

Chaine Logistique Durable

Conclusión del Juego Tomates y Sal - con R-Cran

PALMA Ricardo R. - Universidad Nacional de Cuyo
<rpalma@uncu.edu.ar>

YASSINE Adnan
IDEL - Universite du Havre
<adnan.yassine@univ-lehavre.fr>

October 12, 2017

1 Nutriendo de información al sistema

1.1 Captura de datos

Cargando datos desde la línea de comando

```
> library(readr)
> BSC <- read_csv("~/R-Cran/Juego_Tomate.csv")
> BSC [c(1:5 ,70:73, 126:129) , ]
```

A tibble: 13 × 7

	`PK` <int>	Incremento <dbl>	Mix_P <dbl>	CO_2 <dbl>	Corrup <dbl>	Empresa <chr>	W <chr>
1	1	5.1	3.5	1.4	0.1	Conservadora	C
2	2	4.9	3.0	1.4	0.1	Conservadora	C
3	3	4.7	3.2	1.3	0.1	Conservadora	C
4	4	4.6	3.1	1.5	0.1	Conservadora	C
5	5	5.0	3.6	1.4	0.1	Conservadora	C
6	70	5.6	2.5	3.9	4.8	Trader	T
7	71	5.9	3.2	4.8	10.1	Trader	T
8	72	6.1	2.8	4.0	6.0	Trader	T
9	73	6.3	2.5	4.9	9.0	Trader	T
10	126	7.2	3.2	6.0	10.1	Soudoyer	S
11	127	6.2	2.8	4.8	11.6	Soudoyer	S
12	128	6.1	3.0	4.9	10.8	Soudoyer	S
13	129	6.4	2.8	5.6	15.8	Soudoyer	S

Población Total

Este es el resultado de las multiples oportunidades (o muestras) en las que se ha desarrollado el juego de simulación:

```
> summary(BSC)
```

PK"	Incremento	Mix_P	CO_2
Min. : 1.00	Min. :4.300	Min. :2.000	Min. :1.000
1st Qu.: 38.25	1st Qu.:5.100	1st Qu.:2.800	1st Qu.:1.600
Median : 75.50	Median :5.800	Median :3.000	Median :4.350
Mean : 75.50	Mean :5.843	Mean :3.057	Mean :3.758
3rd Qu.:112.75	3rd Qu.:6.400	3rd Qu.:3.300	3rd Qu.:5.100
Max. :150.00	Max. :7.900	Max. :4.400	Max. :6.900

Corrup	Empresa	W
Min. : 0.000	Length:150	Length:150
1st Qu.: 0.300	Class :character	Class :character
Median : 6.200	Mode :character	Mode :character
Mean : 6.868		
3rd Qu.:11.200		
Max. :20.600		

Resultados de la simulación en ISEL

Table 1: Resultados

Equipos	\$ Final	Transacciones	Tomate	Sandía	Pera	Nuez	Manzana	Zapallo
1	0	5		5	4	2	2	8
2	218,2	3		3	4		3	
3	0		4	4		4	1	6
4	2224	3		4	3		8	
5	3912	3			8		8	8
S	0	9	8	9	5	8		4
R	0	0						

Resultado de la Muestra

Table 2: Comportamiento Muestra

Comportamieto	
Promedio	ISEL
Economía	Abogado
Nro. Condición Matriz	
0,01	
Casos	
151	
Observaciones	
Dificultad para iniciar	
Poco acuerdo en precio	
No alcanza el Equilibrio	

Indicadores de Desempeño

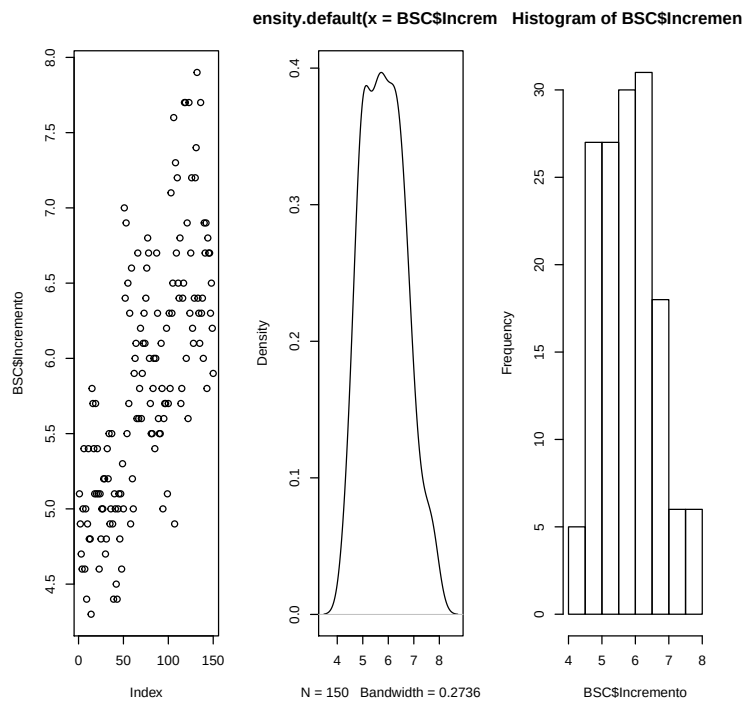
Table 3: Indicadores Índice de Desempeño

Transacciones / Equipo	
Promedio	
93	3,7
Incremento Capital	
Promedio	
5,8	0,23
Mix Producto	
Promedio	
3	5
CO2	
Promedio	
3,75	6,9
Índice Corrupción	
6,68	0,4

Análisis de Histograma - Incremento

Veremos como se comportan las muestras (Incremento) utilizando el histograma.
I=10 implica duplica el capital inicial

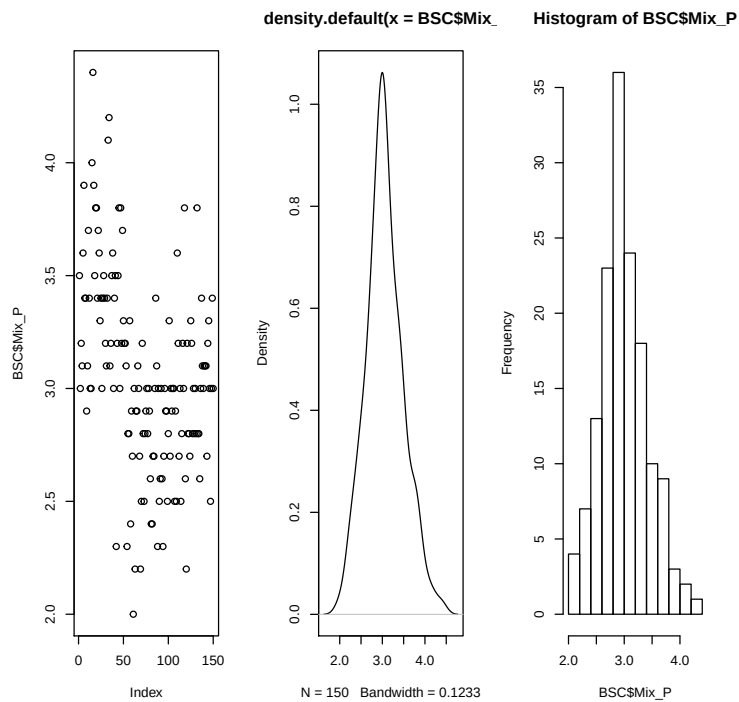
```
> #par(mfrow=c(2,1))
> par(mfrow=c(1,3))
> plot(BSC$Incremento)
> plot(density(BSC$Incremento))
> hist(BSC$Incremento)
>
```



Análisis de Histograma Mix Dinero Producto

Veremos como se comportan las muestras (Incremento) utilizando el histograma.
I=10 implica duplica el capital inicial

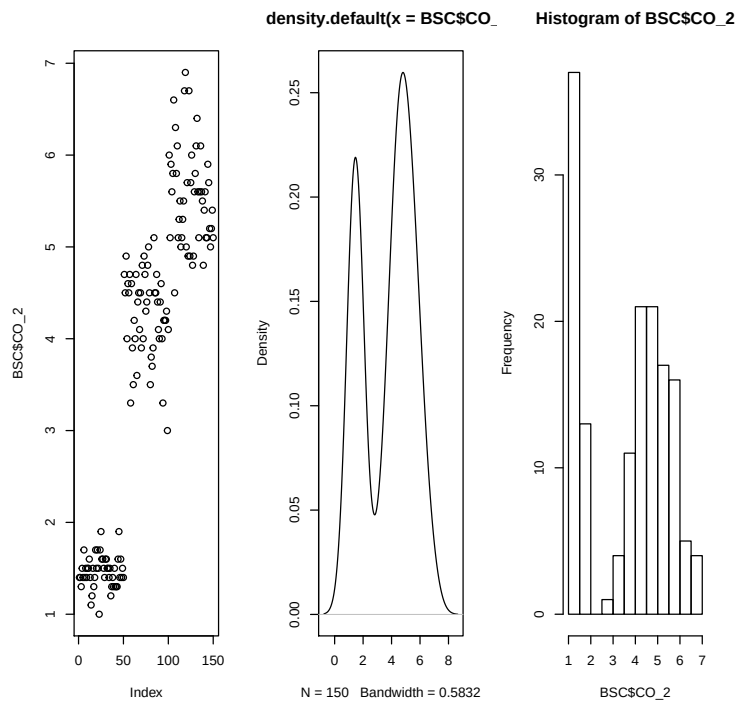
```
> par(mfrow=c(1,3))
> plot(BSC$Mix_P)
> plot(density(BSC$Mix_P))
> hist(BSC$Mix_P)
>
>
```



Análisis de Histograma CO2

Veremos como se comportan las muestras (Incremento) utilizando el histograma.
 $I=10$ implica duplica el capital inicial

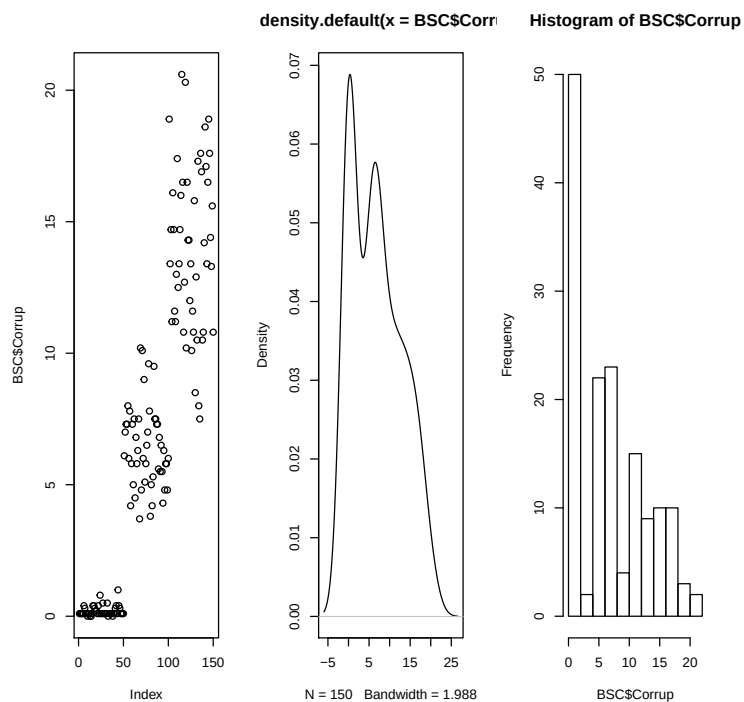
```
> par(mfrow=c(1,3))
> plot(BSC$CO_2)
> plot(density(BSC$CO_2))
> hist(BSC$CO_2)
>
>
```



Análisis de Histograma Corrupcion

Veremos como se comportan las muestras (Incremento) utilizando el histograma.
 $I=10$ implica duplica el capital inicial

```
> par(mfrow=c(1,3))
> plot(BSC$Corrup)
> plot(density(BSC$Corrup))
> hist(BSC$Corrup)
>
>
```

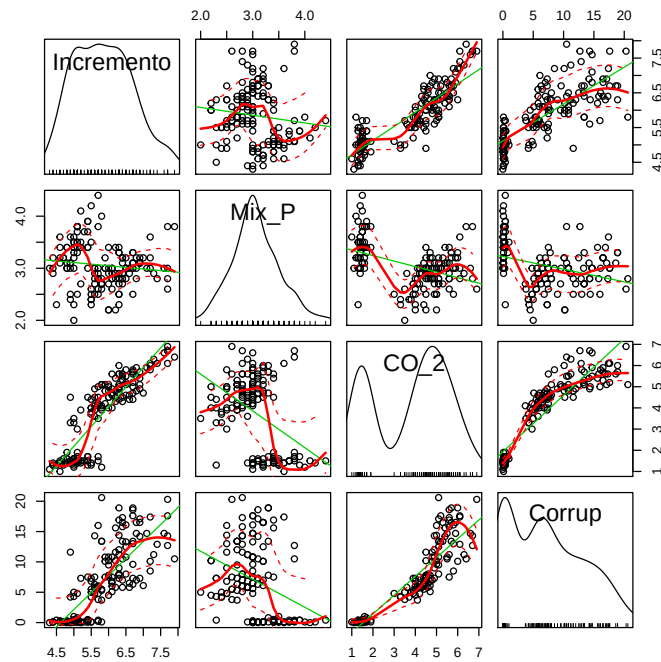


Análisis Multivariado

```
> library(car)
> BSC_Rawdata <- BSC[,c(2,3,4,5)]
> BSC_Rawdata

# A tibble: 150 × 4
  Incremento Mix_P CO_2 Corrup
    <dbl> <dbl> <dbl> <dbl>
1      5.1  3.5  1.4  0.1
2      4.9  3.0  1.4  0.1
3      4.7  3.2  1.3  0.1
4      4.6  3.1  1.5  0.1
5      5.0  3.6  1.4  0.1
6      5.4  3.9  1.7  0.4
7      4.6  3.4  1.4  0.3
8      5.0  3.4  1.5  0.1
9      4.4  2.9  1.4  0.1
10     4.9  3.1  1.5  0.0
# ... with 140 more rows

> par(mfrow=c(1,1))
> scatterplotMatrix(BSC_Rawdata)
```



1.2 Análisis de Componentes Principales

Crearemos un objeto nuevo que se llamará PC1 (por Principal Component 1) y la instrucción con el que crearemos la matriz de correlaciones es princomp.

```
> PC1 <- princomp(BSC_Rawdata)
> PC1
```

Call:

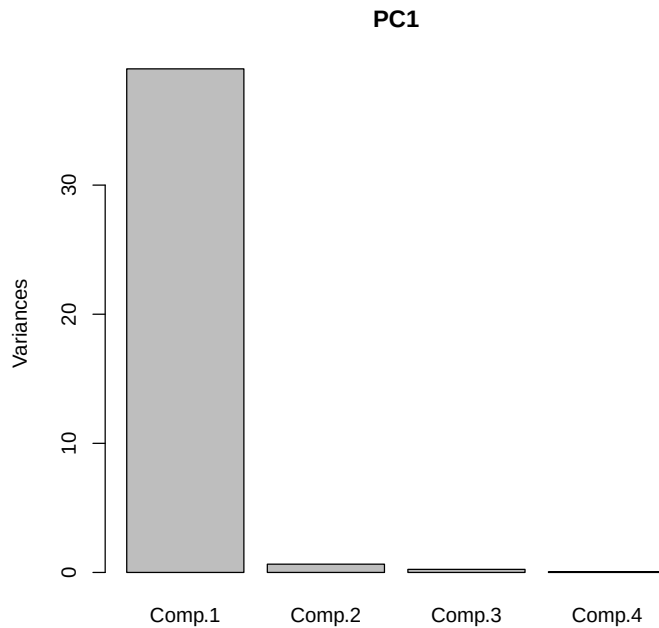
```
princomp(x = BSC_Rawdata)
```

Standard deviations:

Comp.1	Comp.2	Comp.3	Comp.4
6.2453806	0.8006568	0.4873362	0.2255064

4 variables and 150 observations.

```
> plot(PC1)
```



En el ploteo podemos ver que uno de los componentes principales aporta casi el 4 veces más de la información referida al comportamiento de la varianza de todos los casos. Este componente es el que más incluye en la clasificación o posible identificación del comportamiento de cada individuo de la muestra.

```
> summary(PC1)
```

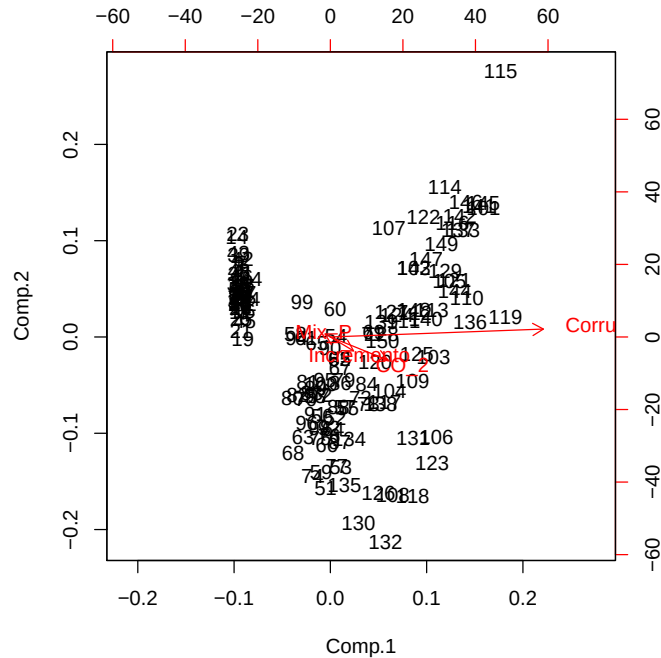
Importance of components:

	Comp.1	Comp.2	Comp.3	Comp.4
Standard deviation	6.2453806	0.8006568	0.487336232	0.225506374
Proportion of Variance	0.9767267	0.0160527	0.005947201	0.001273424
Cumulative Proportion	0.9767267	0.9927794	0.998726576	1.000000000

Si observamos bien el reporte que nos entrega el comando summary nos podemos dar cuenta que con los dos primeros componentes podríamos explicar 97.768521

¿Qué pasaría si representamos a las empresas en un gráfico en el que las variables de los ejes sean los dos componentes principales? , pues tendríamos un primer indicio de la bondad de las dimensiones o variables para agrupar a las muestras Esto lo podemos realizar con el comando biplot

```
> biplot(PC1)
```



Los números que

aparecen en el diagrama son el caso de estudio (renglón en que se encuentra la empresa contratista). A simple vista observamos que hay como tres tipos distintos de empresas (tres nubes claramente diferenciadas). Aquí nos queda claro que el principal componente que ordena o divide a estas colonias es indistintamente el CAPITAL o el EQUIPAMIENTO con que cuentan.

También podemos ver que hay empresas como la 15, 16, 132, 118, 61, 107 sobre las que el gráfico nos recomienda estudiarlas más pues no es capaz de clasificarlas bien (son casos extremos o anómalos). Tal vez con poco capital o sin equipo pueden llegar a ser competitivas o interesantes para las grandes constructoras.

Por último la dimensión referida a la certificación de NORMAS es la dimensión que menos valor aporta. Esto no implica que no certificar sea poco importante, sino que probablemente sea una pregunta irrelevante si todos contestaron que SI certificaron ISO 9000.

1.3 Scores

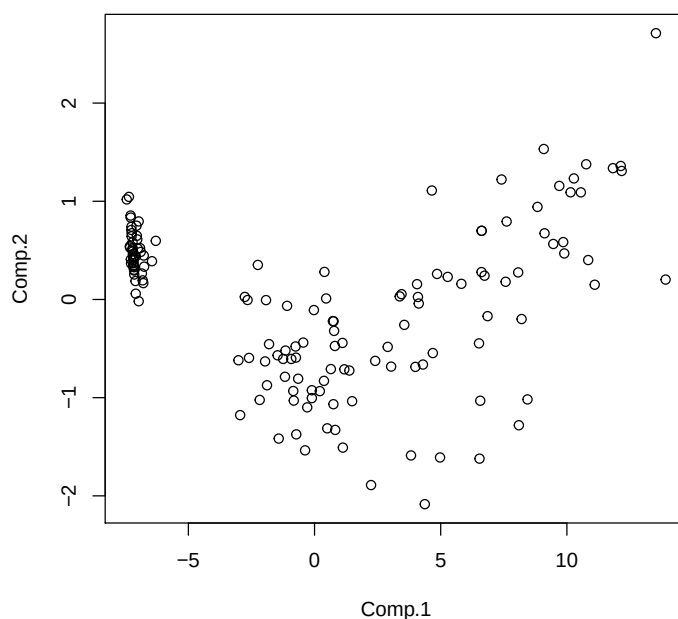
Si el comportamiento del componente va hacia el lado positivo, se debe interpretar como que a mayor desempeño mejor resultado o calificación. Si algún componente apunta para el lado negativo tendremos que pensar que a mayor calificación en esa dimensión pero sería el desempeño. La variable PC1 que usamos tiene mucha información valiosa. Revise todo el contenido, voy a mostrar una dimensión que es el score que indica como se comportarían todos los individuos si sólo los analizásemos con los componentes 1 y 2.

```
> acp1 <- PC1$scores
> acp1 [1:10 , ]
```

	Comp.1	Comp.2	Comp.3	Comp.4
[1,]	-7.196222	0.4619536	0.33559194	0.053357162
[2,]	-7.205173	0.5073358	-0.16573596	0.244487143
[3,]	-7.256808	0.7080014	-0.10653993	0.035769488
[4,]	-7.212830	0.5823004	-0.29208587	-0.052762564
[5,]	-7.209033	0.5212648	0.35277190	-0.073224500
[6,]	-6.808533	0.1945546	0.75177663	-0.133686736
[7,]	-7.054061	0.7513482	-0.03330587	-0.191500696
[8,]	-7.178257	0.4182958	0.17488212	0.009270777
[9,]	-7.254938	0.7411144	-0.53899717	-0.007598331
[10,]	-7.277505	0.4079475	-0.11622695	0.132957196

Voy a realizar el mismo score pero ahora solo con los componentes 1 y 2

```
> acp2 <-PC1$scores[ ,1:2]
> plot(acp2)
```



Aquí ya podemos ver más claramente la división que se produce entre distintos clusters. Para poder diferenciarlos aún más recurriremos a un nuevo tipo de análisis diferenciado que se llama análisis de clusters

2 Agrupamiento Supervisado y No Supervisado

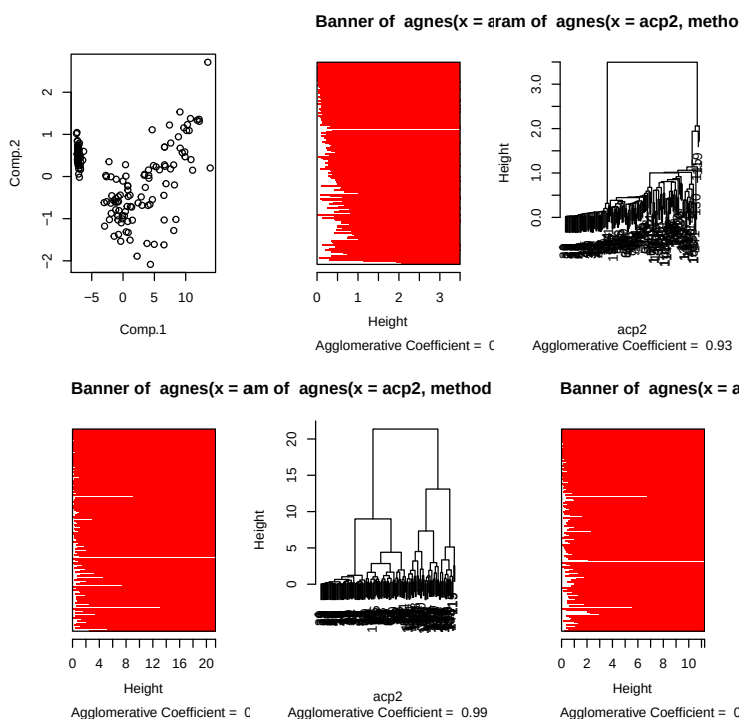
2.1 Análisis de Clusters o Conglomerados

Para realizar este análisis recurriremos a cargar la biblioteca clusters

En el análisis de conglomerados existen dos formas clásicas de estudio. Ambas recorren a las distancias euclídeas entre las muestras. Tenemos aproximaciones Jerárquicas y No Jerárquicas AGNES, CLARA, DIANA, MORA, PAM son nombres de las técnicas que la biblioteca Clusters usa. Todas las técnicas se caracterizan por ser un acrónimo de la combinación de aproximaciones que usan (Single Linkage, Complete Linkage, Average Linkage) .

Todas tienen nombre de mujer, pero esto no quiere necesariamente decir que se trate de una técnica con complicaciones inesperadas, sino más bien que si quieres lo mejor de una de ellas es mejor que la entiendas e indagues en la página del manual.

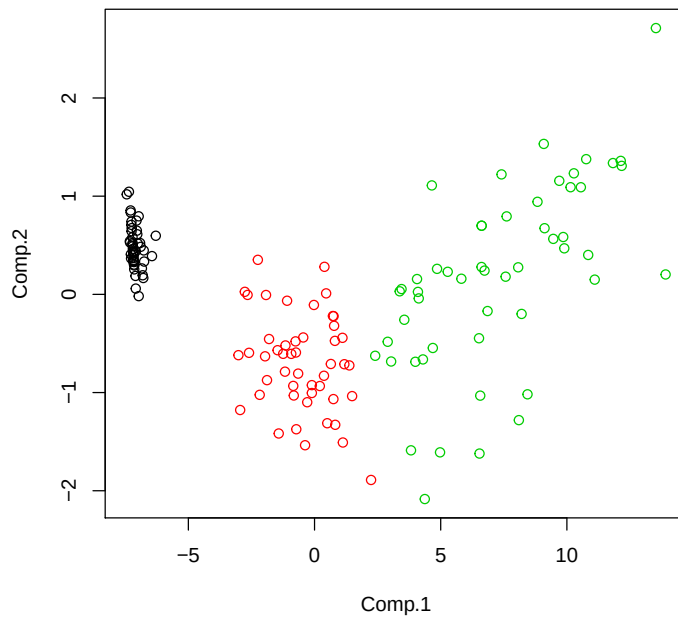
```
> library(cluster)
> agp1 = agnes(acp2,method="single")
> agp2 = agnes(acp2,method="complete")
> agp3 = agnes(acp2,method="average")
> par(mfrow=c(2,3))
> plot(acp2)
> plot(agp1)
> plot(agp2)
> plot(agp3)
```



Pasa asignar las muestras a grupos usará el comando cuttree que me permite valarme de las franjas blancas de corte del los gráficos para armar los clusters

```
> agpcut <- cutree(agp3,3)
> par(mfrow=c(1,1))
> plot(acp2,col=agpcut)
```

```
>
>
```



3 Inteligencia de Negocios

3.1 Indentificación de patrones y reglas

```
> library(car)
> partners <- read.table("~/R-Cran/Juego_Tomate.csv",header=TRUE,sep=",")
> #partners <- read.table("http://ceal.fing.uncu.edu.ar/r-cran/BSC_proveedores.csv",header=
> summary(partners)
```

```
PK.Incremento.Mix_P.CO_2.Corrup.Empresa.W.1.5.1.3.5.1.4.0.1.Conservadora.C.2.4.9.3.1.4.0.
```

```
Min.    : 6
1st Qu.: 42
Median  : 78
Mean    : 78
3rd Qu.:114
Max.    :150
```

Incremento	Mix_P	CO_2	Corrup
Min. :4.300	Min. :2.00	Min. :1.000	Min. : 0.000
1st Qu.:5.200	1st Qu.:2.80	1st Qu.:1.600	1st Qu.: 0.400
Median :5.800	Median :3.00	Median :4.400	Median : 6.500
Mean :5.877	Mean :3.05	Mean :3.839	Mean : 7.101
3rd Qu.:6.400	3rd Qu.:3.30	3rd Qu.:5.100	3rd Qu.:11.600
Max. :7.900	Max. :4.40	Max. :6.900	Max. :20.600

```

      Empresa  W
Conservadora:45 C:45
Soudoyer      :50 S:50
Trader        :50 T:50

```

```
> scatterplotMatrix(partners)
```

4 Aprendizaje Supervisado

4.1 Entrenamiento de árboles de decisión.

Esta técnica utiliza un set de datos representativos de una situación y utilizando recursivamente el teorema de Bayes puede armar un pronosticador o clasificador de datos. Es una técnica parecida a la de clustering, pero más refinada, pues no se basa en reglas sino en aprendizaje del set de datos usado como entrenamiento. En el paquete party existen dos funciones ctree que se utiliza para entrenar y predict que se usa para pronosticar o generar la regla de decisión que debemos usar.

```

> library(party)
> attach(partners)
> str(partners)

```

```

'data.frame':      145 obs. of  7 variables:
 $ PK.Incremento.Mix_P.CO_2.Corrup.Empresa.W.1.5.1.3.5.1.4.0.1.Conservadora.C.2.4.9.3.1.4.
 $ Incremento
 $ Mix_P
 $ CO_2
 $ Corrup
 $ Empresa
 $ W

```

```

> # describe al objeto transit
> ind <- sample(2, nrow(partners), replace=TRUE, prob=c(0.7, 0.3))
> # toma una muestra
> ind

```

```

[1] 2 2 1 1 1 2 1 2 2 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 2
[38] 1 2 1 1 2 1 1 1 2 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 2 2 2 1 1
[75] 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 2 1 1 2
[112] 1 1 1 1 2 2 1 2 1 1 1 1 1 1 1 1 2 1 1 2 1 2 1 1 1 1 1 1 1 1 2 1 2 1 1 1

```

```

> # nos imprime la muestra tomada.

> trainData <- partners [ind==1,]
> # genero un set de entrenamiento
> testData <- partners [ind==2,]
> # genero un set de datos de prueba
> myFormula <- W ~ Incremento + Mix_P + CO_2 + Corrup
> transit_ctree <- ctree(myFormula, data=trainData)
> # creo el motor de entrenamiento
> # Verificar las predicciones
> table(predict(transit_ctree), trainData$W)

```

	C	S	T
C	34	0	0
S	0	39	4
T	0	0	35

```

> print(transit_ctree)

Conditional inference tree with 4 terminal nodes

Response: W
Inputs: Incremento, Mix_P, CO_2, Corrup
Number of observations: 112

1) CO_2 <= 1.9; criterion = 1, statistic = 103.933
  2)* weights = 34
1) CO_2 > 1.9
  3) Corrup <= 10.2; criterion = 1, statistic = 52.719
    4) CO_2 <= 4.8; criterion = 1, statistic = 15.877
      5)* weights = 35
    4) CO_2 > 4.8
      6)* weights = 8
  3) Corrup > 10.2
    7)* weights = 35

```

Regla de clasificación descubierta

```
> plot(transit_ctree)
```

