

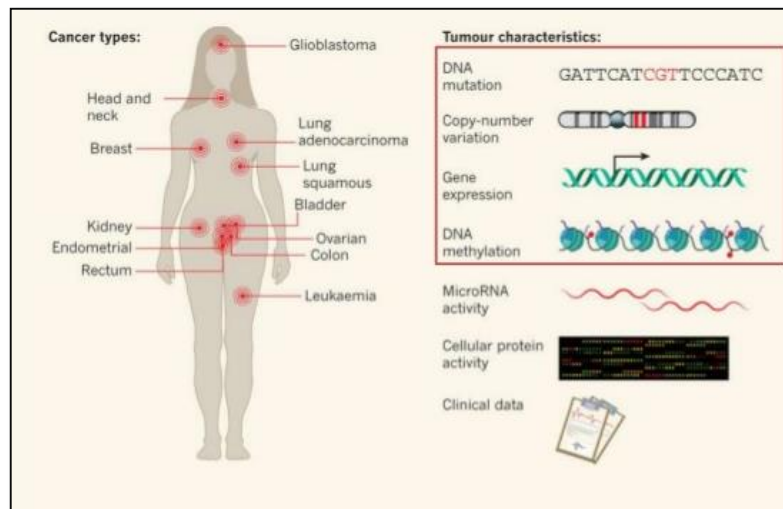
ANÁLISIS DE EXPRESIONES DE GENES DE TUMORES CANCERÍGENOS MEDIANTE TÉCNICAS DE CLUSTERING



José Antonio Pozo Núñez
Técnicas Inteligentes en Bioinformática
Máster Universitario en Lógica, Computación e Inteligencia Artificial

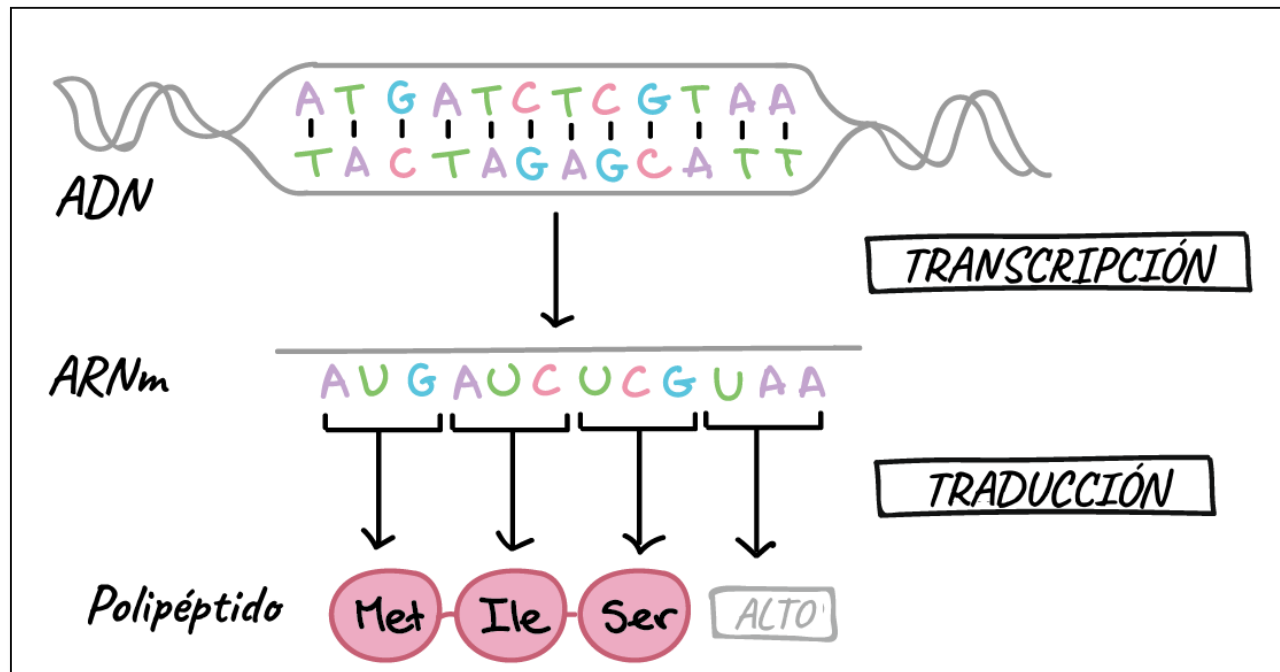
Proyecto Pan-Cáncer

- Enmarcado dentro del proyecto *El Atlas del Genoma del Cáncer*.
- Pretende analizar miles de genomas de pacientes de todo el mundo con diferentes tipos de tumores, con el fin de identificar diferencias y similitudes a nivel genético e identificar terapias más efectivas.



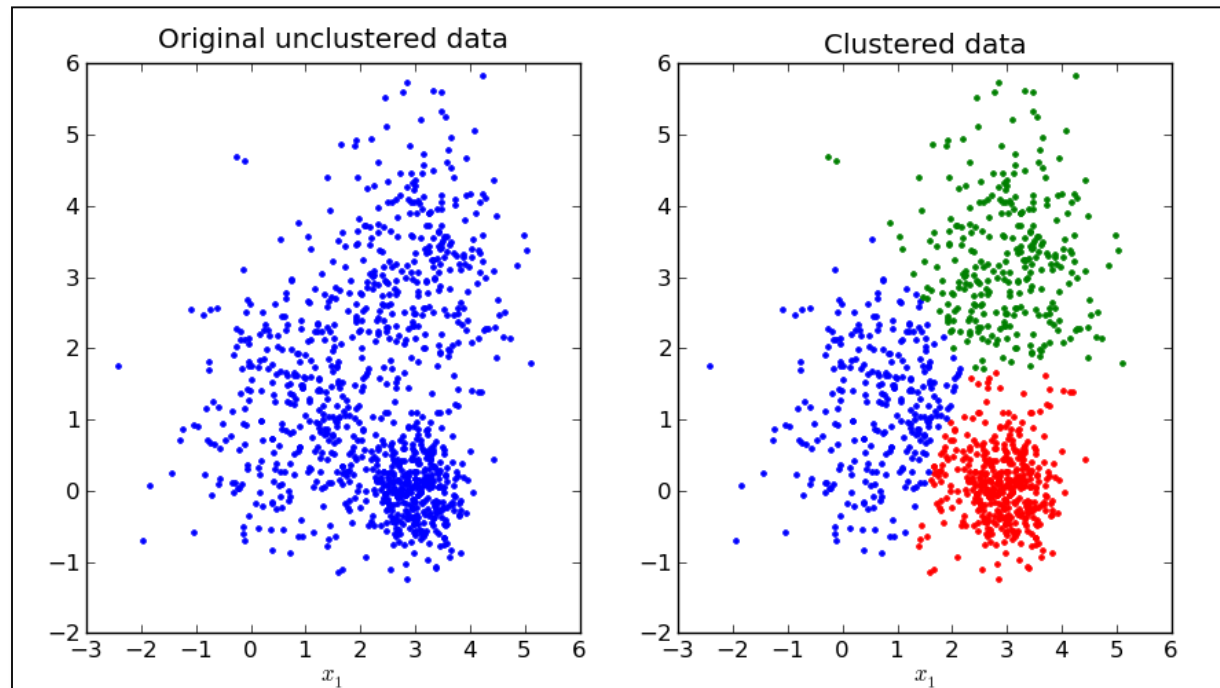
Expresión Génica

- Dogma central de la biología molecular.
- Proceso por el cual las instrucciones genéticas son utilizadas para sintetizar productos del gen.
- Es posible medirlo.



Clustering

- Utilizaremos técnicas Clustering, con R y Weka, para obtener agrupaciones y analizar sus diferencias y similitudes.

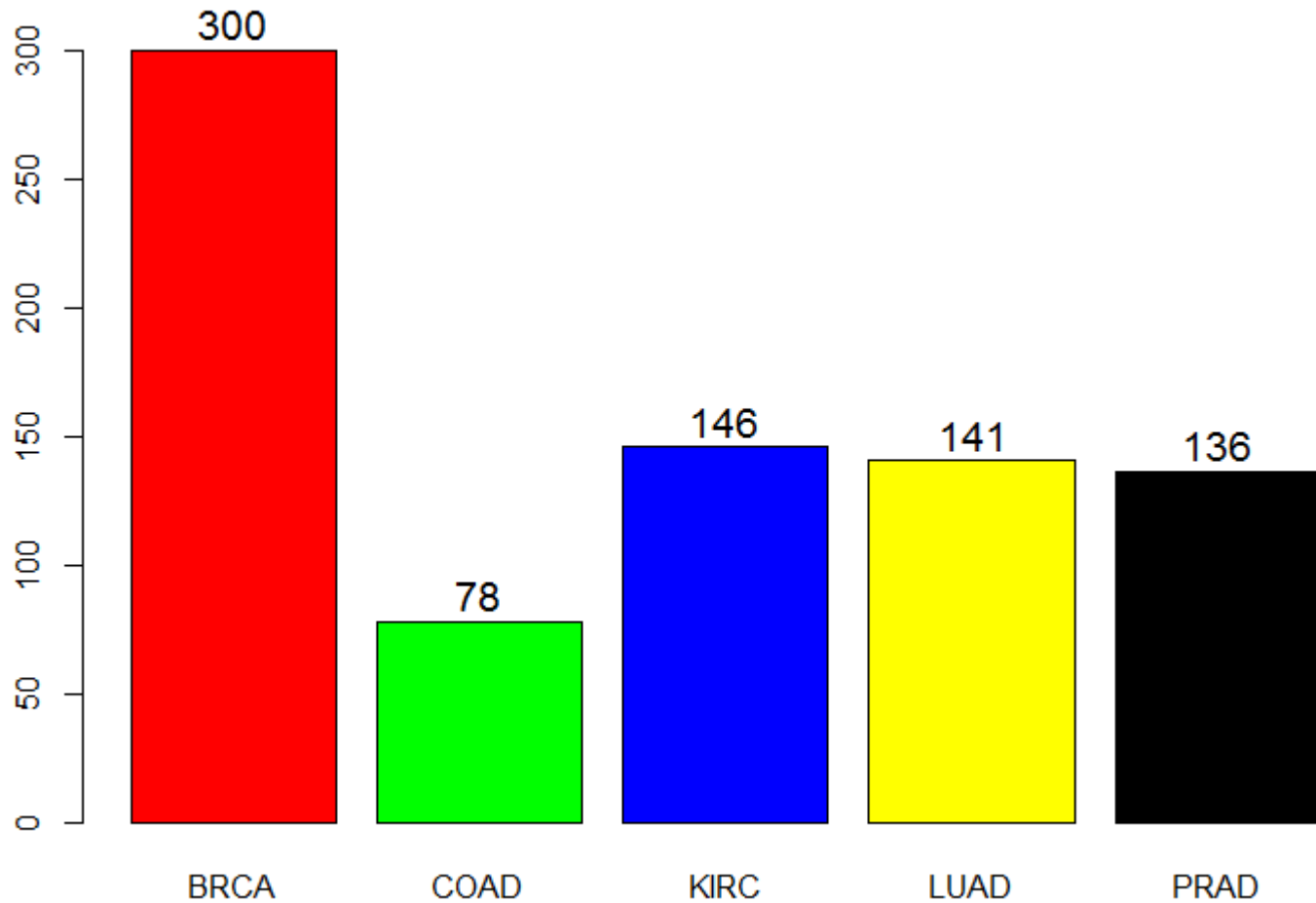


Dataset (1)



- 801 Ejemplos (Distintos pacientes con cáncer).
- 20531 Atributos genéticos (Mediciones de la expresión génica de distintos genes).
- 1 Atributo con la clase de tumor (Usado para validar el modelo):
 - BRCA: Cáncer de mama.
 - COAD: Cáncer de colon.
 - KIRC: Cáncer de células renales.
 - LUAD: Cáncer de pulmón.
 - PRAD: Cáncer de próstata.

Dataset (2)



Preprocesado

- **Comprobación de valores vacíos**

```
> #Comprobamos que no haya valores vacíos, si es así, los informamos con la media de la columna
> genes.cancer.dataset.omit <- na.omit(genes.cancer.dataset)
> porcentaje.valores.vacios <- (1-(nrow(genes.cancer.dataset.omit)/nrow(genes.cancer.dataset)))
> print(paste("Porcentaje de valores vacíos:",porcentaje.valores.vacios))
[1] "Porcentaje de valores vacíos: 0"
```

- **Eliminamos los atributos cuyos valores sean todos cero**

```
> #Eliminamos los Genes (atributos) que tengan todos sus valores a cero
> genes.valores.cero <- obtener_genes_a_cero(genes.cancer.dataset)
> genes.cancer.dataset <- eliminar_atributos(genes.cancer.dataset,genes.valores.cero)
> print(paste("Numero de atributos después de la eliminación:",length(genes.cancer.dataset)))
[1] "Numero de atributos después de la eliminación: 20264"
```

- **Normalización**

```
> #Comprobamos el rango de varios atributo
> apply(genes.cancer.dataset[25:30], 2, range)
      gene_26  gene_27  gene_28  gene_29  gene_30  gene_31
[1,]  0.00000  6.200934  0.00000  2.20389  0.000000  0.000000
[2,] 13.26481 11.537946 10.25141 13.10509  9.962827  9.520505
```

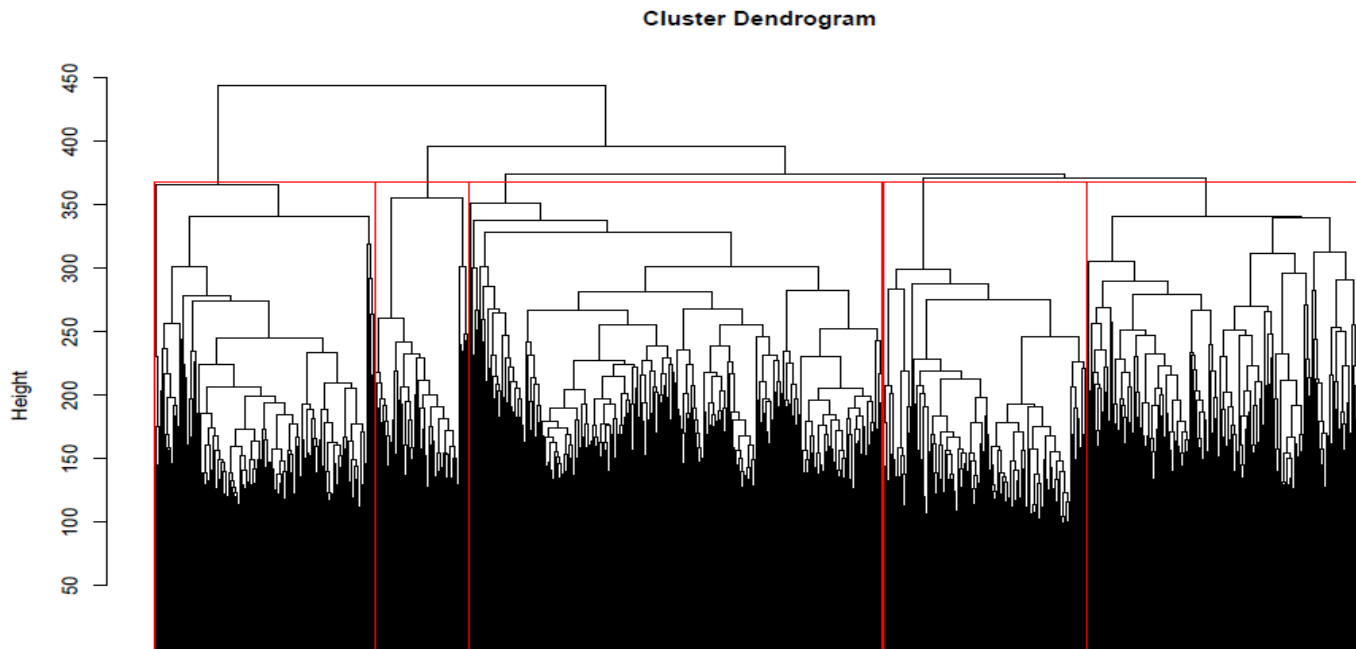
- **Eliminación de outliers**

```
> mvOutlier(genes.cancer.dataset)
Error in covMcd(data, alpha = alpha) : n <= p -- you can't be serious!
```

Construcción del modelo y análisis (1)

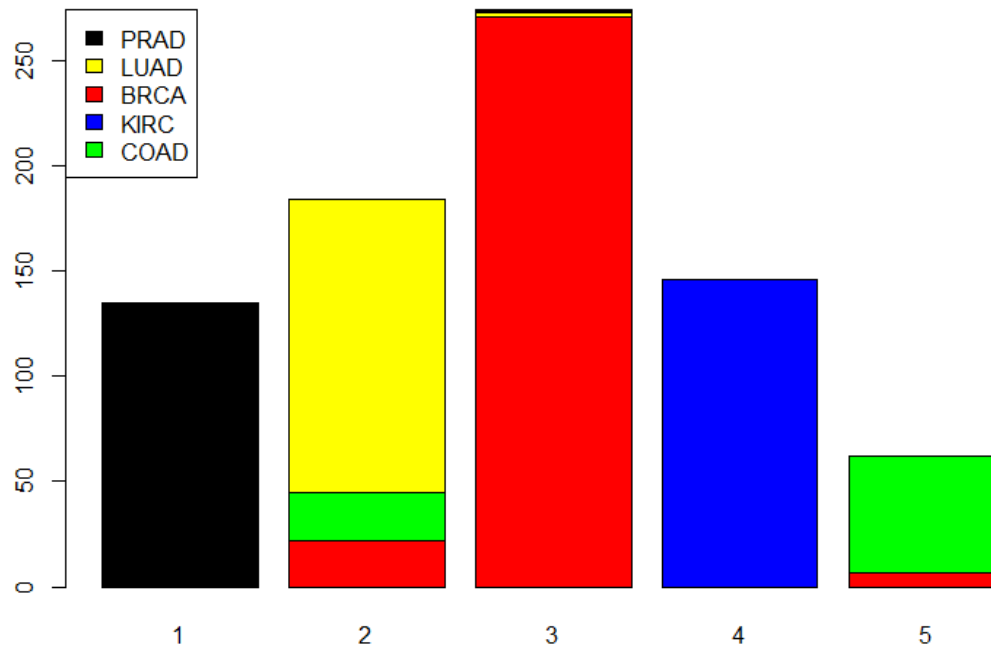
- Algoritmo Jerárquico (distintas configuraciones)

```
> #Calculamos la matriz de distancia, utilizando la distancia euclídea  
> genes.cancer.dist<-dist(genes.cancer.dataset,method = "euclidean")  
> #Construimos el modelo de clustering jerárquico, con distancia completa  
> genes.cancer.modelo.hclust<-hclust(genes.cancer.dist,method = "complete")  
> #Pintamos el árbol jerárquico  
> plot(genes.cancer.modelo.hclust, hang = -1, cex = 0.6)  
> #Dividimos el árbol en 5 clústers, como el número de clases  
> rect.hclust(genes.cancer.modelo.hclust, k = 5, border = "red")
```



Construcción del modelo y análisis (2)

```
> barplot(table(genes.cancer.dataset.groups$Class, genes.cancer.dataset.groups$groups),  
col = c("red", "green", "blue", "yellow", "black"))  
[1] "Porcentaje de elementos bien agrupados con la distancias Euclidea y Completa:  
0.931335830212235"
```

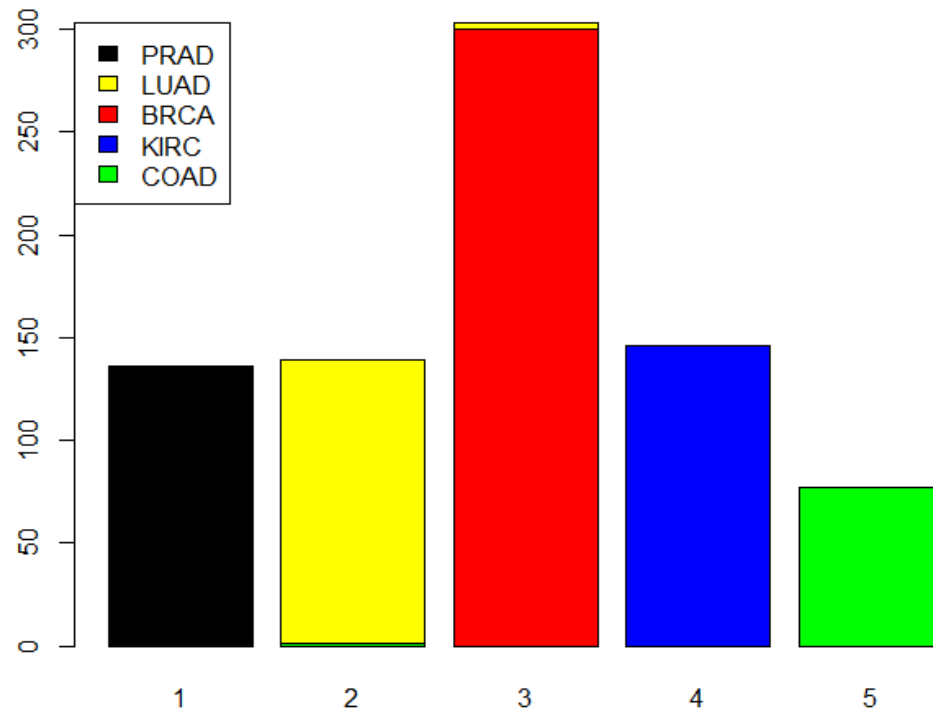


Construcción del modelo y análisis (3)



- Distancia Euclídea y enlace War.D

[1] "Porcentaje de elementos bien agrupados con la distancias Euclidea y ward.D:
0.995006242197253"

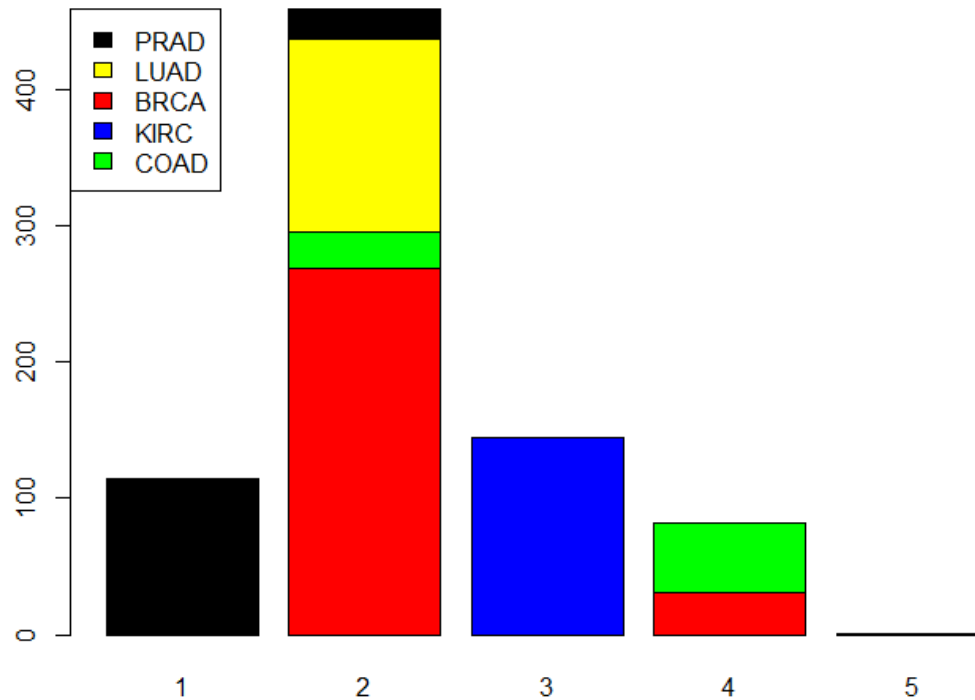


Construcción del modelo y análisis (4)



- Distancia Manhattan y enlace Completo

[1] "Porcentaje de elementos bien agrupados con la distancias Manhattan y Completa: 0.722846441947566"

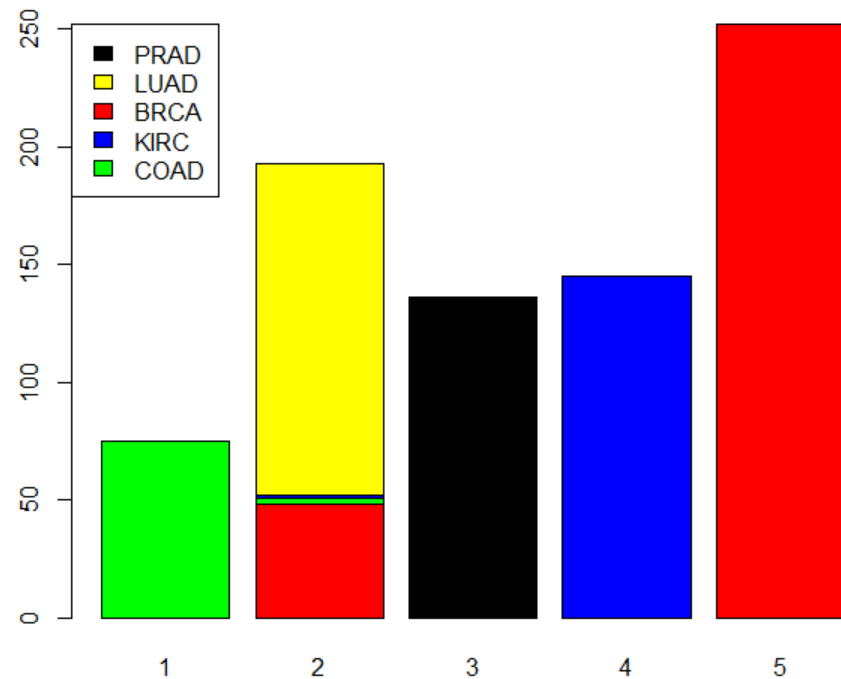


Construcción del modelo y análisis (5)

- Algoritmo K-Medias

```
> #Construimos el modelo kmeans  
> genes.cancer.modelo.kmeans <- kmeans(genes.cancer.dataset,5)  
> table(genes.cancer.modelo.kmeans$cluster, as.vector(t(genes.cancer.dataset.class)))
```

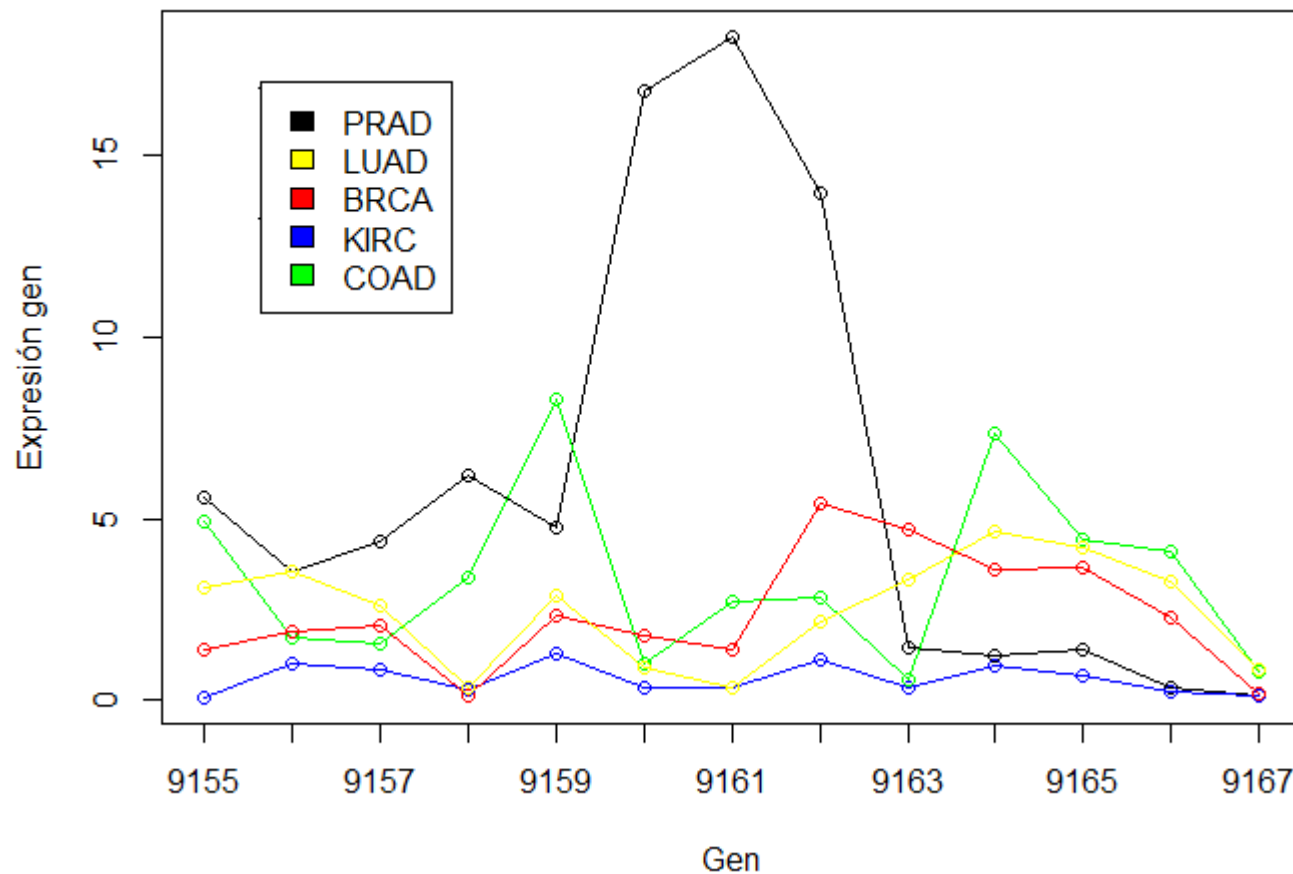
	BRCA	COAD	KIRC	LUAD	PRAD
1	0	75	0	0	0
2	48	3	1	141	0
3	0	0	0	0	136
4	0	0	145	0	0
5	252	0	0	0	0



[1] "Porcentaje de elementos bien agrupados con k-medias: 0.935081148564295"

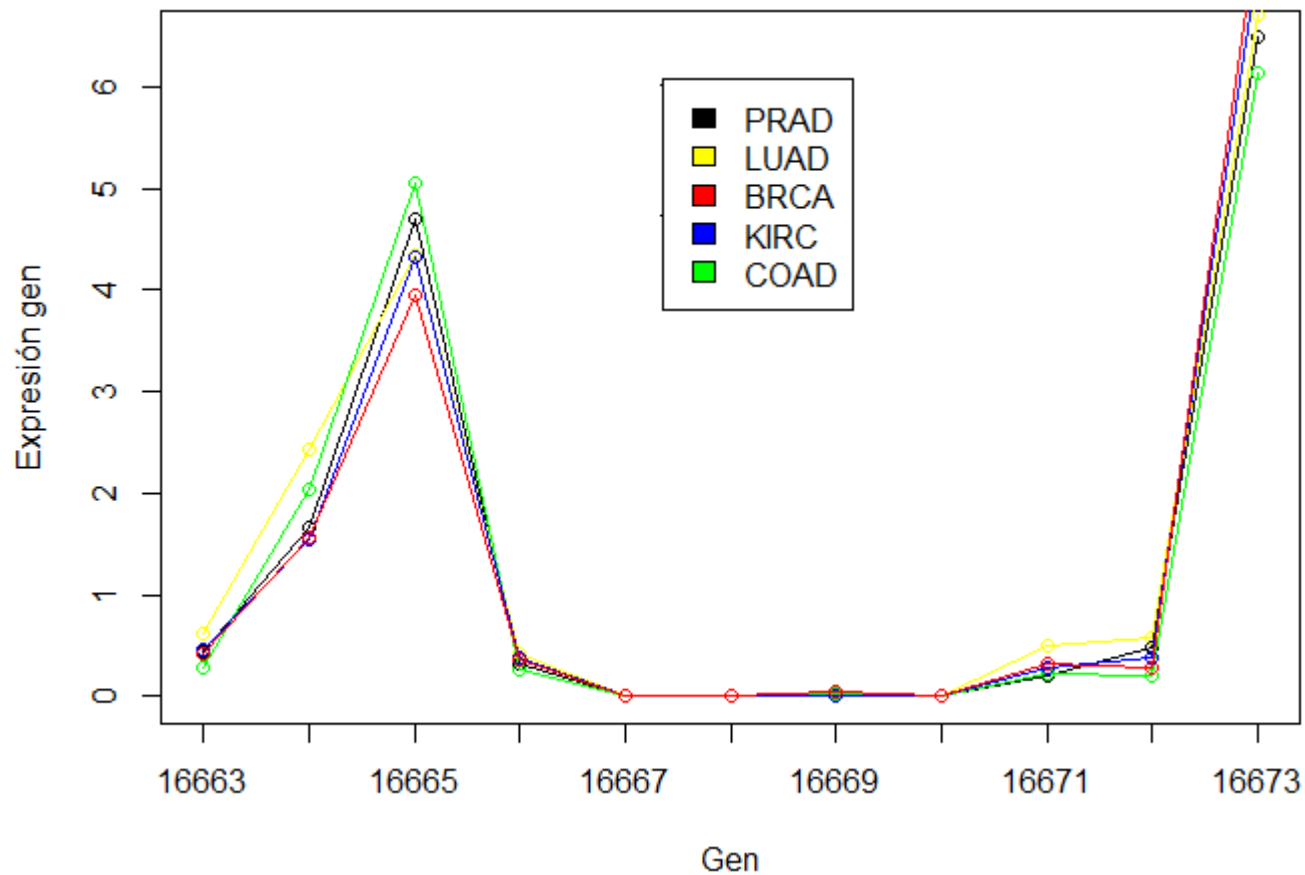
Construcción del modelo y análisis (6)

- Centroides (Genes más dispares)



Construcción del modelo y análisis (6)

- Centroides (Genes más similares)



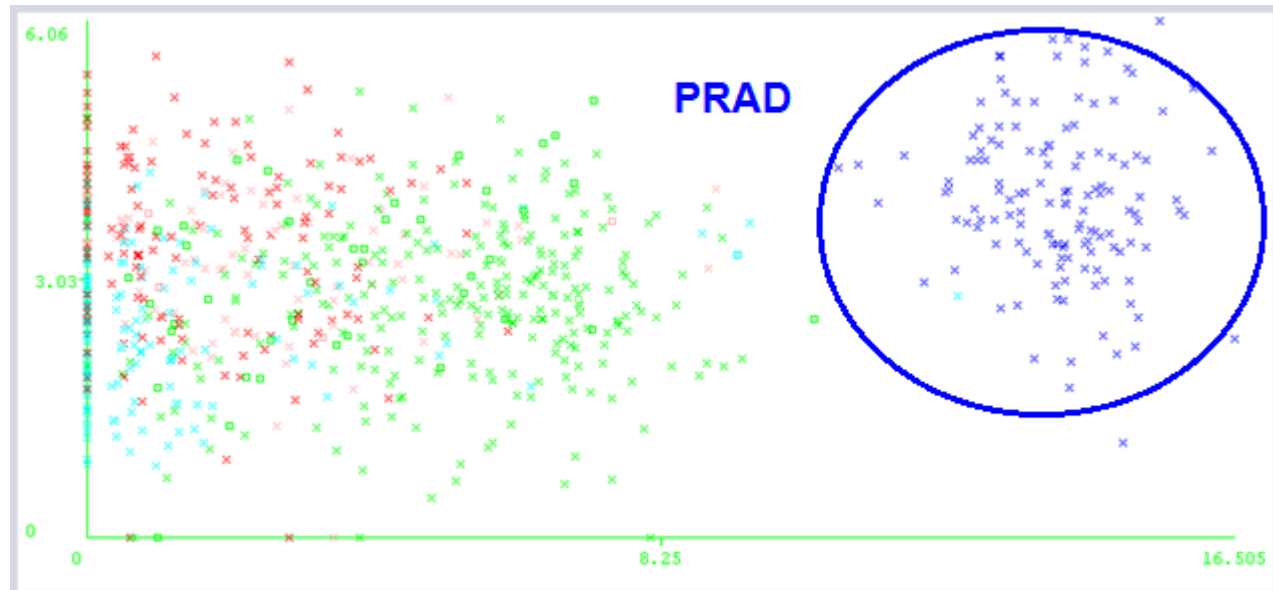
Construcción del modelo y análisis (7)

- Weka (K-Medias)

Classes to Clusters:


	0	1	2	3	4	<-- assigned to cluster
Cluster 0 <-- LUAD	0	0	136	0	0	PRAD
Cluster 1 <-- KIRC	141	0	0	0	0	LUAD
Cluster 2 <-- PRAD	41	0	0	0	259	BRCA
Cluster 3 <-- COAD	1	145	0	0	0	KIRC
Cluster 4 <-- BRCA	4	0	0	74	0	COAD

Incorrectly clustered instances : 46.0 5.7428 %





Conclusiones

- Se hace evidente que existen diferencias y similitudes a nivel génico entre los distintos tipos de tumores.
 - Los tumores BRCA, LUAD y COAD parecen más similares, sin embargo, KIRC y PRAD, parecen diferenciarse de los demás.
 - Información valiosa para los expertos en el dominio del problema.
 - A nivel personal:
 - Complejidad a la hora de tratar con un dataset tan grande.
 - Difícil visualización.
 - Curva de aprendizaje de R.
- 



¡GRACIAS!

**ANÁLISIS DE EXPRESIONES DE
GENES DE TUMORES
CANCERÍGENOS MEDIANTE
TÉCNICAS DE CLUSTERING**



José Antonio Pozo Núñez
Técnicas Inteligentes en Bioinformática
Máster Universitario en Lógica, Computación e Inteligencia Artificial