

Cluster validation & dendrogram adjustment

Hierarchical clustering

- Distance calculation

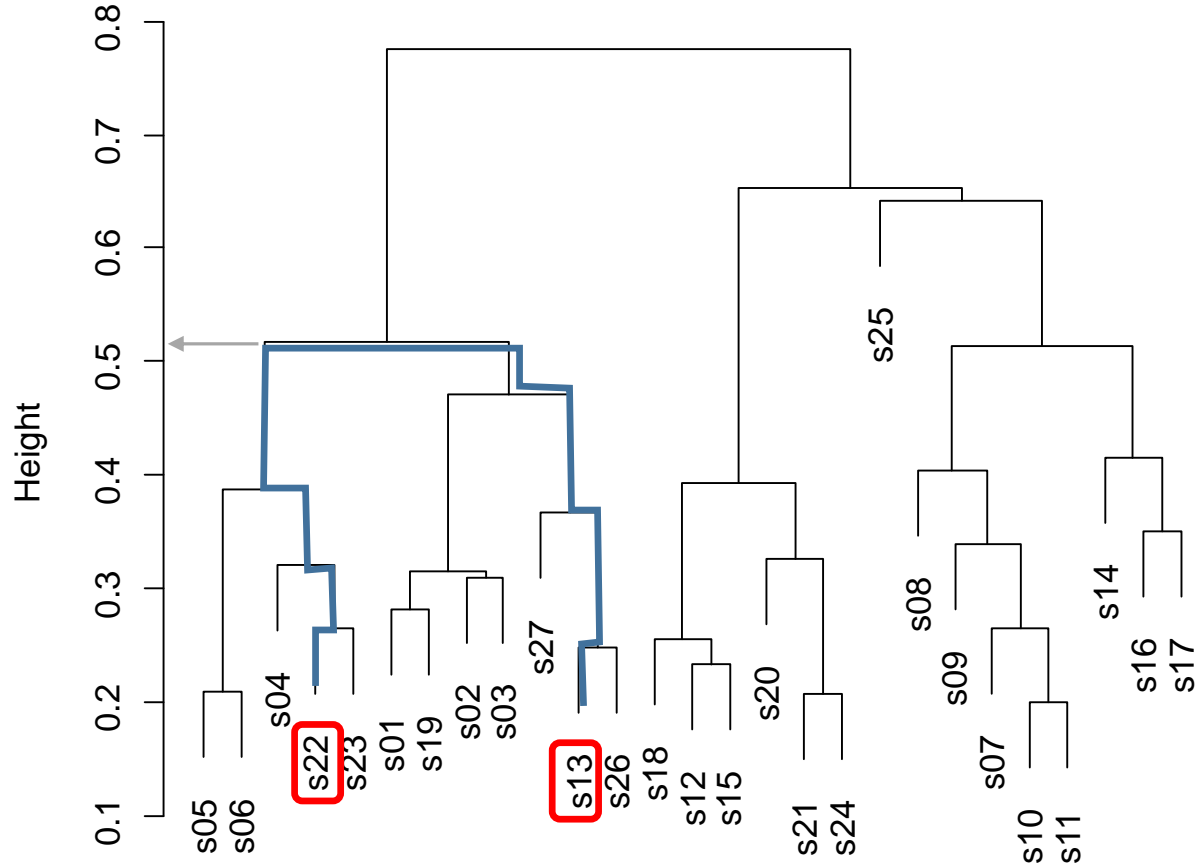
```
distance <- dist (spe, method = )  
vegdist (spe)  
designdist (spe)
```



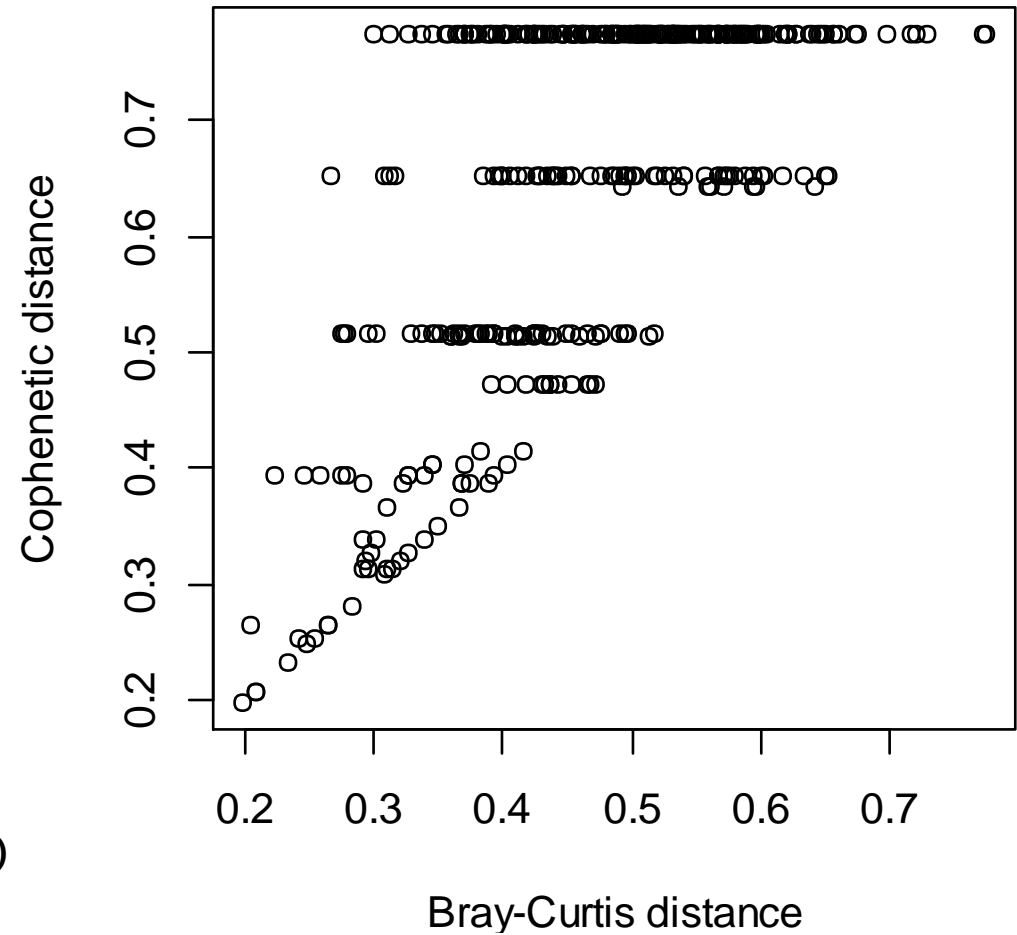
- Clustering

```
dendrogram <- hclust (distance, method = )
```

Cophenetic distance

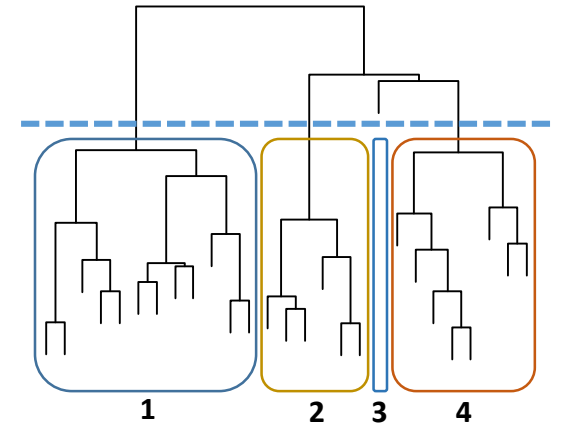


```
cophenetic_distance <- cophenetic (dendrogram)
```



Extract group memberships

- cut the dendrogram (tree) on a certain level



```
groups <- cutree (dendrogram, k = desired_number_of_groups, h = or_height)
```

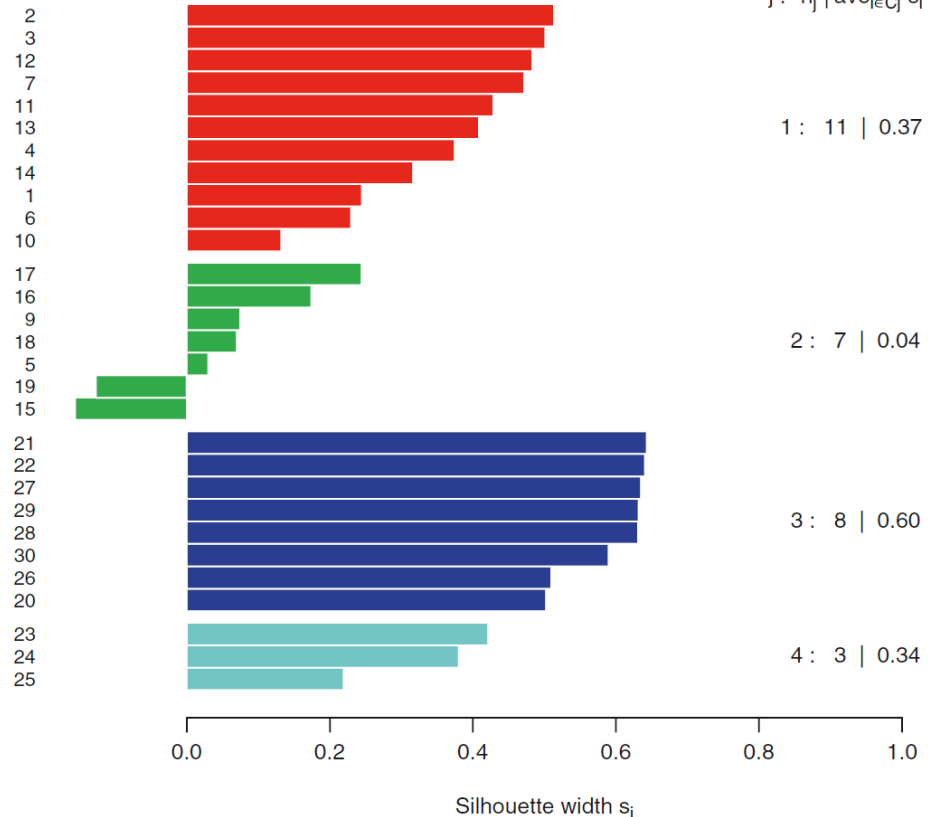
```
> s01 s02 s03 s04 s05 s06 s07 s08 s09 s10 s11 s12 s13 s14 s15 s16 s17  
>  1  1  1  2  2  2  3  3  3  3  3  4  1  3  4  3  3  
> s18 s19 s20 s21 s22 s23 s24 s25 s26 s27  
>  4  1  4  4  2  2  4  5  1  1
```

Silhouette plot

Silhouette plot - Chord - Ward

n = 29

4 clusters C_j
 $j : n_j \mid \text{ave}_{i \in C_j} s_i$



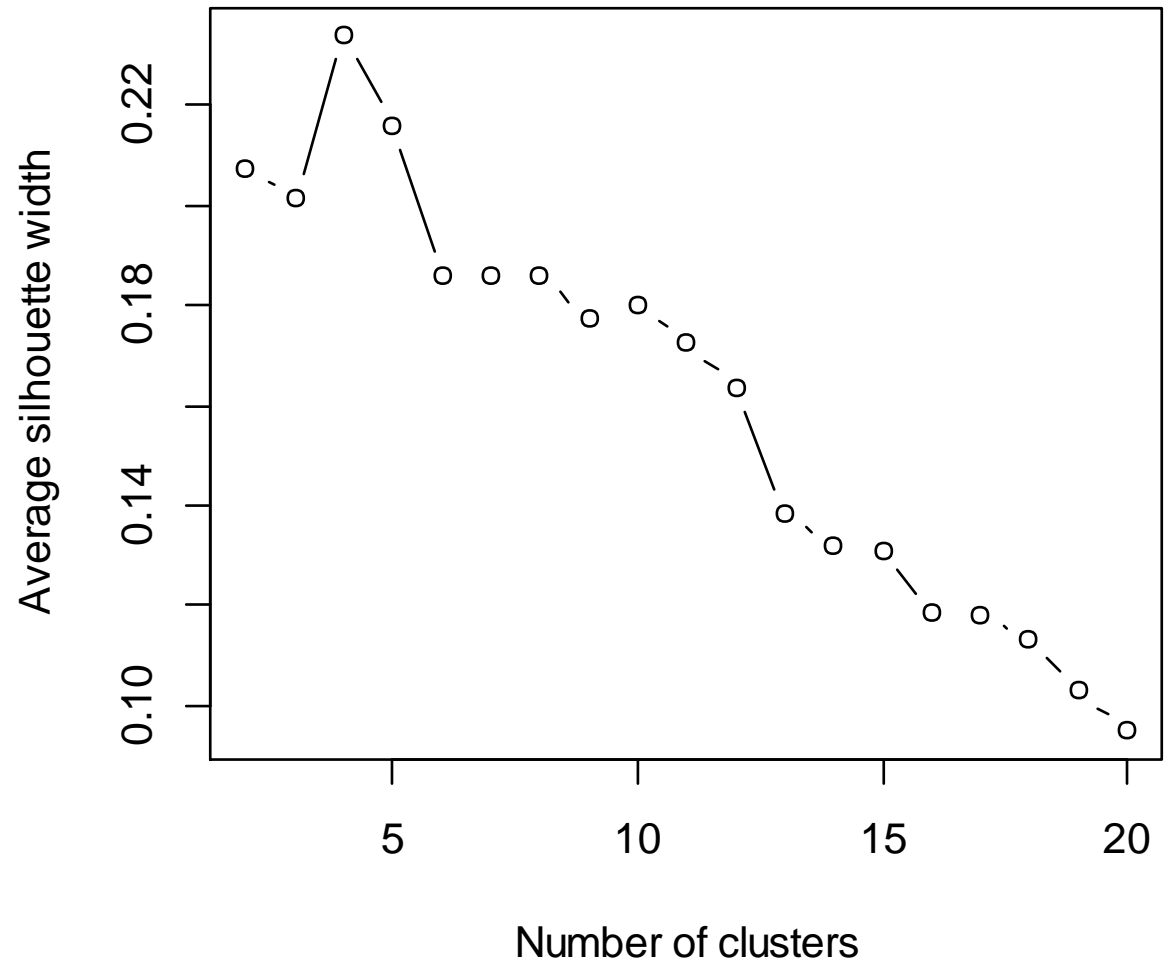
Average silhouette width : 0.35

```
sil <- cluster::silhouette(groups, distance)
```

```
> cluster neighbor sil_width  
> [1,] 1 2 -0.043640504  
> [2,] 1 3 0.018670085  
> [3,] 1 2 0.060293140  
> [4,] 2 1 0.224429327  
> [5,] 2 1 0.230826816  
> [6,] 2 1 0.138005590  
> ...
```

```
plot(sil)
```

Average silhouette



```
plot(2:20,  
     sapply(2:20, function(x) mean(silhouette(cutree(c1_upgma, x), bray)[, "sil_width"])),  
     type = "b")
```

Reordering of clusters

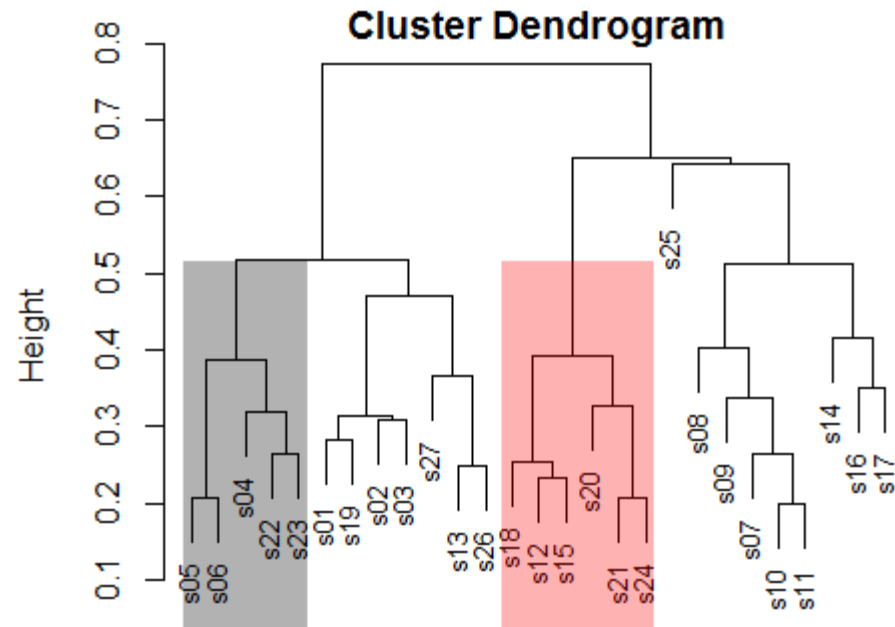
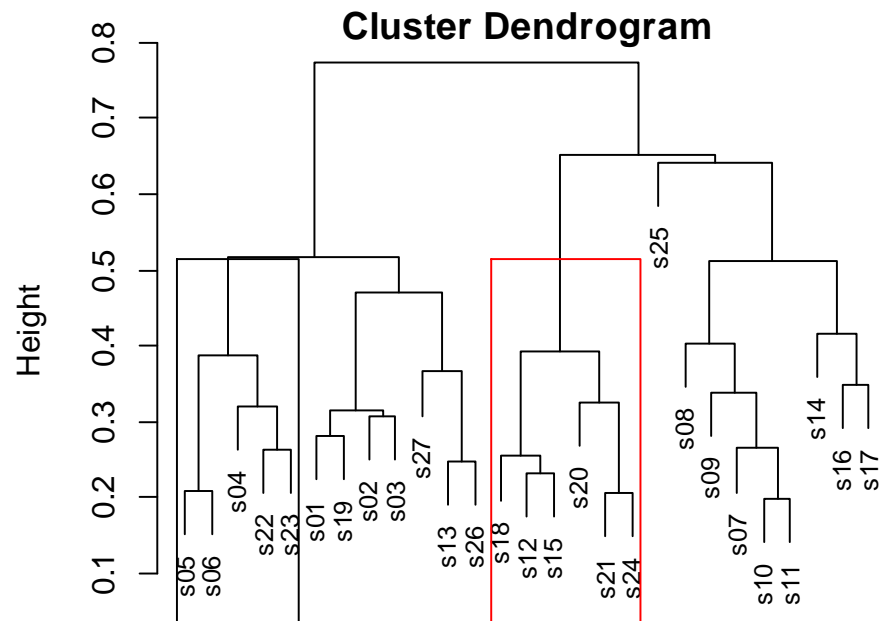
```
dendrogram_reordered <- reorder (dendrogram, ordering_var)
```

Drawing rectangles

```
plot(dendrogram, cex = 0.8, xlab = "")
```

```
rect.hclust(dendrogram, 5, border = 1:2, which = c(1,3))
```

```
my.rect(dendrogram, 5, which= c(1,3), col= adjustcolor(1:2, alpha.f = 0.3),  
border= NA)
```



Heatmap of ordered species data

```
heatmap()
```

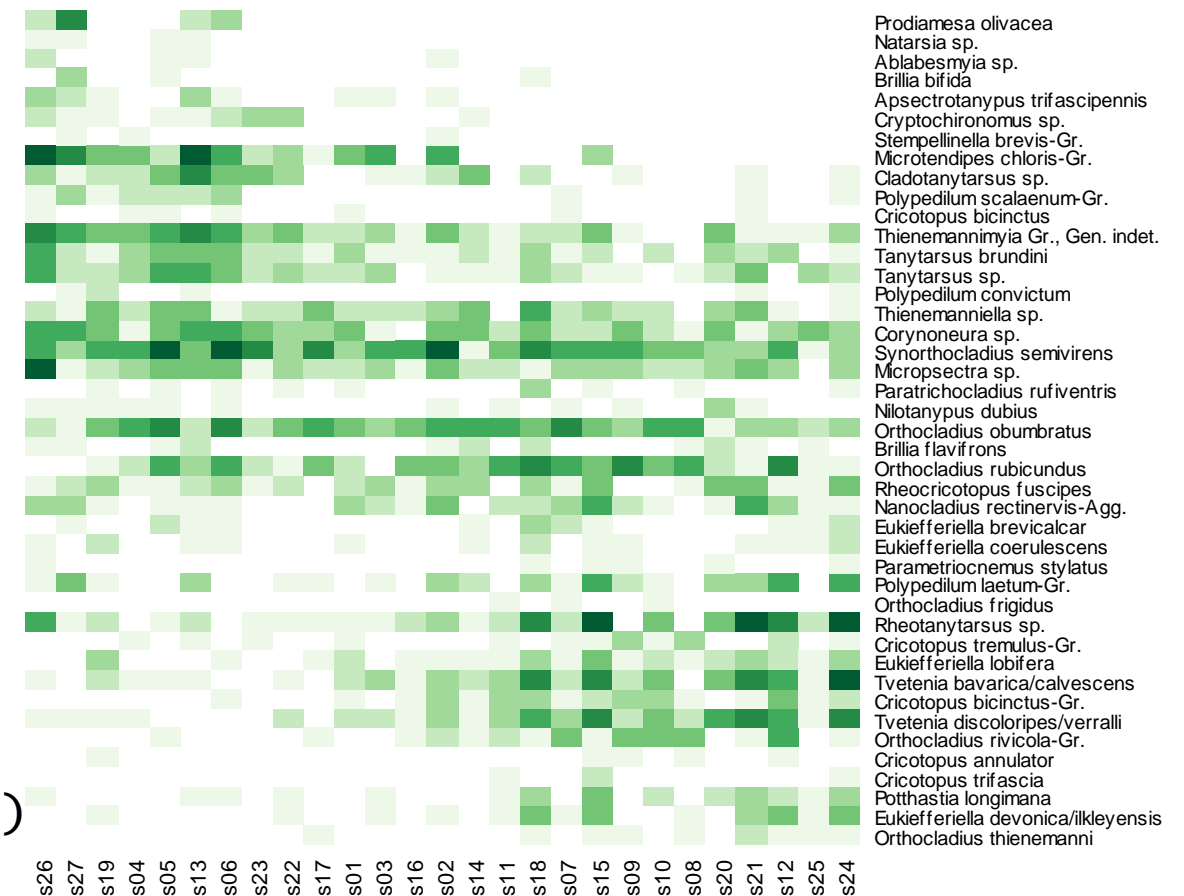
```
heatmap(t(spe[vm$sites, vm$species]),  
        Rowv= NA, Colv= NA,  
        col= vector_of_colours,  
        breaks = optional_breaks)
```

```
tabasco(log1p(spe),  
        use = ordering_var,  
        col= rev(vector_of_colours))
```

vector_of_colours e.g.:

```
brewer.pal(7, "Greens")
```

```
c("grey80", brewer.pal(7, "Greens"))
```



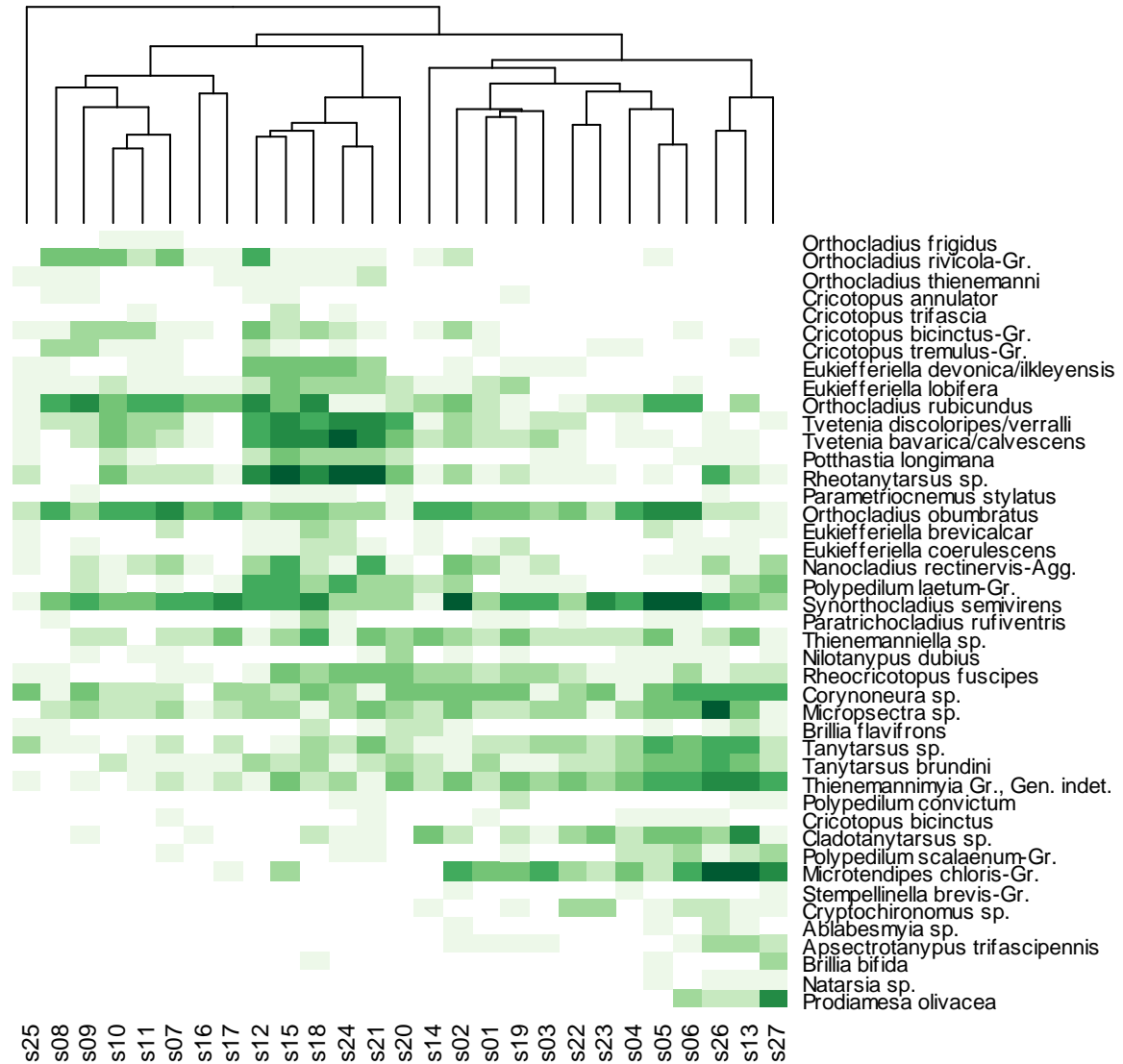
Heatmap with a dendrogram

heatmap():

```
colv = as.dendrogram(dendrogram)
```

tabasco():

```
use = dendrogram
```



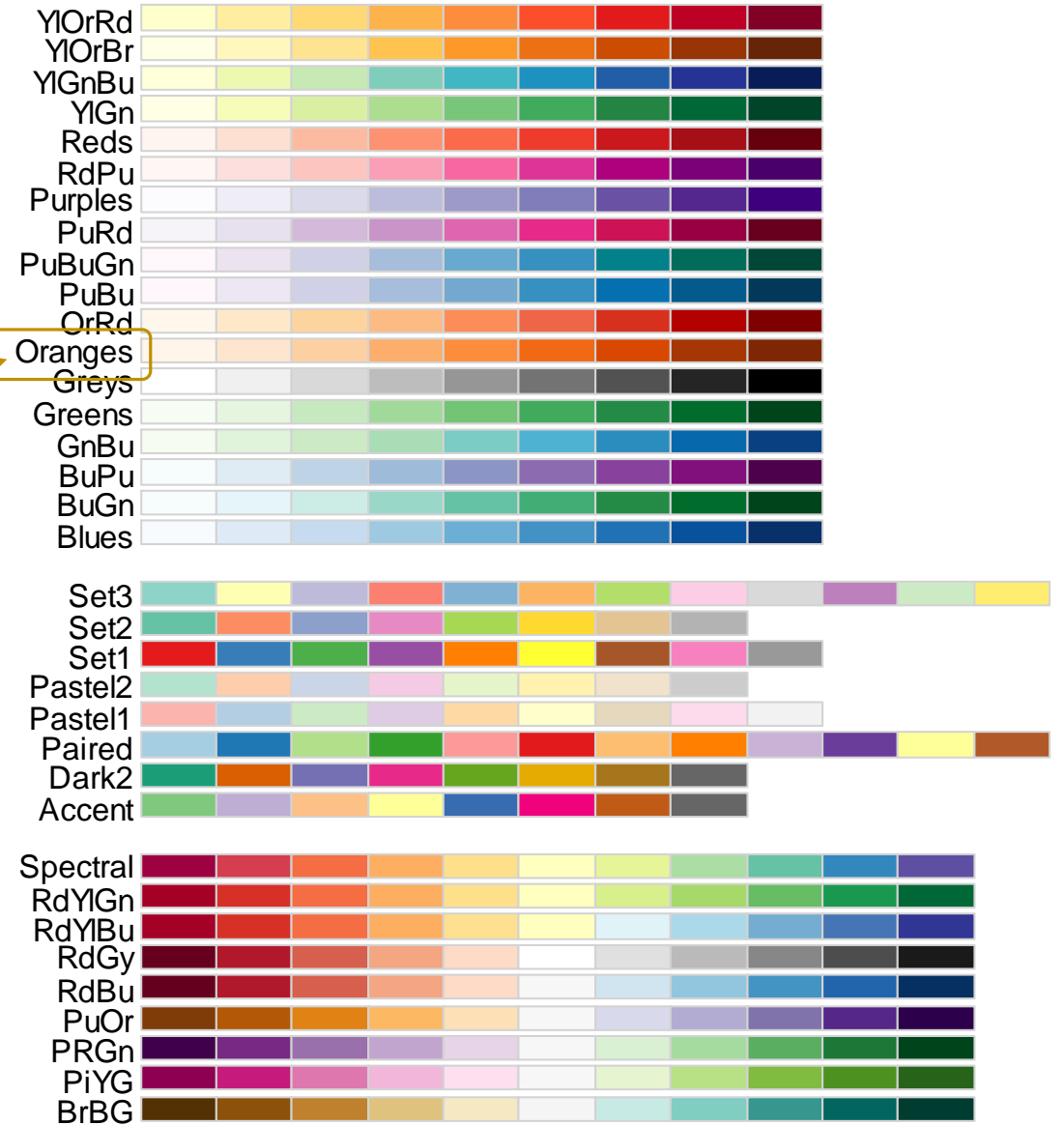
Colour gradients

- package **RColorBrewer**

```
display.brewer.all()
```

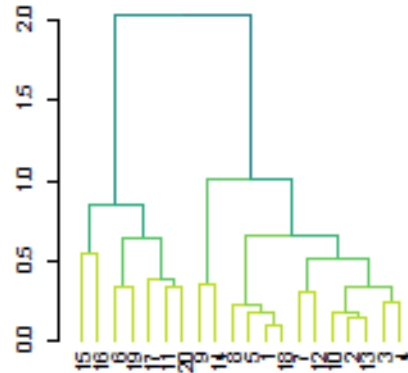
```
brewer.pal(7, "Oranges")
```

```
[1] "#FEEDDE" "#FDD0A2" "#FDAE6B" "#FD8D3C"  
     "#F16913" "#D94801" "#8C2D04")
```

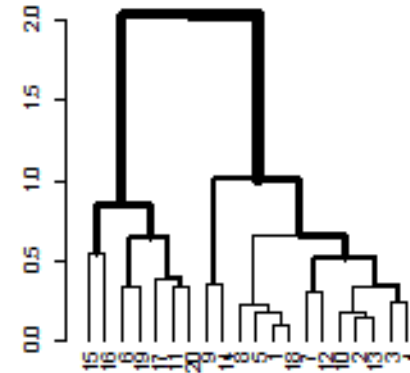


dendextend package

Coloring branches



Emphasizing line-width



Emphasizing color and line-width

