Aberdeen Culm.habit Leaf.pubescence
## 1                     0.7901235   4.000000               1
## 2                     0.7432432   3.000000              NA
## 3                     0.6393443   1.666667               0
## 4                     0.8977273   3.000000               0
## 5                            NA   3.000000              NA
## 6                     0.7222222   2.500000               1
##   Flag.leaf.length Flag.leaf.width Awn.presence Panicle.number.per.plant
## 1         28.37500        1.283333            0                 3.068053
## 2         27.90000        1.000000            1                 3.650658
## 3         27.62222        1.611111            0                 2.978925
## 4         27.62500        1.450000            0                 2.818398
## 5         27.85000        1.100000            1                 3.481240
## 6         30.17500        1.050000            0                 2.957511
##   Plant.height Panicle.length Primary.panicle.branch.number
## 1    110.91667       20.48182                      9.272727
## 2     83.00000       22.16667                     10.333333
## 3    114.88889       23.94444                     11.555556
## 4     86.16667       27.45000                     11.083333
## 5    105.50000       30.75000                     10.500000
## 6     95.08333       24.13333                      9.777778
##   Seed.number.per.panicle Florets.per.panicle Panicle.fertility
## 1                4.785975            4.914658             0.879
## 2                4.110874            4.733270             0.537
## 3                5.032614            5.340738             0.735
## 4                4.894101            5.209031             0.730
## 5                4.600158            4.867534             0.765
## 6                4.881538            4.998900             0.889
##   Seed.length Seed.width Seed.volume Seed.surface.area
## 1    8.064117   3.685183    2.587448          3.914120
## 2    7.859000   3.233250    2.265361          3.729249
## 3    8.138033   3.382633    2.440978          3.850531
## 4    9.632392   2.644467    2.121810          3.778742
## 5    9.805500   2.469600    2.030632          3.750804
## 6    7.528083   3.534583    2.404007          3.784157
##   Brown.rice.seed.length Brown.rice.seed.width Brown.rice.surface.area
## 1               5.794542              3.113958                3.511152
## 2               5.088267              2.937733                3.309970
## 3               5.944467              2.893033                3.467780
## 4               7.147025              2.251503                3.388738
## 5               7.072600              2.044100                3.301659
```

```
## 6              5.298917             3.048408                3.390811
##   Brown.rice.volume Seed.length.width.ratio Brown.rice.length.width.ratio
## 1          7.737358                  2.188                          1.861
## 2          5.912900                  2.431                          1.732
## 3          6.898900                  2.406                          2.055
## 4          4.919557                  3.642                          3.174
## 5          4.130800                  3.970                          3.460
## 6          6.670092                  2.130                          1.738
##   Seed.color Pericarp.color Straighthead.suseptability Blast.resistance
## 1      light          light                   4.833333                8
## 2      light          light                   3.330000                2
## 3      light          light                   6.501667                3
## 4      light          light                   3.331667                8
## 5      light          light                   6.835000                1
## 6      light          light                   5.335000                4
##   Amylose.content Alkali.spreading.value Protein.content
## 1        15.61333               6.083333            8.45
## 2        15.09667               7.000000            9.50
## 3        10.60333               6.958333            8.70
## 4        20.51333               6.000000            9.50
## 5        21.25333               5.000000            8.70
## 6        16.97667               6.916667            7.75
```

```
summary(data.geno.pheno)
```

```
##   Row.names                V1                V2
##  Length:413         Min.   :-25.05   Min.   :-16.720
##  Class :AsIs        1st Qu.:-20.32   1st Qu.: -2.848
##  Mode  :character   Median : 11.16   Median :  1.236
##                     Mean   :  0.00   Mean   :  0.000
##                     3rd Qu.: 15.05   3rd Qu.:  4.524
##                     Max.   : 17.38   Max.   : 11.642
##
##        Accession_Name     Country_of_Origin     Region      Alu.Tol
##  Azucena      :  2    United States: 41     America:83   Min.   :0.0300
##  Carolina Gold:  2    China        : 28     E Asia :73   1st Qu.:0.4000
##  Dom Sufid    :  2    India        : 27     S Asia :64   Median :0.6035
##  M-202        :  2    Bangladesh   : 24     Africa :39   Mean   :0.5874
##  Moroberekan  :  2    Japan        : 17     Europe :34   3rd Qu.:0.7500
##  (Other)      :373    (Other)      :246     (Other):78   Max.   :1.3500
##  NA's         : 30    NA's         : 30     NA's   :42   NA's   :43
##  Flowering.time.at.Arkansas Flowering.time.at.Faridpur
##  Min.   : 54.50             Min.   : 39.00
##  1st Qu.: 79.75             1st Qu.: 66.00
##  Median : 87.71             Median : 74.00
##  Mean   : 87.94             Mean   : 71.77
##  3rd Qu.: 96.83             3rd Qu.: 78.00
##  Max.   :150.50             Max.   :110.00
##  NA's   :39                 NA's   :108
##  Flowering.time.at.Aberdeen FT.ratio.of.Arkansas.Aberdeen
##  Min.   : 45.0              Min.   :0.3724
##  1st Qu.: 81.5              1st Qu.:0.7975
##  Median : 99.0              Median :0.8960
##  Mean   :107.1              Mean   :0.8949
##  3rd Qu.:114.0              3rd Qu.:1.0195
```

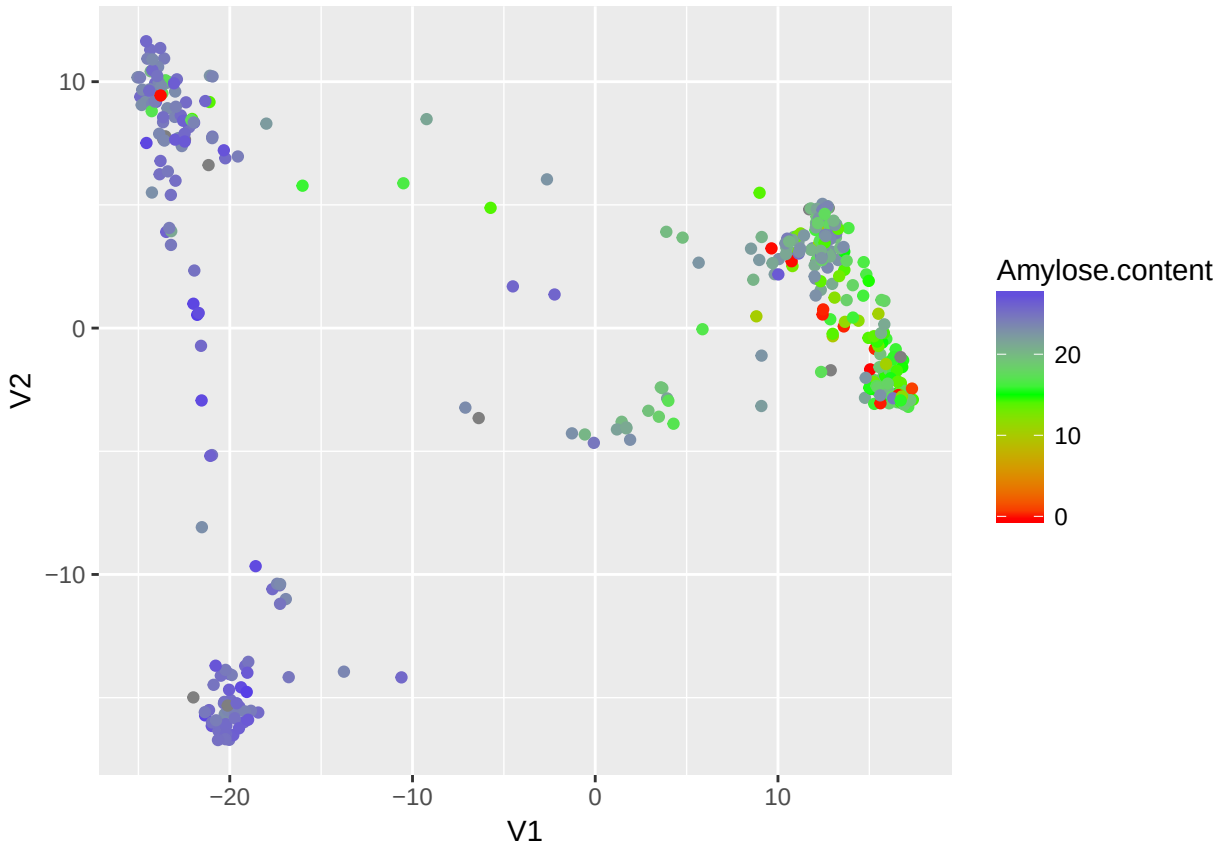
```
##  Max.   :306.0              Max.   :1.7031
##  NA's   :54                 NA's   :64
##  FT.ratio.of.Faridpur.Aberdeen   Culm.habit    Leaf.pubescence
##  Min.   :0.3459                Min.   :1.000   Min.   :0.0000
##  1st Qu.:0.6838                1st Qu.:3.000   1st Qu.:1.0000
##  Median :0.7720                Median :4.000   Median :1.0000
##  Mean   :0.7711                Mean   :4.228   Mean   :0.8507
##  3rd Qu.:0.8671                3rd Qu.:5.500   3rd Qu.:1.0000
##  Max.   :1.2885                Max.   :9.000   Max.   :1.0000
##  NA's   :109                   NA's   :29      NA's   :125
##  Flag.leaf.length Flag.leaf.width   Awn.presence
##  Min.   :15.42    Min.   :0.5917   Min.   :0.0000
##  1st Qu.:26.62    1st Qu.:1.0400   1st Qu.:0.0000
##  Median :30.05    Median :1.1833   Median :0.0000
##  Mean   :30.63    Mean   :1.2217   Mean   :0.2222
##  3rd Qu.:34.55    3rd Qu.:1.4111   3rd Qu.:0.0000
##  Max.   :49.44    Max.   :1.8917   Max.   :1.0000
##  NA's   :36       NA's   :36       NA's   :44
##  Panicle.number.per.plant  Plant.height    Panicle.length
##  Min.   :2.234            Min.   : 67.75   Min.   :15.63
##  1st Qu.:2.931            1st Qu.: 99.75   1st Qu.:22.24
##  Median :3.242            Median :117.50   Median :24.19
##  Mean   :3.247            Mean   :116.58   Mean   :24.37
##  3rd Qu.:3.558            3rd Qu.:131.39   3rd Qu.:26.45
##  Max.   :4.172            Max.   :194.33   Max.   :35.68
##  NA's   :41               NA's   :30       NA's   :38
##  Primary.panicle.branch.number Seed.number.per.panicle Florets.per.panicle
##  Min.   : 5.556                Min.   :3.445           Min.   :3.909
##  1st Qu.: 8.667                1st Qu.:4.679           1st Qu.:4.879
##  Median : 9.917                Median :4.888           Median :5.065
##  Mean   : 9.943                Mean   :4.854           Mean   :5.056
##  3rd Qu.:11.111                3rd Qu.:5.054           3rd Qu.:5.258
##  Max.   :17.000                Max.   :5.635           Max.   :5.836
##  NA's   :38                    NA's   :37              NA's   :36
##  Panicle.fertility  Seed.length       Seed.width     Seed.volume
##  Min.   :0.3720    Min.   : 5.894   Min.   :2.196   Min.   :1.669
##  1st Qu.:0.7830    1st Qu.: 7.680   1st Qu.:2.819   1st Qu.:2.118
##  Median :0.8530    Median : 8.319   Median :3.157   Median :2.280
##  Mean   :0.8240    Mean   : 8.400   Mean   :3.117   Mean   :2.278
##  3rd Qu.:0.8932    3rd Qu.: 9.118   3rd Qu.:3.398   3rd Qu.:2.454
##  Max.   :0.9800    Max.   :12.549   Max.   :3.990   Max.   :2.872
##  NA's   :37        NA's   :36       NA's   :36      NA's   :36
##  Seed.surface.area Brown.rice.seed.length Brown.rice.seed.width
##  Min.   :3.434     Min.   :4.093          Min.   :1.820
##  1st Qu.:3.679     1st Qu.:5.501          1st Qu.:2.388
##  Median :3.764     Median :5.999          Median :2.655
##  Mean   :3.781     Mean   :6.117          Mean   :2.627
##  3rd Qu.:3.883     3rd Qu.:6.753          3rd Qu.:2.876
##  Max.   :4.198     Max.   :8.784          Max.   :3.358
##  NA's   :36        NA's   :36             NA's   :36
##  Brown.rice.surface.area Brown.rice.volume Seed.length.width.ratio
##  Min.   :2.959           Min.   :2.841     Min.   :1.799
##  1st Qu.:3.284           1st Qu.:4.806     1st Qu.:2.327
##  Median :3.363           Median :5.650     Median :2.651
```

```
##  Mean   :3.378           Mean   :5.806     Mean   :2.752
##  3rd Qu.:3.471           3rd Qu.:6.717     3rd Qu.:3.068
##  Max.   :3.766           Max.   :9.861     Max.   :4.467
##  NA's   :36              NA's   :36        NA's   :36
##  Brown.rice.length.width.ratio Seed.color  Pericarp.color
##  Min.   :1.502                 dark :  8   dark : 34
##  1st Qu.:1.997                 light:369   light:343
##  Median :2.287                 NA's : 36   NA's : 36
##  Mean   :2.382
##  3rd Qu.:2.678
##  Max.   :3.878
##  NA's   :36
##  Straighthead.suseptability Blast.resistance Amylose.content
##  Min.   :2.330              Min.   :0.000    Min.   : 0.00
##  1st Qu.:5.998              1st Qu.:2.000    1st Qu.:16.70
##  Median :7.165              Median :5.000    Median :21.13
##  Mean   :6.936              Mean   :5.039    Mean   :19.88
##  3rd Qu.:8.167              3rd Qu.:8.000    3rd Qu.:23.97
##  Max.   :9.000              Max.   :9.000    Max.   :27.96
##  NA's   :73                 NA's   :28       NA's   :12
##  Alkali.spreading.value Protein.content
##  Min.   :2.000          Min.   : 6.500
##  1st Qu.:5.403          1st Qu.: 7.950
##  Median :6.083          Median : 8.450
##  Mean   :5.974          Mean   : 8.593
##  3rd Qu.:6.938          3rd Qu.: 9.050
##  Max.   :7.000          Max.   :14.100
##  NA's   :10             NA's   :20
```

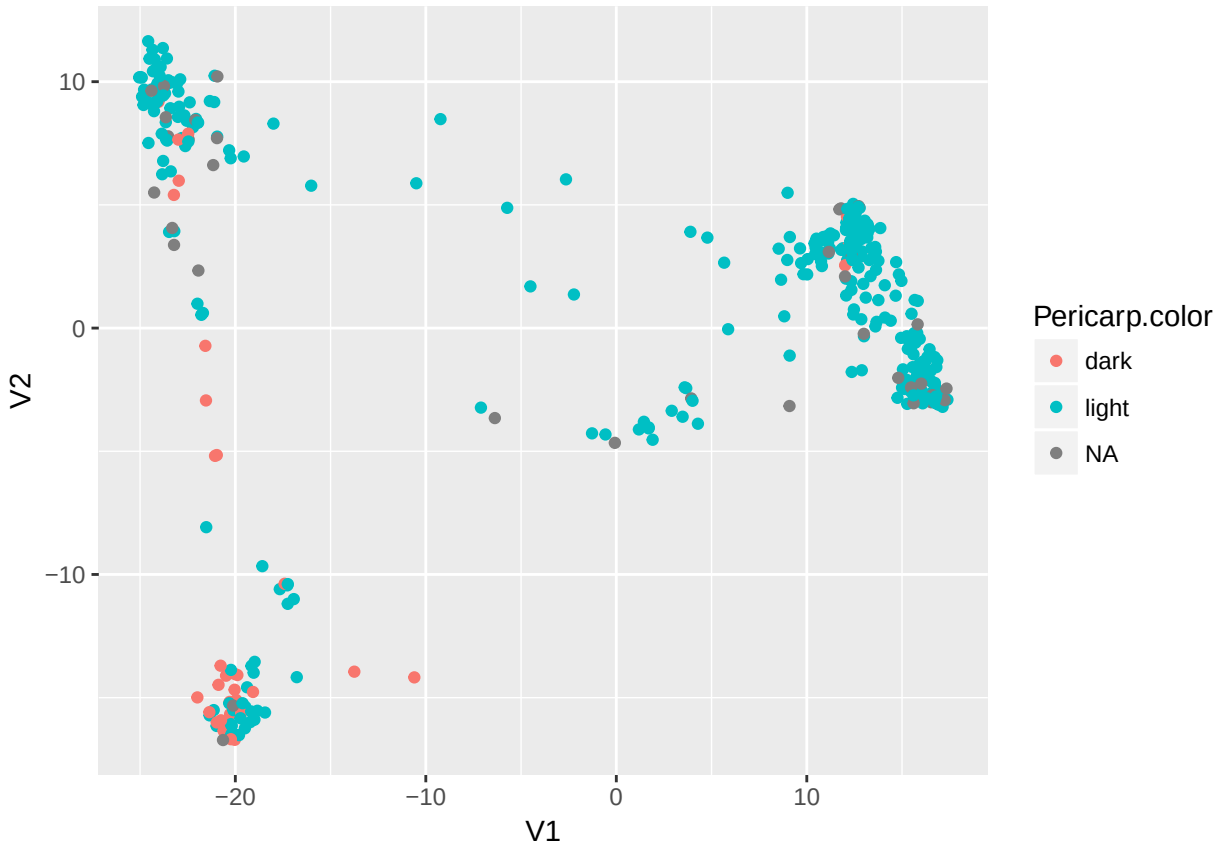
EXERCISE 3: Prepare three different plots to explore if subgroups vary by 1) Amylose content; 2) Pericarp color; 3) Region. Do any of these seem to be associated with the different population groups? Briefly discuss.

Hint 1 use `color=` argument to `qplot` or `ggplot` to color the point by the different traits
Hint 2 use `size=I(3)` as an argument to increase the point size (you can play with different values)
Hint 3 when plotting the Region data, the colors will be easier to interpret if you include the following at the end of the line with your qplot command: `+ scale_color_brewer(type="div")` This specifies that a diverging set of colors is used. (Try plotting with and without this).

```
#Amylose MDS
amylose <- ggplot(data.geno.pheno, aes(x = V1, y = V2, color = Amylose.content))
amylose + geom_point() + scale_colour_gradient2(low = "red", mid = "green",
  high = "blue", midpoint = 15, guide = "colourbar")
```

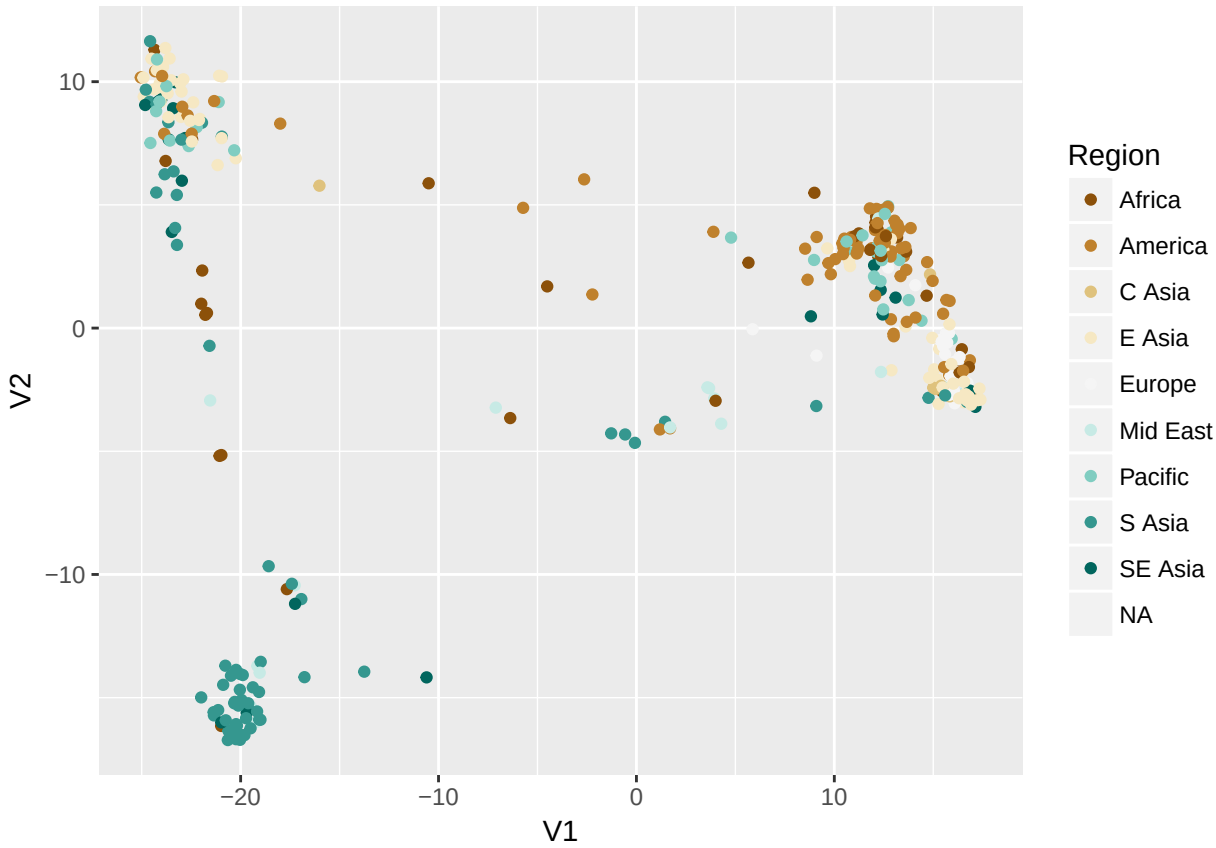



```
#Pericarp MDS
pericarp <- ggplot(data.geno.pheno, aes(x = V1, y = V2, color = Pericarp.color))
pericarp + geom_point()
```




```
#Region MDS
region <- ggplot(data.geno.pheno, aes(x = V1, y = V2, color = Region))
region + geom_point() + scale_color_brewer(type="div")
```

```
## Warning: Removed 42 rows containing missing values (geom_point).
```



EXERCISE 4: Re plot the MDS data, but include the population assignment in an informative way. How do the populations assignments relate to the MDS plot? PSMix: Assigning varieties to populations. From the MDS there looks like there is structure in our population, we will assign individuals to specific populations classes with PSMix package.

Convert our genotypes to PSMix format (separate row for each allele)

```
#Convert to character matrix.  The apply function applies a function (in this case as.character()) eith
data.geno.2500.c <- apply(data.geno.2500,2,as.character)

#Create a new Matrix to hold reformatted data
data.geno.2500.ps <- matrix("",nrow=nrow(data.geno.2500.c)*2,ncol=ncol(data.geno.2500.c))

#for each row of genotypes, create 2 rows, one with the first allele and one with the second allele.
for (i in 1:nrow(data.geno.2500.c)) {
  data.geno.2500.ps[(i-1)*2+1,] <- substr(data.geno.2500.c[i,],1,1)
  data.geno.2500.ps[(i-1)*2+2,] <- substr(data.geno.2500.c[i,],2,2)
  }
```

```
library(PSMix)
load("./RiceSNPData/ps4.2500.RData")

#run on K=4 popultions and 2500 markers; may take 15-30 minutes
#system.time(ps4 <- PSMix(K=4,data.geno.2500.ps,eps=1e-05,verbose=T))
#save(ps4,file="../data/ps4.2500.RData")

#2500 markers K = 5 > 1 hour run time
#system.time(ps5 <- PSMix(K=5,data.geno.2500.ps,eps=1e-05,verbose=T))
#save(ps5,file="../data/ps5.2500.RData")
```

Examine Output

```
names(ps4) #Show us elements within ps4
```

```
## [1] "AmPr" "AmId"
```

```
head(ps4$AmPr)
```

```
##               [,1]          [,2]          [,3]          [,4]
## [1,]  3.906153e-74  1.000000e+00 4.940656e-324 4.940656e-324
## [2,] 1.632141e-223 9.326173e-252  1.000000e+00 1.571730e-111
## [3,]  3.199112e-39  1.623725e-82  1.263693e-01  8.736307e-01
## [4,]  3.846553e-01  2.629090e-01  3.702799e-02  3.154077e-01
## [5,]  1.655290e-02  2.043686e-34  1.336814e-01  8.497657e-01
## [6,]  9.255409e-01  2.628336e-02  4.817578e-02  7.551204e-29
```

```
round(head(ps4$AmPr), 3) #Round to 3 decimal places
```

```
##       [,1]  [,2]  [,3]  [,4]
## [1,] 0.000 1.000 0.000 0.000
## [2,] 0.000 0.000 1.000 0.000
## [3,] 0.000 0.000 0.126 0.874
## [4,] 0.385 0.263 0.037 0.315
## [5,] 0.017 0.000 0.134 0.850
## [6,] 0.926 0.026 0.048 0.000
```

Each row in the AmPr table is an individual and each column represents one of the hypothesized populations. Genomes with substantial contributions from two ancestral genomes are said to be admixed.

The second component, AmID, shows an assignment of each individual to a signle ancestral population.

```
head(ps4$AmId)
```

```
## [1] 2 3 4 1 4 1
```

```
table(ps4$AmId)
```

```
##
##   1   2   3   4
## 133 117  96  67
```

```
ps4.df <- as.data.frame(cbind(round(ps4$AmPr,3),ps4$AmId))
head(ps4.df) #look at the new data frame
```

```
##      V1    V2    V3    V4 V5
## 1 0.000 1.000 0.000 0.000  2
## 2 0.000 0.000 1.000 0.000  3
## 3 0.000 0.000 0.126 0.874  4
## 4 0.385 0.263 0.037 0.315  1
## 5 0.017 0.000 0.134 0.850  4
## 6 0.926 0.026 0.048 0.000  1
```

```
#Next add useful column names
colnames(ps4.df) <- c(paste("pop",1:(ncol(ps4.df)-1),sep=""),"popID")
head(ps4.df) #look at the new data frame
```

```
##    pop1  pop2  pop3  pop4 popID
## 1 0.000 1.000 0.000 0.000     2
## 2 0.000 0.000 1.000 0.000     3
## 3 0.000 0.000 0.126 0.874     4
```

```
## 4 0.385 0.263 0.037 0.315     1
## 5 0.017 0.000 0.134 0.850     4
## 6 0.926 0.026 0.048 0.000     1
```

For plotting it will be helpful to order the samples based on population for each individual. This is done using apply(), which applies functios across every row or column of a dataframe.

```
maxGenome <- apply(ps4$AmPr,1,max)

#now we order the varieties by their prediced population membership and their degree of admixture.
ps4.df <- ps4.df[order(ps4.df$popID,-maxGenome),]
#Add a column for sample index
ps4.df$sampleID <- factor(1:413)
head(ps4.df)
```

```
##    pop1 pop2 pop3 pop4 popID sampleID
## 12    1    0    0    0     1        1
## 18    1    0    0    0     1        2
## 20    1    0    0    0     1        3
## 21    1    0    0    0     1        4
## 22    1    0    0    0     1        5
## 23    1    0    0    0     1        6
```

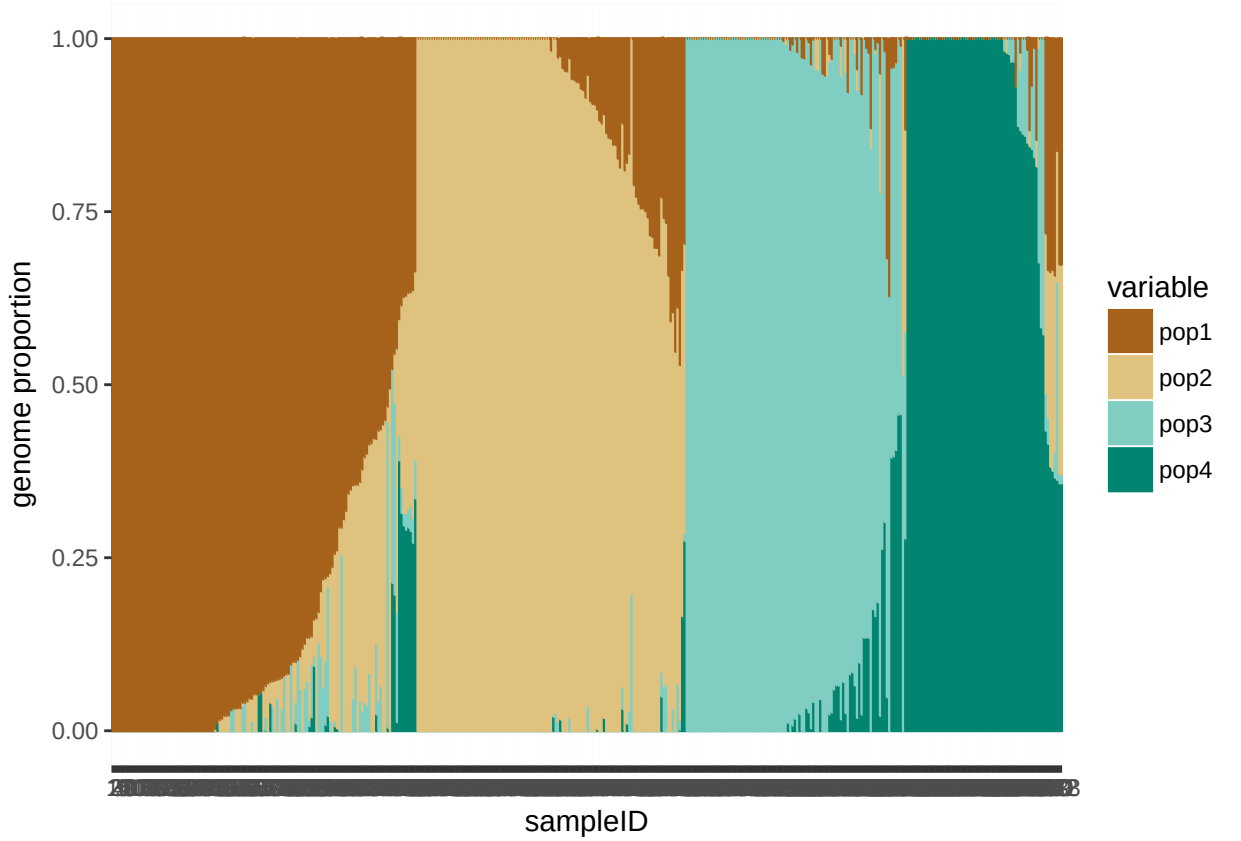
Now take the data from wide to long format as ggplot needs one observation per row.

```
library(reshape2)
ps4.df.melt <- melt(ps4.df,id.vars=c("popID","sampleID"))
head(ps4.df.melt) #look a the melted data set.
```

```
##   popID sampleID variable value
## 1     1        1     pop1     1
## 2     1        2     pop1     1
## 3     1        3     pop1     1
## 4     1        4     pop1     1
## 5     1        5     pop1     1
## 6     1        6     pop1     1
```

Each color is a single rice variety and colors correspond to ancestral genomes.

```
library(ggplot2)

pl <- ggplot(aes(x=sampleID, y=value, color=variable, fill=variable), data=ps4.df.melt)
pl <- pl + geom_bar(stat="identity")
pl <- pl + ylab("genome proportion") + scale_color_brewer(type="div") + scale_fill_brewer(type="div")

pl
```




```
geno.mds$popID <- factor(ps4$AmId)
head(geno.mds$popID)
```

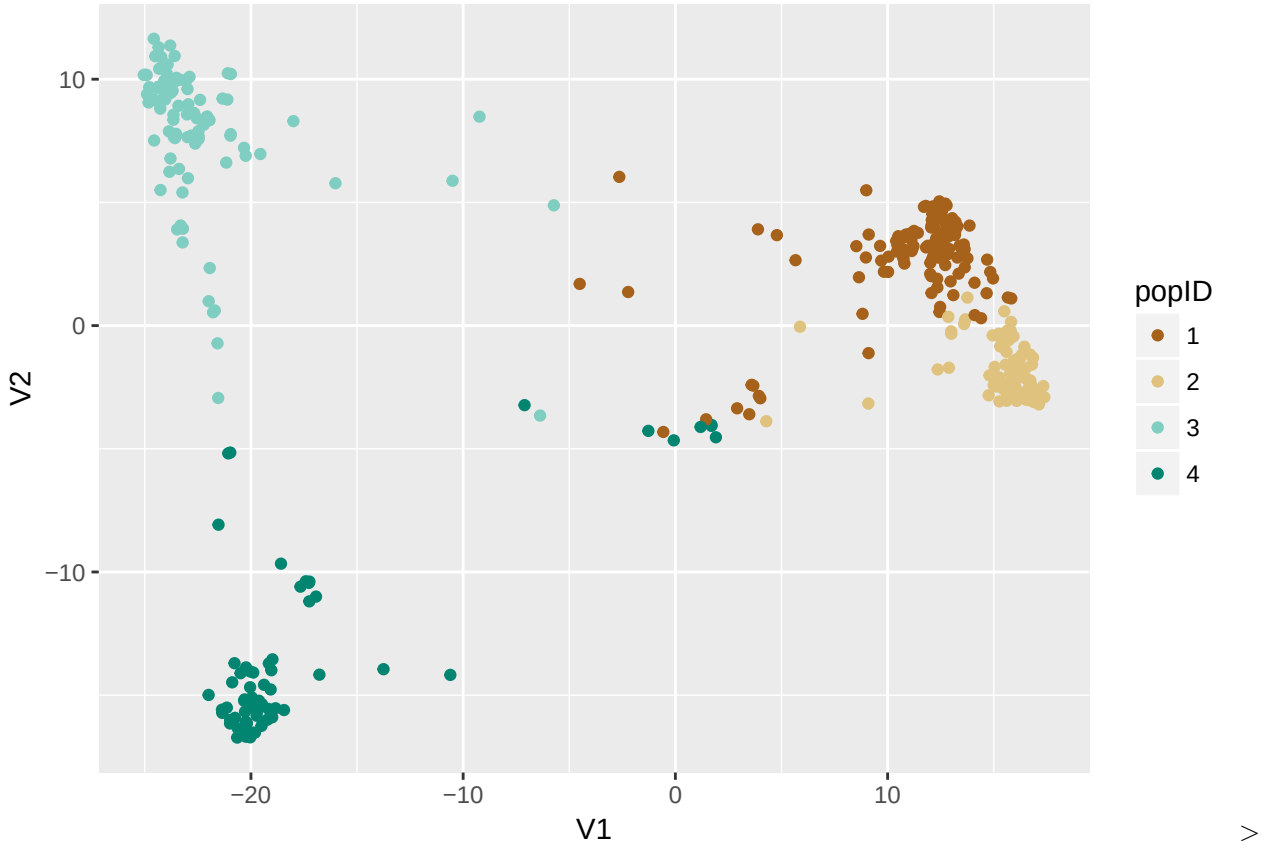
```
## [1] 2 3 4 1 4 1
## Levels: 1 2 3 4
```

```
colnames(ps4$AmPr) <- paste("pr",1:4,sep="")

geno.mds <- cbind(geno.mds,ps4$AmPr)
head(geno.mds)
```

```
##                V1         V2 popID           pr1           pr2
## NSFTV1  16.724400  -2.800265     2  3.906153e-74  1.000000e+00
## NSFTV3 -24.367765   9.690134     3 1.632141e-223 9.326173e-252
## NSFTV4 -17.243869 -10.390602     4  3.199112e-39  1.623725e-82
## NSFTV5   2.909696  -3.357328     1  3.846553e-01  2.629090e-01
## NSFTV6 -16.932450 -10.999276     4  1.655290e-02  2.043686e-34
## NSFTV7  10.620649   3.513382     1  9.255409e-01  2.628336e-02
##                  pr3           pr4
## NSFTV1 4.940656e-324 4.940656e-324
## NSFTV3  1.000000e+00 1.571730e-111
## NSFTV4  1.263693e-01  8.736307e-01
## NSFTV5  3.702799e-02  3.154077e-01
## NSFTV6  1.336814e-01  8.497657e-01
## NSFTV7  4.817578e-02  7.551204e-29
```

```
#PopulationMDS
population <- ggplot(geno.mds, aes(x = V1, y = V2, color = popID))
```

```
population + geom_point() + scale_color_brewer(type="div")
```



This MDS plot shows where the individual plants group together in subpopulations. The sub-populations 1 and 2 are closely related, while 3 and 4 are diverged populations.

```
save(data.pheno,geno.mds,file="data_from_SNP_lab.Rdata")
```

Exercise 5:

- Plot your chosen trait data
- as a **single histogram** for all of the data
- as **separate histograms** for each of the 4 population assignments made by PSMix
- as a **boxplot** separated by population.
- Based on these histograms do you think that your trait varies by population?
- **BONUS** Try using the "violin" geom. What is this showing?

Hint: you will need to use a different binwidth (or don't specify it at all and let R choose the default). Hint: the relevant column names for population are "popID".

Merge Data of amylose

```
data.pheno.mds <- merge(geno.mds,data.pheno,by="row.names",all=T) #even if you already have this object
head(data.pheno.mds)
```

```
##   Row.names        V1         V2 popID           pr1          pr2
## 1    NSFTV1  16.72440 -2.8002646     2  3.906153e-74 1.000000e+00
## 2   NSFTV10  16.57548 -2.6215418     2 5.975221e-105 1.000000e+00
## 3  NSFTV100  15.49797  0.5808688     2  4.524589e-01 5.475411e-01
## 4  NSFTV101  10.86102  3.6953413     1  9.279480e-01 3.621048e-02
## 5  NSFTV102 -23.53711  9.9287307     3  2.664270e-02 9.945103e-07
```

```
## 6  NSFTV103  15.64048 -2.3548656     2  5.139445e-02 9.121458e-01
##             pr3           pr4     Accession_Name Country_of_Origin
## 1 4.940656e-324 4.940656e-324           Agostano             Italy
## 2  0.000000e+00  0.000000e+00 Baghlani Nangarhar       Afghanistan
## 3 1.486226e-251 5.413008e-254           Lacrosse     United States
## 4  3.584154e-02  1.228608e-59             Lemont     United States
## 5  9.733563e-01  6.766981e-74               <NA>             <NA>
## 6  3.645977e-02 4.916478e-141         Luk Takhar       Afghanistan
##     Region Alu.Tol Flowering.time.at.Arkansas Flowering.time.at.Faridpur
## 1   Europe   0.730                   75.08333                         64
## 2 Mid East   0.902                   89.00000                         55
## 3  America   0.800                   84.11111                         78
## 4  America   0.630                   86.16667                         79
## 5     <NA>   0.440                         NA                         NA
## 6 Mid East   0.550                   84.00000                         78
##   Flowering.time.at.Aberdeen FT.ratio.of.Arkansas.Aberdeen
## 1                         81                     0.9269547
## 2                         74                     1.2027027
## 3                        122                     0.6894353
## 4                         88                     0.9791667
## 5                        165                            NA
## 6                        108                     0.7777778
##   FT.ratio.of.Faridpur.Aberdeen Culm.habit Leaf.pubescence
## 1                     0.7901235   4.000000               1
## 2                     0.7432432   3.000000              NA
## 3                     0.6393443   1.666667               0
## 4                     0.8977273   3.000000               0
## 5                            NA   3.000000              NA
## 6                     0.7222222   2.500000               1
##   Flag.leaf.length Flag.leaf.width Awn.presence Panicle.number.per.plant
## 1         28.37500        1.283333            0                 3.068053
## 2         27.90000        1.000000            1                 3.650658
## 3         27.62222        1.611111            0                 2.978925
## 4         27.62500        1.450000            0                 2.818398
## 5         27.85000        1.100000            1                 3.481240
## 6         30.17500        1.050000            0                 2.957511
##   Plant.height Panicle.length Primary.panicle.branch.number
## 1    110.91667       20.48182                      9.272727
## 2     83.00000       22.16667                     10.333333
## 3    114.88889       23.94444                     11.555556
## 4     86.16667       27.45000                     11.083333
## 5    105.50000       30.75000                     10.500000
## 6     95.08333       24.13333                      9.777778
##   Seed.number.per.panicle Florets.per.panicle Panicle.fertility
## 1                4.785975            4.914658             0.879
## 2                4.110874            4.733270             0.537
## 3                5.032614            5.340738             0.735
## 4                4.894101            5.209031             0.730
## 5                4.600158            4.867534             0.765
## 6                4.881538            4.998900             0.889
##   Seed.length Seed.width Seed.volume Seed.surface.area
## 1    8.064117   3.685183    2.587448          3.914120
## 2    7.859000   3.233250    2.265361          3.729249
## 3    8.138033   3.382633    2.440978          3.850531
```

```
## 4    9.632392   2.644467    2.121810          3.778742
## 5    9.805500   2.469600    2.030632          3.750804
## 6    7.528083   3.534583    2.404007          3.784157
##   Brown.rice.seed.length Brown.rice.seed.width Brown.rice.surface.area
## 1               5.794542              3.113958                3.511152
## 2               5.088267              2.937733                3.309970
## 3               5.944467              2.893033                3.467780
## 4               7.147025              2.251503                3.388738
## 5               7.072600              2.044100                3.301659
## 6               5.298917              3.048408                3.390811
##   Brown.rice.volume Seed.length.width.ratio Brown.rice.length.width.ratio
## 1          7.737358                  2.188                         1.861
## 2          5.912900                  2.431                         1.732
## 3          6.898900                  2.406                         2.055
## 4          4.919557                  3.642                         3.174
## 5          4.130800                  3.970                         3.460
## 6          6.670092                  2.130                         1.738
##   Seed.color Pericarp.color Straighthead.suseptability Blast.resistance
## 1      light          light                   4.833333                8
## 2      light          light                   3.330000                2
## 3      light          light                   6.501667                3
## 4      light          light                   3.331667                8
## 5      light          light                   6.835000                1
## 6      light          light                   5.335000                4
##   Amylose.content Alkali.spreading.value Protein.content
## 1        15.61333               6.083333           8.45
## 2        15.09667               7.000000           9.50
## 3        10.60333               6.958333           8.70
## 4        20.51333               6.000000           9.50
## 5        21.25333               5.000000           8.70
## 6        16.97667               6.916667           7.75
```

```
summary(data.pheno.mds)
```

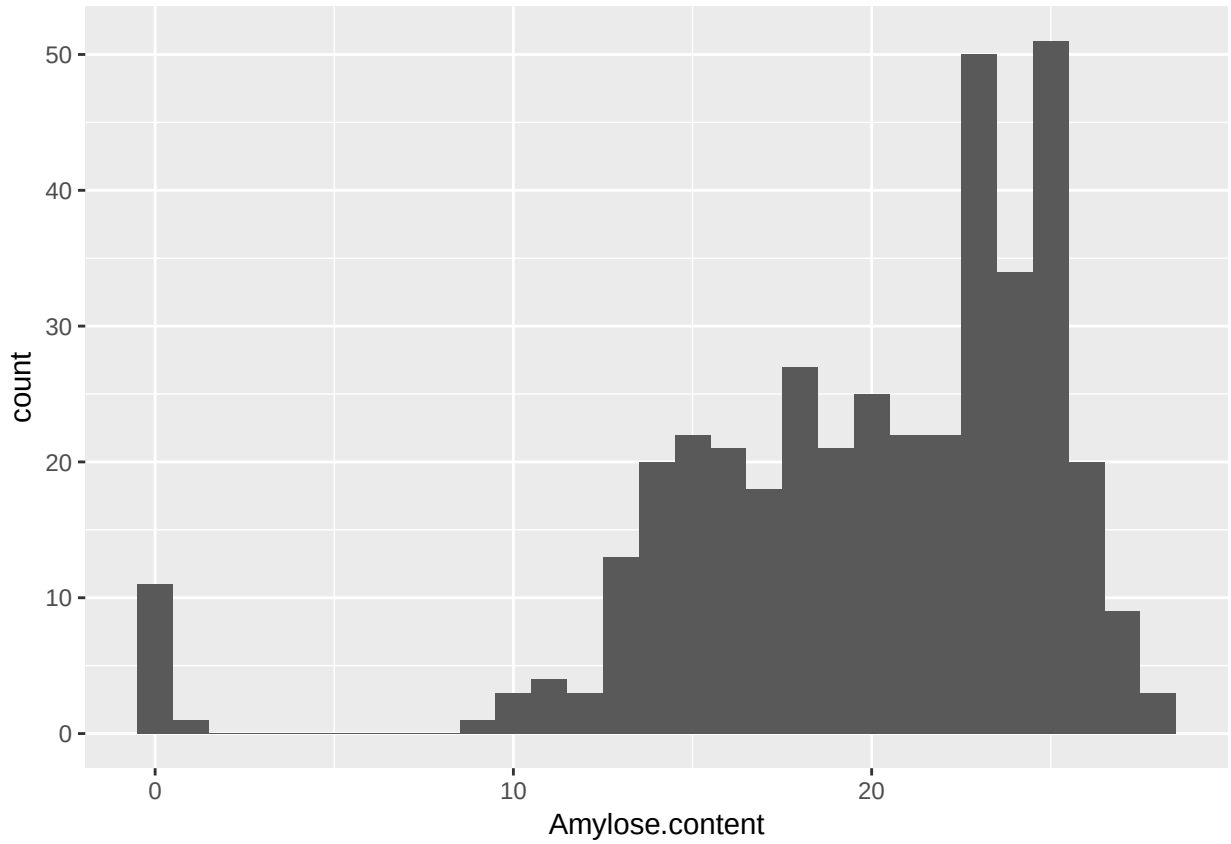
```
##   Row.names              V1                V2          popID
##  Length:413         Min.   :-25.05   Min.   :-16.720   1:133
##  Class :AsIs        1st Qu.:-20.32   1st Qu.: -2.848   2:117
##  Mode  :character   Median : 11.16   Median :  1.236   3: 96
##                     Mean   :  0.00   Mean   :  0.000   4: 67
##                     3rd Qu.: 15.05   3rd Qu.:  4.524
##                     Max.   : 17.38   Max.   : 11.642
##
##       pr1               pr2               pr3              pr4
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.00000
##  Median :0.04214   Median :0.02284   Median :0.0000   Median :0.00000
##  Mean   :0.30717   Mean   :0.29607   Mean   :0.2303   Mean   :0.16649
##  3rd Qu.:0.65722   3rd Qu.:0.69794   3rd Qu.:0.2555   3rd Qu.:0.05348
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.0000   Max.   :1.00000
##
##        Accession_Name     Country_of_Origin     Region      Alu.Tol
##  Azucena      :  2    United States: 41     America:83   Min.   :0.0300
##  Carolina Gold:  2    China        : 28     E Asia :73   1st Qu.:0.4000
##  Dom Sufid    :  2    India        : 27     S Asia :64   Median :0.6035
##  M-202        :  2    Bangladesh   : 24     Africa :39   Mean   :0.5874
```

```
##  Moroberekan  :  2    Japan        : 17     Europe :34   3rd Qu.:0.7500
##  (Other)      :373    (Other)      :246     (Other):78   Max.   :1.3500
##  NA's         : 30    NA's         : 30     NA's   :42   NA's   :43
##  Flowering.time.at.Arkansas Flowering.time.at.Faridpur
##  Min.   : 54.50             Min.   : 39.00
##  1st Qu.: 79.75             1st Qu.: 66.00
##  Median : 87.71             Median : 74.00
##  Mean   : 87.94             Mean   : 71.77
##  3rd Qu.: 96.83             3rd Qu.: 78.00
##  Max.   :150.50             Max.   :110.00
##  NA's   :39                 NA's   :108
##  Flowering.time.at.Aberdeen FT.ratio.of.Arkansas.Aberdeen
##  Min.   : 45.0              Min.   :0.3724
##  1st Qu.: 81.5              1st Qu.:0.7975
##  Median : 99.0              Median :0.8960
##  Mean   :107.1              Mean   :0.8949
##  3rd Qu.:114.0              3rd Qu.:1.0195
##  Max.   :306.0              Max.   :1.7031
##  NA's   :54                 NA's   :64
##  FT.ratio.of.Faridpur.Aberdeen   Culm.habit    Leaf.pubescence
##  Min.   :0.3459                Min.   :1.000   Min.   :0.0000
##  1st Qu.:0.6838                1st Qu.:3.000   1st Qu.:1.0000
##  Median :0.7720                Median :4.000   Median :1.0000
##  Mean   :0.7711                Mean   :4.228   Mean   :0.8507
##  3rd Qu.:0.8671                3rd Qu.:5.500   3rd Qu.:1.0000
##  Max.   :1.2885                Max.   :9.000   Max.   :1.0000
##  NA's   :109                   NA's   :29      NA's   :125
##  Flag.leaf.length Flag.leaf.width   Awn.presence
##  Min.   :15.42    Min.   :0.5917   Min.   :0.0000
##  1st Qu.:26.62    1st Qu.:1.0400   1st Qu.:0.0000
##  Median :30.05    Median :1.1833   Median :0.0000
##  Mean   :30.63    Mean   :1.2217   Mean   :0.2222
##  3rd Qu.:34.55    3rd Qu.:1.4111   3rd Qu.:0.0000
##  Max.   :49.44    Max.   :1.8917   Max.   :1.0000
##  NA's   :36       NA's   :36       NA's   :44
##  Panicle.number.per.plant  Plant.height    Panicle.length
##  Min.   :2.234             Min.   : 67.75   Min.   :15.63
##  1st Qu.:2.931             1st Qu.: 99.75   1st Qu.:22.24
##  Median :3.242             Median :117.50   Median :24.19
##  Mean   :3.247             Mean   :116.58   Mean   :24.37
##  3rd Qu.:3.558             3rd Qu.:131.39   3rd Qu.:26.45
##  Max.   :4.172             Max.   :194.33   Max.   :35.68
##  NA's   :41                NA's   :30       NA's   :38
##  Primary.panicle.branch.number Seed.number.per.panicle Florets.per.panicle
##  Min.   : 5.556                Min.   :3.445           Min.   :3.909
##  1st Qu.: 8.667                1st Qu.:4.679           1st Qu.:4.879
##  Median : 9.917                Median :4.888           Median :5.065
##  Mean   : 9.943                Mean   :4.854           Mean   :5.056
##  3rd Qu.:11.111                3rd Qu.:5.054           3rd Qu.:5.258
##  Max.   :17.000                Max.   :5.635           Max.   :5.836
##  NA's   :38                    NA's   :37              NA's   :36
##  Panicle.fertility  Seed.length       Seed.width     Seed.volume
##  Min.   :0.3720    Min.   : 5.894   Min.   :2.196   Min.   :1.669
##  1st Qu.:0.7830    1st Qu.: 7.680   1st Qu.:2.819   1st Qu.:2.118
```

```
##  Median :0.8530    Median : 8.319   Median :3.157   Median :2.280
##  Mean   :0.8240    Mean   : 8.400   Mean   :3.117   Mean   :2.278
##  3rd Qu.:0.8932    3rd Qu.: 9.118   3rd Qu.:3.398   3rd Qu.:2.454
##  Max.   :0.9800    Max.   :12.549   Max.   :3.990   Max.   :2.872
##  NA's   :37        NA's   :36       NA's   :36      NA's   :36
##  Seed.surface.area Brown.rice.seed.length Brown.rice.seed.width
##  Min.   :3.434     Min.   :4.093          Min.   :1.820
##  1st Qu.:3.679     1st Qu.:5.501          1st Qu.:2.388
##  Median :3.764     Median :5.999          Median :2.655
##  Mean   :3.781     Mean   :6.117          Mean   :2.627
##  3rd Qu.:3.883     3rd Qu.:6.753          3rd Qu.:2.876
##  Max.   :4.198     Max.   :8.784          Max.   :3.358
##  NA's   :36        NA's   :36             NA's   :36
##  Brown.rice.surface.area Brown.rice.volume Seed.length.width.ratio
##  Min.   :2.959           Min.   :2.841     Min.   :1.799
##  1st Qu.:3.284           1st Qu.:4.806     1st Qu.:2.327
##  Median :3.363           Median :5.650     Median :2.651
##  Mean   :3.378           Mean   :5.806     Mean   :2.752
##  3rd Qu.:3.471           3rd Qu.:6.717     3rd Qu.:3.068
##  Max.   :3.766           Max.   :9.861     Max.   :4.467
##  NA's   :36              NA's   :36        NA's   :36
##  Brown.rice.length.width.ratio Seed.color  Pericarp.color
##  Min.   :1.502                 dark :  8   dark : 34
##  1st Qu.:1.997                 light:369   light:343
##  Median :2.287                 NA's : 36   NA's : 36
##  Mean   :2.382
##  3rd Qu.:2.678
##  Max.   :3.878
##  NA's   :36
##  Straighthead.suseptability Blast.resistance Amylose.content
##  Min.   :2.330              Min.   :0.000    Min.   : 0.00
##  1st Qu.:5.998              1st Qu.:2.000    1st Qu.:16.70
##  Median :7.165              Median :5.000    Median :21.13
##  Mean   :6.936              Mean   :5.039    Mean   :19.88
##  3rd Qu.:8.167              3rd Qu.:8.000    3rd Qu.:23.97
##  Max.   :9.000              Max.   :9.000    Max.   :27.96
##  NA's   :73                 NA's   :28       NA's   :12
##  Alkali.spreading.value Protein.content
##  Min.   :2.000          Min.   : 6.500
##  1st Qu.:5.403          1st Qu.: 7.950
##  Median :6.083          Median : 8.450
##  Mean   :5.974          Mean   : 8.593
##  3rd Qu.:6.938          3rd Qu.: 9.050
##  Max.   :7.000          Max.   :14.100
##  NA's   :10             NA's   :20
```

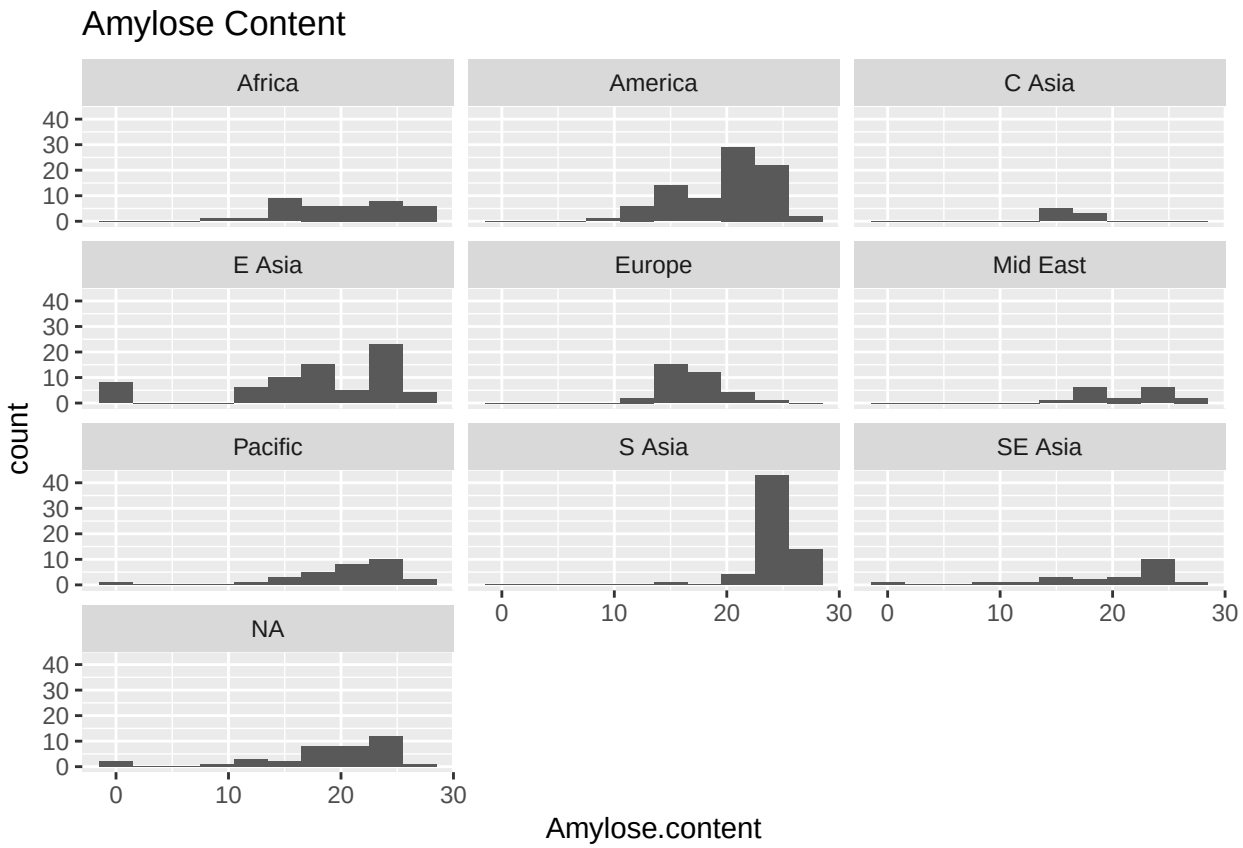
```
pl <- ggplot(data.pheno.mds,aes(x=Amylose.content)) + geom_histogram(binwidth = 1)
pl
```

```
## Warning: Removed 12 rows containing non-finite values (stat_bin).
```



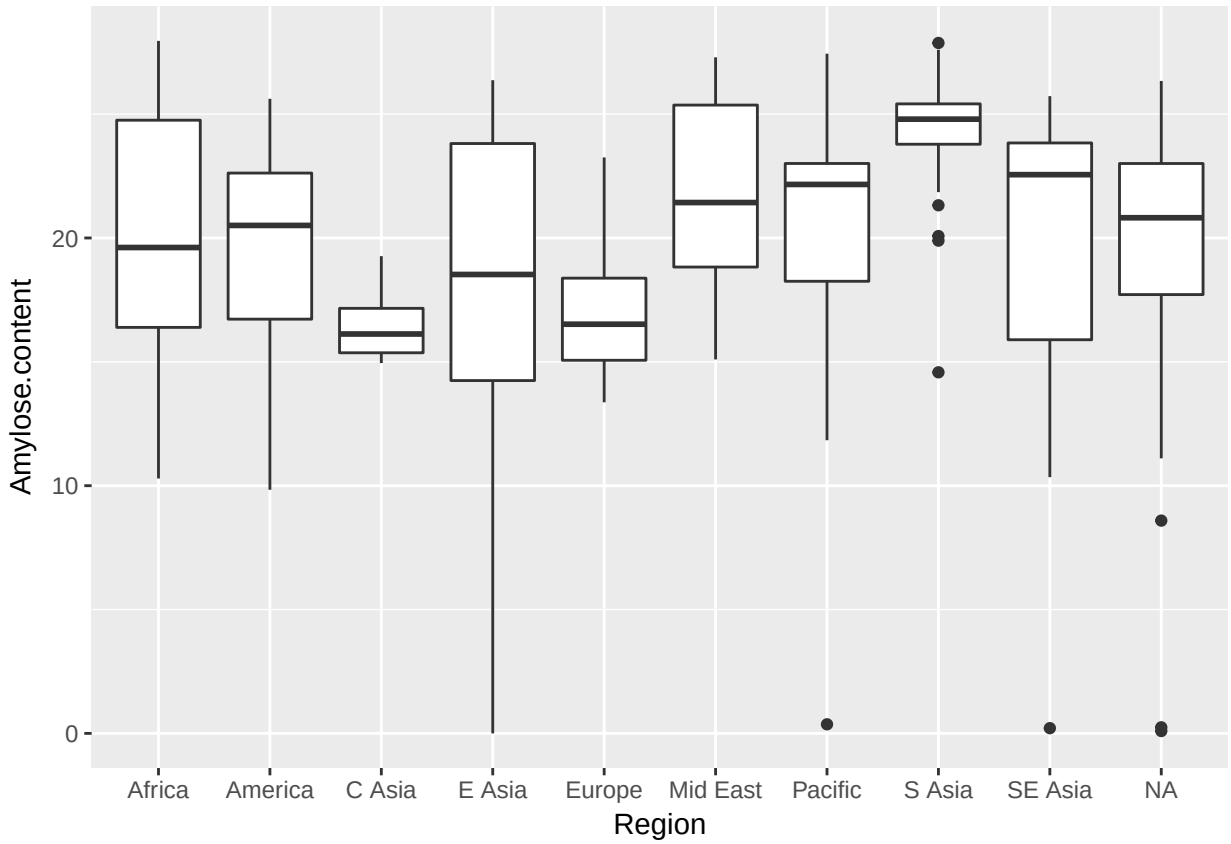

```
pl <- ggplot(data=data.pheno.mds,aes(x=Amylose.content)) #create the basic plot object
pl <- pl + geom_histogram(binwidth=3) #tell R that we want a histogram, with binwidth of 3
pl <- pl + facet_wrap(facets= ~ Region, ncol=3) # a separate plot ("facet") for each region, arranged i
pl <- pl + ggtitle("Amylose Content") #add a title
pl #display the plot
```

```
## Warning: Removed 12 rows containing non-finite values (stat_bin).
```



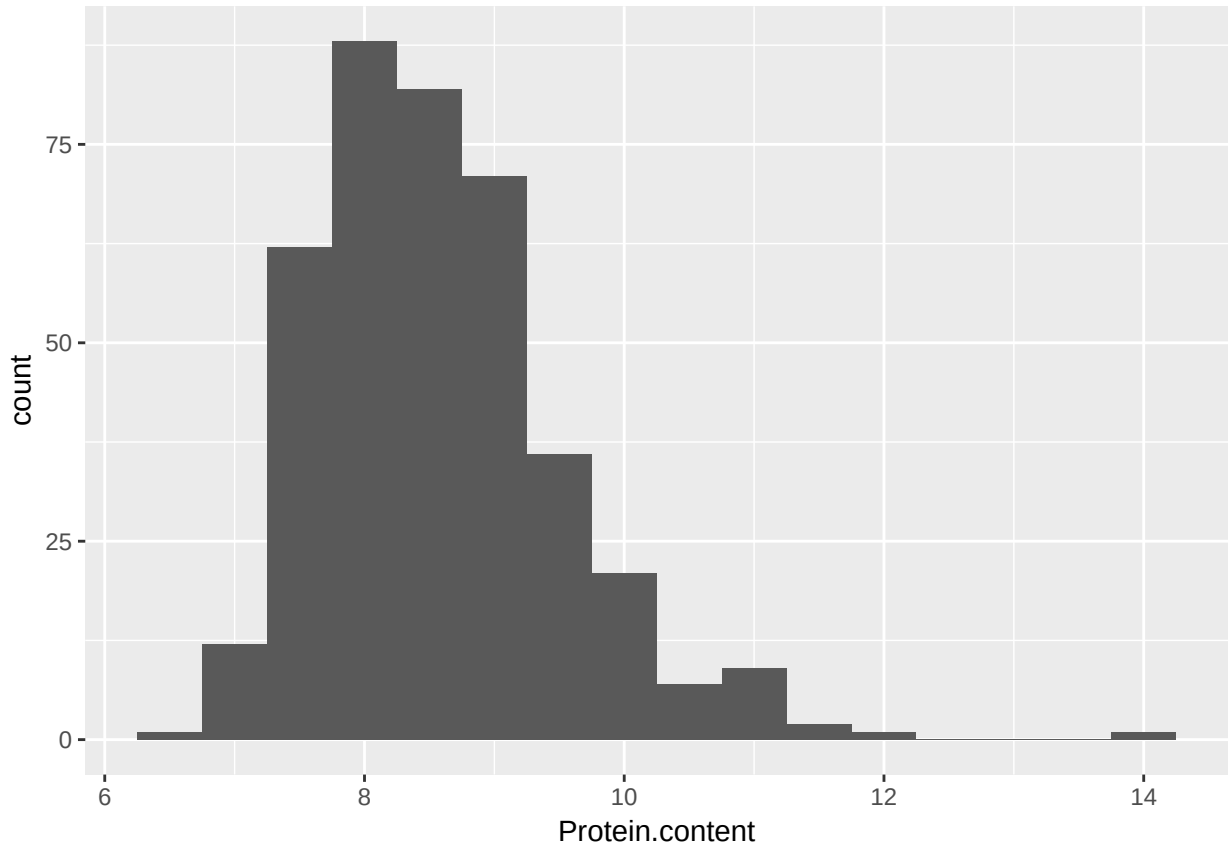

```
pl <- ggplot(data=data.pheno.mds,aes(x=Region,y=Amylose.content)) + geom_boxplot()
pl
```

```
## Warning: Removed 12 rows containing non-finite values (stat_boxplot).
```



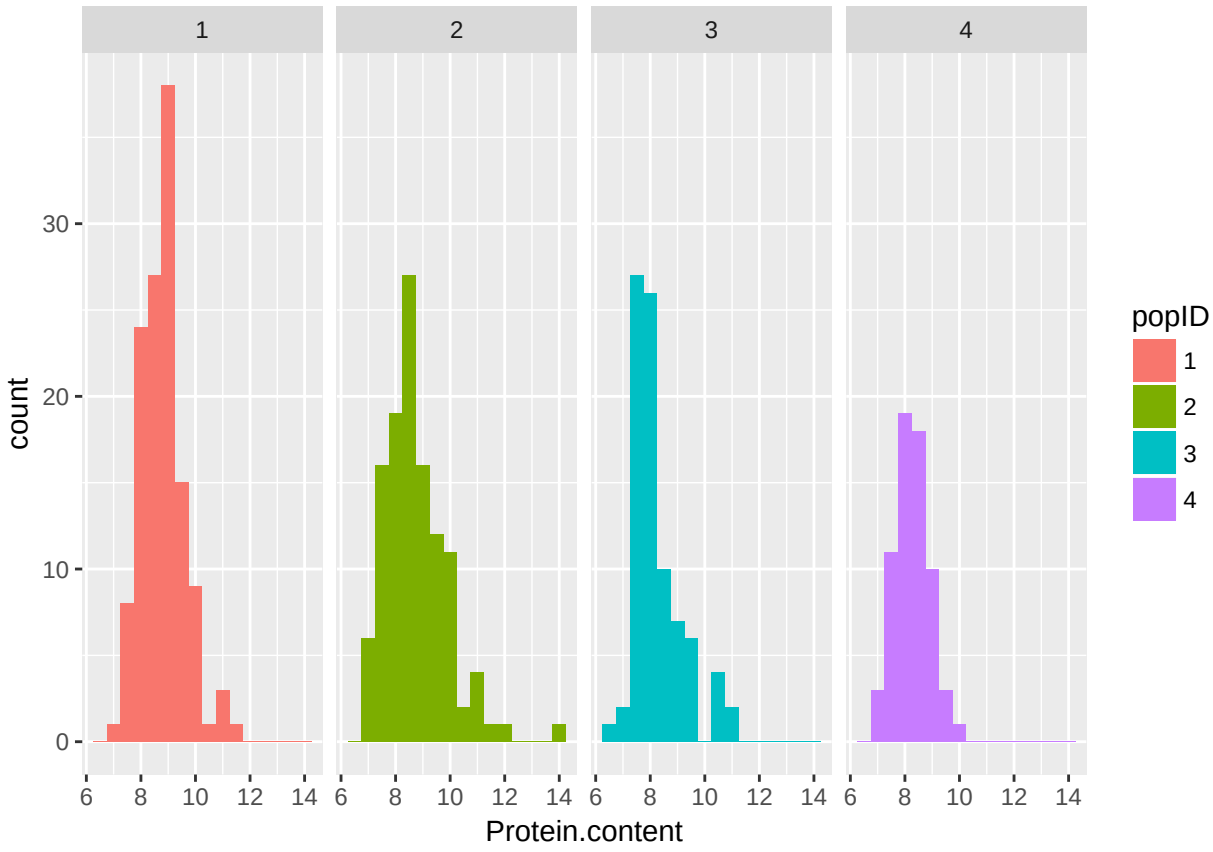

```
# Single Histogram for all of the data
ggplot(data.pheno.mds, aes(x = Protein.content)) + geom_histogram(binwidth = 0.5)
```

```
## Warning: Removed 20 rows containing non-finite values (stat_bin).
```



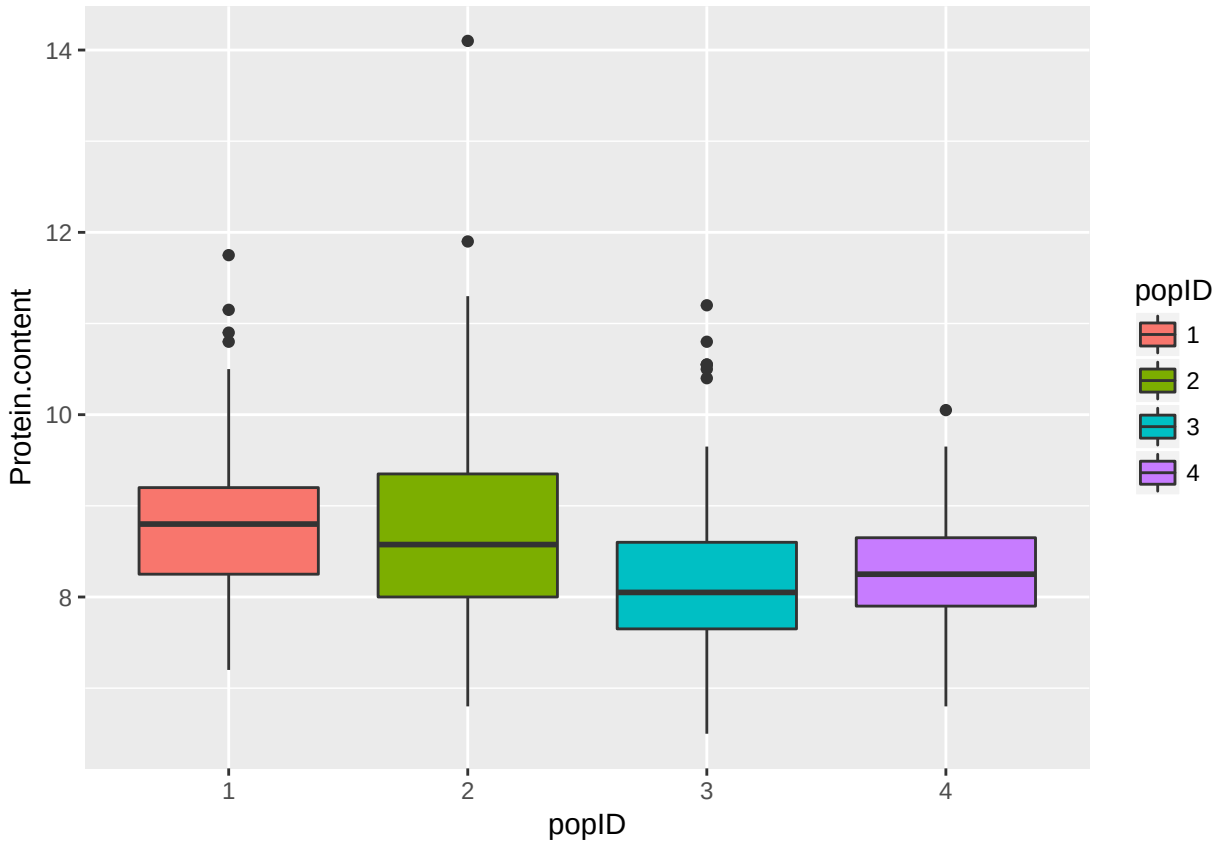

```
# Separate histograms for each of the 4 populations
ggplot(data.pheno.mds, aes(x = Protein.content, fill = popID)) + geom_histogram(binwidth = 0.5) + facet_
```

```
## Warning: Removed 20 rows containing non-finite values (stat_bin).
```



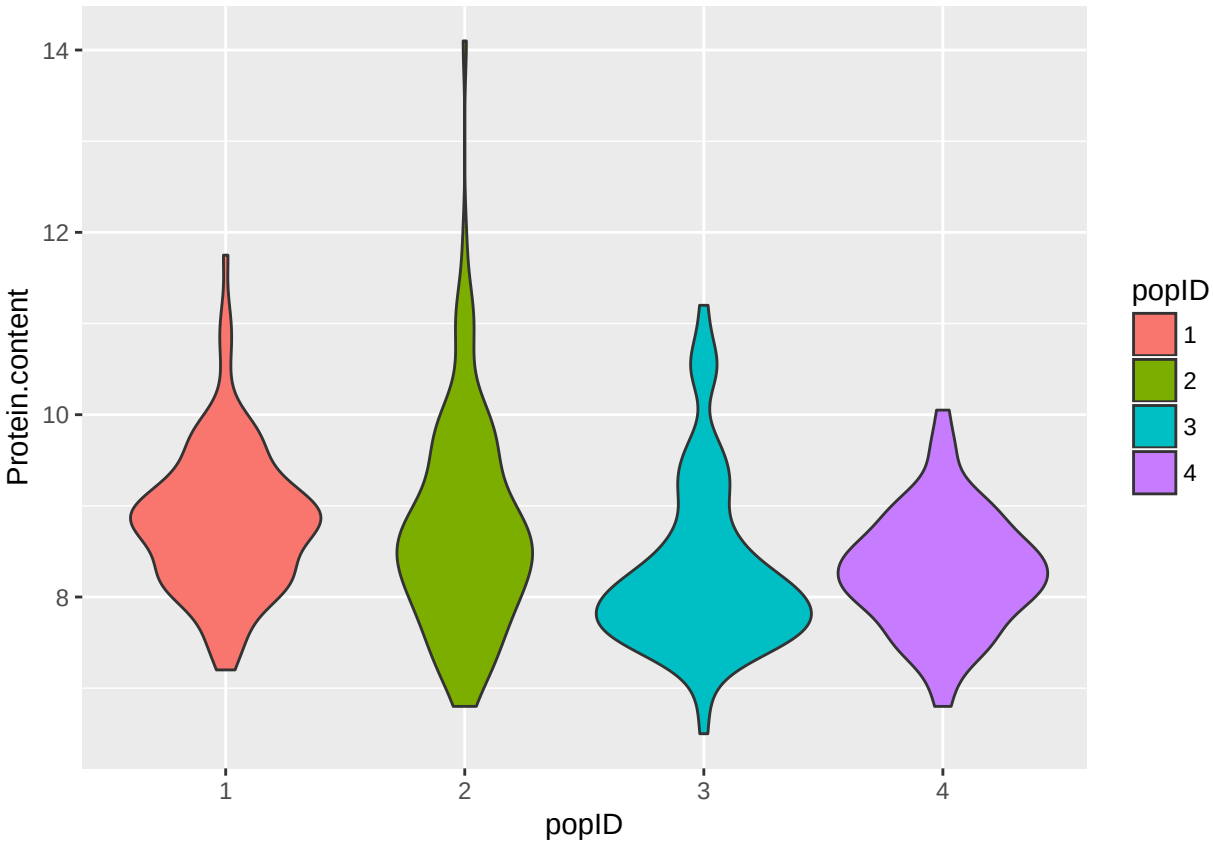

```
# As a boxplot separated by population
ggplot(data.pheno.mds, aes(x = popID, y = Protein.content, fill = popID)) + geom_boxplot()
```

```
## Warning: Removed 20 rows containing non-finite values (stat_boxplot).
```




```
# As a violin plot
ggplot(data.pheno.mds, aes(x = popID, y = Protein.content, fill = popID)) + geom_violin()
```

```
## Warning: Removed 20 rows containing non-finite values (stat_ydensity).
```



The protein content averages indicated by the box plot do not show that much variance by population.

Exercise 6:

- Obtain the mean of your trait for each of the 4 PSMix populations.
- Perform an ANOVA for your trait to test if it varies significantly by population. Show your code, the ANOVA output, and provide an interpretation.
- Discuss: Do your results present a problem for GWAS?

Calculate the mean

```
mean(data.pheno.mds$Protein.content,na.rm=T) #the na.rm argument tells R to ignore missing data coded b
```

```
## [1] 8.592557
```

```
tapply(X=data.pheno.mds$Protein.content,INDEX=data.pheno.mds$popID,FUN=min,na.rm=T)
```

```
##   1   2   3   4
## 7.2 6.8 6.5 6.8
```

The mean of protein content for the overall data is 8.59. The mean of protein content for each population ID is 7.2, 6.8, 6.5, an 6.8.

```
#ANOVA test to check if the differences in mean we see among the diferent population is significant.
aov1 <- aov(Protein.content ~ popID,data=data.pheno.mds)
summary(aov1)
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## popID         3   24.2   8.072    9.74 3.28e-06 ***
## Residuals   389  322.4   0.829
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## 20 observations deleted due to missingness
```

The ANOVA test returns a small p value of 3.28e-06. This is problematic for GWAS studies as the differences among the 4 sub populations is significant, it is a good indicator for population structure.

To be ready to perfrom a GWAS run the following:

```
#load information about snp chromosome and position
snpinfo <- read.csv("./RiceSNPData/snpInfo.csv",row.names=1)

head(snpinfo) #note one column for chromosome and one for position (in base pairs)
```

```
##        snp chr   pos
## 1 X1_13147   1 13147
## 2 X1_73192   1 73192
## 3 X1_74969   1 74969
## 4 X1_75852   1 75852
## 5 X1_75953   1 75953
## 6 X1_91016   1 91016
```

```
# you will need the data.geno file from the previous lab.  If you don't have it in your workspace, relo
data.geno <- read.csv("./RiceSNPData/Rice_44K_genotypes.csv.gz", row.names=1, na.strings=c("NA","00"))

#next merge the genotype information with the phenotype and population info:
head(data.pheno.mds) #note: need to get rownames assigned correctly again (they were lost after the mer
```

```
##   Row.names        V1         V2 popID           pr1          pr2
## 1    NSFTV1  16.72440 -2.8002646     2  3.906153e-74 1.000000e+00
## 2   NSFTV10  16.57548 -2.6215418     2 5.975221e-105 1.000000e+00
## 3  NSFTV100  15.49797  0.5808688     2  4.524589e-01 5.475411e-01
## 4  NSFTV101  10.86102  3.6953413     1  9.279480e-01 3.621048e-02
## 5  NSFTV102 -23.53711  9.9287307     3  2.664270e-02 9.945103e-07
## 6  NSFTV103  15.64048 -2.3548656     2  5.139445e-02 9.121458e-01
##             pr3           pr4     Accession_Name Country_of_Origin
## 1 4.940656e-324 4.940656e-324           Agostano             Italy
## 2  0.000000e+00  0.000000e+00 Baghlani Nangarhar       Afghanistan
## 3 1.486226e-251 5.413008e-254           Lacrosse     United States
## 4  3.584154e-02  1.228608e-59             Lemont     United States
## 5  9.733563e-01  6.766981e-74               <NA>              <NA>
## 6  3.645977e-02 4.916478e-141         Luk Takhar       Afghanistan
##     Region Alu.Tol Flowering.time.at.Arkansas Flowering.time.at.Faridpur
## 1   Europe   0.730                   75.08333                         64
## 2 Mid East   0.902                   89.00000                         55
## 3  America   0.800                   84.11111                         78
## 4  America   0.630                   86.16667                         79
## 5     <NA>   0.440                         NA                         NA
## 6 Mid East   0.550                   84.00000                         78
##   Flowering.time.at.Aberdeen FT.ratio.of.Arkansas.Aberdeen
## 1                         81                     0.9269547
## 2                         74                     1.2027027
## 3                        122                     0.6894353
## 4                         88                     0.9791667
## 5                        165                            NA
## 6                        108                     0.7777778
##   FT.ratio.of.Faridpur.Aberdeen Culm.habit Leaf.pubescence
```

```
## 1                    0.7901235   4.000000              1
## 2                    0.7432432   3.000000             NA
## 3                    0.6393443   1.666667              0
## 4                    0.8977273   3.000000              0
## 5                           NA   3.000000             NA
## 6                    0.7222222   2.500000              1
##   Flag.leaf.length Flag.leaf.width Awn.presence Panicle.number.per.plant
## 1         28.37500        1.283333            0                 3.068053
## 2         27.90000        1.000000            1                 3.650658
## 3         27.62222        1.611111            0                 2.978925
## 4         27.62500        1.450000            0                 2.818398
## 5         27.85000        1.100000            1                 3.481240
## 6         30.17500        1.050000            0                 2.957511
##   Plant.height Panicle.length Primary.panicle.branch.number
## 1    110.91667       20.48182                     9.272727
## 2     83.00000       22.16667                    10.333333
## 3    114.88889       23.94444                    11.555556
## 4     86.16667       27.45000                    11.083333
## 5    105.50000       30.75000                    10.500000
## 6     95.08333       24.13333                     9.777778
##   Seed.number.per.panicle Florets.per.panicle Panicle.fertility
## 1                4.785975            4.914658             0.879
## 2                4.110874            4.733270             0.537
## 3                5.032614            5.340738             0.735
## 4                4.894101            5.209031             0.730
## 5                4.600158            4.867534             0.765
## 6                4.881538            4.998900             0.889
##   Seed.length Seed.width Seed.volume Seed.surface.area
## 1    8.064117   3.685183    2.587448          3.914120
## 2    7.859000   3.233250    2.265361          3.729249
## 3    8.138033   3.382633    2.440978          3.850531
## 4    9.632392   2.644467    2.121810          3.778742
## 5    9.805500   2.469600    2.030632          3.750804
## 6    7.528083   3.534583    2.404007          3.784157
##   Brown.rice.seed.length Brown.rice.seed.width Brown.rice.surface.area
## 1                5.794542              3.113958                3.511152
## 2                5.088267              2.937733                3.309970
## 3                5.944467              2.893033                3.467780
## 4                7.147025              2.251503                3.388738
## 5                7.072600              2.044100                3.301659
## 6                5.298917              3.048408                3.390811
##   Brown.rice.volume Seed.length.width.ratio Brown.rice.length.width.ratio
## 1          7.737358                   2.188                          1.861
## 2          5.912900                   2.431                          1.732
## 3          6.898900                   2.406                          2.055
## 4          4.919557                   3.642                          3.174
## 5          4.130800                   3.970                          3.460
## 6          6.670092                   2.130                          1.738
##   Seed.color Pericarp.color Straighthead.suseptability Blast.resistance
## 1      light          light                   4.833333                8
## 2      light          light                   3.330000                2
## 3      light          light                   6.501667                3
## 4      light          light                   3.331667                8
## 5      light          light                   6.835000                1
```

```
## 6      light          light                   5.335000               4
##   Amylose.content Alkali.spreading.value Protein.content
## 1        15.61333               6.083333            8.45
## 2        15.09667               7.000000            9.50
## 3        10.60333               6.958333            8.70
## 4        20.51333               6.000000            9.50
## 5        21.25333               5.000000            8.70
## 6        16.97667               6.916667            7.75
```

```
rownames(data.pheno.mds) <- data.pheno.mds$Row.names

data.geno.pheno <- merge(data.pheno.mds,data.geno,by="row.names")
```

```
## Warning in merge.data.frame(data.pheno.mds, data.geno, by = "row.names"):
## column name 'Row.names' is duplicated in the result
```

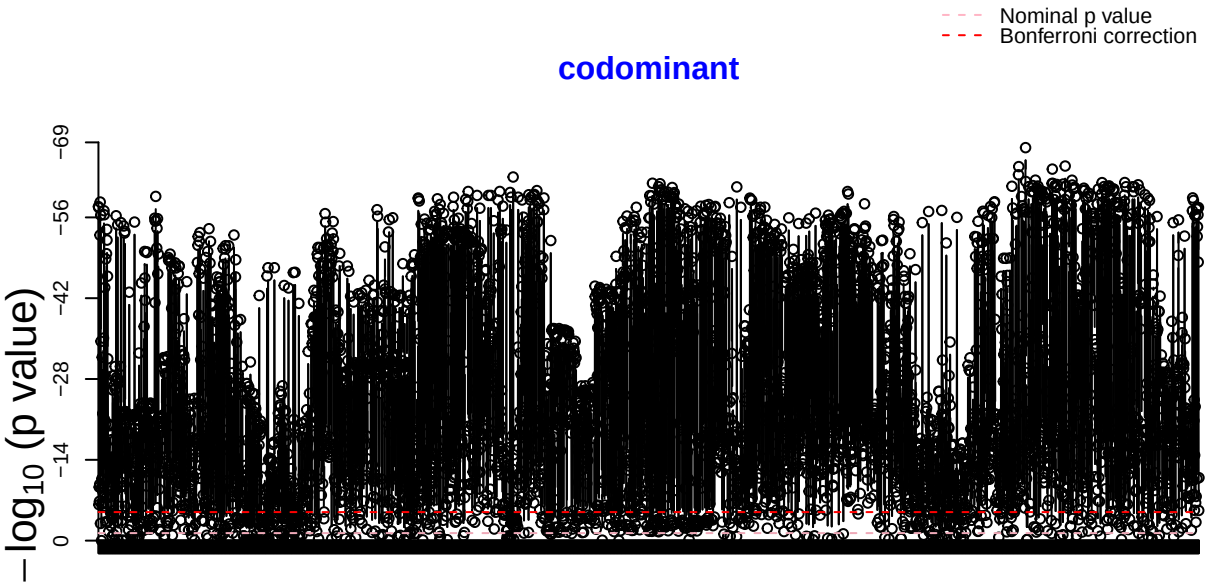
```
#you can ignore the warning
```

```
library(SNPassoc) #load the package that does the associations
```

```
## Loading required package: haplo.stats

## Loading required package: survival

## Loading required package: mvtnorm

## Loading required package: parallel
```

```
#if you get an error that the package is not available, use install.packages("SNPassoc") to install it.

#create new data frames containing only chromosome 3 information.
#grep() is the R version of the linux grep command that you saw in Ian's section.  So the command below
data.geno.pheno3 <- data.geno.pheno[,c(1:47,grep("X3_",colnames(data.geno.pheno)))]
snpinfo3 <- snpinfo[snpinfo$chr==3,]

#convert SNPinfo to a format that SNPassoc can use
snps3 <- setupSNP(data.geno.pheno3,48:ncol(data.geno.pheno3),sort=T,info=snpinfo3,sep="")
```

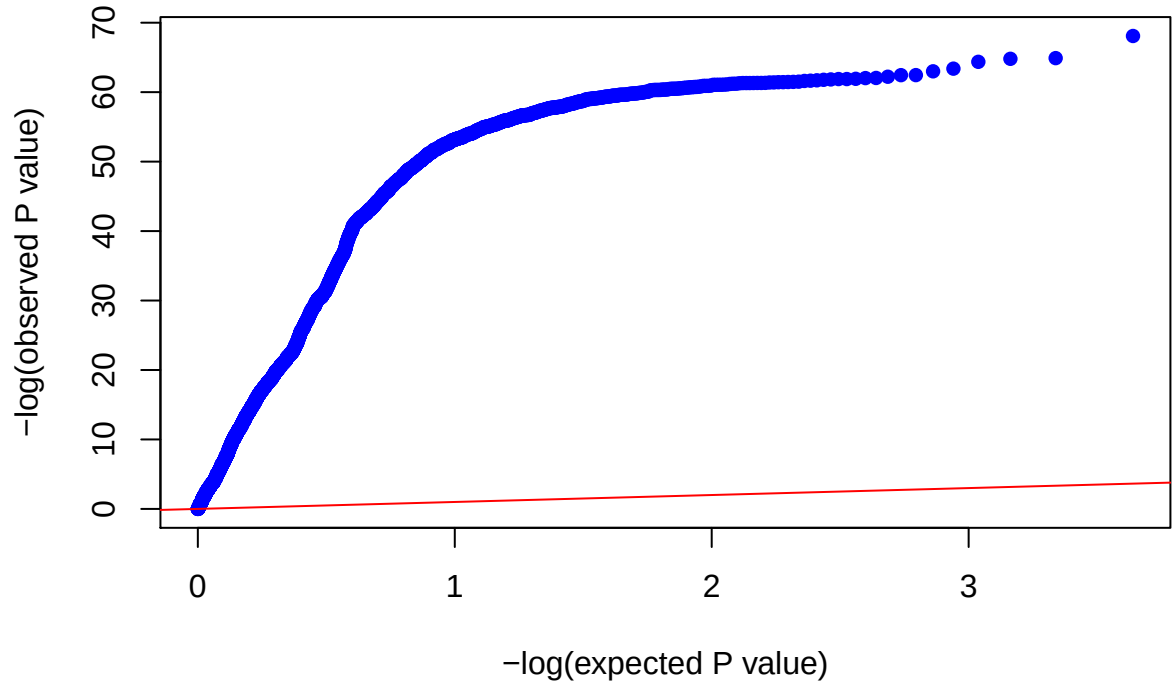
```
#analysis without population structure correction
#this takes ~ 5 minutes to run.
wg3 <- WGassociation(Alu.Tol,data=snps3,model="co",genotypingRate=50)
plot(wg3,print.label.SNPs=FALSE)
```




```
#the p-values for the co-dominant model are extracted by using the codominant() function
#determine the number of significant SNPs (p < 0.00001):
sum(codominant(wg3) < 1e-5)
```

```
## [1] 3664
```

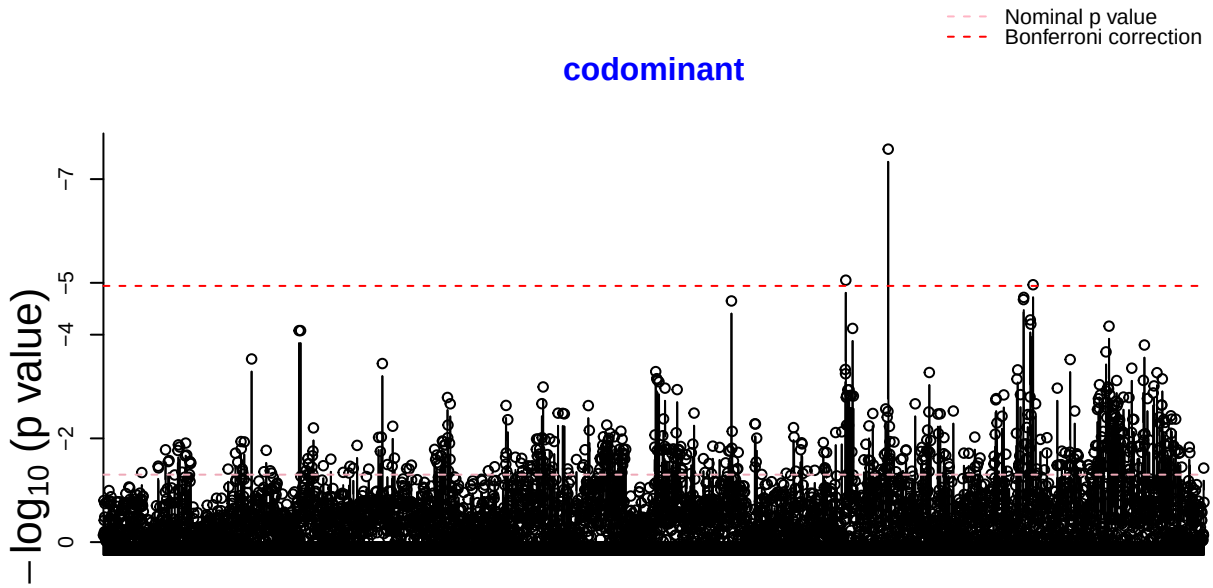
```
qqpval(codominant(wg3)) #remember that codominant(wg3) returns the observed p-values.
```



>

There is inflation in the observed P value vs the expected P value, which is good indication of population structure giving us false positives.

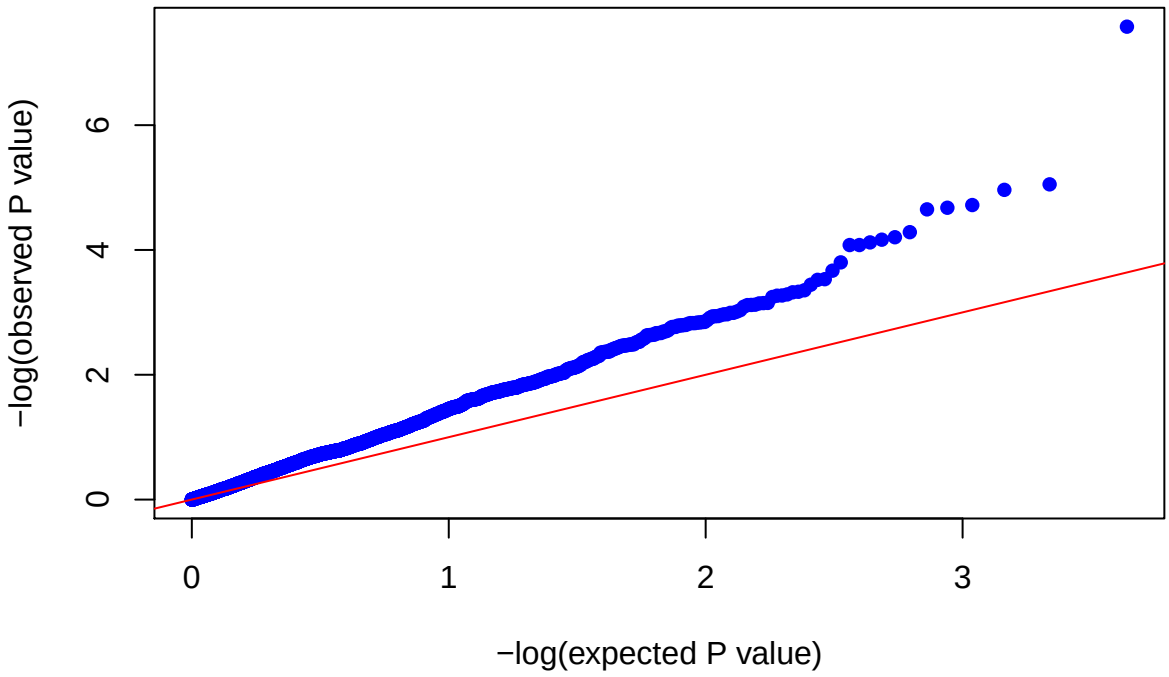
```
#analysis with population structure correction:
wg3.corrected <- WGassociation(Alu.Tol ~ pr1 + pr2 + pr3 + pr4,data=snps3,model="co",genotypingRate=50)

#analysis with population structure correction.
plot(wg3.corrected,print.label.SNPs=FALSE)
```




```
sum(codominant(wg3.corrected) < 1e-5)
```

```
## [1] 2
```

```
qqpval(codominant(wg3.corrected))
```



Exercise 7: Describe and discuss the differences in the analysis with and without population structure correction. Which do you think is the better one to follow-up on, and why?

> When there is population structure the results are inflated with false positive hits of significant SNPs. A population may have diverged and accumulated certain SNP genotypes which is not shared among other populations, thus we cannot compare these SNPs between populations. Taking population structure into account, we see that the results of observed P values are much closer to the expected P values. The difference between the two is significant as the previous test returned 3664 significant SNPs while the adjusted test returned 2 significant SNPs. It is better to study the SNPs returned by the test corrected for population structure as these SNPs are better representive of significant SNPs.

```
#use the square bracked extractions command to extract all rows where the SNP p-value is less than 1e-5
snpinfo3[codominant(wg3.corrected) < 1e-5,]
```

```
##                snp chr      pos
## 13279 X3_27639188   3 27639188
## 13447 X3_27753936   3 27753936
```

```
#if we want to add the pvals to the output:
cbind(snpinfo3[codominant(wg3.corrected) < 1e-5,],codominant(wg3.corrected)[codominant(wg3.corrected) <
```

```
##                snp chr      pos
## 13279 X3_27639188   3 27639188
## 13447 X3_27753936   3 27753936
##       codominant(wg3.corrected)[codominant(wg3.corrected) < 1e-05]
## 13279                                                 8.889402e-06
## 13447                                                 2.643791e-08
```

Exercise 8: Look for genes close to your SNP at the rice genome browser. Pick a significant SNP from your analysis and enter its chromosome and position in the search box. The browser wants you to enter a start and stop position, so for example, you should enter "Chr3:30449857..30449857" and then choose "show 20kb" from the pulldown menu on the left hand side. Report the SNP you chose and the three closest genes. These are candidate genes for determining the phenotype of your trait of interest in the rice population. Briefly discuss these genes as possible candidates for the GWAS peak. (Include a screenshot of the genome browser)

Search for SNP chr 3 position 27639188. The 3 closest genes I found were LOC_Os03g48480, LOS_Os03g48490, LOS_Os03g48520. Looking up the function of LOC_Os03g48480, the gene is involved in metabolic process, found in the peroxisome, and has a hydrolase activity.The proteins hydrolase activity may be playing some role in neutralizing or decreasing the toxicity of the aluminium. LOS_Os03g48490 is a centromeric protein that is involved with sequence specific DNA binding. This protein may be involved with activating certain genes that can increase resistance to aluminium. LOS_Os03g48520 is a protein that contains a protein binding domain. This protein could bind to another protein to form a complex that has a enzymatic function that neutralizes or decreases aluminium toxicity.