# Relacion 2

# Curso 2024-2025

# ${\bf \acute{I}ndice}$

1	EJE	ERCICIO 1	2
	1.1	Apartado a)	2
	1.2	Apartado b)	3
	1.3	Apartado c)	3
	1.4	Apartado d)	4
	1.5	Apartado e)	5
2	EJE	ERCICIO 2	6
	2.1	Apartado a)	6
	2.2	Apartado b)	7
3	EJE	ERCICIO 3	8
	3.1	Apartado a)	8
	3.2	Apartado b)	9
4	EJE	ERCICIO 4	9
	4.1	Funciones de clase	9
	4.2	Diagrama Jerarquias	10
	4.3	Método ahp	11
5	EJE	ERCICIO 5	13
	5.1	Funciones de clase	13
	5.2	Diagrama Jerarquias	14
	5.3	Método ahp	15
##	Load	ding required package: shape	

# 1 EJERCICIO 1

### 1.1 Apartado a)

	a1	a2	a3
a1	1	0	1
a2	1	1	1
a3	0	0	1

• Método de construcción de la función de utilidad "Maximal"

```
(sol01a <- multicriterio.constfuncutilidad.maximales(tab01a)) # suma por filas
```

```
## a1 a2 a3
## 2 3 1
```

Cuánto más alto mejor, por lo tanto nos está diciendo que la mejor alternativa sería a2. La siguiente es la a1 y la peor es a 3. Ahora ordenamos de mejor a peor:

```
sort(sol01a, decreasing = T)
## a2 a1 a3
## 3 2 1
```

• Método de construcción de la función de utilidad "Borroso"

```
(sol01aBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab01a)) #calculo del flujo neto
```

```
## a1 a2 a3
## 0 2 -2
```

```
sort(sol01aBor, decreasing = T)
```

```
## a2 a1 a3
## 2 0 -2
```

La mejor es la alternativa 2

# 1.2 Apartado b)

	a1	a2	a3
a1	1	0	1
a2	1	1	1
a3	1	0	1

• Método de construcción de la función de utilidad "Maximal".

```
(sol01b <- multicriterio.constfuncutilidad.maximales(tab01b)) # suma por filas

## a1 a2 a3
## 2 3 2

sort(sol01b, decreasing = T)

## a2 a1 a3
## 3 2 2</pre>
```

La mejor es la alternativa 2

• Método de construcción de la función de utilidad "Borroso"

```
(sol01bBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab01b)) #calculo del flujo neto
## a1 a2 a3
## -1 2 -1
sort(sol01bBor, decreasing = T)
## a2 a1 a3</pre>
```

La mejor es la alternativa 2

# 1.3 Apartado c)

## 2 -1 -1

```
tab01c <- multicriterio.crea.matrizvaloraciones(rep(1,9), numalternativas = 3)</pre>
```

	a1	a2	a3
a1	1	1	1
a2	1	1	1
a3	1	1	1

• Método de construcción de la función de utilidad "Maximal".

```
(sol01c <- multicriterio.constfuncutilidad.maximales(tab01c)) # suma por filas

## a1 a2 a3
## 3 3 3

sort(sol01c, decreasing = T)

## a1 a2 a3
## 3 3 3</pre>
```

Cualquiera de los 3 es la mejor alternativa

• Método de construcción de la función de utilidad "Borroso"

```
(sol01cBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab01c)) #calculo del flujo neto
## a1 a2 a3
## 0 0 0
sort(sol01cBor, decreasing = T)</pre>
```

## a1 a2 a3 ## 0 0 0

Cualquiera de los 3 es la mejor alternativa

# 1.4 Apartado d)

	a1	a2	a3	a4	a5
a1	1	1	1	0	1
a2	0	1	0	0	1
a3	1	0	1	0	1
a4	1	1	1	1	0
a5	0	0	0	0	1

• Método de construcción de la función de utilidad "Maximal".

```
(sol01d <- multicriterio.constfuncutilidad.maximales(tab01d)) # suma por filas

## a1 a2 a3 a4 a5
## 4 2 4 5 1

sort(sol01d, decreasing = T)

## a4 a1 a3 a2 a5
## 5 4 4 2 1</pre>
La mejor es la alternativa 4
```

• Método de construcción de la función de utilidad "Borroso"

```
(sol01dBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab01d)) #calculo del flujo neto
```

```
## a1 a2 a3 a4 a5
## 1 -1 0 3 -3
```

```
sort(sol01dBor, decreasing = T)
```

```
## a4 a1 a3 a2 a5
## 3 1 0 -1 -3
```

La mejor es la alternativa 4

# 1.5 Apartado e)

	a1	a2	a3	a4	a5
a1	1	1	1	0	1
a2	0	1	0	0	1
a3	1	1	1	0	1
a4	1	1	1	1	0
$a_5$	0	0	0	0	1

• Método de construcción de la función de utilidad "Maximal".

```
(sol01e <- multicriterio.constfuncutilidad.maximales(tab01e)) # suma por filas

## a1 a2 a3 a4 a5
## 4 2 4 5 1

sort(sol01e, decreasing = T)

## a4 a1 a3 a2 a5
## 5 4 4 2 1</pre>
```

• Método de construcción de la función de utilidad "Borroso"

```
(sol01eBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab01e)) #calculo del flujo neto
## a1 a2 a3 a4 a5
## 1 -2 1 3 -3</pre>
```

```
sort(sol01eBor, decreasing = T)
```

```
## a4 a1 a3 a2 a5
## 3 1 1 -2 -3
```

La mejor es la alternativa 4

La mejor es la alternativa 4

# 2 EJERCICIO 2

# 2.1 Apartado a)

	a1	a2	a3
a1	0.0	0.4	0.7
a2	0.2	0.0	0.5
a3	0.3	0.6	0.0

• Método de construcción de la función de utilidad "Maximal".

```
(sol02a <- multicriterio.constfuncutilidad.maximales(tab02a)) # suma por filas

## a1 a2 a3
## 3 1 2

sort(sol02a, decreasing = T)

## a1 a3 a2
## 3 2 1</pre>
```

La mejor es la alternativa 1

• Método de construcción de la función de utilidad "Borroso"

```
(sol02aBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab02a)) #calculo del flujo neto
## a1 a2 a3
## 0.6 -0.3 -0.3
sort(sol02aBor, decreasing = T)</pre>
```

```
## a1 a2 a3
## 0.6 -0.3 -0.3
```

La mejor es la alternativa 1

# 2.2 Apartado b)

	a1	a2	a3
a1	0.0	0.2	0.4
a2	0.9	0.0	0.8
a3	0.1	0.3	0.0

• Método de construcción de la función de utilidad "Maximal".

```
(sol02b <- multicriterio.constfuncutilidad.maximales(tab02b)) # suma por filas
## a1 a2 a3
## 2 3 1</pre>
```

```
sort(sol02b, decreasing = T)
## a2 a1 a3
## 3 2 1
```

La mejor es la alternativa 2

• Método de construcción de la función de utilidad "Borroso"

```
(sol02bBor <- multicriterio.constfuncutilidad.estructuraborrosa(tab02b)) #calculo del flujo neto

## a1 a2 a3
## -0.4 1.2 -0.8

sort(sol02bBor, decreasing = T)

## a2 a1 a3
## 1.2 -0.4 -0.8</pre>
```

La mejor es la alternativa 2

# 3 EJERCICIO 3

	C1	C2	С3	C4	C5
a1	100	15	7	40	50
a2	200	25	7	60	200
a3	100	20	4	25	25
a4	200	30	20	70	350
a5	250	25	25	100	500

# 3.1 Apartado a)

Vamos a homogeneizar las columnas de la tabla de decisión por el método Nadir

```
sol03a <- round(multicriterio.homogeneizacion.nadir(tab03), 4)</pre>
```

	C1	C2	С3	C4	C5
a1	0.0000	0.0000	0.1429	0.2000	0.0526
a2	0.6667	0.6667	0.1429	0.4667	0.3684
a3	0.0000	0.3333	0.0000	0.0000	0.0000
a4	0.6667	1.0000	0.7619	0.6000	0.6842
a5	1.0000	0.6667	1.0000	1.0000	1.0000

### 3.2 Apartado b)

Vamos a homogeneizar las columnas de la tabla de decisión por el método Ptomethee

	C1	C2	С3	C4	C5
a1	-2.5556	-2.7778	-2.0000	-1.250	-1.8333
a2	1.3333	0.7778	-2.0000	-0.125	-0.4167
a3	-2.5556	-1.0000	-2.0000	-2.000	-2.0000
a4	1.3333	2.2222	2.8333	0.625	1.4167
a5	2.4444	0.7778	3.1667	2.750	2.8333

# 4 EJERCICIO 4

#### 4.1 Funciones de clase

#### Introducción datos

#### Cálculo pesos locales

Método mayor autovalor

```
pes1 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab1)
pes2 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab2)
pes3 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab3)</pre>
```

#### Cálculo pesos globales

	Rendimiento	Riesgo	Ponderadores Globales
A	0.7500000	0.3333333	0.6111111
В	0.2500000	0.6666667	0.3888889
Ponder.Criterios	0.6666667	0.3333333	NA

La mejor decisión es la alternativa A (peso global del 61,11%)

```
which.max(tab04[,1])
```

```
## A
## 1
```

Para el rendimiento la mejor alternativa es la A

```
which.max(tab04[,2])
```

```
## B
## 2
```

Para el riesgo la mejor alternativa es la B

Además del método de mayor autovalor, tambien tenemos:

Método de media geométrica

```
pes1 <- multicriterio.metodoAHP.variante2.mediageometrica(tab1)
pes2 <- multicriterio.metodoAHP.variante2.mediageometrica(tab2)
pes3 <- multicriterio.metodoAHP.variante2.mediageometrica(tab3)</pre>
```

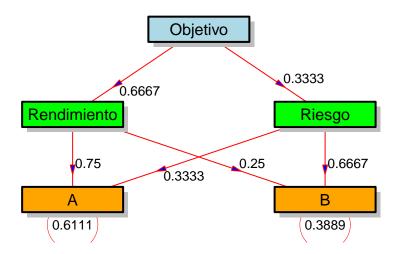
Método básico

```
pes1 <- multicriterio.metodoAHP.variante3.basico(tab1)
pes2 <- multicriterio.metodoAHP.variante3.basico(tab2)
pes3 <- multicriterio.metodoAHP.variante3.basico(tab3)</pre>
```

### 4.2 Diagrama Jerarquias

```
num.alt <- 2
num.crt <- 2
Xmatriznivel2_04 <- array(NA, dim = c(num.alt, num.alt, num.crt))
Xmatriznivel2_04[,,1] <- tab2
Xmatriznivel2_04[,,2] <- tab3
dimnames(Xmatriznivel2_04)[[1]] <- c("A","B")
multicriterio.metodoahp.diagrama(tab1, Xmatriznivel2_04)</pre>
```

# Estructura Jerárquica (AHP)



# 4.3 Método ahp

```
library(ahp)
datos04 = Load("problema4.ahp")
Calculate(datos04)
```

Visualize(datos04)

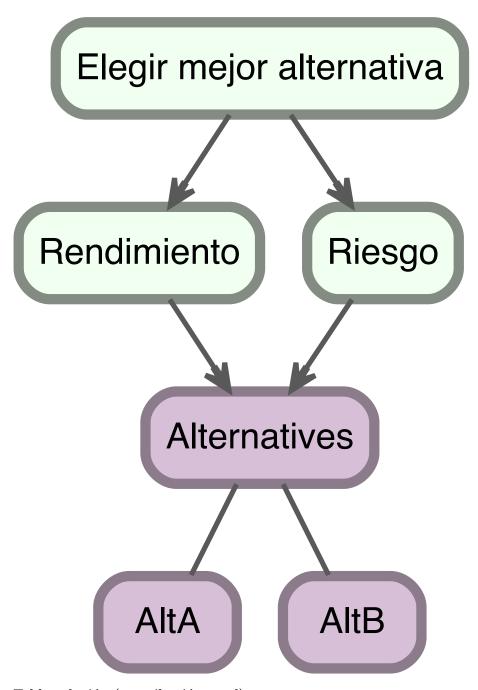


Tabla solución (contribución total)

export\_formattable(AnalyzeTable(datos04), file = "tablaahp104.png")

	Weight	AltA	AltB	Inconsistency
Elegir mejor alternativa	100.0%	61.1%	38.9%	0.0%
Rendimiento	66.7%	50.0%	16.7%	0.0%
Riesgo	33.3%	11.1%	22.2%	0.0%

La mejor decisión es la alternativa A (peso global del 61,11%)

#### Tabla solución (pesos locales)

```
t2 = AnalyzeTable(datos04, variable = "priority")
export_formattable(t2, file = "tablaahp204.png")
```

	Priority	AltA	AltB	Inconsistency
Elegir mejor alternativa	100.0%			0.0%
Rendimiento	66.7%	75.0%	25.0%	0.0%
Riesgo	33.3%	33.3%	66.7%	0.0%

### 5 EJERCICIO 5

#### 5.1 Funciones de clase

#### Introducción datos

#### Cálculo pesos locales

Método mayor autovalor

```
pes1 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab1)
pes2 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab2)
pes3 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab3)
pes4 <- multicriterio.metodoAHP.variante1.autovectormayorautovalor(tab4)</pre>
```

#### Cálculo pesos globales

	Liderazgo	Habilidad personal	Habilidad Gestión	Ponderadores Globales
A	0.800000	0.7500000	0.6666667	0.7263353
В	0.200000	0.2500000	0.3333333	0.2736647
Ponder.Criterios	0.124306	0.5171336	0.3585604	NA

La mejor decisión es la alternativa A (peso global del 72,63%)

```
which.max(tab05[,1])
```

## A

## 1

Para el liderazgo la mejor alternativa es la A

```
which.max(tab05[,2])
```

## A

## 1

Para la habilidad personal la mejor alternativa es la A

```
which.max(tab05[,3])
```

## A

## 1

Para la habilidad gestión la mejor alternativa es la A

Además del método de mayor autovalor, tambien tenemos:

Método de media geométrica

```
pes1 <- multicriterio.metodoAHP.variante2.mediageometrica(tab1)
pes2 <- multicriterio.metodoAHP.variante2.mediageometrica(tab2)
pes3 <- multicriterio.metodoAHP.variante2.mediageometrica(tab3)
pes4 <- multicriterio.metodoAHP.variante2.mediageometrica(tab4)</pre>
```

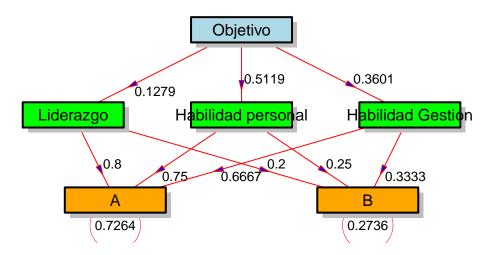
Método básico

```
pes1 <- multicriterio.metodoAHP.variante3.basico(tab1)
pes2 <- multicriterio.metodoAHP.variante3.basico(tab2)
pes3 <- multicriterio.metodoAHP.variante3.basico(tab3)
pes4 <- multicriterio.metodoAHP.variante3.basico(tab4)</pre>
```

# 5.2 Diagrama Jerarquias

```
num.alt <- 2
num.crt <- 3
Xmatriznivel2_05 <- array(NA, dim = c(num.alt, num.alt, num.crt))
Xmatriznivel2_05[,,1] <- tab2
Xmatriznivel2_05[,,2] <- tab3
Xmatriznivel2_05[,,3] <- tab4
dimnames(Xmatriznivel2_05)[[1]] <- c("A","B")
dimnames(Xmatriznivel2_05)[[2]] <- c("A","B")
dimnames(Xmatriznivel2_05)[[3]] <- c("Liderazgo","Habilidad personal", "Habilidad gestión")
multicriterio.metodoahp.diagrama(tab1, Xmatriznivel2_05)</pre>
```

# Estructura Jerárquica (AHP)



# 5.3 Método ahp

```
library(ahp)
datos05 = Load("problema5.ahp")
Calculate(datos05)
Visualize(datos05)
```

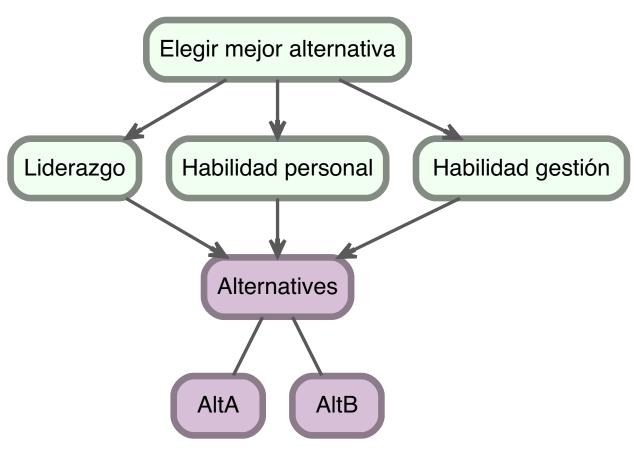


Tabla solución (contribución total)

export\_formattable(AnalyzeTable(datos05), file = "tablaahp105.png")

	Weight	AltA	AltB	Inconsistency
Elegir mejor alternativa	100.0%	72.6%	27.4%	<b>9</b> 10.3%
Habilidad personal	51.7%	38.8%	12.9%	0.0%
Habilidad gestión	35.9%	23.9%	12.0%	0.0%
Liderazgo	12.4%	9.9%	2.5%	0.0%

La mejor decisión es la alternativa A (peso global del  $72,\!63\%)$ 

Tabla solución (pesos locales)

```
t2 = AnalyzeTable(datos05, variable = "priority")
export_formattable(t2, file = "tablaahp205.png")
```

	Priority	AltA	AltB	Inconsistency
Elegir mejor alternativa	100.0%			<b>9</b> 10.3%
Habilidad personal	51.7%	75.0%	25.0%	0.0%
Habilidad gestión	35.9%	66.7%	33.3%	0.0%
Liderazgo	12.4%	80.0%	20.0%	0.0%