Machine Learning no supervisado e ICG

Guillermo Bustos-Pérez

16/4/2020

¿Qué grupos de países, qué les caracteriza, a qué distancia?

Las preguntas que vamos a ntenatar contestar con los datos disponibles son: **¿cómo se agrupan los países siguiendo los indicadores de desarrollo?**, **¿qué les caracteriza?**, **qué distancia hay entre ellos**.
No tenemos grupos pre-definidos y lo que queremos es explorar pocedimientos de agrupación y caracterizar esos grupos. Esto hace que sea muy adecuado para aplicar principios de **"Machine Learning no Supervisado"**.
En este caso vamos a emplear los valores de cada país en los 12 pilares del **Índice de Competitividad Global** definidos por el **Foro Económico Mundial**.

Empezamos por quedarnos con los datos correspondientes a la variable "Pillar" y eliminado las columnas que no aportan nada (unidades de la serie, tipo de serie, o la edición).

```
Pillars <- Comp_2019 %>% filter(`Series type` == "Pillar") %>%
  select(-c(`Series units`, `Series type`, Edition, `Series name`, Attribute))
```

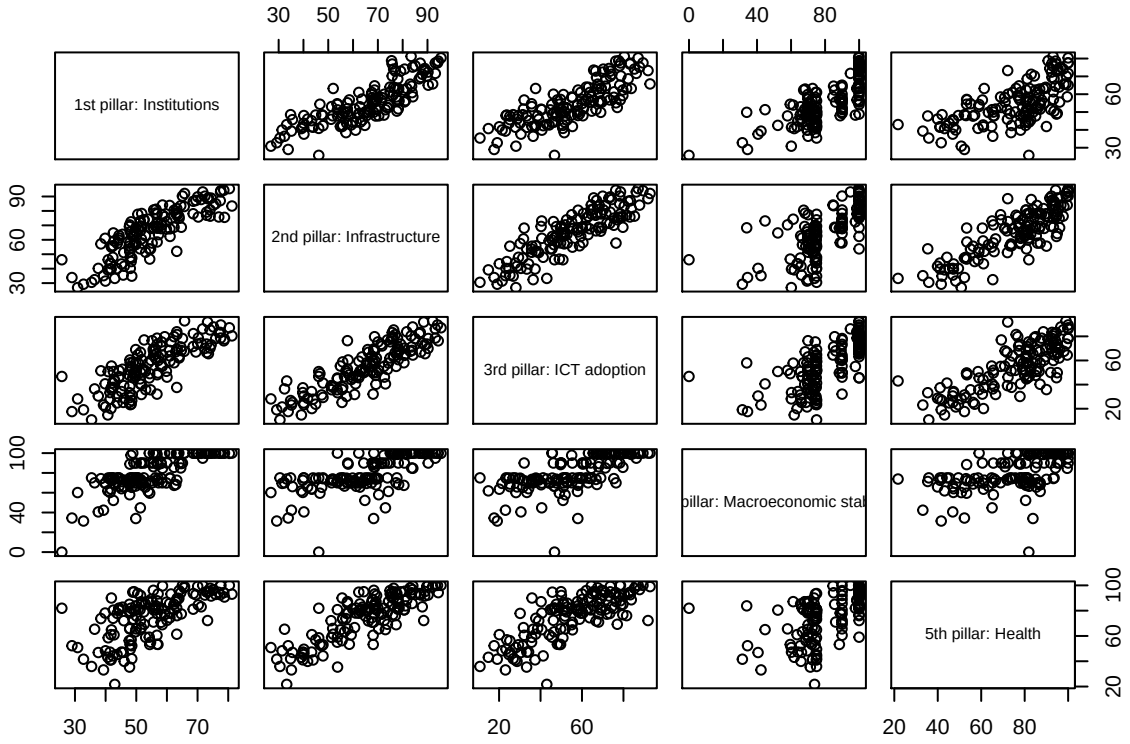
Ahora tansponemos el data frame para que cada pillar sea una columna y cada país una fila. Esto permite que el data frame sea analizable

```
# Transponer el data frame para que sea analizable
n <- Pillars[,1]
Pillars <- Pillars[,-1]
Pillars <- t(Pillars)
colnames(Pillars) <- n

Pillars <- as.data.frame(Pillars)

# Comprobar cinco primeras filas y dos primeras columnas
Pillars[1:5, 1:2]
```

```
##                     1st pillar: Institutions 2nd pillar: Infrastructure
## Angola                                37.61545                   40.19701
## Albania                               51.87481                   57.69324
## United Arab Emirates                  73.25884                   88.48554
## Argentina                             49.85289                   68.29266
## Armenia                               56.24956                   69.40846
```

Una rápida exploración visual de la relación entre los cinco primeros pilares muestra que entre ellos hay un fuerte grado de correlación . Esto implica que al realizar un análisis de componentes principales (PCA) el primer componente poseerá la mayor parte de la varianza

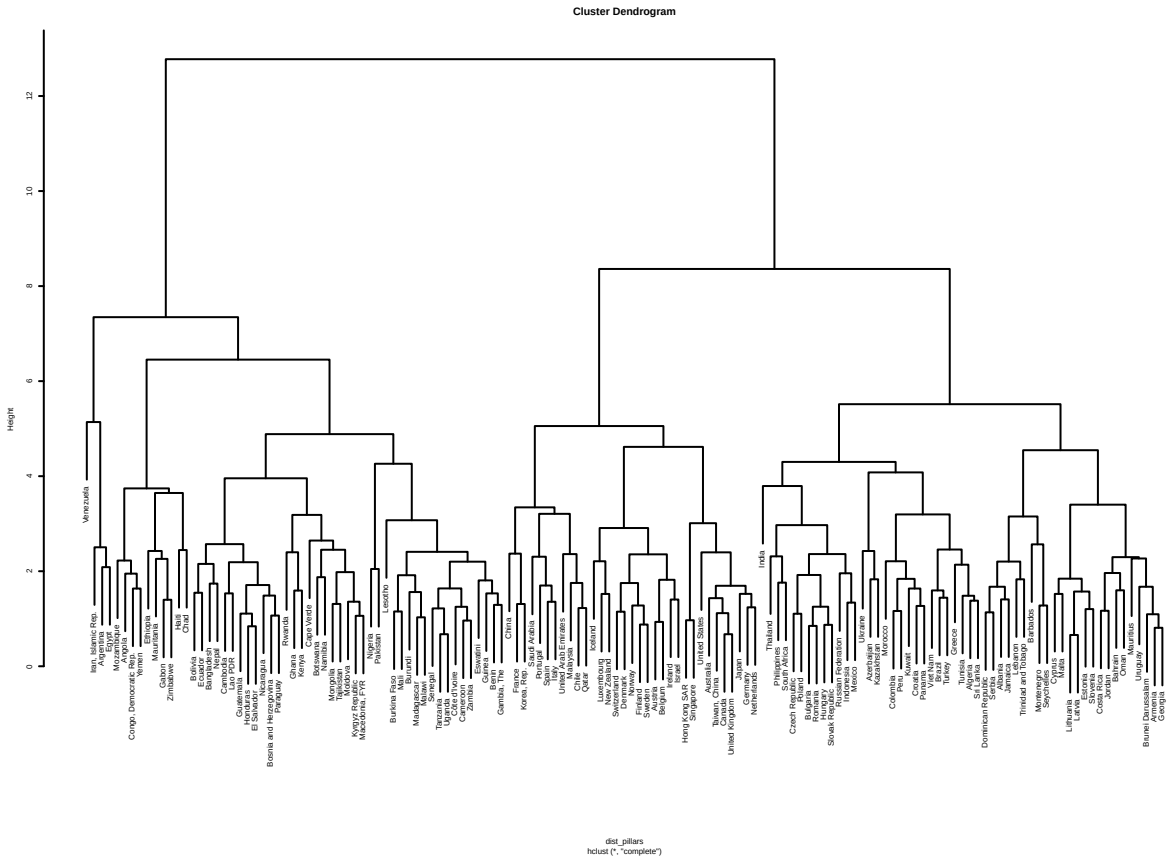
```r
pairs(Pillars[, 1:5])
```



Clústeres jerárqucos y determinación del número de clústeres

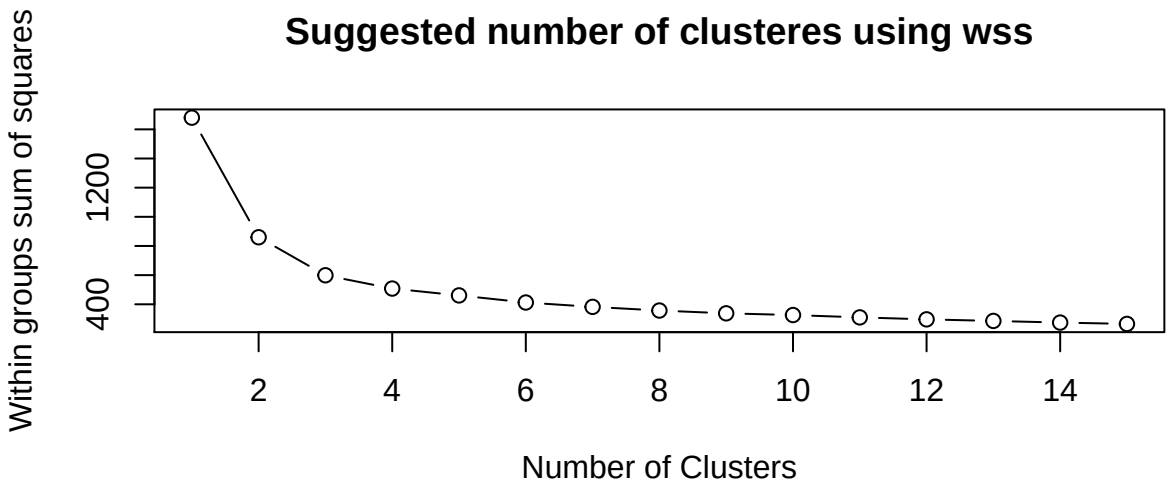
En este caso para realizar el clústering jerárquico basta con escalar/normalizar los datos y computar la distancia eucleidea. A continuación se computa el cluster jerárquico y se genera la representació en forma de dendograma. También aprovechamos para guardar la atribución de clústeres por país, ya que resultará útil más adelante.

```r
Pillars_Scaled <- Pillars %>% scale()

dist_pillars <- dist(Pillars_Scaled, method = "euclidean")

# hc_pillars
hc_pillars <- hclust(dist_pillars)

# Dendograma del HC
par(cex=0.25, mar=c(5, 8, 4, 1))

plot(hc_pillars, labels = Pillars$Countries, horiz = T)
```

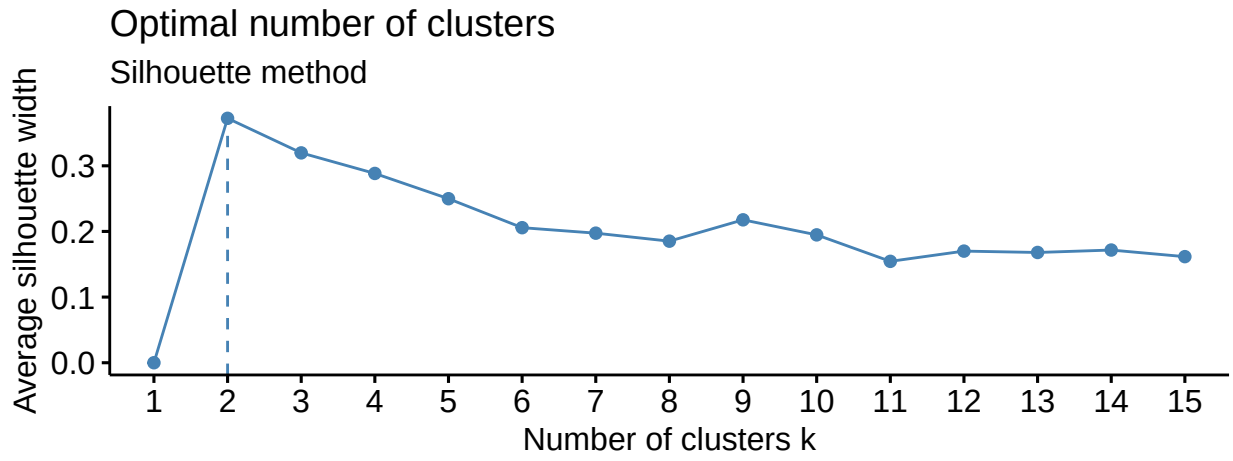


La representación en forma de dendograma permite observar la existencia de al menos 7 clusteres. Aprovechamos para observar las **sugerencias de números de clústeres** usando la **wss y el método del average silhouette**.

```r
# wss para determinear el número de clústeres
wss <- 0

for (i in 1:15) {
  km.out <- kmeans(Pillars_Scaled, centers = i, nstart = 20)
  wss[i] <- km.out$tot.withinss
}

plot(1:15, wss, type = "b",
     xlab = "Number of Clusters",
     ylab = "Within groups sum of squares",
     main = "Suggested number of clusteres using wss")
```




```
# Average silhouette
fviz_nbclust(Pillars_Scaled, kmeans,
                     method = "silhouette", k.max = 15) +
  labs(subtitle = "Silhouette method")
```



Es de señelar que estos dos métodos buscan el número de clústeres que maximizan la distancia. Basándonos en el dendograma esto corresponde a dos grupos. Sin embargo, esto conlleva una pérdida de resolución de los datos ya que podemos ver que dentro de esos dos grupos existen más grupos bien definidos. Por consiguiente **los resultados del wss y average silhouette son indicativos, pero no determinantes**. Resulta preferible emplear la sugerencia de siete clústeres observados en el dendograma. Guardamos la asignación de clústeres.

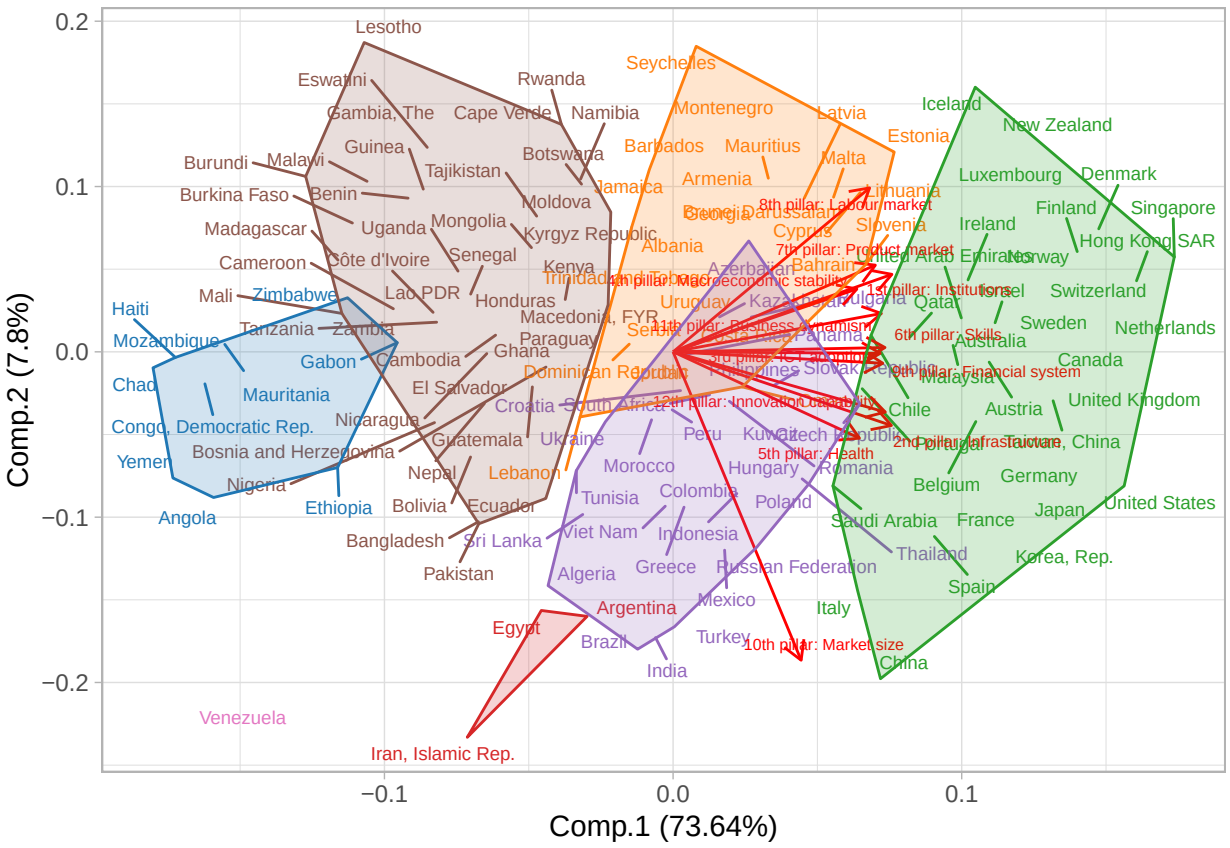
```
# Guardar los clusters en un vector
set.seed(1234)
cluster_assignaments <- cutree(hc_pillars, k = 7)

Pillars <- cbind(Pillars, cluster_assignaments)
Pillars <- Pillars %>% mutate(
  cluster_assignaments = as.factor(cluster_assignaments))
```

```r
# Salvar la atribución de clústeres
Cluster_Countries <- cbind(
  rownames(Pillars_Scaled), cluster_assignaments) %>%
  as.data.frame()
```

PCA usando atribución de grupos

Esta suele ser la parte más agradable del análisis de los datos. Empleamos un PCA con los datos escalados y la atribución de clústeres generada a partir del Clúster Jerárquico. El biplot y las bivariantes convexas permiten observar las agrupaciones de países.

```r
# Autoplot el PCA según los grupos del clúster jerárquico
autoplot(stats::princomp(Pillars_Scaled), x = 1, y = 2, data = Pillars,
         colour = "cluster_assignaments",
         shape = FALSE, frame = TRUE,
         loadings = T, loadings.label = T,
         loadings.label.repel= T, loadings.label.size = 2,
         label.size = 2.5, label.repel = T) +
  scale_color_d3() + scale_fill_d3() +
  theme_light() +
  theme(
    legend.position = "none"
  )
```



Los resultados están en línea con lo esperado. El **Primer Componente Principal supone más del 70% de la varianza** como **consecuencia de la alta correlación entre los diferentes pilares** y los loadings tienden a distribuirse positivamente con respecto a este. Los países con valores más elevados en el Indice de Competitividad se localian en el margen derecho del biplot y no hay sorpresa en los países que aparecen: Estados Unidos, Alemania, Dinamarca, Países Bajos, Canada, Hong Kong, etc. En el margen izquierdo se localizan países que tienen *"dificultades"* o que directamente son considerados estados fallidos

Países y Tecnologías de la información

El informe del **Índice de Competitividad Global** da la oportunidad de explorar los valores de los indicadores correspondientes a los **pilares relacionados con adopción de las tecnologías de la información y desarrollo del mercado laboral**. Estos son principalmente **cuatro pilares**:

- 3er pilar: adopción de ICT
- 6º pilar: habilidades/formación de la población
- 8º pilar: mercado de trabajo
- 12º pilar: capacidad de innovación

Dado que en el apartado anterior hemos visto una clara diferencia entre países, podemos limitar la muestra a países del entorno europeo, determinados países de Asia, y una selección de países de Próximo Oriente. A su vez filtramos para retirar los Pillares, sub.-pilares y sub-sub-sub pilares, ya que se tratan medidas promedio de los indicadores que incluyen

```
# Filtrar para obtener los pilars deseados
TICs <- Comp_2019 %>%
  filter(Pillar == "3rd pillar: ICT adoption" |
         Pillar == "6th pillar: Skills" |
         Pillar == "8th pillar: Labour market")%>%
  filter(`Series type` == "Indicator") %>%
  select(-c(Edition, `Series type`, Pillar, `Series units`, Attribute))


# Transponer el data frame
n <- TICs$`Series name`
TICs <- as.data.frame(t(TICs[,-1]))

colnames(TICs) <- n
TICs$Countries <- rownames(TICs)

# Filtrar para obtener los países de los clusteres 3 y 5
Cluster_Countries <- Cluster_Countries %>% filter(cluster_assignaments == 3 |
                                cluster_assignaments == 5)

TICs <- subset(TICs, Countries %in% Cluster_Countries$V1)

# Observar dimensiones y nueva estructura del data frame
dim(TICs)
```

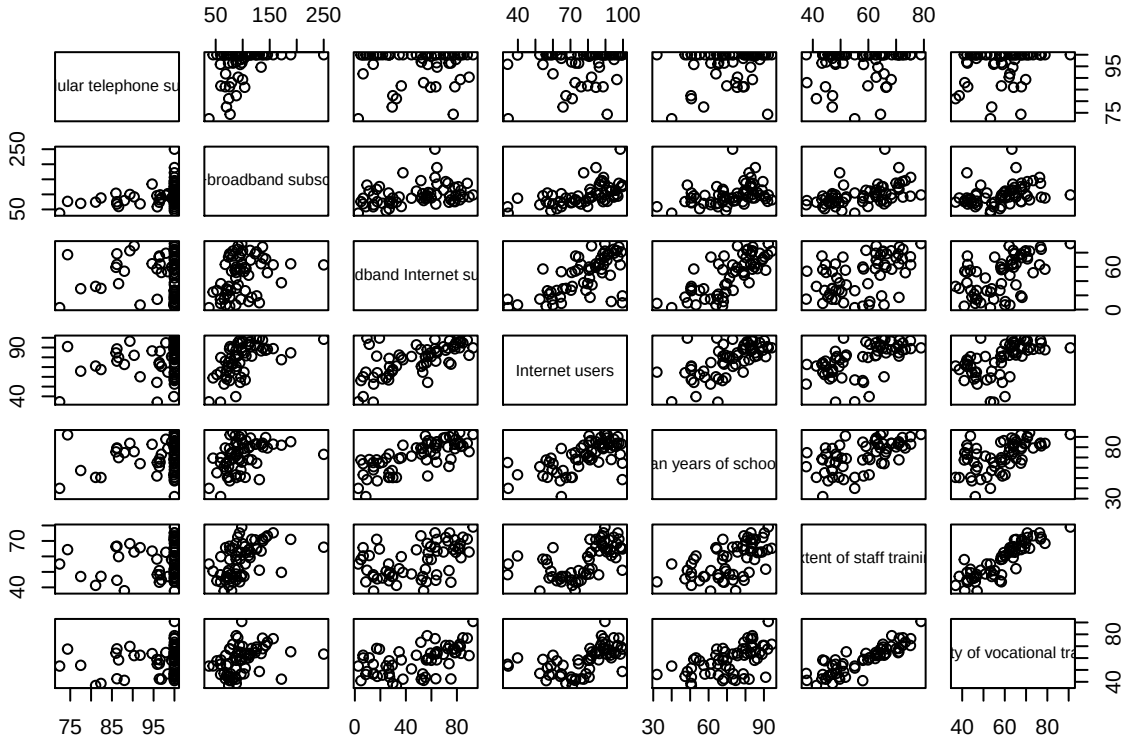
```
## [1] 62 29
```

En este caso no tenemos Na's. SI hubiera Na's una opción sería imputar los datos usando el algoritmo del vecino más cercano (KNN), pero en este caso vamos a prescindir las posibles columnas ya que disponemos de 29 para realizar el análisis.

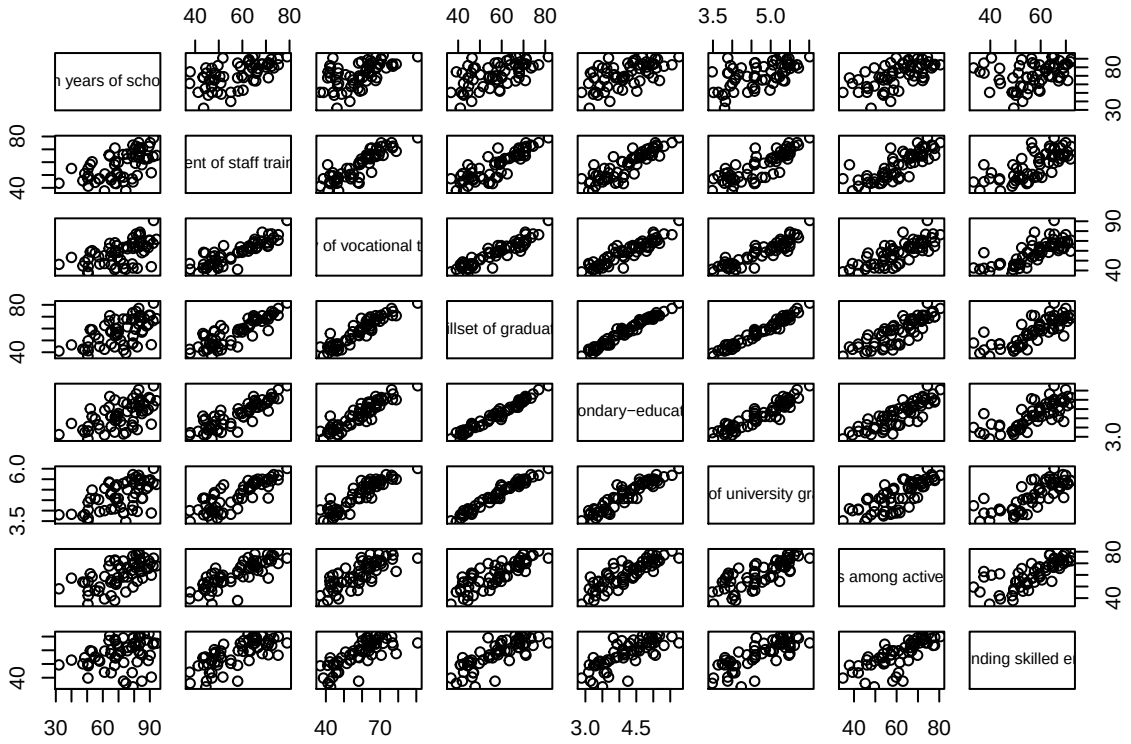
```
# Retirar las columnas que tengan Na
TICs <- TICs %>%
  select_if(~ !any(is.na(.)))
```

Nuevamente podemos emplear la fución pairs() para ver las correlaciones entre las diferentes variables. Algunas variables presentan un fuerte grado de correlación por cuestiones evideentes, como es el caso de la calidad de los docentes (*Quality of vocational training*), habilidad de los graduados (*Skillset of graduates*), habilidad de los graduados de educación secundaria (*Skillset of secondary-education graduates*), y habilidades de los graduados universidtarios (*Skillset of university graduates*).

```
pairs(TICs[1: 7])
```




```
pairs(TICs[5: 12])
```




```
cor(TICs$`Skillset of graduates`,
    TICs$`Skillset of secondary-education graduates`,
    method = "pearson")
```

```
## [1] 0.9841091
```

```
cor(TICs$`Skillset of university graduates`,
    TICs$`Skillset of secondary-education graduates`,
    method = "pearson")
```

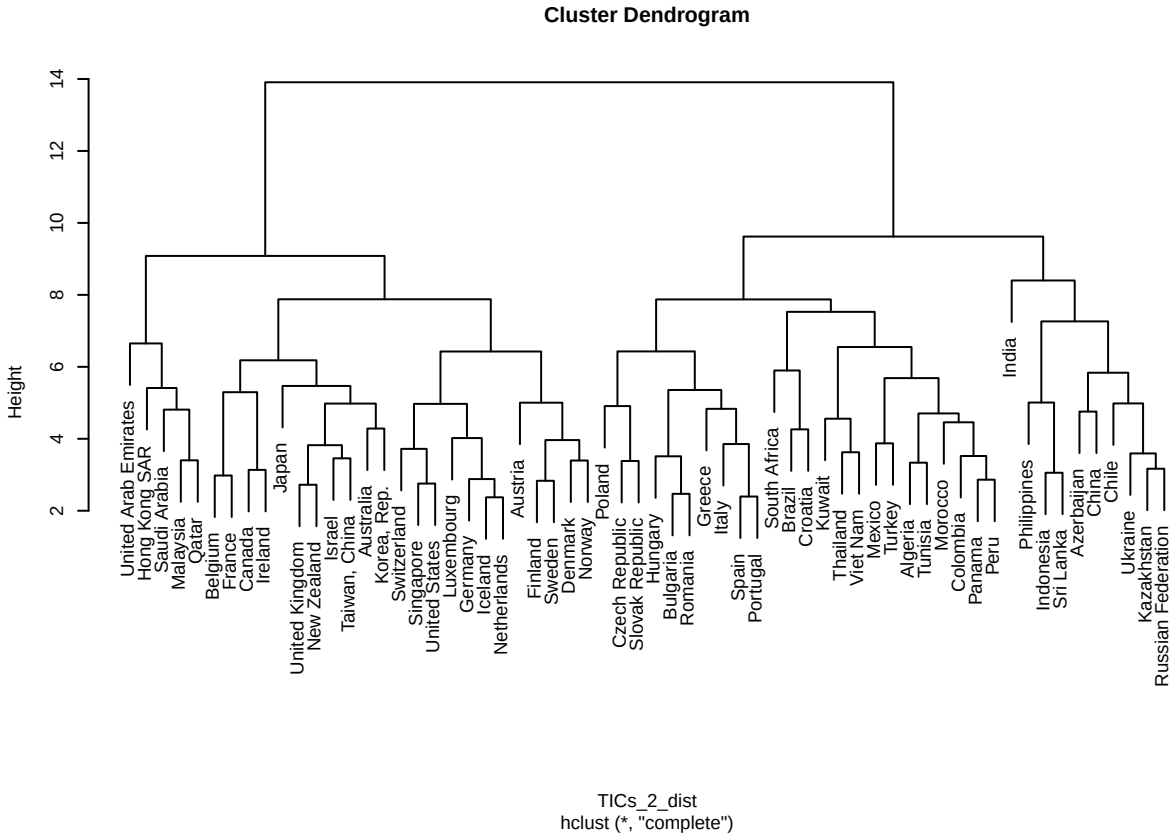
```
## [1] 0.9335985
```

Número de clústeres y PCA

El procedimiento es simialr al previamente visto. Es necesario escalar los datos, calcular la distancia euclídea entre los diferentes casos, generar el clúster jerárquico y representar el dendograma.

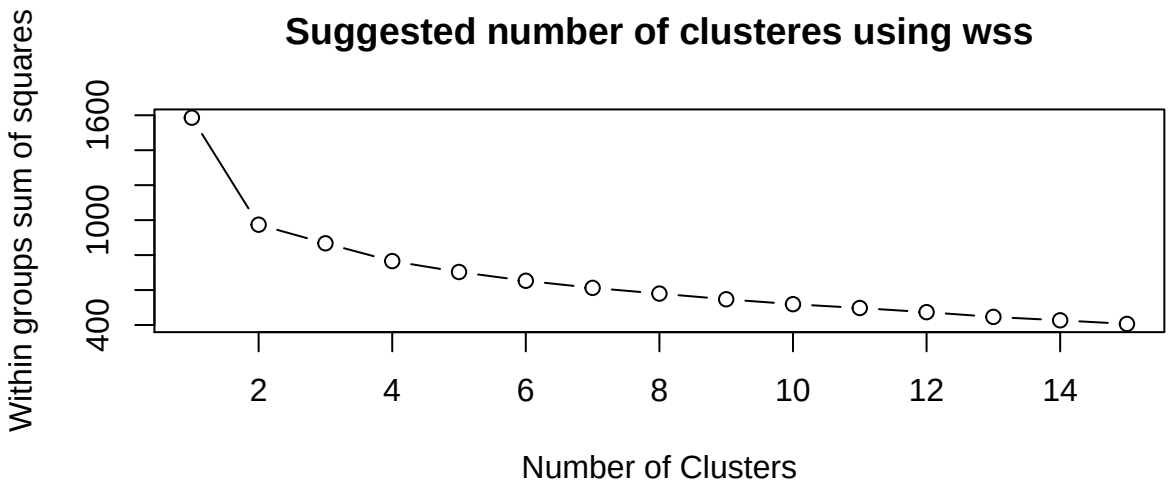
```
# Retirar la columna con el combre de los países y escalar
TICs_2 <-
  TICs %>% select(-c(Countries)) %>% scale() %>%
  as.data.frame()

# calcular la distancia euclídea
TICs_2_dist <- dist(TICs_2, method = "euclidean")

# Convertir en clúster jerárquico
```

```r
hc_TICs <- hclust(TICs_2_dist)

# representar el dendograma
par(cex=0.55)
plot(hc_TICs, labels = TICs$Countries)
```

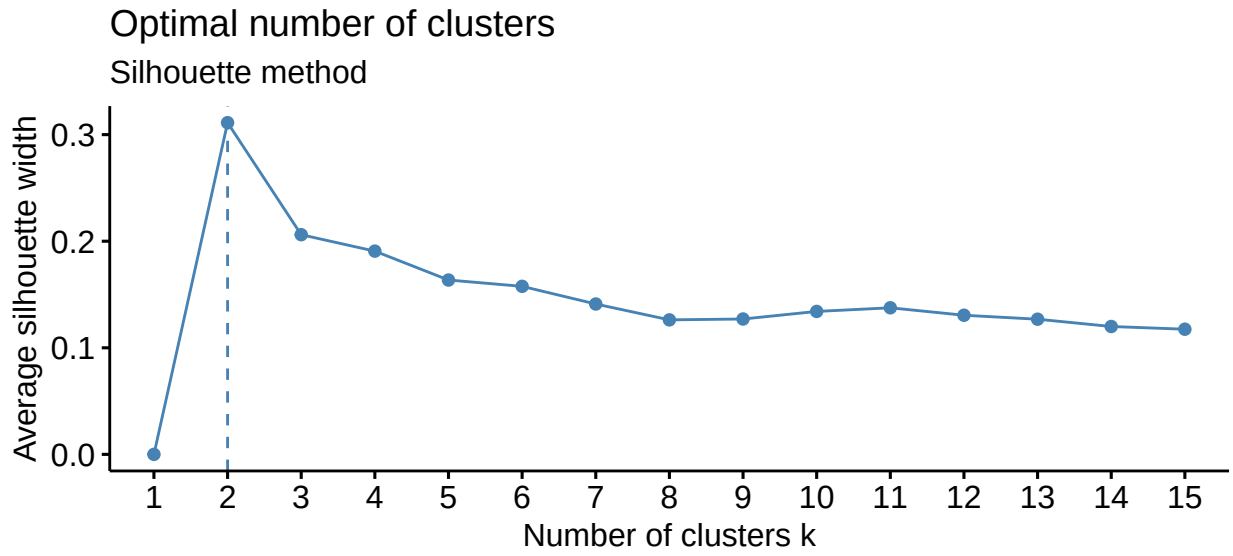


A su vez podemos acompañar la determinación del número de clústeres empleando la **wss y el método del average silhouette**.

```r
# wss para determinear el número de clústeres
wss <- 0

for (i in 1:15) {
  km.out <- kmeans(TICs_2, centers = i, nstart = 20)
  wss[i] <- km.out$tot.withinss
}

plot(1:15, wss, type = "b",
     xlab = "Number of Clusters",
     ylab = "Within groups sum of squares",
     main = "Suggested number of clusteres using wss")
```




```
# Average silhouette
fviz_nbclust(TICs_2, kmeans,
                      method = "silhouette", k.max = 15) +
  labs(subtitle = "Silhouette method")
```

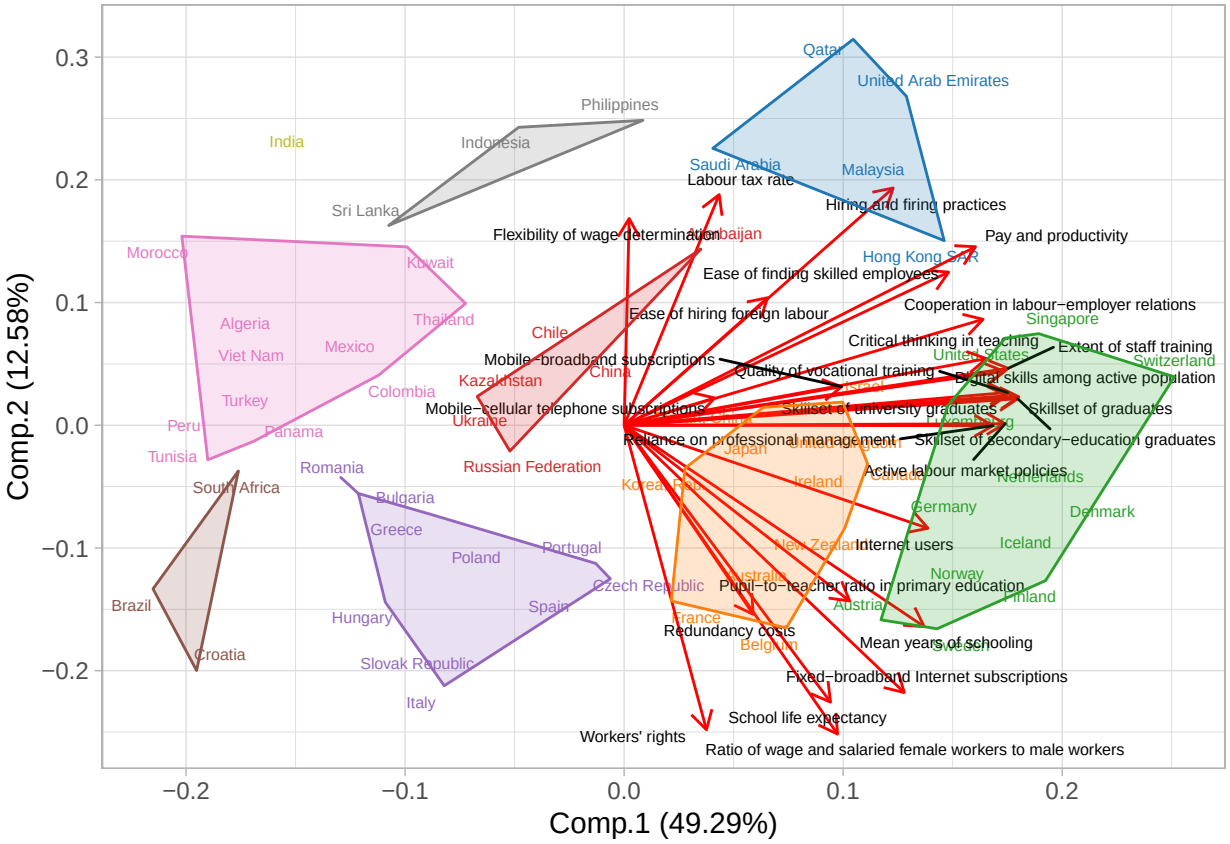


Nuevament hay una confrontación entre el número de clústeres sugerido por los métodos del average silhouette y la wss, y lo observado en el dendograma, donde podemos ver más subgrupos dentro de la sugerencia de doos grupos. Dado que no queremos perder resolución en el análisis vamos a optar por crear los grupos a partir de una altura en el dendograma.

```
# Asignar clusteres según una altura en el dendograma
cluster_assignaments <- cutree(hc_TICs, h = 7)
TICs <- TICs %>% mutate(
  cluster_assignaments = as.factor(cluster_assignaments))
```

Una vez tenemos establecido el número de clústeres podemos realziar un biplot del PCA con las vibariantes convexas de acuerdo a la attribución de clústeres.

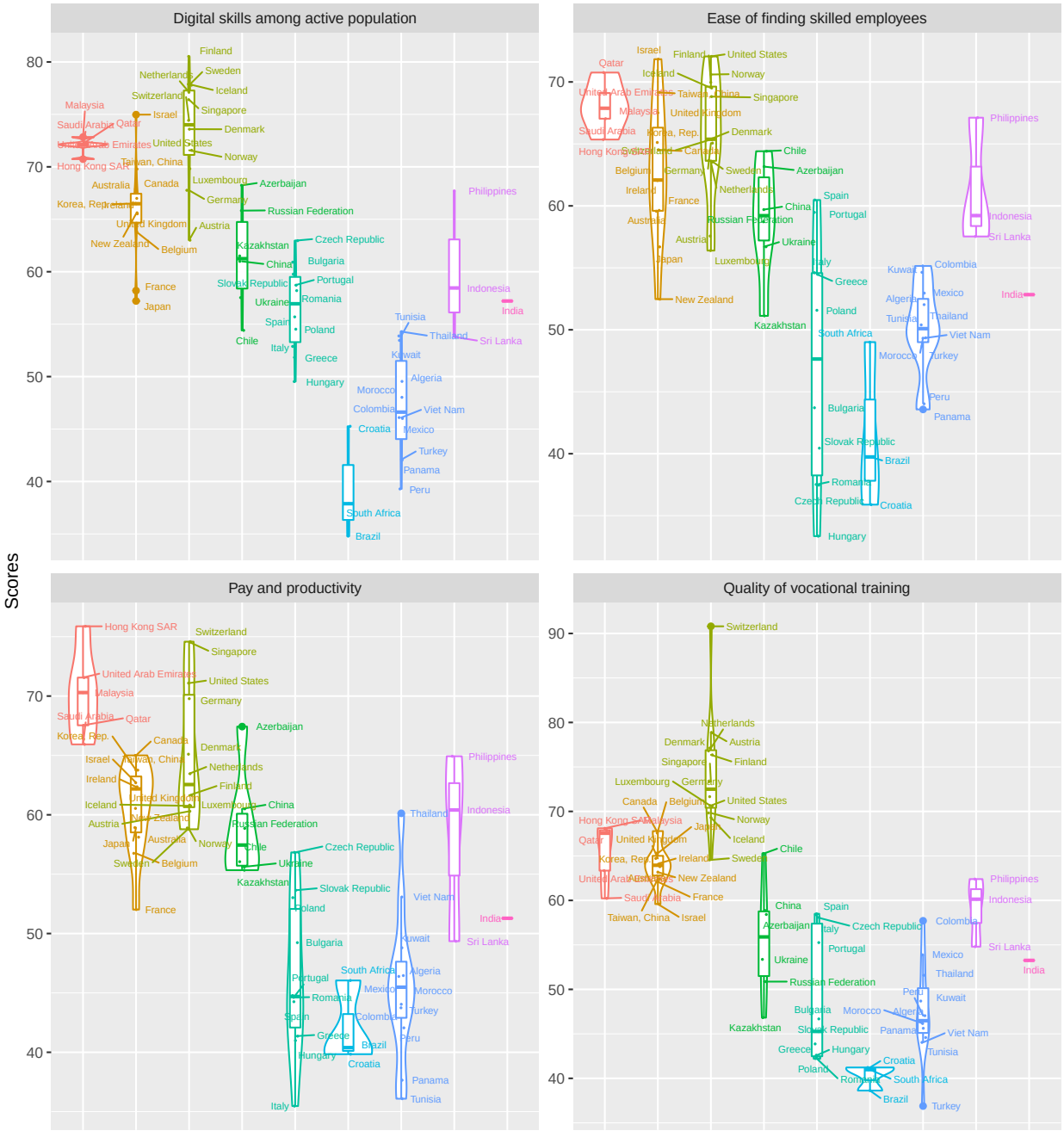
```
# Autoplot PCA
autoplot(stats::princomp(TICs_2), x = 1, y = 2, data = TICs,
         colour = "cluster_assignaments",
         shape = FALSE,
         frame = TRUE,
         loadings = T,
         loadings.label = T,
         loadings.label.size = 2.25,
         loadings.label.colour = "black",

         loadings.label.repel= T,
         label.size = 2.25,
         label.repel = T) +
  scale_color_d3() +
  scale_fill_d3() +
  theme_light() +
  theme(
    legend.position = "none"
  )
```



Visualizar puntuaciones de países dentro de sus clústeres

De momento hemos visto que dentro de los países más desarrollados hay grupos en lo referente a la adopción de nuevas tecnologías, formación y mercado laboral. Hemos agrupado los países según estos grupos. Sin embargo, ahora queremos responder a preguntas concretas. Por ejemplo, queremos comparar los valores obtenidos por los países en determinados indicadores, y queremos hacer esa comparación con respecto a los países de su mismo clúster y los países del resto de clústeres. En este caso el paquete **tidyverse** vuelve a ofrecer una manera flexible de seleccionar y manipular la organización de las variables para podder representarlas gráficamente.

```
TICs %>% select(
  Countries,
  cluster_assignaments,
  `Ease of finding skilled employees`,
  `Digital skills among active population`,
  `Quality of vocational training`,
  `Pay and productivity`) %>%
  pivot_longer(
    `Ease of finding skilled employees`: `Pay and productivity`,
  names_to = "Indicator",
  values_to = "Scores"
) %>%
  ggplot(aes(cluster_assignaments, Scores, color = cluster_assignaments))+
  geom_violin() +
  geom_boxplot(width = 0.2) +
  labs(caption = "Data: 2019 Global Competitiveness Index") +
  geom_jitter(size = 0.5,
    shape = 16, position = position_jitter(0.05)) +
  geom_text_repel(aes(label = Countries),
                  na.rm = TRUE, hjust = -0.3, size = 2.25) +
  facet_wrap(~Indicator, scales = "free", ncol = 2) +
  xlab("Countries grouped according to HC on scores from ICT adoption, Skills and Labour Market") +
  theme(legend.position = "none",
        axis.text.x = element_blank(),
        axis.ticks.x=element_blank(),
        plot.caption = element_text(size = 9, color = "black"))
```


Digital skills among active population
Ease of finding skilled employees
Pay and productivity
Quality of vocational training
Scores
Countries grouped according to HC on scores from ICT adoption, Skills and Labour Market
Data: 2019 Global Competitiveness Index