TRABAJO DE FIN DE GRADO: ANEXO

DISEÑO Y DESARROLLO DE ESTRATEGIAS DE INTEGRACIÓN MULTI-OMICA EN EL CONTEXTO DE LA EVALUACIÓN DE LA RESPUESTA INMUNE HUMORAL EN LEUCEMIA LINFÁTICA CRÓNICA Y SU ESTADIO PREVIO

GRADO EN ESTADÍSTICA

Facultad de Ciencias

Curso 2022/2023

Autora:

Laura Lezaun Pagola

Tutores:

José Manuel Sánchez Santos

Manuel Fuentes García

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1 GRÁFICOS Y TABLAS

Figura S1

Boxplots de los datos originales de Autoanticuerpos IgG

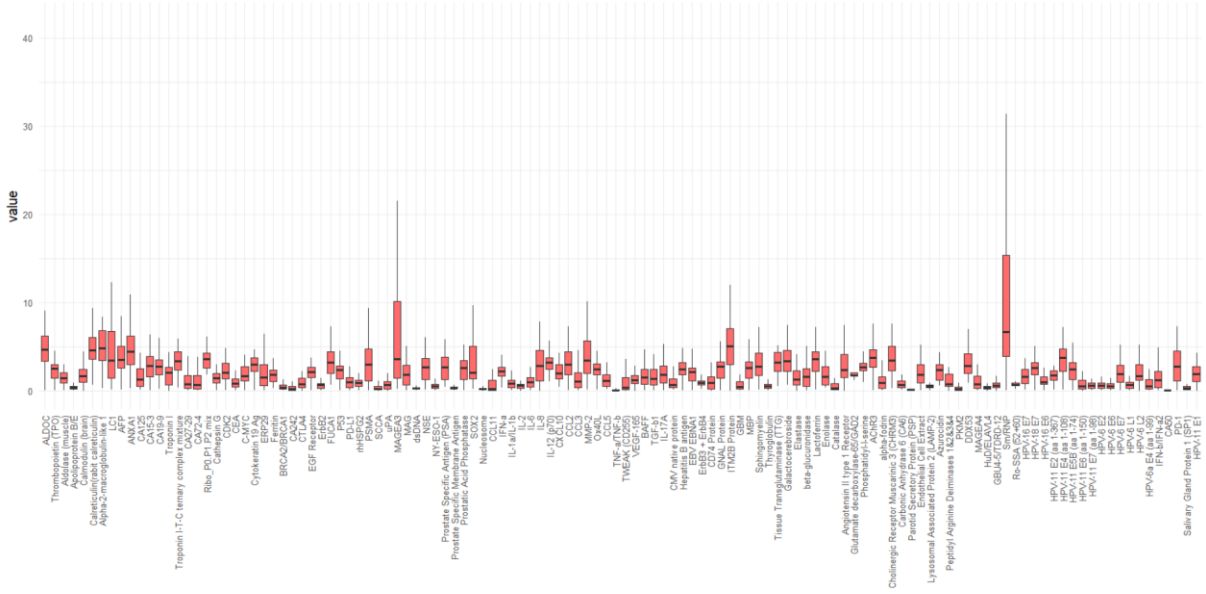

Figura S2

Boxplots de los datos estandarizados de Autoanticuerpos IgG

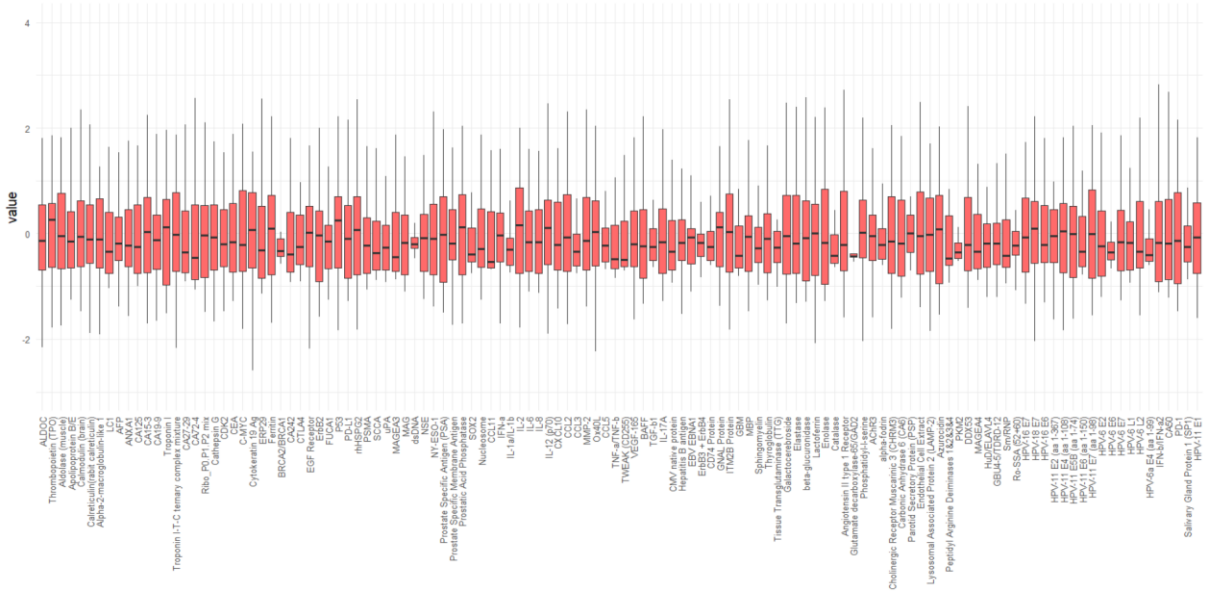

Figura S3

Boxplots de los datos originales de Autoanticuerpos IgM

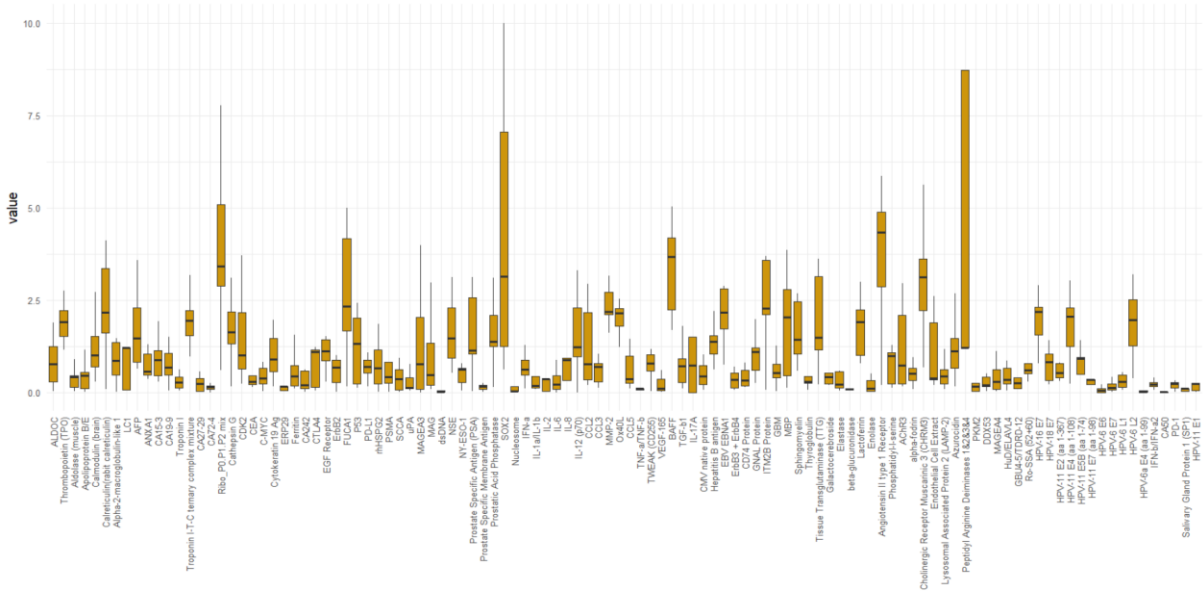

Figura S4

Boxplots de los datos estandarizados de Autoanticuerpos IgM

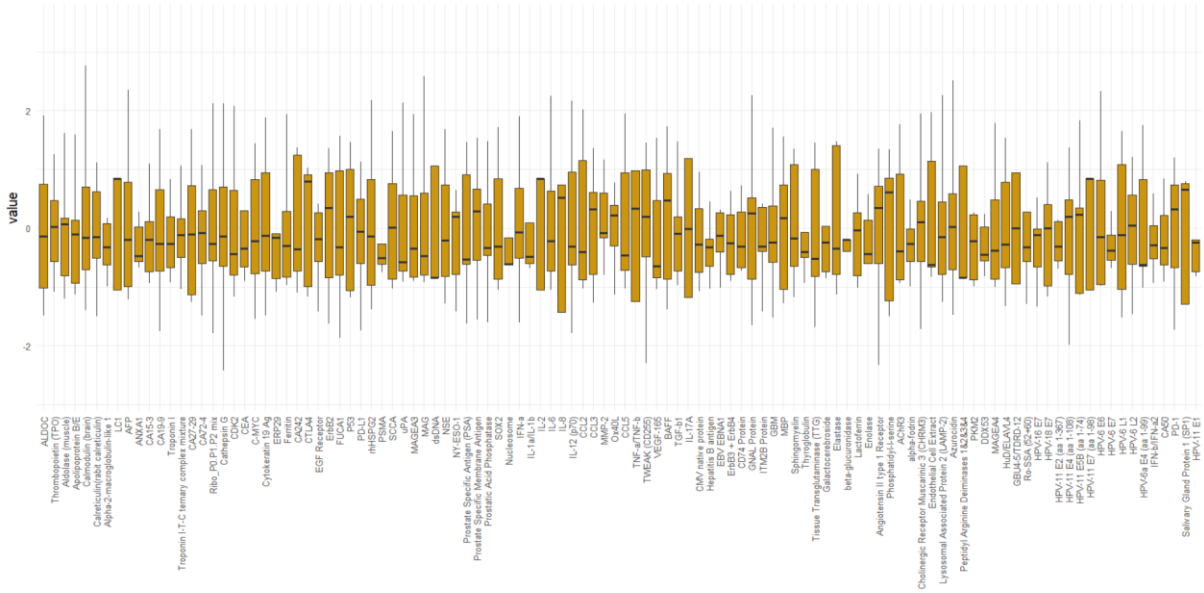

Figura S5

Boxplots de los datos originales de ESPA IgG

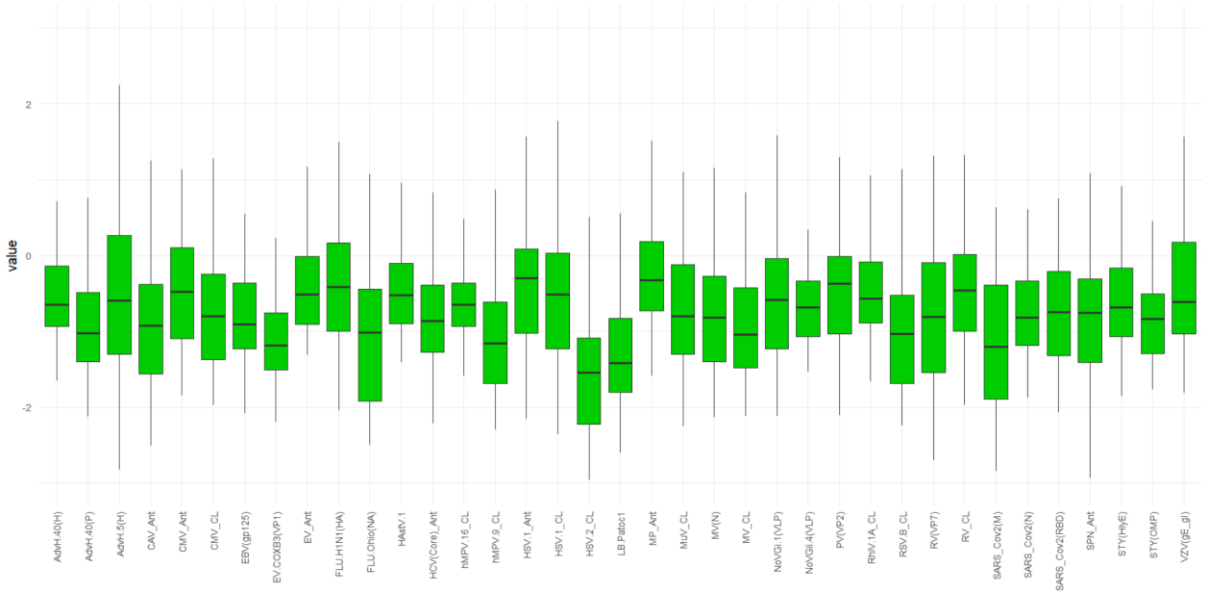

Figura S6

Boxplots de los datos estandarizados de ESPA IgG

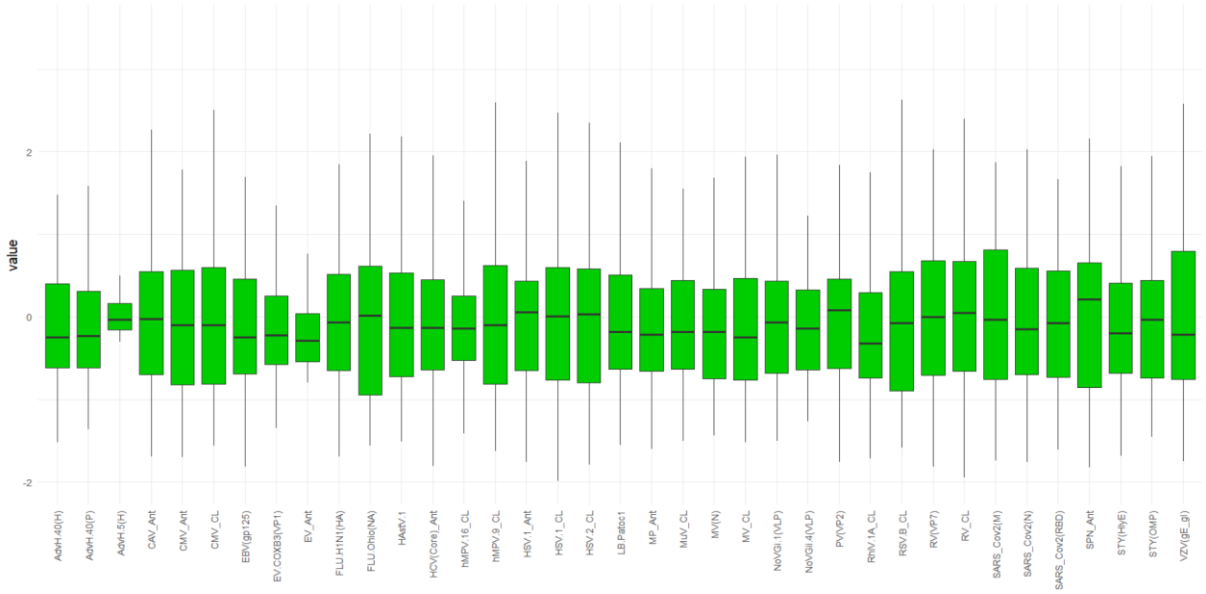

Figura S7

Boxplots de los datos originales de ESPA IgM

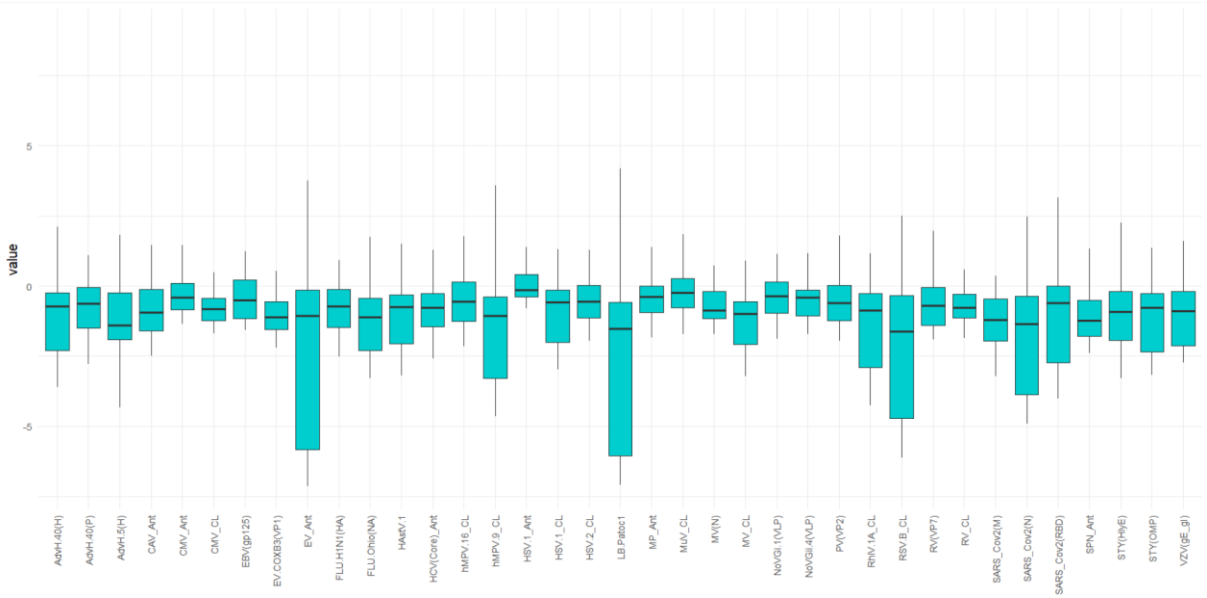

Figura S8

Boxplots de los datos estandarizados de ESPA IgM

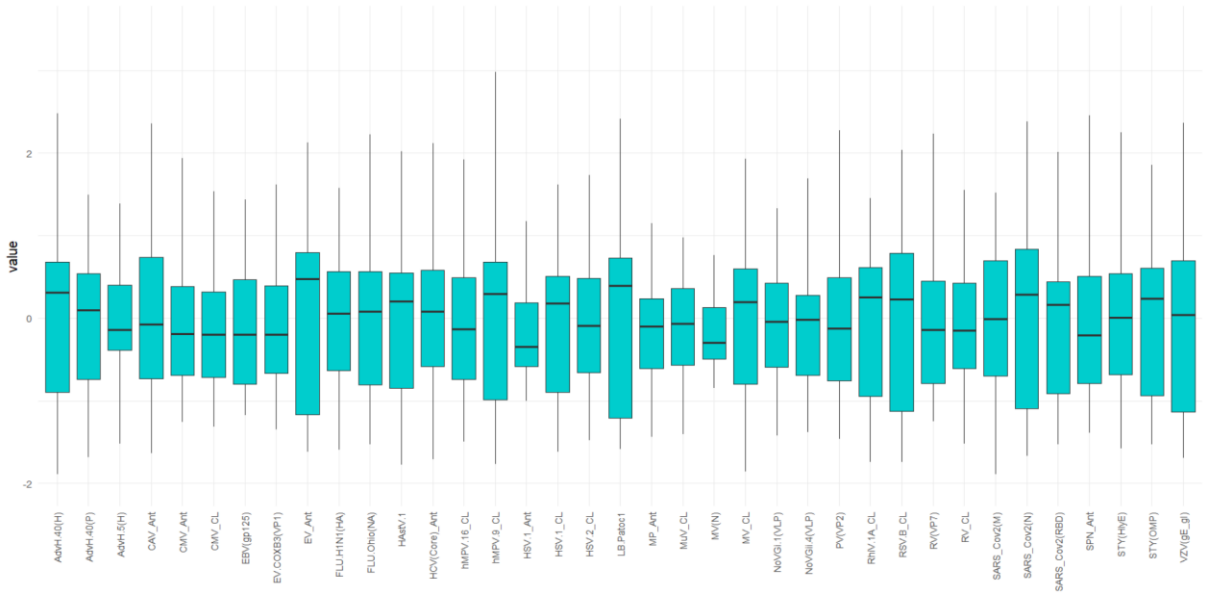

Figura S9

Boxplots de los datos originales de Immunocheckpoints

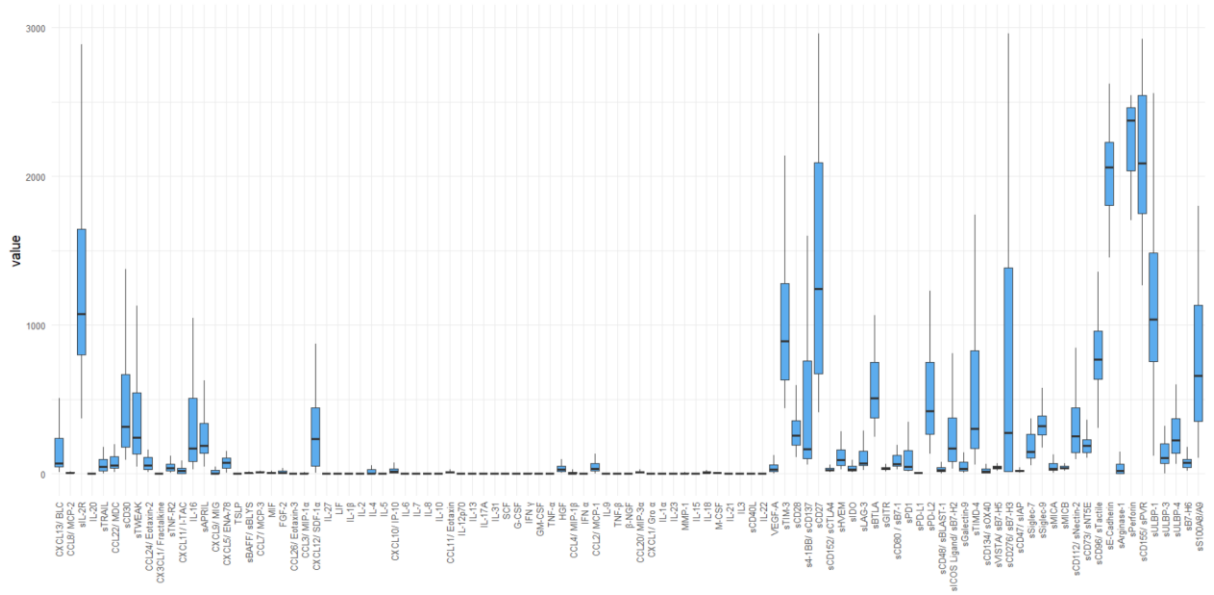

Figura S10

Boxplots de los datos estandarizados de Immunocheckpoints

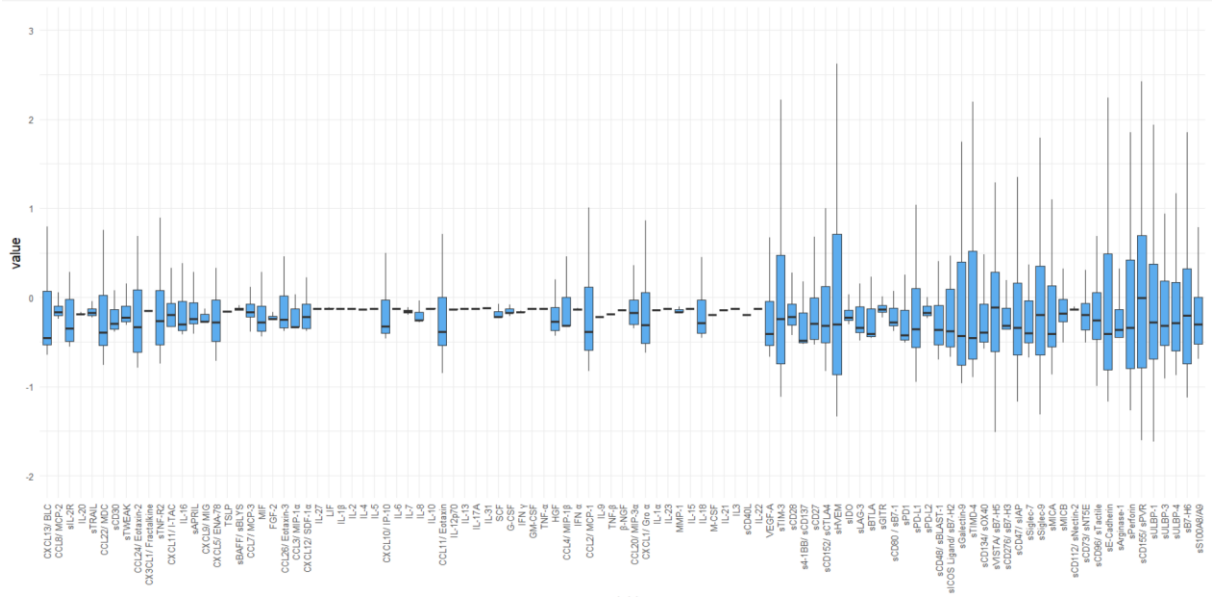

Figura S11

Boxplots de los datos originales de Proteoma

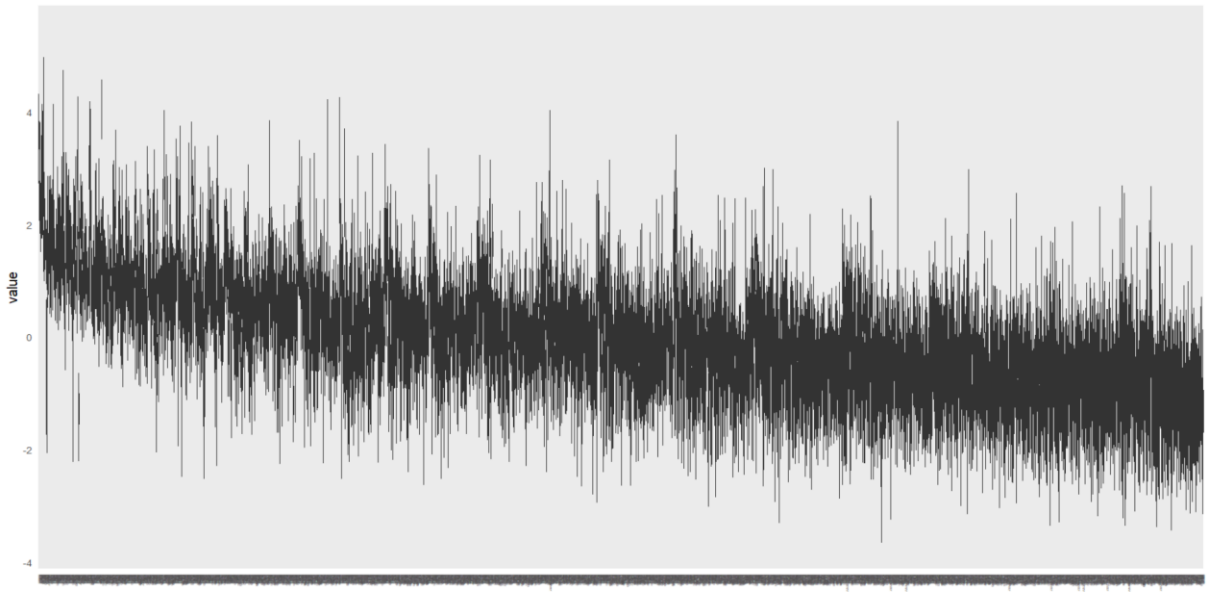

Figura S12

Boxplots de los datos estandarizados de la base de datos Proteoma

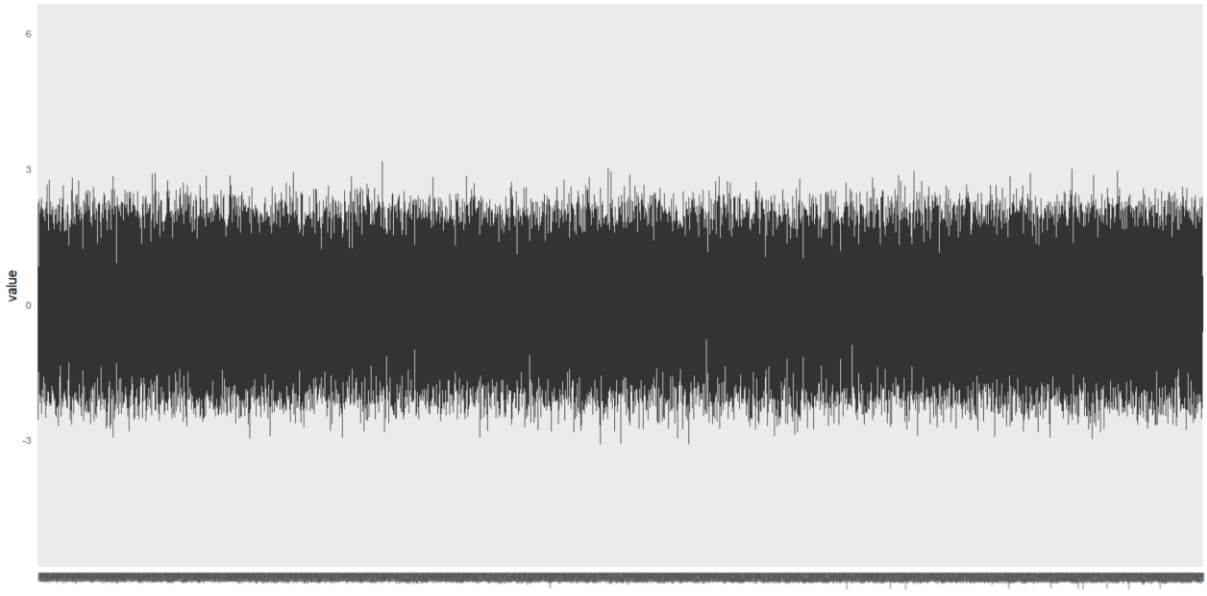

Figura S13

Factores de MOFA que explican al menos el 1.5% de la variabilidad de cada vista

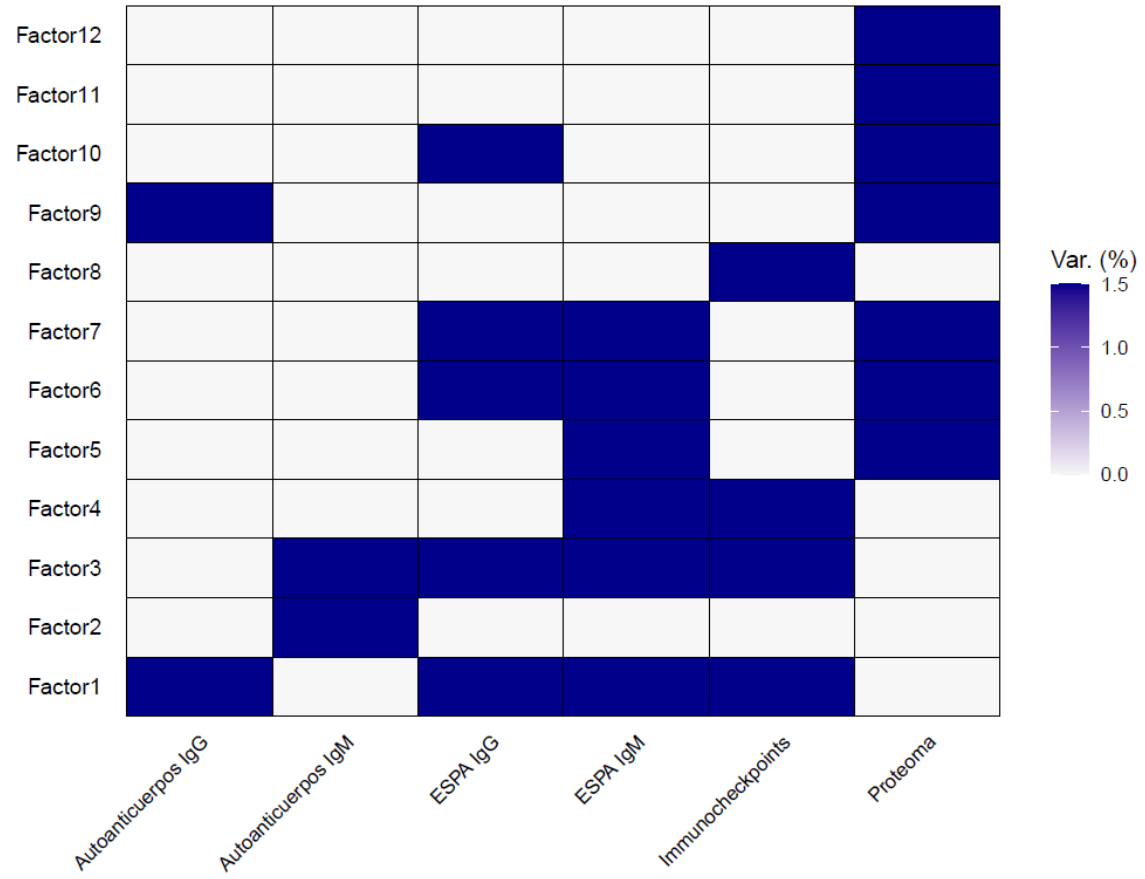

Figura S14

Función de Distribución Acumulativa de Consenso para Autoanticuerpos IgM

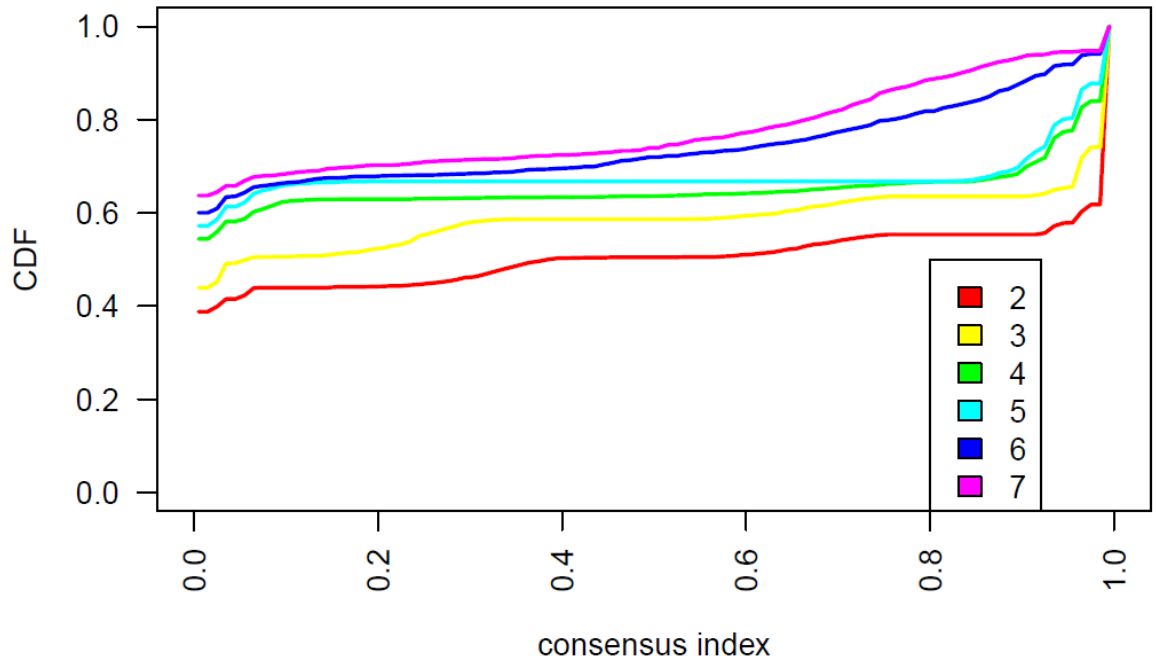

Figura S15

Heatmap de la matriz de consenso sobre Autoanticuerpos IgM para k = 6

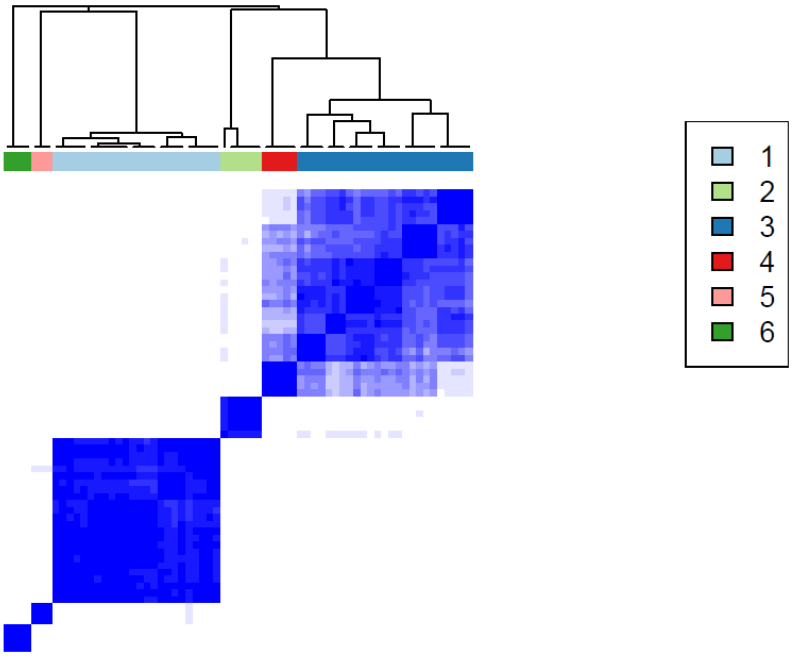

Tabla S1

Individuos y clúster asignado para Autoanticuerpos IgM

ID del paciente	Clúster	ID del paciente	Clúster	ID del paciente	Clúster
LLC-1	1	LLC-38	5	LLC-6	3
LLC-10	1	LLC-39	1	LLC-61	3
LLC-11	1	LLC-4	1	LLC-62	1
LLC-17	2	LLC-40	1	LLC-63	1
LLC-18	3	LLC-41	1	LLC-64	2
LLC-19	4	LLC-42	3	LLC-65	3
LLC-20	3	LLC-43	3	LLC-66	4
LLC-21	5	LLC-44	6	LLC-67	3
LLC-22	1	LLC-45	3	LLC-68	2
LLC-23	1	LLC-46	3	LLC-69	1
LLC-24	1	LLC-47-1	1	LLC-7	6
LLC-25	3	LLC-48	1	LLC-70	1
LLC-26	3	LLC-49	2	LLC-71	1
LLC-28	6	LLC-5	3	LLC-72	6
LLC-29	3	LLC-50	3	LLC-74	3
LLC-3	1	LLC-52	4	LLC-76	1
LLC-30	3	LLC-53	3	LLC-77	2
LLC-32	1	LLC-54	5	LLC-78	3
LLC-33	1	LLC-55	1	LLC-79	4
LLC-34	2	LLC-56	1	LLC-8	3
LLC-35	3	LLC-57	1	LLC-9	3
LLC-36	4	LLC-58	3		
LLC-37	3	LLC-59	3		

Figura S16

Función de Distribución Acumulativa de Consenso para ESPA IgG

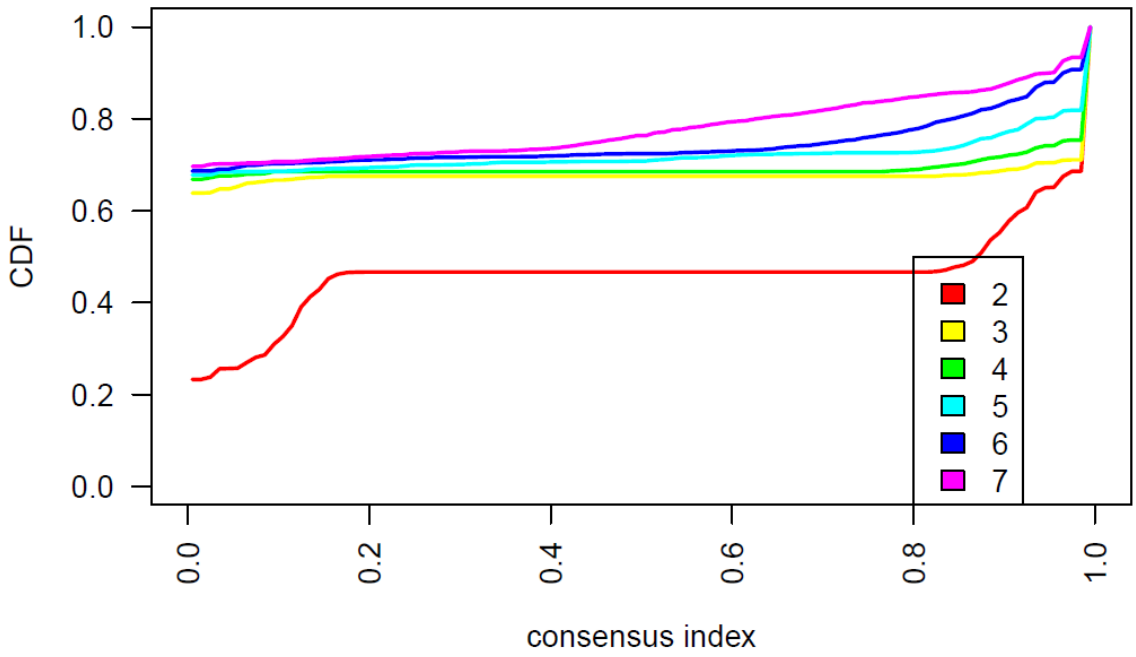

Figura S17

Heatmap de la matriz de consenso sobre ESPA IgG para k = 6

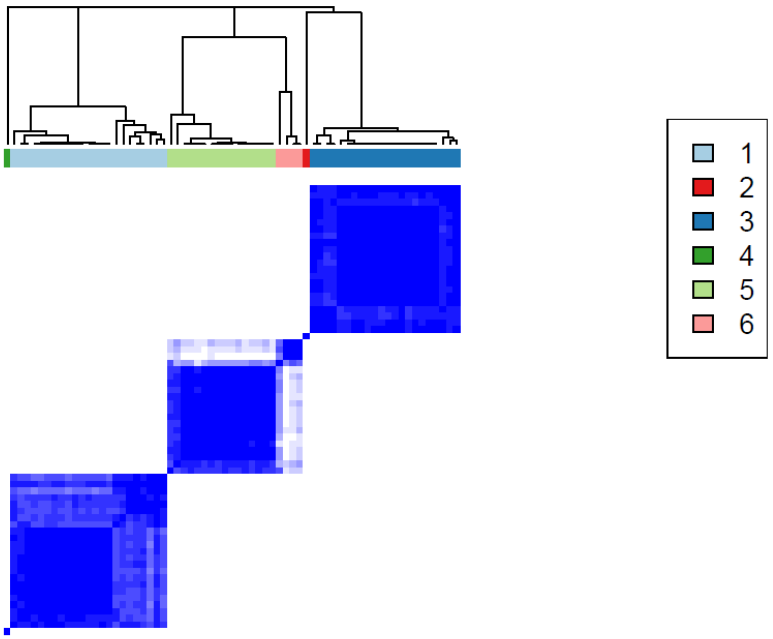

Tabla S2

Individuos y clúster asignado para ESPA IgG

ID del paciente	Clúster	ID del paciente	Clúster	ID del paciente	Clúster
LLC-1	1	LLC-38	3	LLC-6	1
LLC-10	1	LLC-39	3	LLC-61	5
LLC-11	1	LLC-4	1	LLC-62	5
LLC-17	1	LLC-40	3	LLC-63	5
LLC-18	1	LLC-41	3	LLC-64	5
LLC-19	1	LLC-42	3	LLC-65	6
LLC-20	1	LLC-43	3	LLC-66	5
LLC-21	1	LLC-44	3	LLC-67	5
LLC-22	1	LLC-45	3	LLC-68	5
LLC-23	1	LLC-46	3	LLC-69	5
LLC-24	1	LLC-47-1	3	LLC-7	1
LLC-25	1	LLC-48	3	LLC-70	6
LLC-26	1	LLC-49	3	LLC-71	5
LLC-28	1	LLC-5	1	LLC-72	5
LLC-29	1	LLC-50	3	LLC-74	5
LLC-3	1	LLC-52	3	LLC-76	6
LLC-30	1	LLC-53	3	LLC-77	5
LLC-32	2	LLC-54	3	LLC-78	5
LLC-33	3	LLC-55	4	LLC-79	5
LLC-34	3	LLC-56	3	LLC-8	1
LLC-35	3	LLC-57	5	LLC-9	1
LLC-36	3	LLC-58	5		
LLC-37	3	LLC-59	6		

Figura S18

Función de Distribución Acumulativa de Consenso para ESPA IgM

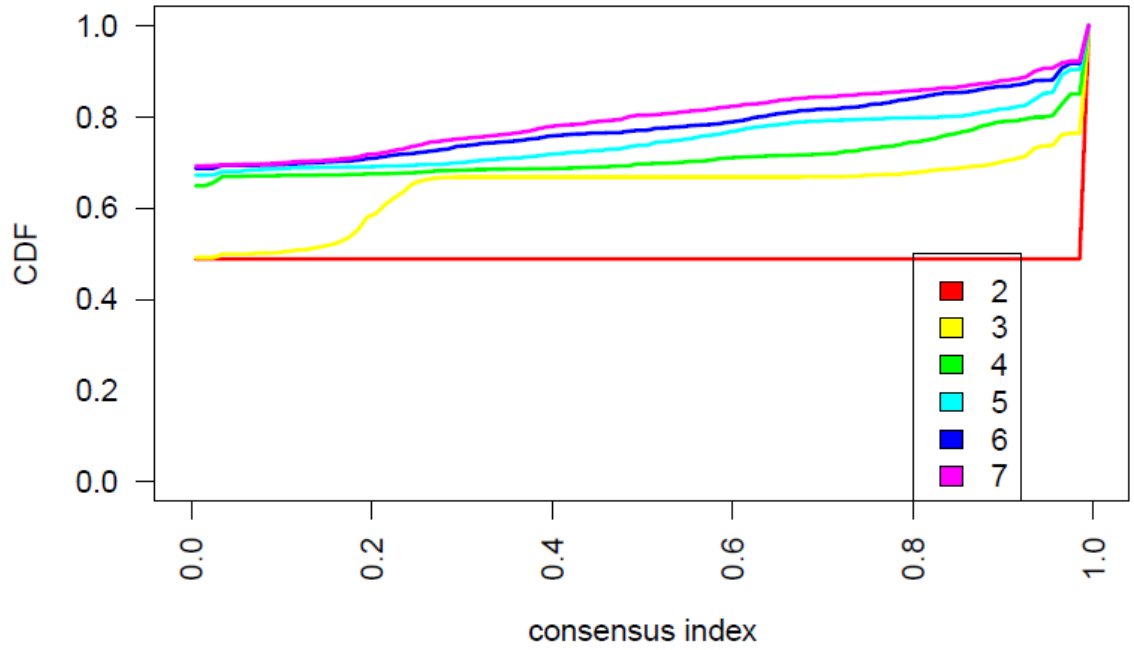

Figura S19

Heatmap de la matriz de consenso sobre ESPA IgM para k = 5

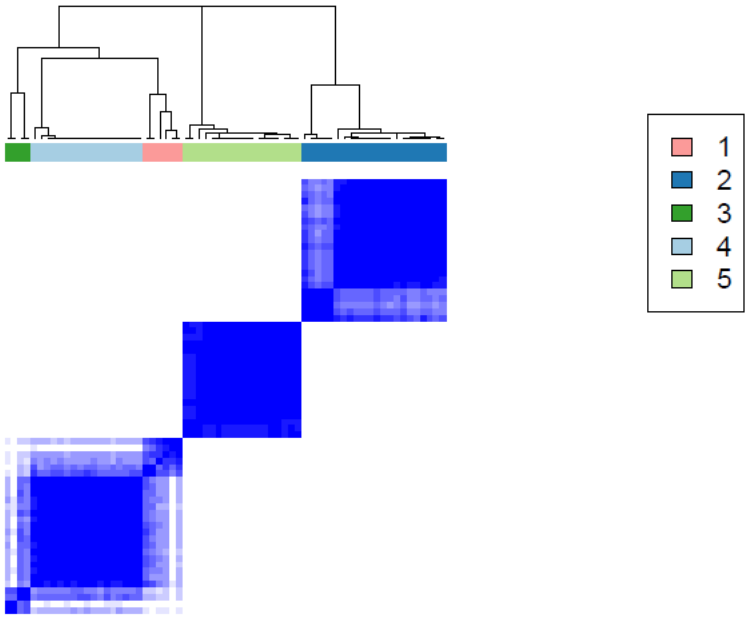

Tabla S3

Individuos y clúster asignado para ESPA IgM

ID del paciente	Clúster	ID del paciente	Clúster	ID del paciente	Clúster
LLC-1	1	LLC-38	4	LLC-6	2
LLC-10	2	LLC-39	4	LLC-61	5
LLC-11	2	LLC-4	2	LLC-62	5
LLC-17	2	LLC-40	4	LLC-63	3
LLC-18	2	LLC-41	4	LLC-64	5
LLC-19	2	LLC-42	4	LLC-65	5
LLC-20	2	LLC-43	4	LLC-66	5
LLC-21	2	LLC-44	4	LLC-67	5
LLC-22	2	LLC-45	3	LLC-68	5
LLC-23	2	LLC-46	4	LLC-69	5
LLC-24	2	LLC-47-1	4	LLC-7	2
LLC-25	2	LLC-48	4	LLC-70	2
LLC-26	2	LLC-49	4	LLC-71	5
LLC-28	2	LLC-5	2	LLC-72	5
LLC-29	2	LLC-50	1	LLC-74	5
LLC-3	2	LLC-52	4	LLC-76	5
LLC-30	3	LLC-53	1	LLC-77	5
LLC-32	4	LLC-54	1	LLC-78	5
LLC-33	4	LLC-55	1	LLC-79	5
LLC-34	4	LLC-56	1	LLC-8	2
LLC-35	4	LLC-57	5	LLC-9	2
LLC-36	3	LLC-58	5		
LLC-37	4	LLC-59	5		

Figura S20

Función de Distribución Acumulativa de Consenso para Immunocheckpoints

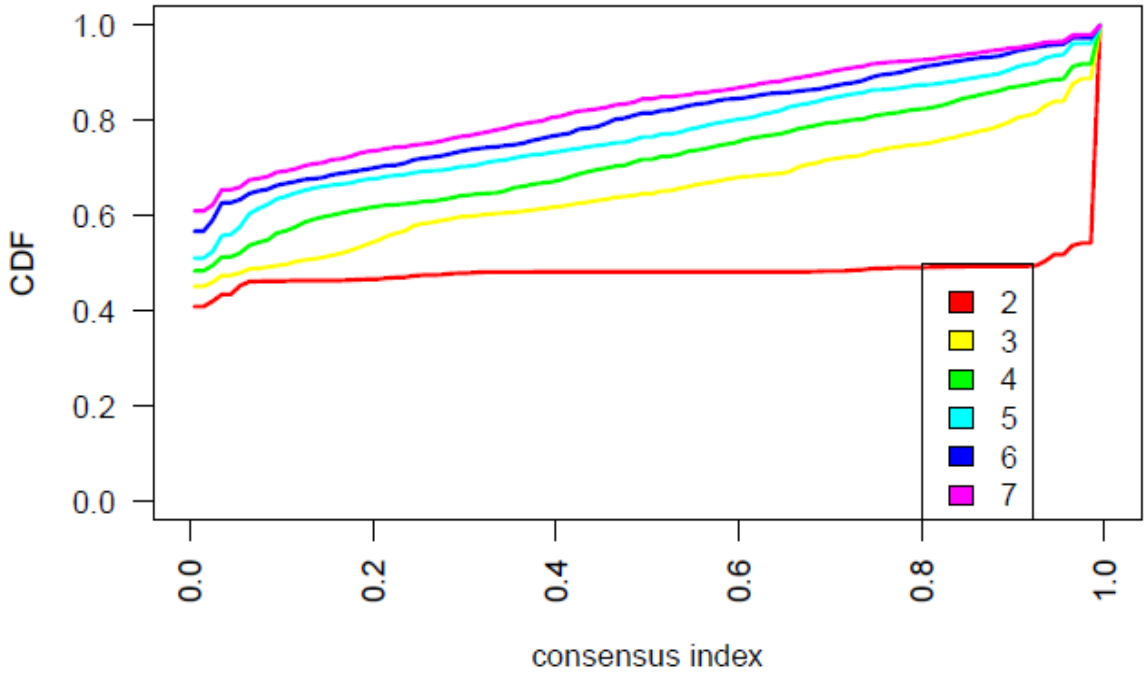

Figura S21

Heatmap de la matriz de consenso sobre Immunocheckpoints para k = 5

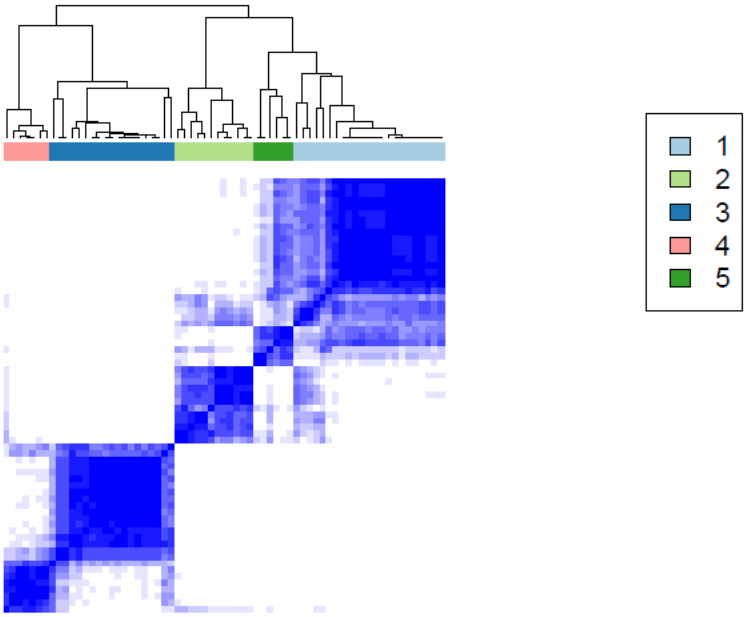

Tabla S4

Individuos y clúster asignado para Immunocheckpoints

ID del paciente	Clúster	ID del paciente	Clúster	ID del paciente	Clúster
LLC-1	1	LLC-38	2	LLC-6	1
LLC-10	2	LLC-39	1	LLC-61	4
LLC-11	3	LLC-4	1	LLC-62	2
LLC-17	3	LLC-40	1	LLC-63	5
LLC-18	1	LLC-41	1	LLC-64	4
LLC-19	1	LLC-42	2	LLC-65	1
LLC-20	3	LLC-43	3	LLC-66	2
LLC-21	1	LLC-44	1	LLC-67	5
LLC-22	3	LLC-45	3	LLC-68	4
LLC-23	3	LLC-46	2	LLC-69	1
LLC-24	3	LLC-47-1	3	LLC-7	1
LLC-25	3	LLC-48	1	LLC-70	4
LLC-26	2	LLC-49	1	LLC-71	3
LLC-28	2	LLC-5	4	LLC-72	2
LLC-29	2	LLC-50	1	LLC-74	5
LLC-3	1	LLC-52	5	LLC-76	3
LLC-30	3	LLC-53	3	LLC-77	3
LLC-32	3	LLC-54	3	LLC-78	1
LLC-33	1	LLC-55	2	LLC-79	5
LLC-34	1	LLC-56	2	LLC-8	4
LLC-35	3	LLC-57	4	LLC-9	3
LLC-36	1	LLC-58	5		
LLC-37	1	LLC-59	1		

Figura S22

Función de Distribución Acumulativa de Consenso para Proteoma

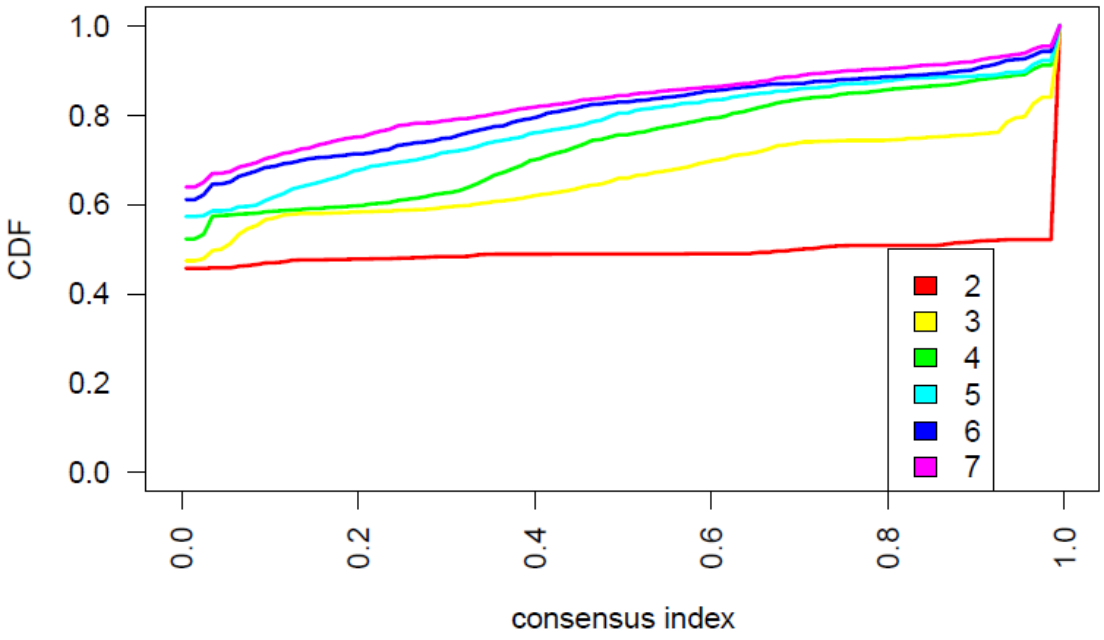

Figura S23

Heatmap de la matriz de consenso sobre Proteoma para k = 6

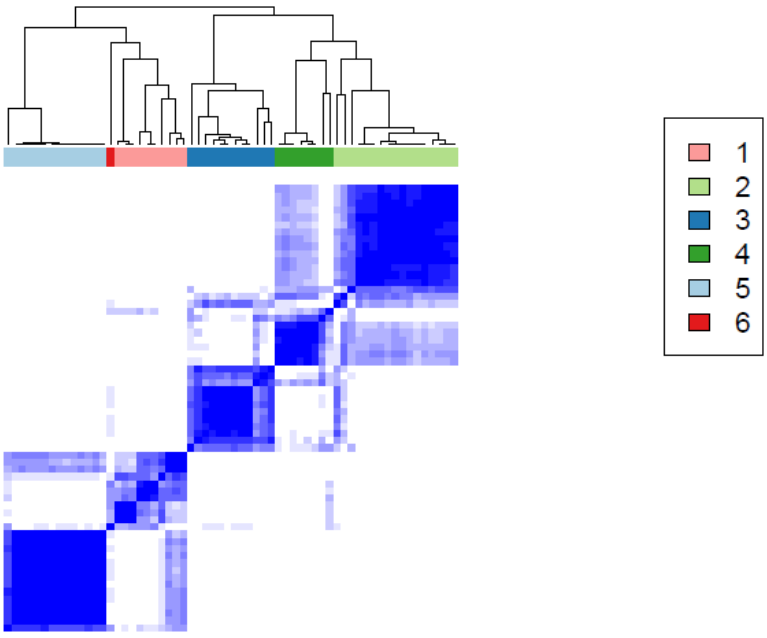

Tabla S5

Individuos y clúster asignado para Proteoma

ID del paciente	Clúster	ID del paciente	Clúster	ID del paciente	Clúster
LLC-10	1	LLC-37	5	LLC-6	1
LLC-11	1	LLC-38	2	LLC-61	2
LLC-17	2	LLC-39	5	LLC-63	2
LLC-18	3	LLC-4	1	LLC-64	2
LLC-19	1	LLC-41	5	LLC-65	2
LLC-20	3	LLC-42	5	LLC-66	5
LLC-21	3	LLC-43	2	LLC-67	5
LLC-22	2	LLC-44	2	LLC-68	2
LLC-23	4	LLC-45	5	LLC-69	3
LLC-24	4	LLC-46	5	LLC-7	1
LLC-25	4	LLC-47-1	5	LLC-70	3
LLC-26	4	LLC-48	2	LLC-71	3
LLC-28	1	LLC-49	2	LLC-72	3
LLC-29	1	LLC-5	4	LLC-74	3
LLC-3	5	LLC-50	2	LLC-76	3
LLC-30	4	LLC-52	5	LLC-77	3
LLC-32	6	LLC-53	2	LLC-78	3
LLC-33	1	LLC-54	5	LLC-79	4
LLC-34	4	LLC-56	2	LLC-8	1
LLC-35	2	LLC-58	2	LLC-9	3
LLC-36	5	LLC-59	5		

2 CÓDIGO EN R

2.1 ANÁLISIS DESCRIPTIVO

Se cargan los paquetes:

```
> library(readxl)
> library(ggplot2)
> library(grid)
> library(gridExtra)
```

Preprocesamiento de datos:

```
> datos <- as.data.frame(read_excel("metadatos.xlsx", na = "NA"))
> rownames(datos)<-datos[,1]
> datos <- datos[,-1]
> datos$Gender <- as.factor(datos$Gender)
> datos$`Disease stage` <- as.factor(datos$`Disease stage`)
> datos$`Disease evolution` <- as.factor(datos$`Disease evolution`)
> datos$`Disease evolution` <- factor(datos$`Disease evolution`, levels = levels(datos$`Disease evolution`)[c(1,3,2)])
> datos$`Dx according to therapy response` <- as.factor(datos$`Dx according to therapy response`)
> datos$Treatment <- as.factor(datos$Treatment)
> datos$`IGHV mutational status` <- as.factor(datos$`IGHV mutational status`)
> datos$Karyotype <- as.factor(datos$Karyotype)
> datos$`Chromosomal aberrations` <-as.factor(datos$`Chromosomal aberrations`)
```

Código de la Figura 24: Análisis descriptivo de las variables clínico-biológicas

```
> g1 <- ggplot(data = datos, aes(x = Gender)) + geom_bar(fill = c("palegreen2", "tan2"), stat = "count", position = position_dodge())+ geom_text(aes(label =..count..), stat = 'count', position = position_dodge(0.9), vjust= -1, size = 4.0) + ylim(c(0,50)) + theme_minimal() + theme(axis.text = element_text(size = 11))+ labs(title = "Gender") + xlab("")
> g2 <- ggplot(data = datos, aes(x =`Disease stage`)) + geom_bar( fill = c("palegreen2", "tan2"),stat = "count", position =
```

```
position_dodge())  +  geom_text(aes(label  =  ..count..),stat  =
'count', position = position_dodge(0.9),Vjust = -1, size = 4.0) +
ylim(c(0,65))    +    theme_minimal()    +    theme(axis.text    =
element_text(size = 11))+ labs(title = "Disease stage") + xlab("")

> g3 <- ggplot(data = datos, aes(x =`Disease evolution`)) +
geom_bar( fill = c("palegreen2", "lightskyblue", "tan2"), stat =
"count", position = position_dodge()) + geom_text(aes(label =
..count..), stat = 'count', position = position_dodge(0.9), vjust
= -1, size = 4.0) + ylim(c(0, 50)) + theme_minimal()+
theme(axis.text = element_text(size = 11))+ labs(title = "Disease
evolution")+ xlab("")

> g4 <- ggplot(data = datos, aes(x =`Dx according to therapy
response`)) + geom_bar( fill = c("palegreen2", "lightskyblue",
"khaki1", "tan2", "indianred"), stat = "count", position =
position_dodge())  +  geom_text(aes(label  =  ..count..),stat  =
'count', position = position_dodge(0.9), vjust = -1, size = 4.0) +
ylim(c(0,50))    +    theme_minimal()+    theme(axis.text    =
element_text(size = 11))+ labs(title = "Dx according to therapy
response") + xlab("")

> g5 <- ggplot(data = datos, aes(x = Treatment)) + geom_bar( fill
=  c("palegreen2",  "tan2"),stat  =  "count",  position  =
position_dodge())  +  geom_text(aes(label  =  ..count..),  stat  =
'count', position = position_dodge(0.9), vjust=-1, size = 4.0) +
ylim(c(0,70))    +    theme_minimal()    +    theme(axis.text    =
element_text(size = 11))+ labs(title = "Treatment") + xlab("")

> g6 <- ggplot(data = datos, aes(x =`IGHV mutational status`)) +
geom_bar( fill = c("palegreen2", "tan2"),stat = "count", position
= position_dodge())  +  geom_text(aes(label  =  ..count..),stat  =
'count', position = position_dodge(0.9), vjust = -1, size = 4.0) +
ylim(c(0,50))    +    theme_minimal()    +    theme(axis.text    =
element_text(size = 11))+ labs(title = "IGHV mutational status") +
xlab("")

> g7 <- ggplot(data = datos, aes(x = Karyotype)) + geom_bar( fill
=  c("palegreen2",  "tan2"),stat  =  "count",  position  =
position_dodge())  +  geom_text(aes(label  =  ..count..),stat  =
'count', position = position_dodge(0.9), vjust = -1, size = 4.0) +
ylim(c(0,50))    +    theme_minimal()    +    theme(axis.text    =
element_text(size = 11))+ labs(title = "Karyotype") + xlab("")

> g8 <- ggplot(data = datos, aes(x =`Chromosomal aberrations`)) +
geom_bar( fill = c("palegreen2", "lightskyblue", "khaki1", "tan2",
"indianred"),stat  =  "count",  position  =  position_dodge())  +
geom_text(aes(label  =  ..count..),  stat  =  'count',  position  =
position_dodge(0.9),  vjust=-1,  size  =  4.0)  +  ylim(c(0,40))  +
theme_minimal()+  theme(axis.text  =  element_text(size  =  11))  +
labs(title = "Chromosomal aberrations") + xlab("")

> g9 <- ggplot(data = datos, aes(x = Age)) + geom_histogram(aes(y
= ..density..), bins = 15, fill = I("lightskyblue"), colour =
I("black") + geom_density() + theme_minimal() + theme(axis.text =
element_text(size = 11)) + labs(title = "Age") + xlab("")

> grid.arrange(g1, g9, g2, g3, g4, g5, g6, g7, g8, ncol = 3)
```

Código de la Figura 7: Boxplots de los datos originales

```
> crudos <- as.data.frame(read_excel("DatosCrudos.xlsx"))

> crudos$variable <- as.factor(crudos$variable)

> crudos$variable <- factor(crudos$variable, levels =
levels(crudos$variable)[c(1, 43, 7, 22, 18, 15, 9, 2, 3, 8, 16,
17, 10, 11, 4, 12, 14, 19, 35, 5, 13, 20, 21, 36, 40, 6, 42, 37,
39, 38, 41, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34 )])

> ggplot(data = crudos) + geom_boxplot(aes(x = variable, y =
value, fill = datos), outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal() +  theme(axis.text =
element_text(size = 11), legend.text = element_text(size = 10),
title = element_text(size=15)) + labs(title = "Datos Originales" )
+ xlab("") + ylim(0,20000)
```

Código de la Figura 8: Boxplots de los datos estandarizados

```
> zscore <- as.data.frame(read_excel("DatosZ.xlsx"))

> zscore$variable <- as.factor(zscore$variable)

> zscore$variable <- factor(zscore$variable, levels =
levels(zscore$variable)[c(1, 43, 7, 22, 18, 15, 9, 2, 3, 8, 16,
17, 10, 11, 4, 12, 14, 19, 35, 5, 13, 20, 21, 36, 40, 6, 42, 37,
39, 38, 41, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34 )])

> ggplot(data = zscore) + geom_boxplot(aes(x = variable, y =
value, fill = datos), outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal()+  theme(axis.text =
element_text(size = 11), legend.text = element_text(size = 10),
title = element_text(size = 15)) + labs(title = "Datos Z-Score") +
xlab("")
```

Código de la Figura S1: Boxplots de los datos originales de la base de datos Autoanticuerpos IgG

```
> Auto_igg <- as.data.frame(read_excel("Datos
Expresión/Autoanticuerpos IgG.xlsx", na = "NA"))

> rownames(Auto_igg) <- Auto_igg[,1]

> Auto_igg <- Auto_igg[,-1]

> x <- as.data.frame(Auto_igg[,c(10:131)])

> z <- as.data.frame(scale(x))

> data_x <- melt(x)

> data_z <- melt(z)

> ggplot(data = data_x) + geom_boxplot(aes(x = variable, y =
value), fill = "indianred1", outlier.shape = NA) + guides(x =
guide_axis(angle = 90)) + theme_minimal() + theme(axis.text =
element_text(size = 7))
```

Código de la Figura S2: Boxplots de los datos estandarizados de la base de datos Autoanticuerpos IgG

```
> ggplot(data = data_z) + geom_boxplot(aes(x = variable, y =
value), fill = "indianred1", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal() +  theme(axis.text =
element_text(size = 7)) + ylim(-3,4)
```

Código de la Figura S3: Boxplots de los datos originales de la base de datos Autoanticuerpos IgM

```
> Auto_igm <- as.data.frame(read_excel("Datos
Expresión/Autoanticuerpos IgM.xlsx", na = "NA"))

> rownames(Auto_igm)<-Auto_igm[,1]

> Auto_igm<-Auto_igm[,-1]

> x <- as.data.frame(Auto_igm[,c(10:131)])

> x <- x[,-c(11,26,48,55,84,86,91,92,102,106,110,112)]

> z <- as.data.frame(scale(x))

> data_x <- melt(x)

> data_z <- melt(z)

> ggplot(data = data_x) + geom_boxplot(aes(x = variable, y =
value),fill = "#CD950C", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal()+ theme(axis.text =
element_text(size=7))
```

Código de la Figura S4: Boxplots de los datos estandarizados de la base de datos Autoanticuerpos IgM

```
> ggplot(data = data_z) + geom_boxplot(aes(x = variable, y =
value), fill = "#CD950C", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal() +  theme(axis.text =
element_text(size = 7))
```

Código de la Figura S5: Boxplots de los datos originales de la base de datos ESPA IgG

```
> ESPA_igg <- as.data.frame(read_excel("Datos Expresión/ESPA
IgG.xlsx", na = "NA"))

> rownames(ESPA_igg)<-ESPA_igg[,1]

> ESPA_igg <- ESPA_igg[,-1]

> x <- as.data.frame(ESPA_igg[,c(10:46)])

> z <- as.data.frame(scale(x))
```

```
> data_x <- melt(x)

> data_z<-melt(z)

> ggplot(data = data_x) + geom_boxplot(aes(x = variable,y =
value),fill = "green3", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal()+ theme(axis.text =
element_text(size = 8)) + ylim(-3,3)
```

Código de la Figura S6: Boxplots de los datos estandarizados de la base de datos ESPA IgG

```
> ggplot(data = data_z) + geom_boxplot(aes(x = variable, y =
value),fill = "green3", outlier.shape = NA) + guides(x =
guide_axis(angle = 90)) + theme_minimal() +  theme(axis.text =
element_text(size = 8)) + ylim(-2,3.5)
```

Código de la Figura S7: Boxplots de los datos originales de la base de datos ESPA IgM

```
> ESPA_igm <- as.data.frame(read_excel("Datos Expresión/ESPA
IgM.xlsx", na = "NA"))

> rownames(ESPA_igm)<-ESPA_igm[,1]

> ESPA_igm<-ESPA_igm[,-1]

> x <- as.data.frame(ESPA_igm[,c(10:46)])

> z <- as.data.frame(scale(x))

> data_x <- melt(x)

> data_z <- melt(z)

> ggplot(data = data_x) + geom_boxplot(aes(x = variable, y =
value),fill = "cyan3", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal()+ theme(axis.text =
element_text(size = 8))
```

Código de la Figura S8: Boxplots de los datos estandarizados de la base de datos ESPA IgM

```
> ggplot(data = data_z) + geom_boxplot(aes(x = variable, y =
value),fill = "cyan3", outlier.shape = NA) + guides(x =
guide_axis(angle = 90)) + theme_minimal() +  theme(axis.text =
element_text(size = 8)) + ylim(-2,3.5)
```

Código de la Figura S9: Boxplots de los datos originales de la base de datos Immunocheckpoints

```
> Immunocheckpoints <- as.data.frame(read_excel("Datos
Expresión/Immunocheckpoints.xlsx", na = "NA"))

> rownames(Immunocheckpoints)<-Immunocheckpoints[,1]

> Immunocheckpoints<-Immunocheckpoints[,-1]

> x <- as.data.frame(Immunocheckpoints[,c(10:112)])

> z <- as.data.frame(scale(x))

> data_x <- melt(x)

> data_z <- melt(z)

> ggplot(data = data_x) + geom_boxplot(aes(x = variable, y =
value), fill = "steelblue2", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal()+ theme(axis.text =
element_text(size = 7)) + ylim(c(0,3000))
```

Código de la Figura S10: Boxplots de los datos estandarizados de la base de datos Immunocheckpoints

```
> ggplot(data = data_z) + geom_boxplot(aes(x = variable, y =
value),fill = "steelblue2", outlier.shape = NA) + guides(x =
guide_axis(angle = 90)) + theme_minimal() +  theme(axis.text =
element_text(size = 8)) + ylim(-2,3)
```

Código de la Figura S11: Boxplots de los datos originales de la base de datos Proteoma

```
> Proteoma <- as.data.frame(read_excel("Datos
Expresión/Proteoma.xlsx", na = "NA"))

> rownames(Proteoma <- Proteoma[,1]

> Proteoma <- Proteoma[,-1]

> x <- as.data.frame(Proteoma[,c(10:2955)])

> z <- as.data.frame(scale(x))

> data_x <- melt(x)

> data_z <- melt(z)

> ggplot(data = data_x) + geom_boxplot(aes(x = variable, y =
value),fill = "orchid2", outlier.shape = NA) + guides(x =
guide_axis(angle = 90))+ theme_minimal()+ theme(axis.text =
element_text(size =8), axis.text.x = element_text(size = 2))
```

Código de la Figura S12: Boxplots de los datos estandarizados de la base de datos Proteoma

```
> ggplot(data = data_z) + geom_boxplot(aes(x = variable, y =
value),fill = "orchid2", outlier.shape = NA) + guides(x =
guide_axis(angle = 90)) + theme_minimal() +  theme(axis.text =
element_text(size = 8), axis.text.x = element_text(size = 2))
```

2.2 MULTI-OMIS FACTOR ANALYSIS (MOFA)

Código para la elaboración del modelo MOFA

```
# Cargamos las librerias

> library(readxl)

> library(ggplot2)

> library(MOFA2)

> library(utils)

> library(corrplot)


# Introducimos y preprocesamos los datos

> metadata <- as.data.frame(read_excel("metadatos.xlsx", na =
"NA"))

> rownames(metadata) <- metadata[,1]

> metadata$Gender <- as.factor(metadata$Gender)

> metadata$`Disease stage` <- as.factor(metadata$`Disease stage`)

> metadata$`Disease evolution` <- as.factor(metadata$`Disease
evolution`)

> metadata$`Dx according to therapy response` <-
as.factor(metadata$`Dx according to therapy response`)

> metadata$Treatment <-as.factor(metadata$Treatment)

> metadata$`IGHV mutational status` <- as.factor(metadata$`IGHV
mutational status`)

> metadata$Karyotype<-as.factor(metadata$Karyotype)

> metadata$`Chromosomal aberrations`<-
as.factor(metadata$`Chromosomal aberrations`)

> datos <- as.data.frame(read_excel("MOFA final.xlsx"))


# Creamos un objeto MOFA

> data_opts <- get_default_data_options(mofaobj)

> model_opts <- get_default_model_options(mofaobj)
```

```
> model_opts$num_factors <- 12

> train_opts <- get_default_training_options(mofaobj)

> train_opts$convergence_mode <- "medium"


# Entrenamos y ejecutamos el modelo creado

> mofaobj <- prepare_mofa(object = mofaobj, data_options =
data_opts, model_options = model_opts, training_options =
train_opts)

> mofaobj <- run_mofa(mofaobj)


# Añadimos las variables clínico-biológicas al modelo en forma de
metadatos

> samples_metadata(mofaobj) <- metadata
```

Código de la Figura 10: Gráfico de correlaciones entre los factores

```
> z <- get_factors(mofaobj)

> x <- do.call(rbind, z)

> y <- do.call(rbind, z)

> mc <- cor(x, y, method = "pearson")

> c <- cor.mtest(as.matrix(x), type = "pearson")

> corrplot(mc, p.mat = c$p, insig = "blank", type = "upper",method
= "circle",diag = F, tl.col = "black")
```

Código de la Figura 11: Varianza explicada por cada uno de los factores en cada vista

```
> plot_variance_explained(mofaobj, max_r2 = 15) + guides(x =
guide_axis(angle = 45))
```

Código de la Figura S25: Factores de MOFA que explican al menos el 1.5% de la variabilidad de cada vista

```
> plot_variance_explained(mofaobj, max_r2 = 1.5, min_r2 = 1.5)+
guides(x = guide_axis(angle = 45))
```

Código de la Figura 12: Varianza explicada por todos los factores en cada vista

```
> plot_variance_explained(mofaobj, plot_total = T)[[2]] + guides(x
= guide_axis(angle = 45))
```

2.3 HEATMAP DE LOS PESOS

```
# Cargamos las librerías

> library(ggplot2)

> library(MOFA2)

> library(tidyverse)

> library(colorspace)


# Top10 características (Más de 1,5% de varianza explicada)

> mofa_weights <- get_weights(mofaobj, views = "all", factors =
"all", abs = FALSE, scale = TRUE, as.data.frame = FALSE)

> top10_F1_AutoIgG <- names(sort(abs(mofa_weights$`Autoanticuerpos
IgG`[, "Factor1"]), decreasing = TRUE )[1:10])

> top10_F9_AutoIgG <- names(sort(abs(mofa_weights$`Autoanticuerpos
IgG`[, "Factor9"]), decreasing = TRUE )[1:10])

> top10_F2_AutoIgM <- names(sort(abs(mofa_weights$`Autoanticuerpos
IgM`[, "Factor2"]), decreasing = TRUE )[1:10])

> top10_F3_AutoIgM <- names(sort(abs(mofa_weights$`Autoanticuerpos
IgM`[, "Factor3"]), decreasing = TRUE )[1:10])

> top10_F1_ESPAIgG <- names(sort(abs(mofa_weights$`ESPA IgG`[,
"Factor1"]), decreasing = TRUE )[1:10])

> top10_F3_ESPAIgG <- names(sort(abs(mofa_weights$`ESPA IgG`[,
"Factor3"]), decreasing = TRUE )[1:10])

> top10_F6_ESPAIgG <- names(sort(abs(mofa_weights$`ESPA IgG`[,
"Factor6"]), decreasing = TRUE )[1:10])

> top10_F7_ESPAIgG <- names(sort(abs(mofa_weights$`ESPA IgG`[,
"Factor7"]), decreasing = TRUE )[1:10])

> top10_F10_ESPAIgG <- names(sort(abs(mofa_weights$`ESPA IgG`[,
"Factor10"]), decreasing = TRUE )[1:10])

> top10_F1_ESPAIgM <- names(sort(abs(mofa_weights$`ESPA IgM`[,
"Factor1"]), decreasing = TRUE )[1:10])

> top10_F3_ESPAIgM <- names(sort(abs(mofa_weights$`ESPA IgM`[,
"Factor3"]), decreasing = TRUE )[1:10])

> top10_F4_ESPAIgM <-names(sort(abs(mofa_weights$`ESPA IgM`[,
"Factor4"]), decreasing = TRUE )[1:10])

> top10_F5_ESPAIgM <- names(sort(abs(mofa_weights$`ESPA IgM`[,
"Factor5"]), decreasing = TRUE )[1:10])

> top10_F6_ESPAIgM <- names(sort(abs(mofa_weights$`ESPA IgM`[,
"Factor6"]), decreasing = TRUE )[1:10])

> top10_F7_ESPAIgM <- names(sort(abs(mofa_weights$`ESPA IgM`[,
"Factor7"]), decreasing = TRUE )[1:10])
```

```
> top10_F1_Immuno <-
names(sort(abs(mofa_weights$Immunocheckpoints[, "Factor1"]),
decreasing = TRUE )[1:10])

> top10_F3_Immuno <-
names(sort(abs(mofa_weights$Immunocheckpoints[, "Factor3"]),
decreasing = TRUE )[1:10])

> top10_F4_Immuno <-
names(sort(abs(mofa_weights$Immunocheckpoints[, "Factor4"]),
decreasing = TRUE )[1:10])

> top10_F8_Immuno <-
names(sort(abs(mofa_weights$Immunocheckpoints[, "Factor8"]),
decreasing = TRUE )[1:10])

> top10_F5_Proteoma <- names(sort(abs(mofa_weights$Proteoma[,
"Factor5"]), decreasing = TRUE )[1:10])

> top10_F6_Proteoma <- names(sort(abs(mofa_weights$Proteoma[,
"Factor6"]), decreasing = TRUE )[1:10])

> top10_F7_Proteoma <- names(sort(abs(mofa_weights$Proteoma[,
"Factor7"]), decreasing = TRUE )[1:10])

> top10_F9_Proteoma <- names(sort(abs(mofa_weights$Proteoma[,
"Factor9"]), decreasing = TRUE )[1:10])

> top10_F10_Proteoma <- names(sort(abs(mofa_weights$Proteoma[,
"Factor10"]), decreasing = TRUE )[1:10])

> top10_F11_Proteoma <- names(sort(abs(mofa_weights$Proteoma[,
"Factor11"]), decreasing = TRUE )[1:10])

> top10_F12_Proteoma<-names(sort(abs(mofa_weights$Proteoma[,
"Factor12"]), decreasing = TRUE )[1:10])


# Juntamos los nombres de todas las características en un vector

> all_symbols<-c(top10_F1_AutoIgG, top10_F9_AutoIgG,
top10_F2_AutoIgM, top10_F3_AutoIgM, top10_F1_ESPAIgG,
top10_F3_ESPAIgG, top10_F6_ESPAIgG, top10_F7_ESPAIgG,
top10_F10_ESPAIgG, top10_F1_ESPAIgM, top10_F3_ESPAIgM,
top10_F4_ESPAIgM, top10_F5_ESPAIgM, top10_F6_ESPAIgM,
top10_F7_ESPAIgM, top10_F1_Immuno, top10_F3_Immuno,
top10_F4_Immuno, top10_F8_Immuno, top10_F5_Proteoma,
top10_F6_Proteoma, top10_F7_Proteoma, top10_F9_Proteoma,
top10_F10_Proteoma,top10_F11_Proteoma,top10_F12_Proteoma)


# Separamos las características por bases de datos

> AutoIgG_symbols <- subset(all_symbols, all_symbols %in%
rownames(mofa_weights$`Autoanticuerpos IgG`)) %>% unique

> AutoIgM_symbols <- subset(all_symbols, all_symbols %in%
rownames(mofa_weights$`Autoanticuerpos IgM`)) %>% unique
```

```
> ESPAIgG_symbols <- subset(all_symbols, all_symbols %in%
rownames(mofa_weights$`ESPA IgG`)) %>% unique

> ESPAIgM_symbols <- subset(all_symbols, all_symbols %in%
rownames(mofa_weights$`ESPA IgM`)) %>% unique

> Immuno_symbols <- subset(all_symbols, all_symbols %in%
rownames(mofa_weights$Immunocheckpoints)) %>% unique

> Proteoma_symbols <- subset(all_symbols, all_symbols %in%
rownames(mofa_weights$Proteoma)) %>% unique


# Se junta toda la información en una única base de datos

> weights_Fs_sel <- bind_rows(

mofa_weights$`Autoanticuerpos IgG`[AutoIgG_symbols, c("Factor1",
"Factor9")] %>%

  as.data.frame() %>%

  rownames_to_column(., var="symbol") %>%

  mutate(omic="Autoanticuerpos IgG"),


mofa_weights$`Autoanticuerpos IgM`[AutoIgM_symbols, c("Factor2",
"Factor3")] %>%

  as.data.frame() %>%

  rownames_to_column(., var="symbol") %>%

  mutate(omic="Autoanticuerpos IgM"),


mofa_weights$`ESPA IgG`[ESPAIgG_symbols, c("Factor1", "Factor3",
"Factor6", "Factor7", "Factor10")] %>%

  as.data.frame() %>%

  rownames_to_column(., var="symbol") %>%

  mutate(omic="ESPA IgG"),


mofa_weights$`ESPA IgM`[ESPAIgM_symbols, c("Factor1","Factor3",
"Factor4", "Factor5", "Factor6", "Factor7")] %>%

  as.data.frame() %>%

  rownames_to_column(., var="symbol") %>%

  mutate(omic="ESPA IgM"),
```

```
mofa_weights$Immunocheckpoints[Immuno_symbols, c("Factor1",
"Factor3", "Factor4", "Factor8")] %>%
  as.data.frame() %>%
  rownames_to_column(., var="symbol") %>%
  mutate(omic="Immunocheckpoints"),


mofa_weights$Proteoma[Proteoma_symbols, c("Factor5", "Factor6",
"Factor7", "Factor9", "Factor10", "Factor11", "Factor12")] %>%
  as.data.frame() %>%
  rownames_to_column(., var="symbol") %>%
  mutate(omic="Proteoma"))


> weights_Fs_sel <- weights_Fs_sel %>%
  as_tibble() %>%
  pivot_longer(c("Factor1","Factor2", "Factor3", "Factor4",
"Factor5","Factor6","Factor7","Factor8","Factor9","Factor10","Fact
or11","Factor12"), names_to = "Factores", values_to = "weights")


# Se ordenan las características y los factores para que aparezcan
en el gráfico en el orden elegido
> order_symbols <- weights_Fs_sel %>%
  filter(!is.na(weights)) %>%
  group_by(symbol) %>%
  arrange(omic, desc(abs(weights) )) %>%
  dplyr::slice(1) %>%
  ungroup() %>%
  arrange(omic, Factores, weights  ) %>%
  .$symbol %>% rev


> heatmap_weights <- weights_Fs_sel %>%
filter(!is.na(weights)) %>%
mutate(symbol = as.factor(symbol)) %>%
mutate(symbol = factor(symbol, levels = order_symbols)) %>%
mutate(Factores = as.factor(Factores)) %>%
mutate(Factores = factor(Factores, levels = c("Factor1","Factor2",
"Factor3", "Factor4", "Factor5", "Factor6", "Factor7", "Factor8",
"Factor9", "Factor10", "Factor11", "Factor12")))
```

```
# Creamos el gráfico

> ggplot(heatmap_weights, aes( omic, symbol, fill = weights )) +
geom_tile(color = "black") + facet_grid(~Factores) +
scale_fill_continuous_divergingx(palette = 'RdBu', mid = 0) +
theme(axis.text.x = element_text(angle = 90, size = 4),
axis.text.y = element_text(size = 2), axis.title =
element_blank(), panel.background = element_rect(fill = "grey90"),
legend.title = element_text(size = 8), legend.key.width  =
unit(0.3, units = "cm"), legend.text = element_text(size = 7)  )
```

2.4 CONSENSUS CLUSTERING

```
# Cargamos los paquetes

> library(MOFA2)

> library(ConsensusClusterPlus)

> library(readxl)

> library(cluster)


# Extraemos los datos de mofa

> datos <- get_data(mofaobj)


# Creamos el algoritmo de clustering

> diana_alg <- function(mat, n){

 x <- diana(mat,diss=TRUE)

 a <- cutree(x, n)

 return(a) }


# Autoanticuerpos IgG

> autoigg <-datos$`Autoanticuerpos IgG`$single_group

      # Sustituir NAs por la media de las columnas

> medias<-rowMeans(autoigg, na.rm = TRUE)

> for (x in 1:122) {

  autoigg[x, is.na(autoigg[x,])] <- medias[x]

}

> n <-ConsensusClusterPlus(autoigg, maxK = 7, reps = 50,
distance="pearson", clusterAlg="diana_alg",seed=856814)
```

```
# Autoanticuerpos IgM
> autoigm <- datos$`Autoanticuerpos IgM`$single_group
      # Sustituir NAs por la media de las columnas
> medias<-rowMeans(autoigm, na.rm = TRUE)
> for (x in 1:110) {
  autoigm[x, is.na(autoigm[x,])] <- medias[x]
}
> n <- ConsensusClusterPlus(autoigm, maxK = 7, reps = 50,
clusterAlg="diana_alg",distance="pearson",seed=856814)


# ESPA IgG
> espaigg <- datos$`ESPA IgG`$single_group
> n <- ConsensusClusterPlus(espaigg, maxK = 7, reps = 50,
clusterAlg="diana_alg",distance="pearson",seed=856814)


# ESPA IgM
> espaigm <- datos$`ESPA IgM`$single_group
> n <- ConsensusClusterPlus(espaigm, maxK = 7, reps = 50,
clusterAlg="diana_alg",distance="pearson",seed=856814)


# Immunocheckpoints
> immuno <- datos$Immunocheckpoints$single_group
> n <- ConsensusClusterPlus(immuno, maxK = 7, reps = 50,
clusterAlg="diana_alg",distance="pearson",seed=856814)


# Proteoma
> proteoma <- t(datos$Proteoma$single_group)
> proteoma <- proteoma[-c(1,27,42,44,49),]
> proteoma <- t(proteoma)
> n <- ConsensusClusterPlus(proteoma, maxK = 7, reps = 50,
clusterAlg="diana_alg",distance="pearson",seed=856814)
```