

# Estadística Multivariante - PEC 2

Alejandro Keymer

## Ejercicio 1 (70 pt.)

(a)

Después de reducir la base de datos original a las  $p = 13$  especies destacadas, nos planteamos elegir una distancia adecuada para el análisis de conglomerados. Con este tipo de datos se suele utilizar la disimilaridad de Bray-Curtis con la siguiente fórmula según la Wikipedia:

$$d_{BC}^{(1)}(i, j) = 1 - \frac{2C_{ij}}{S_i + S_j}$$

donde  $C_{ij}$  es la suma del menor valor para cada una de las especies en común de las dos muestras.  $S_i$  y  $S_j$  son el total de especímenes en cada una de las muestras. También se puede calcular con la fórmula que utiliza la función `vegdist()` del paquete `vegan`

$$d_{BC}^{(2)}(i, j) = \frac{\sum_{k=1}^p |x_{ik} - x_{jk}|}{\sum_{k=1}^p (x_{ik} + x_{jk})}$$

Calcular la disimilaridad de Bray-Curtis entre las dos primeras muestras de la base de datos con ambas fórmulas (sin utilizar la función `vegdist()`). En la primera fórmula nos puede ayudar la función `pmin()` de R. Calcular a continuación la matriz de disimilaridades completa con la función `vegdist()` y comprobamos que el valor hallado para las dos primeras muestras coincide:

En primer lugar se calculan los datos y se crean nombres de fila con las coordenadas de cada muestra:

```
species <- c('carcar', 'corflo', 'faggra',  
            'ileopa', 'liqsty', 'maggra',  
            'nyssyl', 'ostvir', 'oxyarb',  
            'pingla', 'quenig', 'quemmic', 'symtin')  
  
df <-  
  read.table("wood.dat", header=T) %>%  
  select(x, y, species) %>%  
  unite(rowname, x, y, sep = "-") %>%  
  column_to_rownames()
```

Con la base hacemos el cálculo *manual* con las fórmulas descritas utilizando las dos primeras muestras:

```
# formula 1  
(d_bc_1 <- 1 - (2 * sum(pmin(df[1,], df[2,]))) / (sum(df[1,]) + sum(df[2,])))
```

```
## [1] 0.3181818
```

```
# formula 2  
(d_bc_2 <- sum(abs(df[1,] - df[2,])) / sum(df[1,] + df[2,]))
```

```
## [1] 0.3181818
```

A continuación se realiza el cálculo de la matriz de disimilaridad utilizando la función `vegdist` que, por defecto, utiliza la distancia de Bray-Curtis. Finalmente revisamos que la primera distancia, entre las dos primeras muestras, coincide con las calculadas manualmente.

```
df_dist <- vegdist(df)
df_dist[1]
```

```
## [1] 0.3181818
```

Se puede ver que las tres soluciones dan el mismo valor para la distancia entre las primeras dos muestras.

(b)

Sin embargo, la disimilaridad de Bray-Curtis no verifica la desigualdad triangular de forma que no es una distancia. Comprobar este hecho, tal vez con algún contra ejemplo entre dos muestras o con alguna función de R que podamos hallar. Para corregir este problema podemos transformar la disimilaridad de Bray-Curtis calculando la raíz cuadrada de sus valores. Comprobar que dicha raíz cuadrada verifica la desigualdad triangular.

Para verificar la desigualdad triangular se puede utilizar la función `tri.ineq` de la librería `fossil`. Con esta función podemos corroborar que la distancia de Bray-Curtis calculada **no** verifica la desigualdad triangular, pero la raíz cuadrada de las distancias, si lo hacen. Al examinar la función en sí, es posible modificar el procedimiento por el que se verifica la desigualdad triangular, y encontrar un contra ejemplo específico. La fórmula busca si hay casos en que se cumpla que el “cateto” mas grande, sea mayor a la suma de los otros dos “catetos”, lo que implica que no es un triángulo.

```
tri.ineq(df_dist)
```

```
## [1] FALSE
```

*# función modificada para encontrar contraejemplo. Devuelve el primer contraejemplo que encuentra.*

```
mat <- as.matrix(df_dist)
n <- dim(mat)[1]
for (i in 1:(n - 2)) {
  for (j in (i + 1):(n - 1)) {
    for (k in (j + 1):n) {
      sds <- c(mat[j, i], mat[k, i], mat[k, j])
      lng <- max(sds)
      if (lng > (sum(sds) - lng)){
        return()}
      }
    }
  }
}
```

*# no se satisface para las muestras:*

```
rownames(mat)[j]
```

```
## [1] "2-7"
```

```
rownames(mat)[i]
```

```
## [1] "1-2"
```

```
rownames(mat)[k]
```

```
## [1] "5-2"
```

Para la distancia al cuadrado si se cumple

```
# cuadrado de las distancias
sq_dist <- sqrt(df_dist)
tri.ineq(sq_dist)
```

```
## [1] TRUE
```

---

(c)

Muchos procedimientos de Análisis multivariante requieren que la distancia utilizada sea euclídea. Comprobar que la raíz cuadrada de la disimilaridad de Bray-Curtis es una distancia euclídea.

Es posible crear una función con las fórmulas del apartado 8.2 del libro de *C.M. Cuadras*.

```
is_euclidean <- function(diss){
  mat <- as.matrix(diss)
  n <- dim(mat)[1]
  A <- mat * -(1/2)
  H <- diag(n) - (1/n) * (rep(1,n) %*% t(rep(1,n)))
  B <- H %*% A %*% H

  # prueba
  eig <- eigen(B)
  all(eig$values >= 0)
}
```

```
# probamos con la raíz cuadrada de la distancia
is_euclidean(sq_dist)
```

```
## [1] TRUE
```

---

(d)

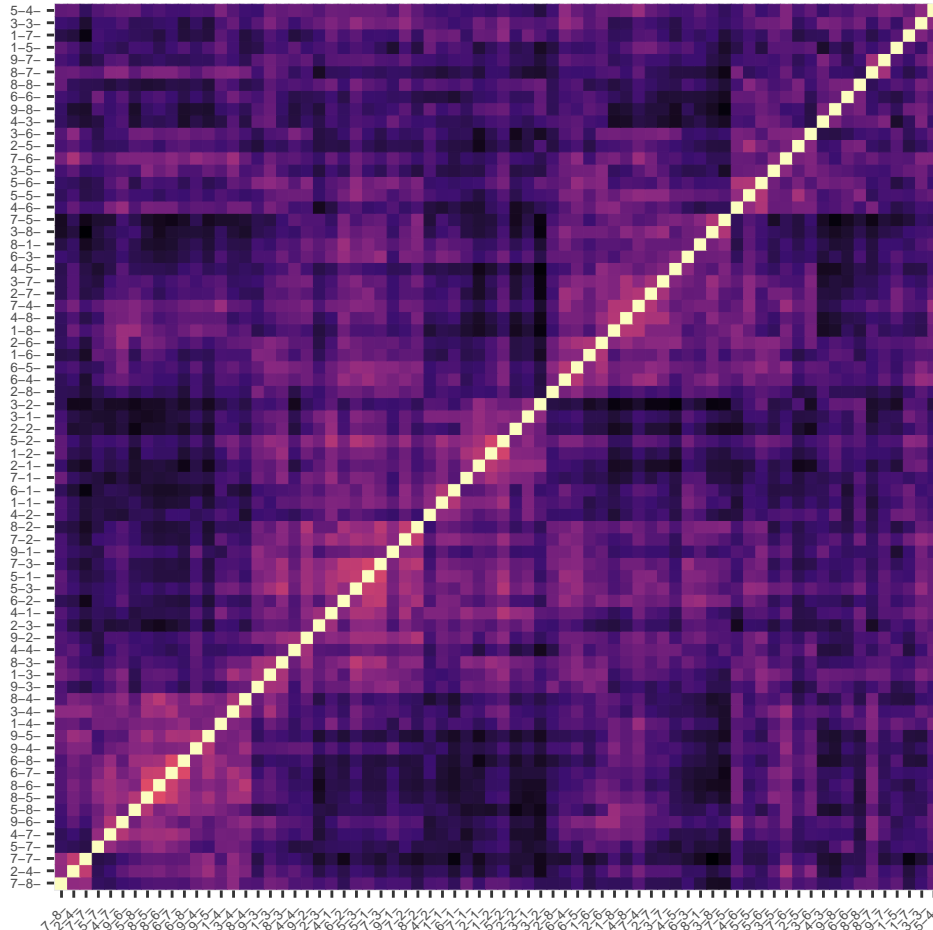
Evaluar la tendencia al agrupamiento con el estadístico de Hopkins que podéis calcular con la función `get_clust_tendency()` del paquete `factoextra`. La función `fviz_dist()` del mismo paquete permite realizar una imagen de la disimilaridad propuesta en el apartado anterior (*ordered dissimilarity image, ODI*) para evaluar gráficamente la tendencia al agrupamiento. También la misma función `get_clust_tendency()` lo hace con una imagen menos vistosa. ¿Cuántos conglomerados se intuyen? Para saber más sobre el análisis de la tendencia al agrupamiento, la fórmula del estadístico de Hopkins y un ejemplo se puede consultar la página <https://www.datanovia.com/en/lessons/assessing-clustering-tendency/>

Se realiza el cálculo del estadístico de Hopkins. En este caso resulta en aprox 0.32, lo que se acerca medianamente a 0. Los datos tienen una tendencia a agruparse moderada. Al mirar la gráfica de las distancias (la raíz de la disimilaridad) se corrobora lo anterior. Si bien los datos no son homogéneos, la forma en como se pueden agrupar no se ve tan claramente. Se podría intuir con el gráfico que hay 1 sector donde se aglomeran bien y luego unos 4 o quizás 5 sectores donde los datos se aglomeran de manera menos estructurada, dependiendo del tamaño que se considere válido para una agrupación.

```
set.seed(41)
get_clust_tendency(df, nrow(df)-1, graph = F)$hopkins_stat
```

```
## [1] 0.320416
```

```
fviz_dist(sq_dist, lab_size = 5) +
  # hay un 'bug' al compilar el pdf, que se ven los bordes de cada 'tile'
  # para arreglarlo hay que mapear 'color' a la variable de valor.
  aes(color = value) +
  scale_fill_viridis_c(direction = -1, option = "A") +
  scale_color_viridis_c(direction = -1, option = "A") +
  guides(fill = F, color = F)
```



(e)

Realizar un análisis de conglomerados jerárquico con el método de *Ward* de la distancia propuesta en el apartado (c). Dibujar el dendrograma resultante.

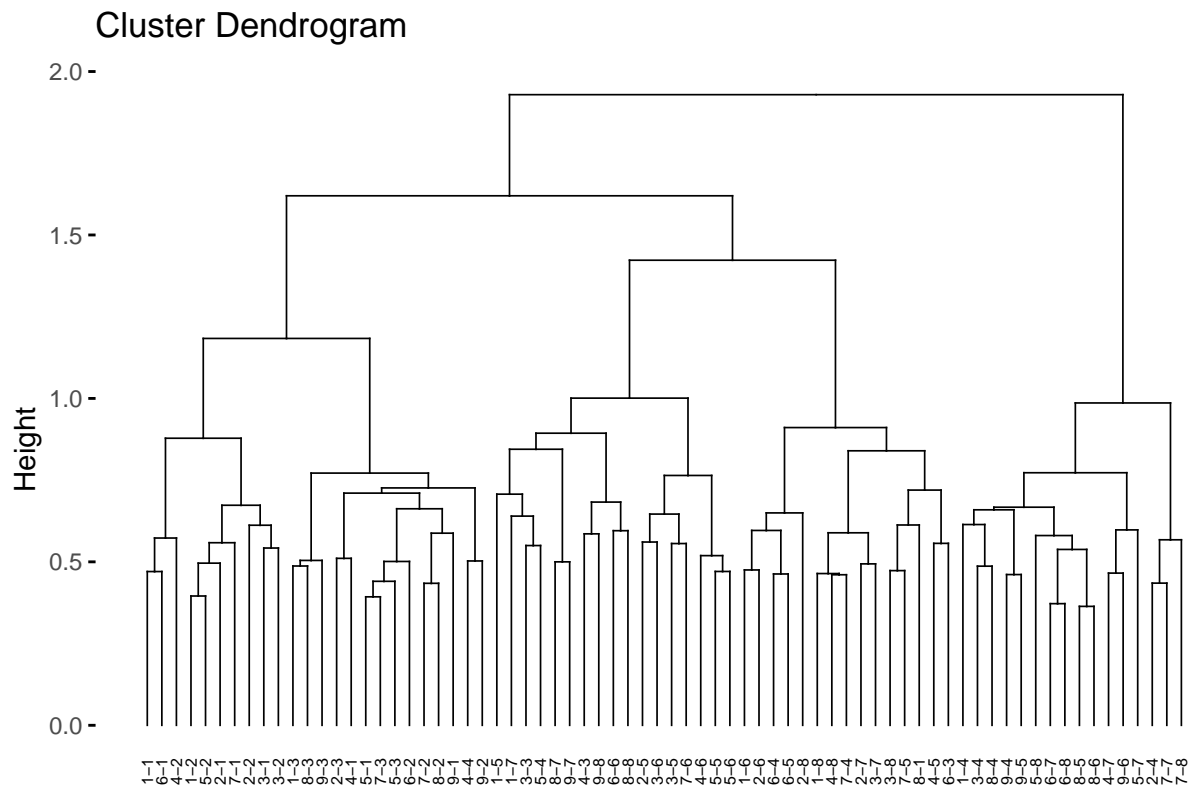
Nota: Justamente el método de *Ward* exige que la distancia sea euclídea, pero no necesariamente la distancia euclídea.

Dibujar también un *heatmap* de este análisis con la función `heatmap()`. Para ello, hay que elegir bien los parámetros `distfun=` y `hclustfun=` y una escala de colores. ¿Para qué sirve el *heatmap*?

Nota: No nos interesa reordenar las variables o columnas y tampoco un dendrograma sobre ellas.

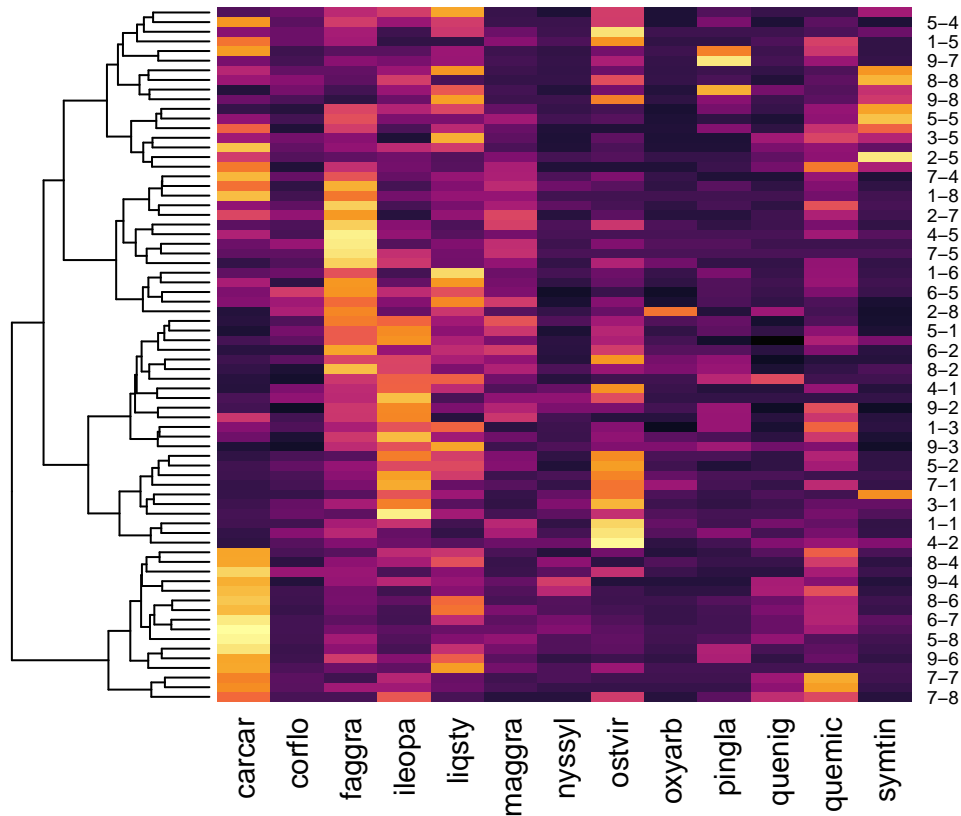
El dendrograma permite detallar mejor el agrupamiento aglomerativo que se intuía en el apartado anterior.

```
clust_1 <- agnes(sq_dist, diss = T, method = "ward")
fviz_dend(clust_1, cex = .4, lwd = .3)
```



La ventaja del “heatmap” es que incluye la información de las columnas. En este caso es posible ver la distribución de las especies en las diferentes muestras. Si además utilizamos un dendrograma y ordenamos las muestras según el dendrograma, obtenemos información visual de cómo se distribuyen las especies en las ramas del dendrograma, y cómo se podrían construir los diferentes clusters.

```
heatmap(as.matrix(df),
        Colv = NA,
        distfun = function(x) sqrt(vegdist(x)),
        hclustfun = hcut,
        col = hcl.colors(256, palette = "Inferno"))
```



```
hcl.pals()
```

```
## [1] "Pastel 1"      "Dark 2"      "Dark 3"      "Set 2"
## [5] "Set 3"        "Warm"        "Cold"        "Harmonic"
## [9] "Dynamic"      "Grays"       "Light Grays" "Blues 2"
## [13] "Blues 3"      "Purples 2"   "Purples 3"   "Reds 2"
## [17] "Reds 3"      "Greens 2"    "Greens 3"    "Oslo"
## [21] "Purple-Blue"  "Red-Purple"  "Red-Blue"    "Purple-Orange"
## [25] "Purple-Yellow" "Blue-Yellow" "Green-Yellow" "Red-Yellow"
## [29] "Heat"        "Heat 2"      "Terrain"     "Terrain 2"
## [33] "Viridis"     "Plasma"      "Inferno"     "Dark Mint"
## [37] "Mint"        "BluGrn"      "Teal"        "TealGrn"
## [41] "Emrld"       "BluYl"       "ag_GrnYl"    "Peach"
## [45] "PinkYl"      "Burg"        "BurgYl"      "RedOr"
## [49] "OrYel"       "Purp"        "PurpOr"      "Sunset"
## [53] "Magenta"     "SunsetDark"  "ag_Sunset"   "BrwnYl"
## [57] "YlOrRd"      "YlOrBr"      "OrRd"        "Oranges"
## [61] "YlGn"        "YlGnBu"      "Reds"        "RdPu"
## [65] "PuRd"        "Purples"     "PuBuGn"      "PuBu"
## [69] "Greens"      "BuGn"        "GnBu"        "BuPu"
## [73] "Blues"       "Lajolla"     "Turku"       "Blue-Red"
## [77] "Blue-Red 2"  "Blue-Red 3"  "Red-Green"   "Purple-Green"
## [81] "Purple-Brown" "Green-Brown" "Blue-Yellow 2" "Blue-Yellow 3"
## [85] "Green-Orange" "Cyan-Magenta" "Tropic"      "Broc"
## [89] "Cork"        "Vik"         "Berlin"      "Lisbon"
## [93] "Tofino"      "ArmyRose"    "Earth"       "Fall"
## [97] "Geyser"      "TealRose"    "Temps"       "PuOr"
## [101] "RdBu"        "RdGy"        "PiYG"        "PRGn"
```

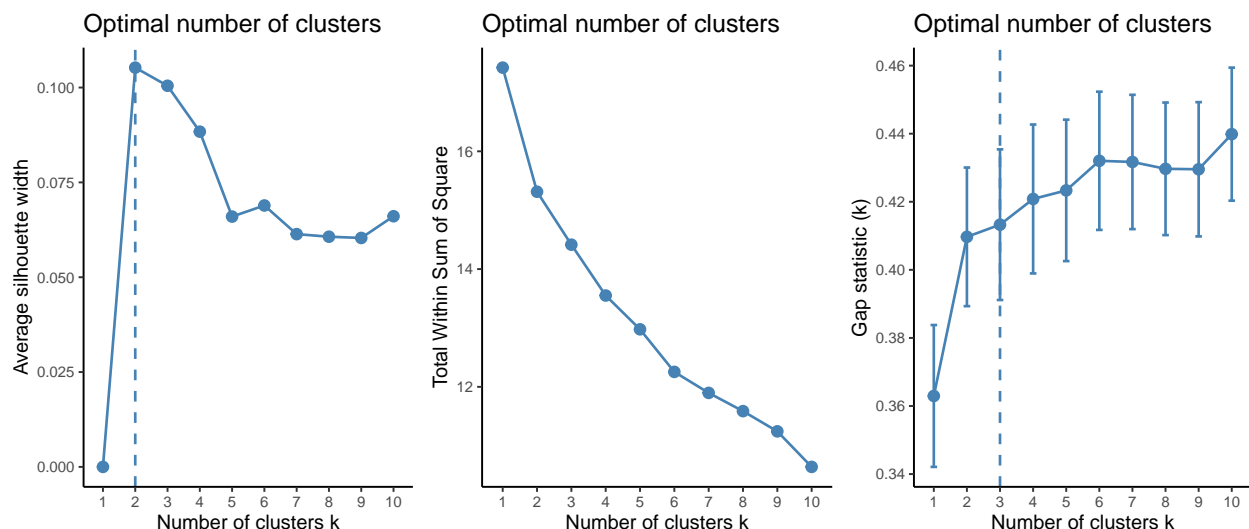
```
## [105] "BrBG"          "RdYlBu"          "RdYlGn"          "Spectral"
## [109] "Zissou 1"      "Cividis"
```

---

(f)

El siguiente paso es estudiar por algún criterio el número óptimo de conglomerados para el análisis jerárquico. Con la distancia del apartado (c) en particular, lo más sencillo es utilizar el criterio de las siluetas. Sin embargo, el criterio de las siluetas y otros métodos como el de la suma de cuadrados dentro de los grupos o WSS o también el estadístico GAP se pueden aplicar con la función `fviz_nbclust()` del paquete `factoextra`. Habrá que especificar que se trata del método de clasificación jerárquico (que por defecto utiliza el método de Ward) y también la distancia raíz cuadrada de la disimilaridad de Bray-Curtis.

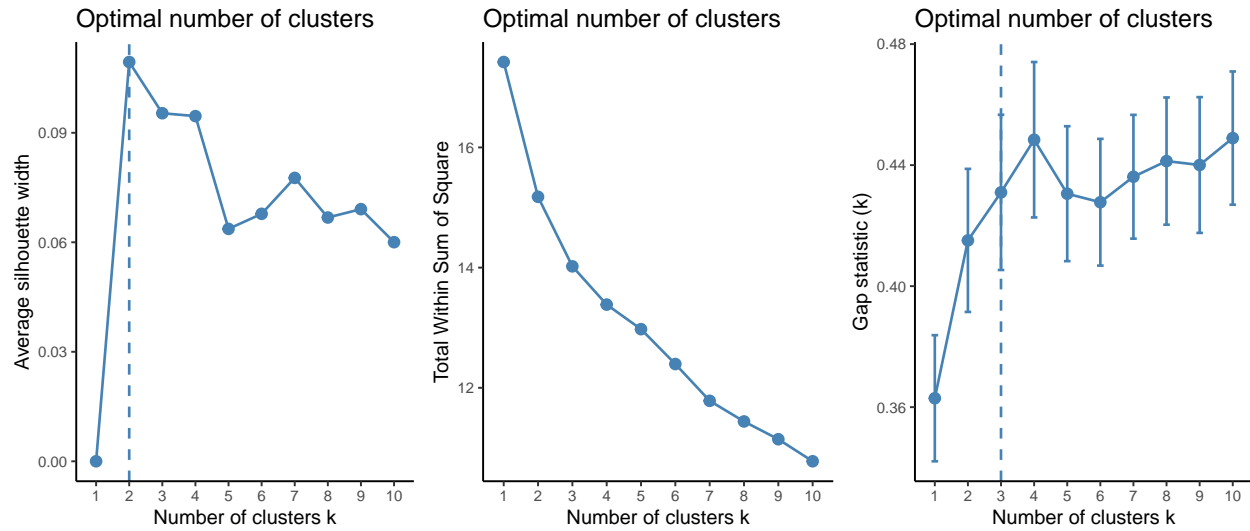
```
grid.arrange(
  fviz_nbclust(df, diss = sq_dist, FUNcluster = hcut, method = "silhouette") +
    theme_classic(base_size = 8),
  fviz_nbclust(df, diss = sq_dist, FUNcluster = hcut, method = "wss") +
    theme_classic(base_size = 8),
  fviz_nbclust(df, diss = sq_dist, FUNcluster = hcut, method = "gap_stat", verbose = F) +
    theme_classic(base_size = 8),
  nrow = 1)
```



(g)

Estudiar con la misma distancia el número óptimo de conglomerados con el método PAM.

```
grid.arrange(
  fviz_nbclust(df, diss = sq_dist, FUNcluster = pam, method = "silhouette") +
    theme_classic(base_size = 8),
  fviz_nbclust(df, diss = sq_dist, FUNcluster = pam, method = "wss") +
    theme_classic(base_size = 8),
  fviz_nbclust(df, diss = sq_dist, FUNcluster = pam, method = "gap_stat", verbose = F) +
    theme_classic(base_size = 8),
  nrow = 1)
```



(h)

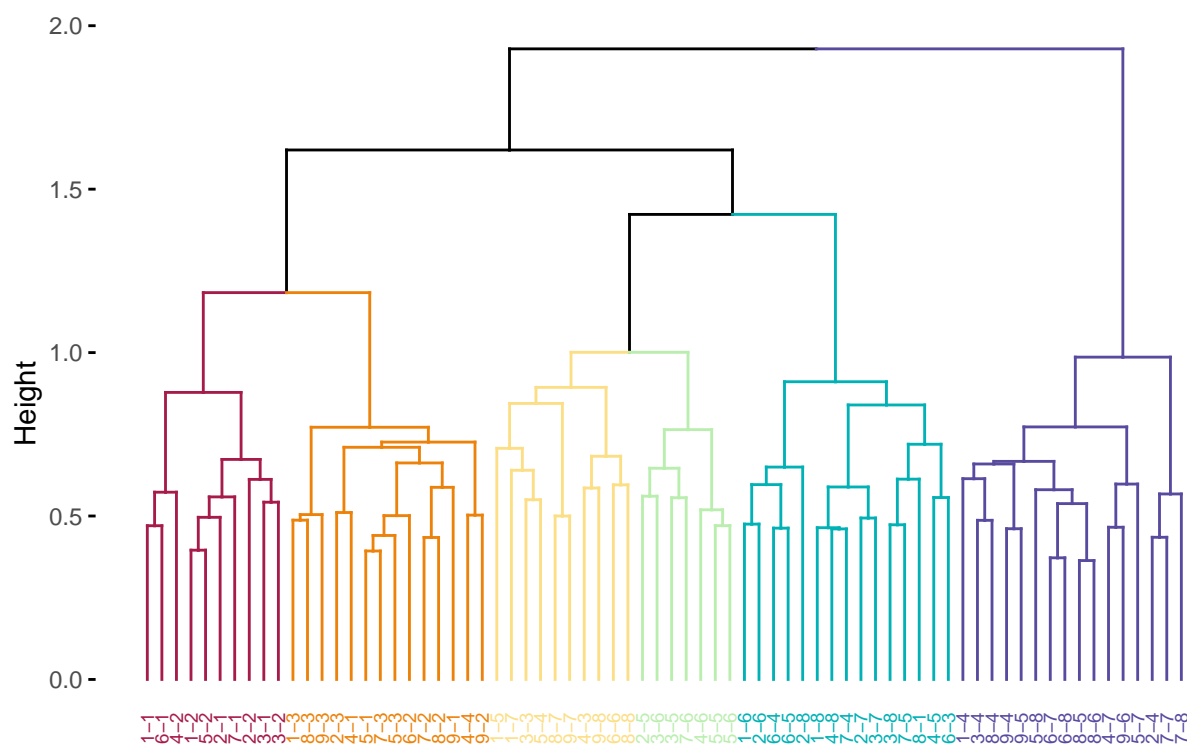
De los apartados anteriores se deduce que no hay un número claro de conglomerados. Un ecólogo experto propone que sean 6, ya que su experiencia así lo ha visto en otros bosques similares. Dibujar el dendograma del apartado (e) con esta partición. Con el objetivo de caracterizar los conglomerados, calcular una tabla con las medianas de las frecuencias de cada especie en cada uno de los seis conglomerados del dendograma anterior. Descartar de esta tabla las especies con una mediana inferior o igual a 4 en todos los conglomerados. Representar la tabla anterior con un análisis de correspondencias y un gráfico asimétrico.

```
clust_6 <- hcut(sq_dist, k = 6, hc_func = "agnes", hc_method = "ward.D2")

fviz_dend(clust_6,
  palette = hcl.colors(palette = "Spectral", 6),
  lwd = .5,
  cex = .5)
```



## Cluster Dendrogram

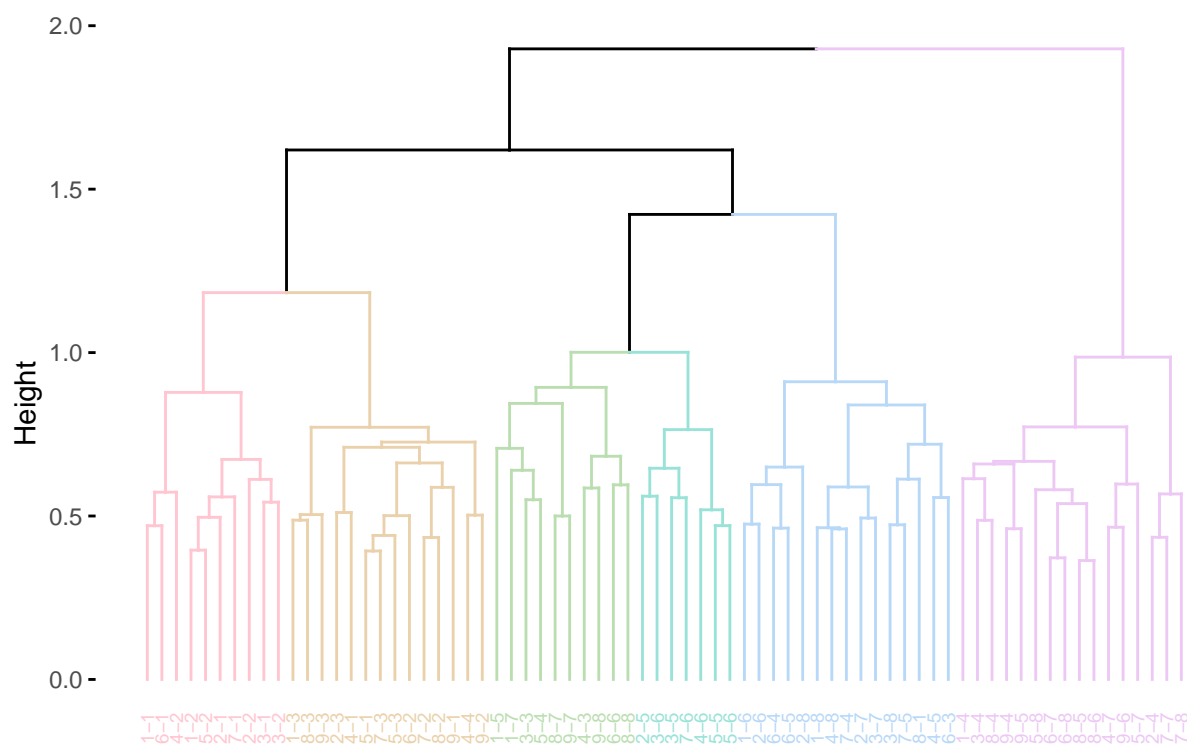


```
dendo <- function(pal){
fviz_dend(clust_6,
  palette = hcl.colors(palette = pal, 6),
  lwd = .5,
  cex = .5)}
```

```
lapply(hcl.pals(), dendo)
```

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

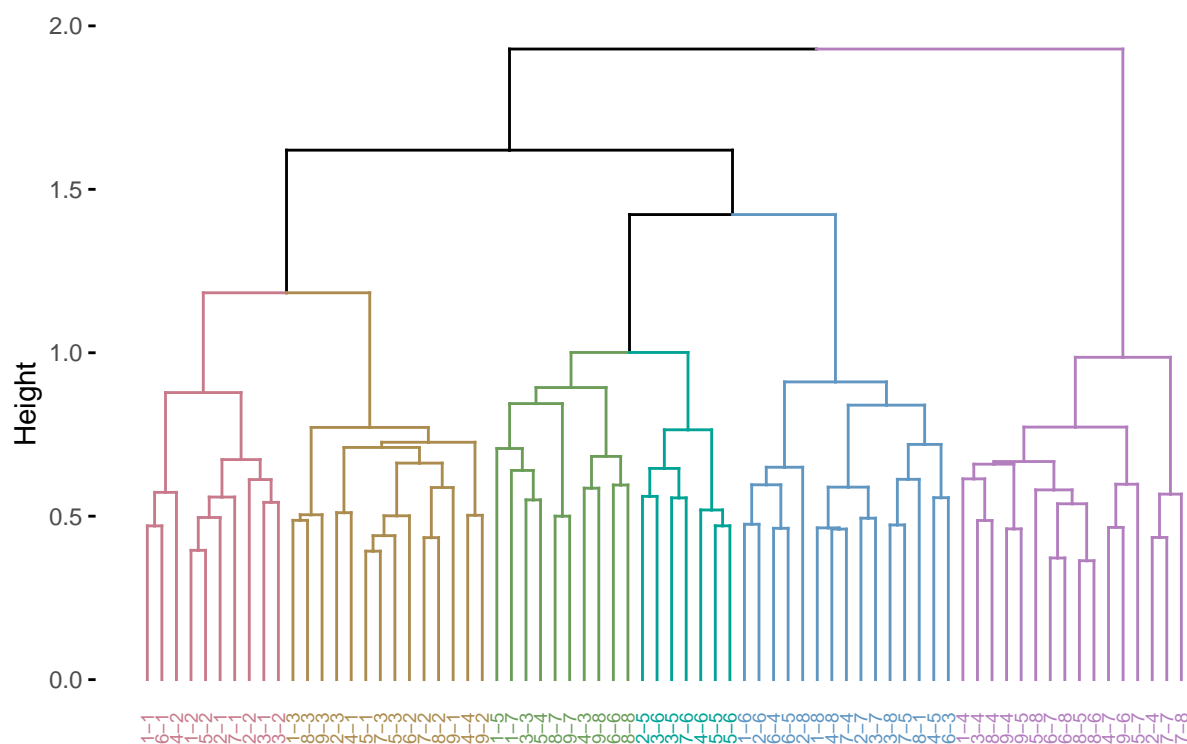
Cluster Dendrogram



##

## [[2]]

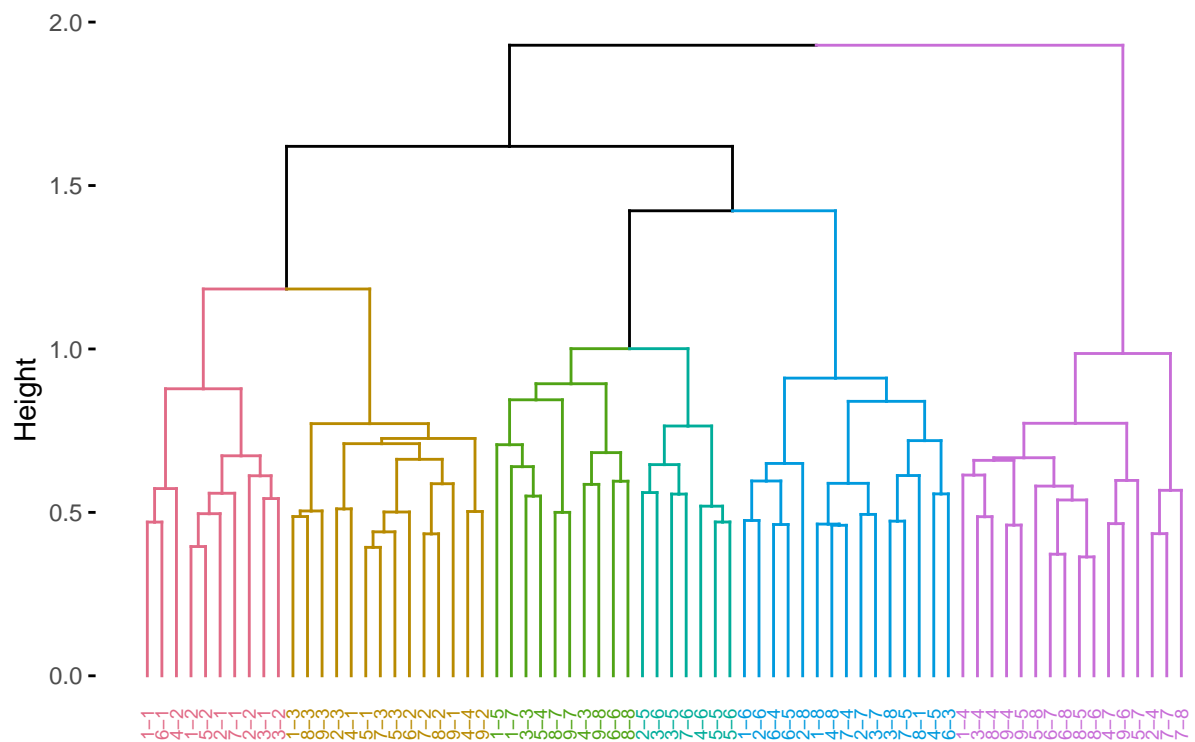
Cluster Dendrogram



##

## [[3]]

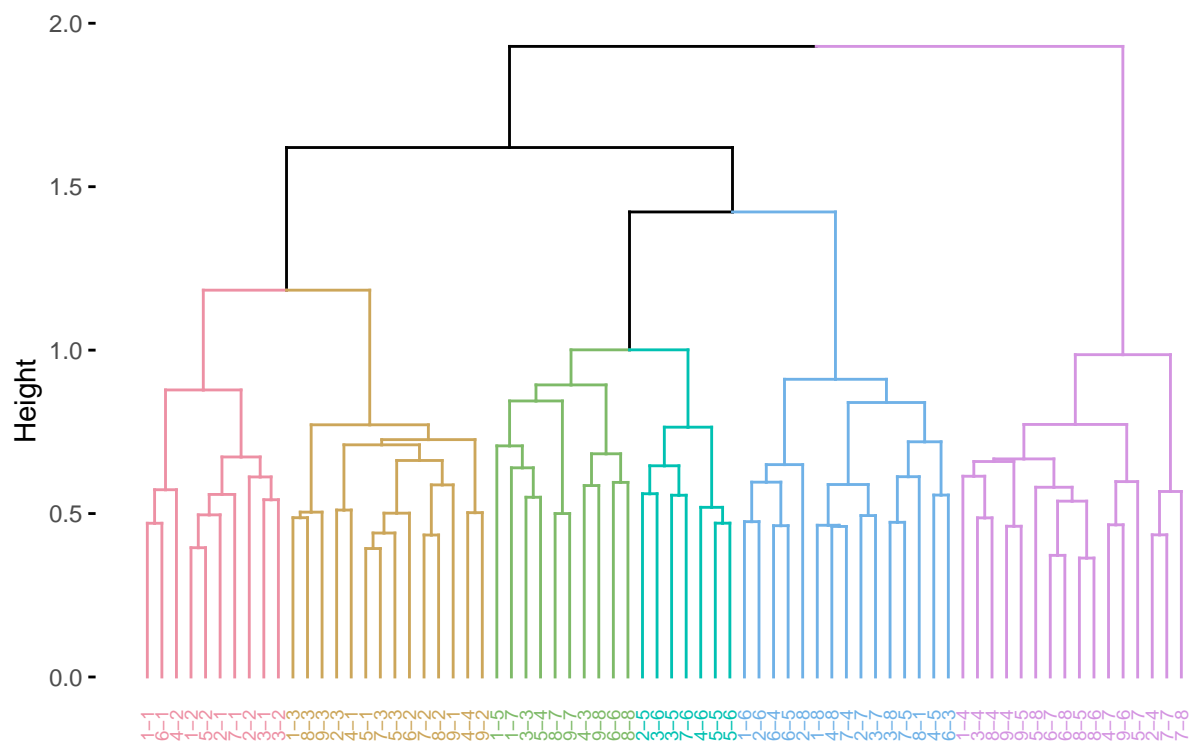
Cluster Dendrogram



##

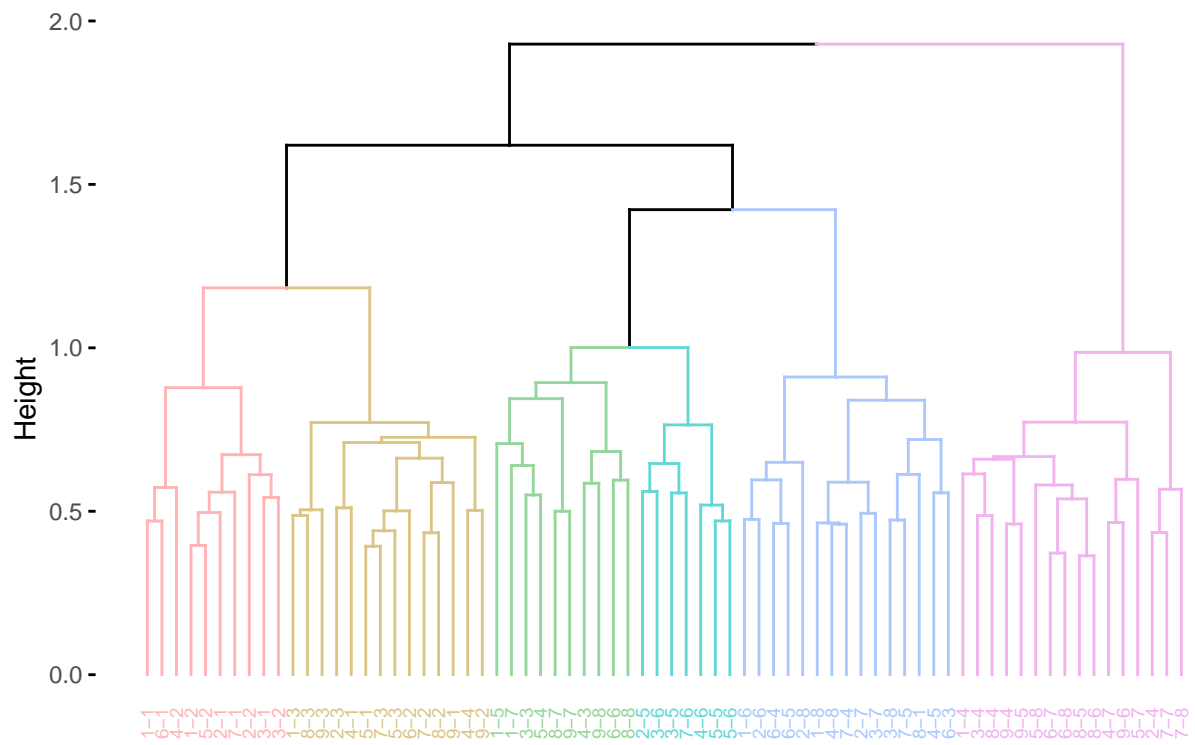
## [[4]]

Cluster Dendrogram



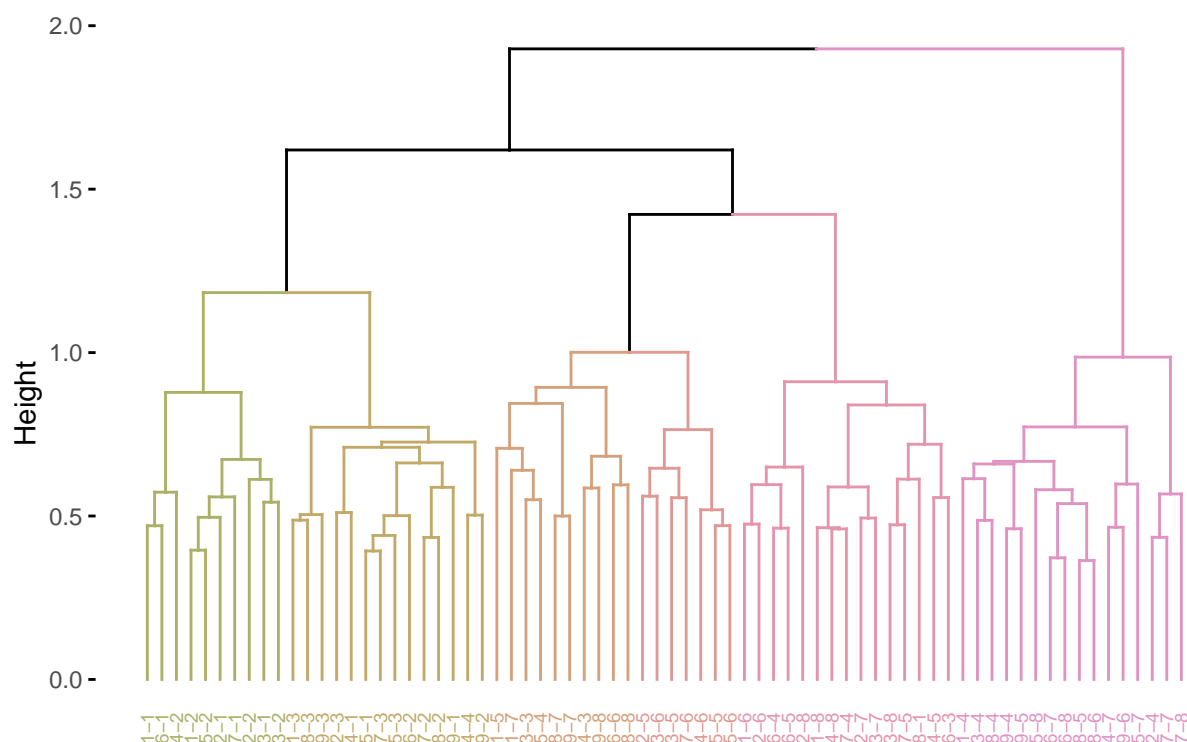
##  
## [[5]]

Cluster Dendrogram



##  
## [[6]]

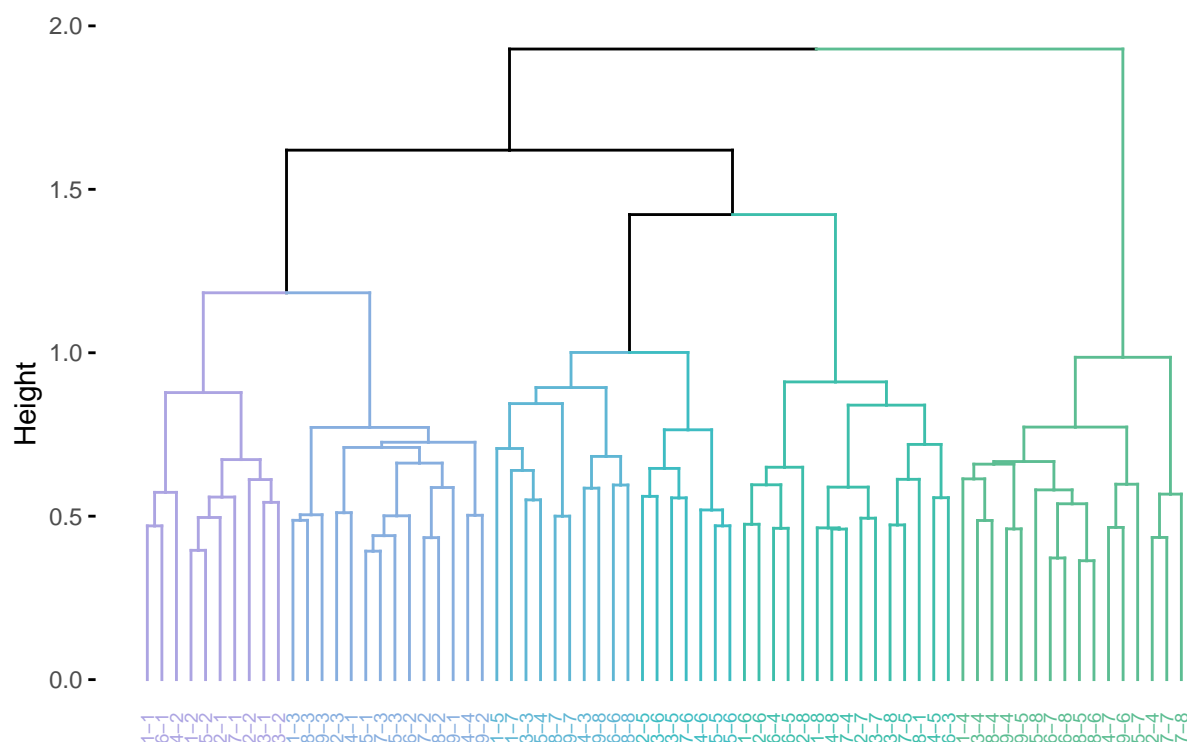
# Cluster Dendrogram



##

## [[7]]

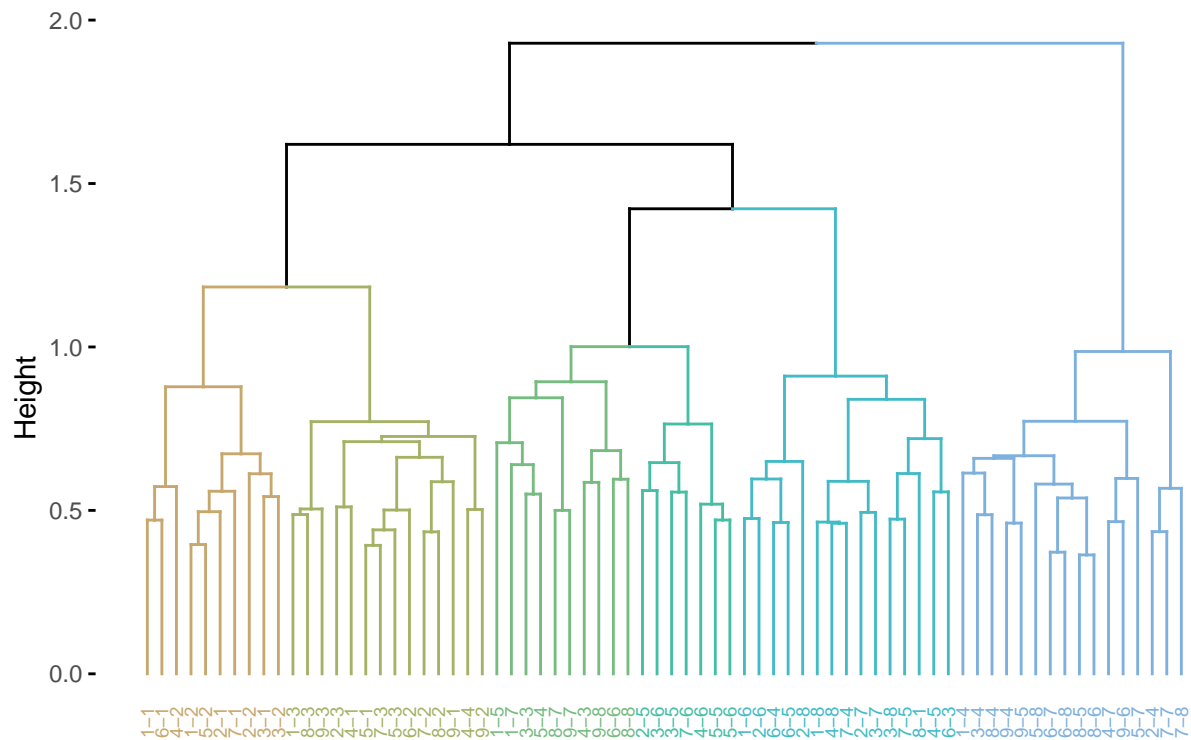
# Cluster Dendrogram



##

## [[8]]

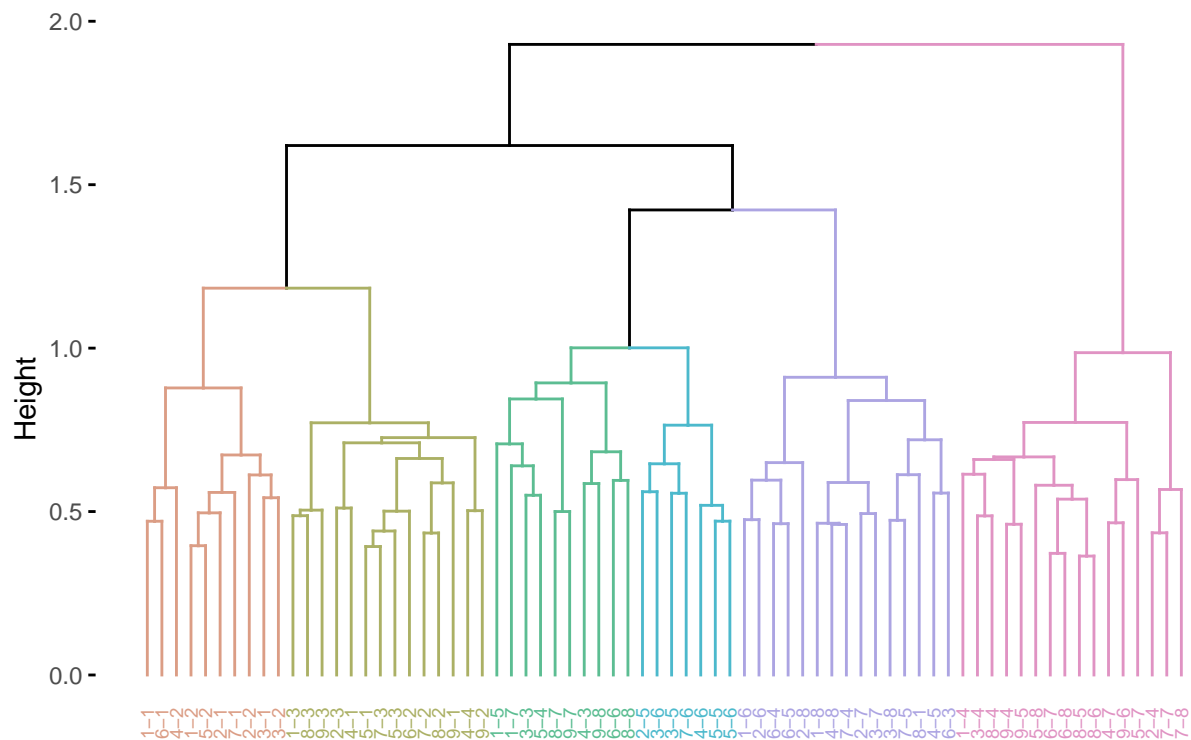
Cluster Dendrogram



##

## [[9]]

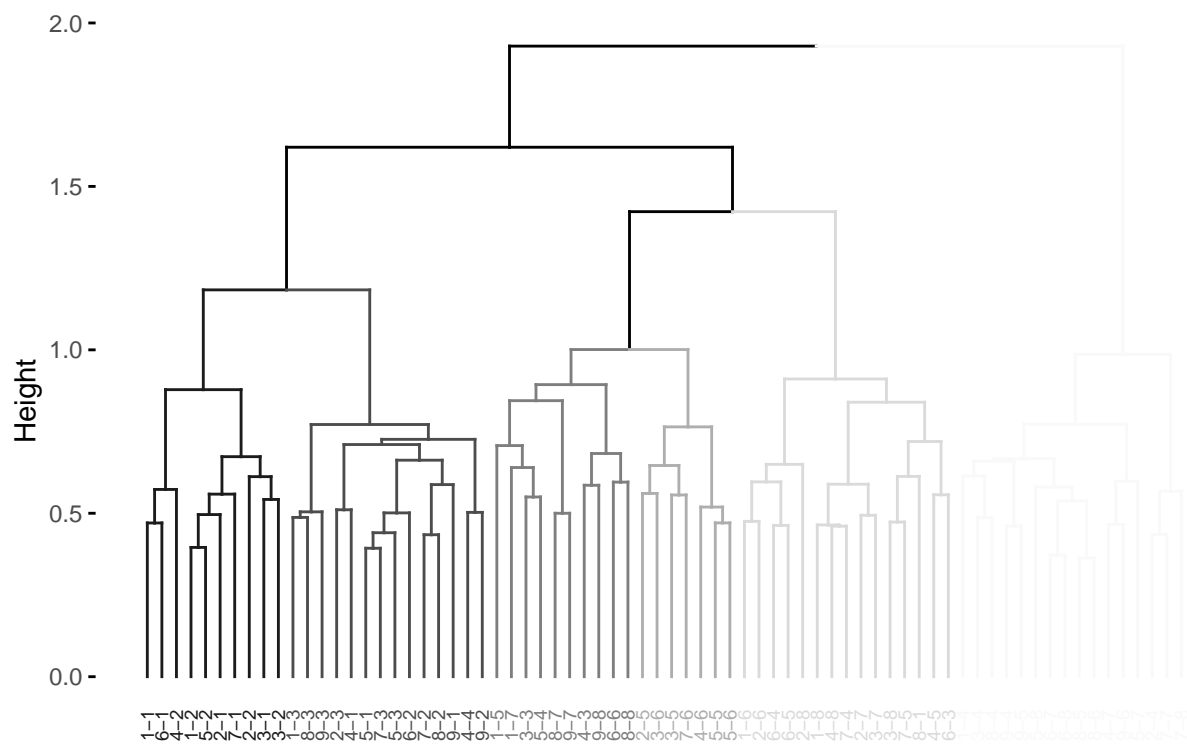
Cluster Dendrogram



##

## [[10]]

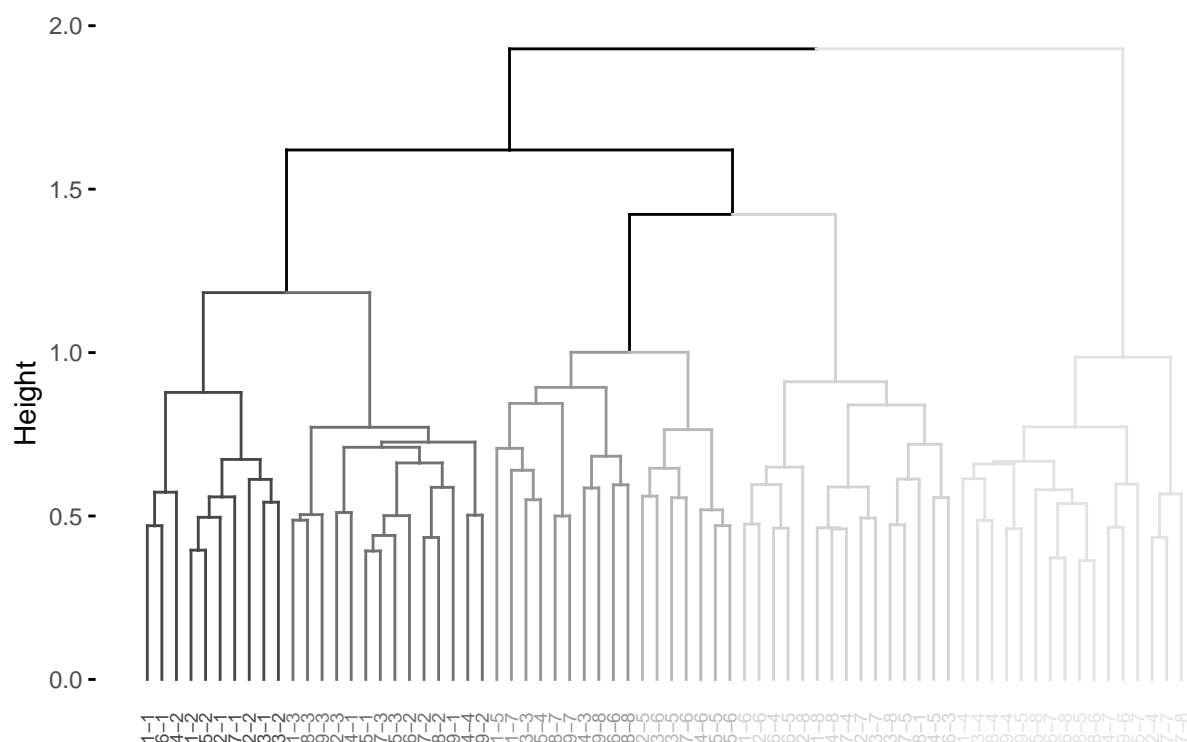
Cluster Dendrogram



##

## [[11]]

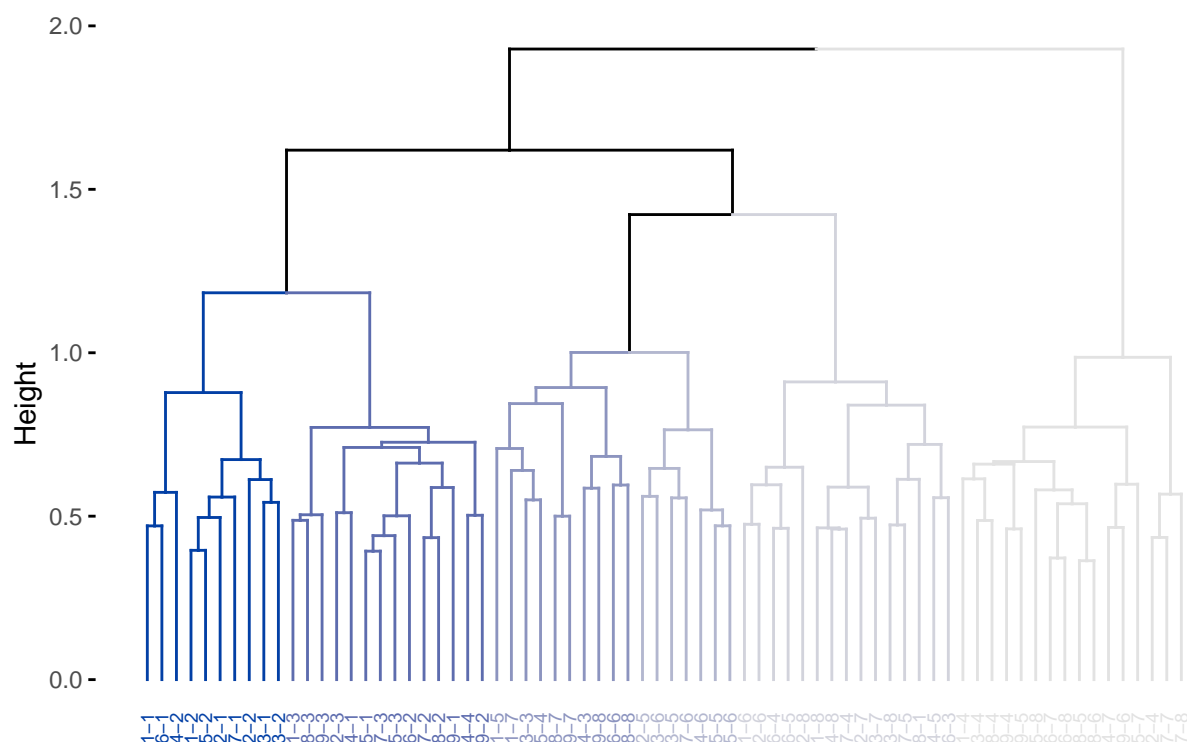
## Cluster Dendrogram



##

```
## [[12]]
```

## Cluster Dendrogram

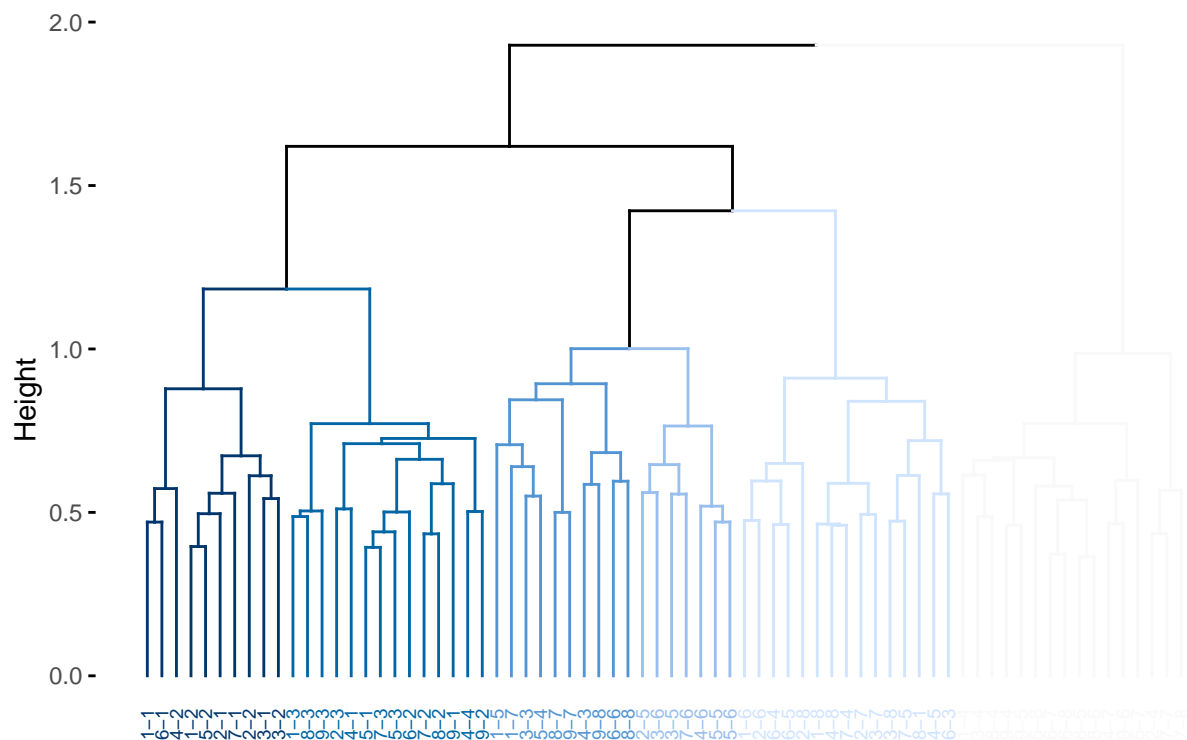


##



## [[13]]

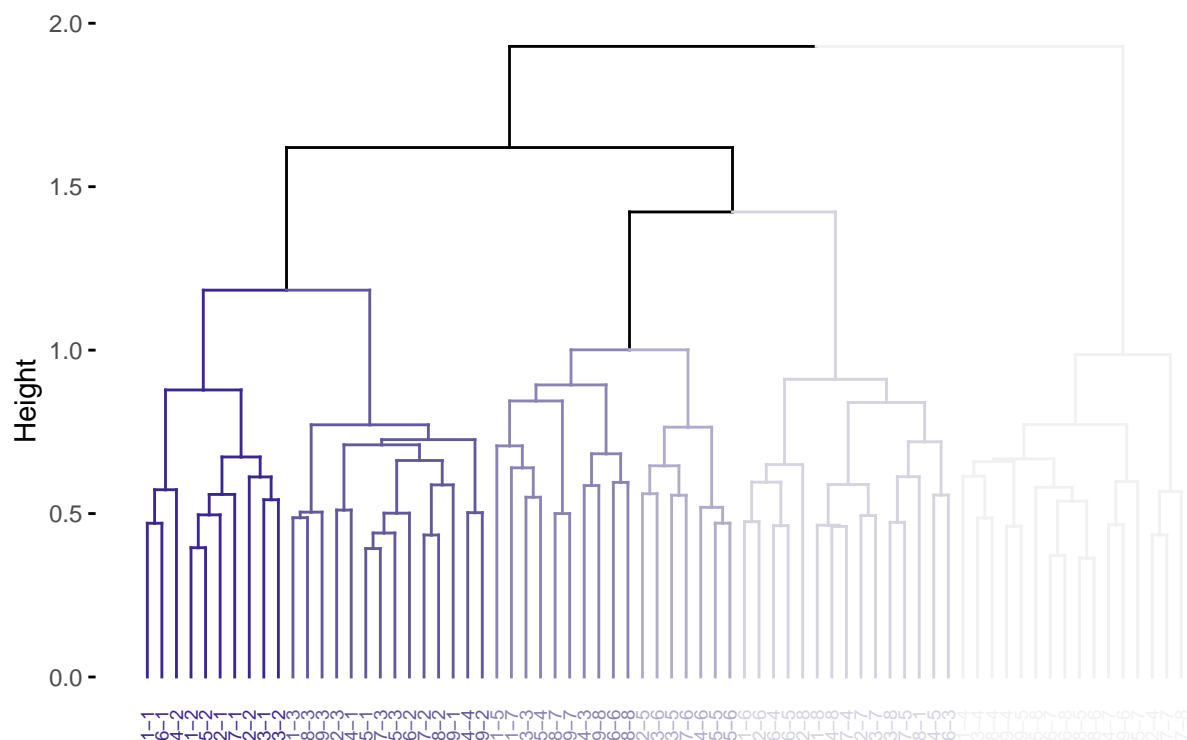
Cluster Dendrogram



##

## [[14]]

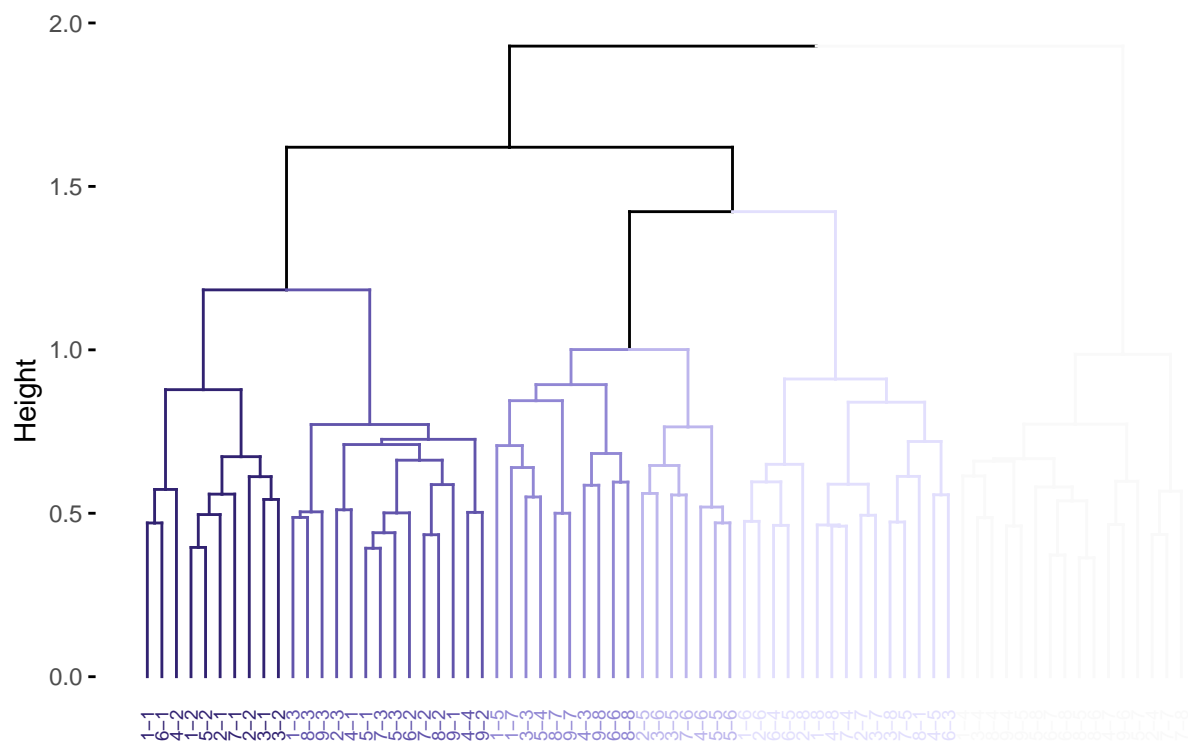
Cluster Dendrogram



##

## [[15]]

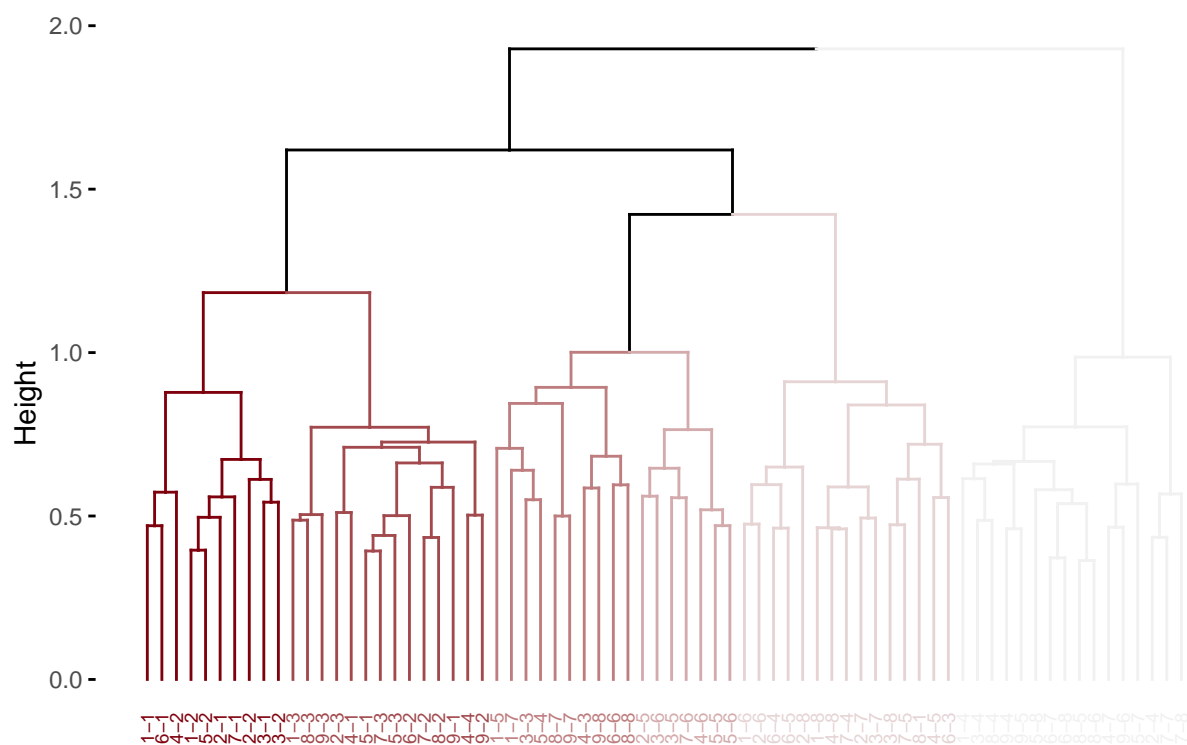
Cluster Dendrogram



##

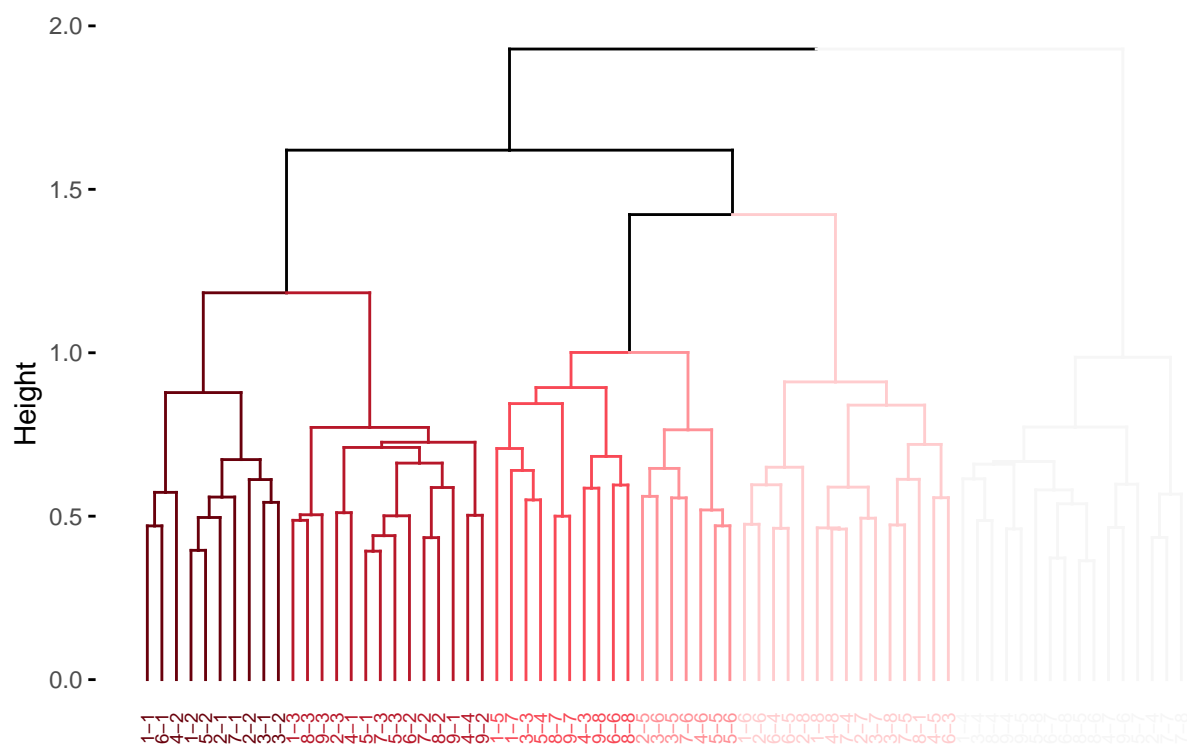
## [[16]]

# Cluster Dendrogram



##  
## [[17]]

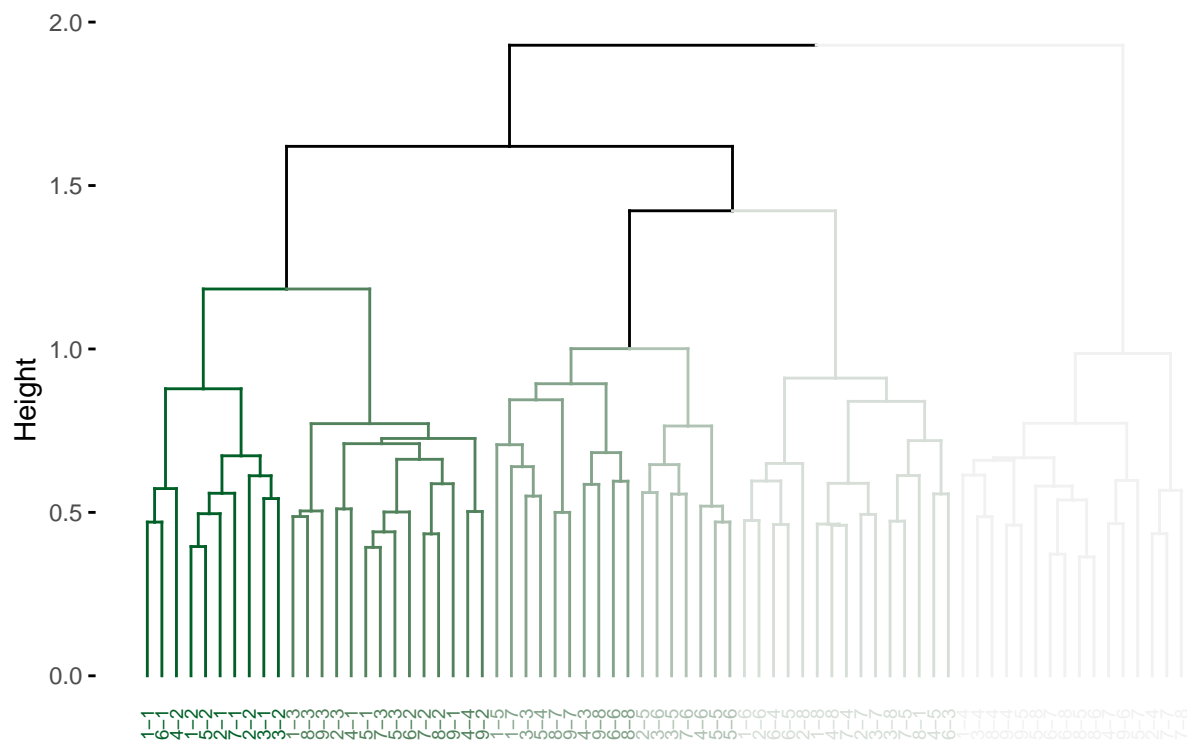
# Cluster Dendrogram



##

## [[18]]

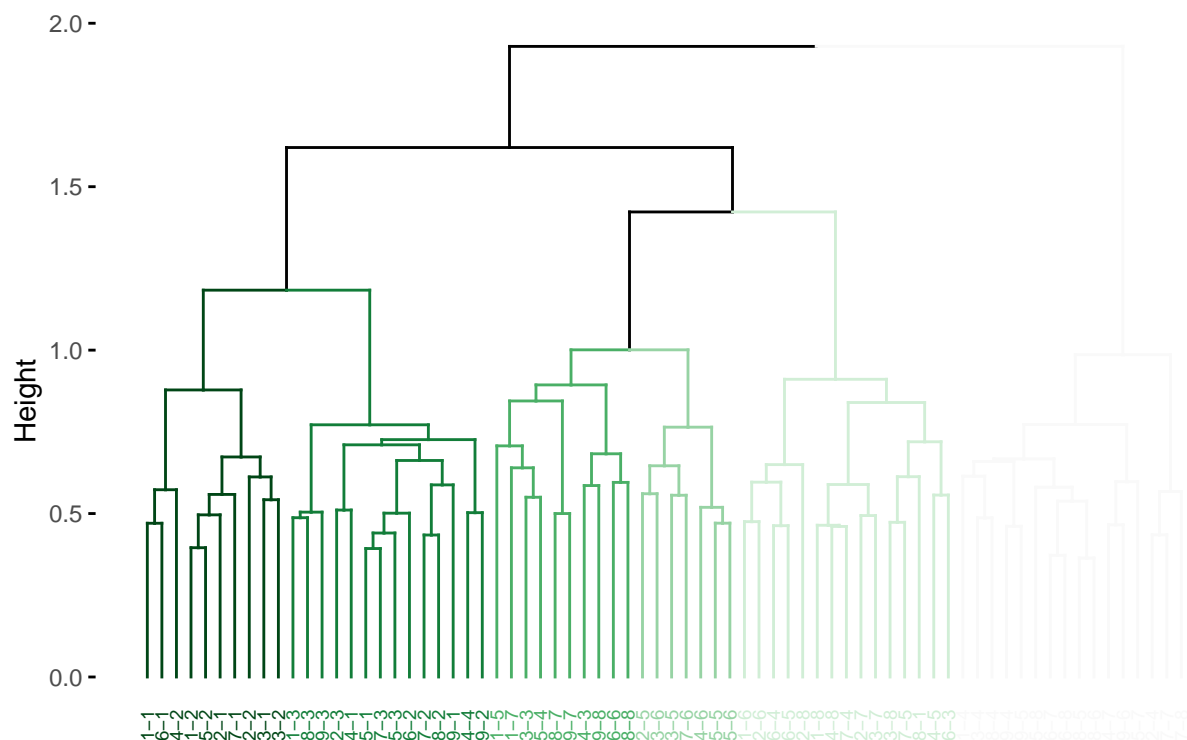
Cluster Dendrogram



##

## [[19]]

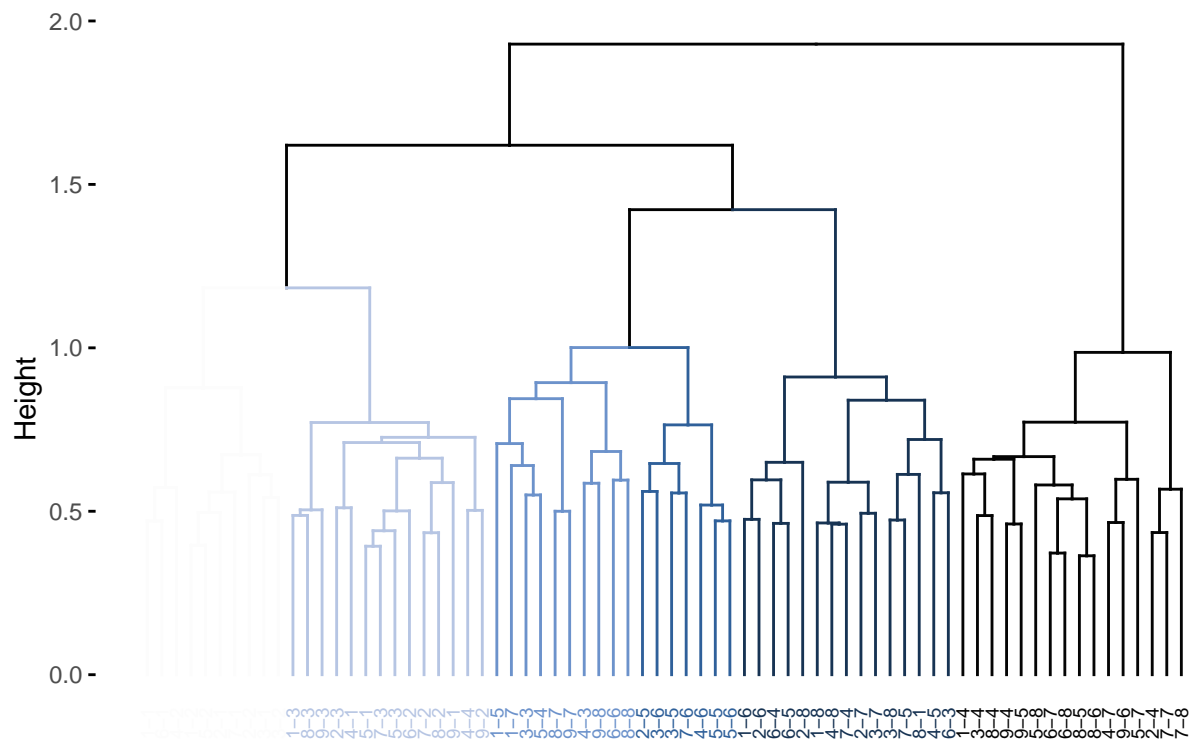
Cluster Dendrogram



##

## [[20]]

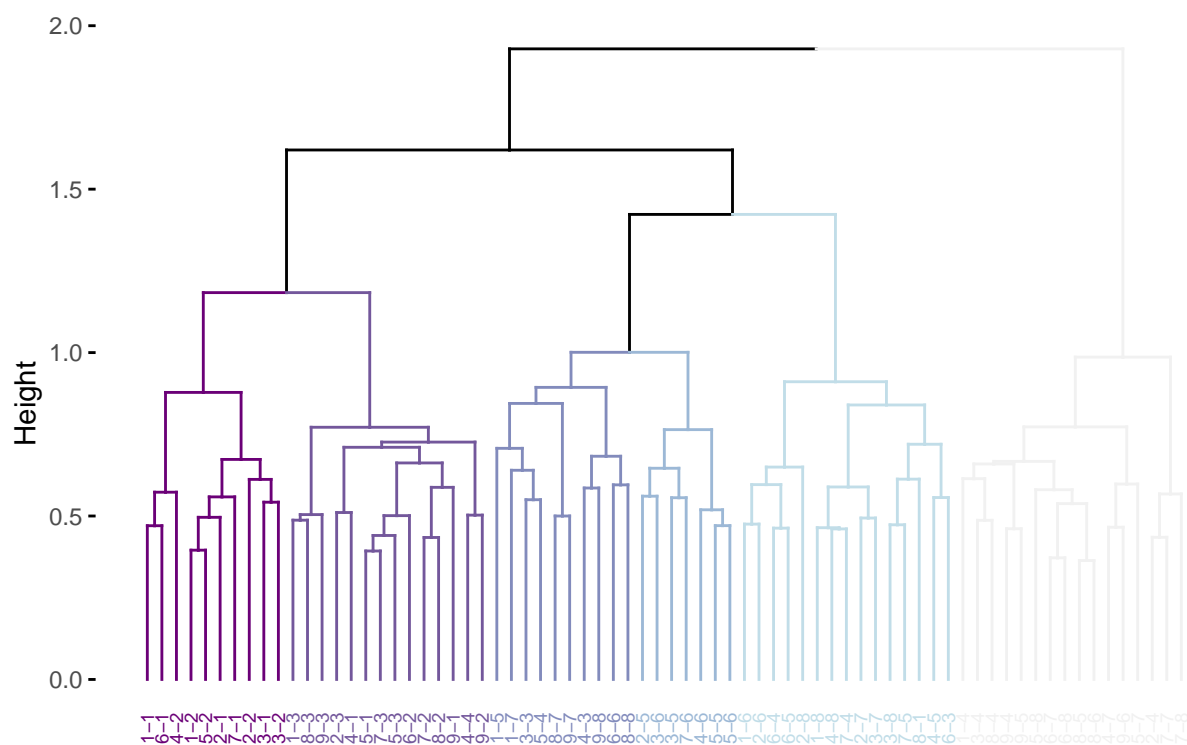
Cluster Dendrogram



##

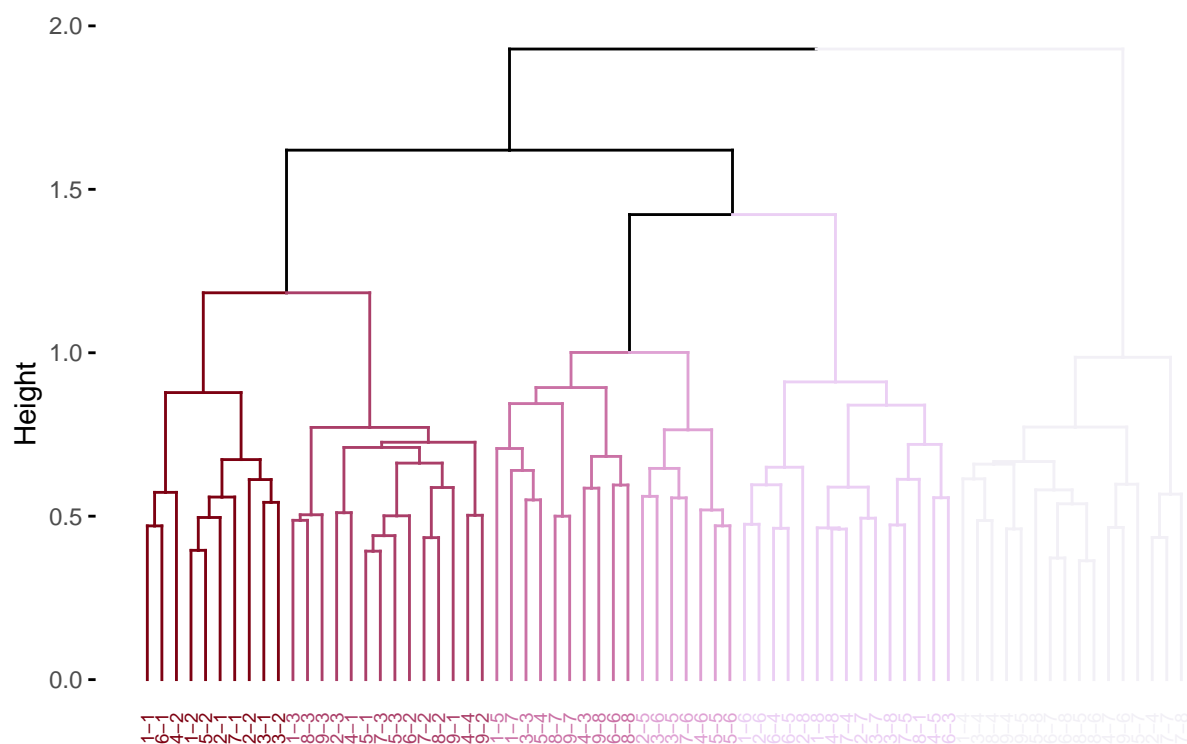
## [[21]]

# Cluster Dendrogram



##  
## [[22]]

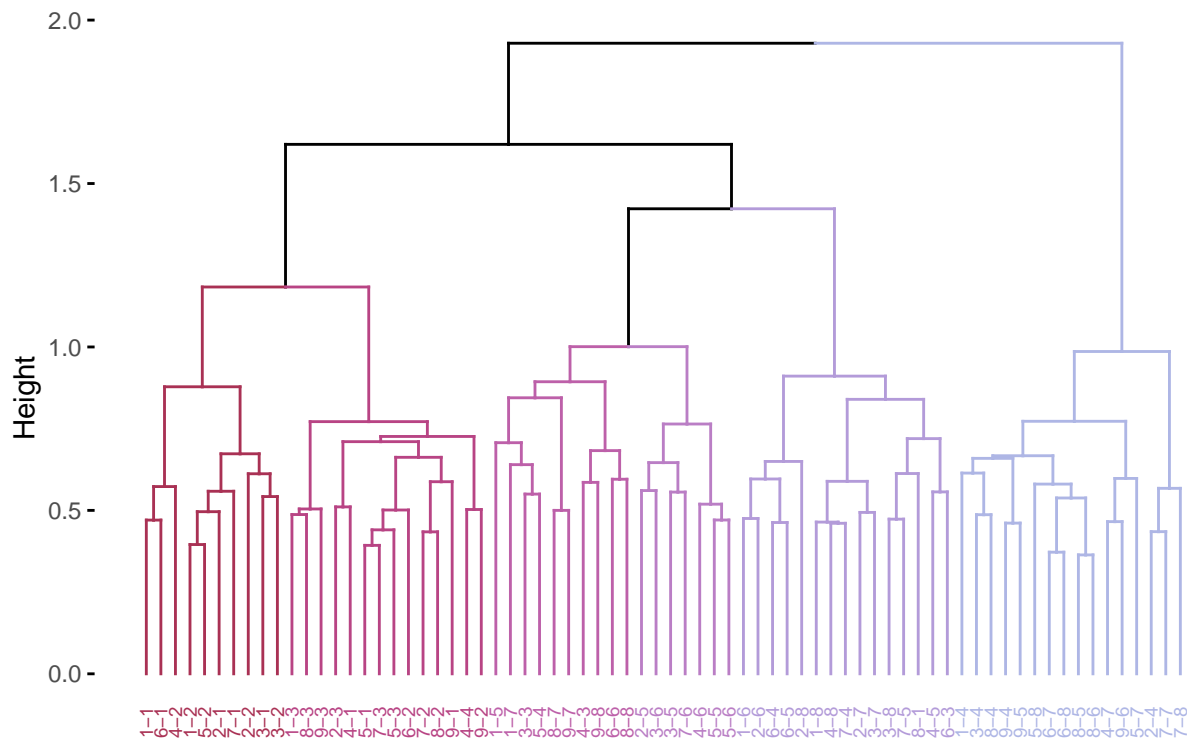
# Cluster Dendrogram



##

## [[23]]

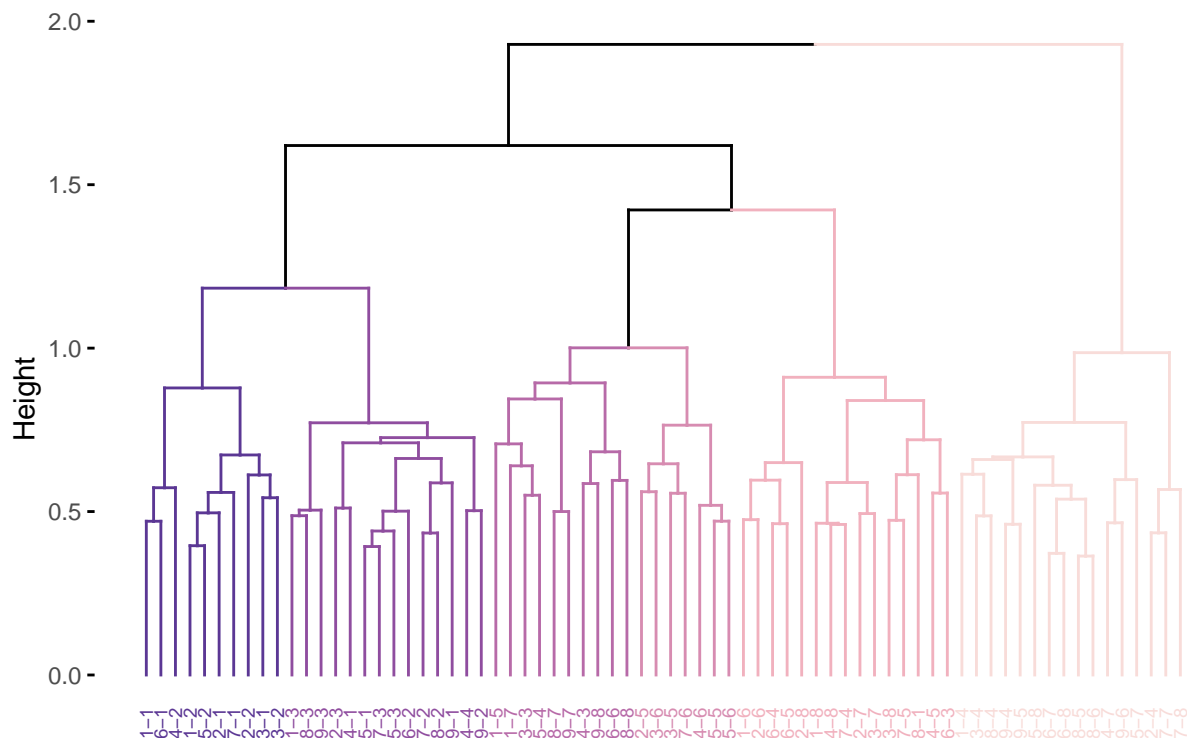
Cluster Dendrogram



##

## [[24]]

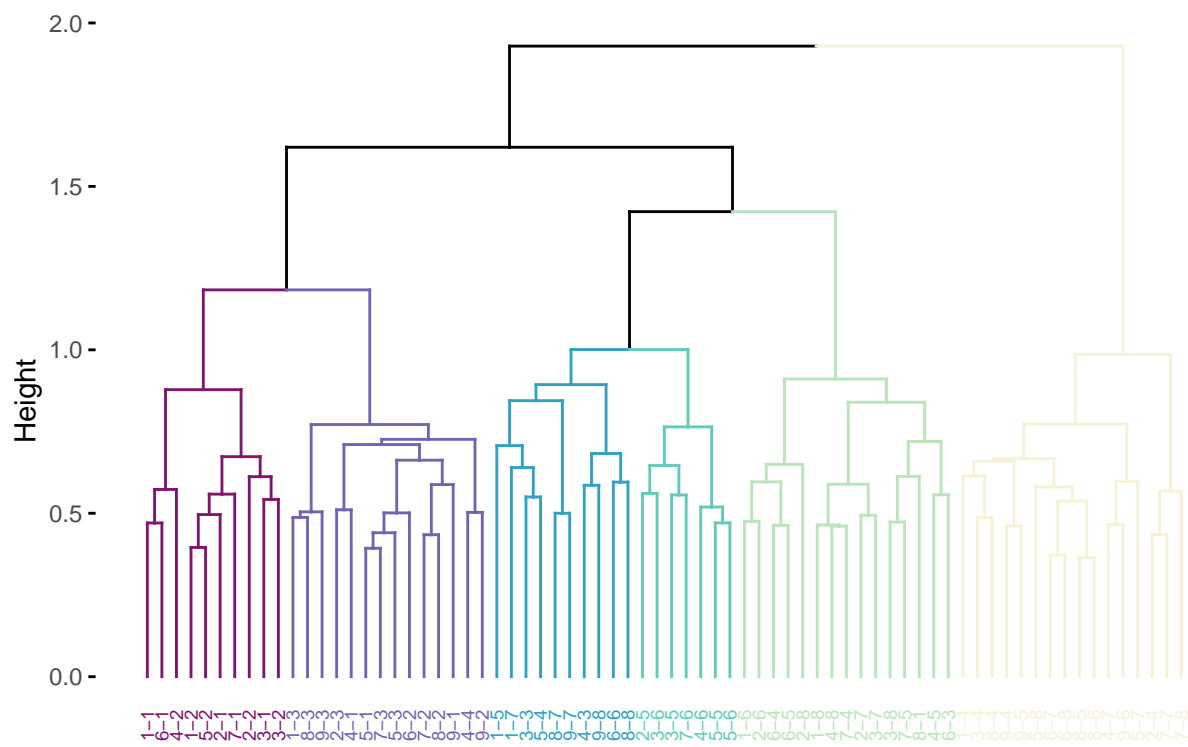
Cluster Dendrogram



##

## [[25]]

Cluster Dendrogram

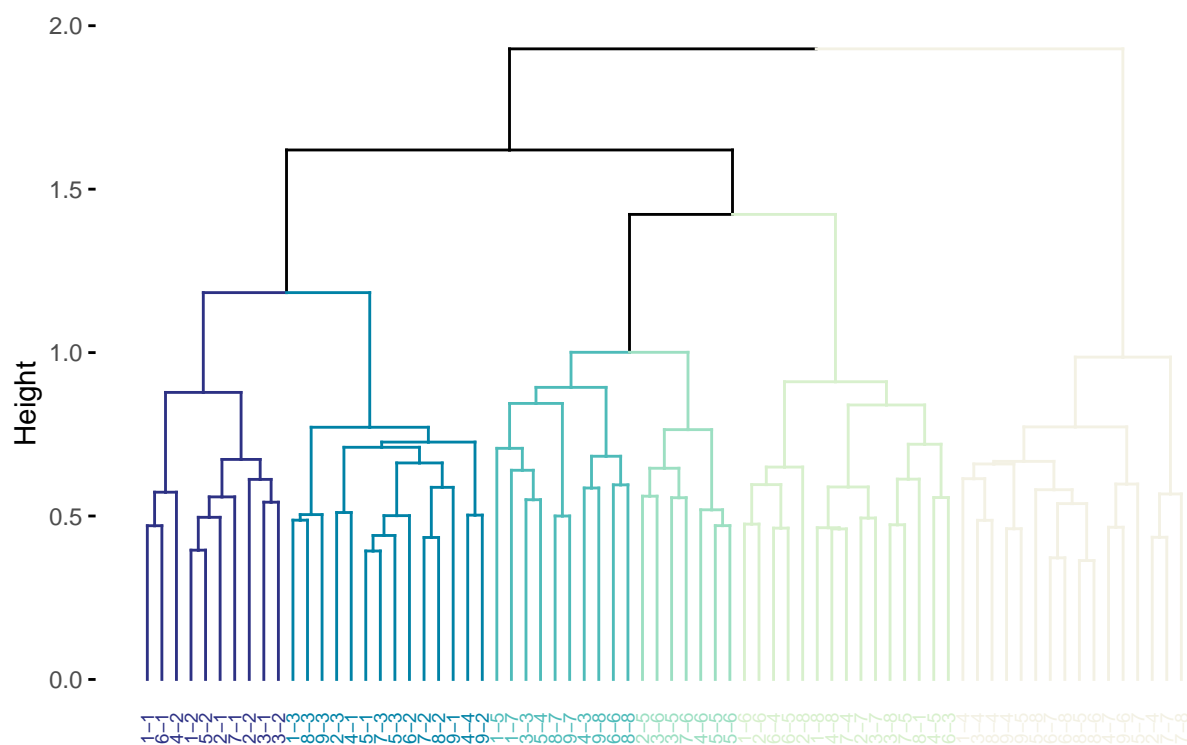


##

## [[26]]



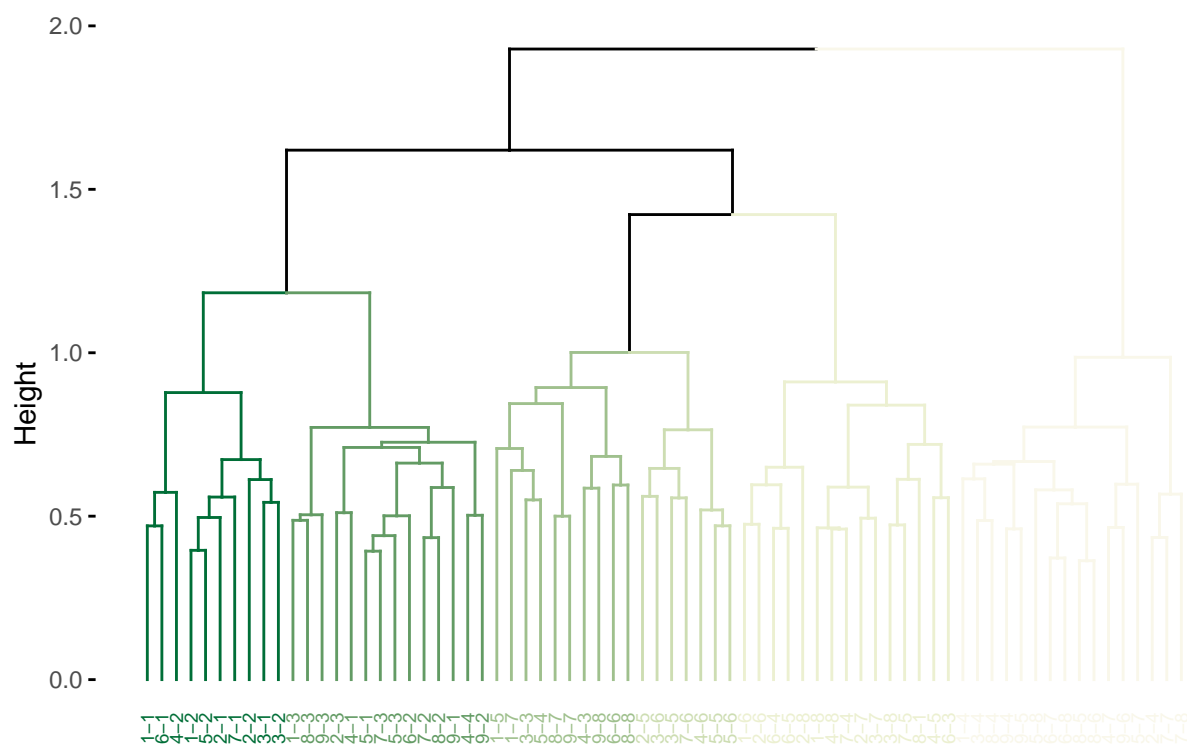
# Cluster Dendrogram



##

## [[27]]

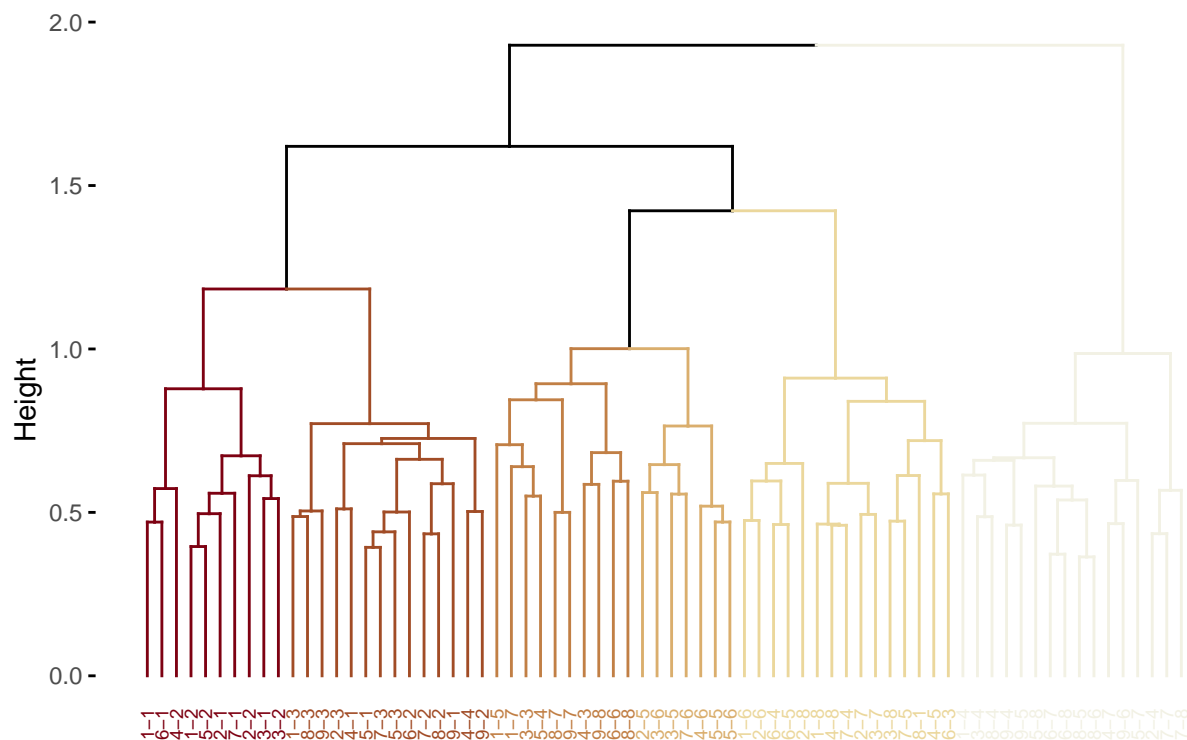
# Cluster Dendrogram



##

## [[28]]

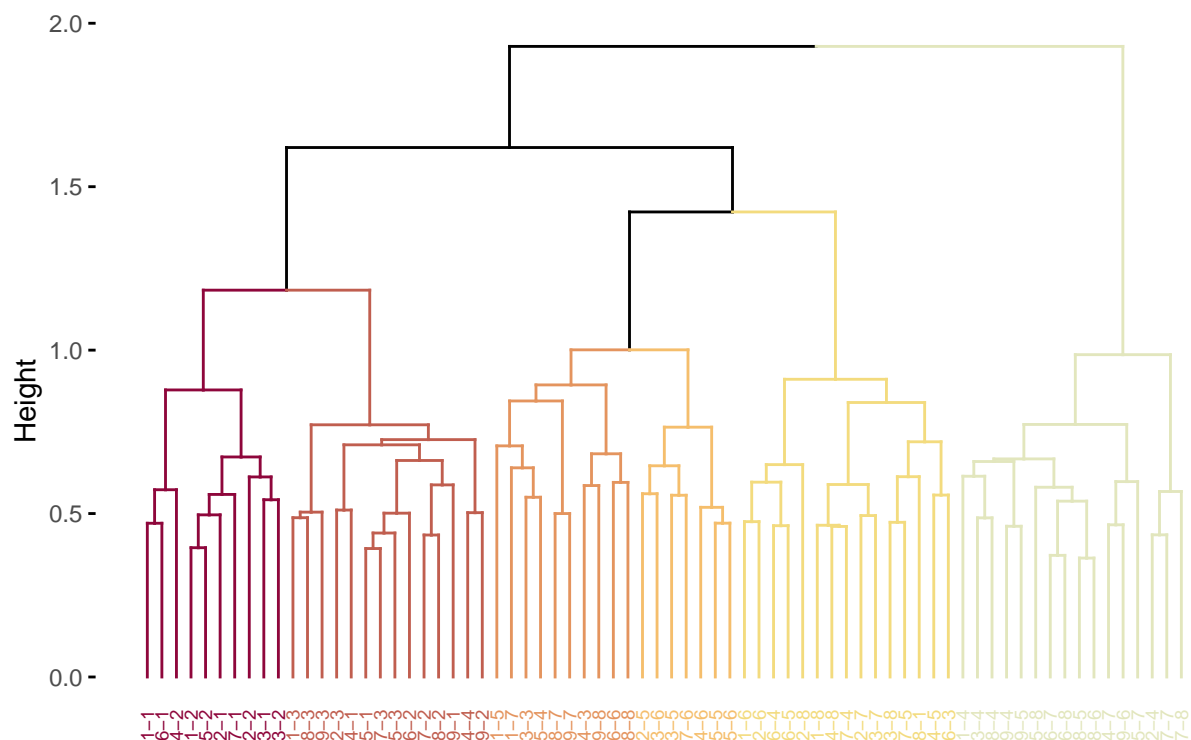
Cluster Dendrogram



##

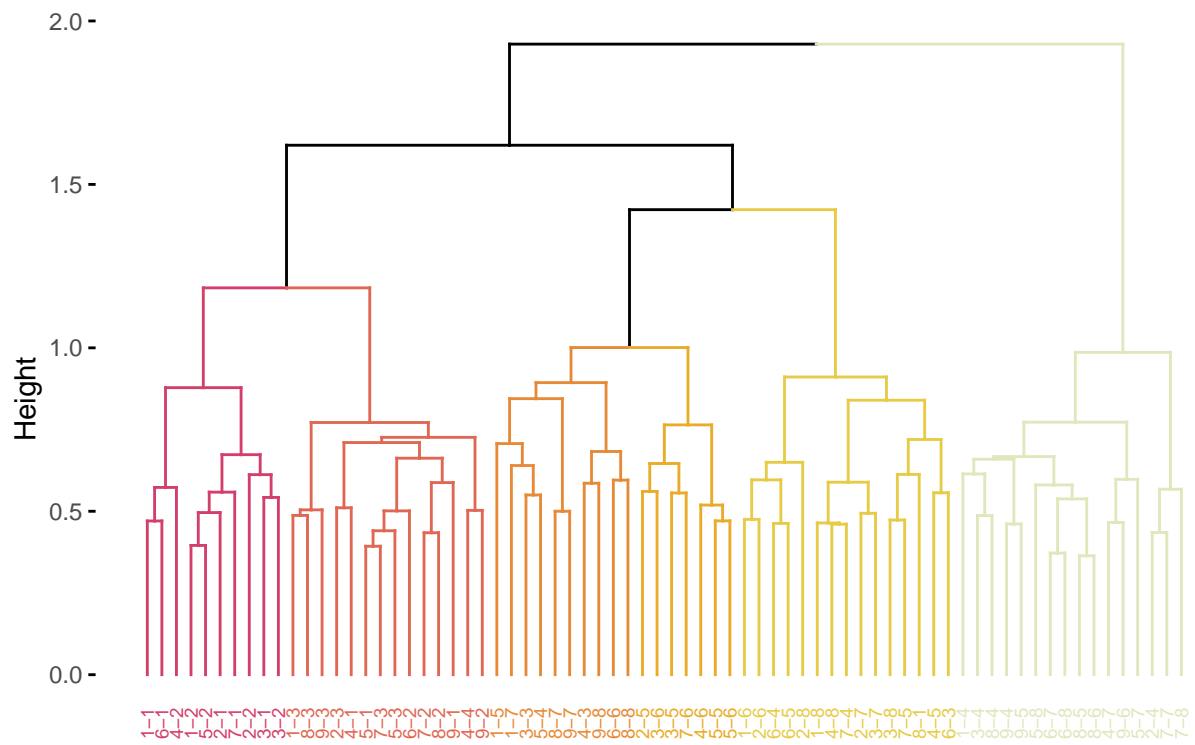
## [[29]]

Cluster Dendrogram



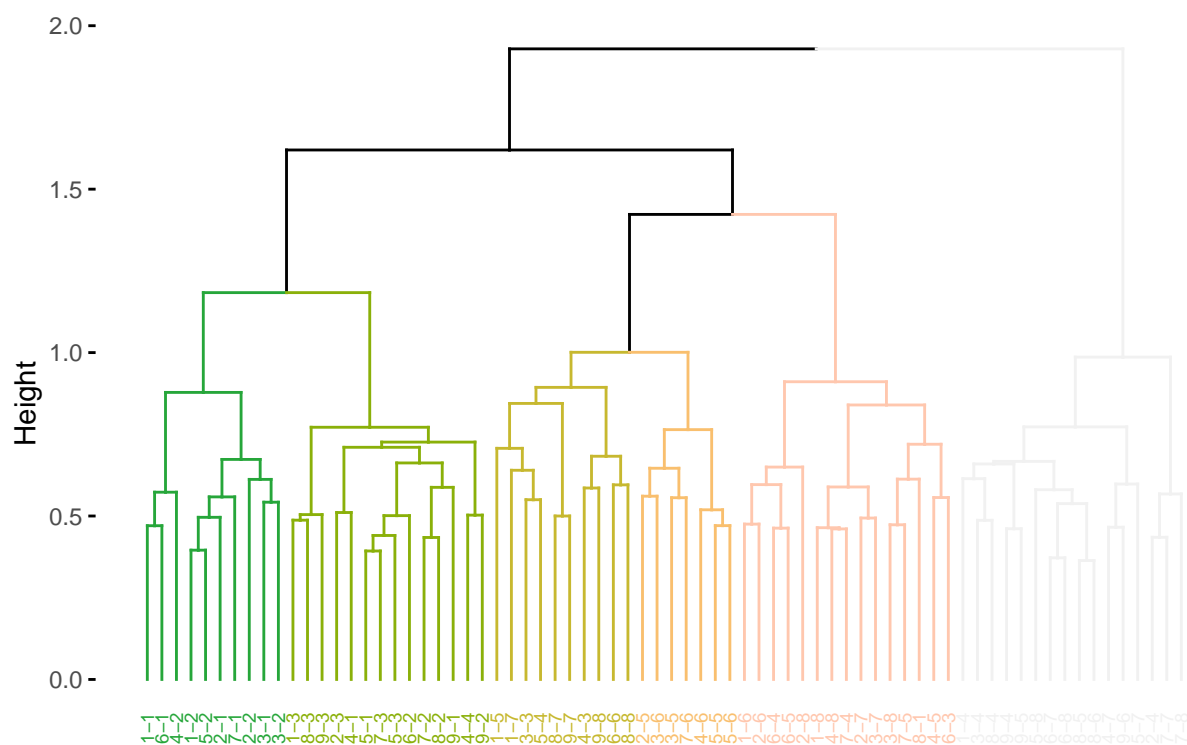
```
##  
## [[30]]
```

Cluster Dendrogram



```
##  
## [[31]]
```

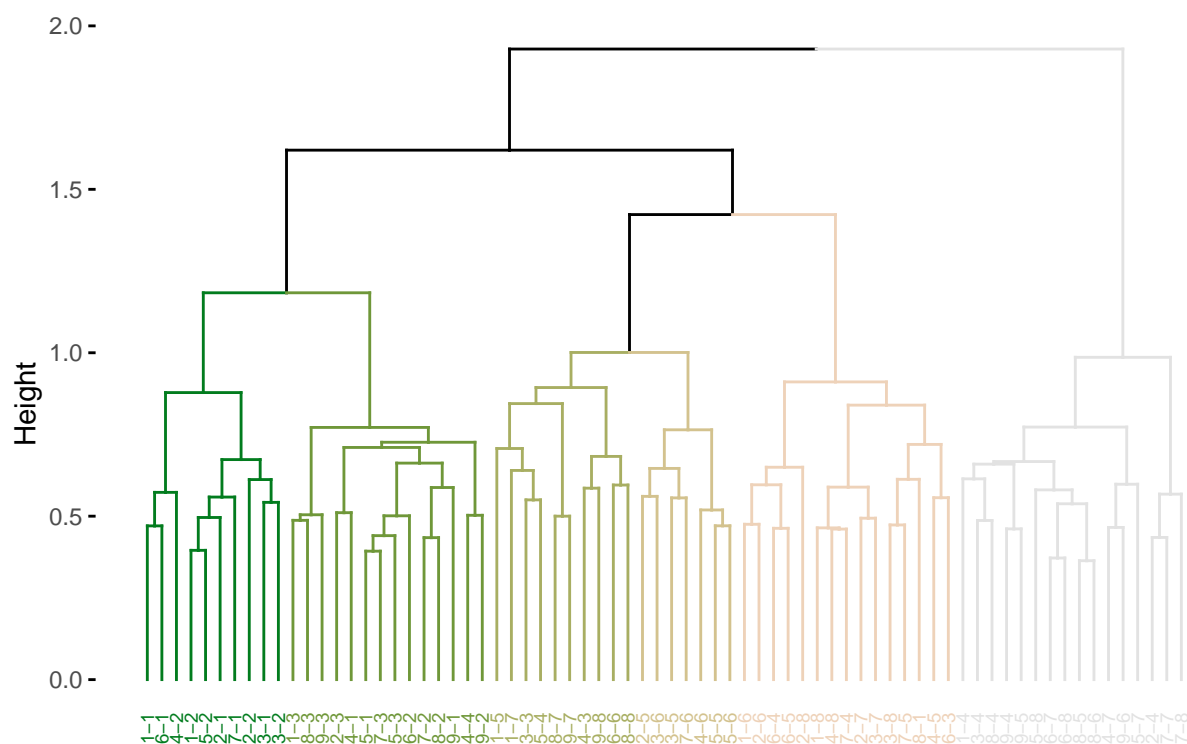
# Cluster Dendrogram



##

## [[32]]

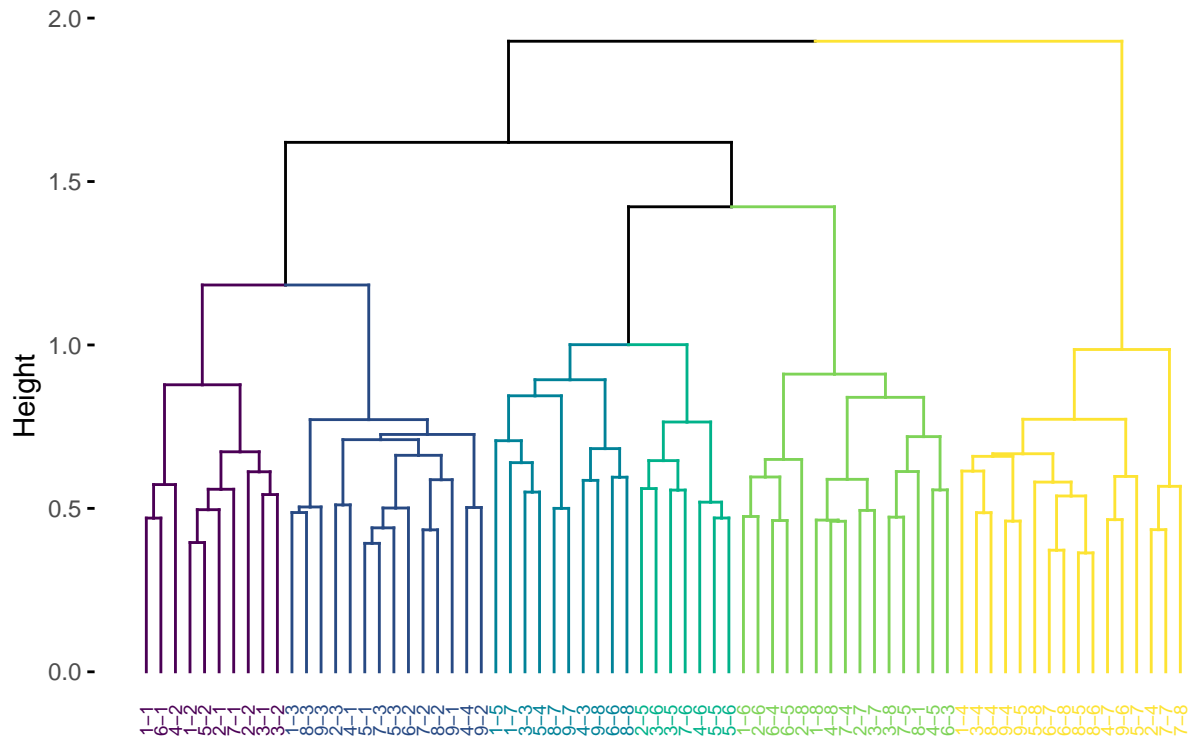
# Cluster Dendrogram



##

## [[33]]

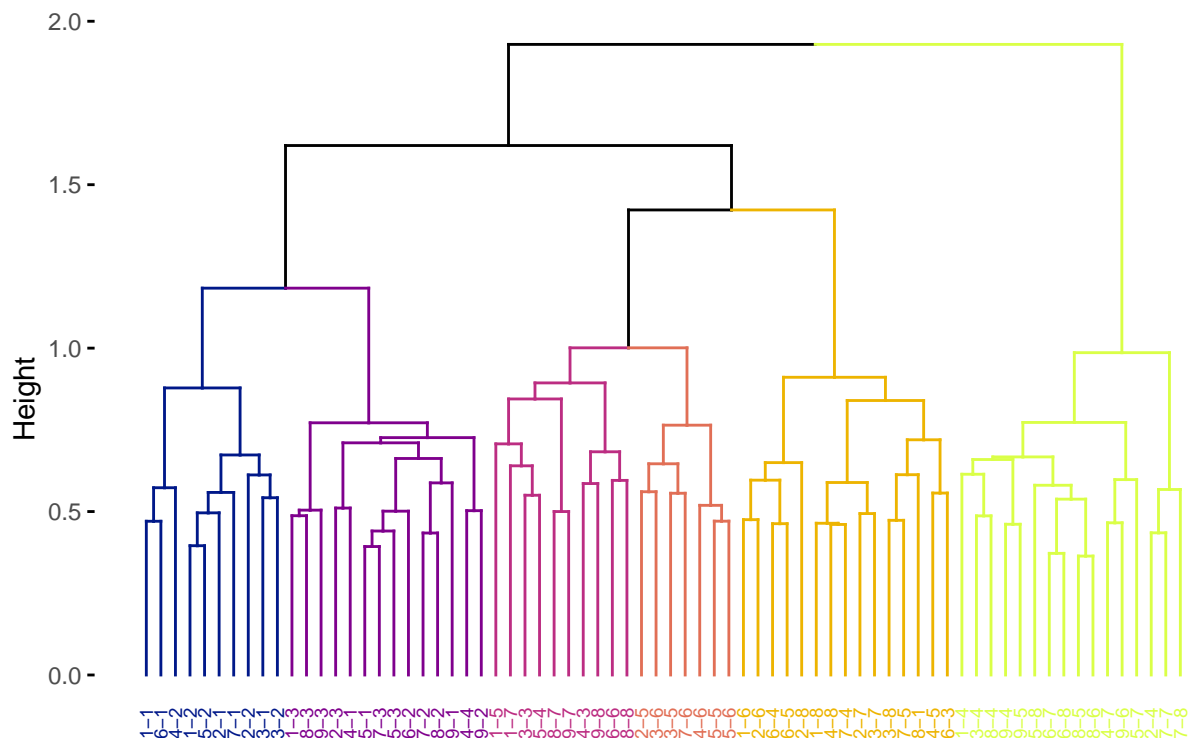
Cluster Dendrogram



##

## [[34]]

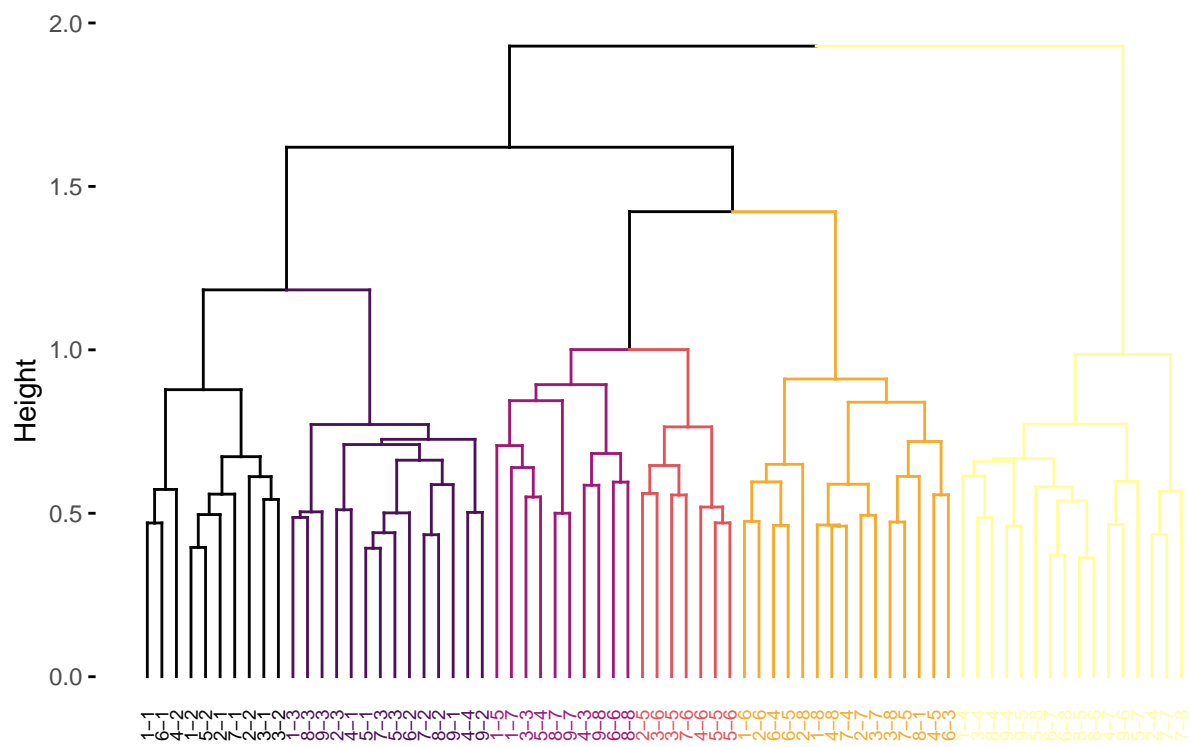
Cluster Dendrogram



##

## [[35]]

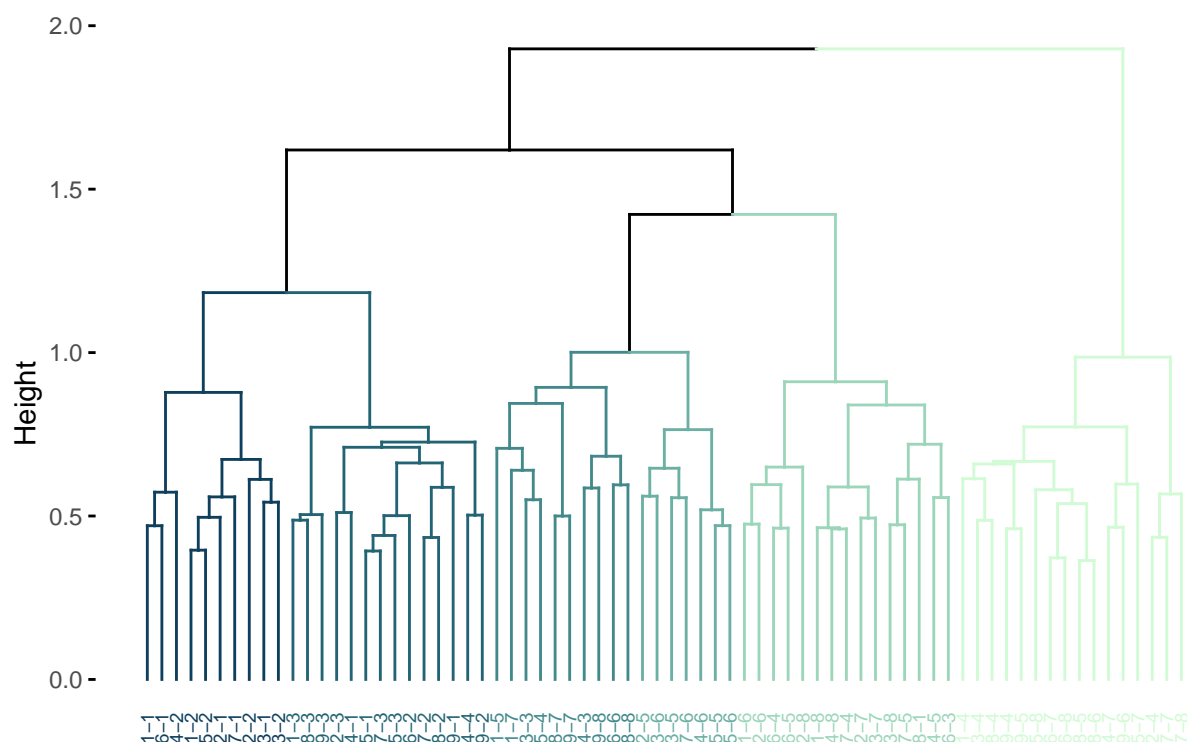
Cluster Dendrogram



##

## [[36]]

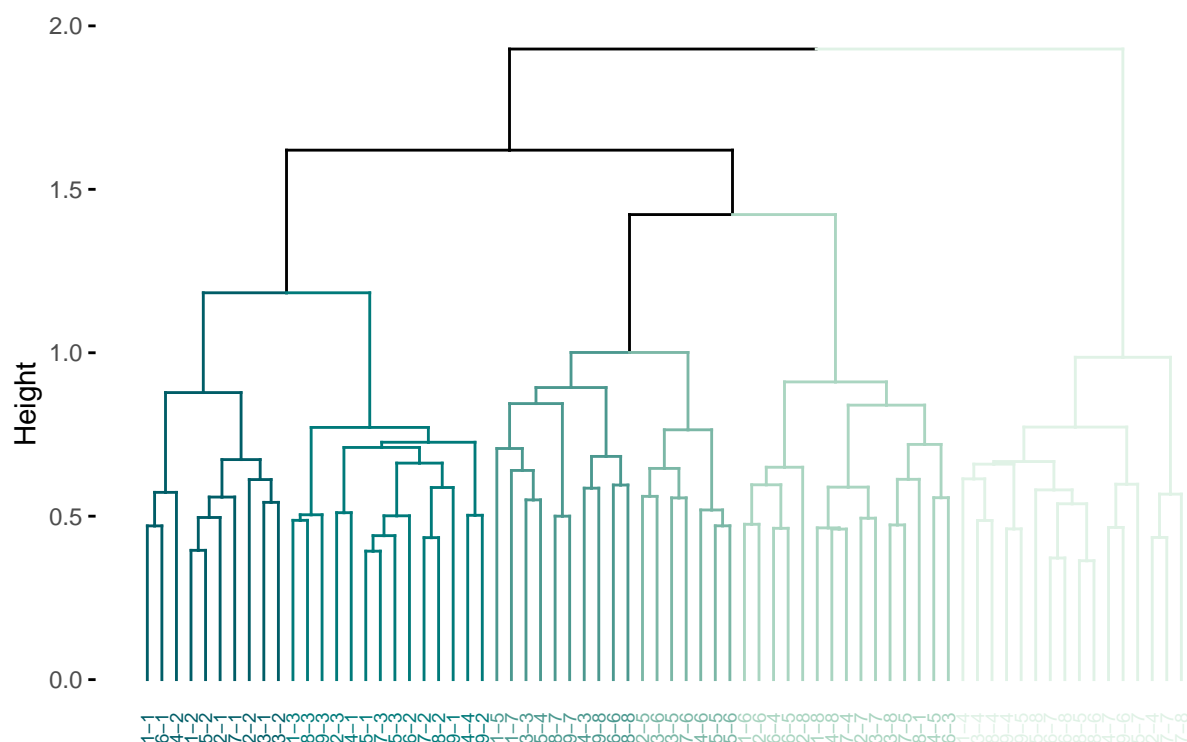
Cluster Dendrogram



##

## [[37]]

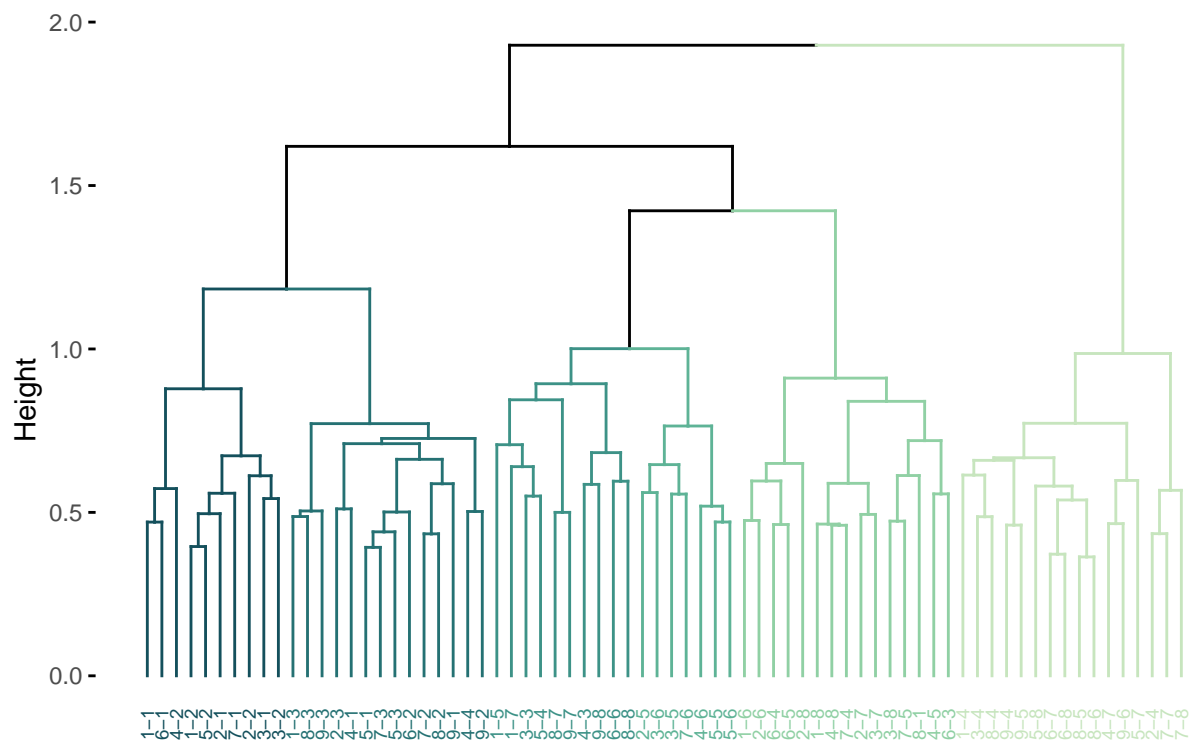
Cluster Dendrogram



##

## [[38]]

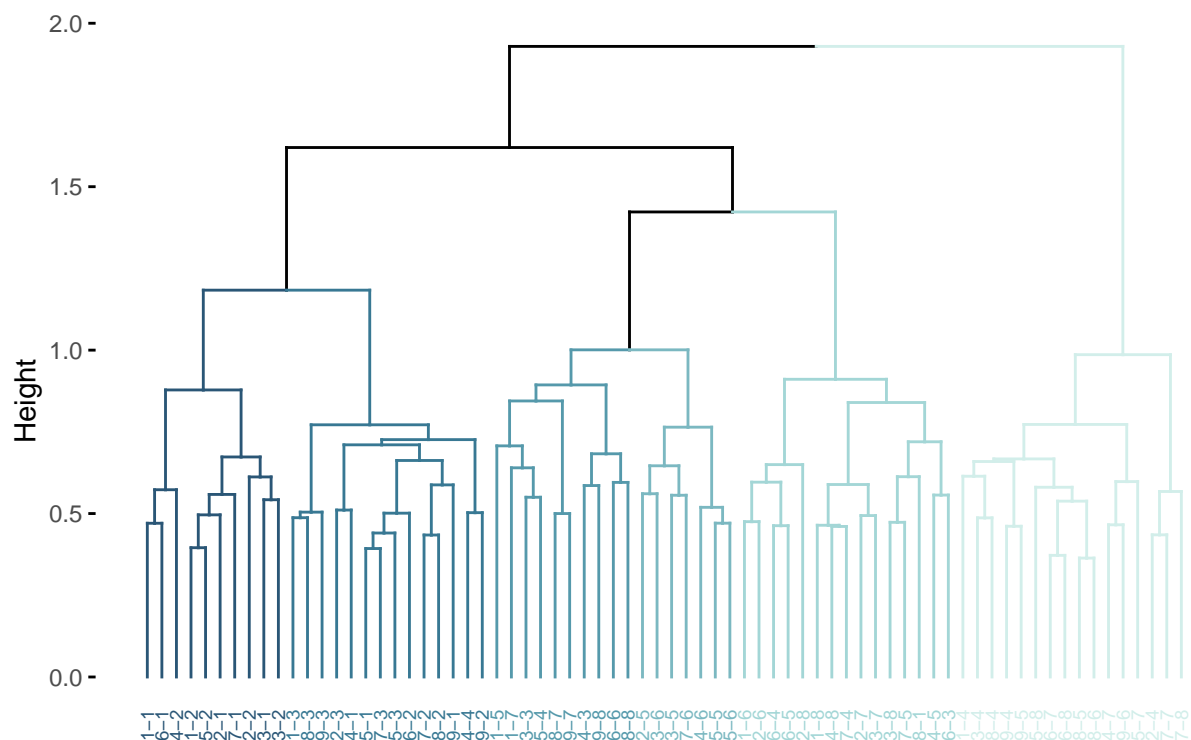
Cluster Dendrogram



##

## [[39]]

Cluster Dendrogram

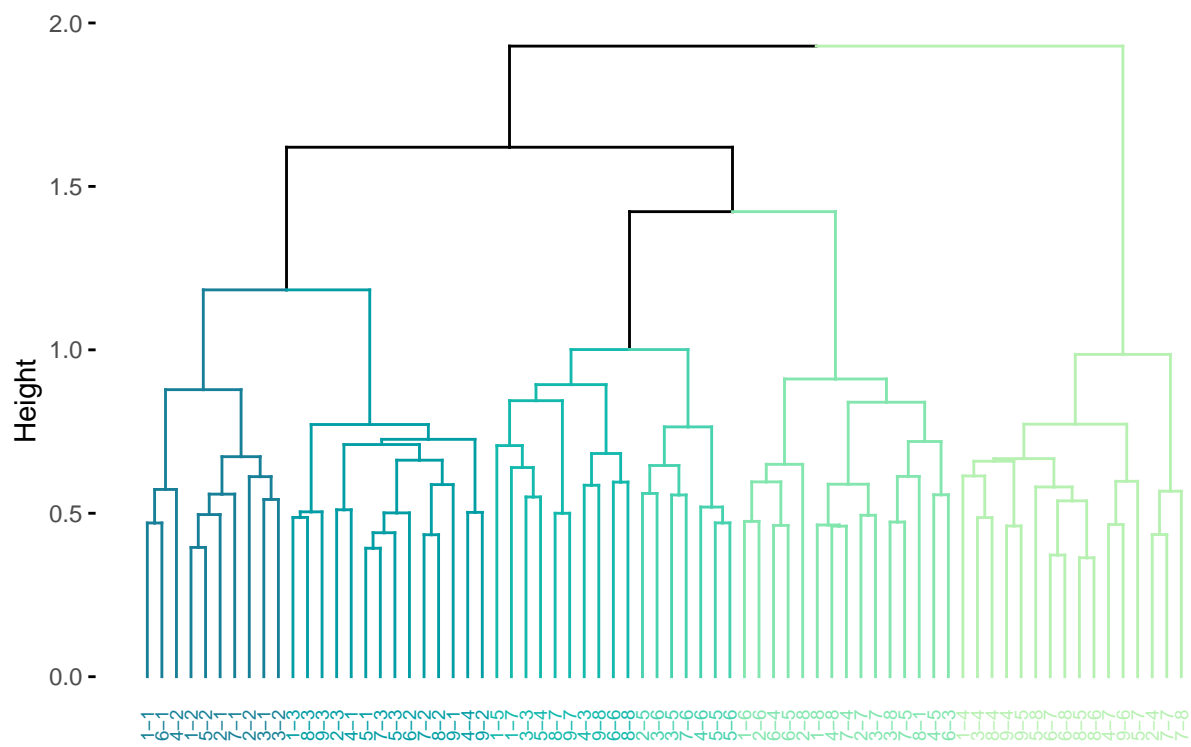




##

## [[40]]

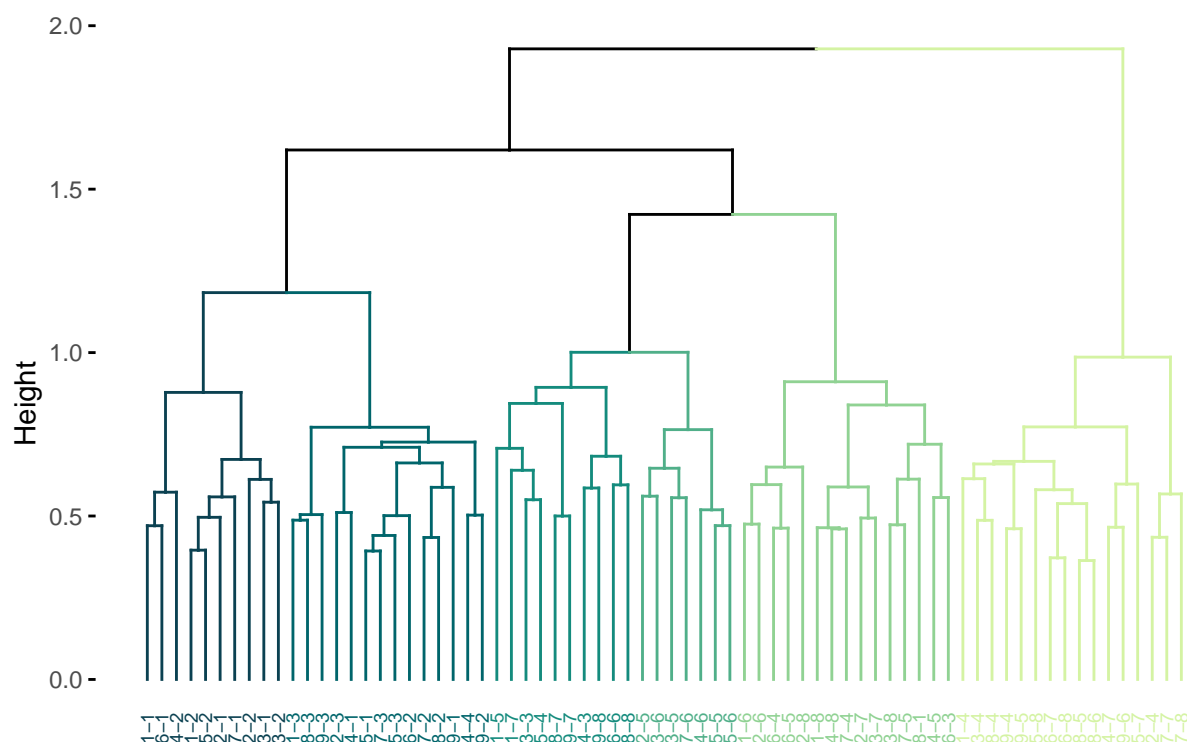
Cluster Dendrogram



##

## [[41]]

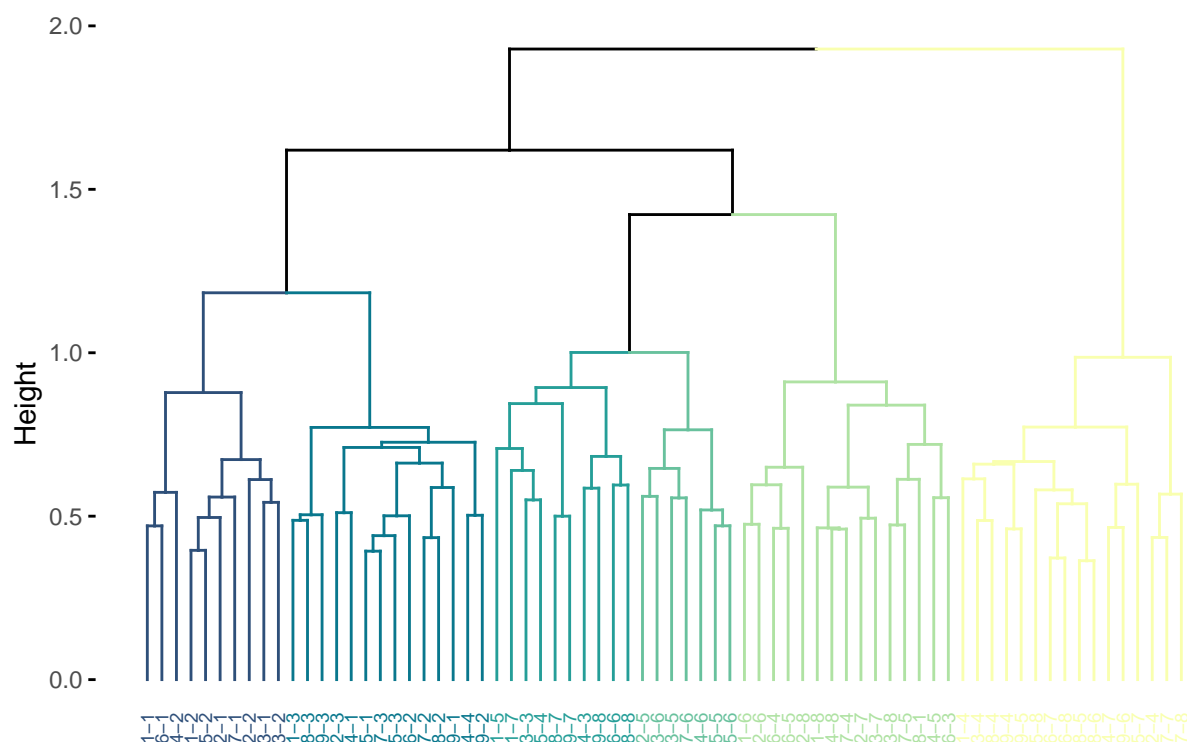
# Cluster Dendrogram



##

## [[42]]

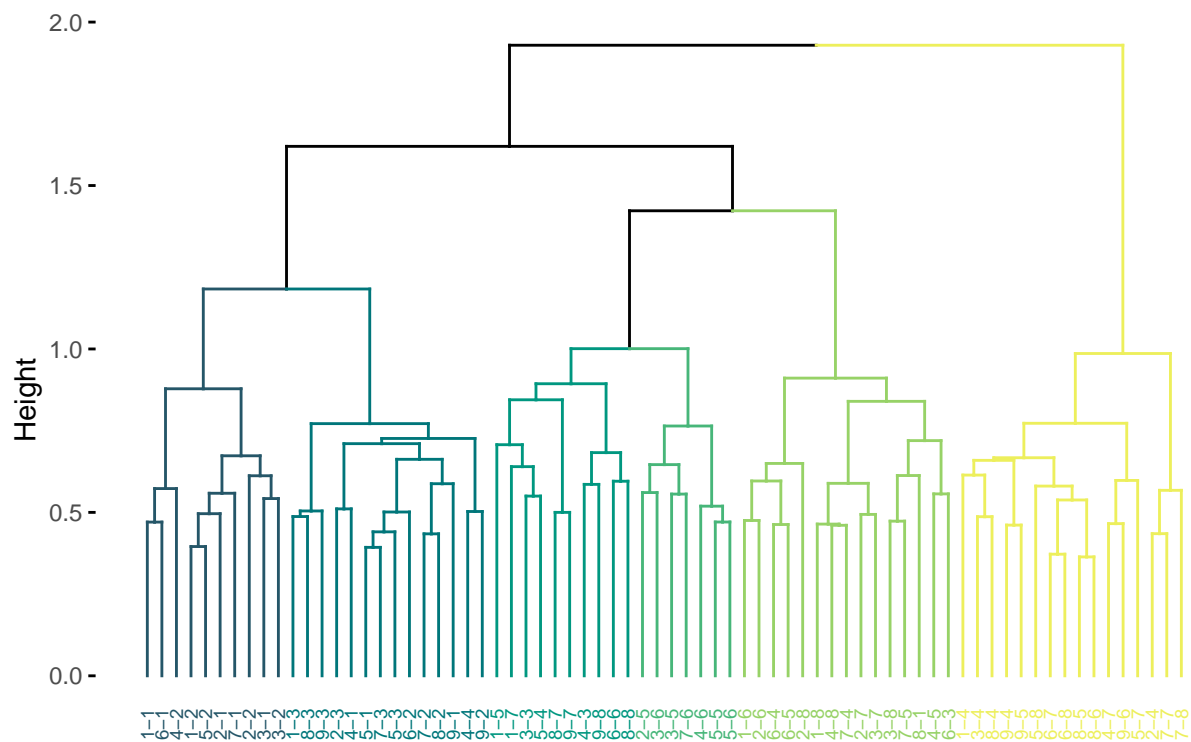
# Cluster Dendrogram



##

## [[43]]

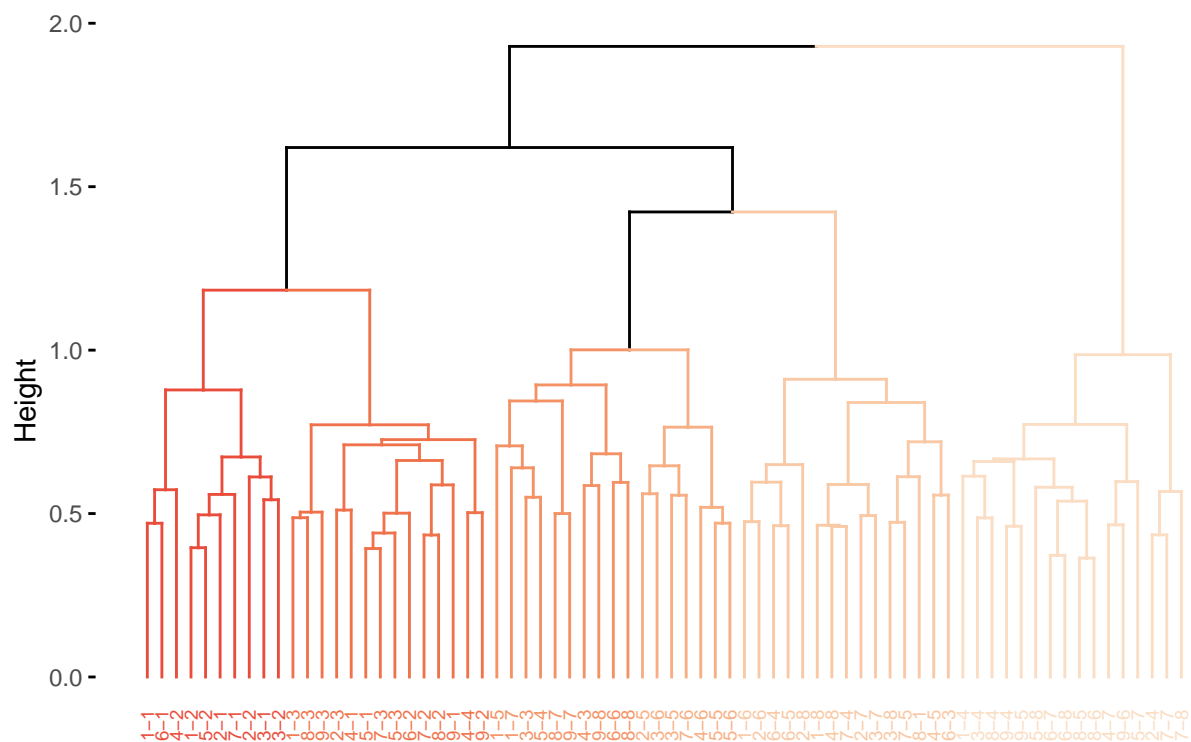
Cluster Dendrogram



##

## [[44]]

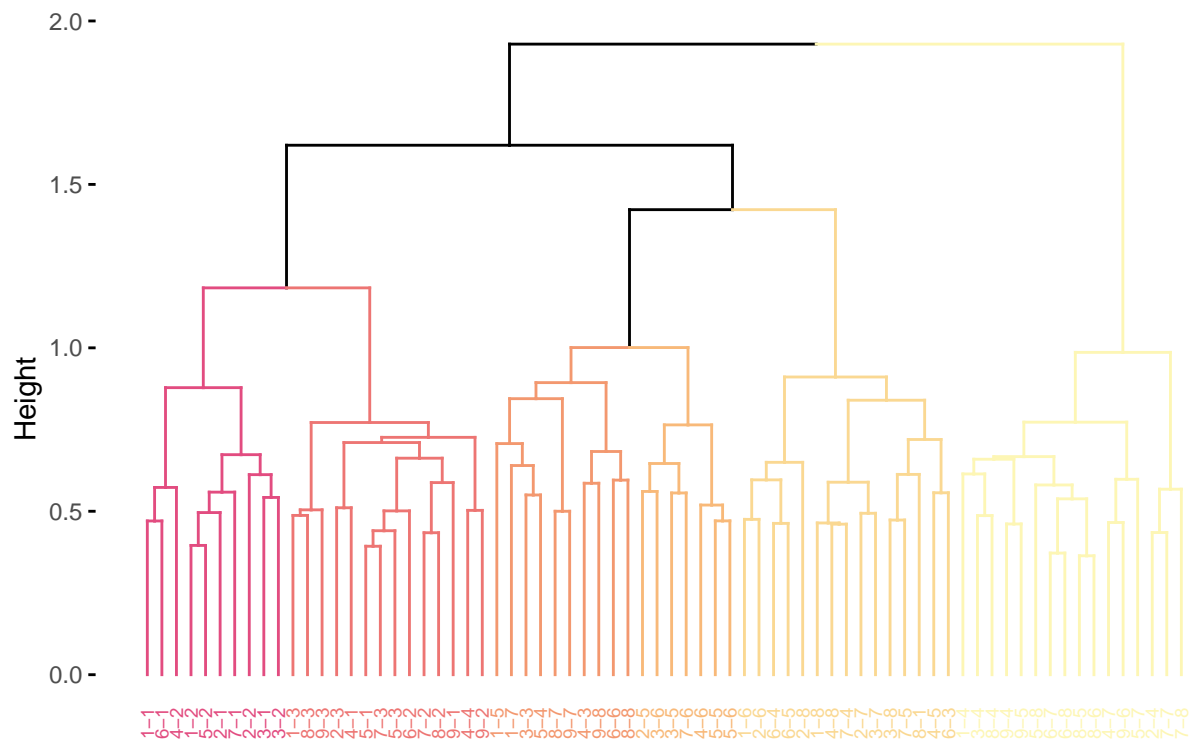
Cluster Dendrogram



##

## [[45]]

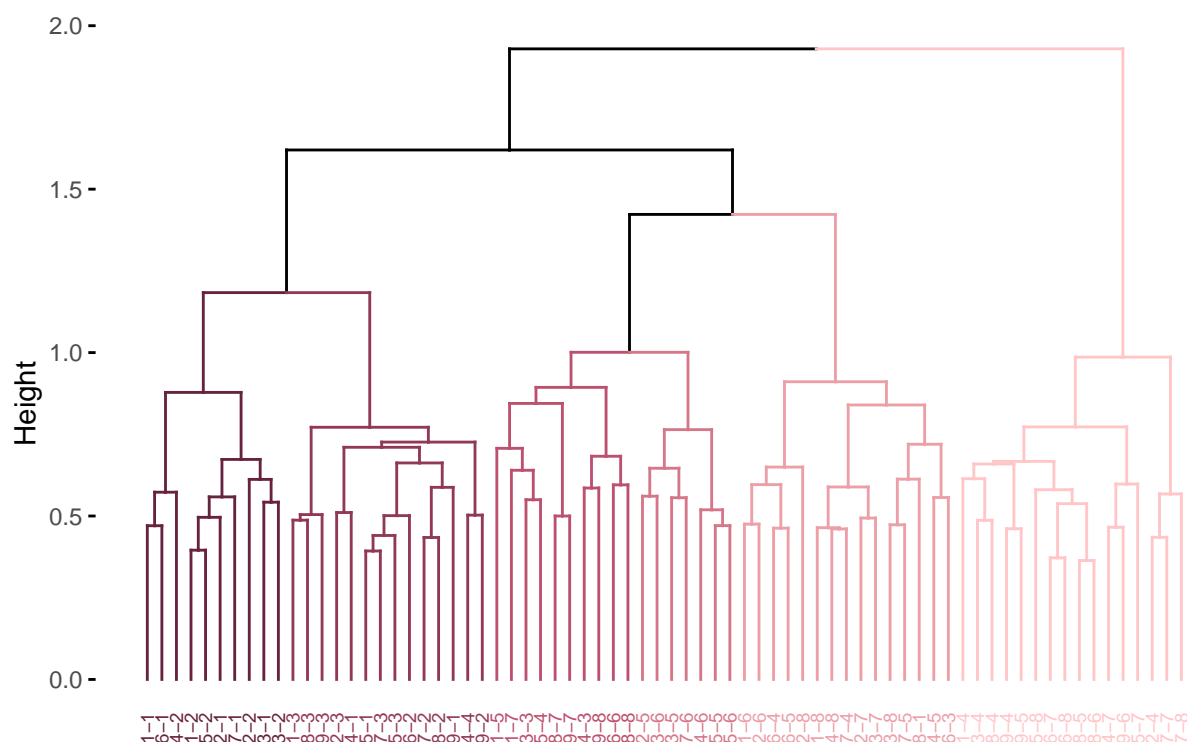
Cluster Dendrogram



##

## [[46]]

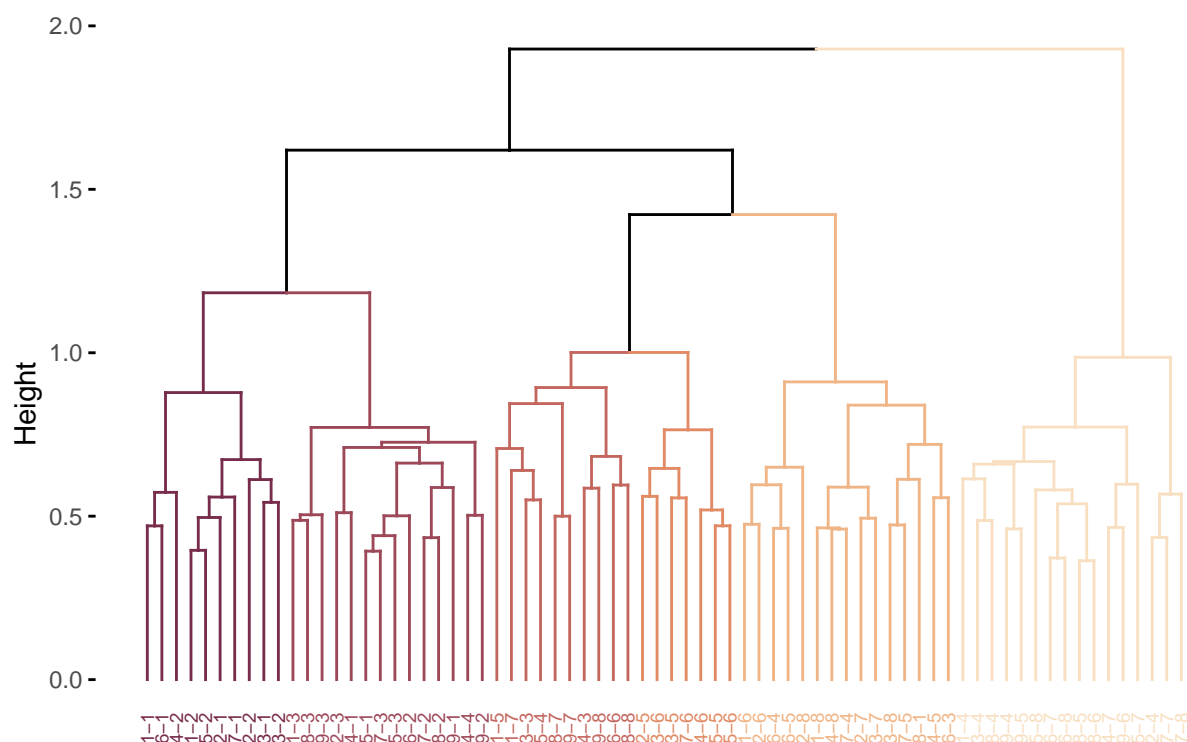
### Cluster Dendrogram



##

## [[47]]

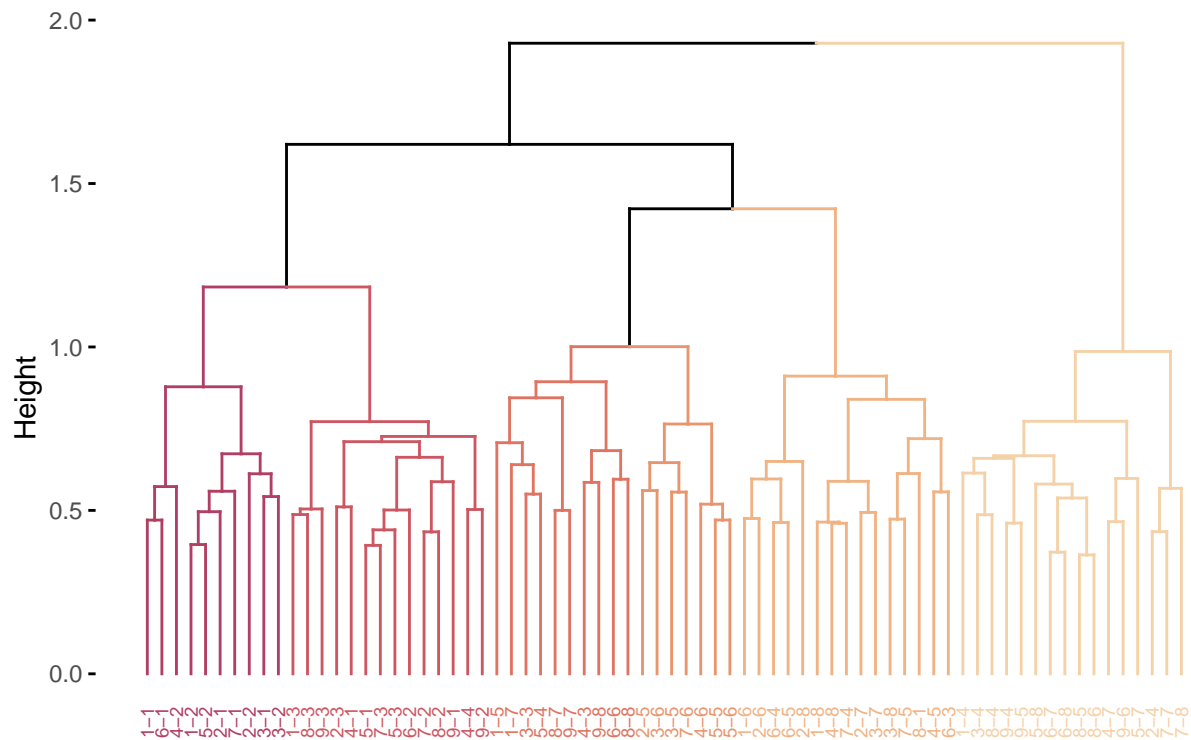
### Cluster Dendrogram



##

## [[48]]

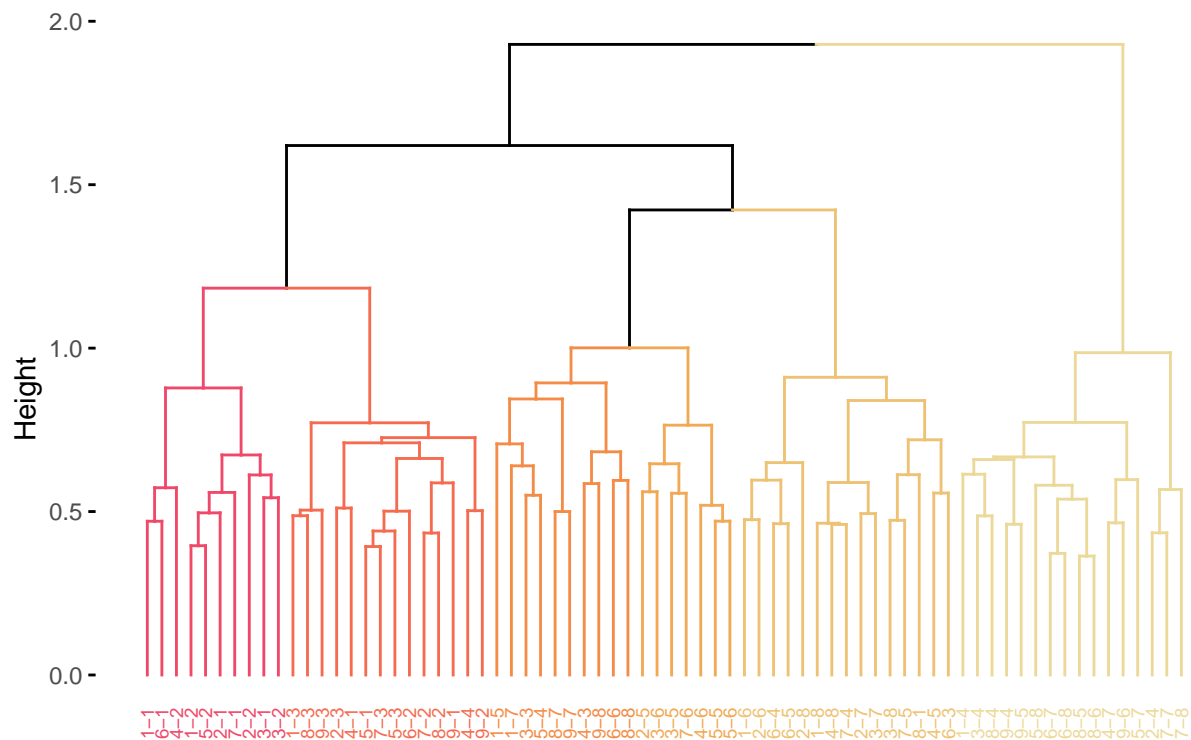
Cluster Dendrogram



##

## [[49]]

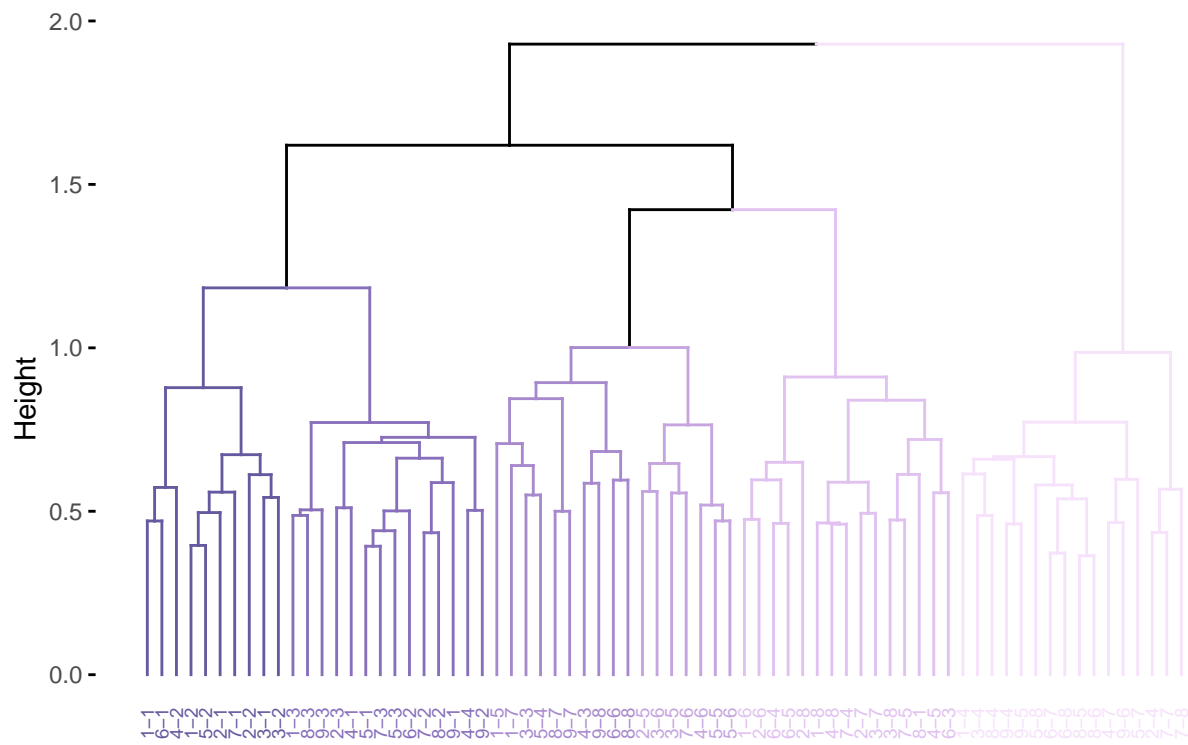
Cluster Dendrogram



##

## [[50]]

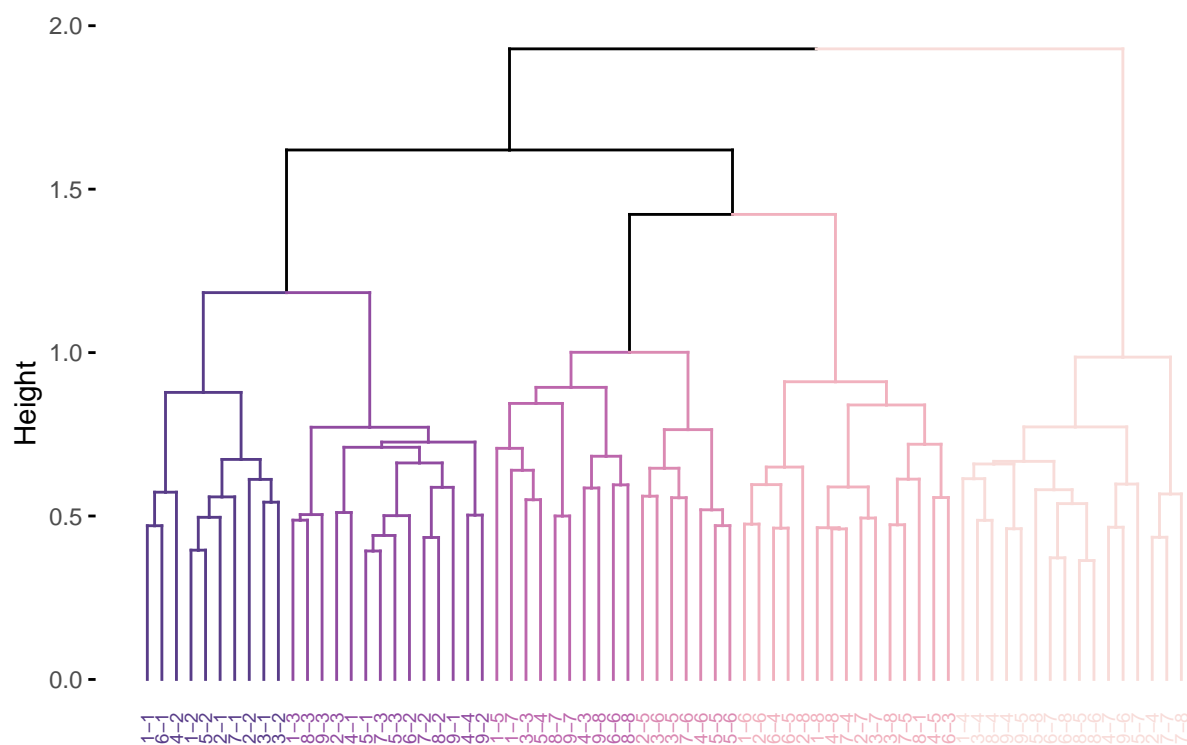
Cluster Dendrogram



##

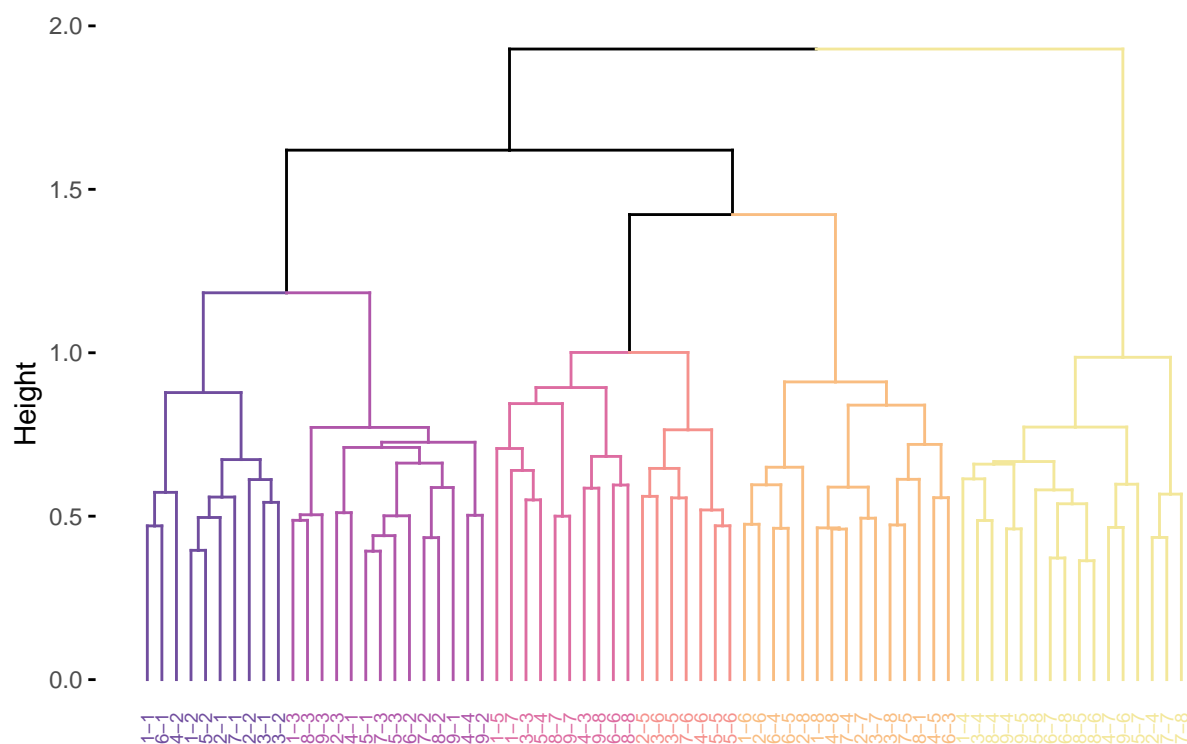
## [[51]]

# Cluster Dendrogram



##  
## [[52]]

# Cluster Dendrogram

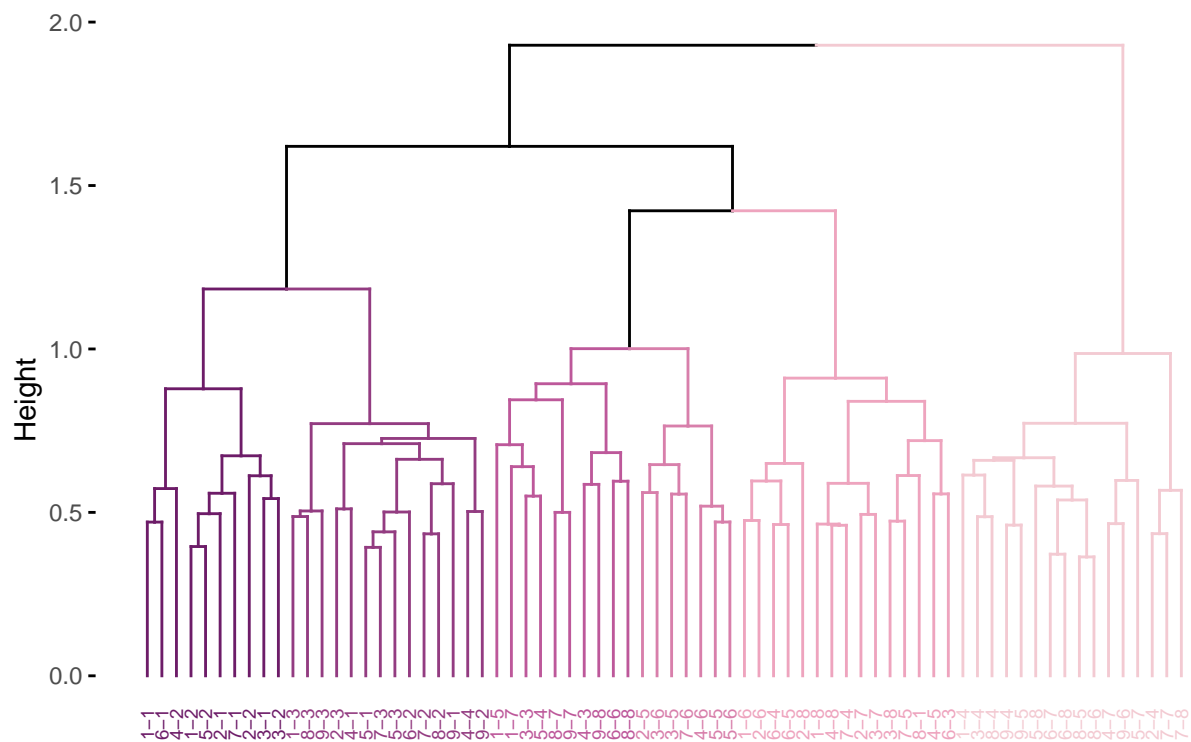


##



## [[53]]

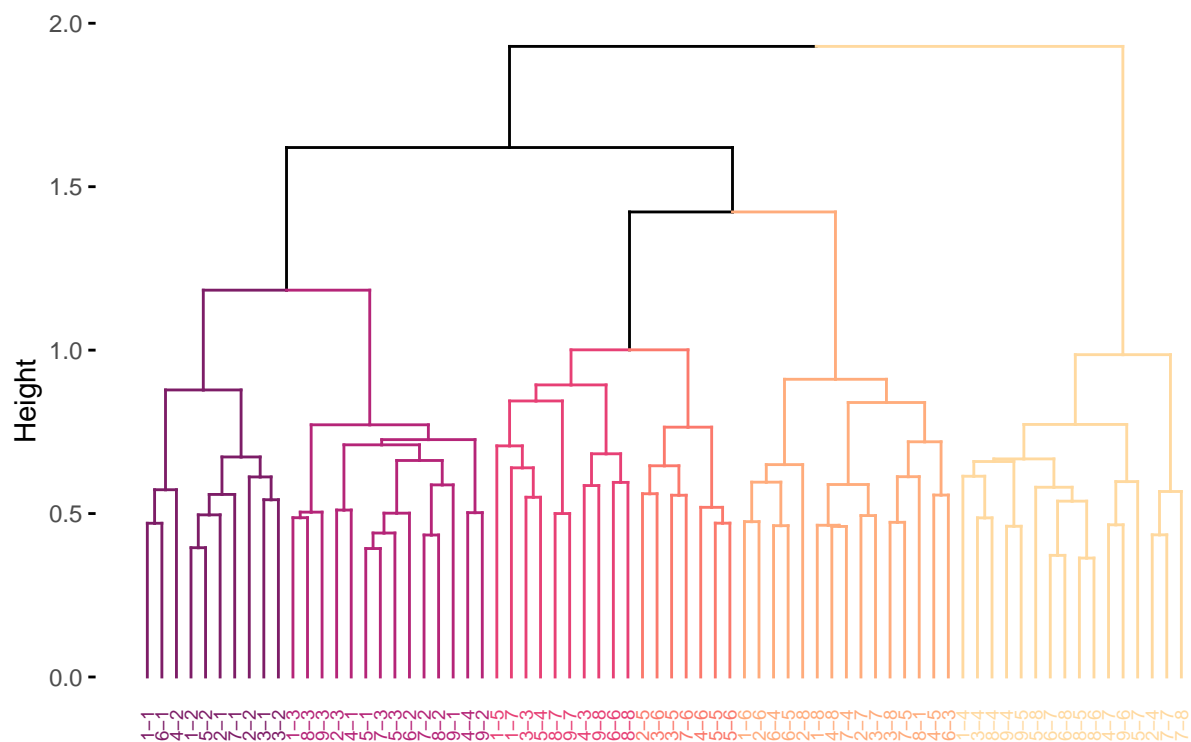
Cluster Dendrogram



##

## [[54]]

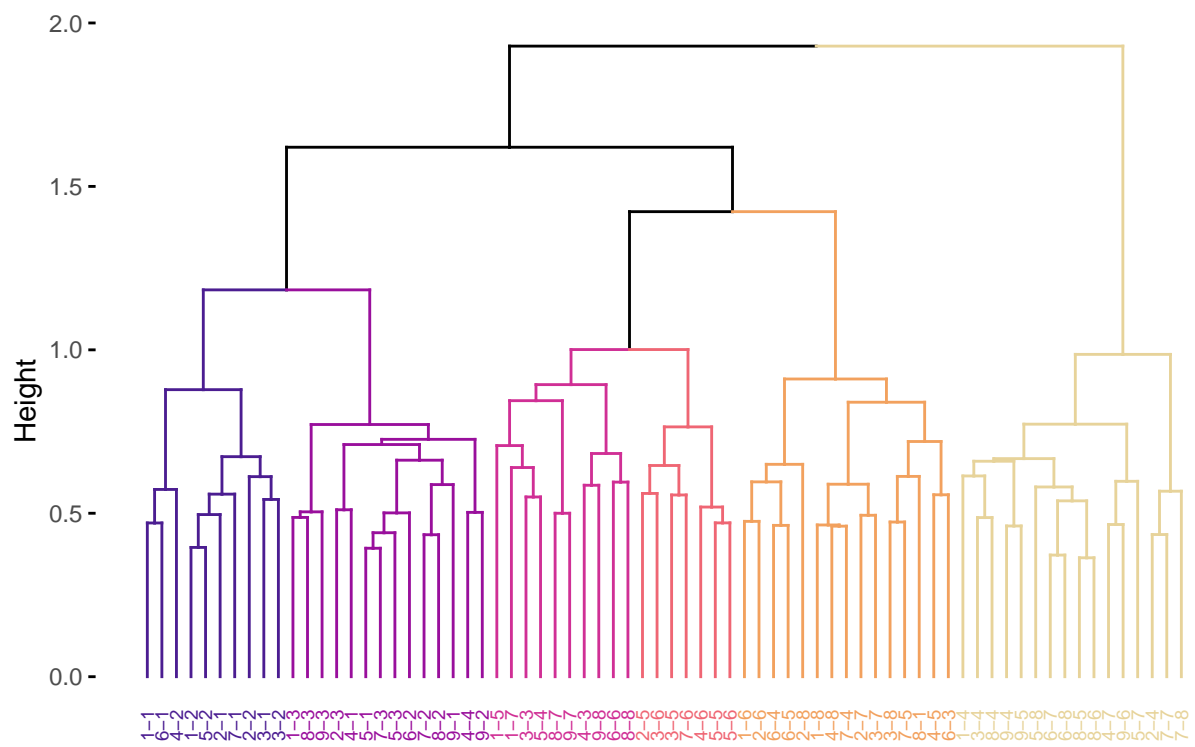
Cluster Dendrogram



##

## [[55]]

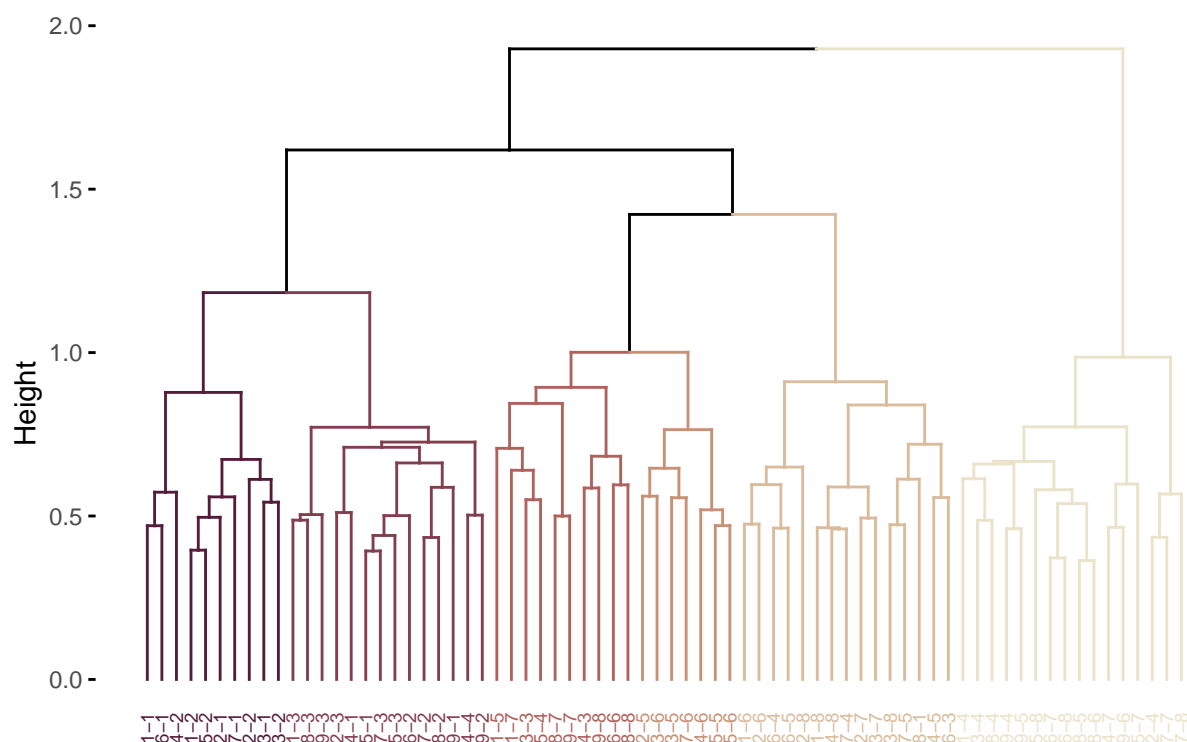
### Cluster Dendrogram



##

## [[56]]

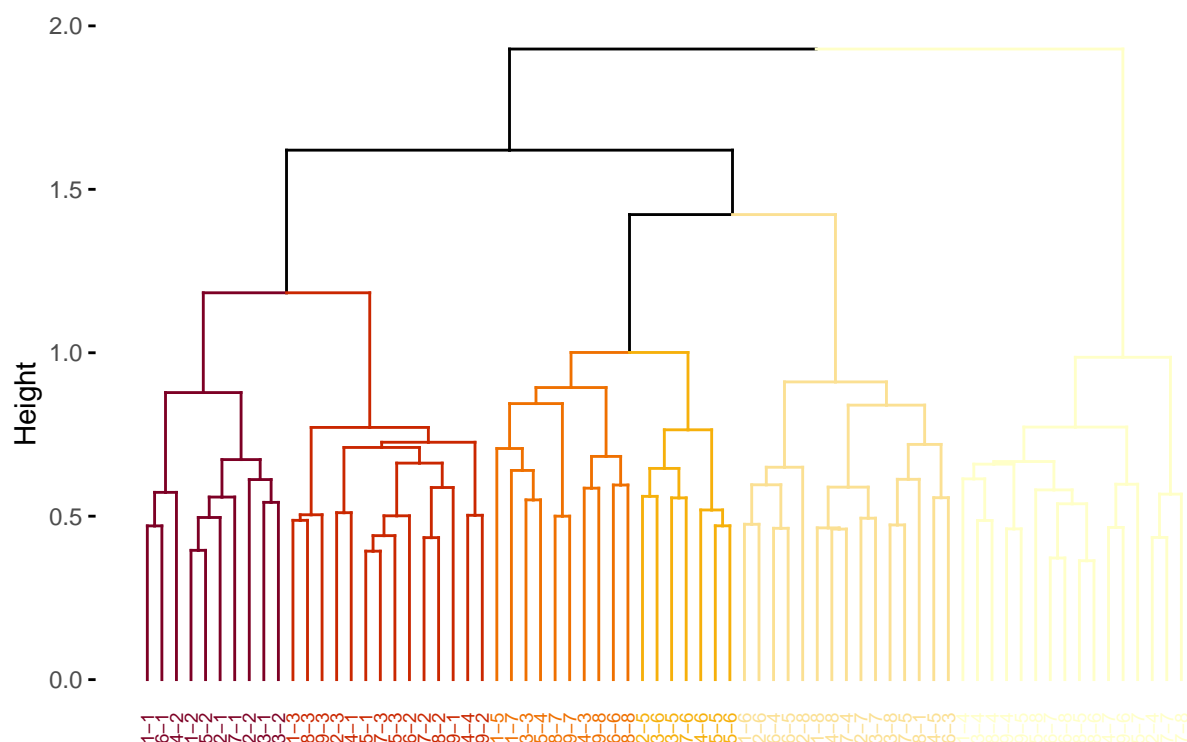
# Cluster Dendrogram



##

## [[57]]

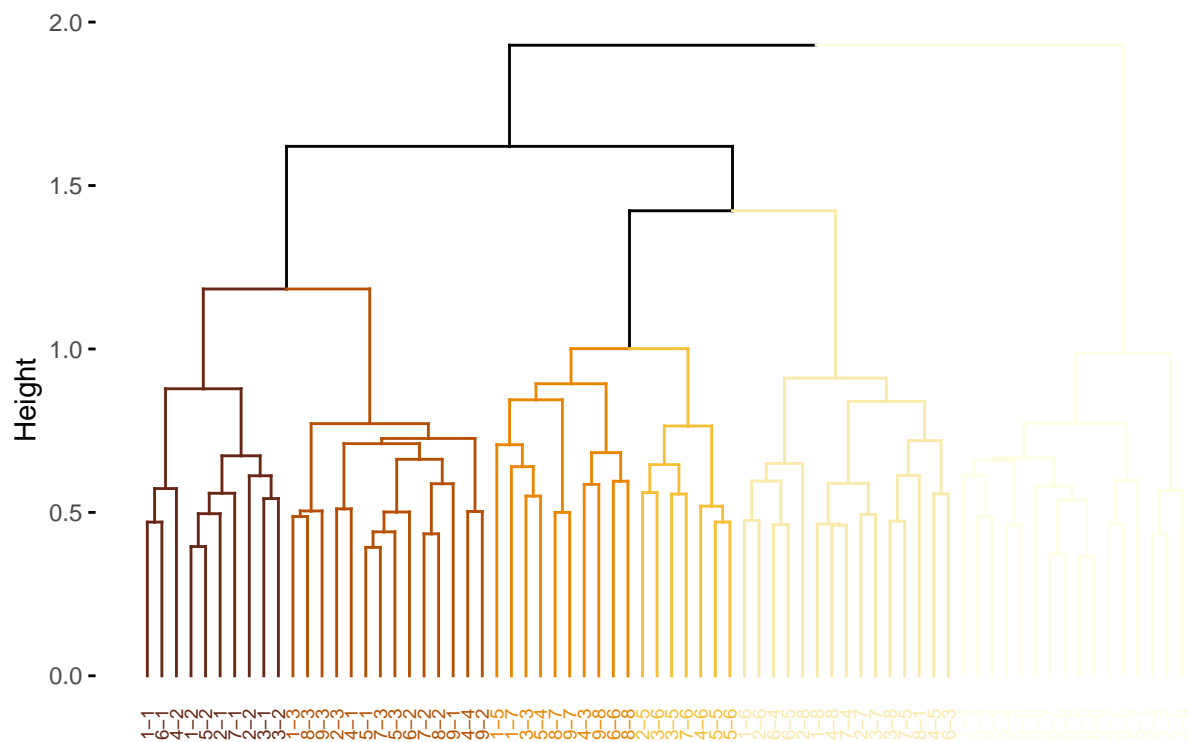
# Cluster Dendrogram



##

## [[58]]

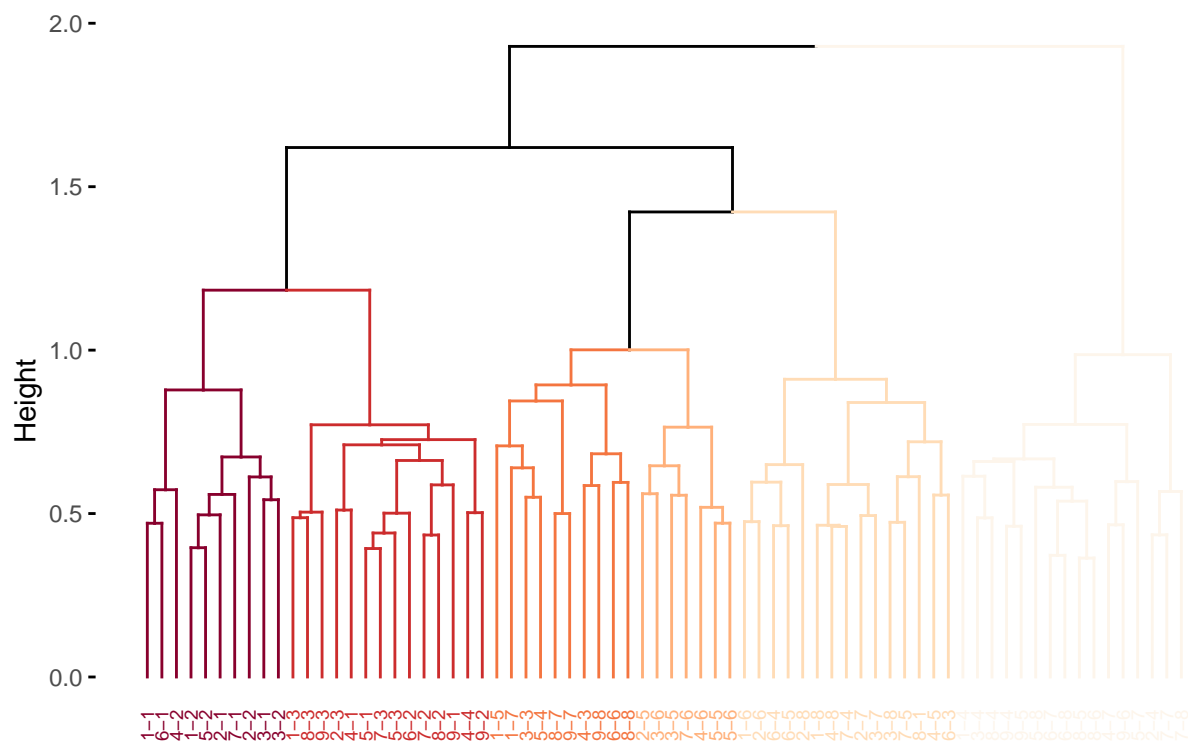
Cluster Dendrogram



##

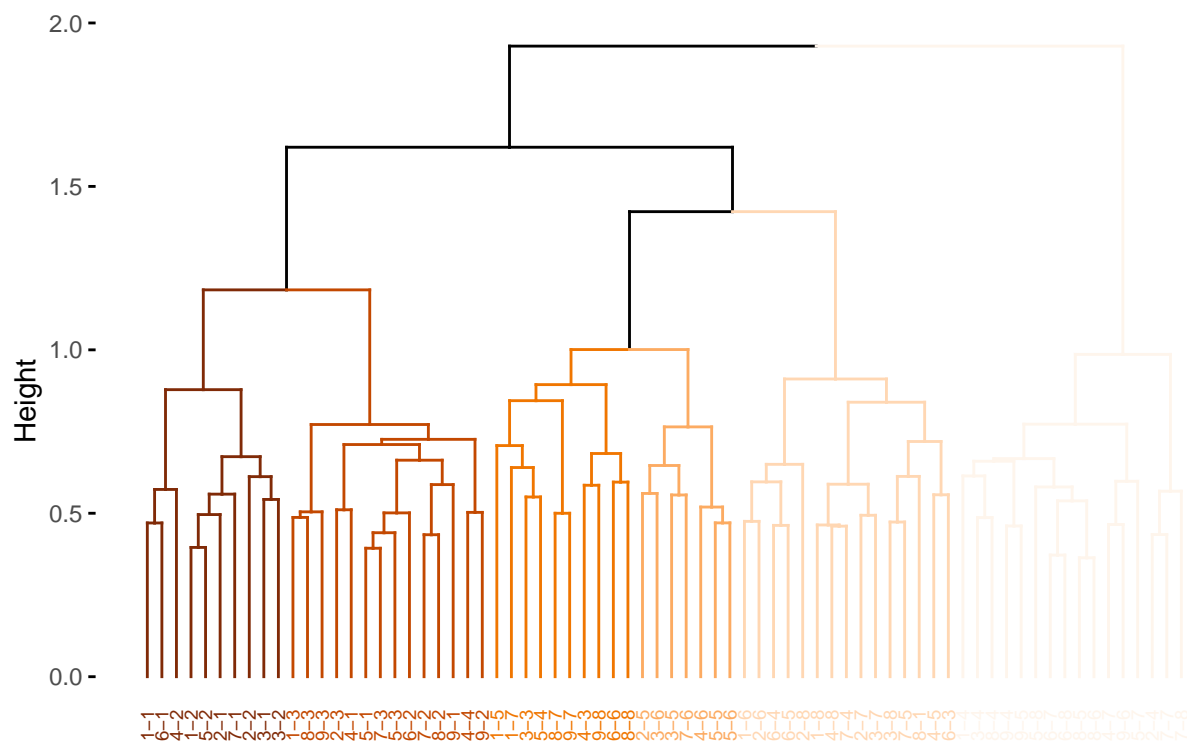
## [[59]]

Cluster Dendrogram



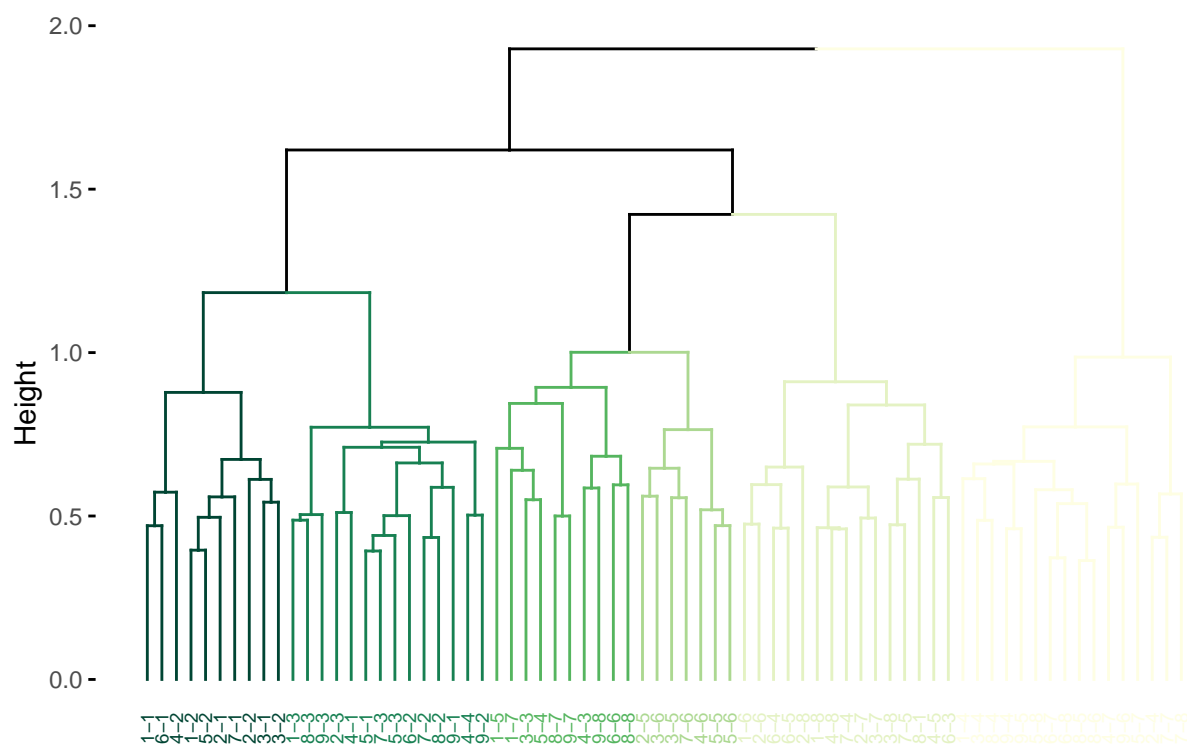
```
## [[60]]
```

### Cluster Dendrogram



```
## [[61]]
```

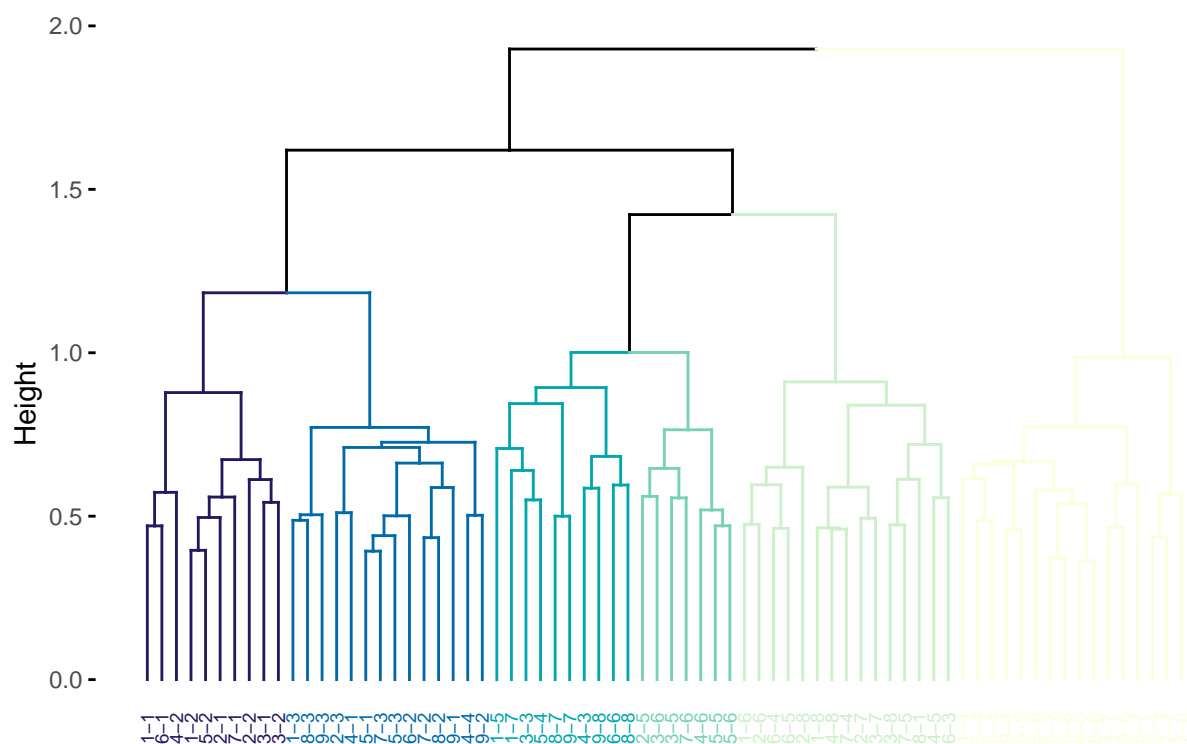
### Cluster Dendrogram



##

## [[62]]

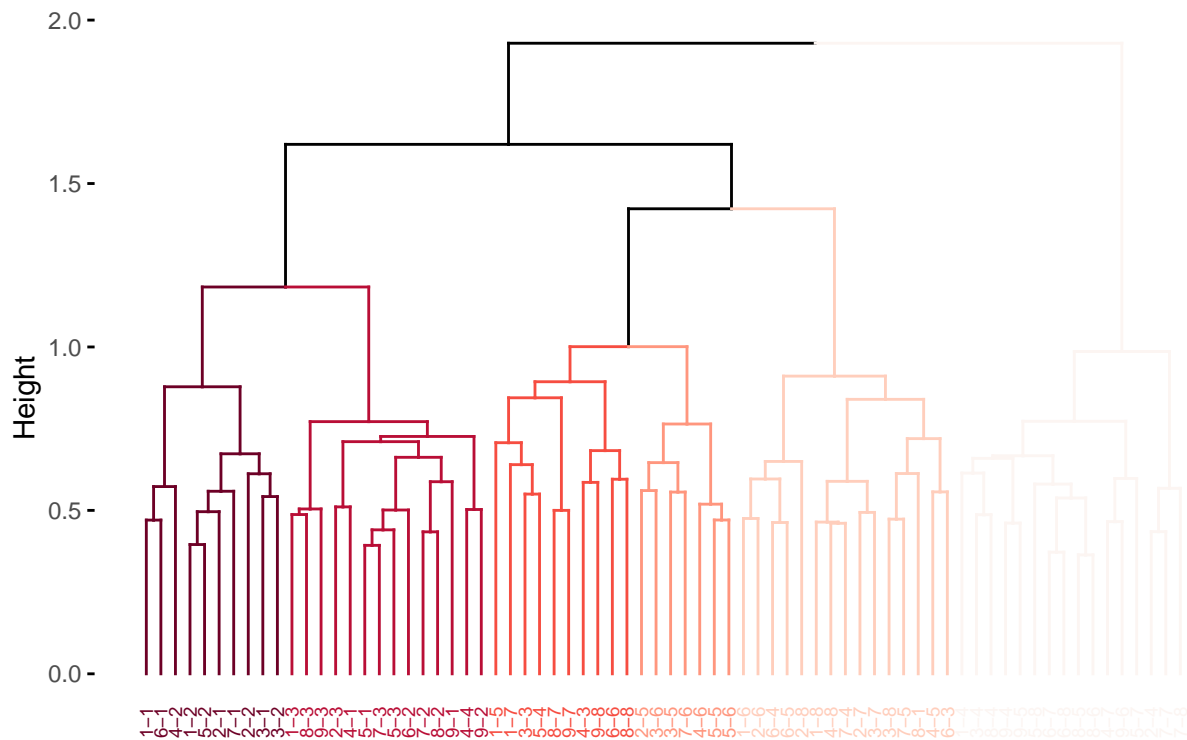
### Cluster Dendrogram



##

## [[63]]

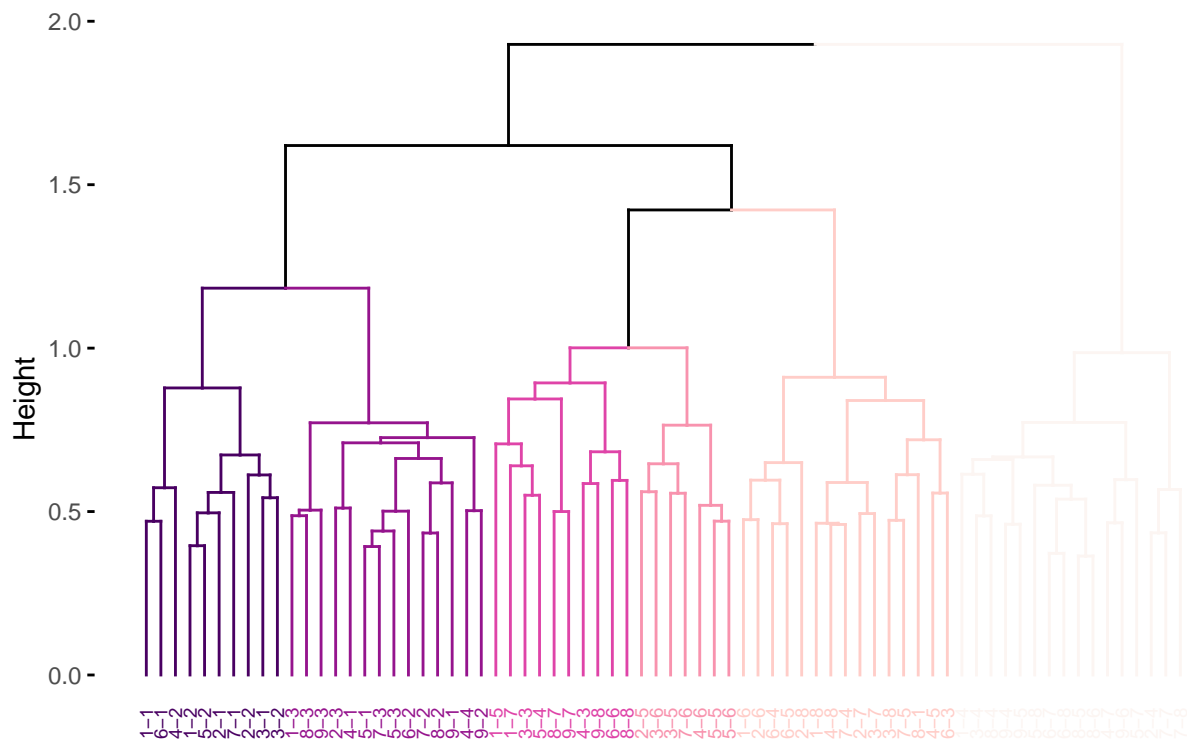
Cluster Dendrogram



##

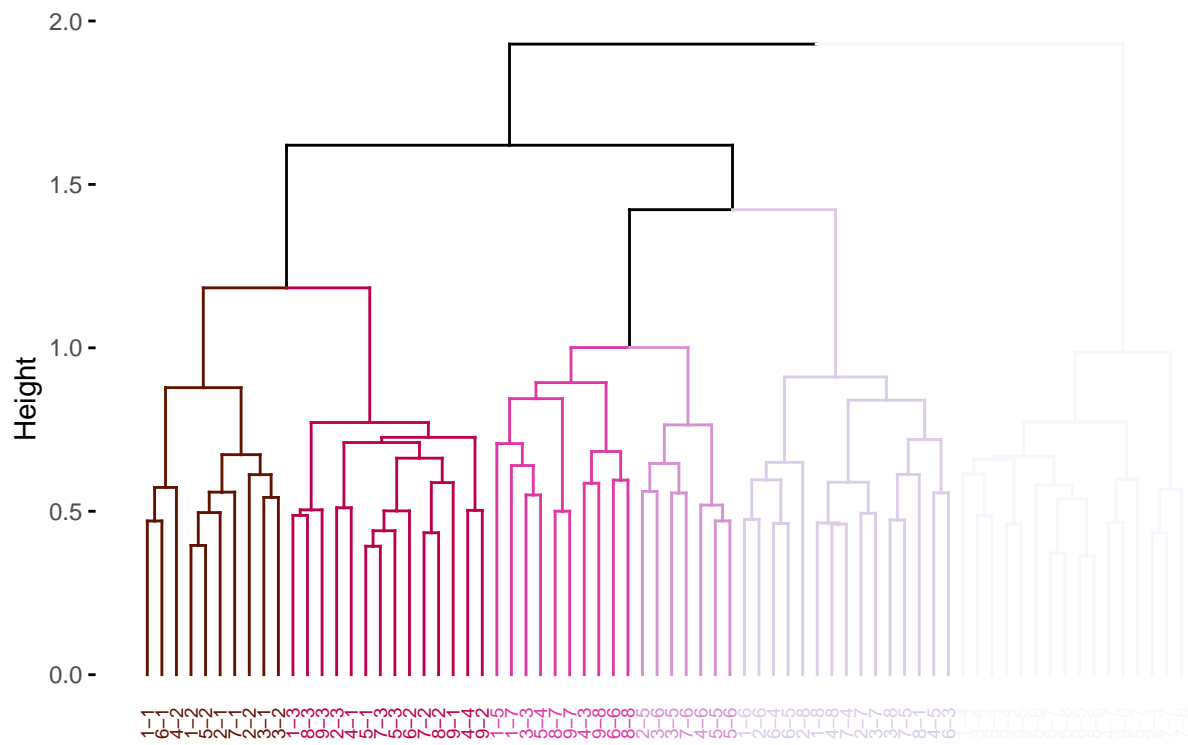
## [[64]]

Cluster Dendrogram



```
##  
## [[65]]
```

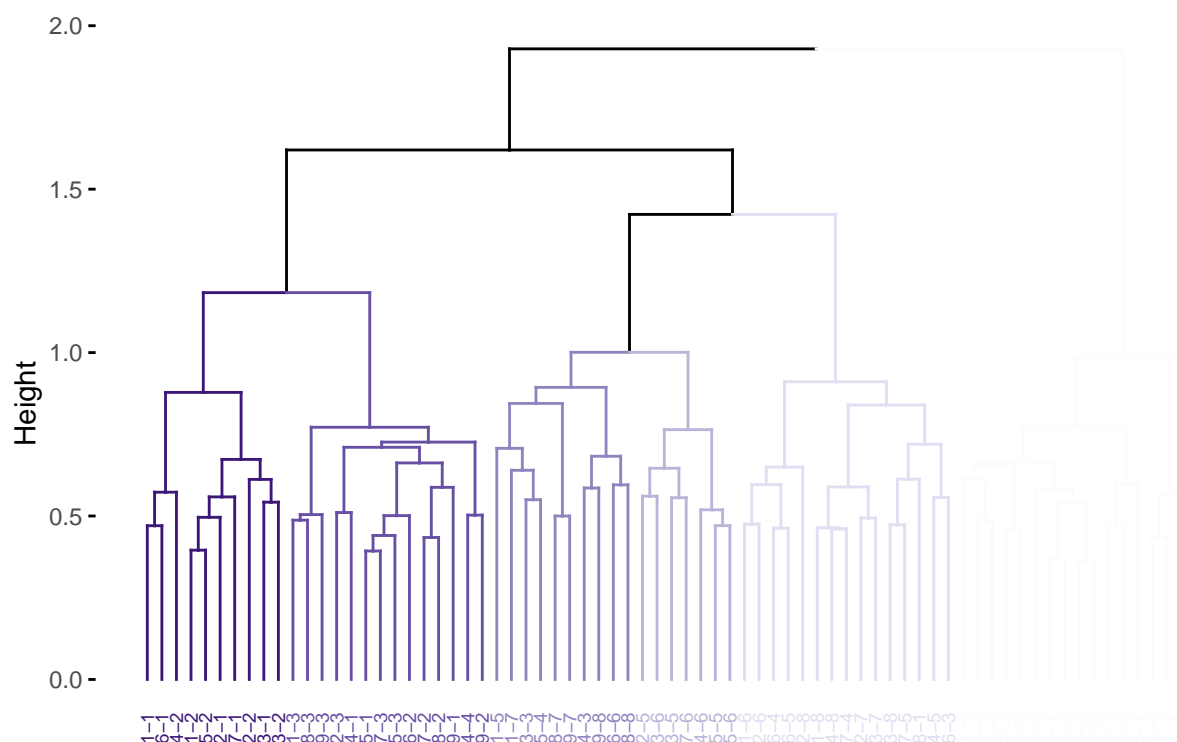
Cluster Dendrogram



```
##  
## [[66]]
```



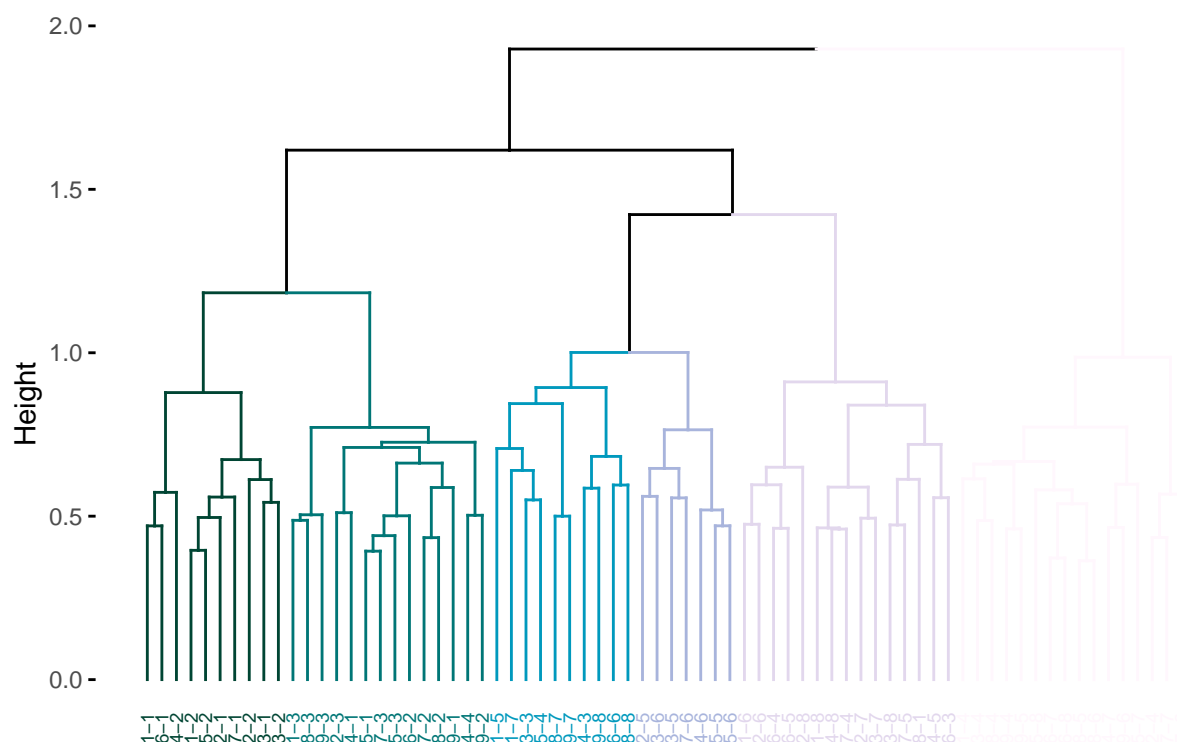
# Cluster Dendrogram



##

## [[67]]

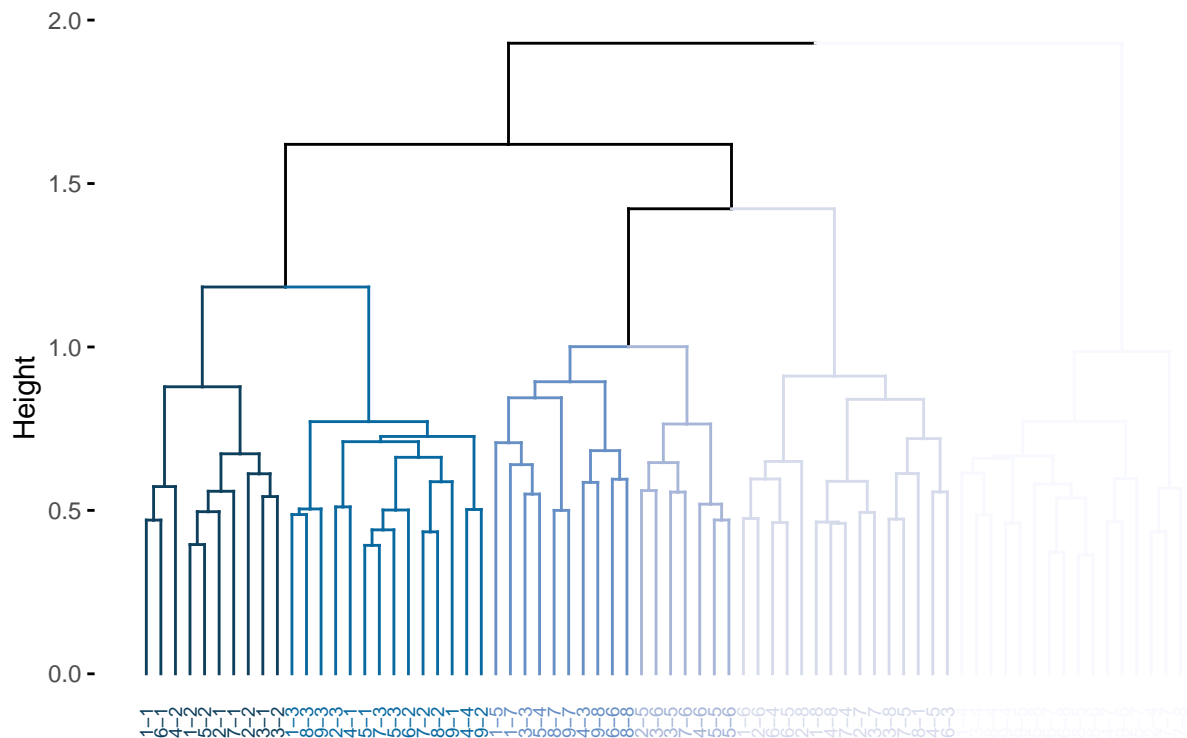
# Cluster Dendrogram



##

## [[68]]

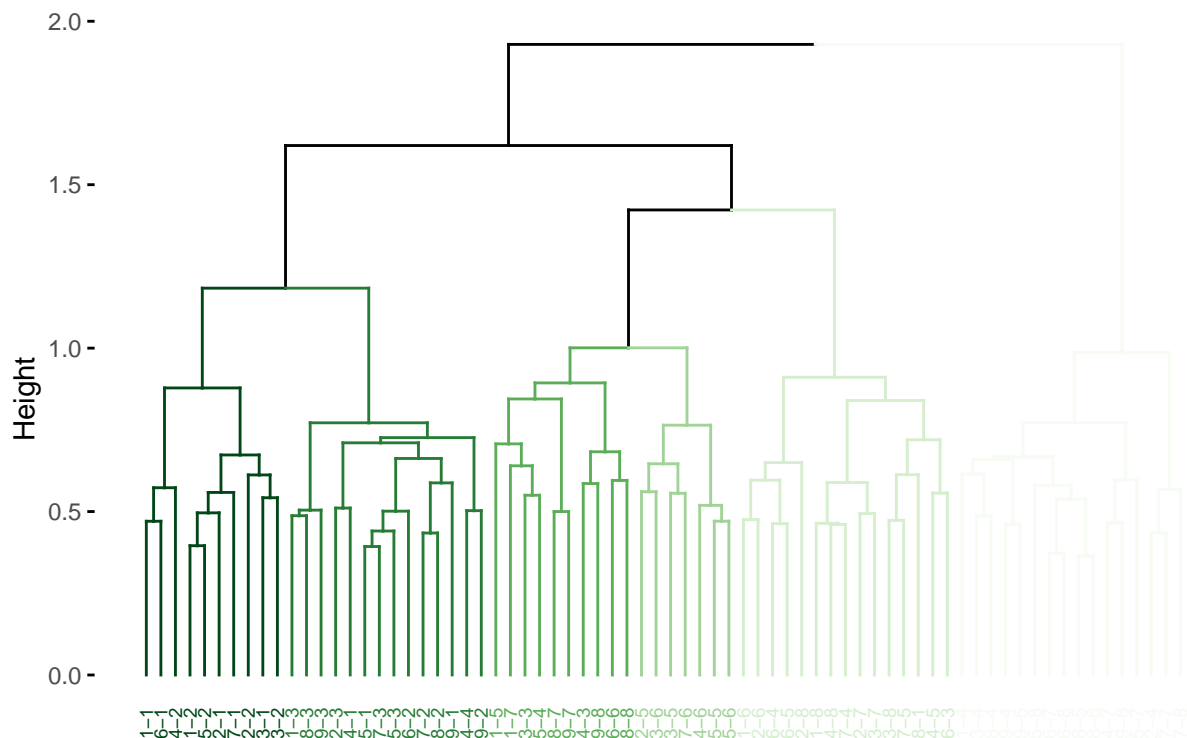
Cluster Dendrogram



##

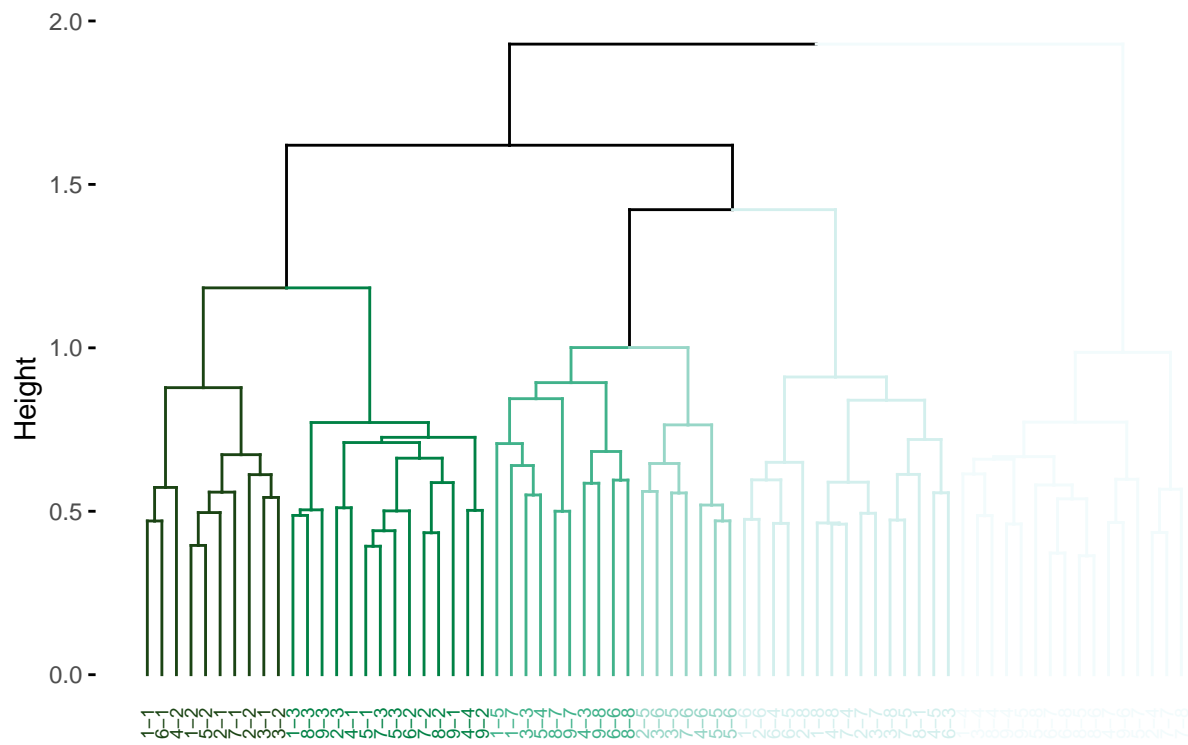
## [[69]]

Cluster Dendrogram



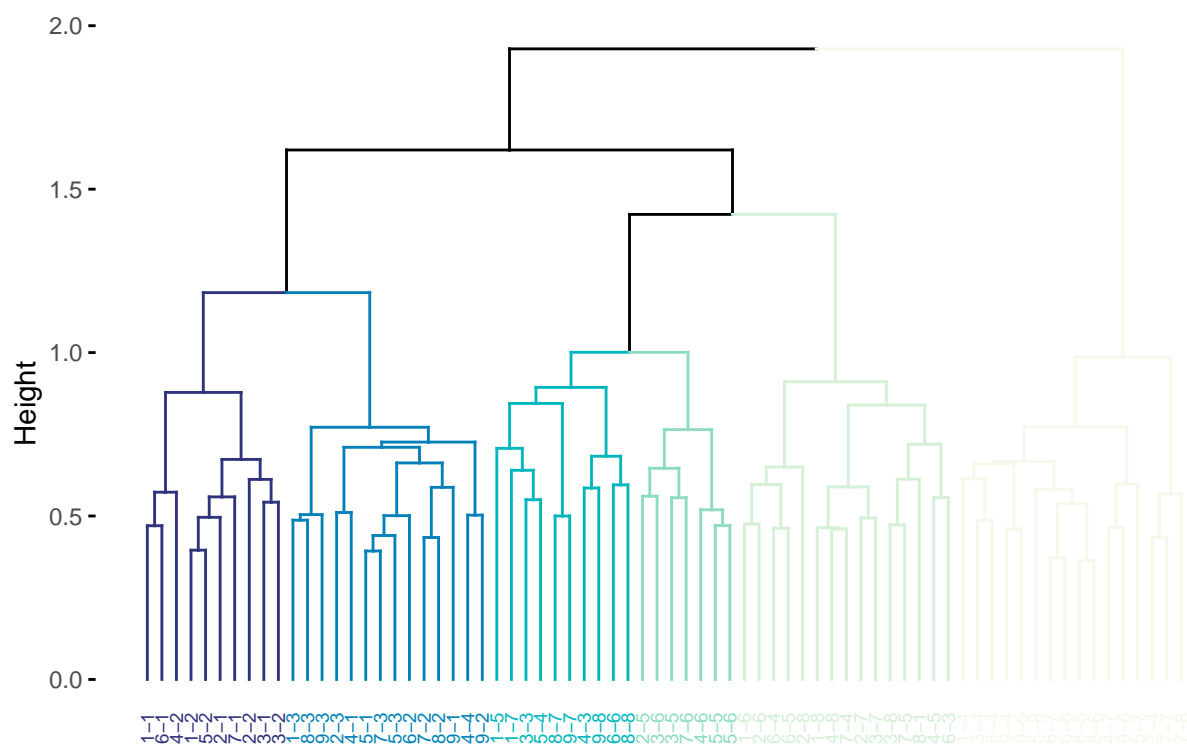
```
##  
## [[70]]
```

Cluster Dendrogram



```
##  
## [[71]]
```

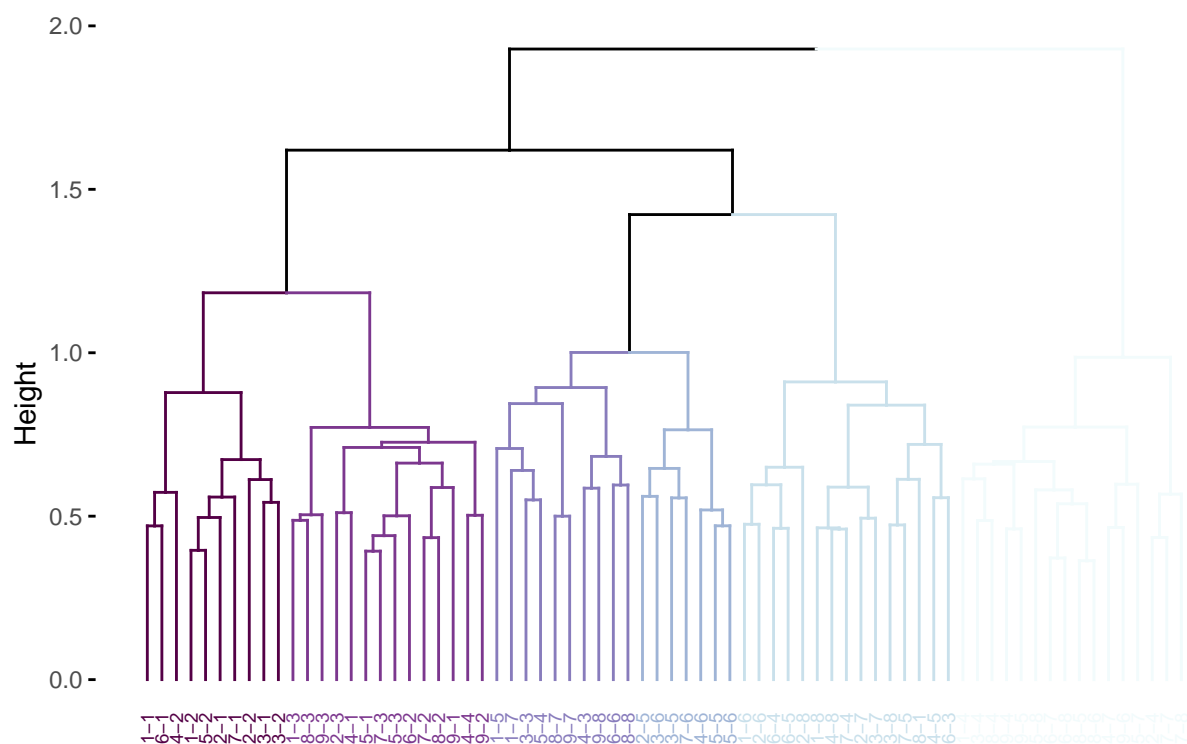
Cluster Dendrogram



##

## [[72]]

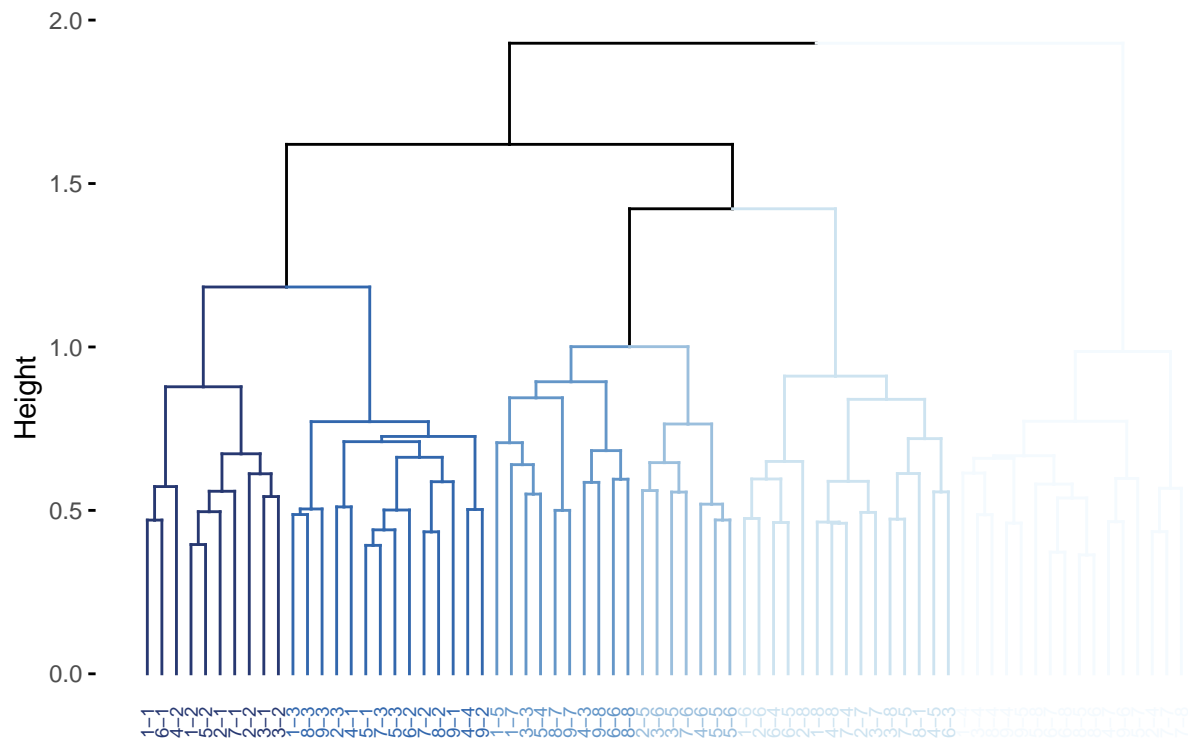
Cluster Dendrogram



##

## [[73]]

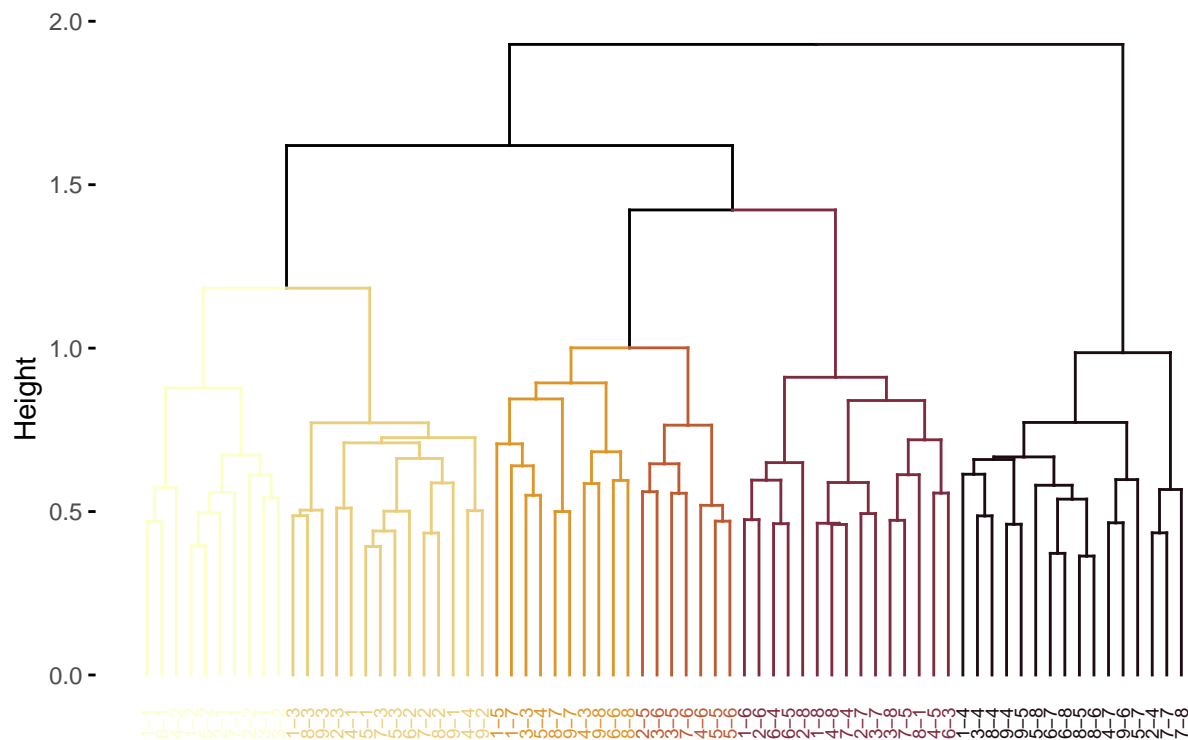
Cluster Dendrogram



##

## [[74]]

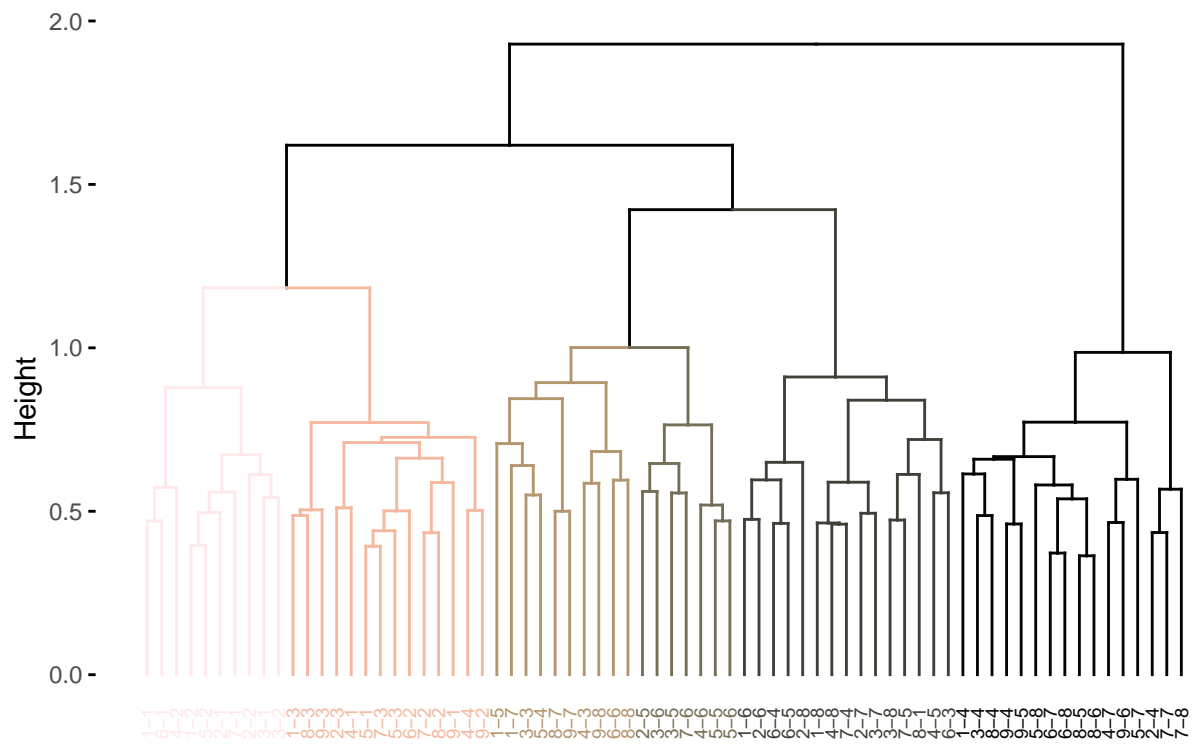
Cluster Dendrogram



##

## [[75]]

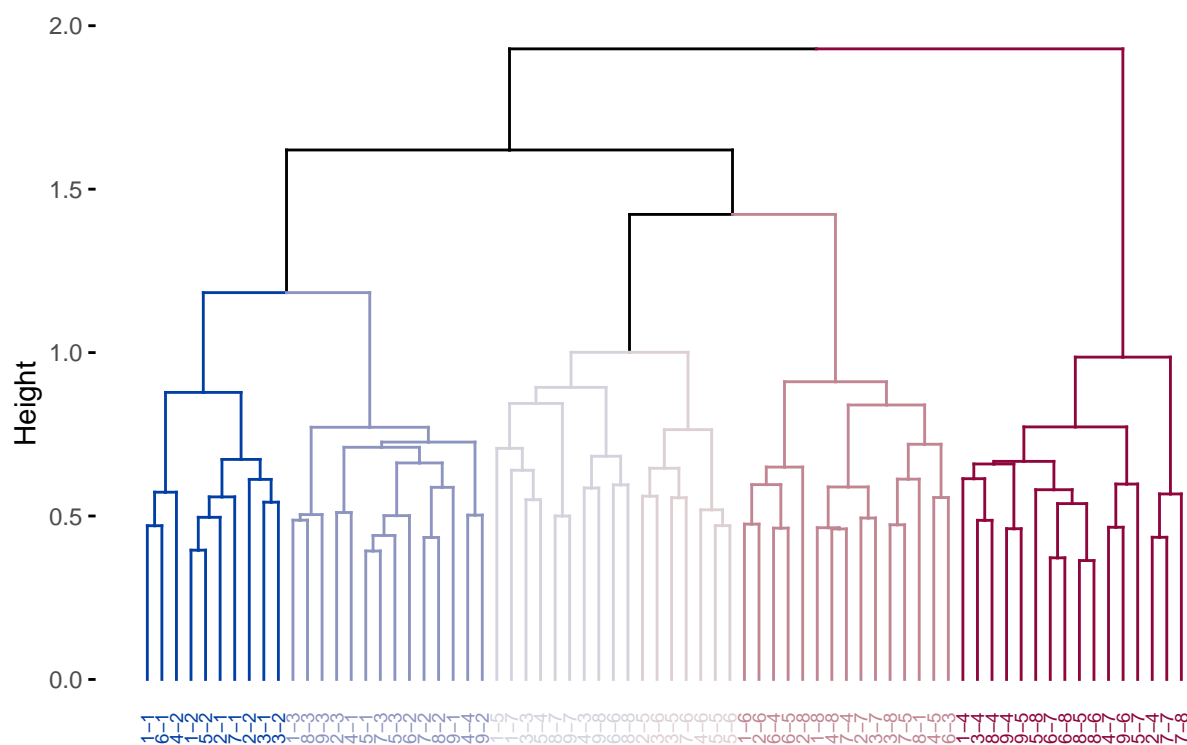
Cluster Dendrogram



##

## [[76]]

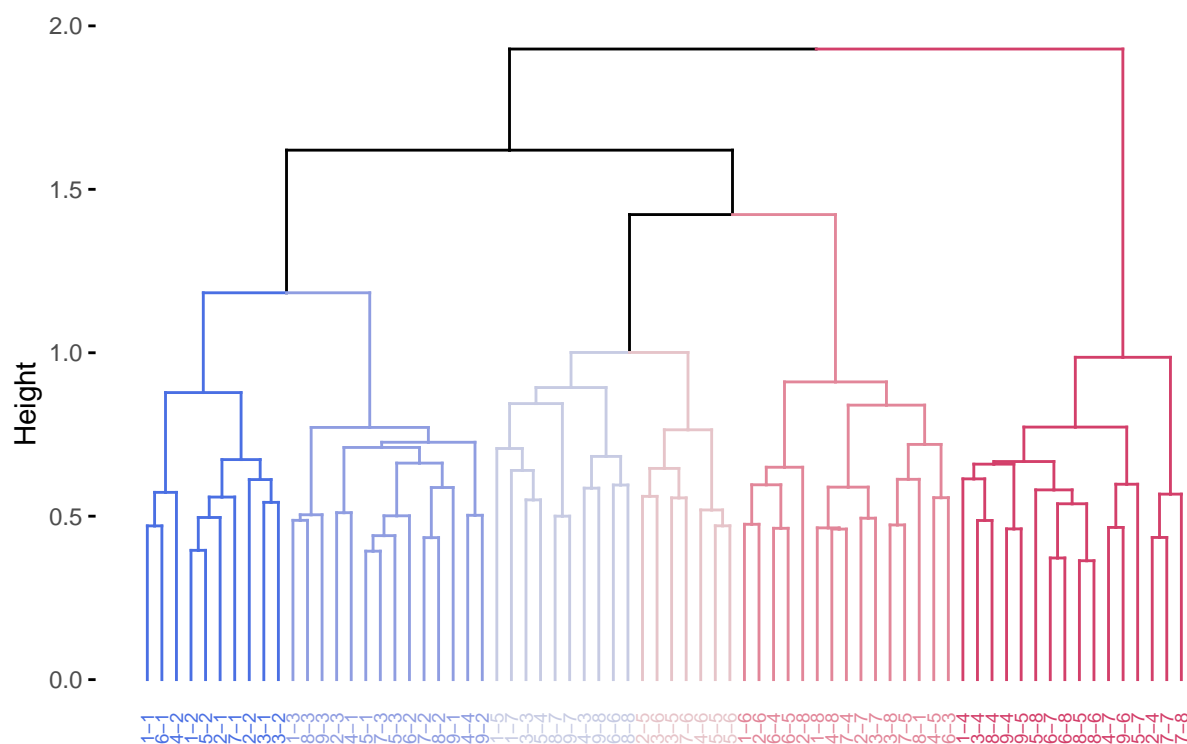
### Cluster Dendrogram



##

## [[77]]

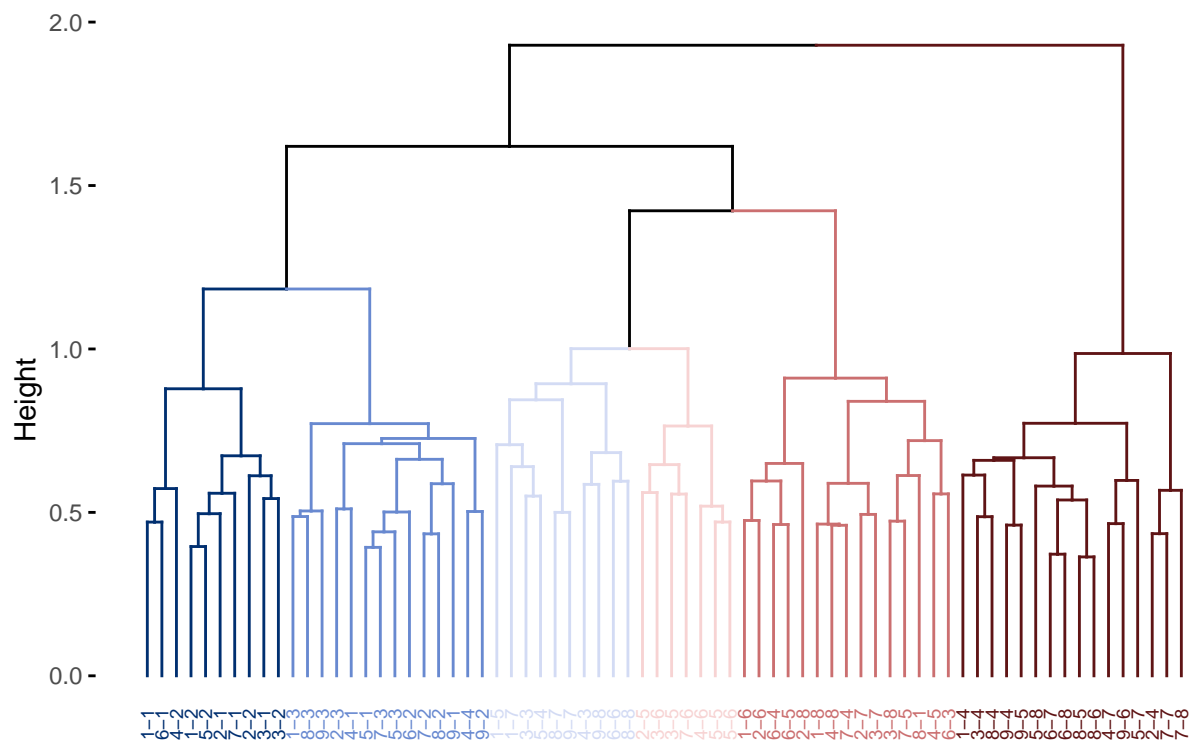
### Cluster Dendrogram



##

## [[78]]

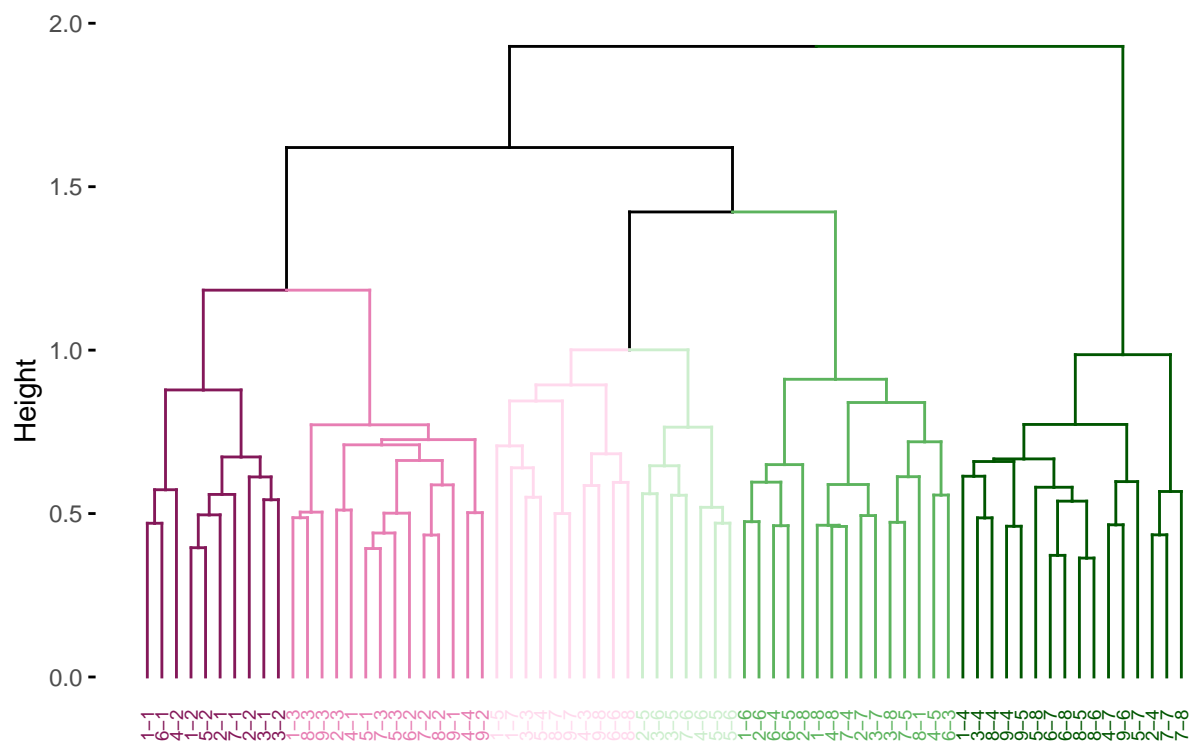
Cluster Dendrogram



##

## [[79]]

Cluster Dendrogram

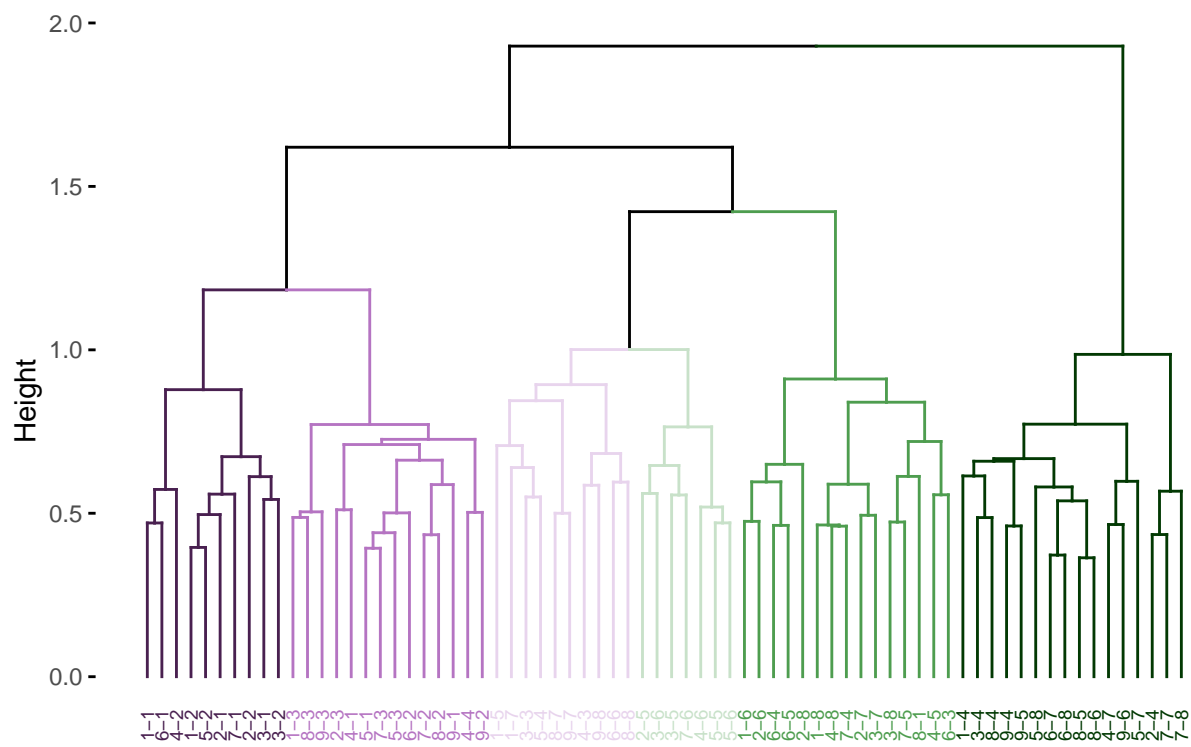




##

## [[80]]

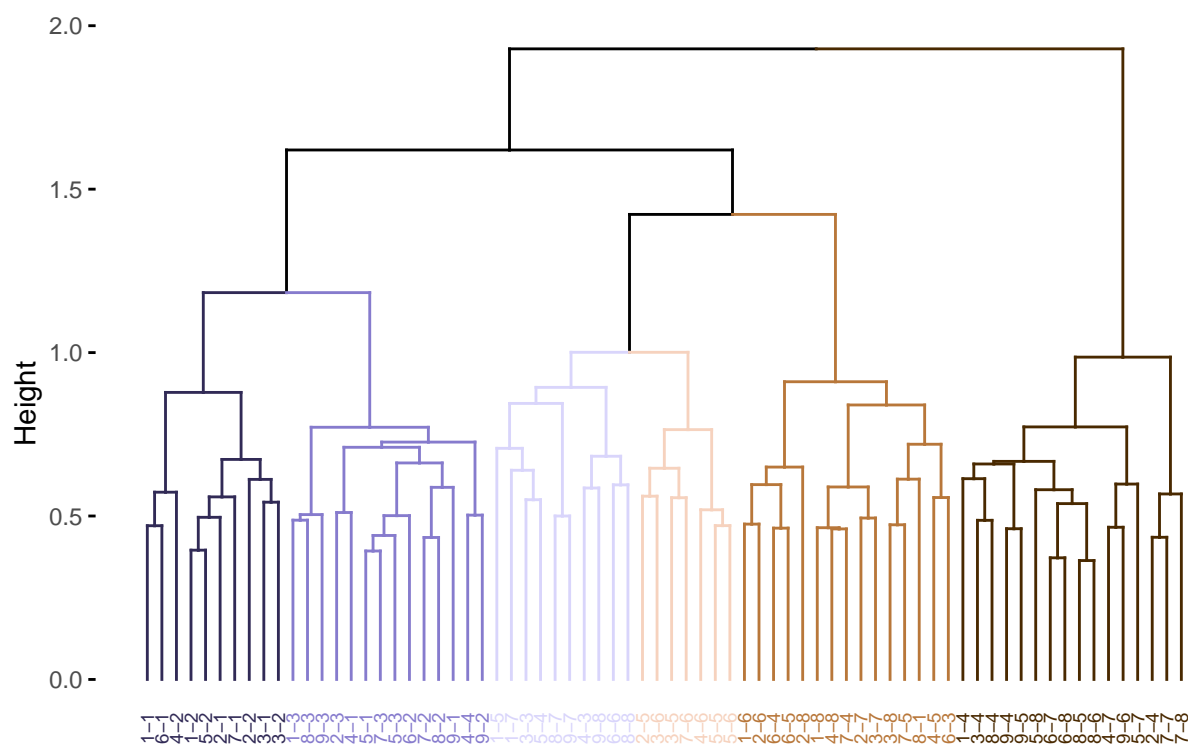
Cluster Dendrogram



##

## [[81]]

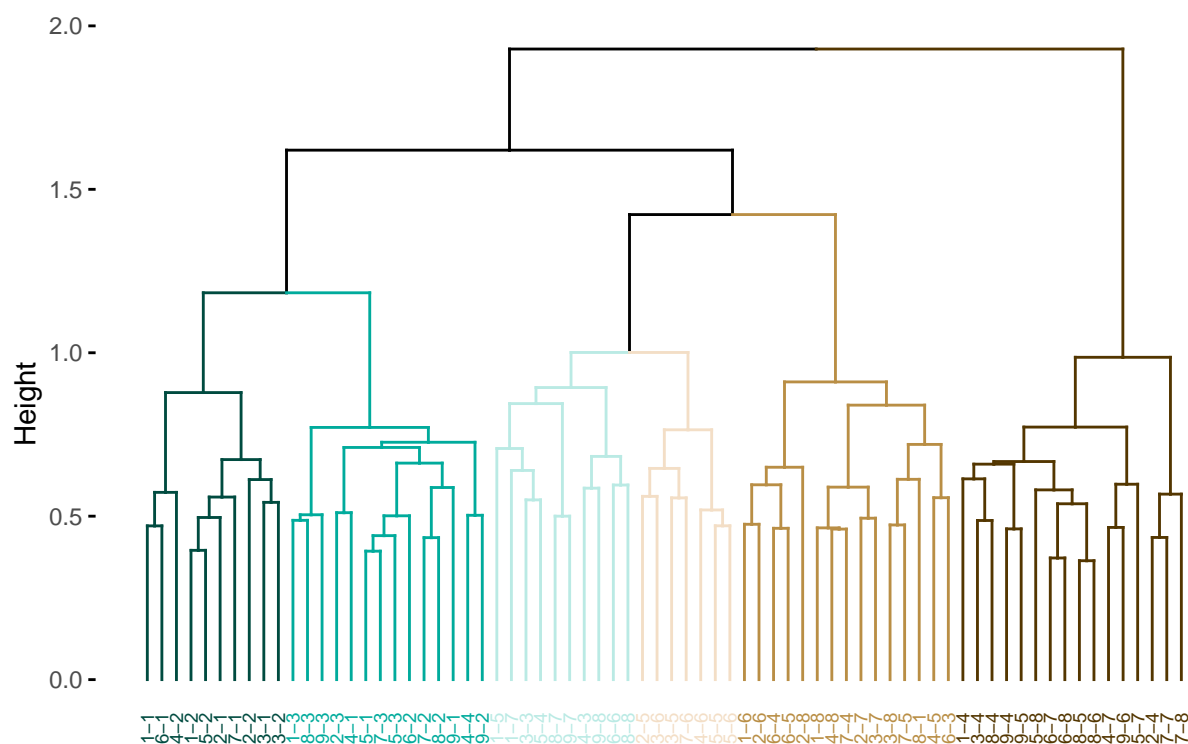
Cluster Dendrogram



##

## [[82]]

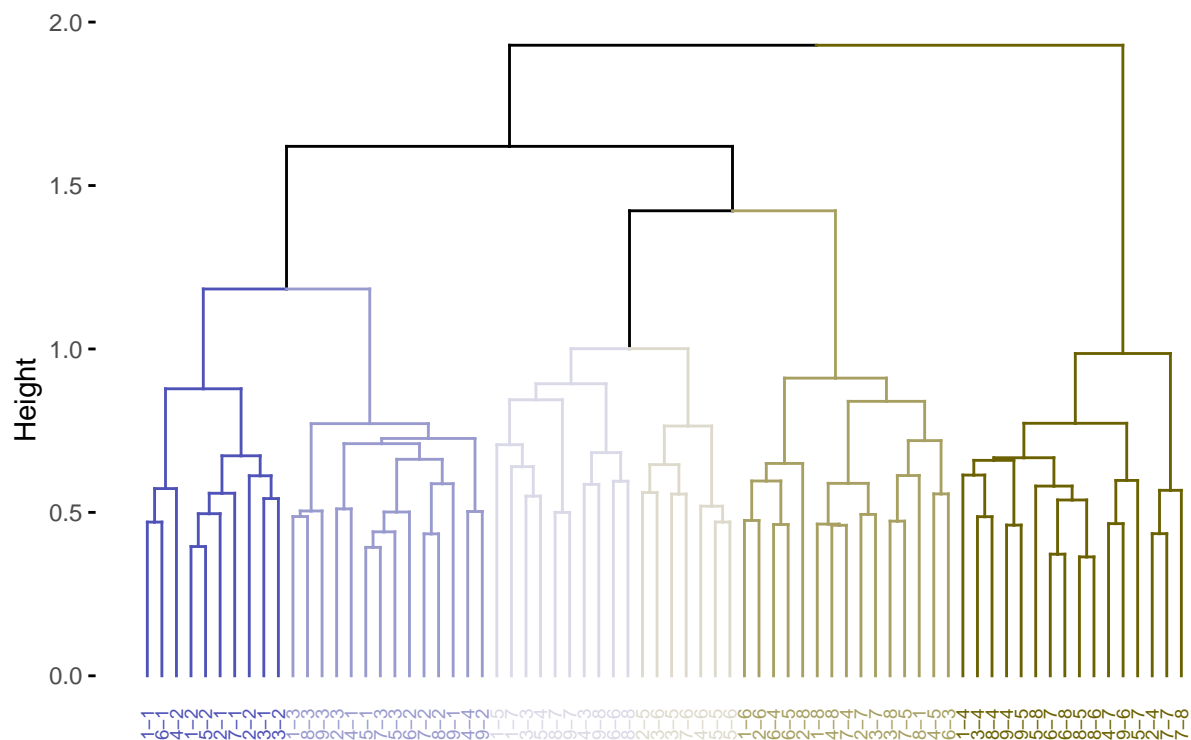
Cluster Dendrogram



##

## [[83]]

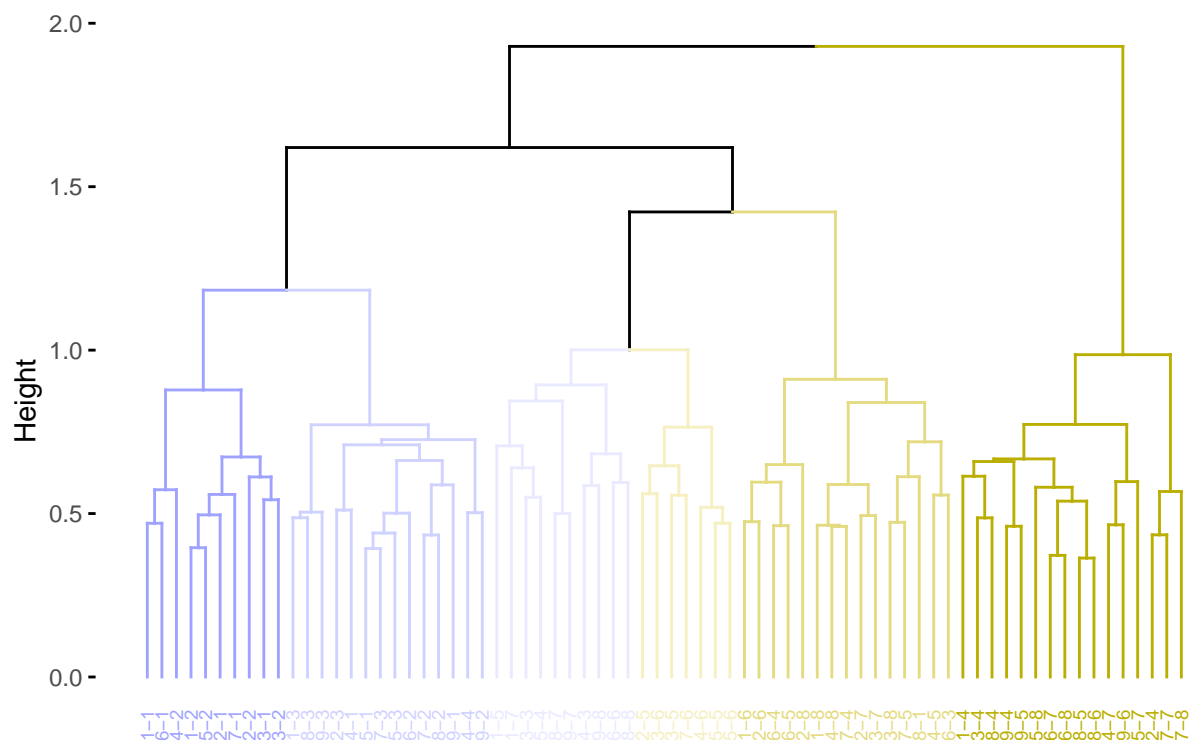
Cluster Dendrogram



##

## [[84]]

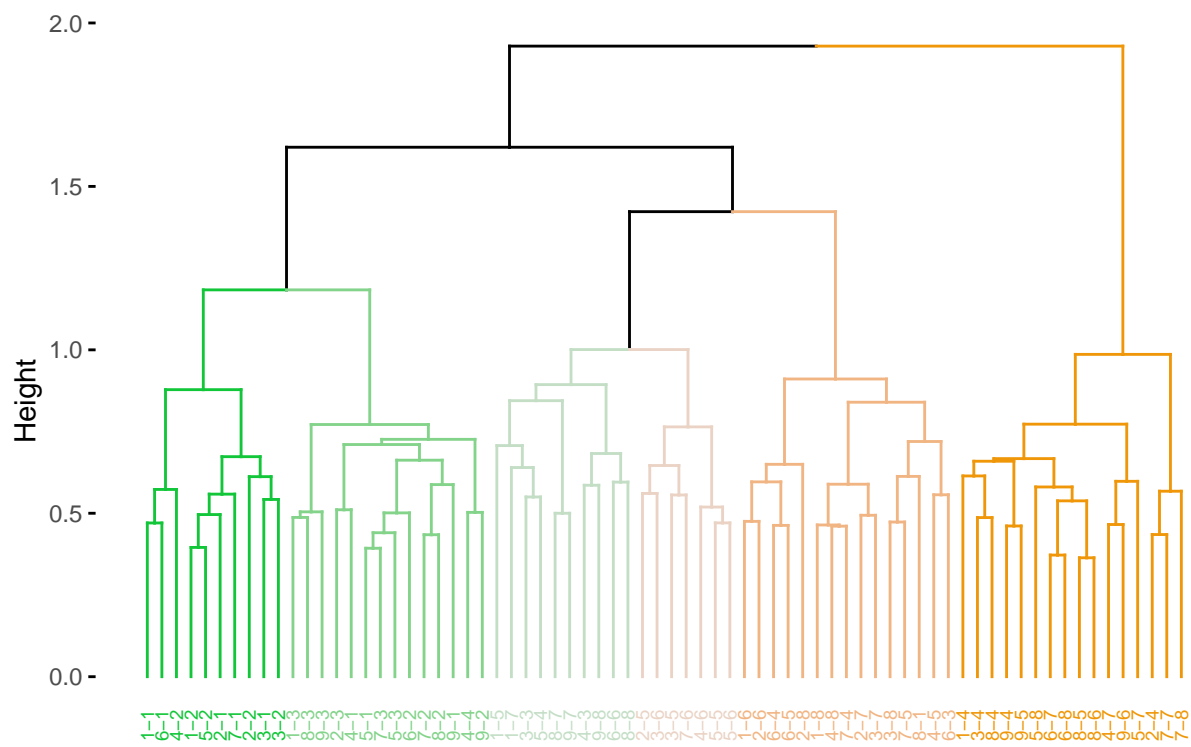
Cluster Dendrogram



##

## [[85]]

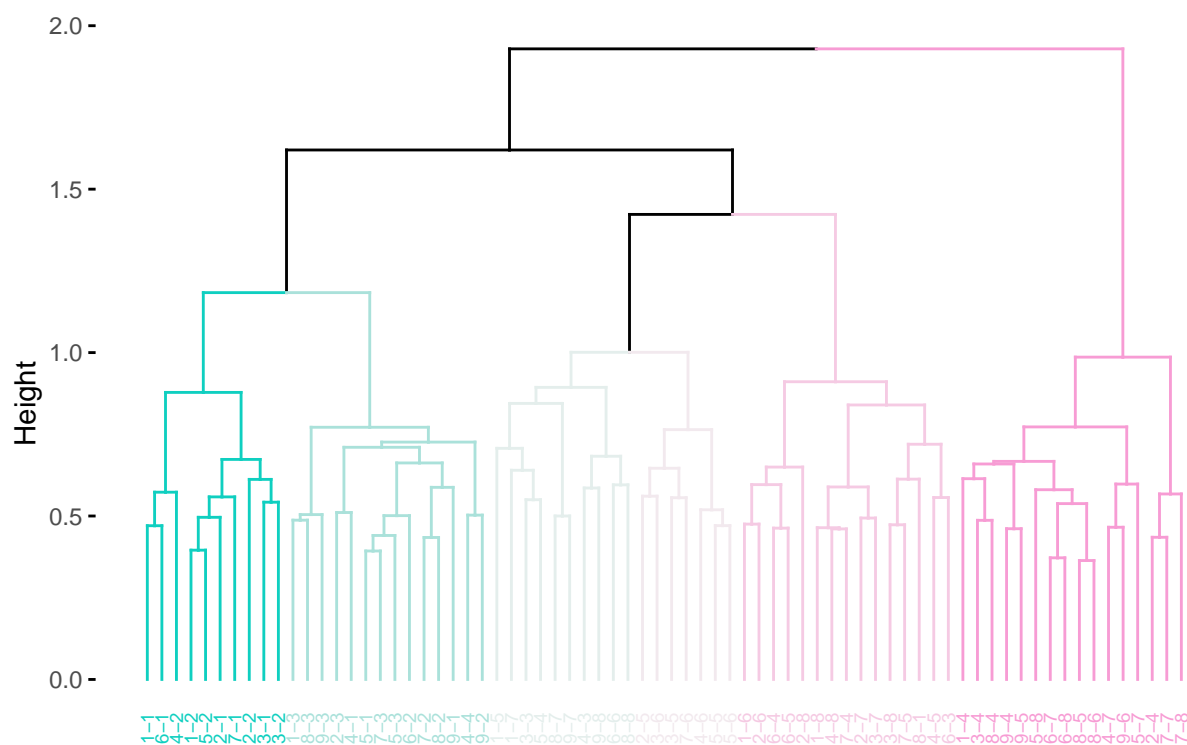
Cluster Dendrogram



##

## [[86]]

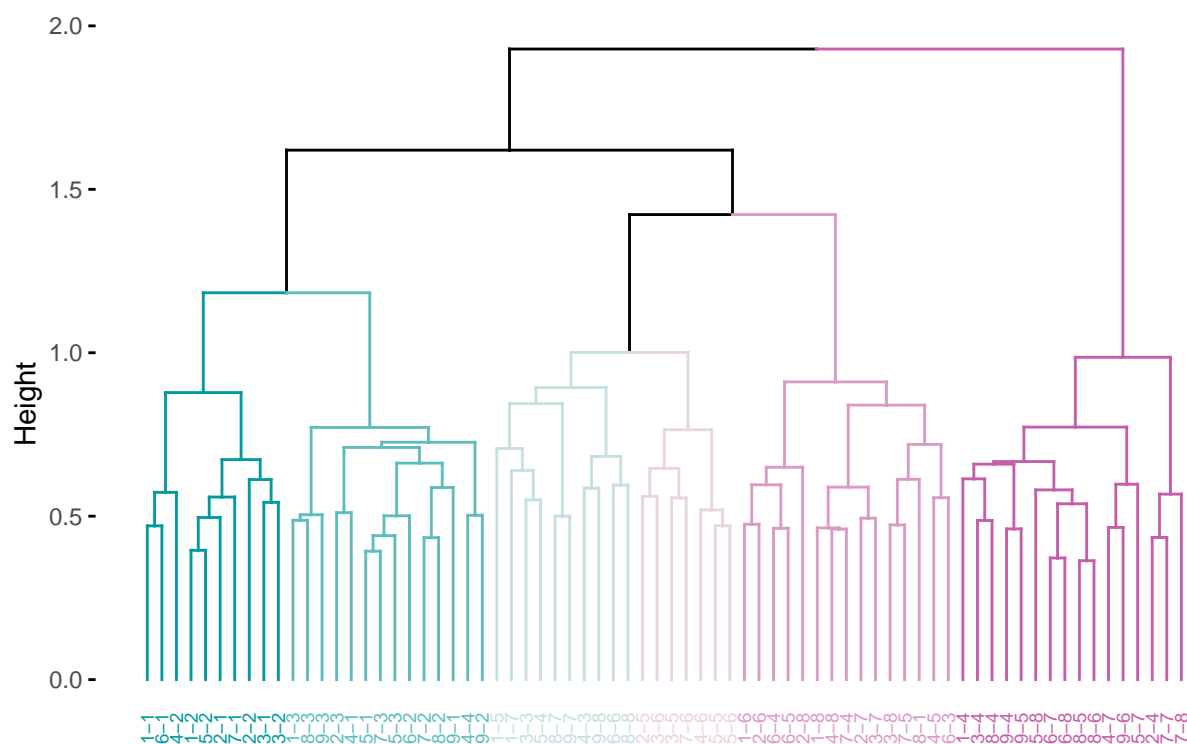
Cluster Dendrogram



##

## [[87]]

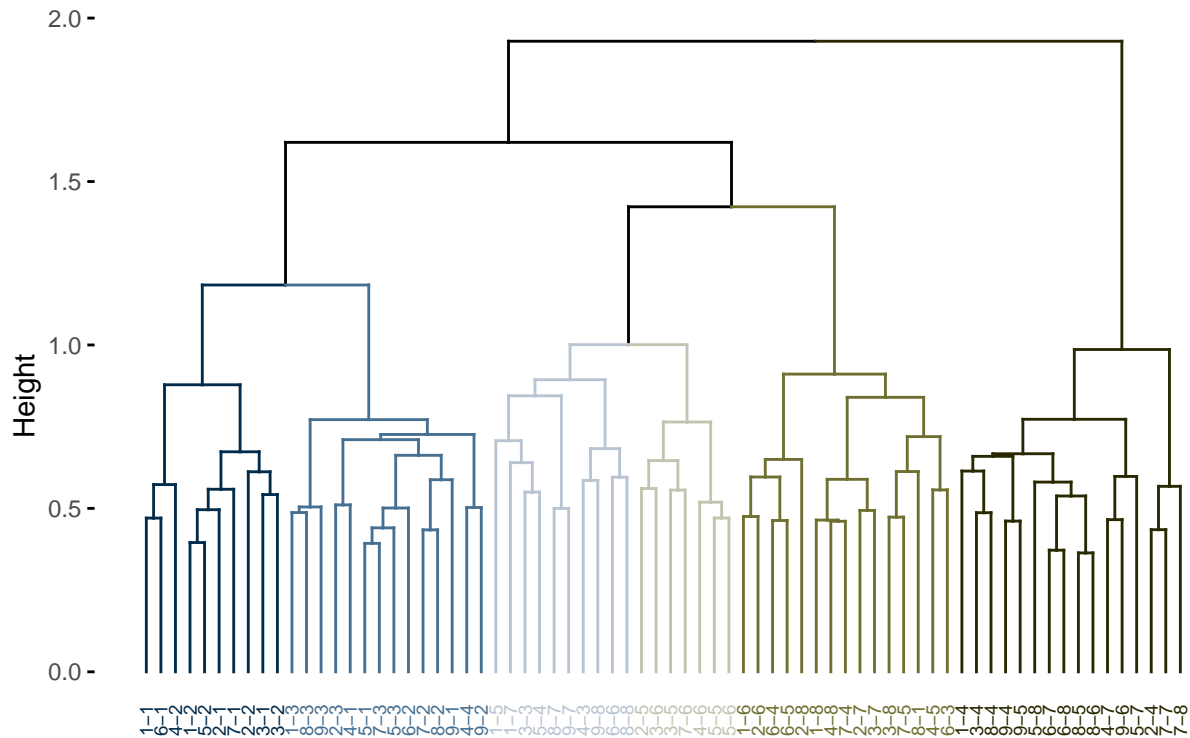
Cluster Dendrogram



##

## [[88]]

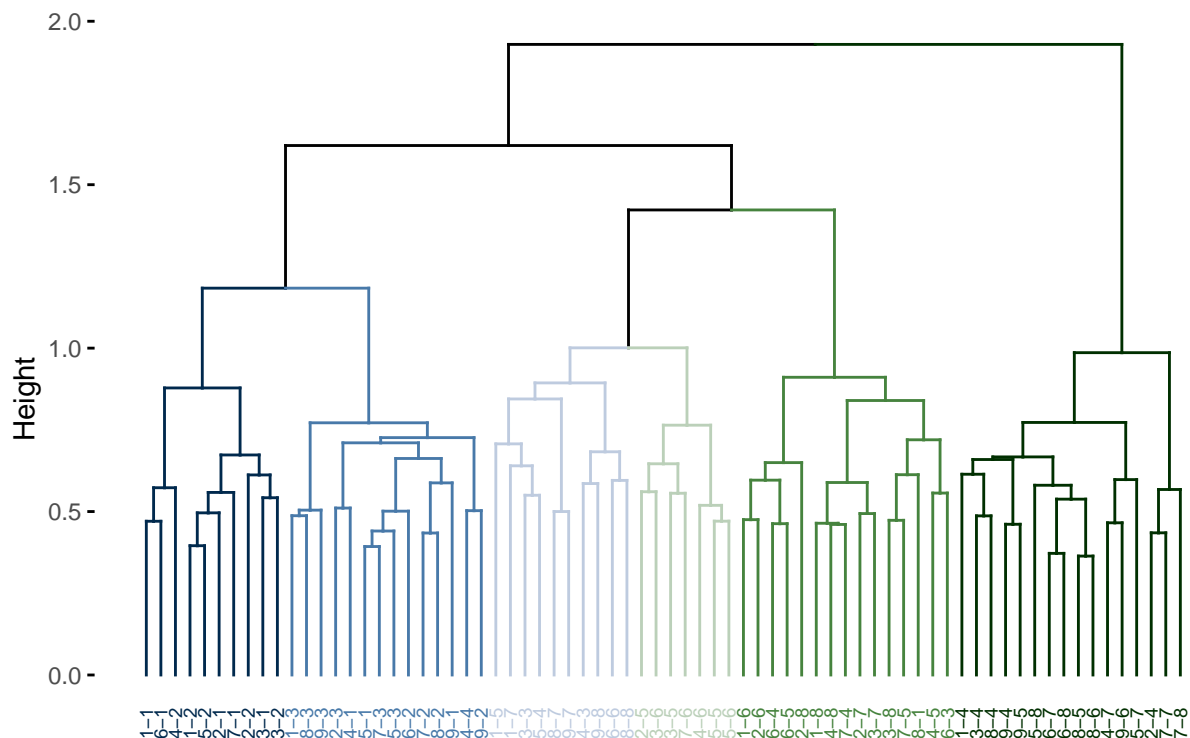
Cluster Dendrogram



##

## [[89]]

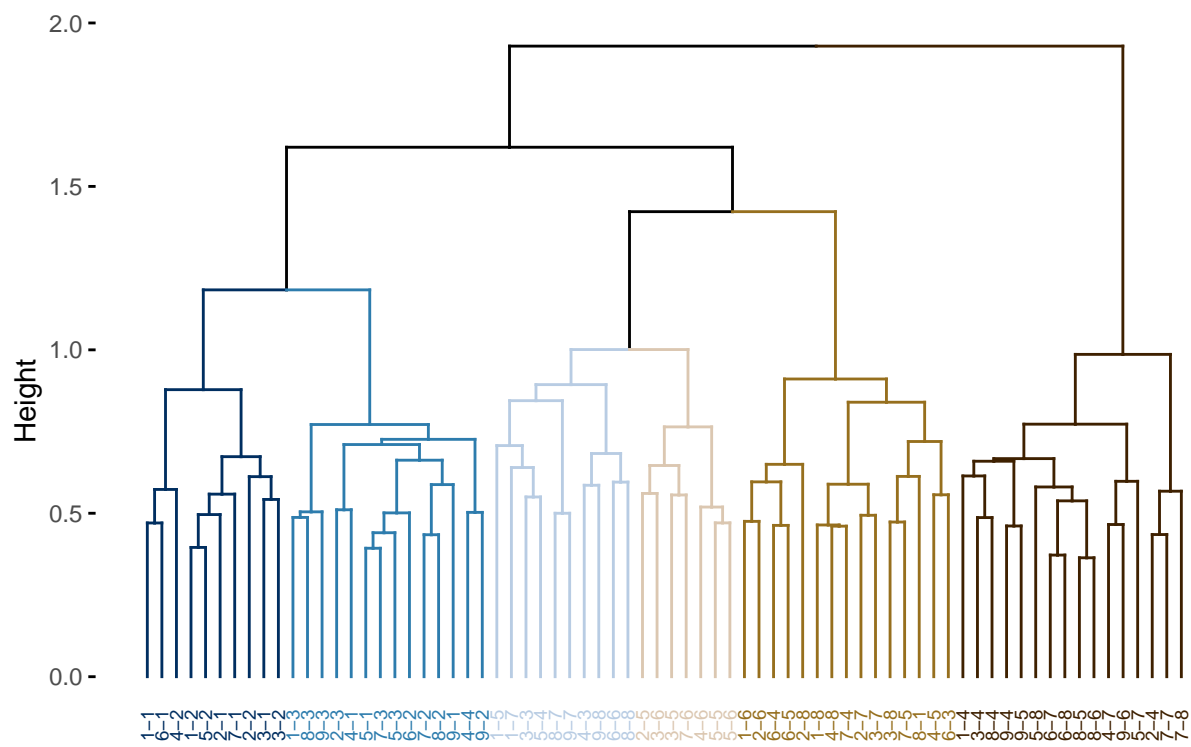
Cluster Dendrogram



##

## [[90]]

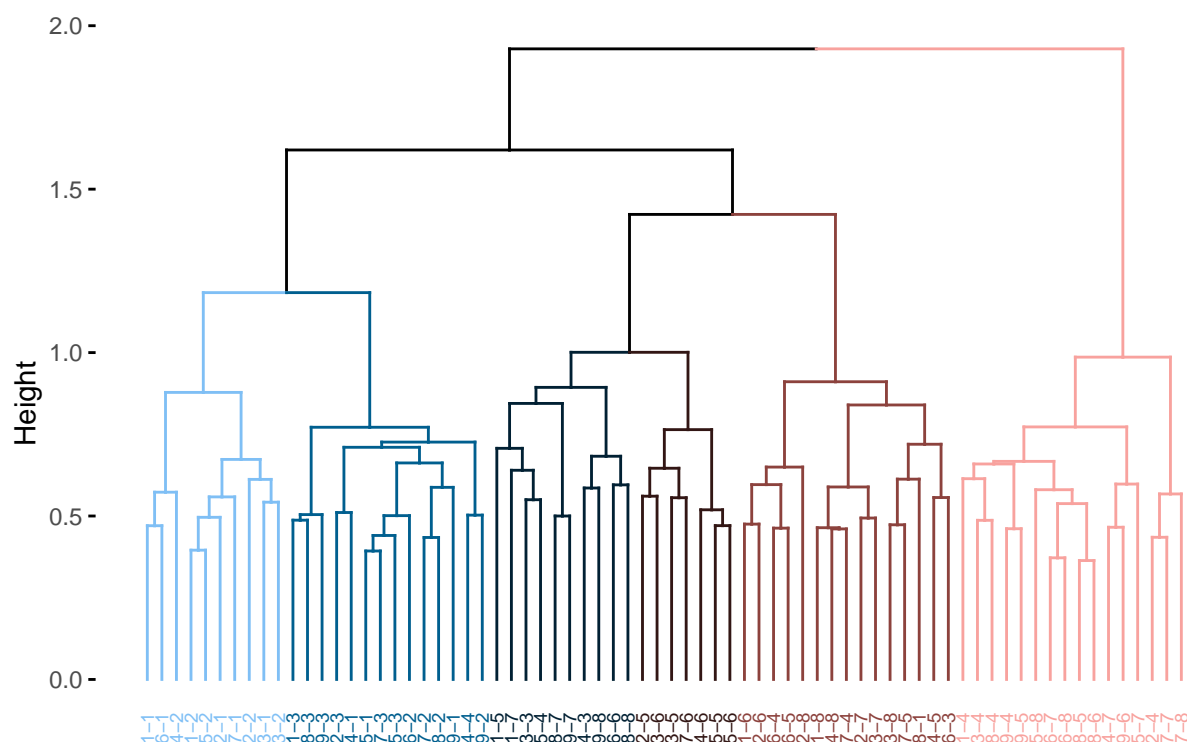
Cluster Dendrogram



##

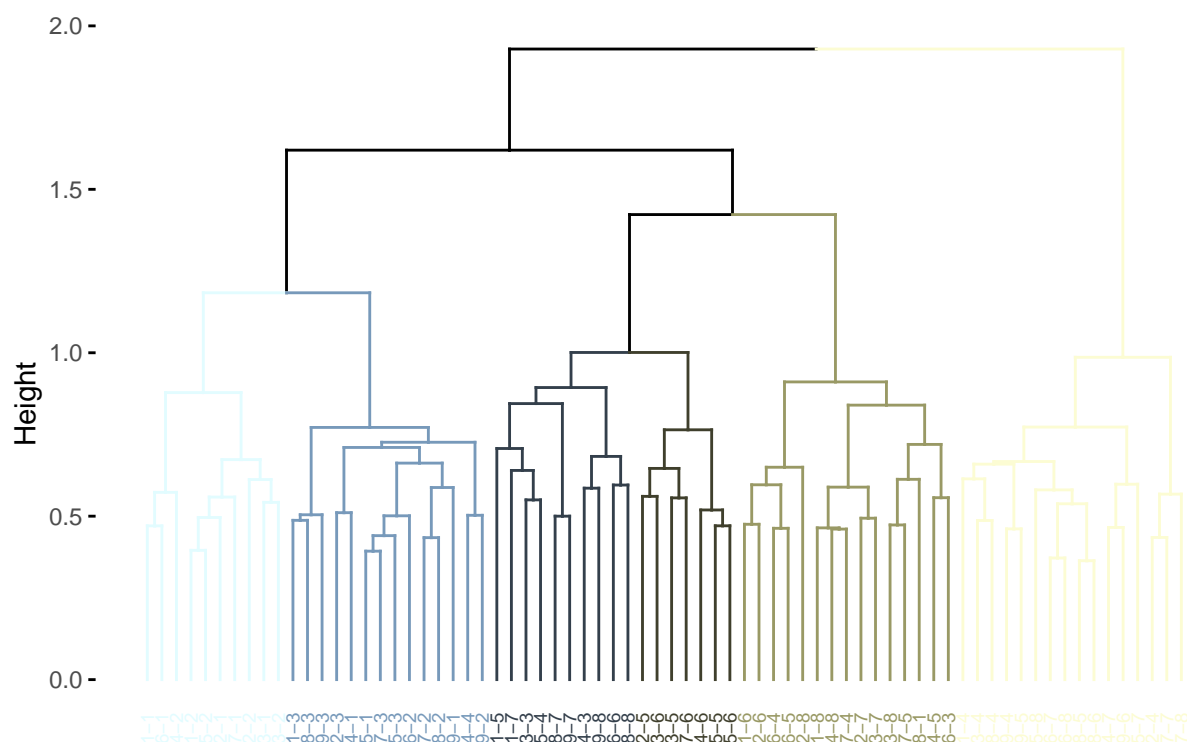
## [[91]]

# Cluster Dendrogram



##  
## [[92]]

# Cluster Dendrogram

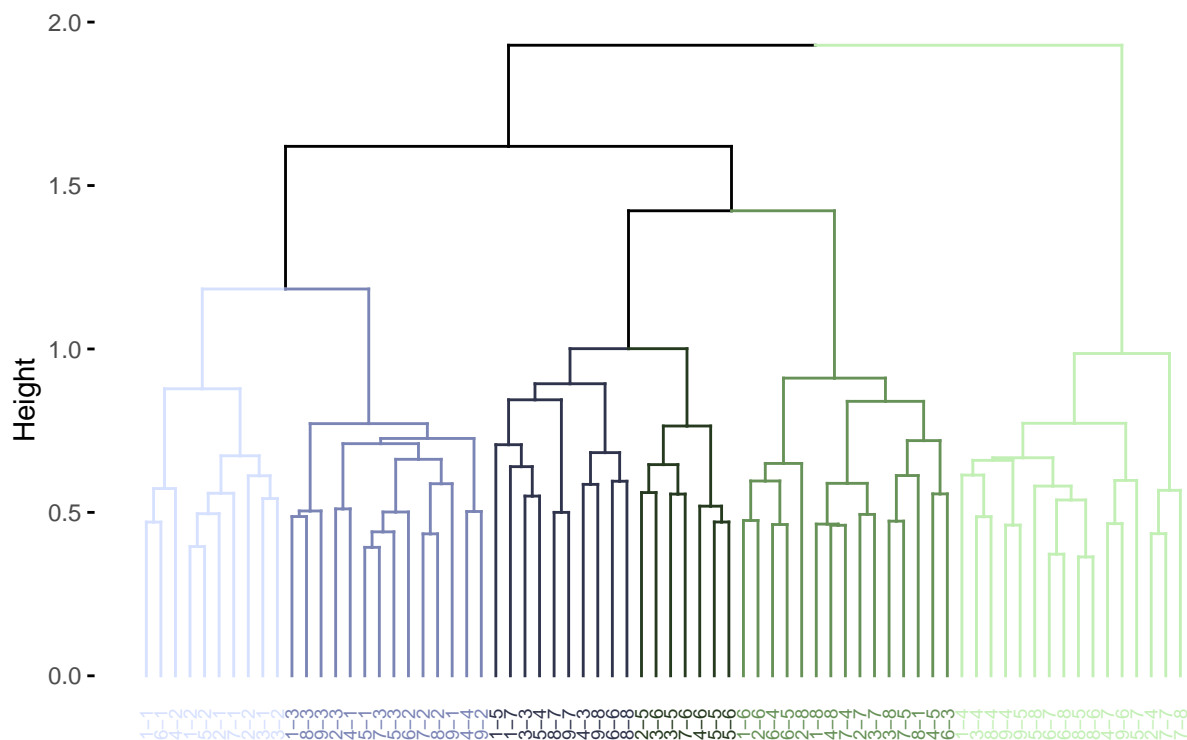


##



## [[93]]

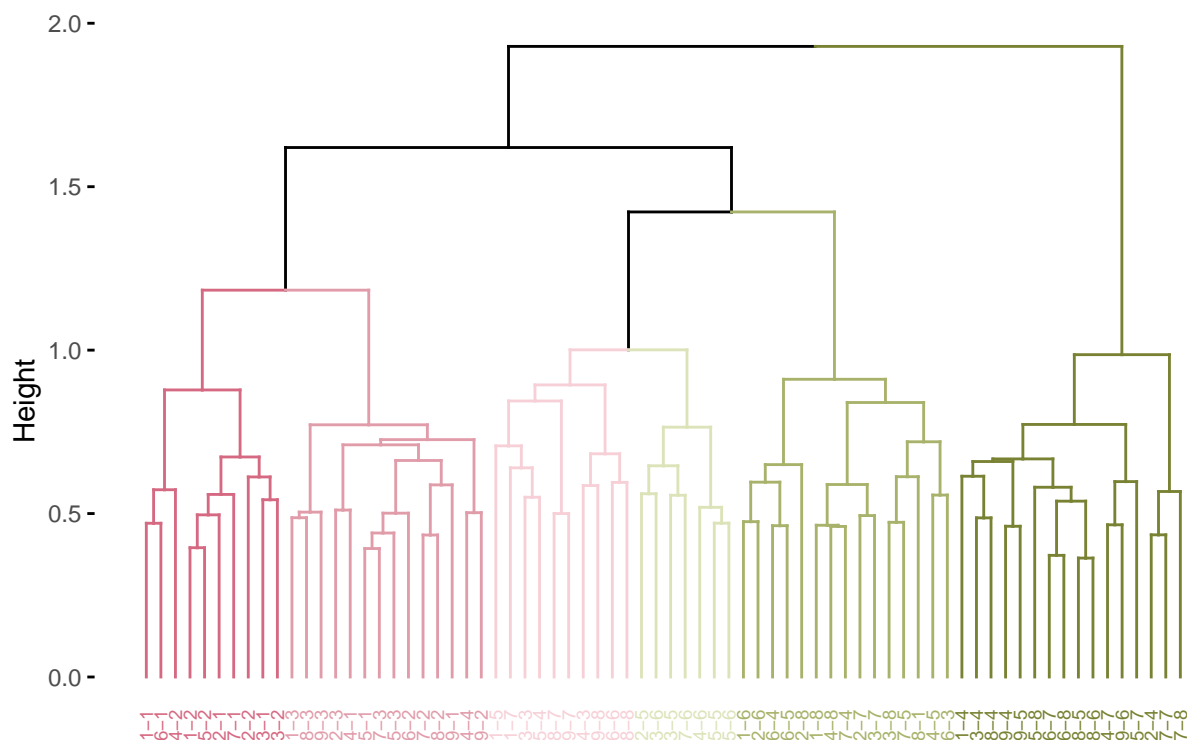
Cluster Dendrogram



##

## [[94]]

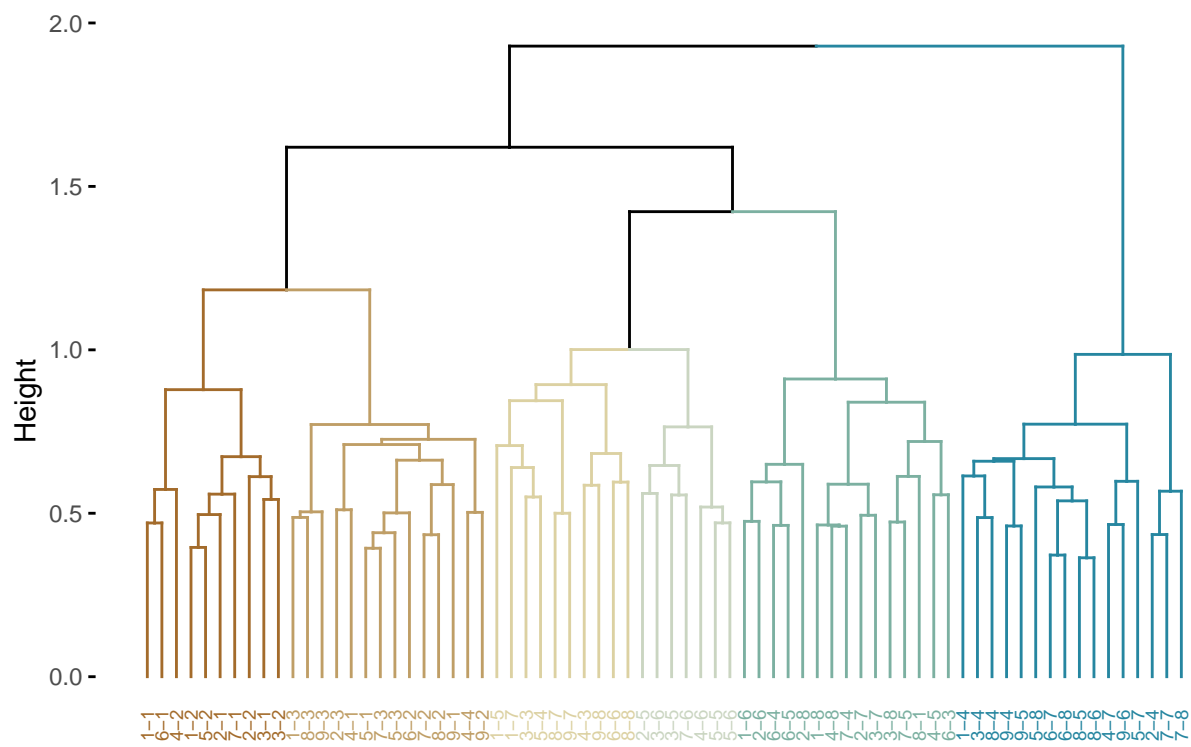
Cluster Dendrogram



##

## [[95]]

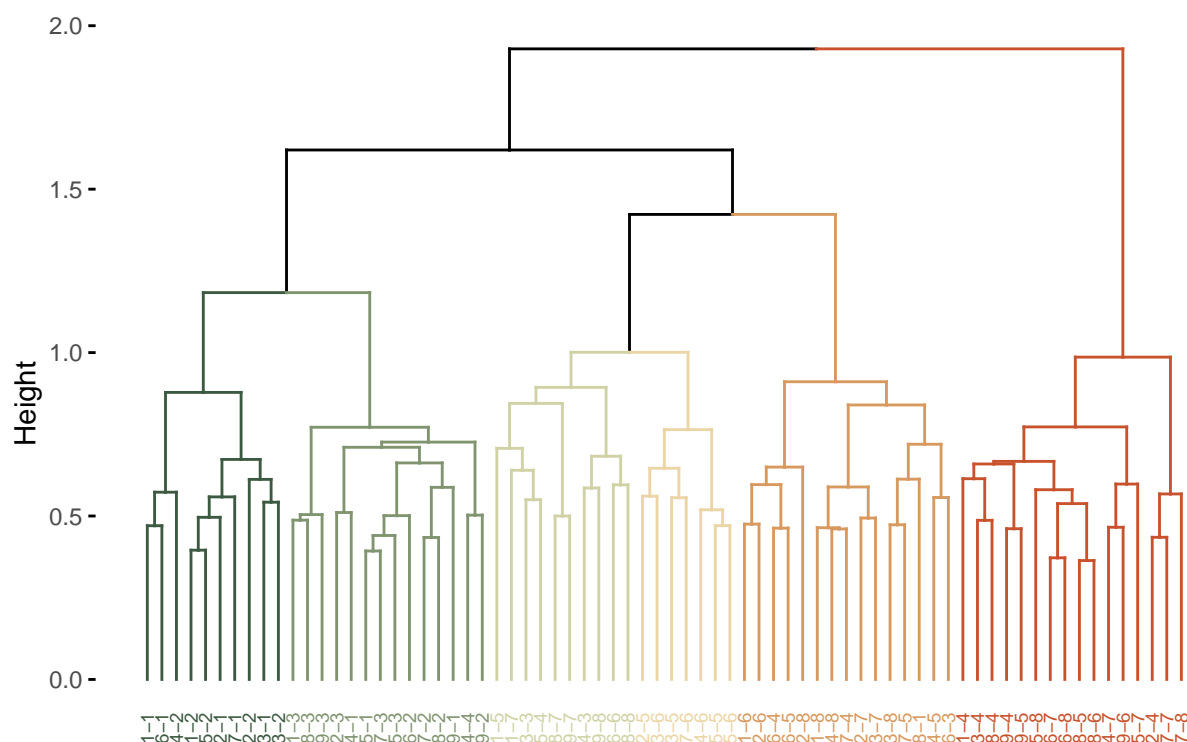
Cluster Dendrogram



##

## [[96]]

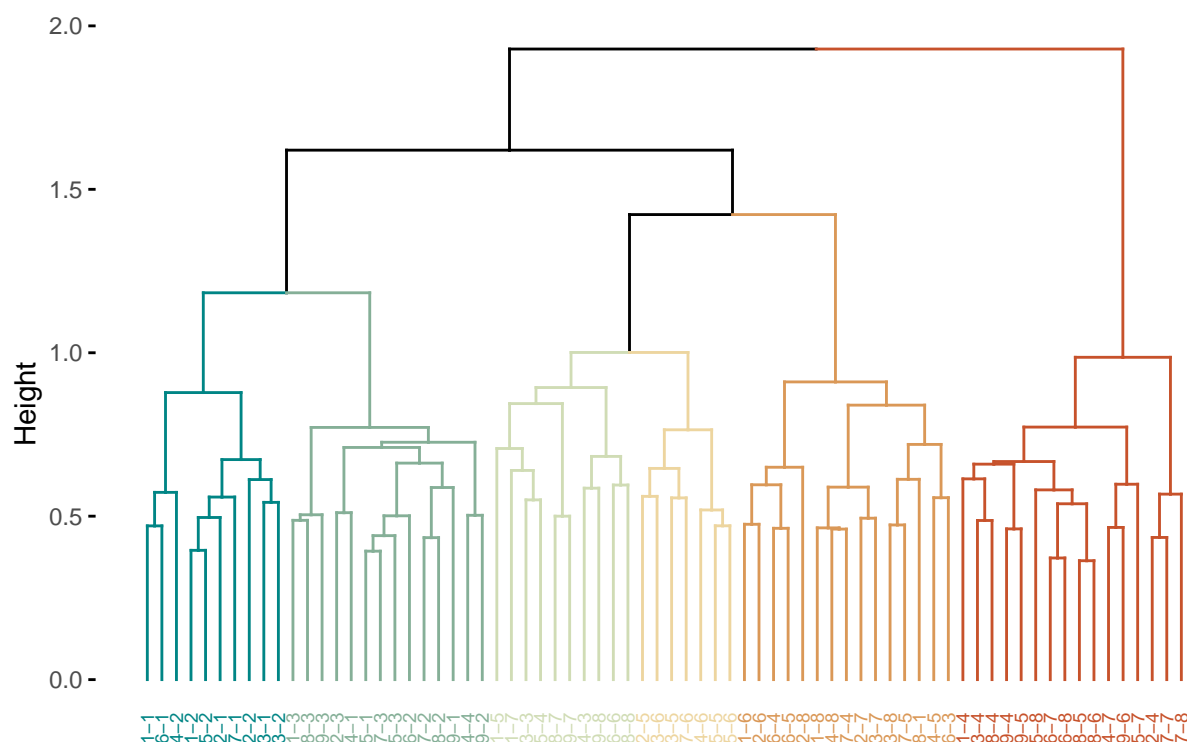
Cluster Dendrogram



##

## [[97]]

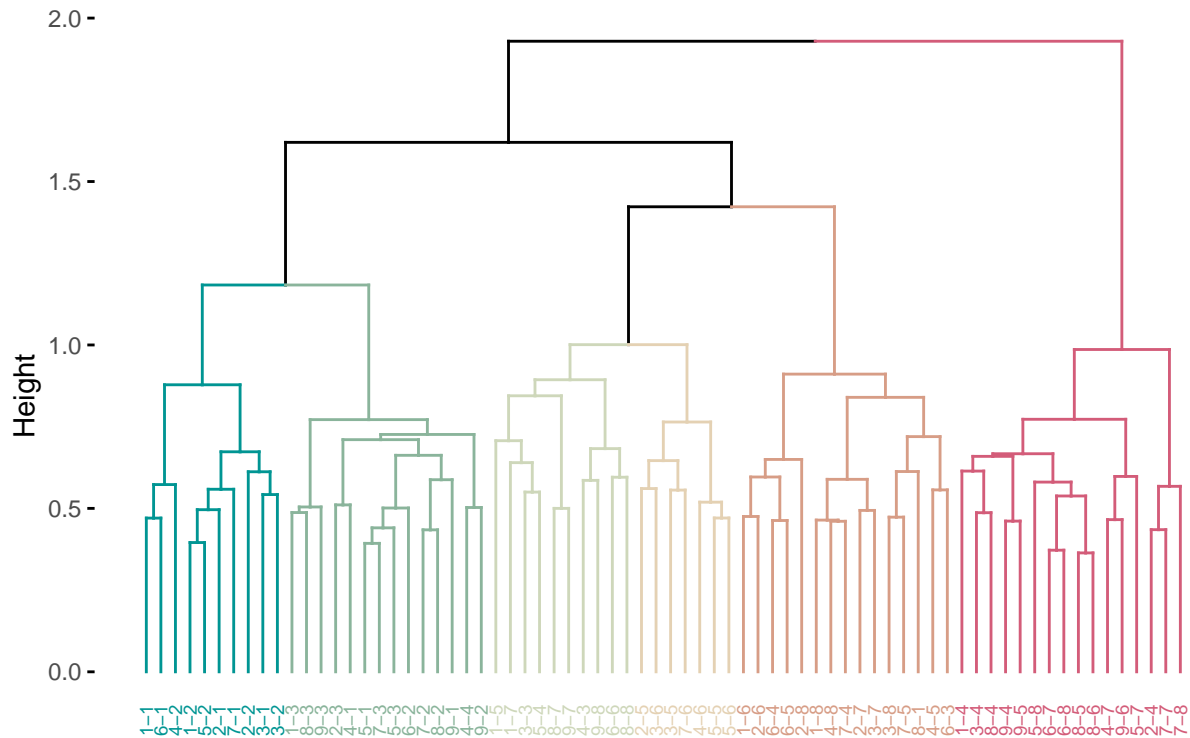
Cluster Dendrogram



##

```
## [[98]]
```

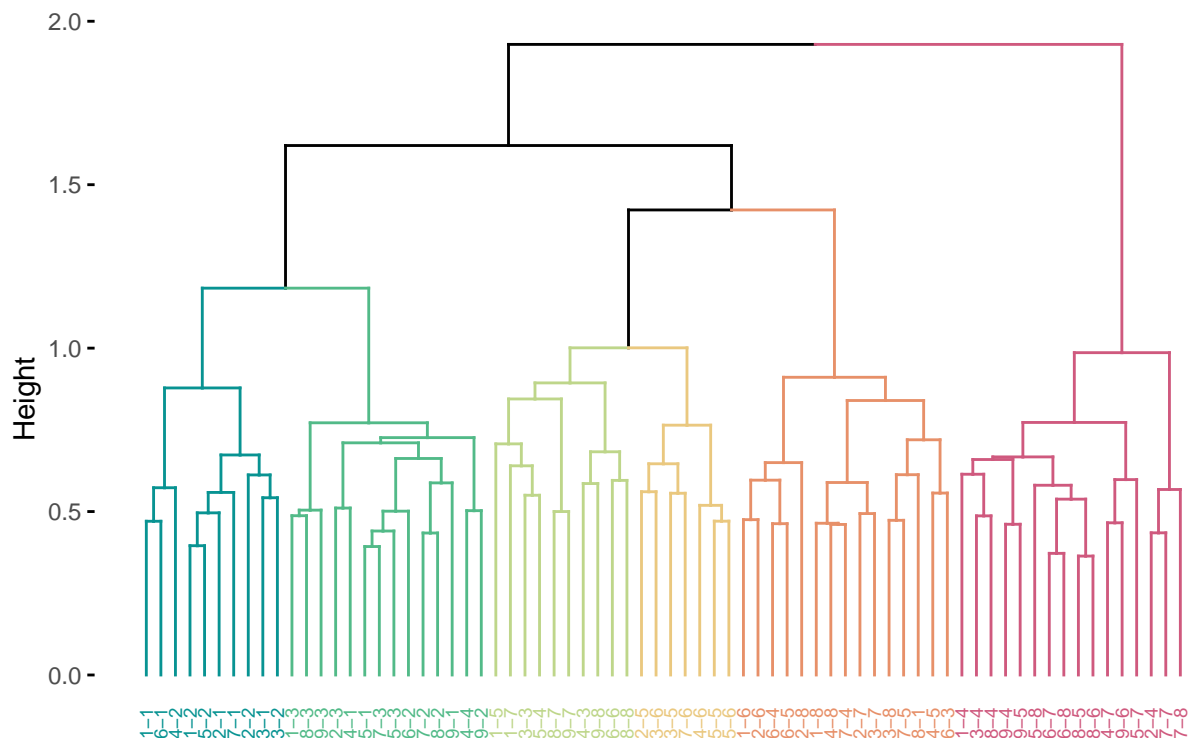
## Cluster Dendrogram



##

```
## [[99]]
```

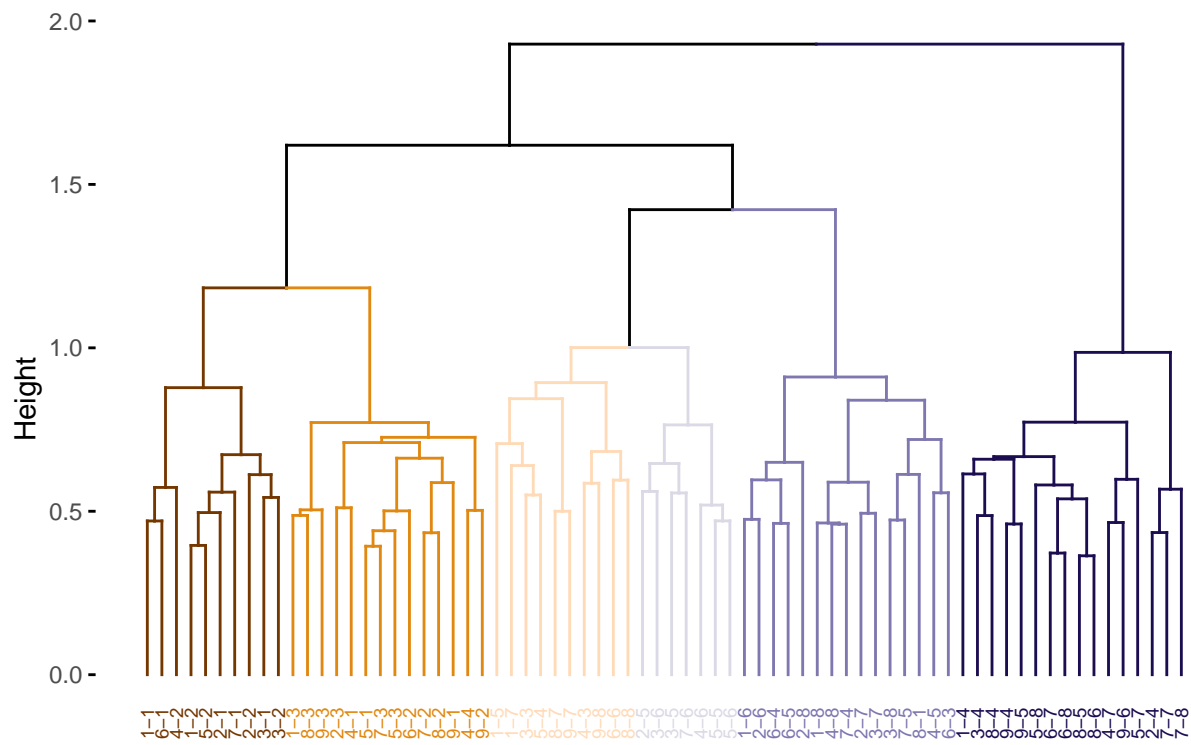
## Cluster Dendrogram



##

## [[100]]

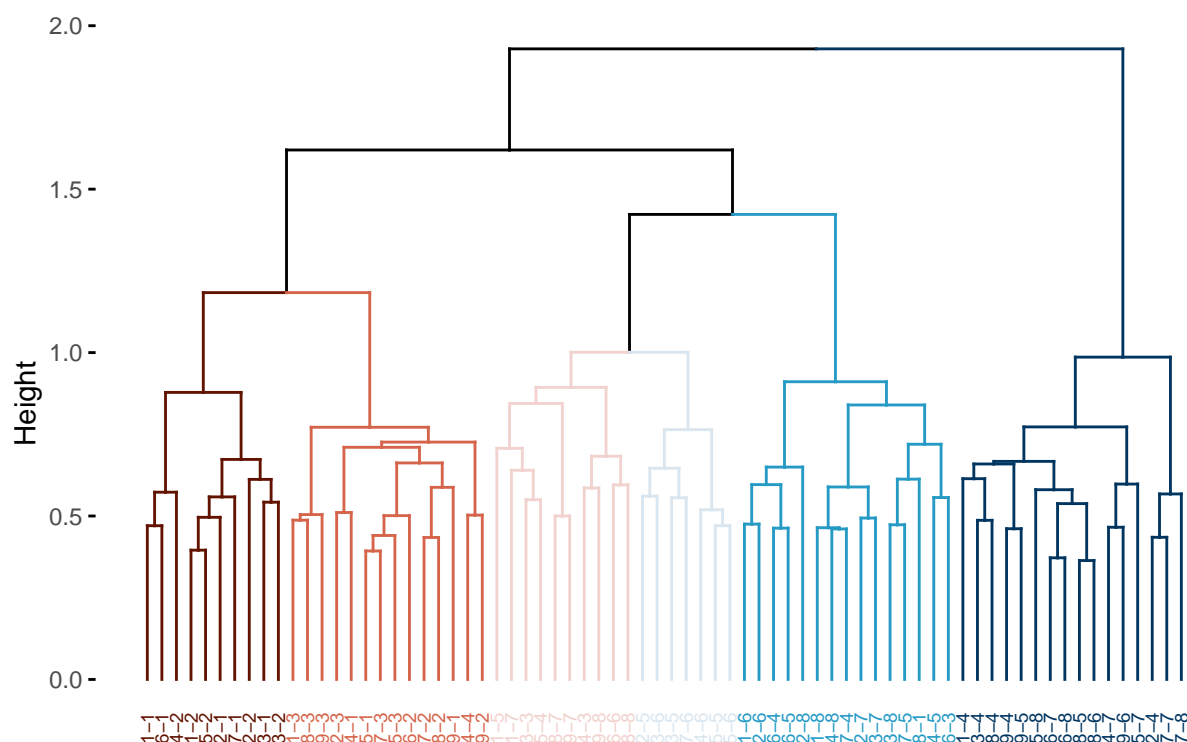
### Cluster Dendrogram



##

## [[101]]

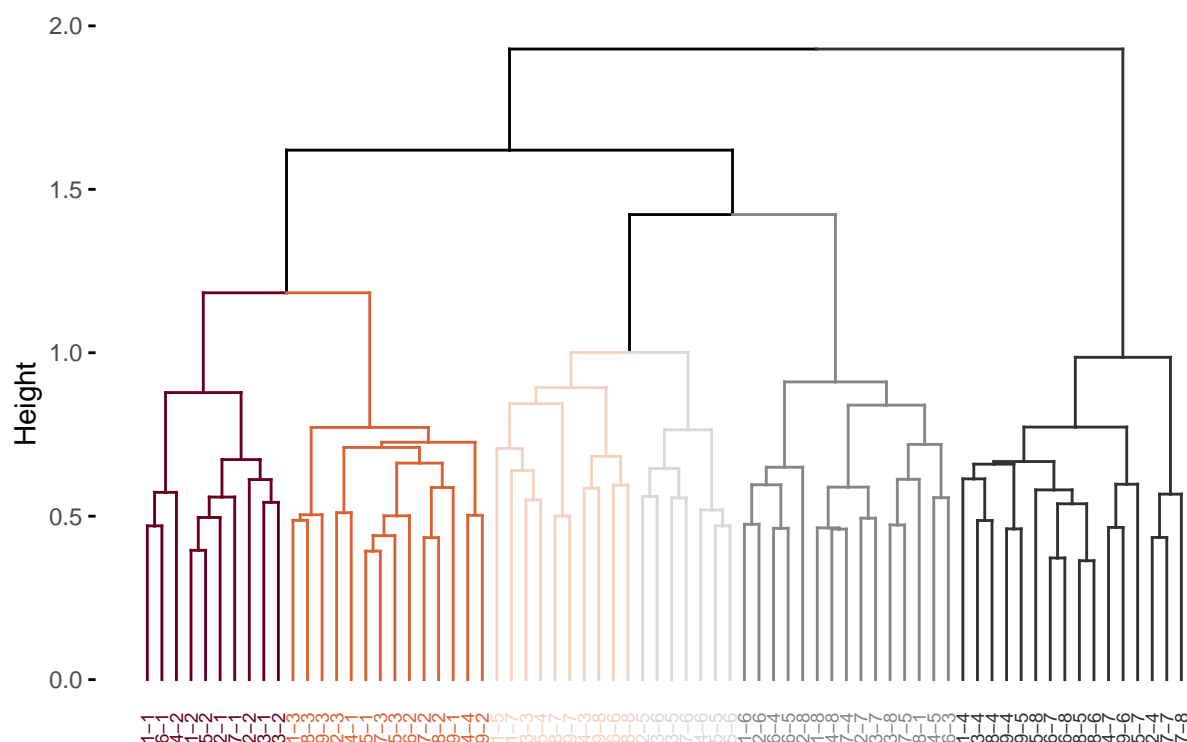
# Cluster Dendrogram



##

## [[102]]

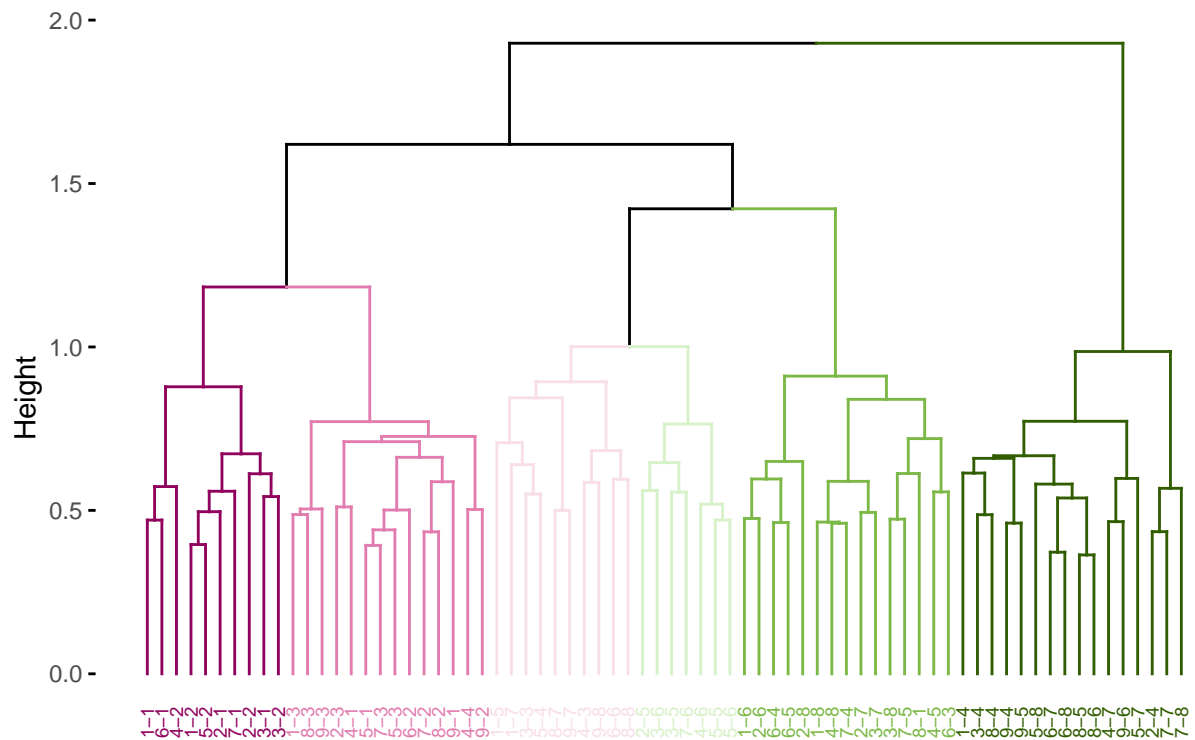
# Cluster Dendrogram



##

```
## [[103]]
```

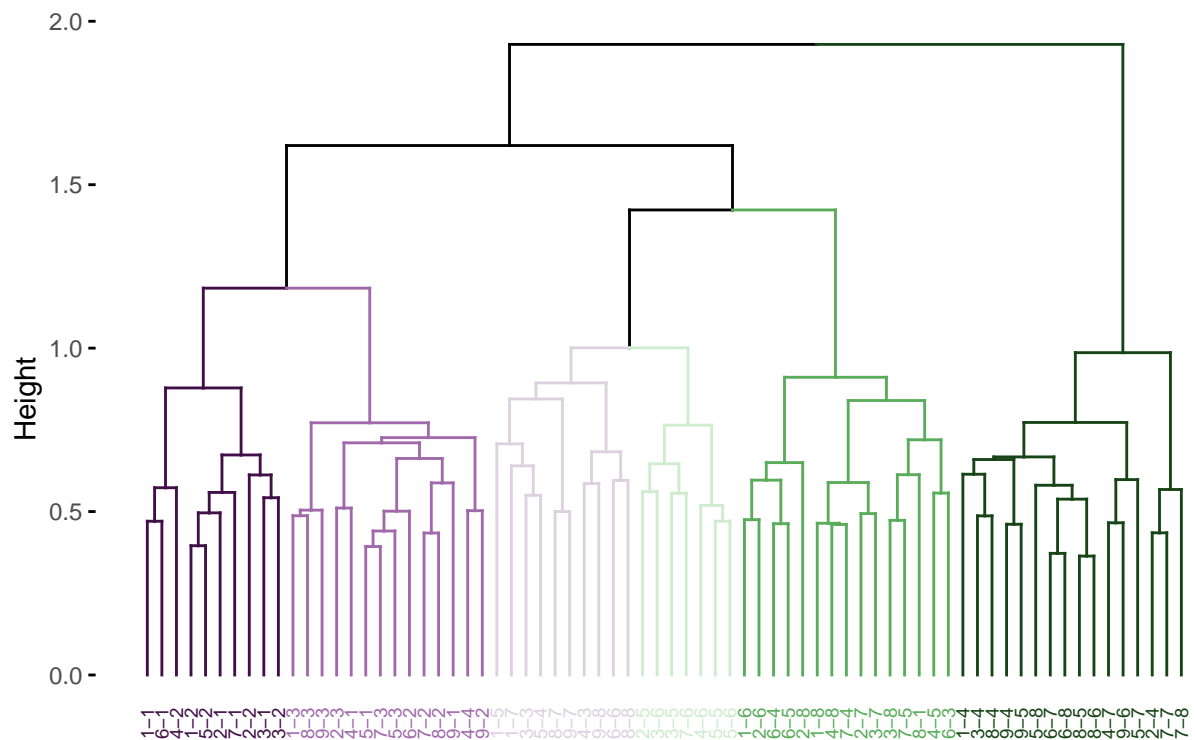
## Cluster Dendrogram



##

```
## [[104]]
```

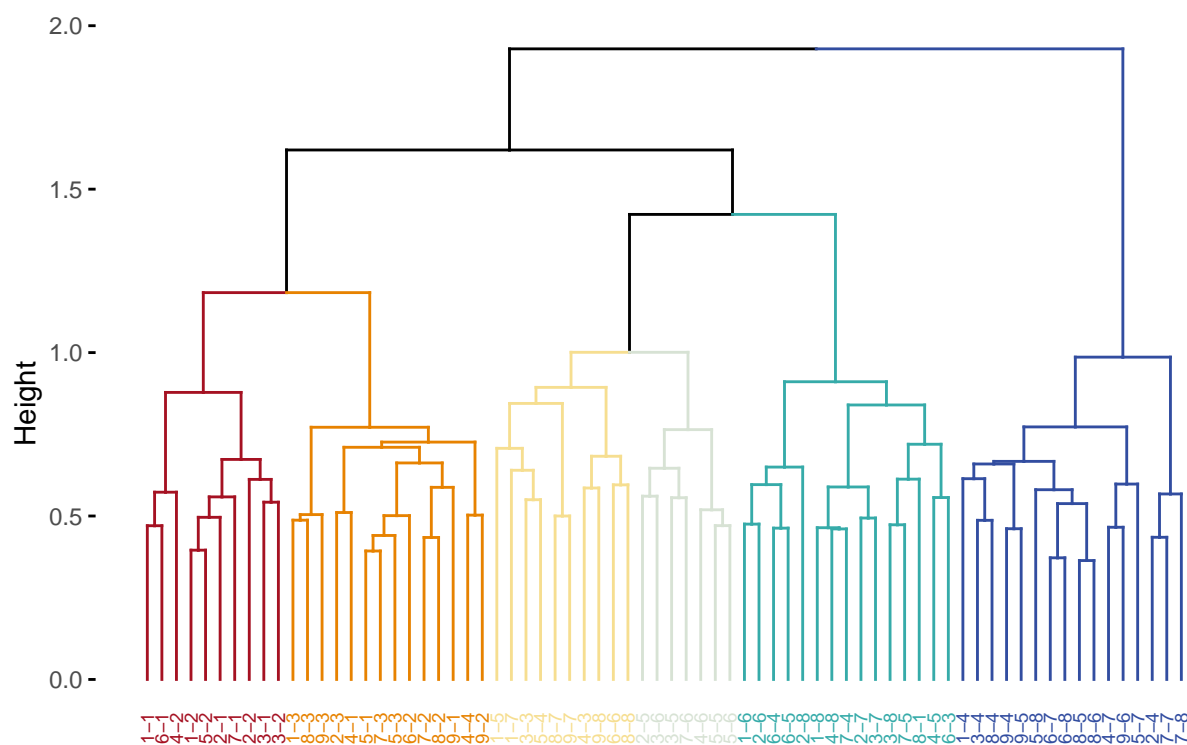
## Cluster Dendrogram







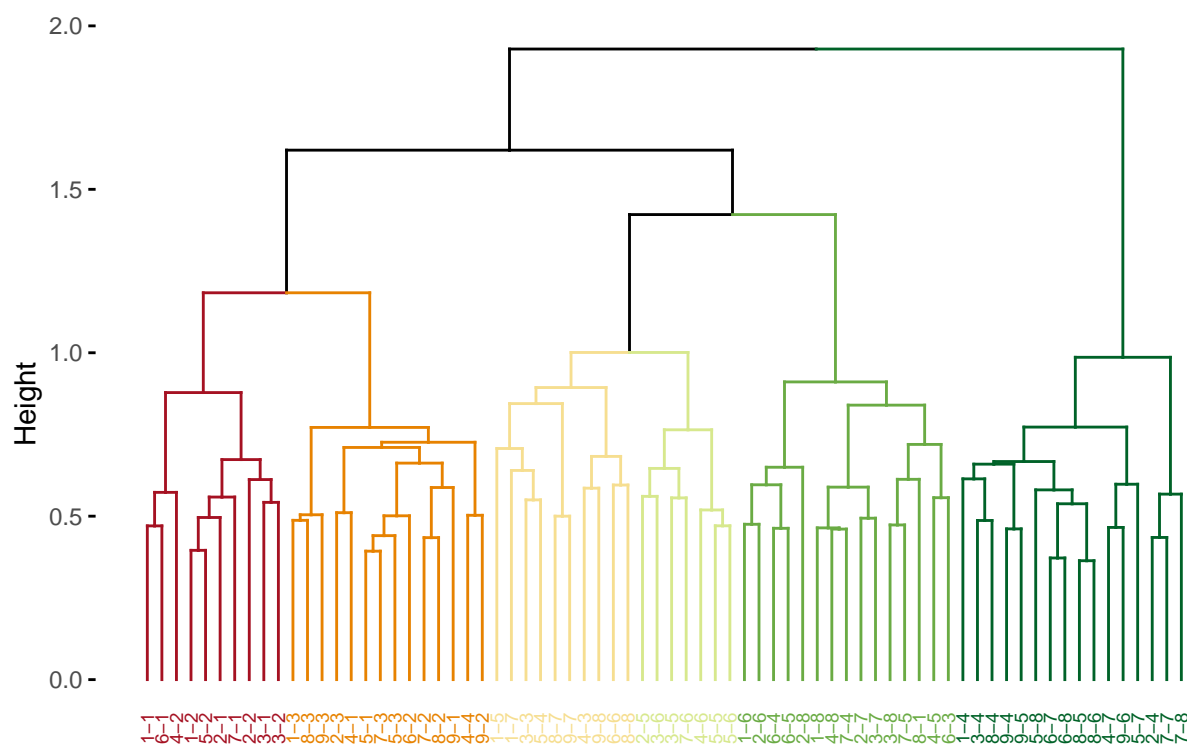
# Cluster Dendrogram



##

## [[107]]

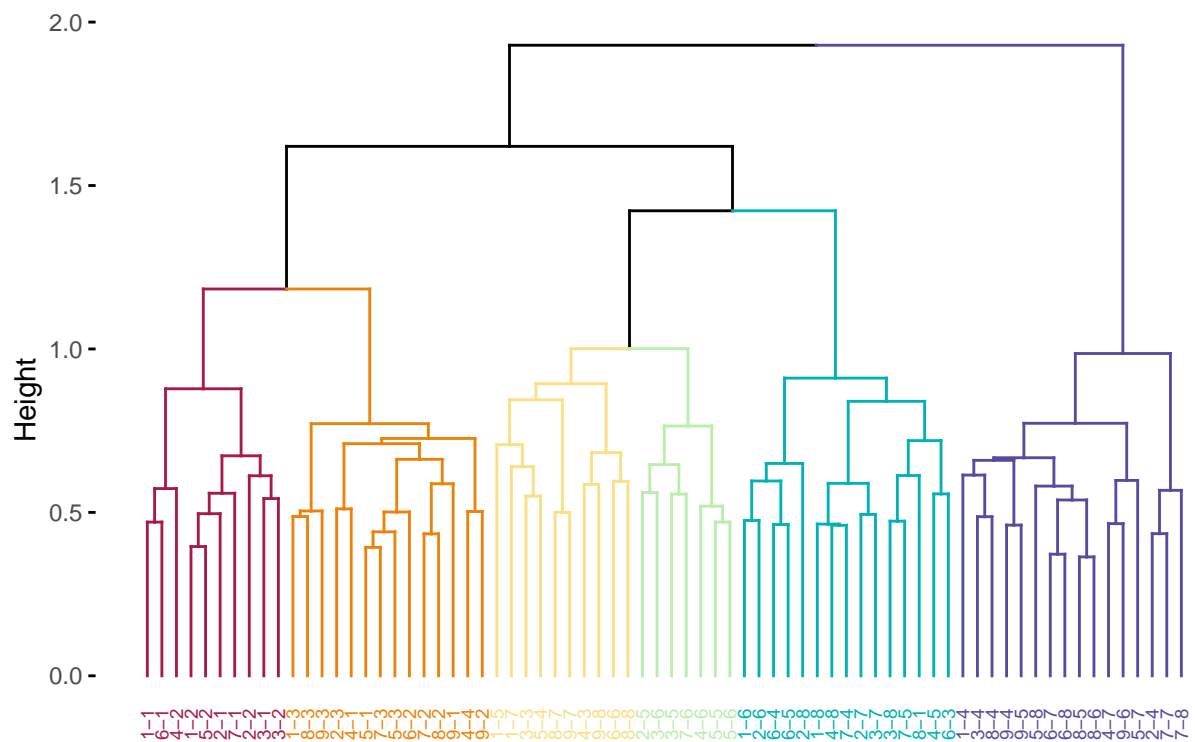
# Cluster Dendrogram



##

## [[108]]

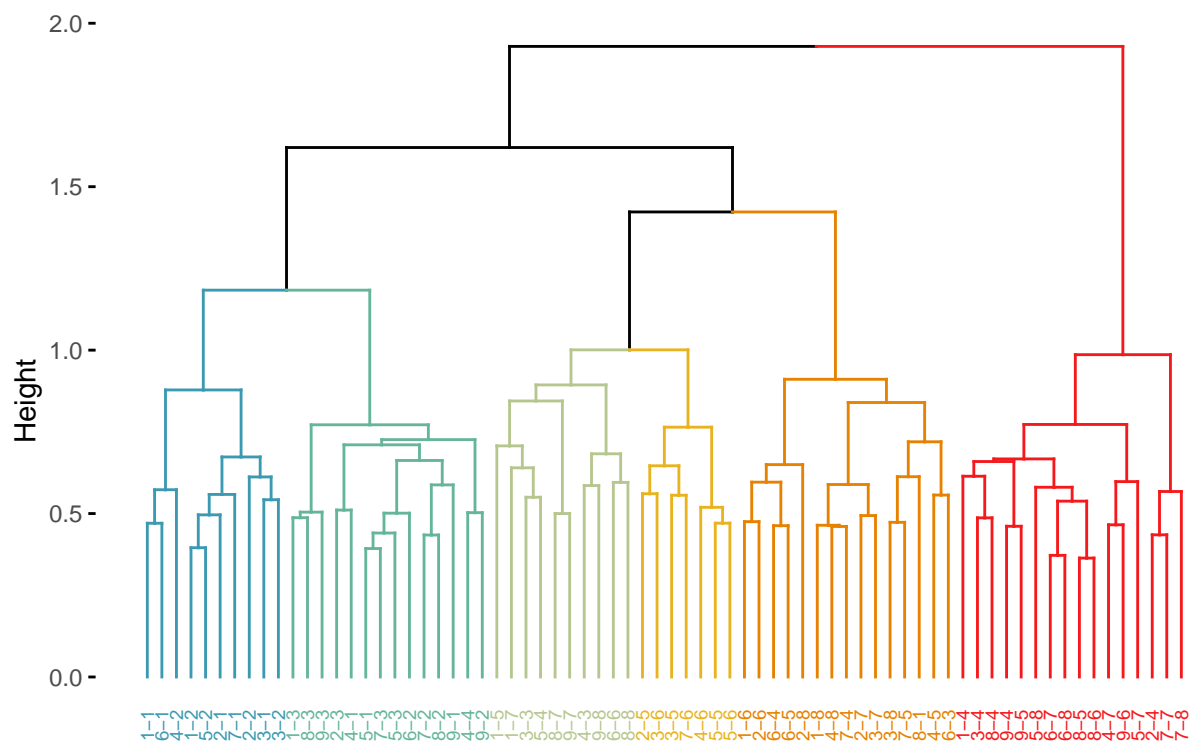
Cluster Dendrogram



##

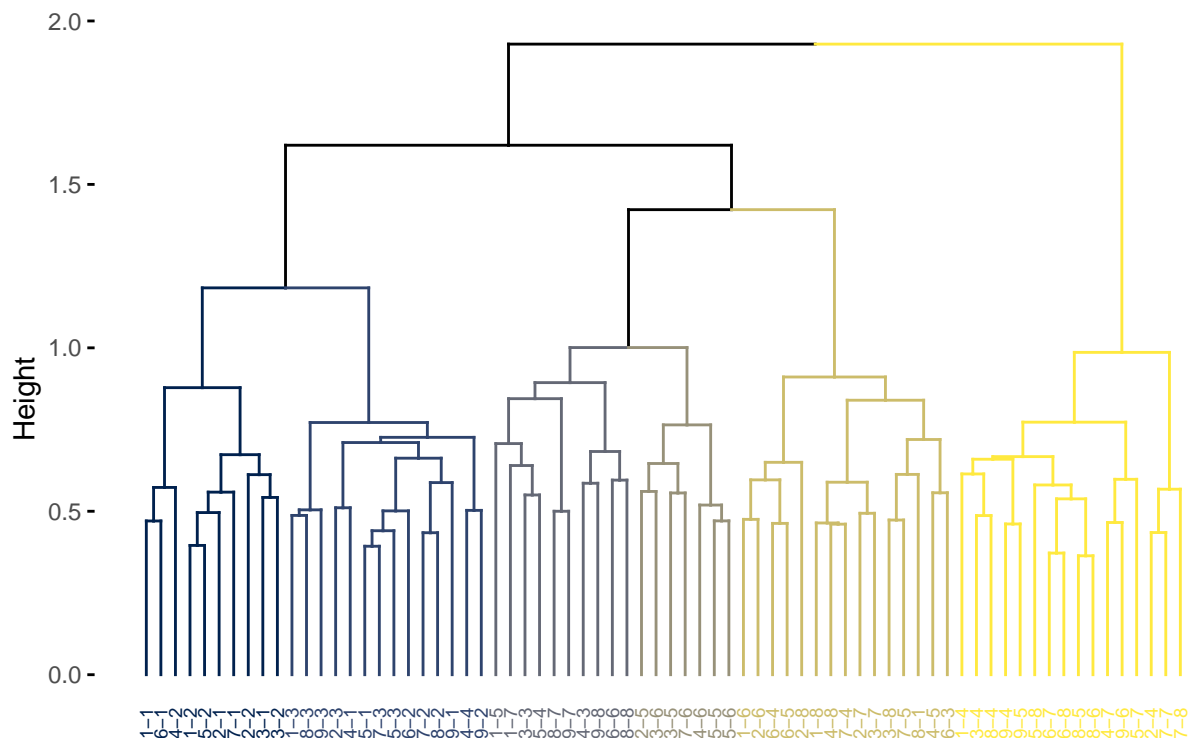
## [[109]]

Cluster Dendrogram



```
##
## [[110]]
```

### Cluster Dendrogram



```
dt <-
  df %>%
  mutate(cluster = factor(clust_6$cluster)) %>%
  group_by(cluster) %>%
  summarise_all(median) %>%
  # trasladamos el tibble para mejor visualización
  gather(especie, value, -cluster) %>%
  spread(cluster, value) %>%
  mutate(chk = `1` <= 4 & `2` <= 4 & `3` <= 4 & `4` <= 4 & `5` <= 4 & `6` <= 4 ) %>%
  filter(!chk) %>% select(-chk) %>%
  column_to_rownames('especie')
```

```
pander(dt)
```

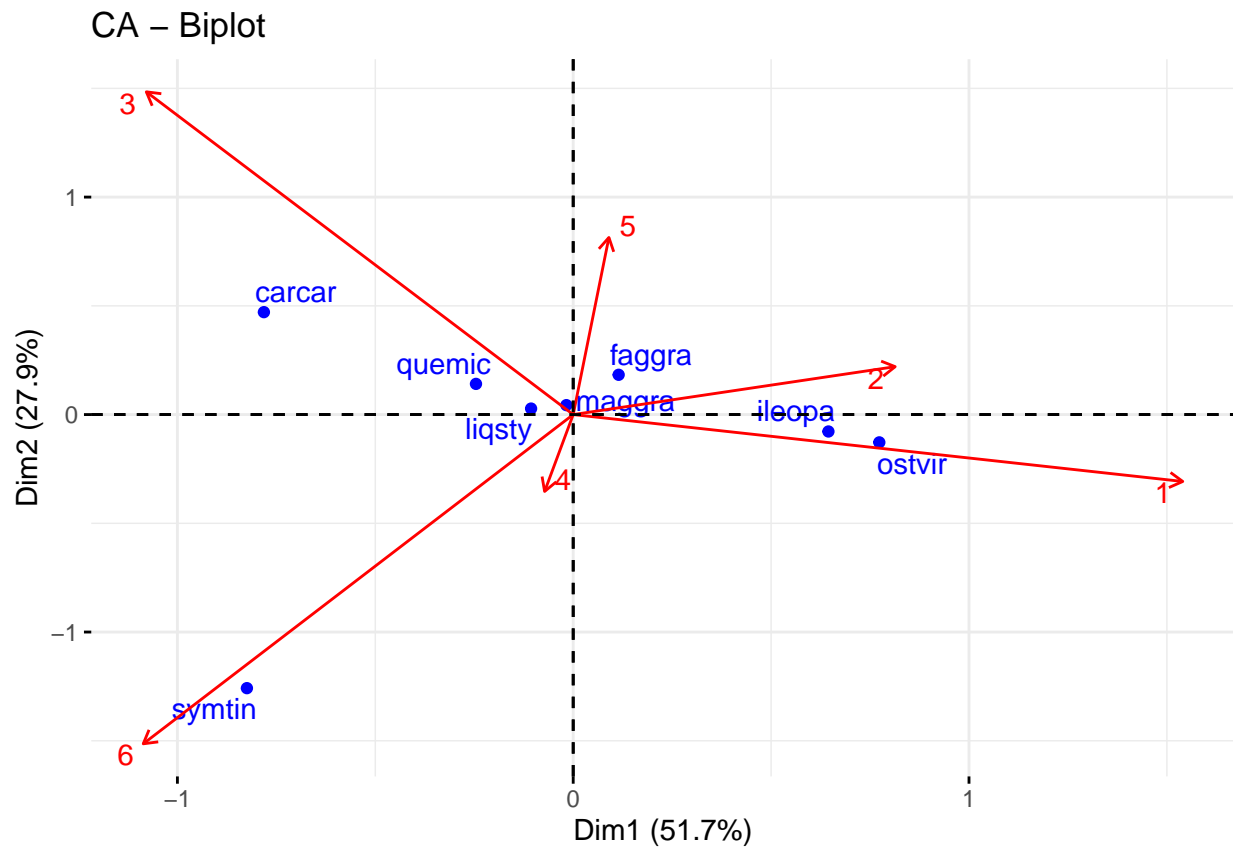
	1	2	3	4	5	6
carcar	0	1	21	5.5	4	9
faggra	5	9.5	4.5	4.5	14	6
ileopa	16.5	11	3	4	3	4
liqsty	5.5	7	9	9	4	8
maggra	2.5	5	3	1	5	5
ostvir	18.5	7	1.5	7	3	2
quemie	4	5	9.5	3	5	8
symtin	0.5	0	0	4	0	16

```
ca_df <-
  FactoMineR::CA(dt, graph = F)
```

```
pander(ca_df$eig)
```

	eigenvalue	percentage of variance	cumulative percentage of variance
dim 1	0.2906	51.68	51.68
dim 2	0.1571	27.94	79.62
dim 3	0.09164	16.3	95.92
dim 4	0.0157	2.792	98.71
dim 5	0.007258	1.291	100

```
fviz_ca_biplot(ca_df, map = "rowprincipal", arrows = c(F,T), repel = T)
```



## Ejercicio 2 (30 pt.)

Es posible utilizar la regresión logística para estudiar la discriminación entre dos grupos dado un conjunto de variables explicativas. Para más detalles podéis consultar el apartado 8.10 del libro de Manly.

(a)

Realizar un análisis discriminante con ayuda de la regresión logística con los datos de los 49 gorrones hembra después de una tormenta. Reproducir con todo detalle la tabla 8.6 de la

página 153 del libro de Manly.

```
load("gorriones.RData")

mod_lg <-
  glm(superviv ~ x1 + x2 + x3 + x4 + x5, gorriones, family = binomial(link = "logit"))

mod_NULL <-
  glm(superviv ~ 1, gorriones, family = binomial(link = "logit"))

# Para reporducir *con todo detalle* se puede utilizar la sintaxis de dplyr y crear la tabla
tbl <-
  tidy(mod_lg) %>%
  mutate(
    chi_2 = (estimate / std.error) ^ 2,
    Variable = c(
      "Constant",
      "Total length",
      "Alar length",
      "Length beak and head",
      "Length humerus",
      "Length keel of sternum"
    ),
    chi_2 = round(chi_2, 2)
  ) %>%
  mutate_at(vars(estimate:p.value), list( ~ round(., 3))) %>%
  select(
    Variable,
    ' estimate' = estimate,
    'Standard \\\ error' = std.error,
    'Chi-squared' = chi_2,
    'p-Value' = p.value
  )

tbl[1,4:5] <- "---"

pander(tbl, justify = "lcccc")
```

Variable	estimate	Standard \ error	Chi-squared	p-Value
Constant	13.58	15.87	–	–
Total length	-0.163	0.14	1.36	0.244
Alar length	-0.028	0.106	0.07	0.794
Length beak and head	-0.084	0.629	0.02	0.894
Length humerus	1.062	1.023	1.08	0.299
Length keel of sternum	0.072	0.417	0.03	0.864

(b)

Contrastar con un test  $\chi^2$  la significación de la regresión y explicar su resultado.

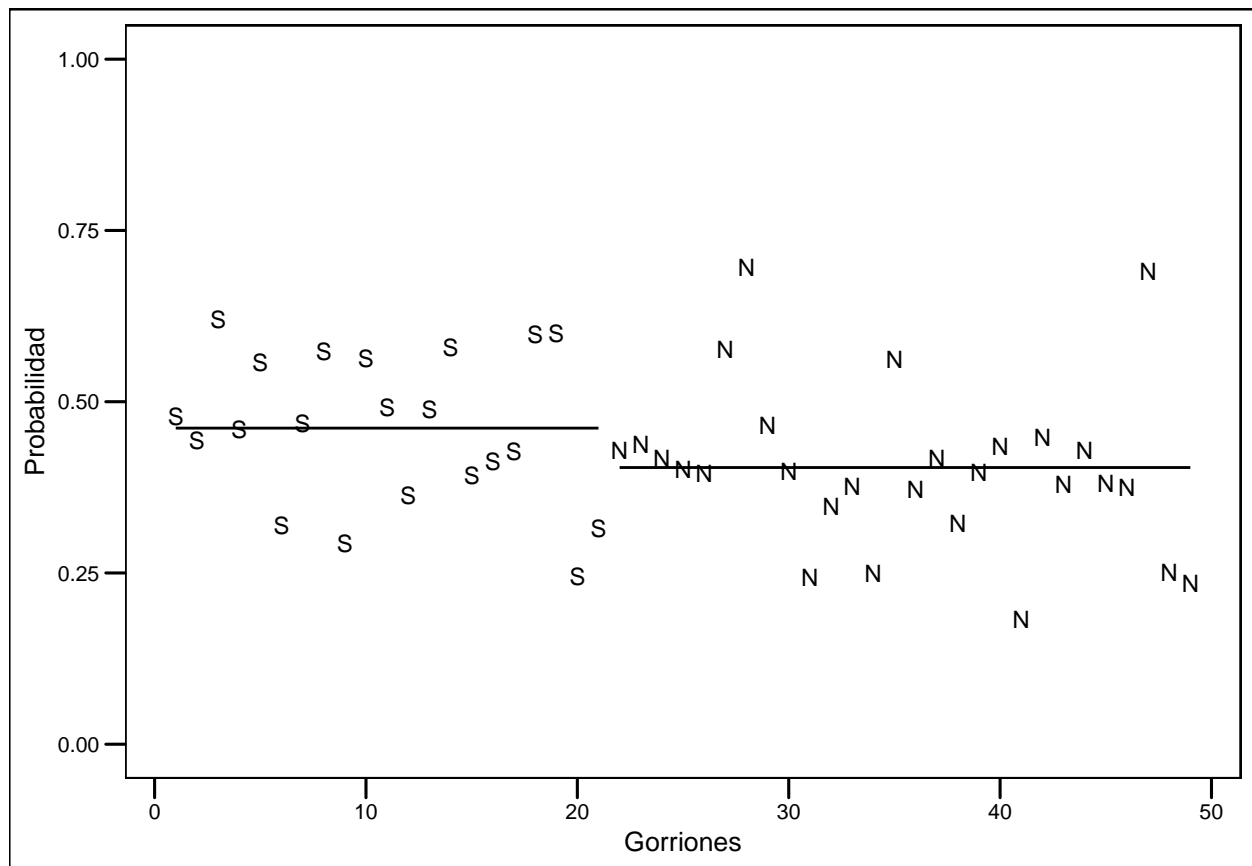
```
anova(mod_NULL, mod_lg, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: superviv ~ 1
## Model 2: superviv ~ x1 + x2 + x3 + x4 + x5
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         48      66.925
## 2         43      64.071  5   2.8536  0.7225
```

(c)

Realizar un gráfico como el de la figura 8.2 de la página 156 del libro de Manly pero con los datos de este ejercicio y valorar el resultado.

```
tibble(
  fitted = mod_lg$fitted.values,
  class = augment(mod_lg)$superviv
) %>%
  rowid_to_column() %>%
  group_by(class) %>%
  mutate(mean = mean(fitted)) %>%
  ggplot(aes(rowid, fitted, shape = class)) +
  geom_point(size = 3) +
  geom_line(aes(y=mean)) +
  scale_shape_manual(values = c("N", "S")) +
  theme_base(base_size = 10) +
  labs(x="Gorriones", y="Probabilidad") +
  scale_y_continuous(limits = c(0,1)) +
  guides(shape = F)
```



(d)

Calcular la tabla de confusión de la clasificación obtenida con la regresión logística.

```
ypred <-
  ifelse(
    predict.glm(mod_lg, type = "response") > 0.5,
    'S', 'N') %>%
  factor()

# para la matriz de confusión se puede utilizar la función de
# la librería `caret`
cm <- confusionMatrix(gorrones$superviv, ypred)

pander(cm$table)
```

	N	S
N	24	4
S	14	7

```
pander(cm$overall)
```

Table 5: Table continues below

Accuracy	Kappa	AccuracyLower	AccuracyUpper	AccuracyNull
0.6327	0.2025	0.4829	0.7658	0.7755

AccuracyPValue	McnemarPValue
0.9926	0.03389