1) Entropy

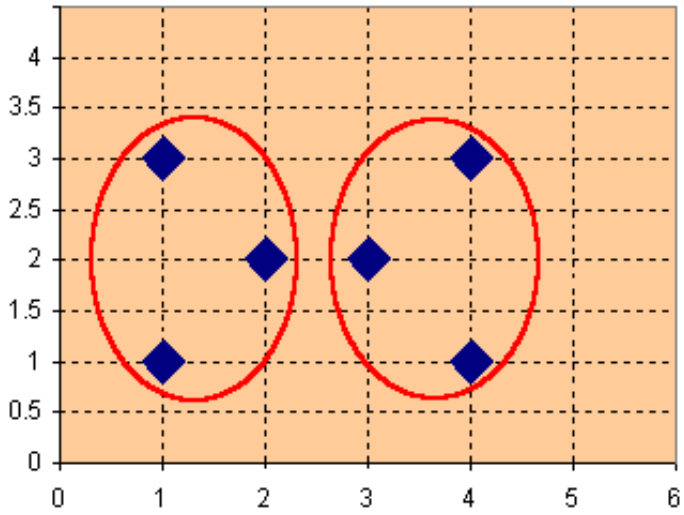
Let us create a simple example: two clusters, we evaluate entropy on a single point.

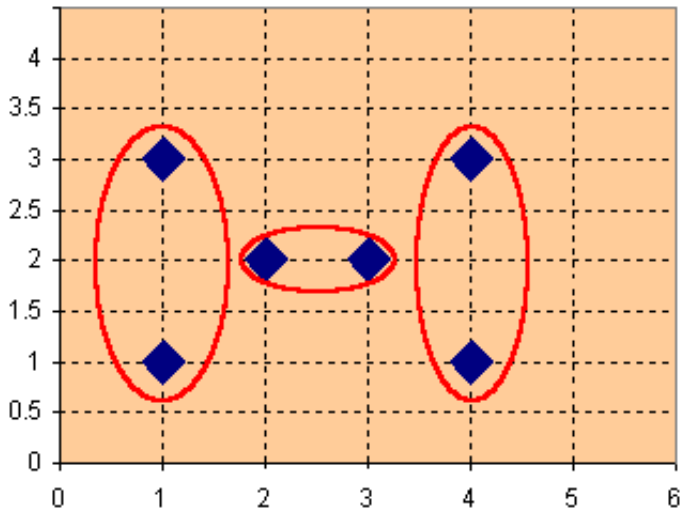
```
# let us simulate a range probabilities that a member of cluster i belongs to class j
p11 = seq(0,1,.01)
# let us suppose that all the elements that are not of class 1 belong to class 2
p12 = 1-p11
p = cbind(p11,p12)

# calculate entropy
e = -rowSums(p * log(p))
# note we might have some NaNs if pij==0 (we have 0*-Inf in this case)
e[is.nan(e)] = 0

# plot entropy as a function of the probability p11 (or 1-p12)
plot(p11,e)
```

---

2) WSS and BSS - Exercises

- Given the results of the two clusterings below, obtained running the same algorithm with K=2 and K=3 clusters, calculate WSS and BSS and tell which result is better and why.





Case K = 2 clusters
Positions of centroids: c1=(4/3,2), c2=(11/3,2)
Mean of all points = (2.5,2)

WSS = [(1+1/9) + (1+1/9) + (4/9)] * 2 = 24/9 * 2 = 48/9 = 5.33

BSS = [3(1/2 + 2/3)^2]*2 = 6[(3/6 + 4/6)^2] = 6 * 49/36 = 49/6 = 8.16

WSS + BSS = 5.33 + 8.16 = 13.5

Also check with R:

```
x = c(1, 1, 2, 3, 4, 4)
y = c(1, 3, 2, 2, 1, 3)
data = cbind(x,y)
plot(data)
C1 = colMeans(data[1:3,])
C2 = colMeans(data[4:6,])
WSS = sum((t(data[1:3,]) - C1)^2) + sum((t(data[4:6,]) - C2)^2)

C = colMeans(data)
BSS = 3*sum((C1-C)^2) + 3*sum((C2-C)^2)

WSS
BSS
WSS+BSS
```

Case K = 3 clusters
Positions of centroids: c1=(1,2), c2=(2.5,2), c3=(4,2)
Mean of all points = (2.5,2) (this always remains the same of course)

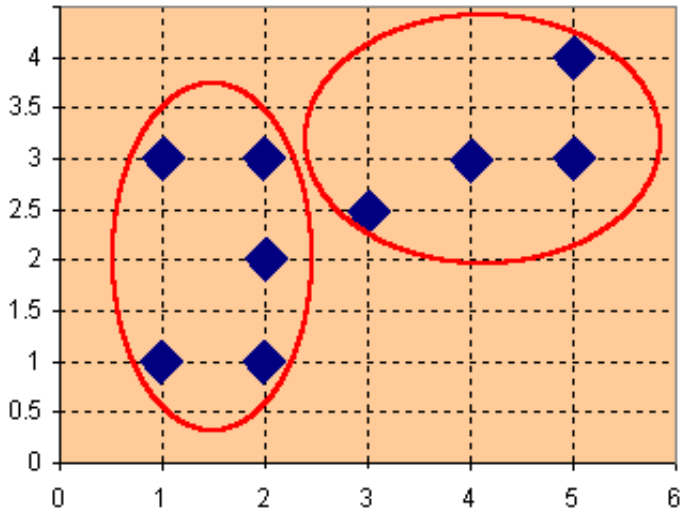
WSS = (3-2)^2 + (1-2)^2 + (2-2.5)^2 + (3-2.5)^2 + (3-2)^2 + (1-2)^2
= 1 + 1 + 0.25 + 0.25 + 1 + 1 = 4.5

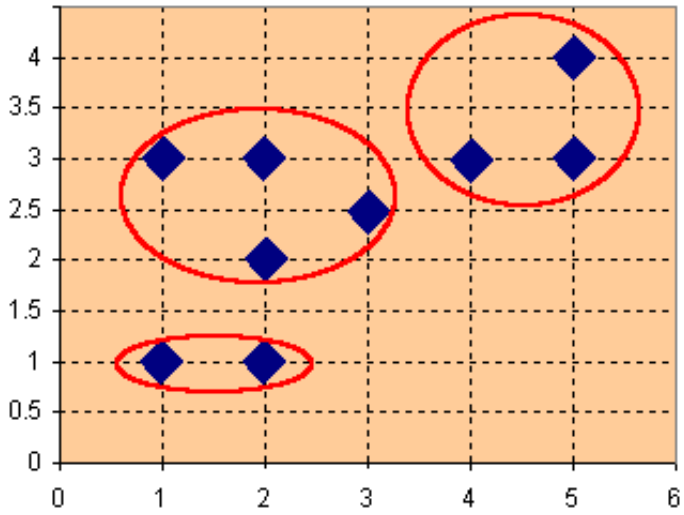
BSS = 2(1-2.5)^2 + 2(0)^2 + 2(4-2.5)^2 = 2*2.25 + 2*2.25 = 9

WSS + BSS = 4.5 + 9 = 13.5
(NOTE: the sum WSS+BSS is CONSTANT!)

The case K=3 is better as it has both a higher cohesion (WSS is lower) and a higher separation (BSS is higher).

- Given the results of the two clusterings below, obtained running the same algorithm with K=2 and K=3 clusters, calculate WSS and BSS and tell which result is better and why.





Case K = 2 clusters
Positions of centroids: c1=(1.6,2), c2=(4.25,3.125)
Mean of all points = (2.78,2.5)

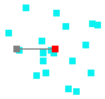
WSS = 2*(1,2)^2+3*(0,2)^2+4+(1,25)^2+0,25^2+2*(0,75)^2+(2,5-3,125)^2+2*(0,125)^2+(4-3,125)^2 = 10.9375
BSS = 5*((2,2-2,78)^2+(2-2,5)^2)+4*((4,25-2,78)^2+(3,125-2,5)^2) = 13.1381
WSS + BSS = 24.0756

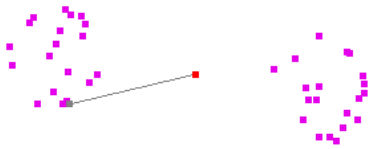
Case K = 3 clusters
Positions of centroids: c1=(1.5,1), c2=(2,2.625), c3=(4.66,3.33)
Mean of all points = (2.78,2.5)

BSS = 2*((1,5-2,78)^2+(1-2,5)^2)+4*((2-2,78)^2+(2,625-2,5)^2)+3*((4,66-2,78)^2+(3,33-2,5)^2) = 22.9428
NOTE: WSS can be calculated more quickly by relying on the fact that the sum WSS+BSS is constant
WSS = 24.0756 - BSS = 24.0756 - 22.9428 = 1.1328

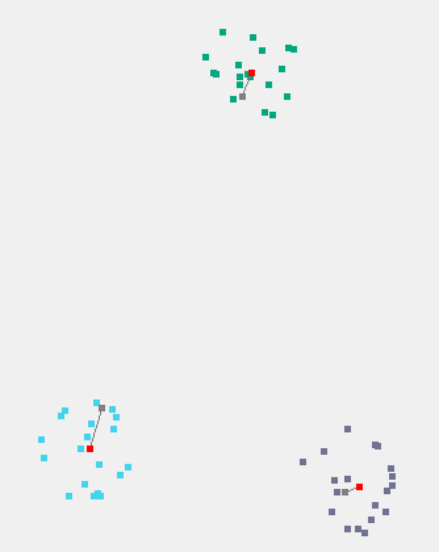
The case K=3 is better as it has both a higher cohesion (WSS is lower) and a higher separation (BSS is higher).

As a sidenote, take into account that WSS always decreases with increasing K if we always take a "reasonable" split like the ones you saw before. If you have a really bad clustering (see e.g. below), you might not have the same decrease you expect from the reasonable case.

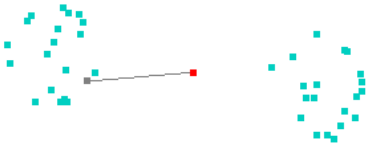




Example with K = 2



Example with K=3 (good)



Example with K=3 (bad)

WSS and BSS - notes on "main centroid" calculation

The "main centroid" is defined as the average position of all the points. Warning: one might think that it could be also calculated more quickly as the mean of the centroids positions, however results are usually not the same:

Two clusters: 1, 3 and 5, 7
centroids: 2, 6
average position of centroids: 4 (= 2+6 / 2)
main centroid (average position of all points): 4 (= 1+3+5+7 / 4)
=> in this case they are the same

Two clusters: 1,3 and 5,7,9
centroids: 2, 7
average position of centroids: 4.5 (= 2+7 / 2)
main centroid (average position of all points): 5 (= 1+3+5+7+9 / 5 = 5)
=> in this case they differ!!!
