Exercises

Andy Pham

April 21, 2016

Exercise 1

```
sum(2000:20000)
```

```
## [1] 198011000
```

The sum is: 198011000

Exercise 2

In the code snippet, a is assigned the value 5, b is assigned the vector from that holds the numbers 2 to 20 incremented by 1. a and b are then printed out.

Exercise 3

```
a <- 5
b <- 2:20
sum(a, b)
```

```
## [1] 214
```

```
a + b
```

```
##  [1]  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
```

I get different results because sum adds all the numbers stored in both a and b while the "+" adds a to every element in b because b is a vector.

Exercise 4

```
sum(b[5:10])
```

```
## [1] 51
```

The sum is 51.

Exercise 5

```r
sum(b[c(3, 8, 10)])
```

```
## [1] 24
```

The sum is 24.

Exercise 6

```r
m <- matrix(data=1:25, ncol=5, byrow=T)
m
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    6    7    8    9   10
## [3,]   11   12   13   14   15
## [4,]   16   17   18   19   20
## [5,]   21   22   23   24   25
```

```r
c(m[, 3], m[, 4], m[, 5])
```

```
##  [1]  3  8 13 18 23  4  9 14 19 24  5 10 15 20 25
```

When extracting from a 2D object, the first number represents the row and the second number represents the column. m[3,] will return all the values in row 3. You extract the columns of m together using c(m[, 3], m[, 4], m[, 5]).

Exercise 7

```r
cbind(m, 101:105)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    2    3    4    5  101
## [2,]    6    7    8    9   10  102
## [3,]   11   12   13   14   15  103
## [4,]   16   17   18   19   20  104
## [5,]   21   22   23   24   25  105
```

```r
n <- 1:5
rbind(n, m)
```

```
##   [,1] [,2] [,3] [,4] [,5]
## n    1    2    3    4    5
##      1    2    3    4    5
##      6    7    8    9   10
##     11   12   13   14   15
##     16   17   18   19   20
##     21   22   23   24   25
```

The cbind command adds a column to the matrix specified with the data specified. In this case, cbind added a column to m with the numbers 101 to 105. Rbind would add another row to the matrix.

```r
fly.worm <- read.delim("../Data/fly2worm.blastp.gz", header=F)
#change the file name and path to match whatever you used.
#the header=F argument tells R that the first row contains data, not column names
#takes a while to read in this large data set.
head(fly.worm)
```

```
##         V1          V2    V3  V4  V5 V6 V7  V8 V9 V10   V11 V12
## 1 2RSSE.1a FBpp0304402 48.35 242 122  2 19 258 31 271 9e-77 242
## 2 2RSSE.1a FBpp0304403 48.35 242 122  2 19 258 31 271 2e-75 243
## 3 2RSSE.1a FBpp0292544 48.35 242 122  2 19 258 31 271 3e-74 243
## 4 2RSSE.1a FBpp0075321 48.35 242 122  2 19 258 31 271 5e-74 243
## 5 2RSSE.1a FBpp0075320 48.35 242 122  2 19 258 31 271 5e-74 243
## 6 2RSSE.1a FBpp0075322 48.35 242 122  2 19 258 31 271 3e-72 243
```

```r
summary(fly.worm)
```

```
##         V1                   V2               V3              V4
##  F15G9.4d:  3909   FBpp0304926:  1080   Min.   :17.97   Min.   :  32.0
##  F15G9.4a:  3745   FBpp0304924:  1078   1st Qu.:26.17   1st Qu.: 203.0
##  C09D1.1b:  3715   FBpp0304923:  1074   Median :29.56   Median : 268.0
##  F15G9.4b:  3687   FBpp0304921:  1072   Mean   :31.83   Mean   : 343.1
##  C09D1.1g:  3617   FBpp0304925:  1070   3rd Qu.:34.62   3rd Qu.: 387.0
##  C09D1.1f:  3545   FBpp0304929:  1069   Max.   :99.33   Max.   :5795.0
##  (Other) :481883   (Other)    :497658
##        V5               V6                V7                V8
##  Min.   :   1.0   Min.   :  0.000   Min.   :    1.0   Min.   :   33
##  1st Qu.: 121.0   1st Qu.:  4.000   1st Qu.:   38.0   1st Qu.:  285
##  Median : 162.0   Median :  7.000   Median :  129.0   Median :  433
##  Mean   : 203.7   Mean   :  9.836   Mean   :  716.2   Mean   : 1039
##  3rd Qu.: 227.0   3rd Qu.: 11.000   3rd Qu.:  444.0   3rd Qu.:  789
##  Max.   :3161.0   Max.   :172.000   Max.   :18345.0   Max.   :18531
##
##        V9               V10               V11                 V12
##  Min.   :    1.0   Min.   :   33.0   Min.   :0.000e+00   Min.   :  55.5
##  1st Qu.:   48.0   1st Qu.:  322.0   1st Qu.:0.000e+00   1st Qu.:  79.0
##  Median :  208.0   Median :  515.0   Median :0.000e+00   Median :  98.2
##  Mean   :  647.7   Mean   :  971.7   Mean   :2.205e-12   Mean   : 135.1
##  3rd Qu.:  544.0   3rd Qu.:  909.0   3rd Qu.:4.000e-15   3rd Qu.: 139.0
##  Max.   :22596.0   Max.   :22891.0   Max.   :1.000e-10   Max.   :5266.0
##
```

```r
colnames(fly.worm) <- c("qid", "sid", "pct", "len", "mis", "gaps", "qb", "qe", "sb", "se", "E", "S")
head(fly.worm)
```

```
##        qid         sid   pct len mis gaps qb  qe sb  se     E   S
## 1 2RSSE.1a FBpp0304402 48.35 242 122    2 19 258 31 271 9e-77 242
## 2 2RSSE.1a FBpp0304403 48.35 242 122    2 19 258 31 271 2e-75 243
## 3 2RSSE.1a FBpp0292544 48.35 242 122    2 19 258 31 271 3e-74 243
## 4 2RSSE.1a FBpp0075321 48.35 242 122    2 19 258 31 271 5e-74 243
## 5 2RSSE.1a FBpp0075320 48.35 242 122    2 19 258 31 271 5e-74 243
## 6 2RSSE.1a FBpp0075322 48.35 242 122    2 19 258 31 271 3e-72 243
```

```r
summary(fly.worm)
```

```
##        qid                  sid              pct             len
##  F15G9.4d:  3909   FBpp0304926:  1080   Min.   :17.97   Min.   :  32.0
##  F15G9.4a:  3745   FBpp0304924:  1078   1st Qu.:26.17   1st Qu.: 203.0
##  C09D1.1b:  3715   FBpp0304923:  1074   Median :29.56   Median : 268.0
##  F15G9.4b:  3687   FBpp0304921:  1072   Mean   :31.83   Mean   : 343.1
##  C09D1.1g:  3617   FBpp0304925:  1070   3rd Qu.:34.62   3rd Qu.: 387.0
##  C09D1.1f:  3545   FBpp0304929:  1069   Max.   :99.33   Max.   :5795.0
##  (Other) :481883   (Other)    :497658
##       mis              gaps               qb                qe
##  Min.   :   1.0   Min.   :  0.000   Min.   :    1.0   Min.   :   33
##  1st Qu.: 121.0   1st Qu.:  4.000   1st Qu.:   38.0   1st Qu.:  285
##  Median : 162.0   Median :  7.000   Median :  129.0   Median :  433
##  Mean   : 203.7   Mean   :  9.836   Mean   :  716.2   Mean   : 1039
##  3rd Qu.: 227.0   3rd Qu.: 11.000   3rd Qu.:  444.0   3rd Qu.:  789
##  Max.   :3161.0   Max.   :172.000   Max.   :18345.0   Max.   :18531
##
##        sb                se                E                   S
##  Min.   :    1.0   Min.   :   33.0   Min.   :0.000e+00   Min.   :  55.5
##  1st Qu.:   48.0   1st Qu.:  322.0   1st Qu.:0.000e+00   1st Qu.:  79.0
##  Median :  208.0   Median :  515.0   Median :0.000e+00   Median :  98.2
##  Mean   :  647.7   Mean   :  971.7   Mean   :2.205e-12   Mean   : 135.1
##  3rd Qu.:  544.0   3rd Qu.:  909.0   3rd Qu.:4.000e-15   3rd Qu.: 139.0
##  Max.   :22596.0   Max.   :22891.0   Max.   :1.000e-10   Max.   :5266.0
##
```

Exercise 8

```r
evalue <- sum(fly.worm["E"] == 0)
pID <- sum(fly.worm["pct"] > 50)
evalue_percent <- sum(fly.worm["E"] == 0)/nrow(fly.worm)
percentID <- sum(fly.worm["pct"] > 50)/nrow(fly.worm)
evalue_pID <- sum(fly.worm["E"] == 0 & fly.worm["pct"] < 50)
minpID <- subset(fly.worm, E == 0)
min(minpID["pct"])
```

```
## [1] 24.07
```

E-value of 0: 7531 hits, 0.01494 **Percent Identity > 50**: 20928 hits, 0.04151 **E-value of 0 and Percent Identity < 50**: 3351 hits **Minimum Percent Identity of Hits with E-value of 0**: 24.07

Exercise 9

I am surprised that low percent identity sequences can still have an E-value of 0. The alignment property that affects the E-value being 0 even when the percent identity is less than 50 is raw score. The higher the raw score, the lower the e-value. Thus, when the raw score is high enough, the e-value is small enough to get rounded to 0.

```
mp <- cbind(fly.worm["E"], fly.worm["S"])
summary(mp)
```

```
##        E                    S          
##  Min.   :0.000e+00   Min.   :  55.5  
##  1st Qu.:0.000e+00   1st Qu.:  79.0  
##  Median :0.000e+00   Median :  98.2  
##  Mean   :2.205e-12   Mean   : 135.1  
##  3rd Qu.:4.000e-15   3rd Qu.: 139.0  
##  Max.   :1.000e-10   Max.   :5266.0  
```

```
mp_small <- subset(mp, E == 0)
mp_big <- subset(mp, E > 0)
mp_middle <- subset(mp, E < 2e-12)
mp_middle2 <- subset(mp, E < 2e-14)
summary(mp_small)
```

```
##        E           S       
##  Min.   :0   Min.   : 511  
##  1st Qu.:0   1st Qu.: 627  
##  Median :0   Median : 766  
##  Mean   :0   Mean   : 961  
##  3rd Qu.:0   3rd Qu.:1078  
##  Max.   :0   Max.   :5266  
```

```
summary(mp_big)
```

```
##        E                    S        
##  Min.   :0.000e+00   Min.   : 55.5  
##  1st Qu.:0.000e+00   1st Qu.: 79.0  
##  Median :0.000e+00   Median : 97.4  
##  Mean   :2.238e-12   Mean   :122.5  
##  3rd Qu.:5.000e-15   3rd Qu.:137.0  
##  Max.   :1.000e-10   Max.   :632.0  
```

```
summary(mp_middle)
```

```
##        E                    S          
##  Min.   :0.000e+00   Min.   :  60.5  
##  1st Qu.:0.000e+00   1st Qu.:  84.3  
##  Median :0.000e+00   Median : 103.0  
##  Mean   :4.081e-14   Mean   : 142.8  
##  3rd Qu.:1.000e-16   3rd Qu.: 147.0  
##  Max.   :1.000e-12   Max.   :5266.0  
```

```
summary(mp_middle2)
```

```
##        E                    S          
##  Min.   :0.000e+00   Min.   :  64.3  
##  1st Qu.:0.000e+00   1st Qu.:  90.9  
##  Median :0.000e+00   Median : 112.0  
##  Mean   :4.136e-16   Mean   : 153.1  
##  3rd Qu.:1.000e-18   3rd Qu.: 156.0  
##  Max.   :1.000e-14   Max.   :5266.0
```

As the summaries show, the lower the E-values become, the higher the mean of the raw score “S” becomes. When E is 0 as shown in mp_small, the mean of the raw scores is many times higher than that if E is greater than 0 as seen in mp_big. Then as the high bounds for E decreeases, the mean of the raw score increases as seen in mp_middle and mp_middle2.

Exercise 10

```r
fly.worm.small <- sample(nrow(fly.worm), 10000)
summary(fly.worm[fly.worm.small, ])
```

```
##        qid                  sid             pct              len        
##  C09D1.1e:  86   FBpp0306552:  28   Min.   :18.63   Min.   :  42.0  
##  C09D1.1g:  82   FBpp0304923:  26   1st Qu.:26.18   1st Qu.: 203.0  
##  C09D1.1a:  74   FBpp0304926:  26   Median :29.67   Median : 268.0  
##  C09D1.1b:  73   FBpp0304920:  25   Mean   :31.90   Mean   : 341.7  
##  F15G9.4b:  73   FBpp0304924:  25   3rd Qu.:34.68   3rd Qu.: 386.2  
##  F15G9.4d:  73   FBpp0304929:  23   Max.   :98.55   Max.   :4506.0  
##  (Other) :9539   (Other)    :9847                                   
##       mis              gaps               qb                 qe       
##  Min.   :   2.0   Min.   :  0.000   Min.   :    1.0   Min.   :   42  
##  1st Qu.: 121.0   1st Qu.:  4.000   1st Qu.:   38.0   1st Qu.:  286  
##  Median : 162.0   Median :  7.000   Median :  125.0   Median :  426  
##  Mean   : 202.6   Mean   :  9.787   Mean   :  721.2   Mean   : 1043  
##  3rd Qu.: 227.0   3rd Qu.: 11.000   3rd Qu.:  433.2   3rd Qu.:  773  
##  Max.   :2678.0   Max.   :125.000   Max.   :18345.0   Max.   :18522  
##                                                                     
##        sb                 se                  E                    S         
##  Min.   :    1.0   Min.   :   65.0   Min.   :0.000e+00   Min.   :  59.7  
##  1st Qu.:   49.0   1st Qu.:  324.0   1st Qu.:0.000e+00   1st Qu.:  79.7  
##  Median :  216.5   Median :  521.0   Median :0.000e+00   Median :  99.0  
##  Mean   :  654.4   Mean   :  977.2   Mean   :2.146e-12   Mean   : 135.6  
##  3rd Qu.:  551.0   3rd Qu.:  913.0   3rd Qu.:3.000e-15   3rd Qu.: 140.0  
##  Max.   :18735.0   Max.   :19531.0   Max.   :1.000e-10   Max.   :2773.0  
## 
```

Exercise 11

```r
stereotype_db <- read.delim("stereotypes.csv", sep =",")
summary(stereotype_db)
```

```
##      population     gender        coffee         computer    
##  hippie   :200   female:400   Min.   : 0.00   Min.   : 0.00  
##  hipster  :199   male  :399   1st Qu.: 6.00   1st Qu.:12.00  
##  metalhead:200                Median :12.00   Median :24.00
##  nerd     :200                 Mean   :13.84   Mean   :28.65
##                                3rd Qu.:22.00   3rd Qu.:43.00
##                                Max.   :36.00   Max.   :95.00
##     shower            beer           tacos            age
##  Min.   : 0.000   Min.   : 0.00   Min.   : 0.00   Min.   :17.00
##  1st Qu.: 4.000   1st Qu.: 6.00   1st Qu.:11.00   1st Qu.:20.00
##  Median : 7.000   Median :17.00   Median :14.00   Median :22.00
##  Mean   : 6.607   Mean   :21.13   Mean   :14.03   Mean   :22.34
##  3rd Qu.: 9.000   3rd Qu.:33.00   3rd Qu.:17.00   3rd Qu.:24.00
##  Max.   :18.000   Max.   :80.00   Max.   :42.00   Max.   :27.00
```

```
head(stereotype_db)
```

```
##   population gender coffee computer shower beer tacos age
## 1     hippie female      1        1      8    0     8  17
## 2       nerd female      2       13      7    0     4  17
## 3       nerd female      6       40      8    0     5  17
## 4       nerd female      3       41     10    0     6  17
## 5       nerd female     15       42      9    0    11  17
## 6       nerd female      6       42      9    0     6  17
```

Activity Break for Intro to Plotting

AB1-1

```
nerds_and_metal <- subset(stereotype_db, population == "nerd" | population == "metalhead")
summary(nerds_and_metal)
```

```
##      population     gender        coffee         computer
##  hippie   :  0   female:200   Min.   : 0.00   Min.   : 1.00
##  hipster  :  0   male  :200   1st Qu.: 5.00   1st Qu.:16.00
##  metalhead:200                Median :11.00   Median :24.50
##  nerd     :200                Mean   :11.23   Mean   :33.24
##                               3rd Qu.:17.00   3rd Qu.:51.00
##                               Max.   :30.00   Max.   :95.00
##     shower          beer           tacos            age
##  Min.   : 0.00   Min.   : 0.00   Min.   : 0.00   Min.   :17.00
##  1st Qu.: 4.00   1st Qu.: 4.00   1st Qu.: 8.00   1st Qu.:20.00
##  Median : 8.00   Median : 7.50   Median :14.00   Median :22.00
##  Mean   : 7.28   Mean   :20.23   Mean   :14.49   Mean   :22.37
##  3rd Qu.:10.00   3rd Qu.:37.00   3rd Qu.:19.00   3rd Qu.:24.00
##  Max.   :18.00   Max.   :80.00   Max.   :42.00   Max.   :27.00
```

AB1-2

```
males.only <- subset(stereotype_db, gender == "male" & beer > 25 & tacos > 20)
summary(males.only)
```

```
##      population    gender       coffee         computer    
##  hippie   : 0   female: 0   Min.   : 6.00   Min.   :11.00  
##  hipster  : 4   male  :72   1st Qu.:16.75   1st Qu.:16.00  
##  metalhead:68               Median :19.00   Median :18.00  
##  nerd     : 0               Mean   :19.29   Mean   :19.19  
##                             3rd Qu.:22.00   3rd Qu.:21.00  
##                             Max.   :31.00   Max.   :48.00  
##      shower            beer           tacos            age       
##  Min.   : 0.000   Min.   :26.00   Min.   :21.00   Min.   :19.00  
##  1st Qu.: 2.000   1st Qu.:36.00   1st Qu.:23.00   1st Qu.:21.00  
##  Median : 4.000   Median :43.00   Median :25.00   Median :22.00  
##  Mean   : 4.722   Mean   :45.47   Mean   :26.15   Mean   :21.81  
##  3rd Qu.: 6.000   3rd Qu.:56.00   3rd Qu.:28.00   3rd Qu.:23.00  
##  Max.   :16.000   Max.   :80.00   Max.   :42.00   Max.   :25.00
```

There are more males in the metalhead category who binge drink and eat over 20 tacos a week.

AB1-3

```
print("How many females total spend more than 2 hours on the computer and drink more than 12 cups of co
```

```
## [1] "How many females total spend more than 2 hours on the computer and drink more than 12 cups of c
```

```
females.ask <- subset(stereotype_db, gender == "female" & computer > 2 & coffee > 12)
summary(females.ask)
```

```
##      population    gender        coffee         computer    
##  hippie   : 2   female:161   Min.   :13.00   Min.   : 3.00  
##  hipster  :99   male  :  0   1st Qu.:17.00   1st Qu.:16.00  
##  metalhead:55                Median :23.00   Median :37.00  
##  nerd     : 5                Mean   :22.08   Mean   :30.76  
##                              3rd Qu.:26.00   3rd Qu.:41.00  
##                              Max.   :36.00   Max.   :74.00  
##      shower            beer           tacos           age       
##  Min.   : 0.000   Min.   : 0.00   Min.   : 1.0   Min.   :17.00  
##  1st Qu.: 5.000   1st Qu.:17.00   1st Qu.:13.0   1st Qu.:20.00  
##  Median : 8.000   Median :28.00   Median :15.0   Median :22.00  
##  Mean   : 7.391   Mean   :28.58   Mean   :14.8   Mean   :22.22  
##  3rd Qu.:10.000   3rd Qu.:39.00   3rd Qu.:17.0   3rd Qu.:23.00  
##  Max.   :18.000   Max.   :73.00   Max.   :20.0   Max.   :27.00
```

161 females total.

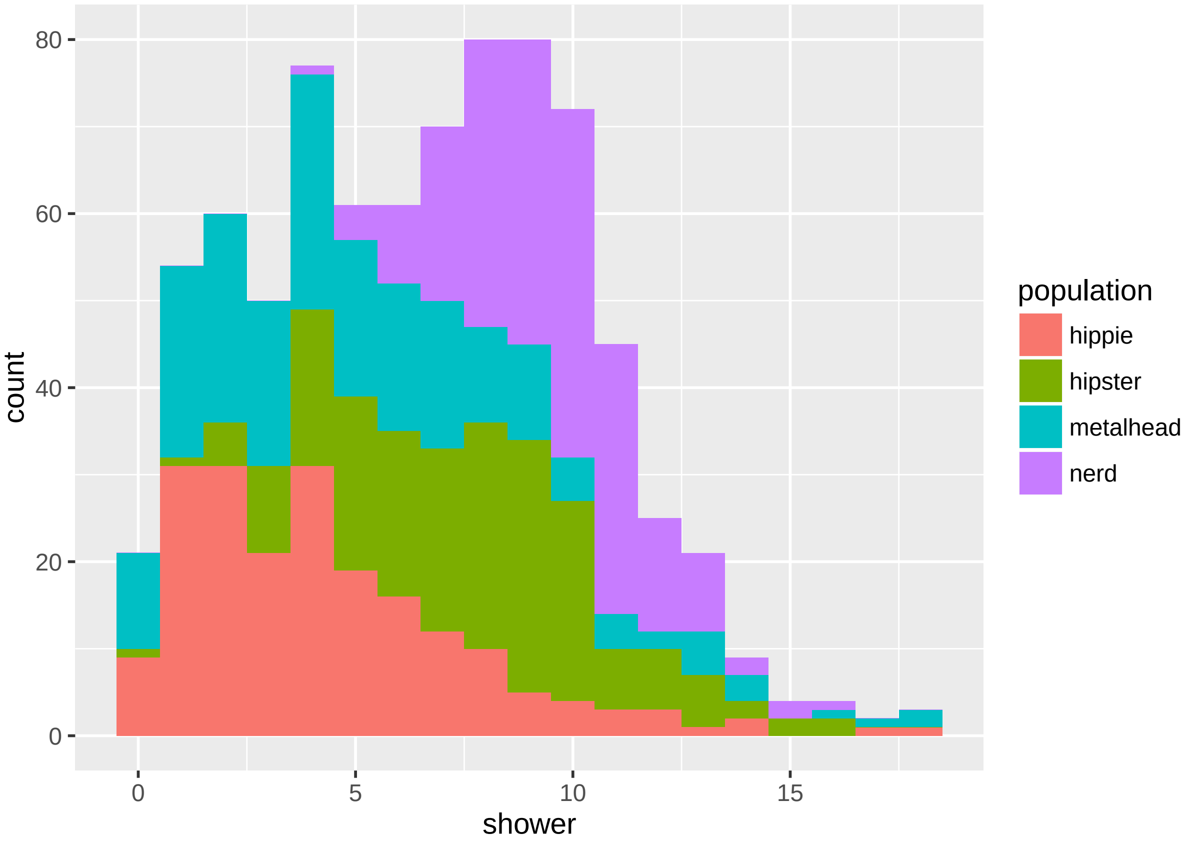
AB_4:

In R, “==” means if left hand side is equal to right side, then return true, else return false.

Visualization with ggplot2 (Plotting tutorial)

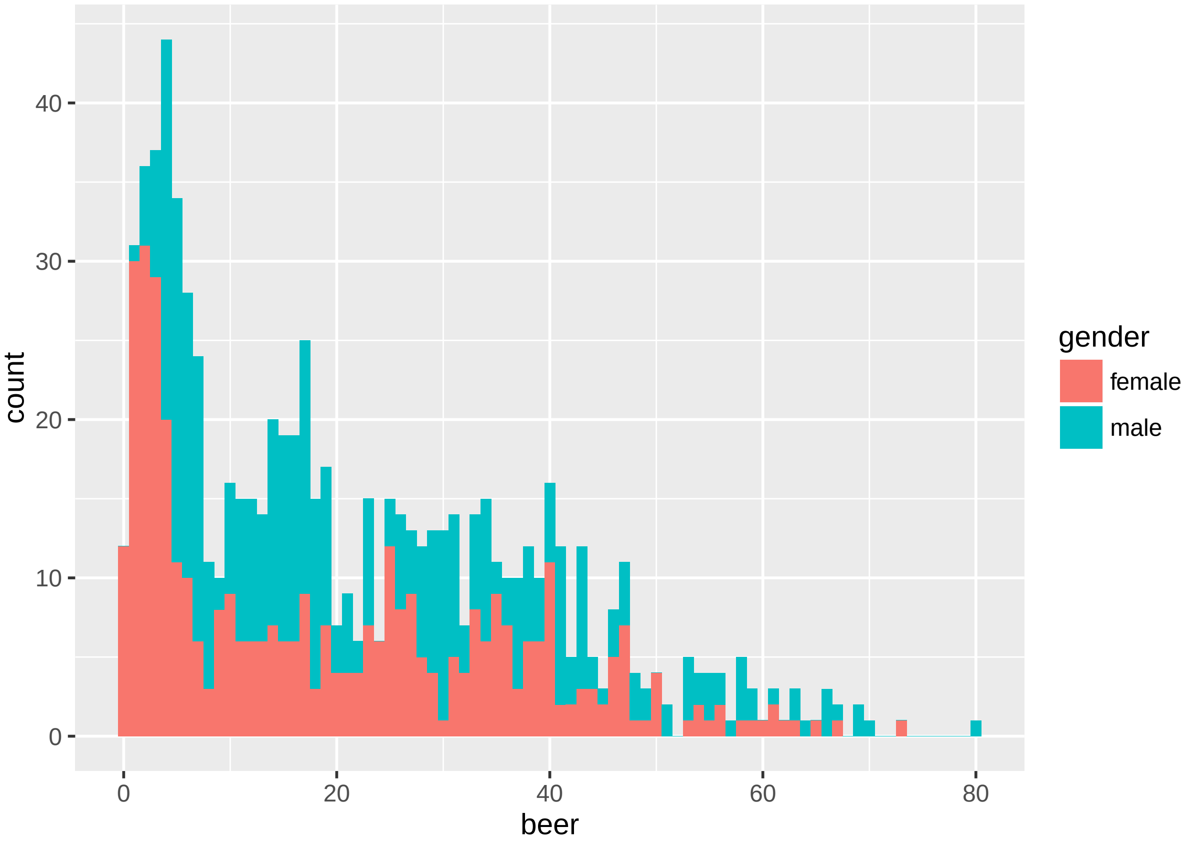
AB2-1

```
qplot(shower, data=stereotype_db, geom="histogram", fill=population, binwidth= 1)
```



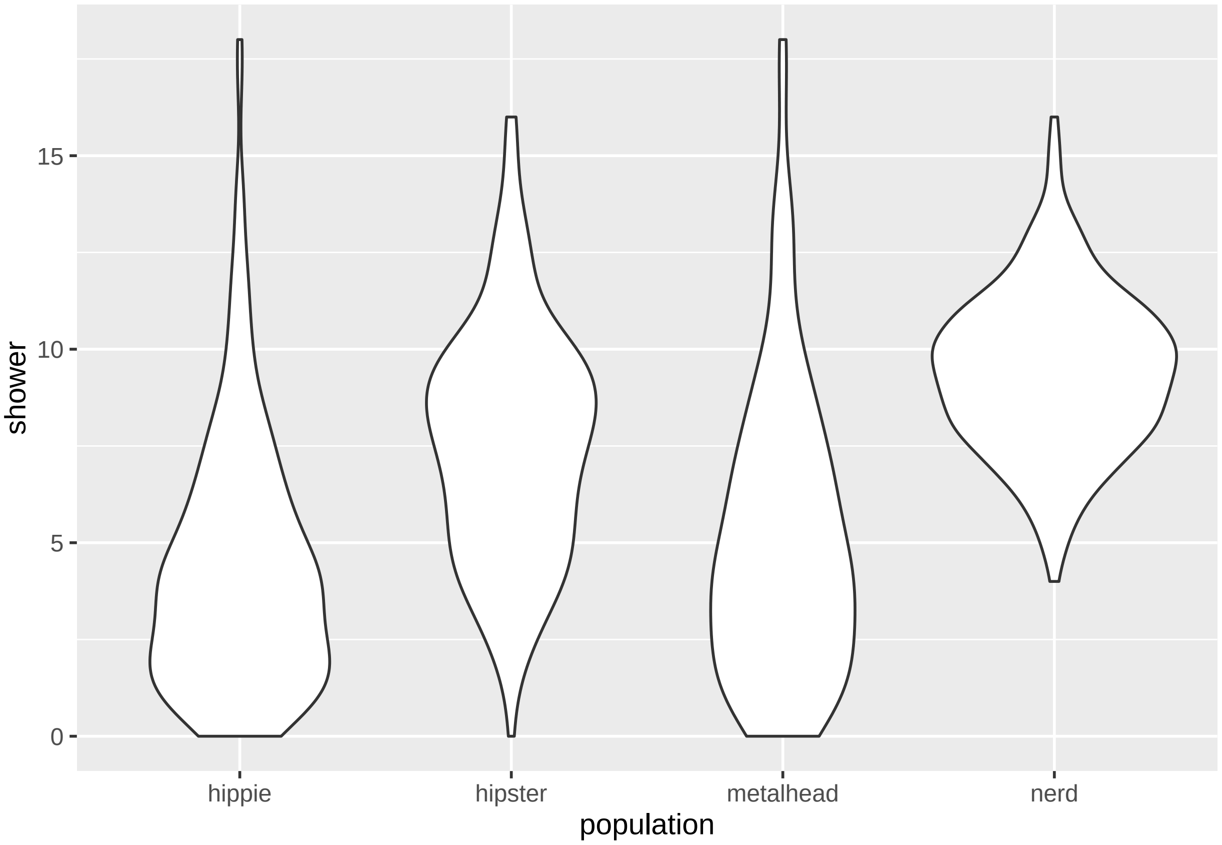
AB2-2

```
qplot(beer, data=stereotype_db, geom="histogram", fill=gender, binwidth= 1)
```



AB2-3

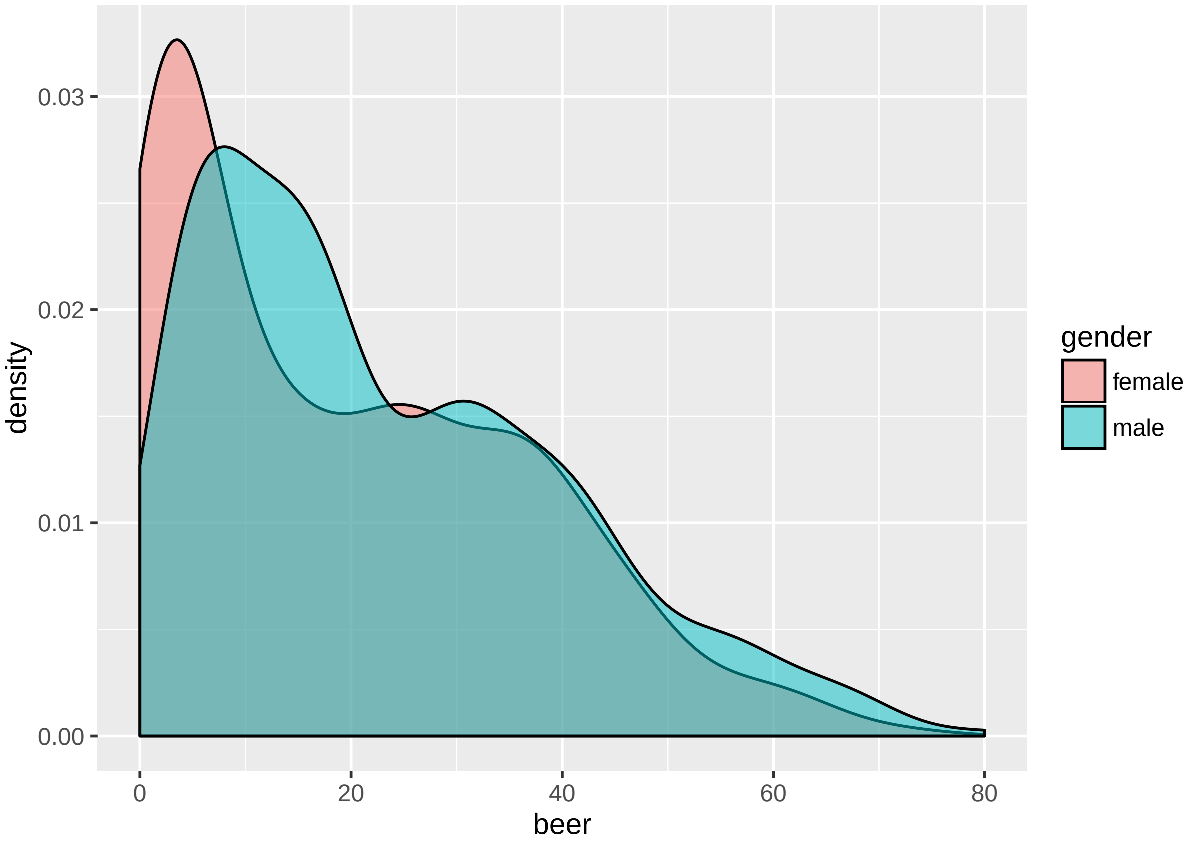
```
ggplot(stereotype_db, aes(population, shower)) + geom_violin()
```



Combining Subset with Visualization (Plotting tutorial)

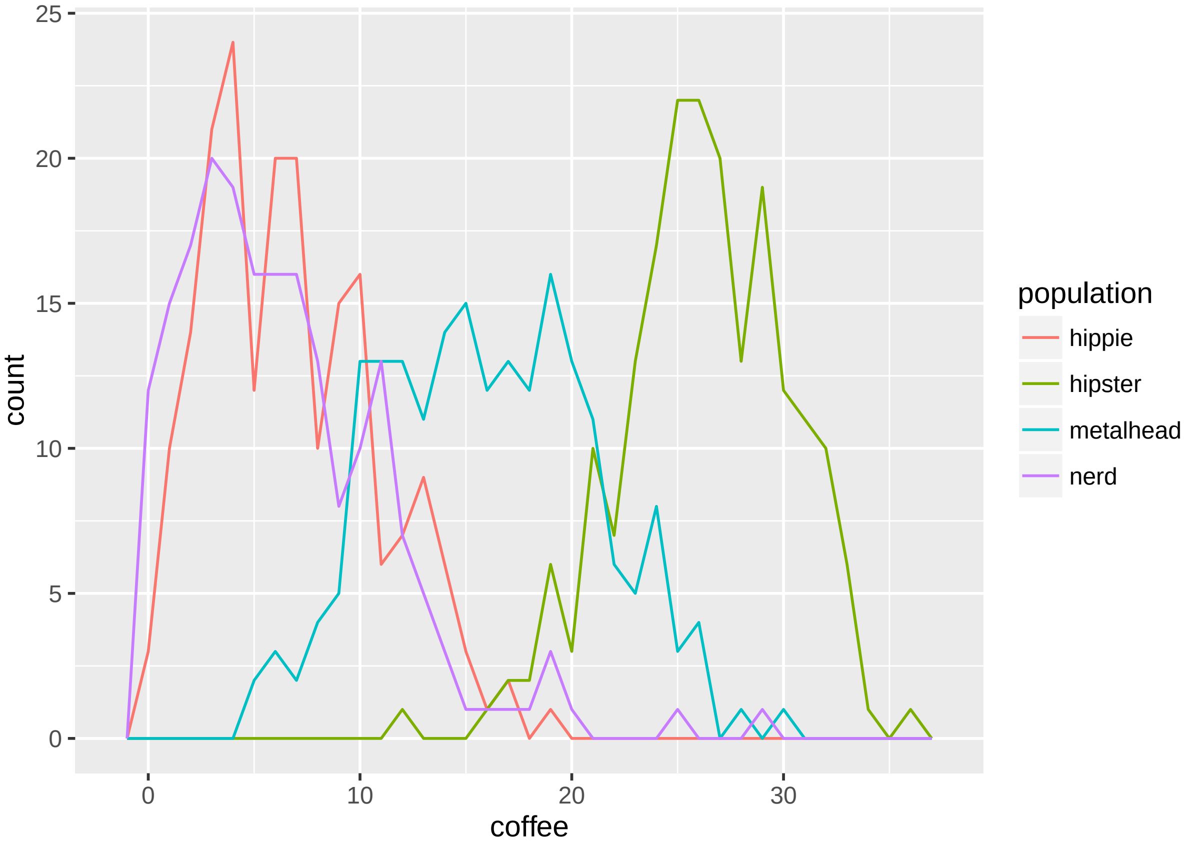
AB3-1

```
nerds_only <- subset(stereotype_db, population == "nerd")
qplot(beer, data=stereotype_db, geom ="density", fill=gender, alpha=I(0.5))
```

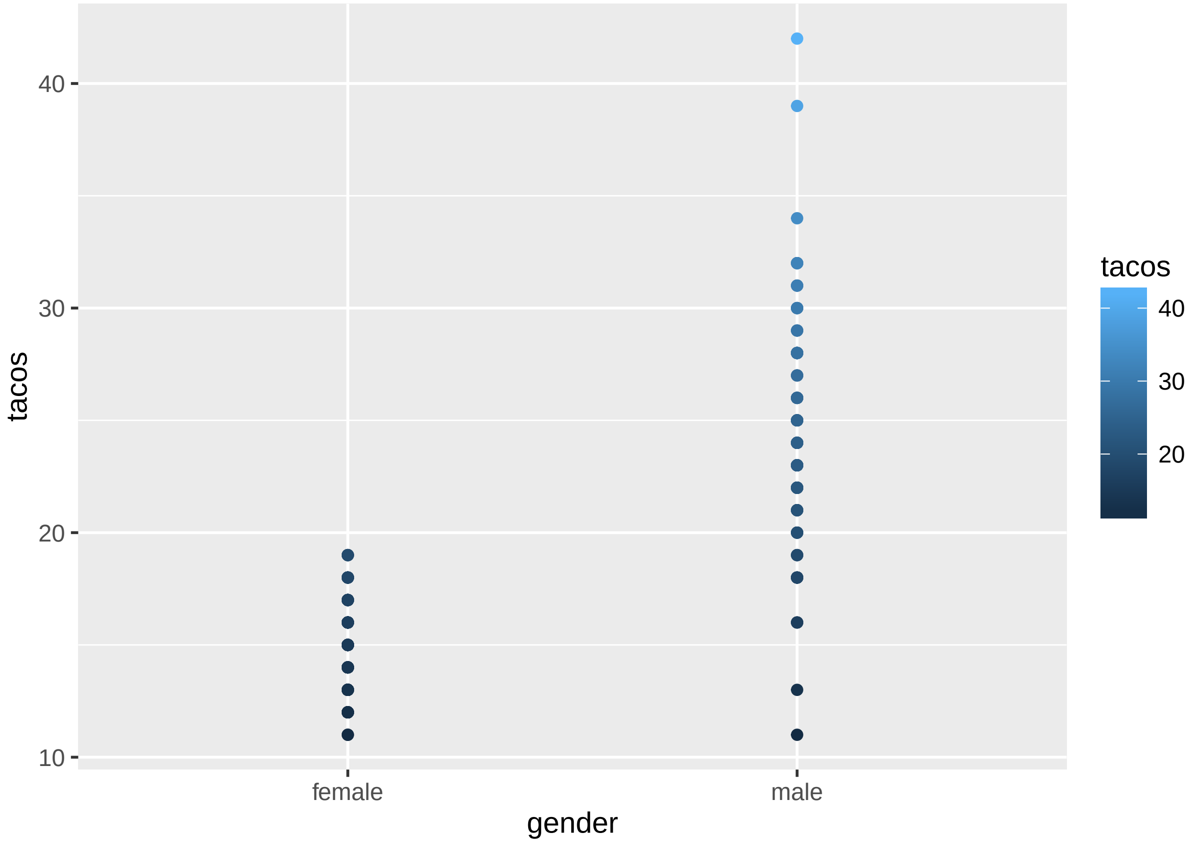


AB3-2

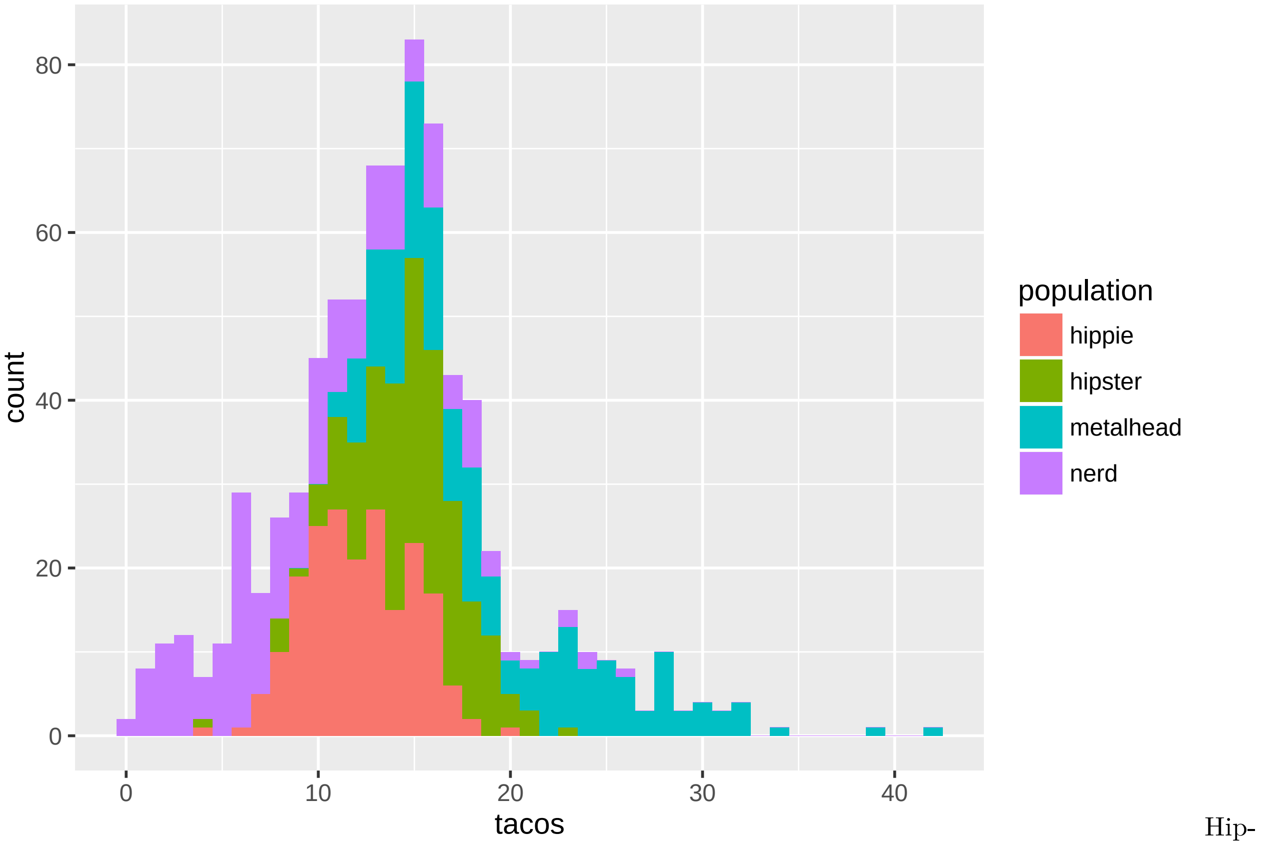
```
qplot(coffee, data=stereotype_db, geom="freqpoly", colour = population, binwidth = 1)
```



```
metal <- subset(stereotype_db, population == "metalhead")
qplot(gender, tacos, data=metal, geom = "point", colour=tacos)
```



```
qplot(tacos,  data=stereotype_db, geom = "histogram", fill=population, binwidth=1)
```



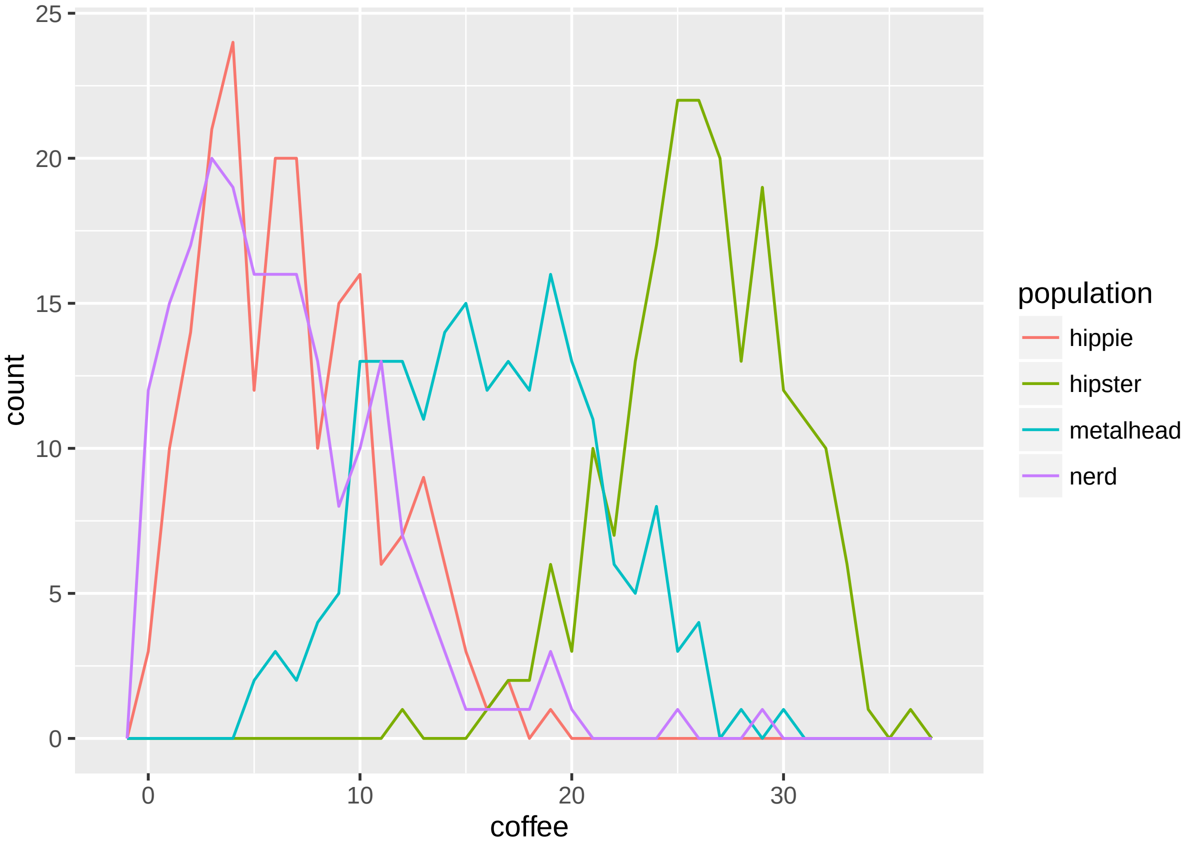
Hipsters consume the most coffee per individual out of the entire population. Of the metalheads, the males eat the most tacos and have a wider distribution. Hippies eat the least amount of tacos.

AB3-3

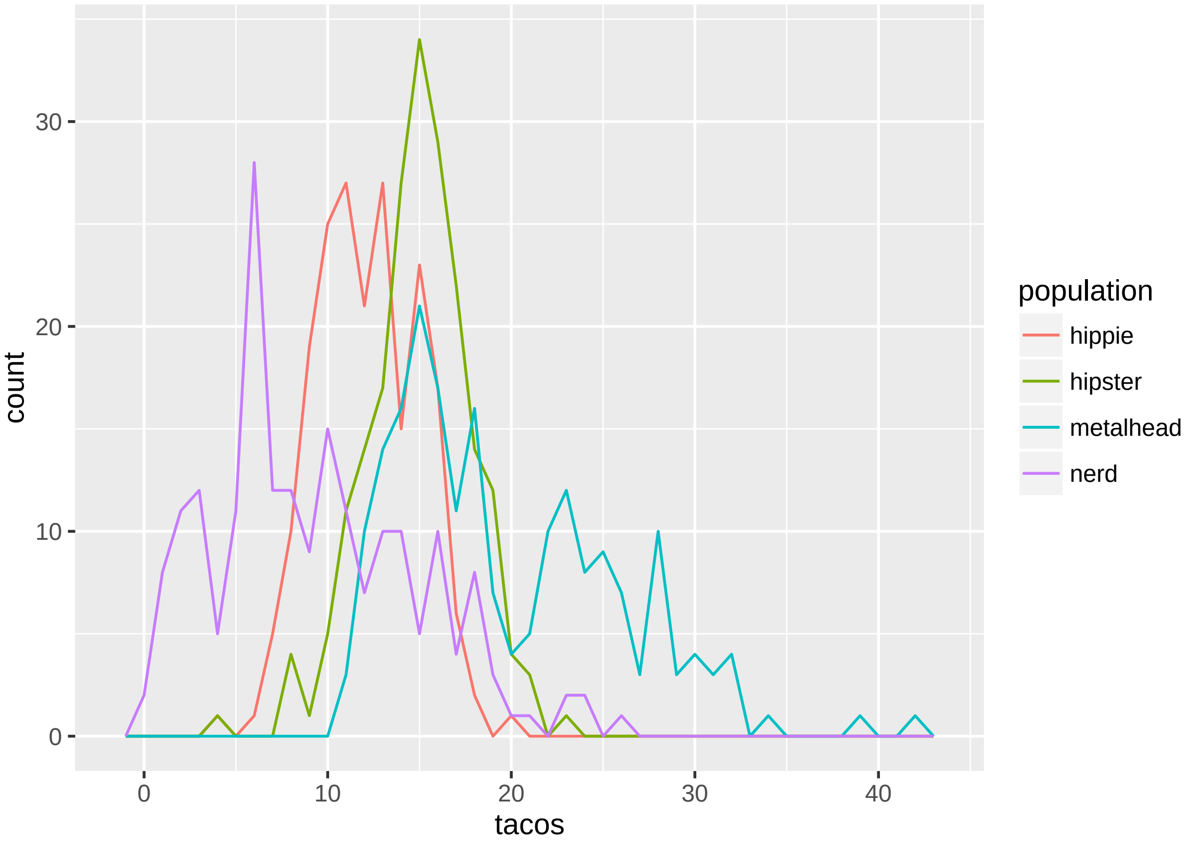
```
summary(stereotype_db)
```

```
##      population     gender        coffee         computer    
##  hippie   :200   female:400   Min.   : 0.00   Min.   : 0.00  
##  hipster  :199   male  :399   1st Qu.: 6.00   1st Qu.:12.00  
##  metalhead:200                Median :12.00   Median :24.00  
##  nerd     :200                Mean   :13.84   Mean   :28.65  
##                               3rd Qu.:22.00   3rd Qu.:43.00  
##                               Max.   :36.00   Max.   :95.00  
##      shower            beer           tacos            age       
##  Min.   : 0.000   Min.   : 0.00   Min.   : 0.00   Min.   :17.00  
##  1st Qu.: 4.000   1st Qu.: 6.00   1st Qu.:11.00   1st Qu.:20.00  
##  Median : 7.000   Median :17.00   Median :14.00   Median :22.00  
##  Mean   : 6.607   Mean   :21.13   Mean   :14.03   Mean   :22.34  
##  3rd Qu.: 9.000   3rd Qu.:33.00   3rd Qu.:17.00   3rd Qu.:24.00  
##  Max.   :18.000   Max.   :80.00   Max.   :42.00   Max.   :27.00
```

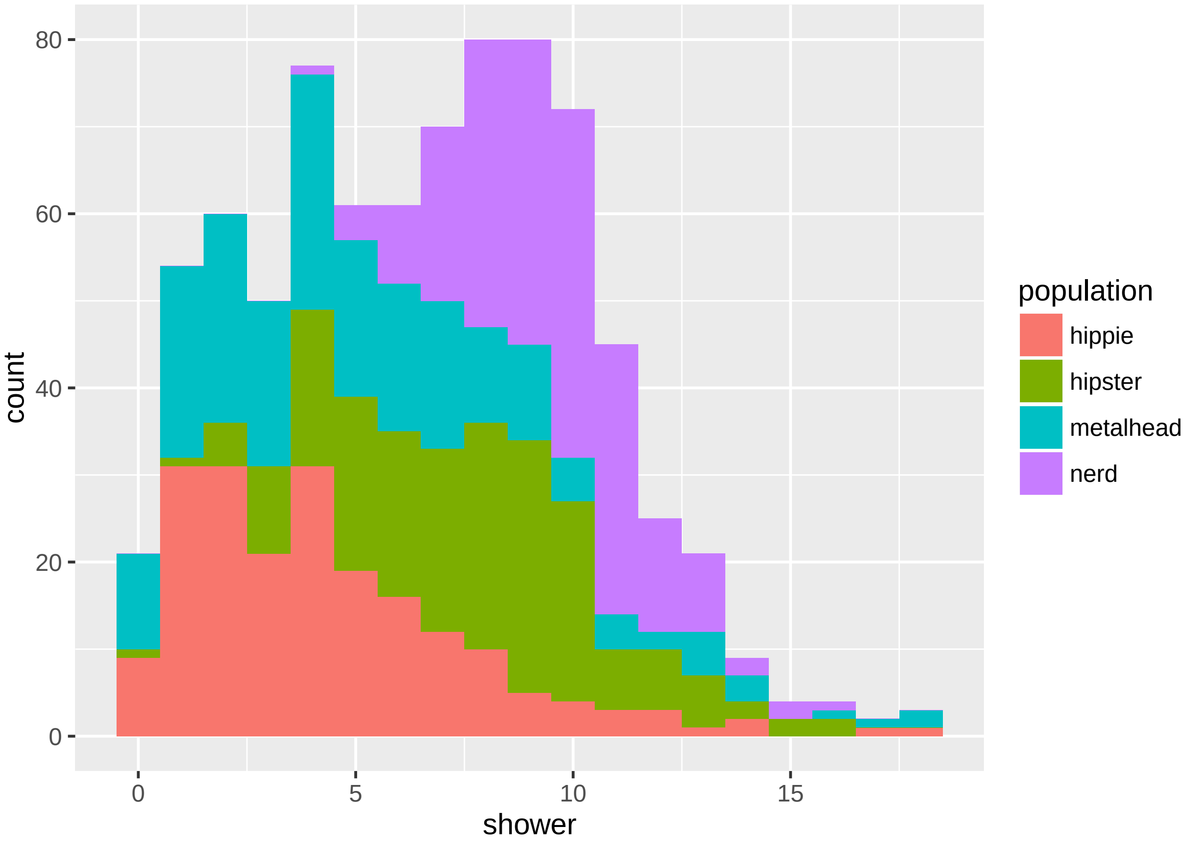
```
qplot(coffee, data=stereotype_db, geom ="freqpoly", colour=population, binwidth=1)
```



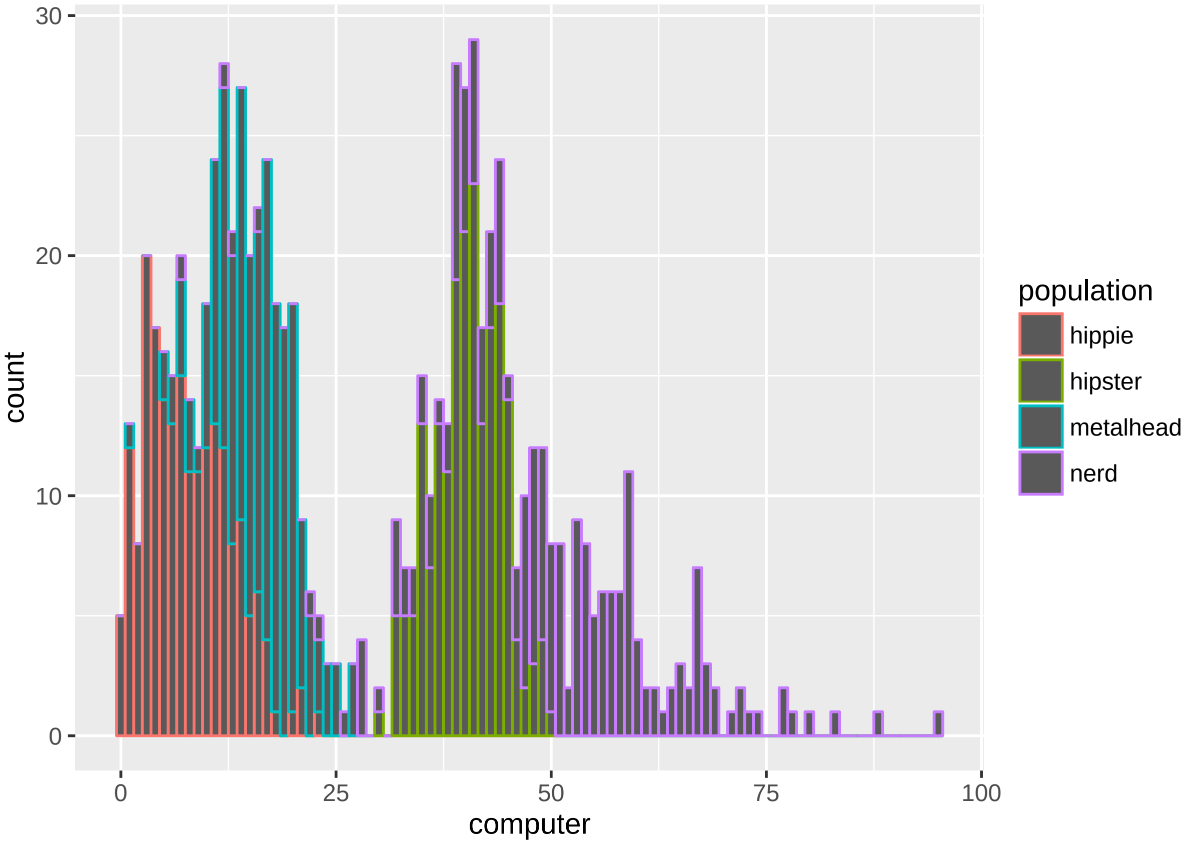
```
qplot(tacos, data=stereotype_db, geom ="freqpoly", colour=population, binwidth=1)
```



```
qplot(shower, data=stereotype_db, geom="histogram", fill=population, binwidth=1)
```



```
qplot(computer, data=stereotype_db, geom="histogram", colour = population, binwidth=1)
```

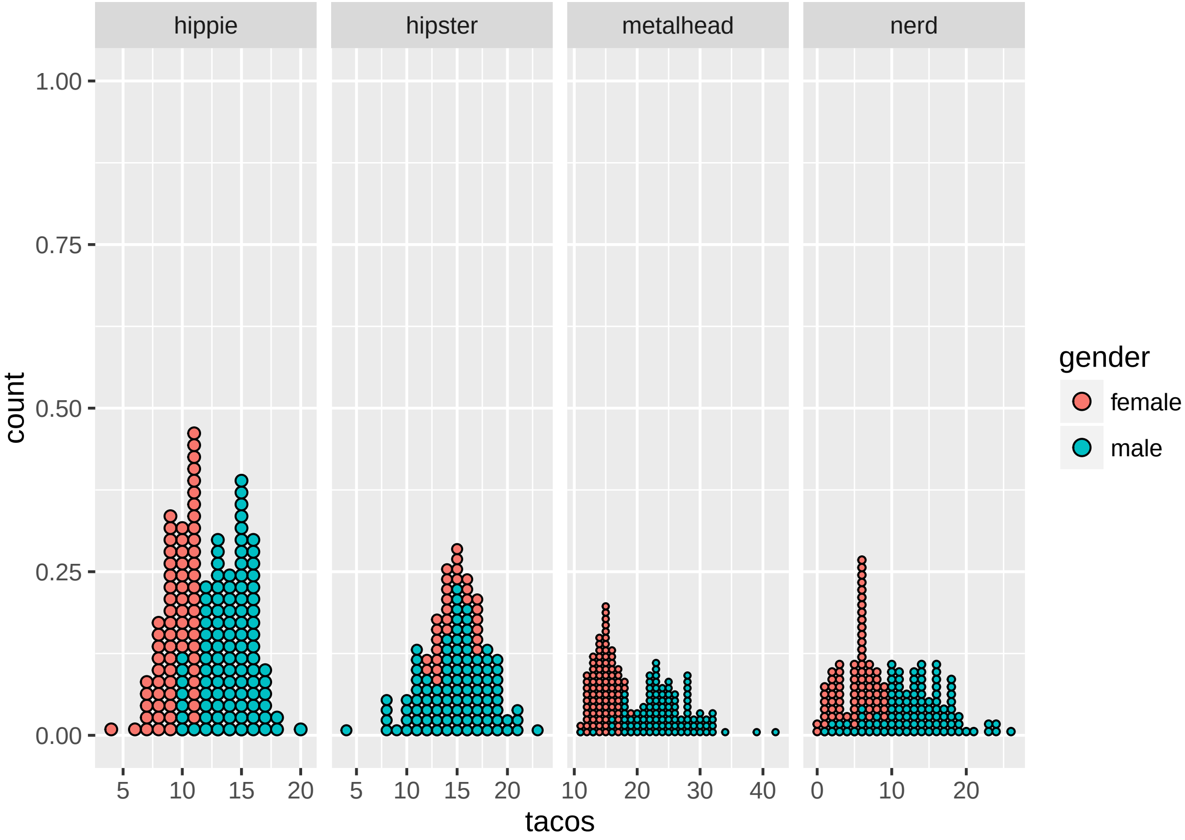


Three hypotheses: 1. The more a person showers and goes on the computer, the more likely it is that they will be a nerd. 2. As the cups of coffee increase, the number of hipsters increase. 3. As the number of tacos increase, the amount of nerds, hippies, and hipsters decrease.

Visualization with the ggplot function (Plotting tutorial)

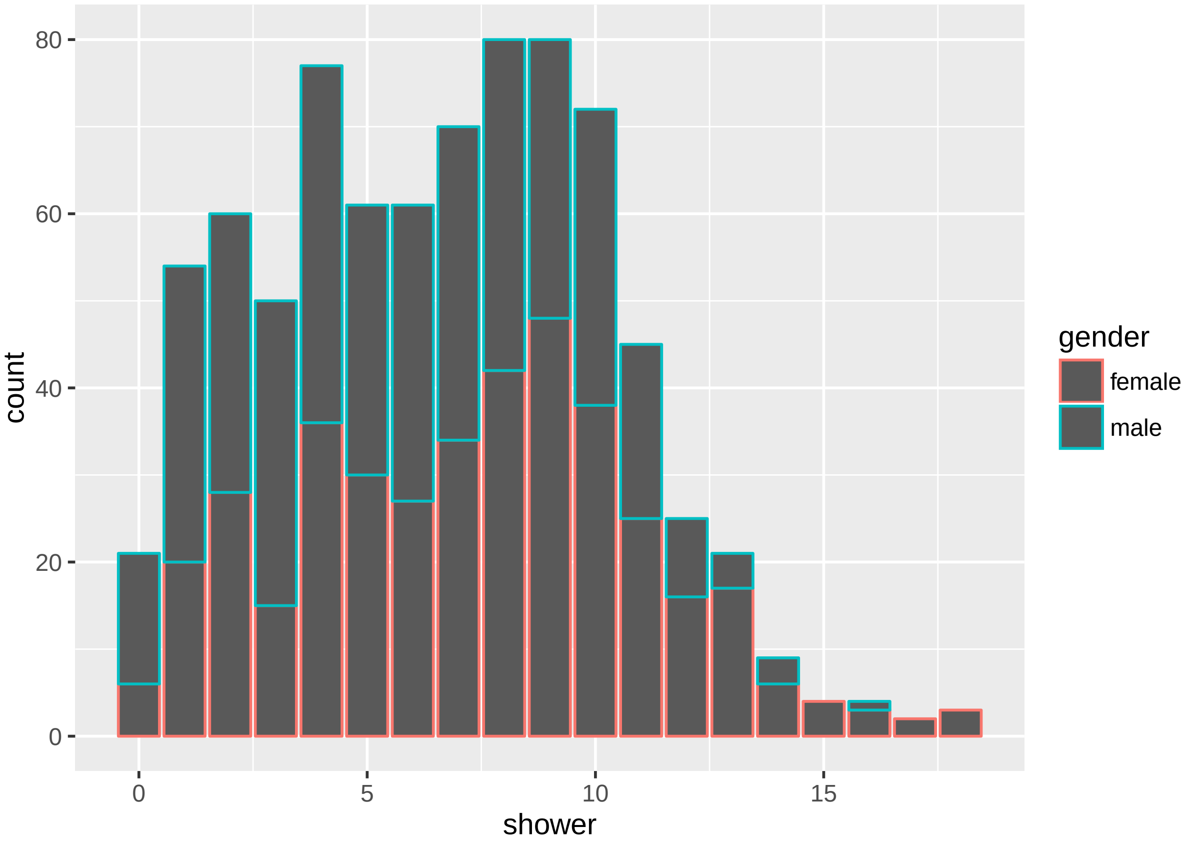
AB4-1

```
p <- ggplot(stereotype_db, aes(tacos, fill=gender))
p + geom_dotplot(binwidth=1) + facet_grid(~population, scales="free")
```

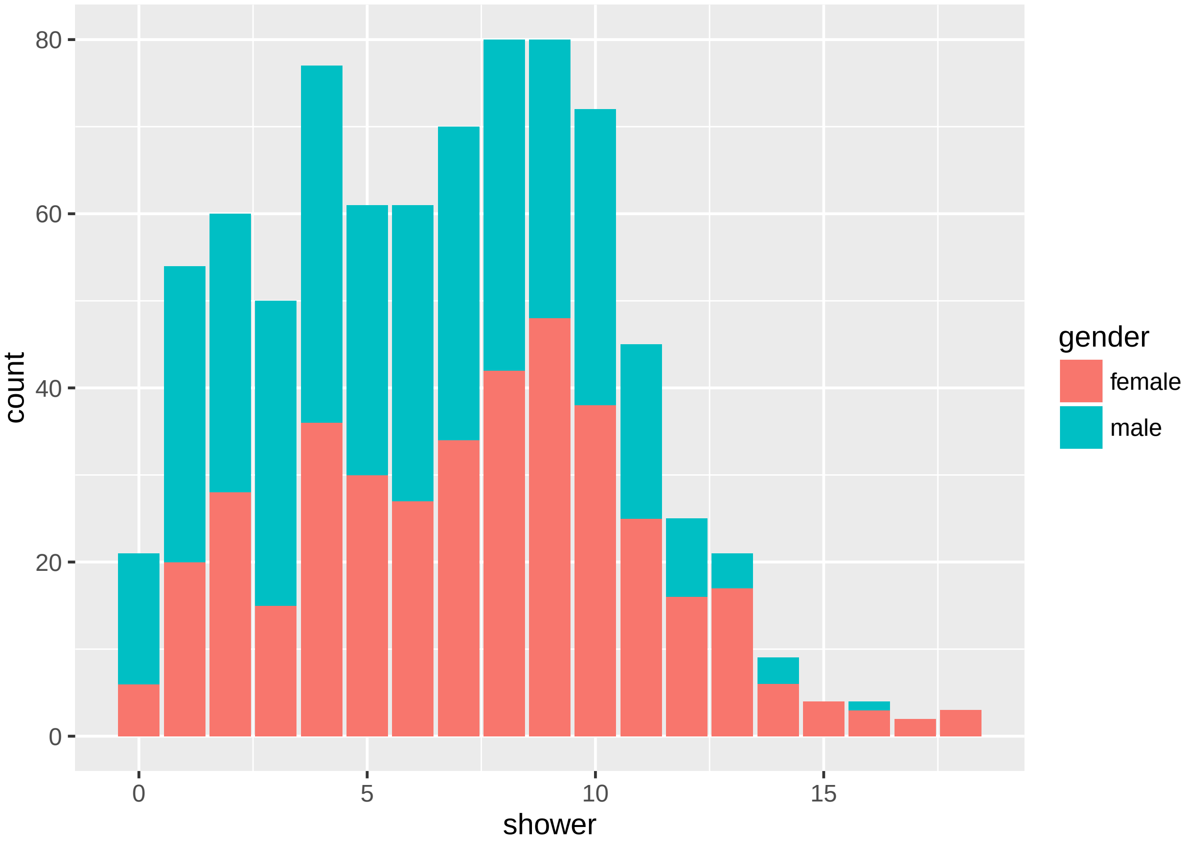


AB4-2

```
a <- ggplot(stereotype_db, aes(shower, colour=gender))
b <- ggplot(stereotype_db, aes(shower, fill=gender))
a + geom_bar()
```



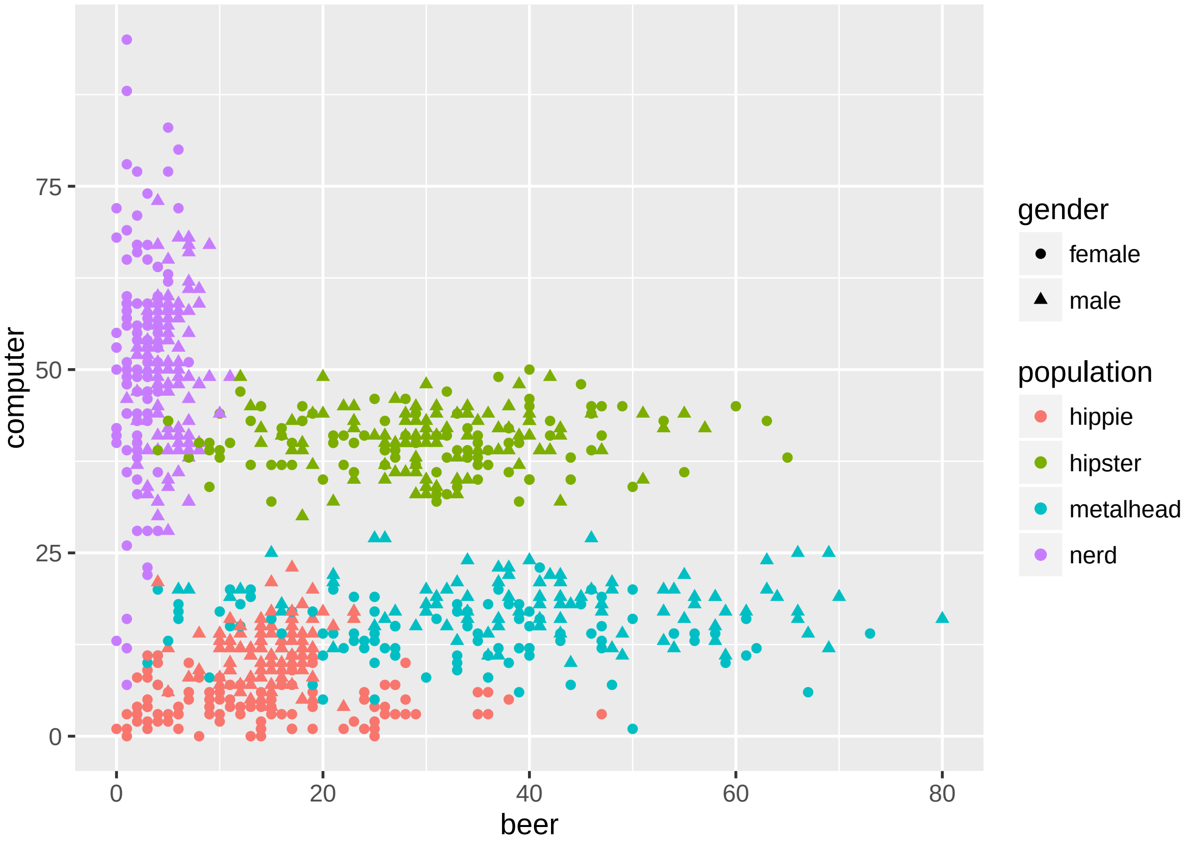
```
b + geom_bar()
```



color= means to outline the edge of the unique attributes with different colors while fill= means to fill the whole bar or symbol with color, edges and all.

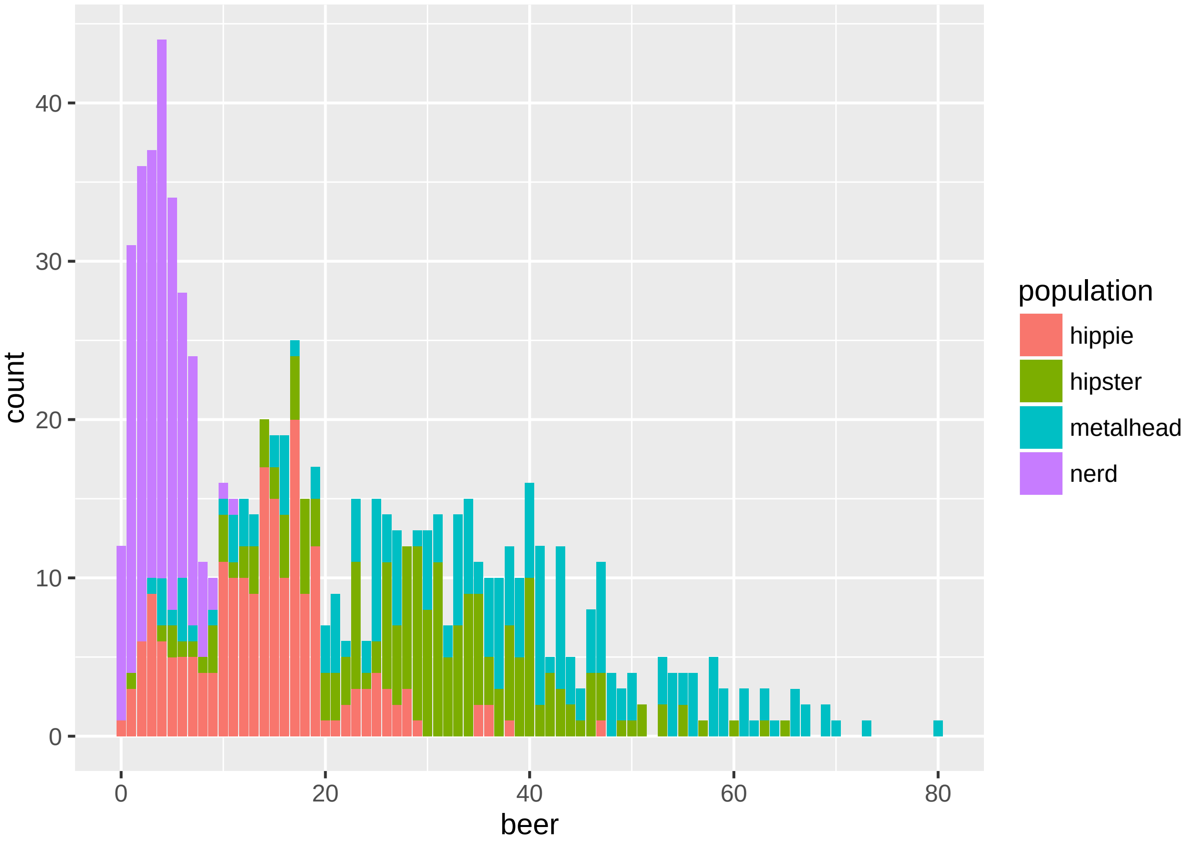
AB5-1

```
c <- ggplot(stereotype_db, aes(beer, computer, color=population, shape=gender))
c+ geom_point()
```



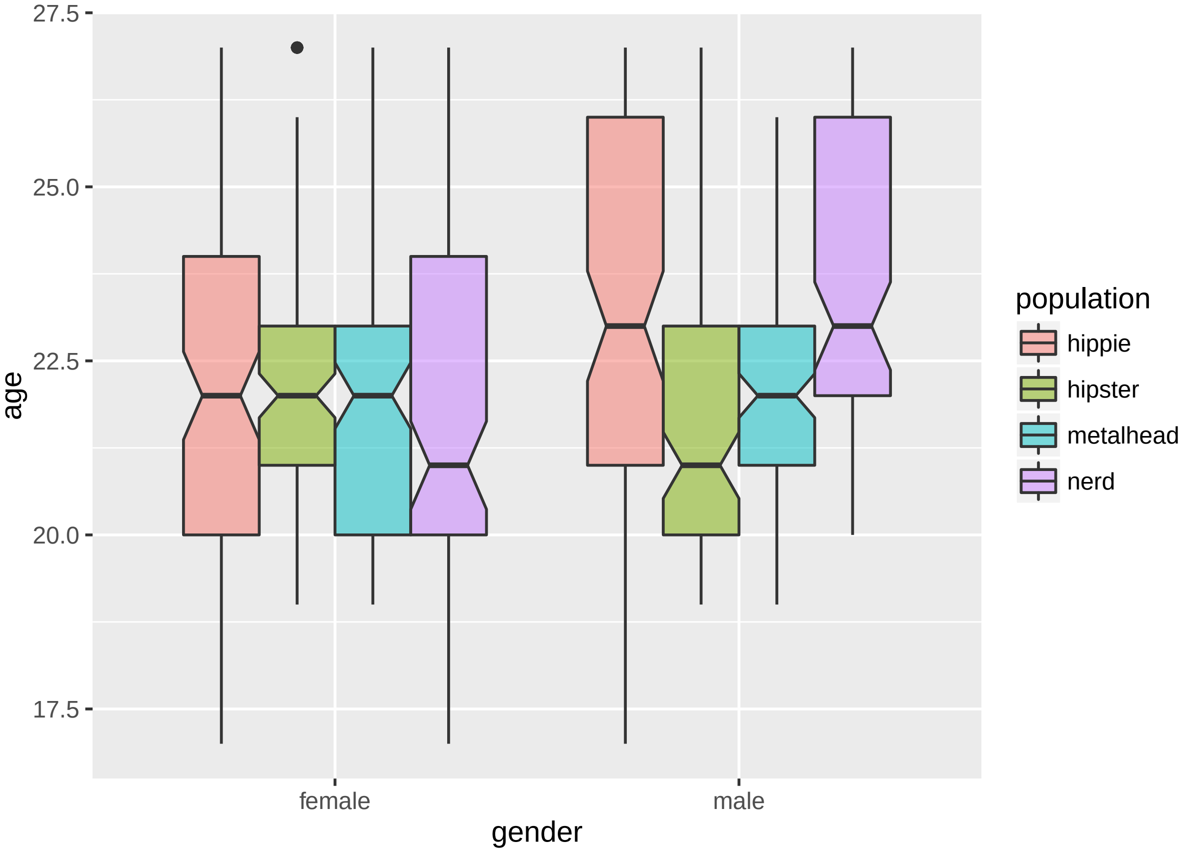
AB5-2

```
d <- ggplot(stereotype_db, aes(beer, fill=population, shape=gender), colour="clarity")
d + geom_bar() + scale_color_hue(l=90, c=50)
```



AB5-3

```
f <- ggplot(stereotype_db, aes(gender, age))
f + geom_boxplot(aes(shape=population, fill=population), notch=TRUE, alpha=0.5)
```



Visualization of the BLAST dataset

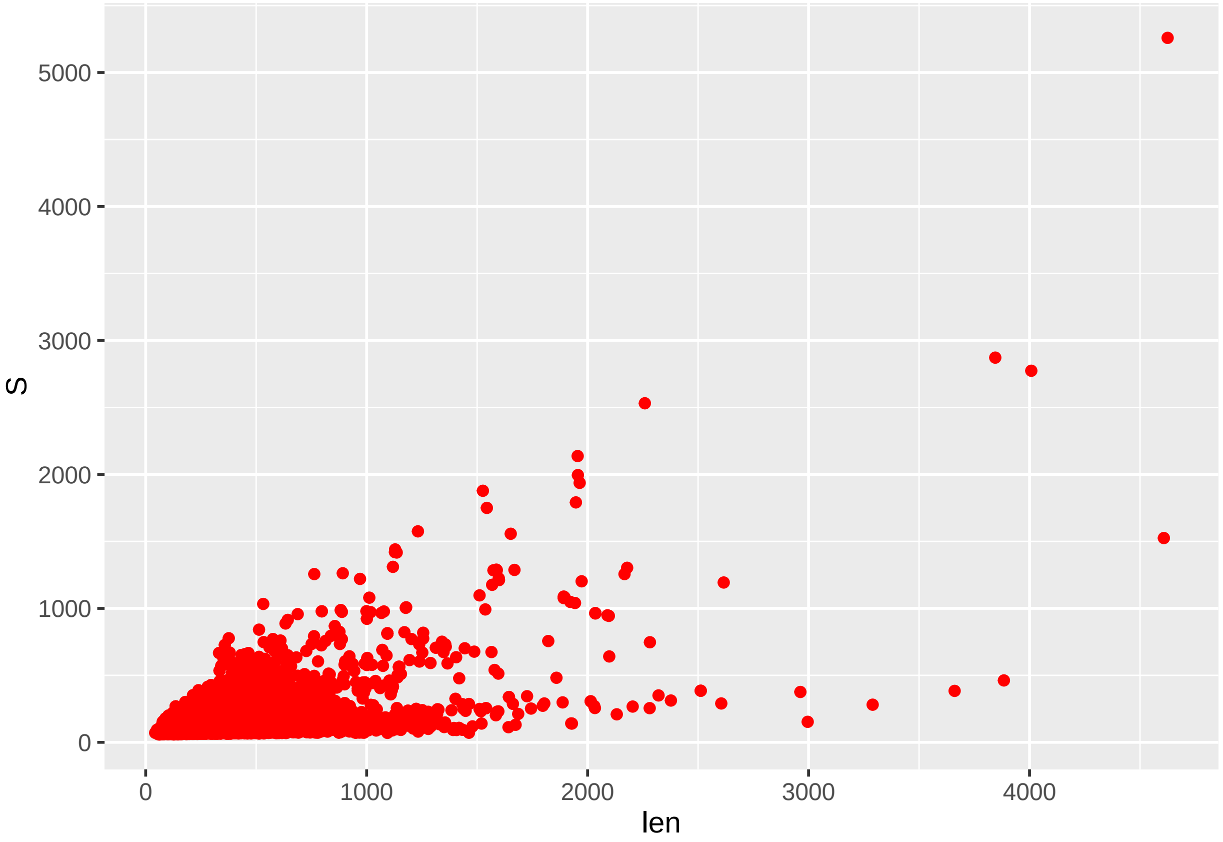
Exercise 12

```
fly.worm.sample <- sample(nrow(fly.worm), 10000)
fw.db <- fly.worm[fly.worm.sample, ]
summary(fw.db)
```

```
##        qid                 sid             pct             len
##  F15G9.4b:  82   FBpp0304921:  29   Min.   :18.41   Min.   :  43.0
##  F15G9.4d:  79   FBpp0304926:  25   1st Qu.:26.15   1st Qu.: 205.0
##  C09D1.1b:  71   FBpp0088734:  24   Median :29.55   Median : 269.0
##  F15G9.4a:  65   FBpp0304922:  24   Mean   :31.76   Mean   : 344.3
##  C09D1.1a:  64   FBpp0304925:  24   3rd Qu.:34.55   3rd Qu.: 384.0
##  C09D1.1e:  63   FBpp0306553:  24   Max.   :98.50   Max.   :4625.0
##  (Other) :9576   (Other)    :9850
##       mis             gaps              qb                qe
##  Min.   :   2   Min.   :  0.000   Min.   :    1.0   Min.   :   58.0
##  1st Qu.: 123   1st Qu.:  4.000   1st Qu.:   36.0   1st Qu.:  285.0
##  Median : 163   Median :  7.000   Median :  125.5   Median :  431.0
##  Mean   : 204   Mean   :  9.909   Mean   :  719.6   Mean   : 1043.8
##  3rd Qu.: 224   3rd Qu.: 12.000   3rd Qu.:  430.2   3rd Qu.:  773.2
##  Max.   :2685   Max.   :162.000   Max.   :18225.0   Max.   :18528.0
##
```

```
##        sb                se                E                   S
##  Min.   :    1.0   Min.   :   57.0   Min.   :0.000e+00   Min.   :  58.5
##  1st Qu.:   46.0   1st Qu.:  320.0   1st Qu.:0.000e+00   1st Qu.:  79.0
##  Median :  203.0   Median :  508.0   Median :0.000e+00   Median :  98.6
##  Mean   :  661.5   Mean   :  986.5   Mean   :2.279e-12   Mean   : 135.9
##  3rd Qu.:  531.0   3rd Qu.:  910.0   3rd Qu.:4.000e-15   3rd Qu.: 140.0
##  Max.   :22005.0   Max.   :22772.0   Max.   :1.000e-10   Max.   :5259.0
##
```

```
score.length <- ggplot(fw.db, aes(len, S))
score.length + geom_point(colour="red")
```

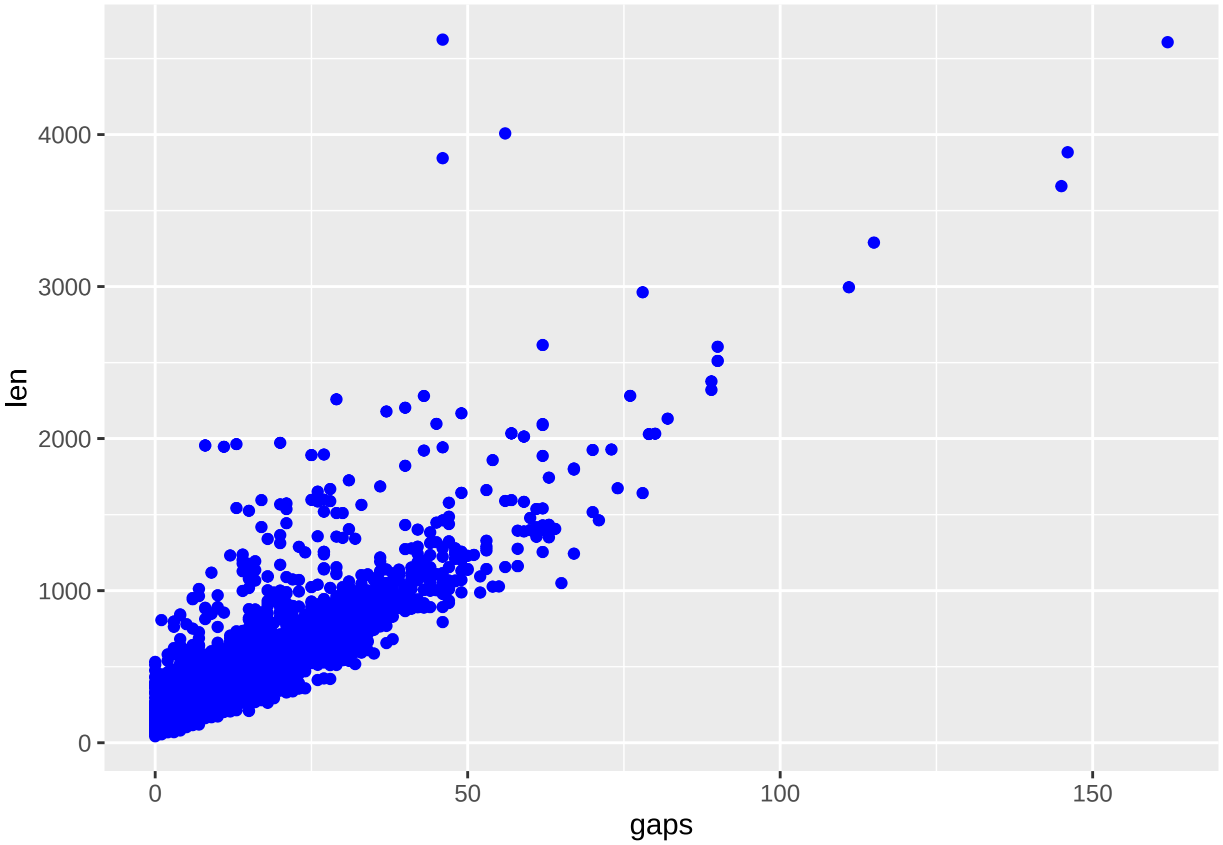


Using a random sampling of 10,000 points, we can roughly see the relationship between score and alignment length. The relationship is that as the alignment length increases, the score increases linearly.

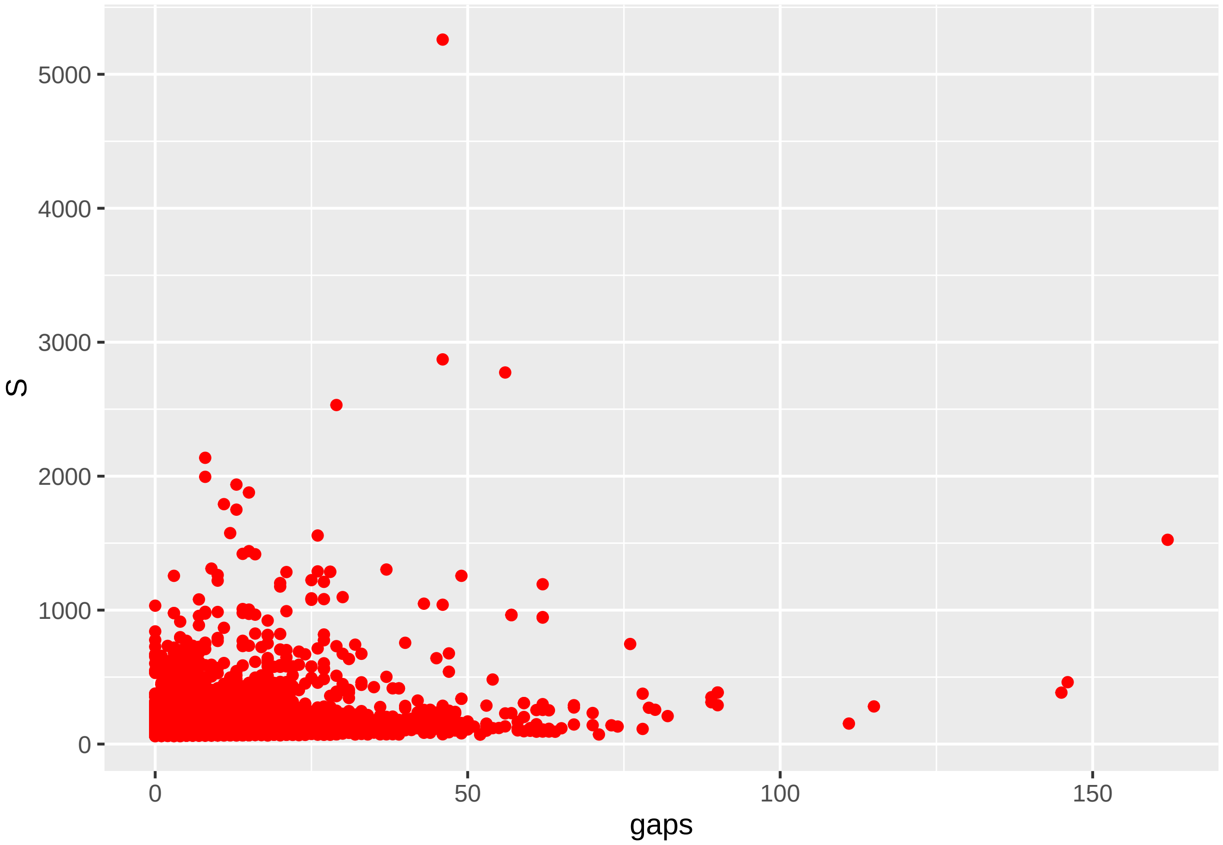
Exercise 13

Hypothesis: As the amount of gaps increase, the alignment length increases and as a result, the scores increase. So the bigger the gaps, the bigger the score.

```
length.gaps <- ggplot(fw.db, aes(x=gaps, y=len))
length.gaps + geom_point(colour = "blue")
```



```
score.gaps <- ggplot(fw.db, aes(gaps, S))
score.gaps + geom_point(colour = "red")
```



Conclusion: As the amount of gaps increase, the alignment length does increase and thus the score increases. However, the increase in gaps does not directly result in the increase of score.