Home Work 3

Hari

30 November 2016

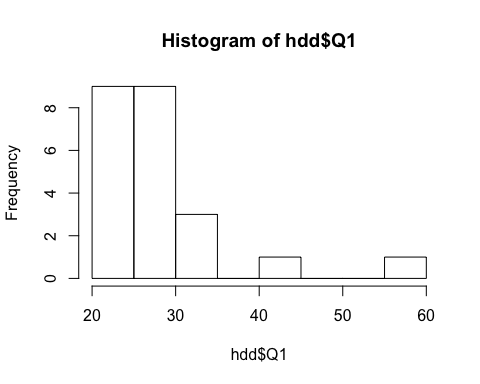
Q1.

hdd <- read.csv("hd.csv", header = TRUE)  
  
summary(hdd)

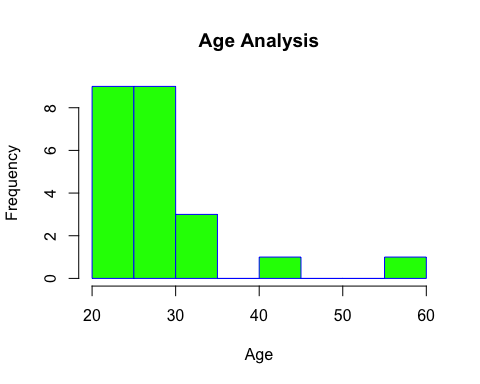
## Qname Q1 Q2 Q3   
## Abdelhamid: 1 Min. :22.00 Min. : 5.00 Min. :0.000   
## Alex : 1 1st Qu.:24.50 1st Qu.: 8.00 1st Qu.:1.000   
## Ayat : 1 Median :26.00 Median : 9.00 Median :1.000   
## Bobby : 1 Mean :28.65 Mean :10.54 Mean :1.587   
## Chris : 1 3rd Qu.:29.00 3rd Qu.:11.00 3rd Qu.:2.000   
## David : 1 Max. :58.00 Max. :41.00 Max. :7.000   
## (Other) :17   
## Q4 Q5 Q6 Q7   
## Min. : 150.0 Min. : 0 Min. : 1.00 Min. :10.00   
## 1st Qu.: 410.5 1st Qu.: 115 1st Qu.: 50.00 1st Qu.:40.00   
## Median :2694.0 Median : 230 Median : 70.00 Median :50.00   
## Mean :3963.3 Mean :1147 Mean : 60.52 Mean :55.04   
## 3rd Qu.:7050.0 3rd Qu.:2300 3rd Qu.: 80.00 3rd Qu.:77.50   
## Max. :9304.0 Max. :2400 Max. :100.00 Max. :92.00   
##   
## Q8 Q9 Q10 Q11   
## Min. : 1.00 Min. : 1.0 Min. : 10.0 Min. : 0.00   
## 1st Qu.: 55.00 1st Qu.: 22.5 1st Qu.: 90.0 1st Qu.: 16.50   
## Median : 72.00 Median : 40.0 Median : 300.0 Median : 30.00   
## Mean : 63.35 Mean : 48.0 Mean : 554.3 Mean : 51.74   
## 3rd Qu.: 85.00 3rd Qu.: 71.5 3rd Qu.: 525.0 3rd Qu.: 85.00   
## Max. :100.00 Max. :100.0 Max. :3650.0 Max. :213.00   
##   
## Q12 Q13 Q14 Q15   
## Min. : 20.0 Min. : 0.0 Min. : 0.00 Min. : 69.00   
## 1st Qu.: 250.0 1st Qu.: 142.0 1st Qu.:41.50 1st Qu.: 80.00   
## Median : 500.0 Median : 295.0 Median :60.00 Median : 90.00   
## Mean : 931.9 Mean : 344.9 Mean :55.39 Mean : 89.22   
## 3rd Qu.:1173.0 3rd Qu.: 468.0 3rd Qu.:76.50 3rd Qu.:100.00   
## Max. :3000.0 Max. :1150.0 Max. :99.00 Max. :100.00   
##   
## Q16 Q17 Q18 Q19   
## Min. : 50.00 Min. : 1.000 Min. : 2.00 Min. : 2.000   
## 1st Qu.: 64.50 1st Qu.: 3.500 1st Qu.: 9.00 1st Qu.: 3.000   
## Median : 83.00 Median : 5.000 Median :16.00 Median : 5.000   
## Mean : 80.48 Mean : 6.478 Mean :16.52 Mean : 6.913   
## 3rd Qu.:100.00 3rd Qu.:10.000 3rd Qu.:21.00 3rd Qu.: 9.000   
## Max. :100.00 Max. :15.000 Max. :49.00 Max. :22.000   
##   
## Q20 Q21 Q22 Q23   
## Min. :32.00 Min. : 1.00 Min. : 0.000 Min. : 0.00   
## 1st Qu.:69.00 1st Qu.: 30.00 1st Qu.: 1.000 1st Qu.: 0.00   
## Median :72.00 Median : 71.00 Median : 3.500 Median : 2.00   
## Mean :70.83 Mean : 58.87 Mean : 5.748 Mean : 27.39   
## 3rd Qu.:75.50 3rd Qu.: 80.00 3rd Qu.: 9.000 3rd Qu.: 10.50   
## Max. :89.00 Max. :100.00 Max. :34.000 Max. :427.00   
##   
## Q24 Q25 Q26 Q27   
## Min. : 0.000 Min. : 2.00 Min. : 0.000 Min. : 0   
## 1st Qu.: 1.000 1st Qu.:10.00 1st Qu.: 2.000 1st Qu.: 10052   
## Median : 3.000 Median :14.00 Median : 3.000 Median : 45000   
## Mean : 9.739 Mean :18.35 Mean : 4.609 Mean : 75014   
## 3rd Qu.: 7.500 3rd Qu.:25.00 3rd Qu.: 5.500 3rd Qu.:122356   
## Max. :100.000 Max. :47.00 Max. :20.000 Max. :245000   
##

Q2

hist(hdd$Q1)



hist(hdd$Q1, main="Age Analysis", xlab="Age", border="blue", col="green")



hist(hdd$Q2)



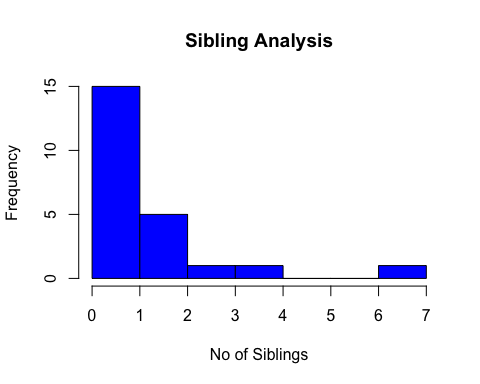
hist(hdd$Q2, main="Shoe size Analysis", xlab="Shoe Size", border="blue", col="red")



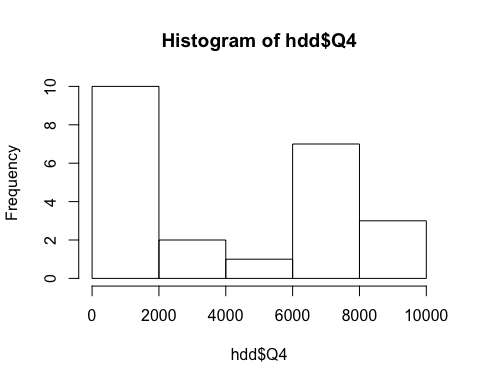
hist(hdd$Q3)



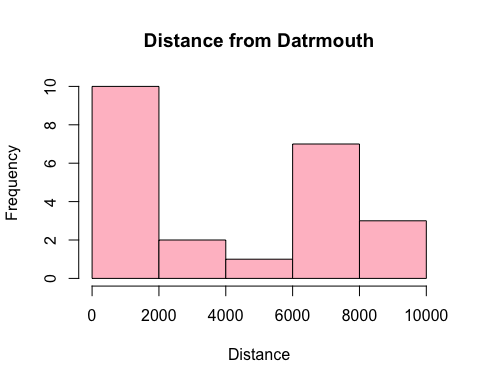
hist(hdd$Q3, main="Sibling Analysis", xlab="No of Siblings", border="black", col="blue")



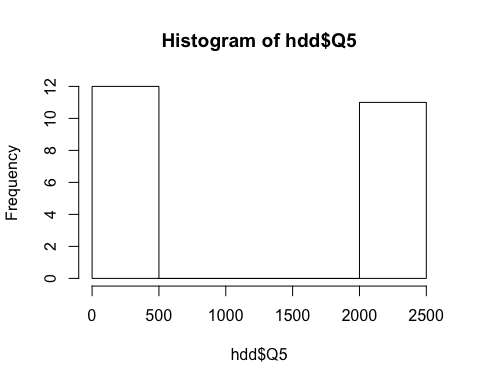
hist(hdd$Q4)



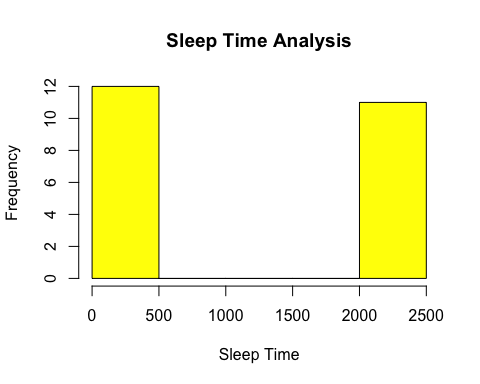
hist(hdd$Q4, main="Distance from Datrmouth", xlab="Distance", border="black", col="pink")



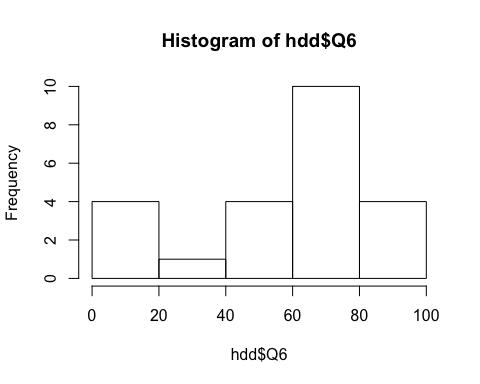
hist(hdd$Q5)



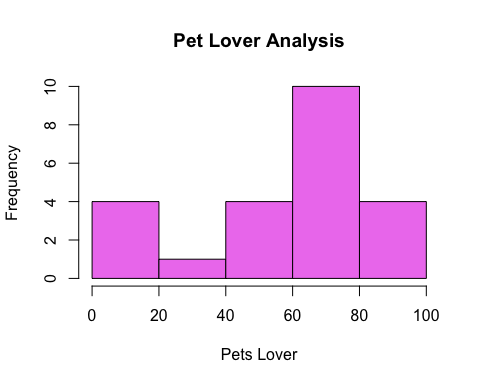
hist(hdd$Q5, main="Sleep Time Analysis", xlab="Sleep Time", border="black", col="yellow")



hist(hdd$Q6)



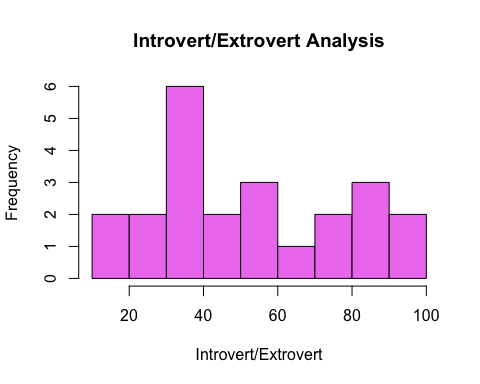
hist(hdd$Q6, main="Pet Lover Analysis", xlab="Pets Lover", border="black", col="violet")



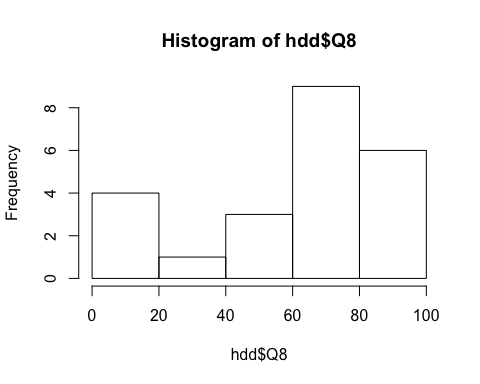
hist(hdd$Q7)



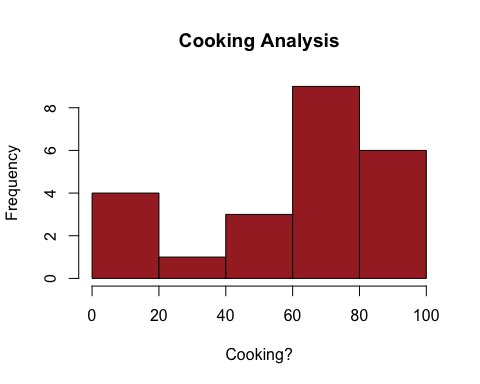
hist(hdd$Q7, main="Introvert/Extrovert Analysis", xlab="Introvert/Extrovert", border="black", col="violet")



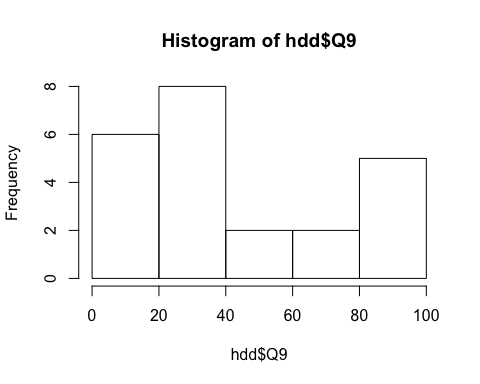
hist(hdd$Q8)



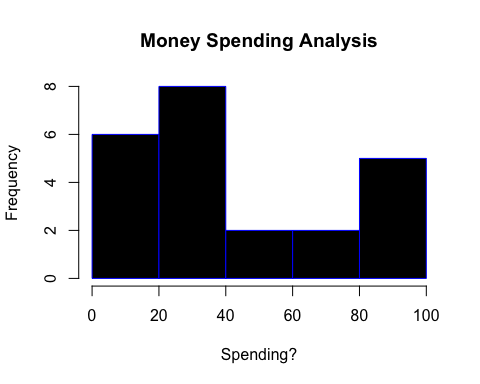
hist(hdd$Q8, main="Cooking Analysis", xlab="Cooking?", border="black", col="brown")



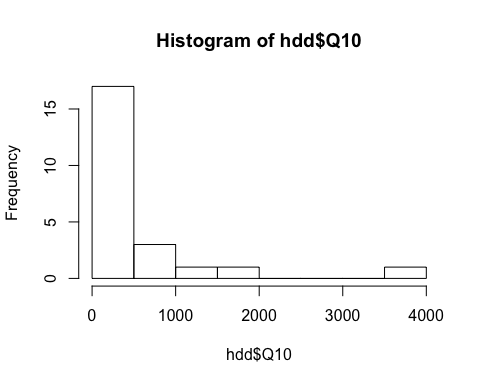
hist(hdd$Q9)



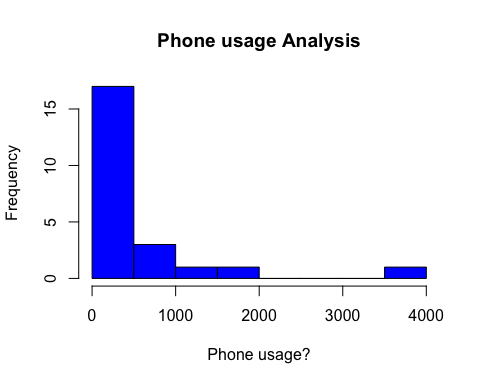
hist(hdd$Q9, main="Money Spending Analysis", xlab="Spending?", border="blue", col="black")



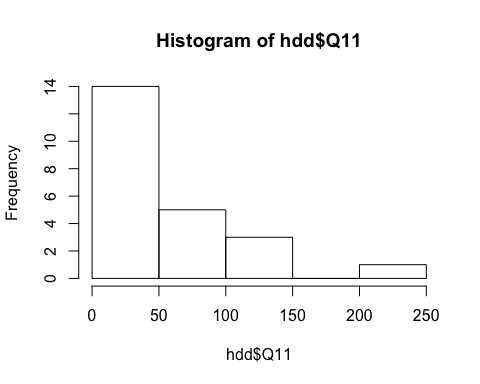
hist(hdd$Q10)



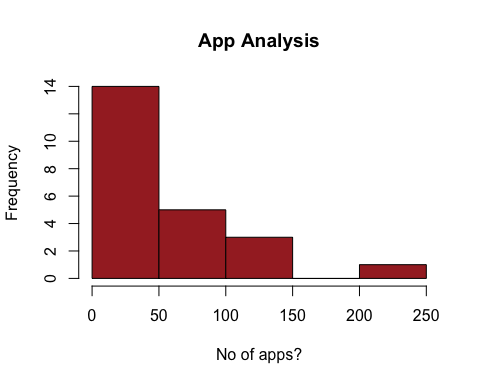
hist(hdd$Q10, main="Phone usage Analysis", xlab="Phone usage?", border="black", col="blue")



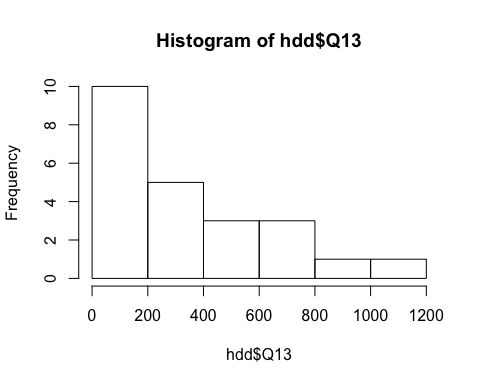
hist(hdd$Q11)



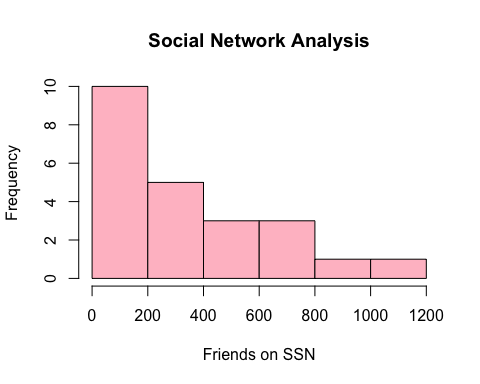
hist(hdd$Q11, main="App Analysis", xlab="No of apps?", border="black", col="brown")



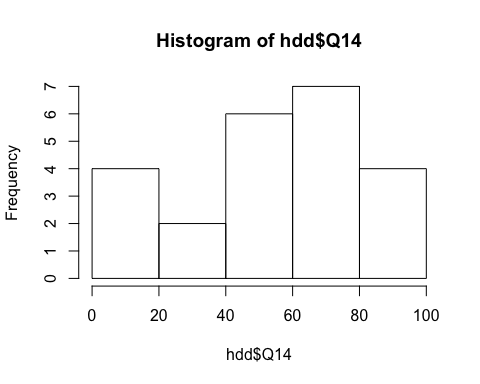
hist(hdd$Q13)



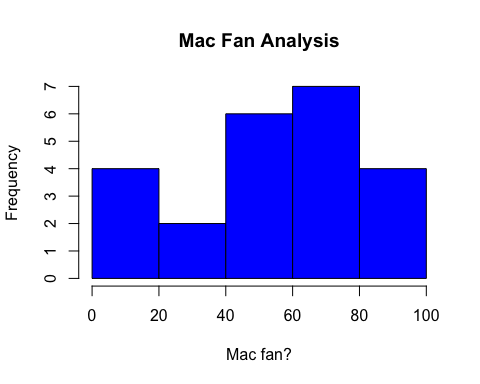
hist(hdd$Q13, main="Social Network Analysis", xlab="Friends on SSN", border="black", col="pink")



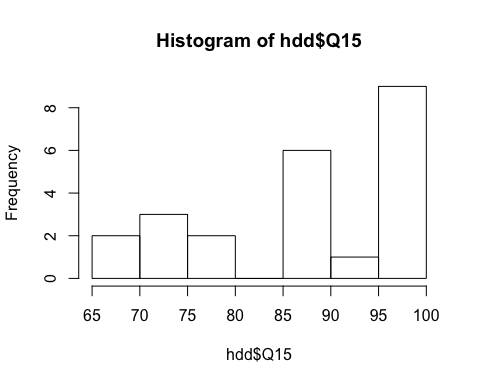
hist(hdd$Q14)



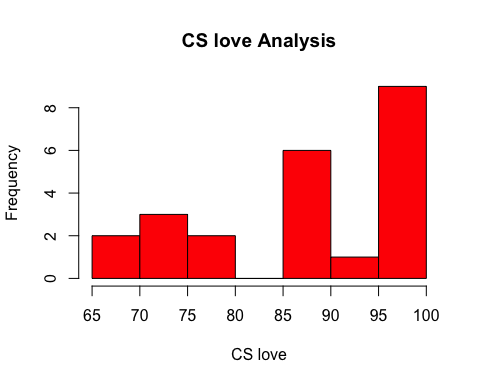
hist(hdd$Q14, main="Mac Fan Analysis", xlab="Mac fan?", border="black", col="blue")



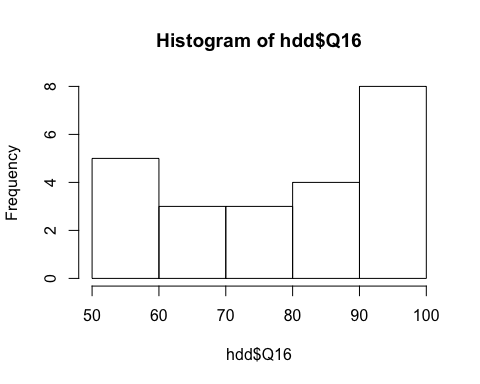
hist(hdd$Q15)



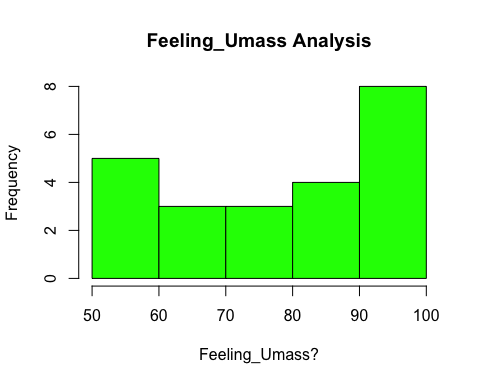
hist(hdd$Q15, main="CS love Analysis", xlab="CS love", border="black", col="red")



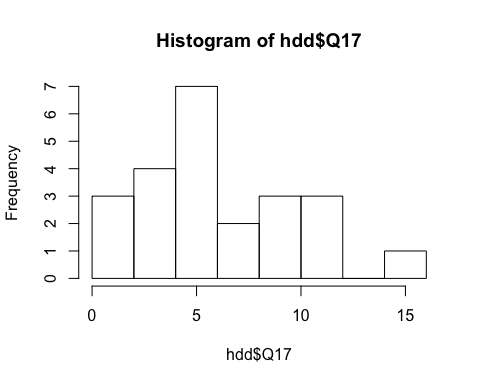
hist(hdd$Q16)



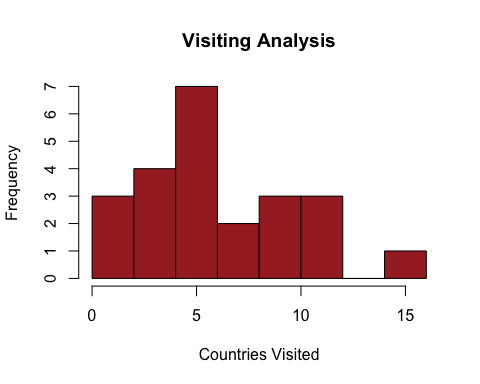
hist(hdd$Q16, main="Feeling\_Umass Analysis", xlab="Feeling\_Umass?", border="black", col="green")



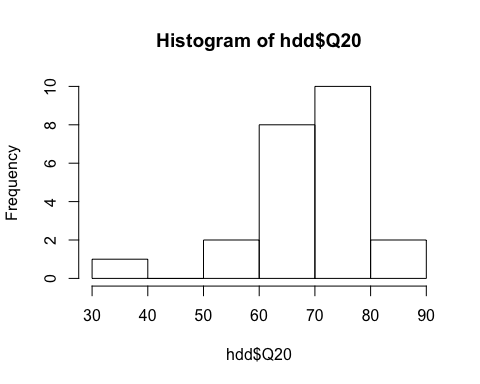
hist(hdd$Q17)



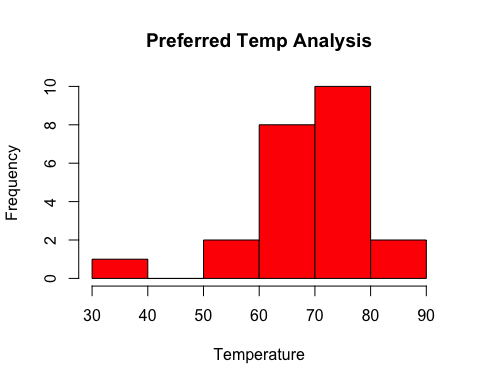
hist(hdd$Q17, main="Visiting Analysis", xlab="Countries Visited", border="black", col="brown")



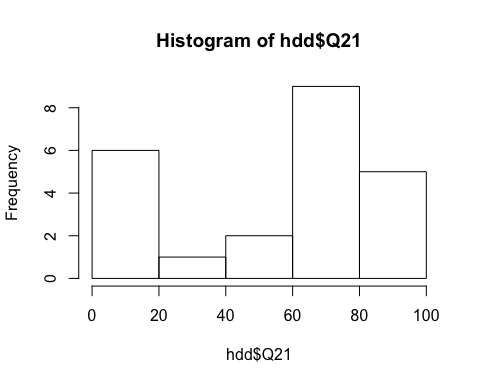
hist(hdd$Q20)



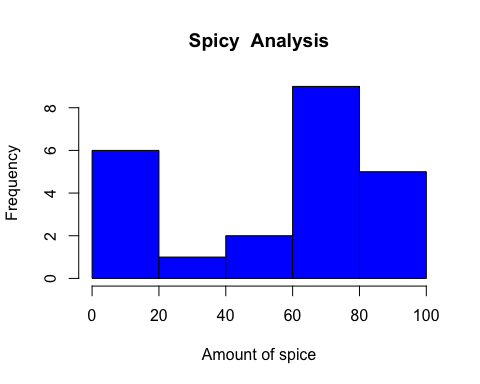
hist(hdd$Q20, main="Preferred Temp Analysis", xlab="Temperature", border="black", col="red")



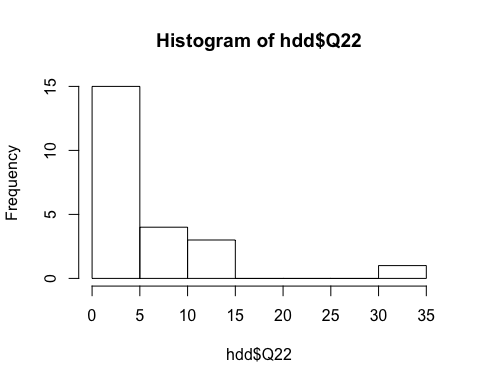
hist(hdd$Q21)



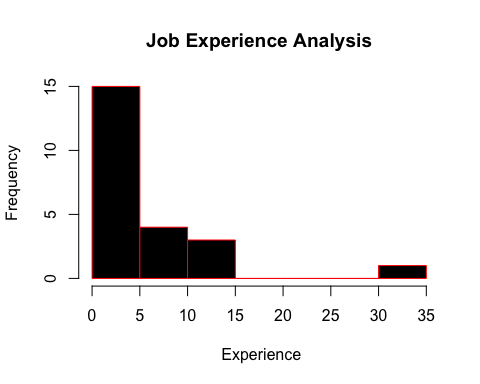
hist(hdd$Q21, main="Spicy Analysis", xlab="Amount of spice", border="black", col="blue")



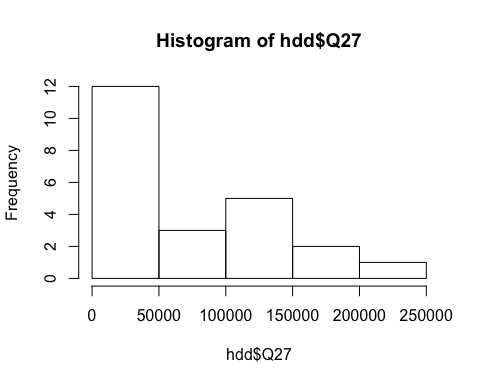
hist(hdd$Q22)



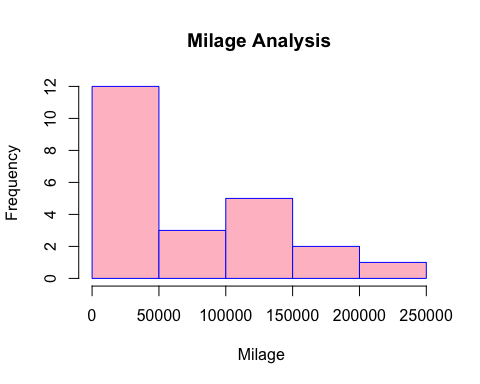
hist(hdd$Q22, main="Job Experience Analysis", xlab=" Experience", border="red", col="black")



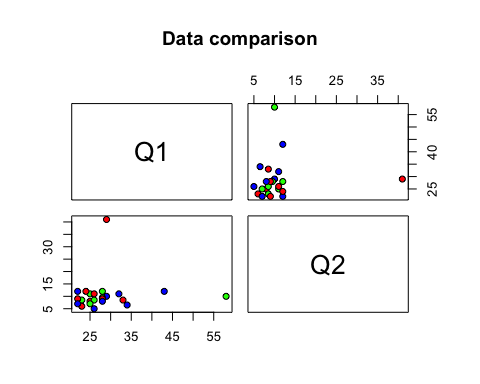
hist(hdd$Q27)



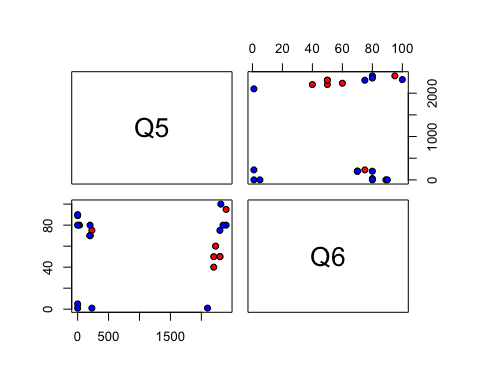
hist(hdd$Q27, main="Milage Analysis", xlab=" Milage", border="blue", col="pink")



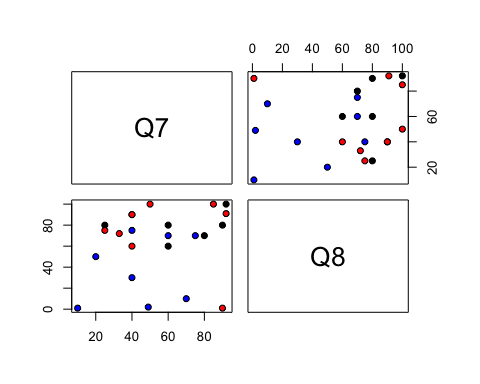
pairs(~Q1+Q2, data = hdd, main = "Data comparison", pch= 21 , bg = c("blue", "red", "green"))



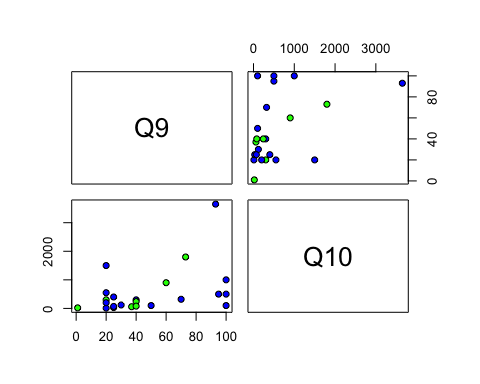
pairs(~Q5+Q6, data = hdd, pch= 21, bg = c("blue", "red", "blue"))



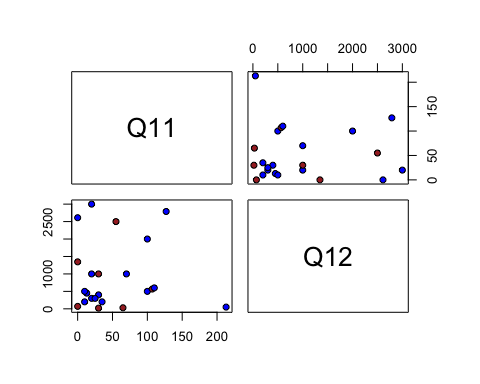
pairs(~Q7+Q8, data = hdd, pch= 21, bg = c("blue", "red", "black"))



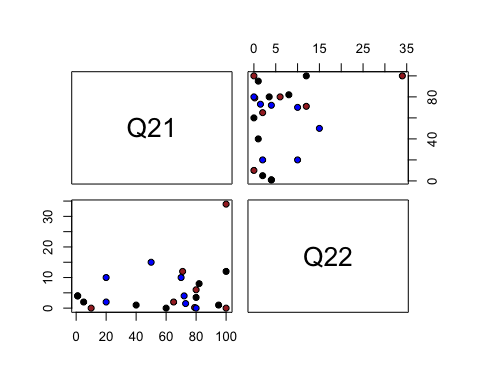
pairs(~Q9+Q10, data = hdd, pch= 21, bg = c("blue", "blue", "green"))



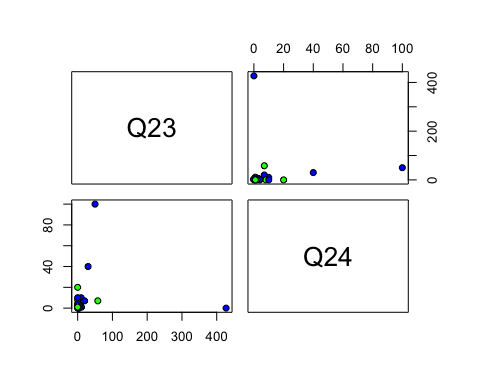
pairs(~Q11+Q12, data = hdd, pch= 21, bg = c("blue", "blue", "brown"))



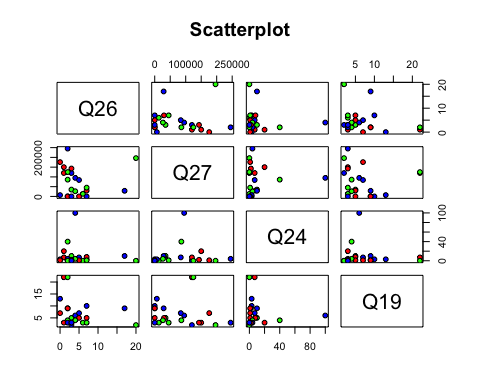
pairs(~Q21+Q22, data = hdd, pch= 21, bg = c("blue", "black", "brown"))



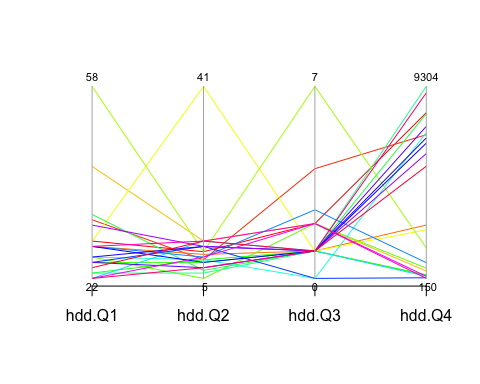
pairs(~Q23+Q24, data = hdd, pch= 21, bg = c("blue", "green", "blue"))



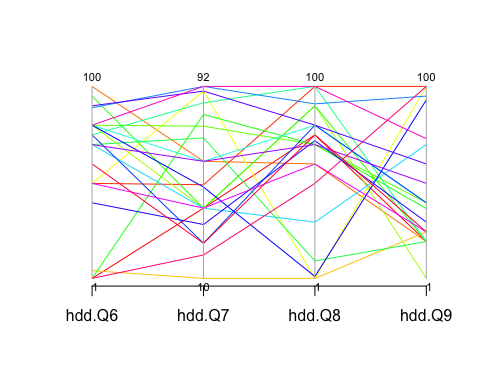
pairs(~Q26+Q27+Q24+Q19, data = hdd, main= "Scatterplot",pch= 21, bg = c("blue", "red", "green"))



library(MASS)  
pair1 <- data.frame(hdd$Q1, hdd$Q2, hdd$Q3, hdd$Q4)  
  
parcoord(pair1, var.label = TRUE, col = rainbow(length(pair1[,1]))) #c("red","green", "blue"))

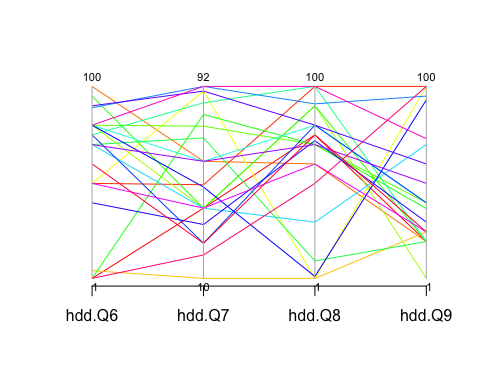


t<-data.frame(hdd$Q6, hdd$Q7, hdd$Q8, hdd$Q9)  
#colnames("Pet lover", "Introvert/Extrovert", "Cook", " Money Saver")  
#parcoord(t, col=rainbow(length(t[,1])), var.label=TRUE)  
  
parcoord(t, var.label = TRUE, col = rainbow(length(t[,1])))



#c("red","green", "blue") )

t1<-data.frame(hdd$Q20, hdd$Q22, hdd$Q23, hdd$Q27)  
parcoord(t, var.label = TRUE, col = rainbow(length(t1[,1])))



row.names(hdd) <- hdd[,1]  
hdd1 <- hdd[,-1]  
normalize <- scale(hdd1, scale=TRUE)  
head(normalize)

## Q1 Q2 Q3 Q4 Q5 Q6  
## Alex 0.04349035 -0.07824244 0.2846389 1.1776196 0.8548309 -1.9049899  
## Ayat 0.54362936 -0.29419159 1.6628906 0.8750532 0.9445462 -0.3367477  
## Bobby -0.08154440 -0.15022549 -0.4044869 -0.3657570 1.0477188 1.2634995  
## Chris 1.79397690 0.20968975 0.2846389 -0.9979766 -1.0291906 -1.7769701  
## Elias 0.04349035 4.38470661 -0.4044869 -0.4312554 1.0342615 -0.3367477  
## Harold 3.66949821 -0.07824244 3.7302681 -0.6810023 -1.0022761 0.6234006  
## Q7 Q8 Q9 Q10 Q11 Q12  
## Alex -0.5935306 0.3586387 -0.8863245 -0.6603888 -0.5991015 0.07058334  
## Ayat -0.1989871 1.1281059 1.6460312 0.5406538 0.9109626 -0.44741819  
## Bobby 0.1955563 -0.1030417 -0.8863245 -0.3085682 -0.4103435 -0.94469965  
## Chris -1.7771609 -1.9189845 -0.7280523 -0.6361253 0.3446885 0.07058334  
## Elias 1.3791866 -1.9189845 1.6460312 -0.5512031 -0.7878595 -0.75821910  
## Harold -0.5935306 0.8203190 -1.4877590 -0.6482571 0.2503095 -0.93433962  
## Q13 Q14 Q15 Q16 Q17 Q18  
## Alex 1.1948527 1.1096729 0.97477565 -1.6542721 -0.3784090 0.03974396  
## Ayat -0.6558742 -1.7439709 0.97477565 1.0595837 -1.4023392 -1.20677100  
## Bobby -0.2016049 0.4684046 0.97477565 1.0595837 0.1335561 0.70455193  
## Chris -1.1269683 1.3982436 0.07074985 0.5168125 0.1335561 2.61587487  
## Elias -0.4876263 -1.7439709 0.97477565 1.0595837 0.9015038 -0.54196303  
## Harold -1.1572530 -0.1728636 0.97477565 1.0595837 0.9015038 2.69897587  
## Q19 Q20 Q21 Q22 Q23 Q24  
## Alex 0.5536930 0.10685384 -1.724062 -0.2286354 -0.25283162 -0.31232993  
## Ayat 0.3743277 -0.98542982 1.225367 0.8178551 -0.28670615 -0.35867566  
## Bobby -0.3431337 -3.53409167 1.225367 -0.7518807 -0.19637408 0.01209019  
## Chris -0.1637683 0.37992475 -0.264244 1.2102891 0.25528630 4.18320598  
## Elias -0.7018643 -0.07519344 -1.724062 -0.2286354 -0.30928917 0.47554750  
## Harold -0.5224990 0.83504294 1.225367 3.6957042 0.02945611 1.40246212  
## Q25 Q26 Q27  
## Alex -0.2653021 0.4902266 -1.0297044  
## Ayat -1.0577629 -0.5347926 -1.0297044  
## Bobby 0.9233891 -0.3297888 -0.5492659  
## Chris 0.9233891 -0.1247850 0.2743429  
## Elias -1.0577629 -0.7397965 1.0293176  
## Harold -1.2955012 -0.5347926 0.1508016

Q3

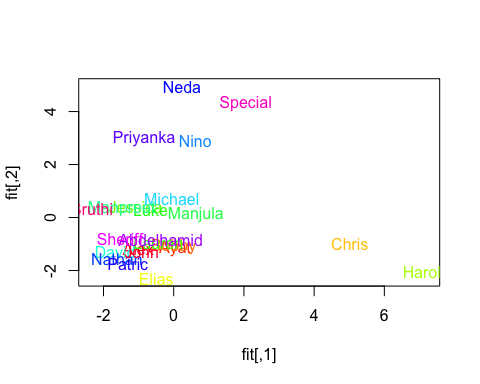
aa <- dist(normalize, method = "euclidean", diag = FALSE, upper = FALSE)  
print(aa)

## Alex Ayat Bobby Chris Elias Harold  
## Ayat 7.081259   
## Bobby 7.132254 6.672798   
## Chris 8.099717 9.591771 8.557536   
## Elias 8.404103 8.161979 8.336198 9.528454   
## Harold 9.807577 8.585894 9.513487 7.591897 10.856015   
## Jessica 5.423777 6.761043 5.591036 8.347032 8.089530 10.009185  
## Lauren 5.458503 6.100273 5.175505 7.929764 7.983310 8.599683  
## Luke 6.500680 6.982805 6.225259 8.027256 8.557779 10.061631  
## Manjula 5.783260 6.555485 6.071308 7.557627 8.433576 8.759948  
## Manpreet 5.998123 5.709198 5.921135 9.668146 8.700588 9.779337  
## David 4.229962 5.969088 5.796540 8.545341 7.053541 9.698909  
## Michael 7.522507 7.663266 7.234314 8.295266 8.158642 9.824892  
## Nino 7.324890 6.696555 7.477909 8.939331 8.825268 9.384535  
## Nathan 6.093882 6.753545 6.601508 8.845222 7.724745 9.925909  
## Neda 8.406100 8.967689 9.038122 9.211299 10.186744 11.452956  
## Patric 3.604155 5.973367 6.333908 8.128201 7.170609 9.580970  
## Priyanka 8.077307 9.190068 8.795337 10.292246 9.654203 10.782895  
## Abdelhamid 5.999680 6.471333 6.435202 8.298486 6.750905 9.017079  
## Sheriff 4.992051 7.426456 6.480438 8.576085 8.208652 10.050842  
## Special 8.887918 8.916798 8.627911 9.669321 9.932237 10.217758  
## Sruthi 6.814386 5.995489 7.967705 9.290619 8.598325 11.250092  
## John 5.684607 6.418547 6.238563 8.349653 6.631934 9.287143  
## Jessica Lauren Luke Manjula Manpreet David  
## Ayat   
## Bobby   
## Chris   
## Elias   
## Harold   
## Jessica   
## Lauren 4.329426   
## Luke 5.826314 5.745993   
## Manjula 6.197691 5.504934 6.063826   
## Manpreet 5.750066 4.681386 6.449141 5.367192   
## David 4.300300 3.584483 6.293567 5.279875 4.388672   
## Michael 7.435158 6.273908 5.240318 6.424638 7.663434 6.778874  
## Nino 5.850056 6.334607 6.104530 6.917877 6.879508 6.775898  
## Nathan 5.782968 3.747085 5.320529 6.672849 6.467621 4.794061  
## Neda 7.744923 8.170769 8.133352 7.635958 8.002346 8.056700  
## Patric 5.231969 4.372397 5.665883 5.696587 5.704007 3.069190  
## Priyanka 7.388739 7.144261 8.166806 6.799575 6.644834 7.419318  
## Abdelhamid 5.188847 5.440832 6.815207 6.004772 5.234062 4.684327  
## Sheriff 5.187242 4.362625 5.712199 6.753121 6.257178 5.170504  
## Special 7.438358 7.925571 8.433107 8.638172 8.217954 8.512909  
## Sruthi 6.391592 6.756158 6.112619 7.222264 6.140584 6.178490  
## John 5.816082 4.522847 6.270725 6.516247 6.052672 4.394771  
## Michael Nino Nathan Neda Patric Priyanka  
## Ayat   
## Bobby   
## Chris   
## Elias   
## Harold   
## Jessica   
## Lauren   
## Luke   
## Manjula   
## Manpreet   
## David   
## Michael   
## Nino 6.042761   
## Nathan 5.934213 6.828701   
## Neda 7.457429 7.096589 8.870789   
## Patric 6.328146 6.786514 4.172949 8.018905   
## Priyanka 7.572228 7.395270 8.295173 8.406801 8.650352   
## Abdelhamid 6.898763 6.690272 6.377795 8.800831 5.706180 6.770119  
## Sheriff 5.238897 6.386962 4.541168 8.626762 4.971908 7.374285  
## Special 9.150708 6.120952 8.869344 7.936528 8.515540 9.105634  
## Sruthi 7.643622 7.418543 6.501392 7.976100 6.131437 8.092368  
## John 5.797240 6.790495 4.328141 8.065627 3.454866 7.784032  
## Abdelhamid Sheriff Special Sruthi  
## Ayat   
## Bobby   
## Chris   
## Elias   
## Harold   
## Jessica   
## Lauren   
## Luke   
## Manjula   
## Manpreet   
## David   
## Michael   
## Nino   
## Nathan   
## Neda   
## Patric   
## Priyanka   
## Abdelhamid   
## Sheriff 5.549136   
## Special 8.178914 8.968400   
## Sruthi 6.615520 7.145691 9.592108   
## John 4.994213 4.943915 8.171835 6.593794

#as.dist(normalize, diag = FALSE, upper = FALSE)

Q4.

fit <- cmdscale(aa, k=2)  
#print(fit)  
  
plot(fit,type="n")  
  
text(fit[,1], fit[,2], labels(aa), col = rainbow(length(fit[,1])))



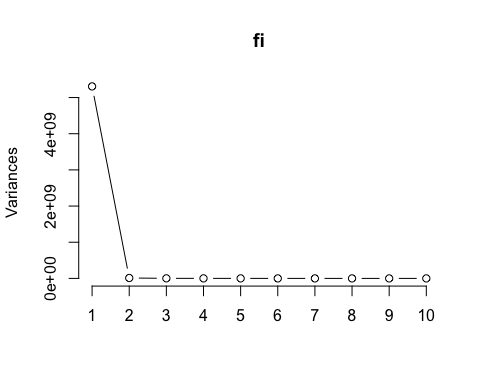
fi <- prcomp(hdd1)  
summary(fi) # print variance accounted for

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5  
## Standard deviation 7.286e+04 3.354e+03 1259.5770 841.08615 620.43074  
## Proportion of Variance 9.974e-01 2.110e-03 0.0003 0.00013 0.00007  
## Cumulative Proportion 9.974e-01 9.995e-01 0.9998 0.99991 0.99999  
## PC6 PC7 PC8 PC9 PC10 PC11 PC12 PC13  
## Standard deviation 242.05782 79.37 39.98 39.07 32.73 25.2 23.4 16.04  
## Proportion of Variance 0.00001 0.00 0.00 0.00 0.00 0.0 0.0 0.00  
## Cumulative Proportion 1.00000 1.00 1.00 1.00 1.00 1.0 1.0 1.00  
## PC14 PC15 PC16 PC17 PC18 PC19 PC20 PC21  
## Standard deviation 15.47 9.927 7.871 6.973 5.125 4.789 3.445 1.802  
## Proportion of Variance 0.00 0.000 0.000 0.000 0.000 0.000 0.000 0.000  
## Cumulative Proportion 1.00 1.000 1.000 1.000 1.000 1.000 1.000 1.000  
## PC22 PC23  
## Standard deviation 1.606 3.271e-13  
## Proportion of Variance 0.000 0.000e+00  
## Cumulative Proportion 1.000 1.000e+00

loadings(fi) # pc loadings

## NULL

plot(fi,type="lines") # scree plot



fi$scores # the principal components

## NULL

biplot(fi)

## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped  
  
## Warning in arrows(0, 0, y[, 1L] \* 0.8, y[, 2L] \* 0.8, col = col[2L], length  
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped

