Introduction to Principal Component Analysis and other multivariate statistics

Dan Chitwood

April 17, 2016

Load in ggplot2 for making graphs

library(ggplot2)

If needed, install.packages("ggrepel") Load in ggrepel, for making labels on your ggplots

library(ggrepel)

If needed, install.packages("ggdendro") Load in ggdendro, for making nice dendrograms out of your hierarchical clustering results

library(ggdendro)

If needed, install.packages("ape") Load in ape, for making nice dendrograms out of your hierarchical clustering results

library(ape)

# Principal Component Analysis (PCA)

Let's read in the dataset for the nutrition labels for our 75 candy types to learn how to do Principal Component Analysis (PCA) in R

data <- read.table("./candy\_nutrition.txt", header=TRUE)

Check the names of our data

names(data)

## [1] "id" "name" "company"   
## [4] "class" "serving\_size\_g" "calories"   
## [7] "calories\_fat" "total\_fat\_g" "saturated\_fat\_g"   
## [10] "cholesterol\_mg" "sodium\_mg" "total\_carb\_g"   
## [13] "dietary\_fiber\_g" "sugars\_g" "protein\_g"   
## [16] "primary\_ingredient"

The dataset includes names of the candies, the company that made them, their general class, calories, serving size, and a number of nutritional values, in addition to primary ingredient

Let's first normalize the nutritional values by dividing them by serving\_size\_g. We'll create new variables

data$total\_fat\_per\_serv <- data$total\_fat\_g/data$serving\_size\_g  
data$saturated\_fat\_per\_serv <- data$saturated\_fat\_g/data$serving\_size\_g  
data$cholesterol\_per\_serv <- data$cholesterol\_mg/data$serving\_size\_g  
data$sodium\_per\_serv <- data$sodium\_mg/data$serving\_size\_g  
data$total\_carb\_per\_serv <- data$total\_carb\_g/data$serving\_size\_g  
data$dietary\_fiber\_per\_serv <- data$dietary\_fiber\_g/data$serving\_size\_g  
data$sugars\_per\_serv <- data$sugars\_g/data$serving\_size\_g  
data$protein\_per\_serv <- data$protein\_g/data$serving\_size\_g

Now, let's perform a PCA using the prcomp() function. PCA is a dimension-reduction technique that reorients the axes of your data so that they explain the maximum amount of variance in the fewest number of dimensions. One way to think about PCAs is that if you measure your data using variables x, y, z in 3D, then imagine reorienting the angle of a camera to take a 2D photo that maximizes the viewable variance in the photo (that is, 2D). PCA is a lot like this, except often it is performed on much higher dimensional data than just 3 dimensions. To get a feeling for how PCAs work, try rotaing the 3D dataset in this [http://setosa.io/ev/principal-component-analysis/](example) to see how the data remains fundamentally the same, but is now described by axes that explain greater amounts of variance.

Let's now perform a PCA using the prcomp function. Check out ?prcomp. prcomp requires input data, but also the variables center and scale. to be specified. Let's set center to TRUE and scale. to TRUE as well. This is because the nutritional information is collected in different units

N.B.: You could manually scale your data if you wanted to. For example, you could use the scale() function, which sets the mean of a data column to 0 and the variance to 1. You would create an object, scaled\_data <- scale(data[17:24]) and could use that as the input into the PCA as well.

Make sure that the column numbers correspond to the new normalized traits we created! This should be columns 17-24 ?prcomp

pca <- prcomp(data[17:24], center=TRUE, scale.=TRUE)

Now that we've done our PCA, let's look at some of the outputs, which you can look up with ?prcomp

But first, let's do a summary to see the percent variance explained by each PC

summary(pca)

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6  
## Standard deviation 2.2679 1.0742 0.77439 0.72721 0.49683 0.48787  
## Proportion of Variance 0.6429 0.1442 0.07496 0.06611 0.03086 0.02975  
## Cumulative Proportion 0.6429 0.7872 0.86215 0.92826 0.95911 0.98886  
## PC7 PC8  
## Standard deviation 0.25479 0.15547  
## Proportion of Variance 0.00811 0.00302  
## Cumulative Proportion 0.99698 1.00000

Nice! The first four PCs explain >90% of variance in our data. PCs 1 and 2 explain >75%!

What do these PCs mean? What combination of our variables consitute each of the PCs? To figure this out, we look at the loadings, which you can find using the rotation output

pca$rotation

## PC1 PC2 PC3 PC4  
## total\_fat\_per\_serv -0.4236773 0.13269577 -0.1568617 0.05851302  
## saturated\_fat\_per\_serv -0.3922951 0.23421654 -0.2184548 -0.10138886  
## cholesterol\_per\_serv -0.2925627 0.50479458 -0.0466282 -0.61129836  
## sodium\_per\_serv -0.2281962 -0.58836586 -0.7210734 -0.09112375  
## total\_carb\_per\_serv 0.4232201 0.09612676 -0.1627624 0.03660121  
## dietary\_fiber\_per\_serv -0.3677174 0.15269038 0.0145450 0.52409455  
## sugars\_per\_serv 0.2688428 0.53504086 -0.5411732 0.41511592  
## protein\_per\_serv -0.3772636 -0.08785818 0.2932223 0.39493205  
## PC5 PC6 PC7 PC8  
## total\_fat\_per\_serv -0.1942906 0.2400786 0.09644378 0.818640342  
## saturated\_fat\_per\_serv -0.4346917 0.4720047 -0.29859715 -0.482015890  
## cholesterol\_per\_serv 0.3838829 -0.3527579 -0.10883972 0.008904835  
## sodium\_per\_serv 0.1990342 -0.1631544 -0.06810743 -0.051275071  
## total\_carb\_per\_serv -0.1003685 -0.1016358 -0.82985714 0.273398215  
## dietary\_fiber\_per\_serv -0.2888437 -0.6830207 -0.06725374 -0.110053733  
## sugars\_per\_serv 0.2998563 0.1441395 0.24303059 -0.080692914  
## protein\_per\_serv 0.6347403 0.2662573 -0.36442260 -0.037556471

The contribution of a variable to a PC is proportional to the absolute value of the loading, and positively or negatively contributes towards the PC based on its sign. For example, total\_fat\_per\_serv is negatively associated with PC1 and total\_carb\_per\_serv is postively associated

Now, let's get the PC scores

scores <- as.data.frame(pca$x)  
head(scores)

## PC1 PC2 PC3 PC4 PC5 PC6  
## 1 -1.360340 1.2253820 -0.2491928 -0.2812836 0.2172947 0.34206721  
## 2 0.483663 -1.6948183 1.3240001 -0.8875114 -0.2790751 -0.28996199  
## 3 2.292019 0.5537819 0.2028140 0.4763461 0.2948282 0.07010778  
## 4 2.350462 -0.2731776 -1.5442805 0.1312974 0.2260422 -0.41877992  
## 5 1.259095 -0.6934915 0.3911153 -0.1329304 0.2391956 -0.11518484  
## 6 -1.326783 1.0001694 -0.4836604 -0.7715705 0.2412477 1.10978569  
## PC7 PC8  
## 1 0.1100176 -0.05937806  
## 2 0.3795090 -0.02460115  
## 3 -0.3638014 0.11301354  
## 4 -0.3298129 0.09024576  
## 5 0.3493761 -0.18264299  
## 6 0.1913661 0.13902258

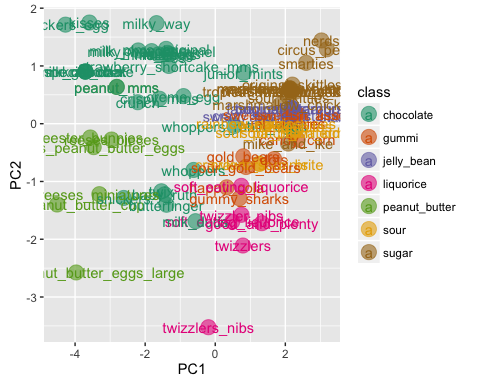
That's nice, but it would be better if the scores also had the associated data from the original dataset linked to the PC scores. Let's use cbind() to combine these columns into a single dataset

pca\_scores <- cbind(data, scores)  
head(pca\_scores)

## id name company class serving\_size\_g  
## 1 id\_1 mini\_eggs cadbury chocolate 40  
## 2 id\_2 soft\_eating\_liquorice darrell\_lea liquorice 42  
## 3 id\_3 raspberries haribo sugar 39  
## 4 id\_4 candy\_corn nice gummi 41  
## 5 id\_5 crawlers\_minis trolli sour 40  
## 6 id\_6 strawberry\_shortcake\_mms mars chocolate 42  
## calories calories\_fat total\_fat\_g saturated\_fat\_g cholesterol\_mg  
## 1 190 70 8 5 5  
## 2 140 10 1 0 0  
## 3 140 0 0 0 0  
## 4 160 160 0 0 0  
## 5 130 0 0 0 0  
## 6 210 100 10 6 5  
## sodium\_mg total\_carb\_g dietary\_fiber\_g sugars\_g protein\_g  
## 1 30 28 0.5 27 2  
## 2 40 30 0.0 16 1  
## 3 0 36 0.0 29 1  
## 4 75 39 0.0 32 0  
## 5 35 31 0.0 24 1  
## 6 40 29 0.0 28 2  
## primary\_ingredient total\_fat\_per\_serv saturated\_fat\_per\_serv  
## 1 chocolate 0.20000000 0.1250000  
## 2 syrup 0.02380952 0.0000000  
## 3 sugar 0.00000000 0.0000000  
## 4 sugar 0.00000000 0.0000000  
## 5 syrup 0.00000000 0.0000000  
## 6 chocolate 0.23809524 0.1428571  
## cholesterol\_per\_serv sodium\_per\_serv total\_carb\_per\_serv  
## 1 0.1250000 0.750000 0.7000000  
## 2 0.0000000 0.952381 0.7142857  
## 3 0.0000000 0.000000 0.9230769  
## 4 0.0000000 1.829268 0.9512195  
## 5 0.0000000 0.875000 0.7750000  
## 6 0.1190476 0.952381 0.6904762  
## dietary\_fiber\_per\_serv sugars\_per\_serv protein\_per\_serv PC1  
## 1 0.0125 0.6750000 0.05000000 -1.360340  
## 2 0.0000 0.3809524 0.02380952 0.483663  
## 3 0.0000 0.7435897 0.02564103 2.292019  
## 4 0.0000 0.7804878 0.00000000 2.350462  
## 5 0.0000 0.6000000 0.02500000 1.259095  
## 6 0.0000 0.6666667 0.04761905 -1.326783  
## PC2 PC3 PC4 PC5 PC6 PC7  
## 1 1.2253820 -0.2491928 -0.2812836 0.2172947 0.34206721 0.1100176  
## 2 -1.6948183 1.3240001 -0.8875114 -0.2790751 -0.28996199 0.3795090  
## 3 0.5537819 0.2028140 0.4763461 0.2948282 0.07010778 -0.3638014  
## 4 -0.2731776 -1.5442805 0.1312974 0.2260422 -0.41877992 -0.3298129  
## 5 -0.6934915 0.3911153 -0.1329304 0.2391956 -0.11518484 0.3493761  
## 6 1.0001694 -0.4836604 -0.7715705 0.2412477 1.10978569 0.1913661  
## PC8  
## 1 -0.05937806  
## 2 -0.02460115  
## 3 0.11301354  
## 4 0.09024576  
## 5 -0.18264299  
## 6 0.13902258

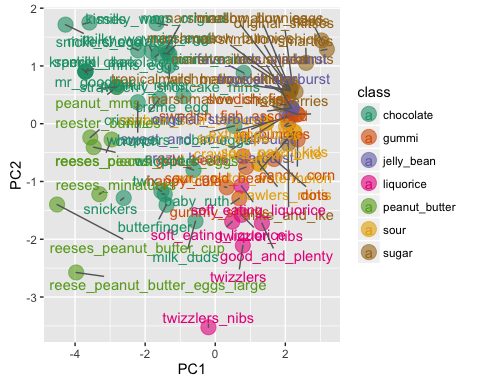
That's better. Now we can visualize our PCA results to see how it separates different candy types by their nutritional lable information

p <- ggplot(pca\_scores, aes(PC1, PC2, colour=class))  
p + geom\_point(size=5, alpha=0.6) + geom\_text(data=pca\_scores, aes(x=PC1, y=PC2, label=name)) + scale\_colour\_brewer(type="qual", palette=2)



The overlap of the labels with the data isn't good, so let's use ggrepel to fix that

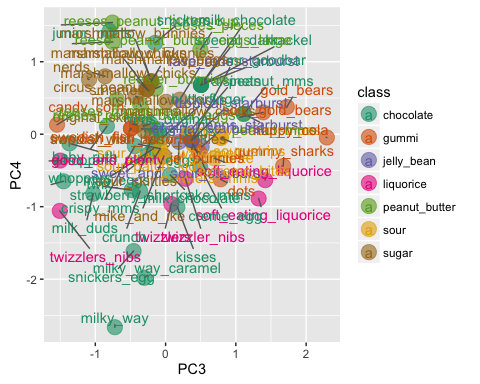
p <- ggplot(pca\_scores, aes(PC1, PC2, colour=class))  
p + geom\_point(size=5, alpha=0.6) + geom\_text\_repel(data=pca\_scores, aes(x=PC1, y=PC2, label=name)) + scale\_colour\_brewer(type="qual", palette=2)



Nice! The PCA does a good job of separating by groups we know to exist, like chocolate, peanut butter, gummi, helly bean, liquorice, sour, and sugar candies.

We can also see what PC3 and PC4 look like too

p <- ggplot(pca\_scores, aes(PC3, PC4, colour=class))  
p + geom\_point(size=5, alpha=0.6) + geom\_text\_repel(data=pca\_scores, aes(x=PC3, y=PC4, label=name)) + scale\_colour\_brewer(type="qual", palette=2)



# Hierarchical Clustering

As mentioned above, it is important to scale our data (column means=0 and variance=1) so that scaling effects are not detected, just patterns of variance between variables. Let's use the scale function here to scale our data before performing hierarchical clustering. The column names in the object data currently correspond to nutrition information. So, the samples to be clustered will be the nutrition informtion itself. But after scaling the nutrition information, let's transpose that dataset and analyze a scaled matrix where the candies are the columns, so that we will cluster by candy type. We will analyze the two datasets, scaled\_nutrition and scaled\_candies in parallel.

scaled\_nutrition <- scale(data[17:24])  
scaled\_candies <- scale(t(scaled\_nutrition))  
colnames(scaled\_candies) <- as.matrix(data[2])  
  
head(scaled\_nutrition)

## total\_fat\_per\_serv saturated\_fat\_per\_serv cholesterol\_per\_serv  
## [1,] 0.7634841 0.9663872 1.1503643  
## [2,] -0.6883584 -0.9029575 -0.5626088  
## [3,] -0.8845533 -0.9029575 -0.5626088  
## [4,] -0.8845533 -0.9029575 -0.5626088  
## [5,] -0.8845533 -0.9029575 -0.5626088  
## [6,] 1.0773960 1.2334364 1.0687941  
## sodium\_per\_serv total\_carb\_per\_serv dietary\_fiber\_per\_serv  
## [1,] -0.22223971 -0.59177589 0.2390135  
## [2,] -0.01984752 -0.47038623 -0.6266729  
## [3,] -0.97228137 1.30377031 -0.6266729  
## [4,] 0.85708852 1.54290566 -0.6266729  
## [5,] -0.09723277 0.04551981 -0.6266729  
## [6,] -0.01984752 -0.67270233 -0.6266729  
## sugars\_per\_serv protein\_per\_serv  
## [1,] 0.4539947 0.4125308  
## [2,] -1.8929612 -0.3875676  
## [3,] 1.0014472 -0.3316167  
## [4,] 1.2959509 -1.1149298  
## [5,] -0.1446216 -0.3511995  
## [6,] 0.3874818 0.3397946

head(scaled\_candies)

## mini\_eggs soft\_eating\_liquorice raspberries  
## total\_fat\_per\_serv 0.6247621 0.01015079 -0.7143345  
## saturated\_fat\_per\_serv 0.9701606 -0.38152273 -0.7349530  
## cholesterol\_per\_serv 1.2833416 0.23966130 -0.3536552  
## sodium\_per\_serv -1.0532188 1.23027675 -0.8126176  
## total\_carb\_per\_serv -1.6822740 0.40798048 1.7372779  
## dietary\_fiber\_per\_serv -0.2680353 0.12273539 -0.4254271  
## candy\_corn crawlers\_minis strawberry\_shortcake\_mms  
## total\_fat\_per\_serv -0.7657406 -1.2313422 0.9686218  
## saturated\_fat\_per\_serv -0.7826166 -1.2823811 1.1759707  
## cholesterol\_per\_serv -0.4705286 -0.3385209 0.9571914  
## sodium\_per\_serv 0.8312848 0.9520672 -0.4894129  
## total\_carb\_per\_serv 1.4601555 1.3479508 -1.3569368  
## dietary\_fiber\_per\_serv -0.5292732 -0.5161844 -1.2957722  
## milk\_chocolate milk\_duds marshmallow\_chicks  
## total\_fat\_per\_serv 0.6900816 0.5508796 -0.7670138  
## saturated\_fat\_per\_serv 0.6346596 0.6135935 -0.7880509  
## cholesterol\_per\_serv 0.2653056 -0.5088880 -0.3990132  
## sodium\_per\_serv -0.6119204 2.0490322 -0.4590349  
## total\_carb\_per\_serv -1.5293963 -0.6148121 1.0939546  
## dietary\_fiber\_per\_serv 1.4187958 -0.5806734 -0.4722420  
## crazy\_beans\_starburst creme\_egg mms\_eggs  
## total\_fat\_per\_serv -0.58098095 0.7934789 0.7089045  
## saturated\_fat\_per\_serv -0.61223115 1.2553936 0.7024987  
## cholesterol\_per\_serv -0.03432204 0.5928605 1.0226337  
## sodium\_per\_serv -0.72994244 -0.9790771 -1.1962131  
## total\_carb\_per\_serv 1.71969022 -0.9963272 -1.5514798  
## dietary\_fiber\_per\_serv -0.14310228 -1.1330215 0.9447997  
## gold\_bears original\_skittles crawlers\_sour\_brite  
## total\_fat\_per\_serv -0.6517637 -0.4290930 -1.2313422  
## saturated\_fat\_per\_serv -0.6761136 -0.1649326 -1.2823811  
## cholesterol\_per\_serv -0.2258115 -0.4144500 -0.3385209  
## sodium\_per\_serv -0.7678333 -0.5951565 0.9520672  
## total\_carb\_per\_serv 0.8021868 1.7164702 1.3479508  
## dietary\_fiber\_per\_serv -0.3105722 -0.4869597 -0.5161844  
## reeses\_pieces milky\_way\_caramel raisinets  
## total\_fat\_per\_serv 0.4032352 0.3556871 0.4621047  
## saturated\_fat\_per\_serv 1.2972586 0.6664809 0.7496776  
## cholesterol\_per\_serv -0.8213735 1.9782369 1.0744053  
## sodium\_per\_serv -0.2495609 -0.2344105 -1.5043053  
## total\_carb\_per\_serv -1.3540784 -1.0780661 -1.2759934  
## dietary\_fiber\_per\_serv 0.5113097 -0.9672375 0.9974636  
## circus\_peanuts sweet\_and\_sour\_starburst  
## total\_fat\_per\_serv -0.5940916 -0.6433016  
## saturated\_fat\_per\_serv -0.6072432 -0.6711654  
## cholesterol\_per\_serv -0.3640293 -0.1558821  
## sodium\_per\_serv -0.4687186 -0.3976031  
## total\_carb\_per\_serv 1.4367796 1.7296733  
## dietary\_fiber\_per\_serv -0.4098097 -0.2528743  
## reeses\_peanut\_butter\_cup whoppers\_robin\_eggs  
## total\_fat\_per\_serv 0.39908807 -0.002694724  
## saturated\_fat\_per\_serv -0.06731952 1.184399770  
## cholesterol\_per\_serv -0.43154124 -1.026618608  
## sodium\_per\_serv 0.94956668 0.730947067  
## total\_carb\_per\_serv -1.66084978 0.774338434  
## dietary\_fiber\_per\_serv 0.99689223 -1.119259034  
## twizzlers nerds sour\_patch\_watermelon  
## total\_fat\_per\_serv -0.3837281 -0.5516585 -0.6601577  
## saturated\_fat\_per\_serv -0.5319207 -0.5653804 -0.6832339  
## cholesterol\_per\_serv -0.1164805 -0.3116231 -0.2564865  
## sodium\_per\_serv 1.9774459 -0.6170669 0.0135461  
## total\_carb\_per\_serv 0.8851214 1.1448908 2.1041687  
## dietary\_fiber\_per\_serv -0.1946791 -0.3593880 -0.3368135  
## crispy\_mms snickers\_egg swedish\_fish\_assorted  
## total\_fat\_per\_serv 0.04211436 0.3177629 -0.69934765  
## saturated\_fat\_per\_serv 0.46741840 0.3185124 -0.72065503  
## cholesterol\_per\_serv 1.21379022 1.7602410 -0.32661819  
## sodium\_per\_serv 0.21708791 -0.1153662 0.02609201  
## total\_carb\_per\_serv -1.48693012 -1.6140235 1.65399497  
## dietary\_fiber\_per\_serv 1.11976952 0.3798587 -0.40078802  
## sour\_skittles rainbow\_sour\_stripes peanut\_mms  
## total\_fat\_per\_serv -0.4058366 -0.3878780 0.7742583  
## saturated\_fat\_per\_serv -0.1181373 -0.2933576 0.2556515  
## cholesterol\_per\_serv -0.3898887 -0.3615561 0.5893067  
## sodium\_per\_serv -0.5866977 -1.1950542 -0.8239140  
## total\_carb\_per\_serv 1.9309141 2.1723185 -1.5315691  
## dietary\_fiber\_per\_serv -0.4688596 -0.4918975 0.5472760  
## baby\_ruth junior\_mints marshmallow\_bunnies  
## total\_fat\_per\_serv 0.72913244 -0.6840201 -0.7898629  
## saturated\_fat\_per\_serv 0.72312786 -0.2966324 -0.8101718  
## cholesterol\_per\_serv -0.92908588 -1.0685705 -0.4346016  
## sodium\_per\_serv 1.61547265 -0.6264816 -0.5418110  
## total\_carb\_per\_serv -1.31973972 0.8249680 1.1741231  
## dietary\_fiber\_per\_serv -0.04697188 1.0970156 -0.5052954  
## fave\_reds\_starburst reeses\_peanut\_butter\_eggs  
## total\_fat\_per\_serv -0.5786255 0.9417943  
## saturated\_fat\_per\_serv -0.6056776 0.7235978  
## cholesterol\_per\_serv -0.1054048 -0.2183600  
## sodium\_per\_serv -0.7075755 0.4499203  
## total\_carb\_per\_serv 1.7252235 -1.8671754  
## dietary\_fiber\_per\_serv -0.1995715 0.5027051  
## butterfinger milk\_chocolate special\_dark mr\_goodbar  
## total\_fat\_per\_serv 0.8437179 1.0226564 0.6900816 0.6900816  
## saturated\_fat\_per\_serv 0.6282753 1.3720806 0.6346596 0.6346596  
## cholesterol\_per\_serv -0.8273575 0.7798505 0.2653056 0.2653056  
## sodium\_per\_serv 1.1768329 -0.9247076 -0.6119204 -0.6119204  
## total\_carb\_per\_serv -0.5964164 -1.4965845 -1.5293963 -1.5293963  
## dietary\_fiber\_per\_serv 0.1828065 -0.1775223 1.4187958 1.4187958  
## snickers twix marshmallow\_chicks  
## total\_fat\_per\_serv 0.573977821 1.1608674 -0.7670138  
## saturated\_fat\_per\_serv 0.005825866 0.9295415 -0.7880509  
## cholesterol\_per\_serv -0.866524822 -0.7619254 -0.3990132  
## sodium\_per\_serv 0.864832622 0.8592227 -0.4590349  
## total\_carb\_per\_serv -1.515933517 -0.9116942 1.0939546  
## dietary\_fiber\_per\_serv 0.891065641 -0.8327969 -0.4722420  
## marshmallow\_bunnies whoppers twizzlers\_nibs  
## total\_fat\_per\_serv -0.7898629 0.4020778 -0.35114768  
## saturated\_fat\_per\_serv -0.8101718 1.6240890 -0.37555585  
## cholesterol\_per\_serv -0.4346016 -0.7732742 -0.27563274  
## sodium\_per\_serv -0.5418110 1.2932608 2.38867861  
## total\_carb\_per\_serv 1.1741231 -0.2884618 -0.01790401  
## dietary\_fiber\_per\_serv -0.5052954 -0.8426784 -0.31735953  
## swedish\_fish dots twizzler\_nibs jel\_bunnies  
## total\_fat\_per\_serv -0.69934765 -0.55235711 -0.4879646 -0.59022754  
## saturated\_fat\_per\_serv -0.72065503 -0.59081639 -0.5526435 -0.61940310  
## cholesterol\_per\_serv -0.32661819 0.12041018 -0.2878586 -0.07986019  
## sodium\_per\_serv 0.02609201 0.04799853 1.8098892 -0.54495639  
## total\_carb\_per\_serv 1.65399497 2.27905434 1.1283863 2.03543739  
## dietary\_fiber\_per\_serv -0.40078802 -0.01346450 -0.3984298 -0.18141873  
## gummy\_sharks good\_and\_plenty mike\_and\_ike  
## total\_fat\_per\_serv -0.49588358 -0.7564517 -0.6852924  
## saturated\_fat\_per\_serv -0.52780342 -0.7740060 -0.7109985  
## cholesterol\_per\_serv 0.06248932 -0.4493752 -0.2356156  
## sodium\_per\_serv 0.15869446 2.0214855 0.2397959  
## total\_carb\_per\_serv 1.00582544 0.9411548 2.0973638  
## dietary\_fiber\_per\_serv -0.04862190 -0.5104807 -0.3250973  
## sour\_patch\_kids sour\_strips kisses krackel  
## total\_fat\_per\_serv -0.6601577 -0.3878780 0.5640830 0.6900816  
## saturated\_fat\_per\_serv -0.6832339 -0.2933576 0.6564900 0.6346596  
## cholesterol\_per\_serv -0.2564865 -0.3615561 1.5044583 0.2653056  
## sodium\_per\_serv 0.0135461 -1.1950542 -0.6715016 -0.6119204  
## total\_carb\_per\_serv 2.1041687 2.1723185 -1.6024616 -1.5293963  
## dietary\_fiber\_per\_serv -0.3368135 -0.4918975 0.2151779 1.4187958  
## reester\_bunnies marshmallow\_chicks happy\_cola  
## total\_fat\_per\_serv 0.9502916 -0.7670138 -0.37560637  
## saturated\_fat\_per\_serv 0.5426627 -0.7880509 -0.39881442  
## cholesterol\_per\_serv -0.1726143 -0.3990132 0.03037146  
## sodium\_per\_serv 0.2429543 -0.4590349 -0.48623306  
## total\_carb\_per\_serv -1.8418005 1.0939546 0.26147177  
## dietary\_fiber\_per\_serv 0.6384235 -0.4722420 -0.05041450  
## mms\_original smarties soft\_eating\_liquorice  
## total\_fat\_per\_serv 0.3826890 -0.5466747 -0.091906779  
## saturated\_fat\_per\_serv 0.7450218 -0.5639333 -0.189137192  
## cholesterol\_per\_serv 1.1055817 -0.2447711 0.155368005  
## sodium\_per\_serv -1.3934494 -0.6289418 0.144791434  
## total\_carb\_per\_serv -1.4125001 0.9800433 1.284414721  
## dietary\_fiber\_per\_serv 1.0179193 -0.3048472 -0.004264772  
## tropical\_wild\_berry\_skittles sour\_gold\_bears  
## total\_fat\_per\_serv -0.4190331 -0.7622266  
## saturated\_fat\_per\_serv -0.1428986 -0.7873358  
## cholesterol\_per\_serv -0.4037264 -0.3229913  
## sodium\_per\_serv -0.5926242 -0.3827465  
## total\_carb\_per\_serv 1.8237862 1.1358232  
## dietary\_fiber\_per\_serv -0.4795229 -0.4103952  
## peanut\_mms extreme\_bites crunch milky\_way  
## total\_fat\_per\_serv 0.7742583 -0.542000238 0.9290599 0.04530828  
## saturated\_fat\_per\_serv 0.2556515 -0.573287073 1.1363936 0.42777271  
## cholesterol\_per\_serv 0.5893067 0.005299495 0.8974378 2.23277955  
## sodium\_per\_serv -0.8239140 -0.691136354 0.1140189 -0.11943278  
## total\_carb\_per\_serv -1.5315691 2.122498274 -1.5621533 -0.79349944  
## dietary\_fiber\_per\_serv 0.5472760 -0.103608265 -0.2493862 -0.85756236  
## marshmallow\_eggs tropical\_starburst  
## total\_fat\_per\_serv -0.7898629 -0.5786255  
## saturated\_fat\_per\_serv -0.8101718 -0.6056776  
## cholesterol\_per\_serv -0.4346016 -0.1054048  
## sodium\_per\_serv -0.5418110 -0.7075755  
## total\_carb\_per\_serv 1.1741231 1.7252235  
## dietary\_fiber\_per\_serv -0.5052954 -0.1995715  
## reeses\_miniatures reese\_peanut\_butter\_eggs\_large  
## total\_fat\_per\_serv 0.7648285 0.5073612  
## saturated\_fat\_per\_serv 0.1521519 -0.1210199  
## cholesterol\_per\_serv -0.3197383 -0.6691984  
## sodium\_per\_serv 1.1163536 1.3244299  
## total\_carb\_per\_serv -1.7301019 -1.4970192  
## dietary\_fiber\_per\_serv 0.2747974 0.4352581  
## original\_starburst marshmallow\_chicks  
## total\_fat\_per\_serv -0.5786255 -0.8113357  
## saturated\_fat\_per\_serv -0.6056776 -0.8373981  
## cholesterol\_per\_serv -0.1054048 -0.3554263  
## sodium\_per\_serv -0.7075755 -0.4297861  
## total\_carb\_per\_serv 1.7252235 0.9211802  
## dietary\_fiber\_per\_serv -0.1995715 -0.4461482

The way similarity between variables is calculated is through correlation. Let's use the cor function to create a pairwise correlation matirx for our data, the correlation of all variable to each other. Let's be conservative and instead of using a parametric correlation parameter like "pearson", we'll use a non-parametric one, like "spearman", which makes no assumptions about the distributions of our data

corell\_nutrition <- cor(scaled\_nutrition, method="spearman")  
corell\_candies <- cor(scaled\_candies, method="spearman")  
  
head(corell\_nutrition)

## total\_fat\_per\_serv saturated\_fat\_per\_serv  
## total\_fat\_per\_serv 1.0000000 0.9368297  
## saturated\_fat\_per\_serv 0.9368297 1.0000000  
## cholesterol\_per\_serv 0.7231604 0.7391776  
## sodium\_per\_serv 0.5766610 0.5181643  
## total\_carb\_per\_serv -0.8333848 -0.7492149  
## dietary\_fiber\_per\_serv 0.8152061 0.7129886  
## cholesterol\_per\_serv sodium\_per\_serv  
## total\_fat\_per\_serv 0.7231604 0.5766610  
## saturated\_fat\_per\_serv 0.7391776 0.5181643  
## cholesterol\_per\_serv 1.0000000 0.2921460  
## sodium\_per\_serv 0.2921460 1.0000000  
## total\_carb\_per\_serv -0.6637718 -0.5644365  
## dietary\_fiber\_per\_serv 0.6441498 0.4431272  
## total\_carb\_per\_serv dietary\_fiber\_per\_serv  
## total\_fat\_per\_serv -0.8333848 0.8152061  
## saturated\_fat\_per\_serv -0.7492149 0.7129886  
## cholesterol\_per\_serv -0.6637718 0.6441498  
## sodium\_per\_serv -0.5644365 0.4431272  
## total\_carb\_per\_serv 1.0000000 -0.7332656  
## dietary\_fiber\_per\_serv -0.7332656 1.0000000  
## sugars\_per\_serv protein\_per\_serv  
## total\_fat\_per\_serv -0.5303592 0.7249913  
## saturated\_fat\_per\_serv -0.4071991 0.6302843  
## cholesterol\_per\_serv -0.2347752 0.5786847  
## sodium\_per\_serv -0.5220287 0.4196859  
## total\_carb\_per\_serv 0.7290826 -0.8234517  
## dietary\_fiber\_per\_serv -0.3983687 0.7196518

head(corell\_candies)

## mini\_eggs soft\_eating\_liquorice raspberries  
## mini\_eggs 1.0000000 -0.54761905 -0.21428571  
## soft\_eating\_liquorice -0.5476190 1.00000000 -0.07142857  
## raspberries -0.2142857 -0.07142857 1.00000000  
## candy\_corn -0.4523810 0.04761905 0.40476190  
## crawlers\_minis -0.7142857 0.54761905 0.50000000  
## strawberry\_shortcake\_mms 0.9047619 -0.57142857 -0.40476190  
## candy\_corn crawlers\_minis  
## mini\_eggs -0.45238095 -0.7142857  
## soft\_eating\_liquorice 0.04761905 0.5476190  
## raspberries 0.40476190 0.5000000  
## candy\_corn 1.00000000 0.7380952  
## crawlers\_minis 0.73809524 1.0000000  
## strawberry\_shortcake\_mms -0.52380952 -0.7380952  
## strawberry\_shortcake\_mms milk\_chocolate  
## mini\_eggs 0.9047619 0.4523810  
## soft\_eating\_liquorice -0.5714286 -0.2857143  
## raspberries -0.4047619 -0.5476190  
## candy\_corn -0.5238095 -0.6666667  
## crawlers\_minis -0.7380952 -0.8809524  
## strawberry\_shortcake\_mms 1.0000000 0.4047619  
## milk\_duds marshmallow\_chicks  
## mini\_eggs 0.1666667 -0.3571429  
## soft\_eating\_liquorice 0.1904762 0.1666667  
## raspberries -0.9047619 0.8333333  
## candy\_corn -0.1904762 0.5952381  
## crawlers\_minis -0.3571429 0.8095238  
## strawberry\_shortcake\_mms 0.3333333 -0.5000000  
## crazy\_beans\_starburst creme\_egg mms\_eggs  
## mini\_eggs -0.07142857 0.6904762 0.7142857  
## soft\_eating\_liquorice -0.38095238 -0.2380952 -0.3095238  
## raspberries 0.64285714 -0.2380952 -0.2619048  
## candy\_corn 0.76190476 -0.7142857 -0.5000000  
## crawlers\_minis 0.30952381 -0.5476190 -0.7619048  
## strawberry\_shortcake\_mms -0.26190476 0.8333333 0.4761905  
## gold\_bears original\_skittles crawlers\_sour\_brite  
## mini\_eggs -0.14285714 0.09523810 -0.7142857  
## soft\_eating\_liquorice 0.26190476 -0.54761905 0.5476190  
## raspberries 0.80952381 0.47619048 0.5000000  
## candy\_corn -0.02380952 0.59523810 0.7380952  
## crawlers\_minis 0.28571429 0.14285714 1.0000000  
## strawberry\_shortcake\_mms -0.40476190 0.07142857 -0.7380952  
## reeses\_pieces milky\_way\_caramel raisinets  
## mini\_eggs 0.30952381 0.8809524 0.7142857  
## soft\_eating\_liquorice -0.07142857 -0.1428571 -0.3809524  
## raspberries -0.57142857 -0.4523810 -0.1190476  
## candy\_corn -0.92857143 -0.5714286 -0.4761905  
## crawlers\_minis -0.73809524 -0.6190476 -0.7619048  
## strawberry\_shortcake\_mms 0.42857143 0.8571429 0.4523810  
## circus\_peanuts sweet\_and\_sour\_starburst  
## mini\_eggs -0.1904762 -0.2857143  
## soft\_eating\_liquorice -0.2380952 -0.1190476  
## raspberries 0.5476190 0.5714286  
## candy\_corn 0.9047619 0.9285714  
## crawlers\_minis 0.5238095 0.5714286  
## strawberry\_shortcake\_mms -0.3809524 -0.4761905  
## reeses\_peanut\_butter\_cup whoppers\_robin\_eggs  
## mini\_eggs -0.1666667 -0.02380952  
## soft\_eating\_liquorice 0.3333333 -0.38095238  
## raspberries -0.6428571 -0.04761905  
## candy\_corn -0.5238095 0.42857143  
## crawlers\_minis -0.3095238 0.09523810  
## strawberry\_shortcake\_mms -0.1190476 0.19047619  
## twizzlers nerds sour\_patch\_watermelon  
## mini\_eggs -0.5476190 0.02380952 -0.45238095  
## soft\_eating\_liquorice 0.8571429 -0.50000000 0.04761905  
## raspberries -0.0952381 0.61904762 0.40476190  
## candy\_corn 0.3809524 0.73809524 1.00000000  
## crawlers\_minis 0.5476190 0.26190476 0.73809524  
## strawberry\_shortcake\_mms -0.6428571 -0.16666667 -0.52380952  
## crispy\_mms snickers\_egg swedish\_fish\_assorted  
## mini\_eggs 0.52380952 0.6666667 -0.45238095  
## soft\_eating\_liquorice -0.02380952 -0.1666667 0.04761905  
## raspberries -0.59523810 -0.4285714 0.40476190  
## candy\_corn -0.35714286 -0.5476190 1.00000000  
## crawlers\_minis -0.59523810 -0.7619048 0.73809524  
## strawberry\_shortcake\_mms 0.33333333 0.4523810 -0.52380952  
## sour\_skittles rainbow\_sour\_stripes peanut\_mms  
## mini\_eggs 0.09523810 -0.04761905 0.4761905  
## soft\_eating\_liquorice -0.54761905 -0.23809524 0.1190476  
## raspberries 0.47619048 0.83333333 -0.2380952  
## candy\_corn 0.59523810 0.23809524 -0.8333333  
## crawlers\_minis 0.14285714 0.33333333 -0.5714286  
## strawberry\_shortcake\_mms 0.07142857 -0.07142857 0.4047619  
## baby\_ruth junior\_mints marshmallow\_bunnies  
## mini\_eggs 0.04761905 -0.4285714 -0.33333333  
## soft\_eating\_liquorice 0.11904762 -0.4761905 0.07142857  
## raspberries -0.85714286 0.2380952 0.90476190  
## candy\_corn -0.50000000 0.4761905 0.54761905  
## crawlers\_minis -0.35714286 0.1904762 0.71428571  
## strawberry\_shortcake\_mms 0.35714286 -0.3571429 -0.52380952  
## fave\_reds\_starburst reeses\_peanut\_butter\_eggs  
## mini\_eggs -0.07142857 0.4047619  
## soft\_eating\_liquorice -0.38095238 -0.1666667  
## raspberries 0.64285714 -0.6428571  
## candy\_corn 0.76190476 -0.8809524  
## crawlers\_minis 0.30952381 -0.8095238  
## strawberry\_shortcake\_mms -0.26190476 0.5952381  
## butterfinger milk\_chocolate special\_dark  
## mini\_eggs -0.1428571 0.8571429 0.4523810  
## soft\_eating\_liquorice 0.3571429 -0.4523810 -0.2857143  
## raspberries -0.8095238 -0.4285714 -0.5476190  
## candy\_corn -0.4523810 -0.7857143 -0.6666667  
## crawlers\_minis -0.2619048 -0.9285714 -0.8809524  
## strawberry\_shortcake\_mms 0.1666667 0.8809524 0.4047619  
## mr\_goodbar snickers twix  
## mini\_eggs 0.4523810 -0.09523810 0.3571429  
## soft\_eating\_liquorice -0.2857143 0.38095238 0.0952381  
## raspberries -0.5476190 -0.47619048 -0.7857143  
## candy\_corn -0.6666667 -0.71428571 -0.6666667  
## crawlers\_minis -0.8809524 -0.30952381 -0.5952381  
## strawberry\_shortcake\_mms 0.4047619 -0.04761905 0.5952381  
## marshmallow\_chicks marshmallow\_bunnies  
## mini\_eggs -0.3571429 -0.33333333  
## soft\_eating\_liquorice 0.1666667 0.07142857  
## raspberries 0.8333333 0.90476190  
## candy\_corn 0.5952381 0.54761905  
## crawlers\_minis 0.8095238 0.71428571  
## strawberry\_shortcake\_mms -0.5000000 -0.52380952  
## whoppers twizzlers\_nibs swedish\_fish  
## mini\_eggs 0.07142857 -0.61904762 -0.45238095  
## soft\_eating\_liquorice -0.19047619 0.97619048 0.04761905  
## raspberries -0.57142857 -0.02380952 0.40476190  
## candy\_corn 0.14285714 0.21428571 1.00000000  
## crawlers\_minis -0.14285714 0.61904762 0.73809524  
## strawberry\_shortcake\_mms 0.38095238 -0.64285714 -0.52380952  
## dots twizzler\_nibs jel\_bunnies gummy\_sharks  
## mini\_eggs -0.3571429 -0.61904762 -0.2857143 -0.52380952  
## soft\_eating\_liquorice 0.3095238 0.97619048 -0.1190476 0.92857143  
## raspberries 0.2380952 -0.02380952 0.5714286 0.21428571  
## candy\_corn 0.8333333 0.21428571 0.9285714 -0.02380952  
## crawlers\_minis 0.5714286 0.61904762 0.5714286 0.52380952  
## strawberry\_shortcake\_mms -0.5476190 -0.64285714 -0.4761905 -0.59523810  
## good\_and\_plenty mike\_and\_ike sour\_patch\_kids  
## mini\_eggs -0.6190476 -0.45238095 -0.45238095  
## soft\_eating\_liquorice 0.4761905 0.04761905 0.04761905  
## raspberries 0.2619048 0.40476190 0.40476190  
## candy\_corn 0.8571429 1.00000000 1.00000000  
## crawlers\_minis 0.9047619 0.73809524 0.73809524  
## strawberry\_shortcake\_mms -0.6904762 -0.52380952 -0.52380952  
## sour\_strips kisses krackel reester\_bunnies  
## mini\_eggs -0.04761905 0.8571429 0.4523810 0.2380952  
## soft\_eating\_liquorice -0.23809524 -0.1666667 -0.2857143 0.0000000  
## raspberries 0.83333333 -0.4523810 -0.5476190 -0.5000000  
## candy\_corn 0.23809524 -0.6904762 -0.6666667 -0.8571429  
## crawlers\_minis 0.33333333 -0.7857143 -0.8809524 -0.6666667  
## strawberry\_shortcake\_mms -0.07142857 0.7619048 0.4047619 0.3571429  
## marshmallow\_chicks happy\_cola mms\_original  
## mini\_eggs -0.3571429 -0.1428571 0.7619048  
## soft\_eating\_liquorice 0.1666667 0.5000000 -0.4761905  
## raspberries 0.8333333 0.4523810 -0.2619048  
## candy\_corn 0.5952381 -0.2619048 -0.3809524  
## crawlers\_minis 0.8095238 0.0952381 -0.7619048  
## strawberry\_shortcake\_mms -0.5000000 -0.3333333 0.5238095  
## smarties soft\_eating\_liquorice  
## mini\_eggs 0.02380952 -0.4523810  
## soft\_eating\_liquorice -0.50000000 0.8333333  
## raspberries 0.61904762 0.3571429  
## candy\_corn 0.73809524 0.1190476  
## crawlers\_minis 0.26190476 0.5238095  
## strawberry\_shortcake\_mms -0.16666667 -0.5952381  
## tropical\_wild\_berry\_skittles sour\_gold\_bears  
## mini\_eggs 0.09523810 -0.4523810  
## soft\_eating\_liquorice -0.54761905 0.7857143  
## raspberries 0.47619048 0.5238095  
## candy\_corn 0.59523810 0.1666667  
## crawlers\_minis 0.14285714 0.6666667  
## strawberry\_shortcake\_mms 0.07142857 -0.6428571  
## peanut\_mms extreme\_bites crunch milky\_way  
## mini\_eggs 0.4761905 -0.07142857 0.7142857 0.8095238  
## soft\_eating\_liquorice 0.1190476 -0.38095238 -0.1666667 -0.2142857  
## raspberries -0.2380952 0.64285714 -0.7619048 -0.4523810  
## candy\_corn -0.8333333 0.76190476 -0.6428571 -0.2142857  
## crawlers\_minis -0.5714286 0.30952381 -0.7619048 -0.4285714  
## strawberry\_shortcake\_mms 0.4047619 -0.26190476 0.8333333 0.8333333  
## marshmallow\_eggs tropical\_starburst  
## mini\_eggs -0.33333333 -0.07142857  
## soft\_eating\_liquorice 0.07142857 -0.38095238  
## raspberries 0.90476190 0.64285714  
## candy\_corn 0.54761905 0.76190476  
## crawlers\_minis 0.71428571 0.30952381  
## strawberry\_shortcake\_mms -0.52380952 -0.26190476  
## reeses\_miniatures reese\_peanut\_butter\_eggs\_large  
## mini\_eggs -0.09523810 -0.09523810  
## soft\_eating\_liquorice 0.47619048 0.47619048  
## raspberries -0.69047619 -0.69047619  
## candy\_corn -0.57142857 -0.57142857  
## crawlers\_minis -0.19047619 -0.19047619  
## strawberry\_shortcake\_mms 0.07142857 0.07142857  
## original\_starburst marshmallow\_chicks  
## mini\_eggs -0.07142857 -0.3571429  
## soft\_eating\_liquorice -0.38095238 0.1666667  
## raspberries 0.64285714 0.8333333  
## candy\_corn 0.76190476 0.5952381  
## crawlers\_minis 0.30952381 0.8095238  
## strawberry\_shortcake\_mms -0.26190476 -0.5000000

Next, let's create a distance matrix using the as.dist funciton, which assigns distances between our samples using the correlation matrix we just calculated. There are many different methods to calculate distance, and the default is Eucledian, which we'll be using. We will take the absolute value of the correlation matirx (because strong negative as well as positive correlations indicate similarity) and also subtract that value from 1, as strong correlation (e.g., rho=1) means a closer distance (e.g., 0)

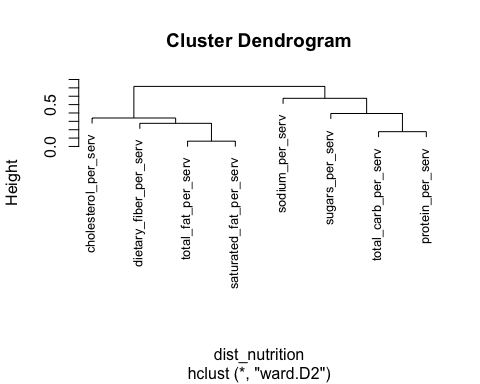
dist\_nutrition <- as.dist(1-abs(corell\_nutrition))  
dist\_candies <- as.dist(1-abs(corell\_candies))

Finally, using our distance matrix, let's convert it to a dendrogram using the hclust function. Check out ?hclust, as there are many different methods by which to cluster your results. For this example, we'll start by using the method ward.D

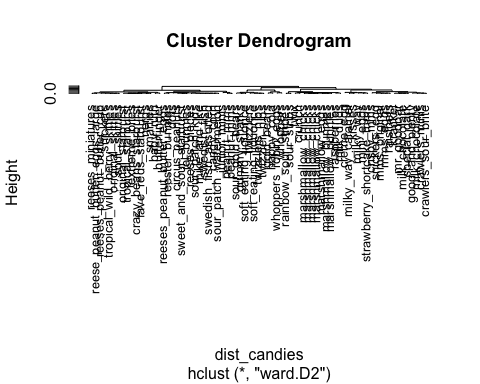
hc\_nutrition <- hclust(dist\_nutrition, method="ward.D2")  
hc\_candies <- hclust(dist\_candies, method="ward.D2")

Let's plot out our dendrograms!

plot(hc\_nutrition, cex=0.8)

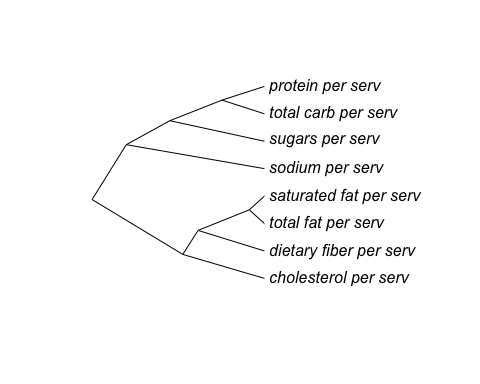


plot(hc\_candies, cex=0.8)

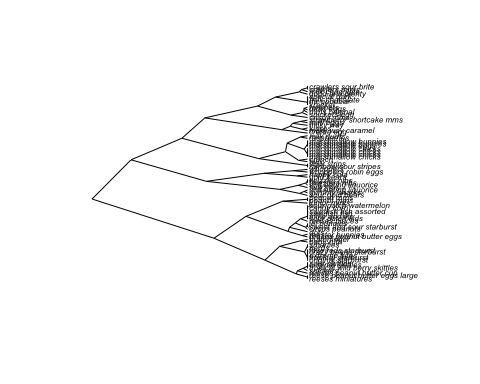


Using the packages ggdendro and ape you can plot out nicer looking dendrograms if you like. Use the "cex" option to change the size of the text if you need to

plot(as.phylo(hc\_nutrition), type="cladogram", label.offset=0.01)



plot(as.phylo(hc\_candies), type="cladogram", label.offset=0.01, cex=0.5)



# Shape descriptors and allometry

Let's read in a datafile with basic shape information about individual candy pieces

shape\_desc <- read.table("./shape\_descriptors.txt", header=TRUE)  
names(shape\_desc)

## [1] "label" "id" "candy\_no" "area" "cm2" "circ"   
## [7] "ar" "round" "solid"

head(shape\_desc)

## label id candy\_no area cm2 circ ar round solid  
## 1 ID\_52\_22.jpg id\_52 22 5721 0.8362881 0.563 2.881 0.347 0.947  
## 2 ID\_18\_15.jpg id\_18 15 5964 0.8718095 0.824 1.451 0.689 0.973  
## 3 ID\_52\_12.jpg id\_52 12 7036 1.0285130 0.697 2.242 0.446 0.970  
## 4 ID\_52\_14.jpg id\_52 14 7239 1.0581873 0.663 2.267 0.441 0.954  
## 5 ID\_52\_07.jpg id\_52 7 7383 1.0792370 0.714 2.197 0.455 0.962  
## 6 ID\_71\_10.jpg id\_71 10 7433 1.0865459 0.828 1.579 0.633 0.973

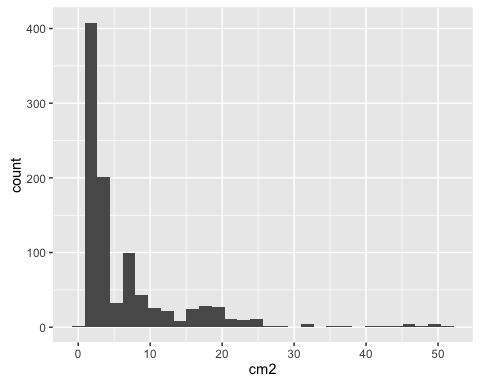
The variables in this dataset are:

* label: the name of the file measured
* id: the candy ID
* candy\_no: for a given candy ID, the individual candy piece in question
* area: area in pixels
* cm2: area in cm^2
* circ: circularity, which is 4pi(area/perimeter^2). More lanky, jagged things have smaller circ values. More smoother, rounder things have higher circ values
* ar: aspect ratio. The ratio of the major to minor axes of the best fitted ellipse. Longer things have higher aspect ratios
* round: inversely related to ar
* solidity: the ratio of the area to convex area. More jagged things have lower solidity values

Let's first look at the distribution of cm2 values using a histogram

p <- ggplot(shape\_desc, aes(x=cm2))  
p + geom\_histogram()

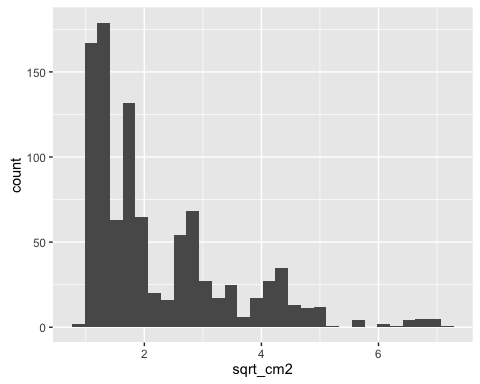
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



As is typical for area, the distribution is skewed. Let's create a transformed value for area by taking the square root and see how that looks

shape\_desc$sqrt\_cm2 <- sqrt(shape\_desc$cm2)  
  
p <- ggplot(shape\_desc, aes(x=sqrt\_cm2))  
p + geom\_histogram()

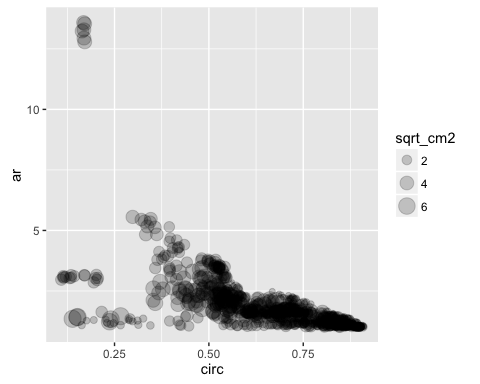
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



That's a little more normal looking, assuming discrete populations in our data (which there are). Let's use this square root value in the future.

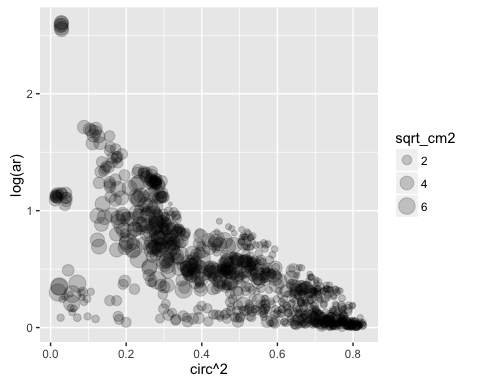
Let's get an idea for how our data looks and plot aspect ratio vs. circularity, making the size of the points sqrt\_cm2

p <- ggplot(shape\_desc, aes(x=circ, y=ar, size=sqrt\_cm2))  
p + geom\_point(alpha=0.2)



Interesting! Let's get a little more separation and transform our variables. I tried squaring circ and taking the log of ar, but you can try your own transformations!

p <- ggplot(shape\_desc, aes(x=circ^2, y=log(ar), size=sqrt\_cm2))  
p + geom\_point(alpha=0.2)



Wouldn't it be nice if we had the other information for the candies associated with this dataset? Let's use the merge function to merge our nutritional label dataset with the shape descriptor dataset

Check out ?merge

But the gist is that you input an x dataset that will be merged with a y dataset. Using by.x and by.y you can specify the columns by which to merge in each dataset. all.x or all.y or all set to TRUE will insure that every row specifed by all is included in the merge, even if there is no corresponding data in the other dataset to merge with. Let's merge our shape descriptor individual candy dataset shape\_desc with the nutritional information for each of our candy types, data

mdata <- merge(x=shape\_desc, y=data, by.x="id", by.y="id", all.x=TRUE)  
summary(mdata)

## id label candy\_no area   
## id\_16 : 50 ID\_01\_01.jpg: 1 Min. : 1.00 Min. : 5721   
## id\_14 : 30 ID\_01\_02.jpg: 1 1st Qu.: 5.00 1st Qu.: 10825   
## id\_52 : 30 ID\_01\_03.jpg: 1 Median : 9.00 Median : 20989   
## id\_64 : 28 ID\_01\_04.jpg: 1 Mean :10.73 Mean : 44678   
## id\_05 : 27 ID\_01\_05.jpg: 1 3rd Qu.:15.00 3rd Qu.: 53384   
## id\_20 : 25 ID\_01\_06.jpg: 1 Max. :50.00 Max. :356531   
## (Other):789 (Other) :973   
## cm2 circ ar round   
## Min. : 0.8363 Min. :0.1090 Min. : 1.003 Min. :0.0740   
## 1st Qu.: 1.5824 1st Qu.:0.5490 1st Qu.: 1.137 1st Qu.:0.4860   
## Median : 3.0681 Median :0.7120 Median : 1.513 Median :0.6610   
## Mean : 6.5310 Mean :0.6737 Mean : 1.817 Mean :0.6624   
## 3rd Qu.: 7.8035 3rd Qu.:0.8270 3rd Qu.: 2.058 3rd Qu.:0.8785   
## Max. :52.1172 Max. :0.9090 Max. :13.587 Max. :0.9970   
##   
## solid sqrt\_cm2 name   
## Min. :0.7120 Min. :0.9145 reeses\_pieces : 50   
## 1st Qu.:0.9200 1st Qu.:1.2579 good\_and\_plenty : 30   
## Median :0.9560 Median :1.7516 original\_skittles : 30   
## Mean :0.9385 Mean :2.2339 tropical\_wild\_berry\_skittles: 28   
## 3rd Qu.:0.9720 3rd Qu.:2.7935 mms\_original : 25   
## Max. :0.9880 Max. :7.2192 (Other) :689   
## NA's :127   
## company class serving\_size\_g calories   
## hershey :215 chocolate :217 Min. : 7.00 Min. : 25.0   
## wrigley :167 sugar :135 1st Qu.:40.00 1st Qu.:140.0   
## mars : 99 gummi :123 Median :40.00 Median :150.0   
## haribo : 62 jelly\_bean :109 Mean :39.22 Mean :158.7   
## just\_born: 60 peanut\_butter:101 3rd Qu.:41.00 3rd Qu.:190.0   
## (Other) :249 (Other) :167 Max. :45.00 Max. :220.0   
## NA's :127 NA's :127 NA's :127 NA's :127   
## calories\_fat total\_fat\_g saturated\_fat\_g cholesterol\_mg   
## Min. : 0.00 Min. : 0.000 Min. :0.000 Min. : 0.000   
## 1st Qu.: 0.00 1st Qu.: 0.000 1st Qu.:0.000 1st Qu.: 0.000   
## Median : 10.00 Median : 1.000 Median :0.500 Median : 0.000   
## Mean : 31.95 Mean : 3.639 Mean :2.228 Mean : 1.206   
## 3rd Qu.: 70.00 3rd Qu.: 8.000 3rd Qu.:5.000 3rd Qu.: 0.000   
## Max. :110.00 Max. :13.000 Max. :8.000 Max. :10.000   
## NA's :127 NA's :127 NA's :127 NA's :127   
## sodium\_mg total\_carb\_g dietary\_fiber\_g sugars\_g   
## Min. : 0.00 Min. : 6.00 Min. :0.0000 Min. : 6.00   
## 1st Qu.: 10.00 1st Qu.:26.00 1st Qu.:0.0000 1st Qu.:21.00   
## Median : 20.00 Median :32.00 Median :0.0000 Median :24.00   
## Mean : 34.19 Mean :30.59 Mean :0.3427 Mean :23.83   
## 3rd Qu.: 45.00 3rd Qu.:34.00 3rd Qu.:1.0000 3rd Qu.:27.00   
## Max. :180.00 Max. :38.00 Max. :2.0000 Max. :35.00   
## NA's :127 NA's :127 NA's :127 NA's :127   
## protein\_g primary\_ingredient total\_fat\_per\_serv  
## Min. :0.000 chocolate:193 Min. :0.00000   
## 1st Qu.:0.000 dextrose : 9 1st Qu.:0.00000   
## Median :1.000 peanuts : 6 Median :0.02703   
## Mean :1.407 sugar :484 Mean :0.09263   
## 3rd Qu.:3.000 syrup :160 3rd Qu.:0.20455   
## Max. :5.000 NA's :127 Max. :0.32353   
## NA's :127 NA's :127   
## saturated\_fat\_per\_serv cholesterol\_per\_serv sodium\_per\_serv   
## Min. :0.00000 Min. :0.00000 Min. :0.0000   
## 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:0.2500   
## Median :0.01351 Median :0.00000 Median :0.4881   
## Mean :0.05664 Mean :0.03108 Mean :0.8735   
## 3rd Qu.:0.11905 3rd Qu.:0.00000 3rd Qu.:1.1250   
## Max. :0.20000 Max. :0.32051 Max. :4.5000   
## NA's :127 NA's :127 NA's :127   
## total\_carb\_per\_serv dietary\_fiber\_per\_serv sugars\_per\_serv   
## Min. :0.5294 Min. :0.00000 Min. :0.4186   
## 1st Qu.:0.7059 1st Qu.:0.00000 1st Qu.:0.5250   
## Median :0.8250 Median :0.00000 Median :0.6000   
## Mean :0.7800 Mean :0.00838 Mean :0.6097   
## 3rd Qu.:0.8580 3rd Qu.:0.02273 3rd Qu.:0.6750   
## Max. :1.0000 Max. :0.04651 Max. :0.9211   
## NA's :127 NA's :127 NA's :127   
## protein\_per\_serv   
## Min. :0.00000   
## 1st Qu.:0.00000   
## Median :0.02500   
## Mean :0.03574   
## 3rd Qu.:0.06977   
## Max. :0.11765   
## NA's :127

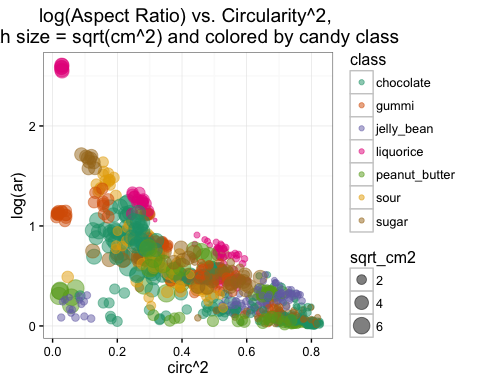
Using summary, it seems that our merge was successful. However, you will notice that there are some NAs. It is important we deal with these for subsequent analyses, because some methods, like PCA, can't handle missing data this way. Let's get rid of the NAs using na.omit

nonas\_mdata <- na.omit(mdata)  
summary(nonas\_mdata)

## id label candy\_no area   
## id\_16 : 50 ID\_10\_01.jpg: 1 Min. : 1.00 Min. : 5721   
## id\_14 : 30 ID\_10\_02.jpg: 1 1st Qu.: 5.00 1st Qu.: 10328   
## id\_52 : 30 ID\_10\_03.jpg: 1 Median : 9.00 Median : 19701   
## id\_64 : 28 ID\_10\_04.jpg: 1 Mean :10.88 Mean : 46853   
## id\_20 : 25 ID\_10\_05.jpg: 1 3rd Qu.:15.00 3rd Qu.: 56382   
## id\_61 : 25 ID\_10\_06.jpg: 1 Max. :50.00 Max. :356531   
## (Other):664 (Other) :846   
## cm2 circ ar round   
## Min. : 0.8363 Min. :0.1090 Min. : 1.003 Min. :0.0740   
## 1st Qu.: 1.5098 1st Qu.:0.5490 1st Qu.: 1.160 1st Qu.:0.4925   
## Median : 2.8799 Median :0.7170 Median : 1.528 Median :0.6540   
## Mean : 6.8489 Mean :0.6725 Mean : 1.819 Mean :0.6614   
## 3rd Qu.: 8.2418 3rd Qu.:0.8293 3rd Qu.: 2.030 3rd Qu.:0.8618   
## Max. :52.1172 Max. :0.9090 Max. :13.587 Max. :0.9970   
##   
## solid sqrt\_cm2 name   
## Min. :0.7120 Min. :0.9145 reeses\_pieces : 50   
## 1st Qu.:0.9210 1st Qu.:1.2287 good\_and\_plenty : 30   
## Median :0.9570 Median :1.6970 original\_skittles : 30   
## Mean :0.9388 Mean :2.2666 tropical\_wild\_berry\_skittles: 28   
## 3rd Qu.:0.9720 3rd Qu.:2.8709 mms\_original : 25   
## Max. :0.9880 Max. :7.2192 sweet\_and\_sour\_starburst : 25   
## (Other) :664   
## company class serving\_size\_g calories   
## hershey :215 chocolate :217 Min. : 7.00 Min. : 25.0   
## wrigley :167 gummi :123 1st Qu.:40.00 1st Qu.:140.0   
## mars : 99 jelly\_bean :109 Median :40.00 Median :150.0   
## haribo : 62 liquorice : 83 Mean :39.22 Mean :158.7   
## just\_born: 60 peanut\_butter:101 3rd Qu.:41.00 3rd Qu.:190.0   
## nestle : 42 sour : 84 Max. :45.00 Max. :220.0   
## (Other) :207 sugar :135   
## calories\_fat total\_fat\_g saturated\_fat\_g cholesterol\_mg   
## Min. : 0.00 Min. : 0.000 Min. :0.000 Min. : 0.000   
## 1st Qu.: 0.00 1st Qu.: 0.000 1st Qu.:0.000 1st Qu.: 0.000   
## Median : 10.00 Median : 1.000 Median :0.500 Median : 0.000   
## Mean : 31.95 Mean : 3.639 Mean :2.228 Mean : 1.206   
## 3rd Qu.: 70.00 3rd Qu.: 8.000 3rd Qu.:5.000 3rd Qu.: 0.000   
## Max. :110.00 Max. :13.000 Max. :8.000 Max. :10.000   
##   
## sodium\_mg total\_carb\_g dietary\_fiber\_g sugars\_g   
## Min. : 0.00 Min. : 6.00 Min. :0.0000 Min. : 6.00   
## 1st Qu.: 10.00 1st Qu.:26.00 1st Qu.:0.0000 1st Qu.:21.00   
## Median : 20.00 Median :32.00 Median :0.0000 Median :24.00   
## Mean : 34.19 Mean :30.59 Mean :0.3427 Mean :23.83   
## 3rd Qu.: 45.00 3rd Qu.:34.00 3rd Qu.:1.0000 3rd Qu.:27.00   
## Max. :180.00 Max. :38.00 Max. :2.0000 Max. :35.00   
##   
## protein\_g primary\_ingredient total\_fat\_per\_serv  
## Min. :0.000 chocolate:193 Min. :0.00000   
## 1st Qu.:0.000 dextrose : 9 1st Qu.:0.00000   
## Median :1.000 peanuts : 6 Median :0.02703   
## Mean :1.407 sugar :484 Mean :0.09263   
## 3rd Qu.:3.000 syrup :160 3rd Qu.:0.20455   
## Max. :5.000 Max. :0.32353   
##   
## saturated\_fat\_per\_serv cholesterol\_per\_serv sodium\_per\_serv   
## Min. :0.00000 Min. :0.00000 Min. :0.0000   
## 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:0.2500   
## Median :0.01351 Median :0.00000 Median :0.4881   
## Mean :0.05664 Mean :0.03108 Mean :0.8735   
## 3rd Qu.:0.11905 3rd Qu.:0.00000 3rd Qu.:1.1250   
## Max. :0.20000 Max. :0.32051 Max. :4.5000   
##   
## total\_carb\_per\_serv dietary\_fiber\_per\_serv sugars\_per\_serv   
## Min. :0.5294 Min. :0.000000 Min. :0.4186   
## 1st Qu.:0.7059 1st Qu.:0.000000 1st Qu.:0.5250   
## Median :0.8250 Median :0.000000 Median :0.6000   
## Mean :0.7800 Mean :0.008381 Mean :0.6097   
## 3rd Qu.:0.8580 3rd Qu.:0.022727 3rd Qu.:0.6750   
## Max. :1.0000 Max. :0.046512 Max. :0.9211   
##   
## protein\_per\_serv   
## Min. :0.00000   
## 1st Qu.:0.00000   
## Median :0.02500   
## Mean :0.03574   
## 3rd Qu.:0.06977   
## Max. :0.11765   
##

Great! All the NAs are gone. Let's look at our graph of aspect ratio, circularity, and sqrt\_cm2 again, this time by candy class

p <- ggplot(nonas\_mdata, aes(x=circ^2, y=log(ar), size=sqrt\_cm2, colour=class))  
p + geom\_point(alpha=0.5) + scale\_colour\_brewer(type="qual", palette=2) + theme\_bw() + ggtitle(label="log(Aspect Ratio) vs. Circularity^2, \nwith size = sqrt(cm^2) and colored by candy class")



If you want to save your graph, then ggsave("all\_candies.jpg")

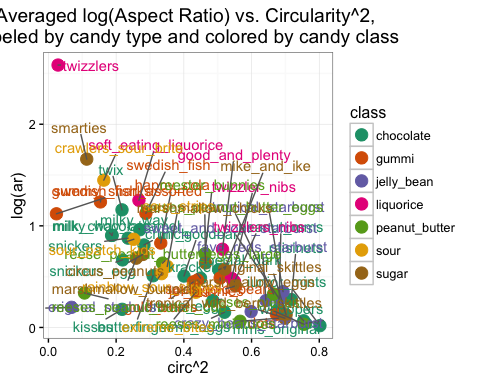
To get a feel for the shapes of the candies, let's take averages by candy type and replot this graph with labels

averaged <- aggregate(nonas\_mdata[c(4:10,11,13,14:33)], by=list(nonas\_mdata$name, nonas\_mdata$class), FUN=mean)

Now let's plot the averaged shape descriptor values and label by candy type and class

p <- ggplot(averaged, aes(x=circ^2, y=log(ar), colour=Group.2))  
p + geom\_point(alpha=1, size=4) + scale\_colour\_brewer(type="qual", palette=2) + geom\_text\_repel(data=averaged, aes(x=circ^2, y=log(ar), label=Group.1)) + scale\_colour\_brewer(type="qual", palette=2, guide=guide\_legend(title="class")) + theme\_bw() + ggtitle(label="Averaged log(Aspect Ratio) vs. Circularity^2, \nlabeled by candy type and colored by candy class")

## Scale for 'colour' is already present. Adding another scale for  
## 'colour', which will replace the existing scale.



If you want to save your graph, then ggsave("averaged\_candies.jpg")

# Final project

Previously, you should have isolated the RGB values per each candy piece. Your final project is to consider all the candy data as a whole. The three main datasets are:

1. Nutrition label information and candy class for each of the 75 candy types
2. Shape descriptor information and area for each of ~980 individual candy pieces
3. RGB color information for ~980 individual candy pieces (you should be in possession of this data)

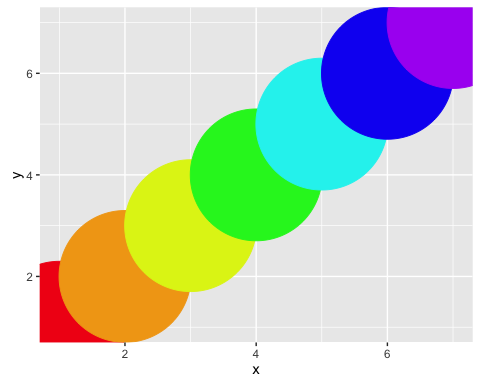
You should first merge the three datasets together.

Then, considering data formating (NAs!) and scaling and centering of data as we discussed, use PCA and hierarchical clustering and provide commentary and notes on the results to discern patterns of relatedness of the candies. Please save both your code and commentary for the final project.

Provide ample data visualization to convey your results and make them understandable to others. Be sure to title and label your graphs too.

Additionally: because you have color information, in all your graphs (PCAs, scatterplots of individual variables) consider coloring your datapoints by the actual color of the individual candies. To do this, consider the following example data:

sample\_colors <- read.table("./sample\_colors.txt", header=TRUE)  
  
p <- ggplot(sample\_colors, aes(x,y, colour=rgb(r,g,b, maxColorValue=255)))  
p + geom\_point(size=46) + scale\_colour\_identity()



Coloring your data by actual candy piece rgb values should make for a striking graph!

Have fun, apply what you've learned, and analyze the multivariate data you worked so hard to collect!!! Email Dan your scripts, including commentary, no later than 5pm pon Thursday. Provide lots of analyses and data visualization, as well as explanation, for what you think is going on with your data!!!