A3.1 SVM y Multiple Testing

En esta actividad trabajarás con la base de datos de la que se habló en clase, que consiste de 83 muestras y 2308 variables de entrada, que consisten en la expresión génica estandarizada de distintos genes. La variable de salida cuenta con valores numéricos del 1 al 4 que corresponden a distintos tipos de cáncer.

Desarrolla los siguientes puntos en una *Jupyter Notebook*, tratando, dentro de lo posible, que cada punto se trabaje en una celda distinta. Los comentarios en el código siempre son bienvenidos, de preferencia, aprovecha el *markdown* para generar cuadros de descripción que ayuden al lector a comprender el trabajo realizado.

1. Importa los datos a tu ambiente de trabajo y revisa que no haya huecos. Calcula la diferencia de promedios entre las clases 2 y 4 para todos los genes, e imprime los 10 genes con la mayor diferencia de medias. Indica qué crees que esta diferencia podría implicar en términos de un estudio de inferencia.

2. Calcula el estadístico t y el p-value para comparar las medias de todos los genes entre la clase 2 y la clase 4 de la base de datos. Usa la metodología de Bonferroni, de Holm, y de Benjamini-Hochberg para corregir por múltiples pruebas e indica, para cada una, qué genes tienen una expresión significativamente distinta entre las clases (maneja un control de 0.05). Te recomiendo usar la función multipletests de statsmodels.stats.multitest

3. Realiza un experimento similar, pero ahora comparando las medias de las 4 clases de la base de datos. Para lograrlo, en vez de trabajar con el estadístico t, te recomiendo realizar pruebas de análisis de varianza (ANOVA). Dicha prueba la puedes realizar con la función f_oneway de scipy.stats, pero revisa bien cómo se deben ingresar los datos a dicha función, necesitarás primero estratificarlos por clase.

4. Separa los datos en entrenamiento y prueba, construye y entrena un modelo de SVM con un kernel lineal, con un kernel polinomial de orden 3, y con un kernel radial (puedes usar los parámetros que gustes, no necesitas optimizar con validación cruzada). Para evitar que el tiempo de procesamiento sea exagerado, puedes seleccionar solamente algunas variables, partiendo de los resultados que obtuviste en los puntos anteriores. Esta no es una práctica adecuada, pues estamos cayendo en una situación de fuga de datos. Lo ideal sería que la selección de características se basara solamente en experimentos realizados con los datos de entrenamiento. Pero, en este caso, obviaremos este detalle.

5. Calcula, para los 3 modelos, las métricas que consideres importantes para comparar los desempeños. Indica qué opinas sobre los resultados, especificando si crees que uno de los kernels es mejor para esta tarea específica.