

Isabella Salgado 201730418  
Johan Hernández 201729696  
Juan Pablo Naranjo 201730006

## Tarea 7

### Punto 1

a.

En este primer punto, se obtuvieron los datos presentados en el archivo *datosIris.txt*, que corresponden a 150 observaciones del largo y ancho de los sépalos de flores tipo Iris. En la figura 1 se puede observar una representación gráfica de estos datos.

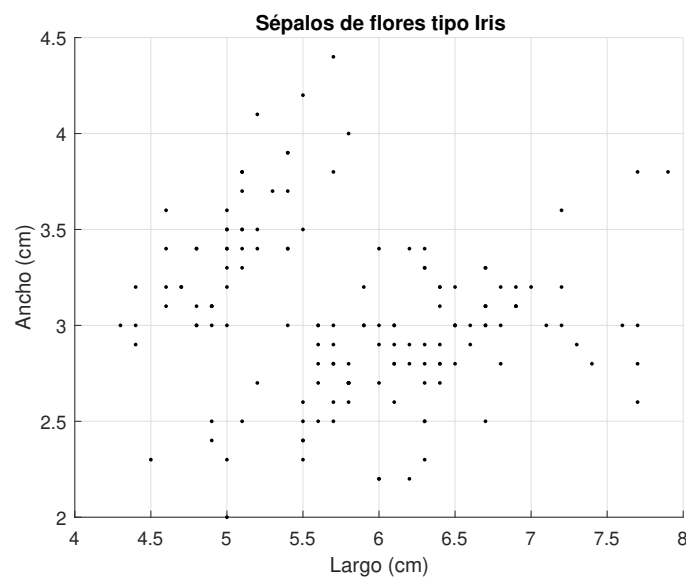


Figura 1: Largo y ancho de los sépalos de flores tipo Iris

Se puede notar que la mayoría de las observaciones del largo y ancho de los sépalos están distribuidos alrededor de un largo entre los 5.5 cm y los 7 cm para un alto entre los 2 cm y los 3.5 cm. Por otro lado, también se puede notar una pequeña concentración de datos para largos entre los 4 cm y los 6 cm y un ancho entre los 3 cm y los 4.5 cm.

b.

Se implementó el algoritmo EM para estimar la función de densidad de probabilidad que genera los datos presentados en el literal anterior como una combinación de tres distribuciones Gaussianas, es decir, para obtener una distribución de densidad de probabilidad dada por

$$p_1(x) = \sum_{k=1}^3 \alpha_k N(x; \mu_k, Q_k) \quad (1)$$

donde  $\mu_k$  corresponde al vector de medias,  $Q_k$  corresponde a la matriz de covarianza de cada distribución independiente y  $\alpha_k$  corresponde al peso correspondiente de cada función independiente.

El resultado obtenido al implementar el algoritmo y traslapar las curvas de nivel de la función combinada obtenida sobre la distribución de los datos se muestra en la figura 2.

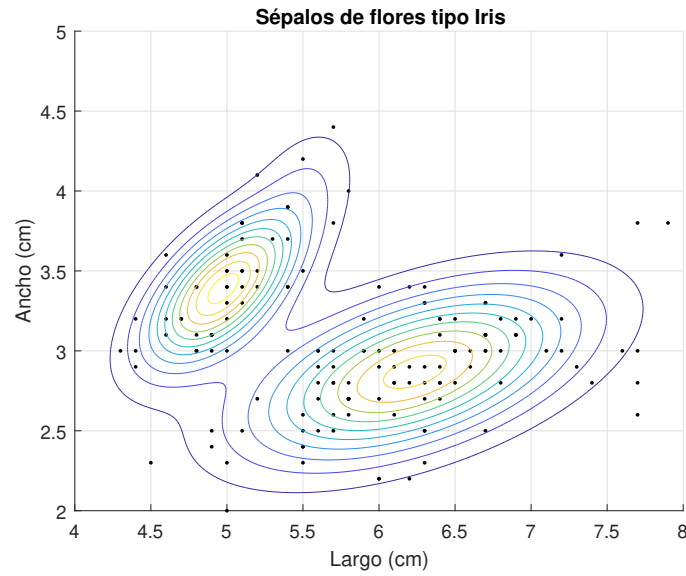


Figura 2: Curvas de nivel de la función  $p_1(x)$  (combinación de 3 distribuciones Gaussianas)

Esta función es el resultado de combinar tres distribuciones Gaussianas, cuyas curvas de nivel independientes se muestran en la figura 3.

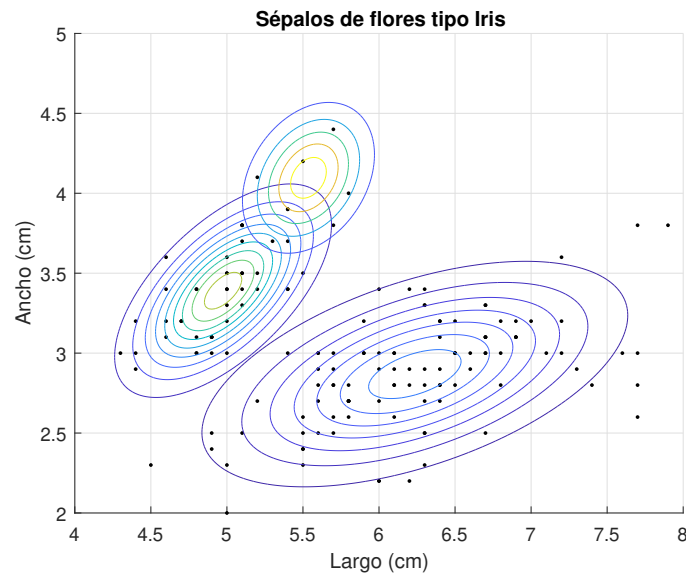


Figura 3: Curvas de nivel de las funciones que sumadas generan la función  $p_1(x)$

### C.

Finalmente, se volvió a aplicar el algoritmo EM sobre los datos pero ahora para generar una función  $p_2(x)$ , que corresponde a una combinación de cuatro distribuciones Gaussianas, como se muestra en la ecuación 2:

$$p_2(x) = \sum_{k=1}^4 \alpha_k N(x; \mu_k, Q_k) \quad (2)$$

donde  $\mu_k$  corresponde al vector de medias,  $Q_k$  corresponde a la matriz de covarianza de cada distribución independiente y  $\alpha_k$  corresponde al peso correspondiente de cada función independiente.

En la figura 4 se pueden observar las curvas de nivel de la función  $p_2(x)$  sobre la distribución de los datos.

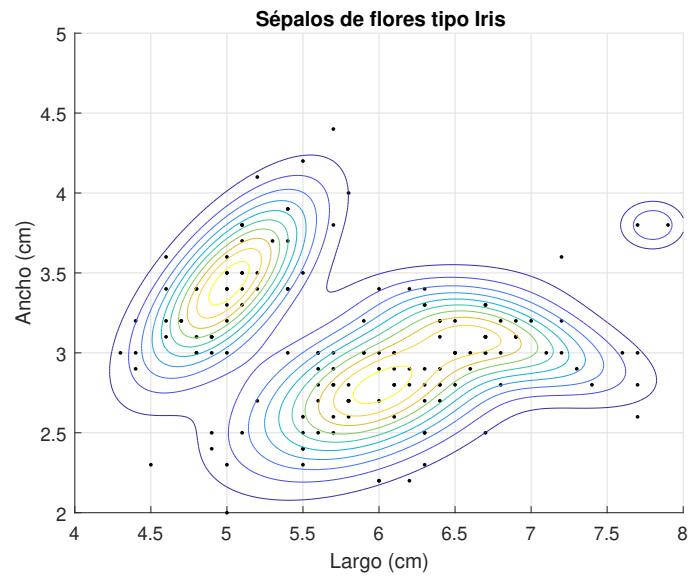


Figura 4: Curvas de nivel de la función  $p_2(x)$  (combinación de 4 distribuciones Gaussianas)

Finalmente, en la figura 5 se pueden observar las curvas de nivel correspondientes a cada una de las funciones independientes que conforman a la función  $p_2(x)$ .

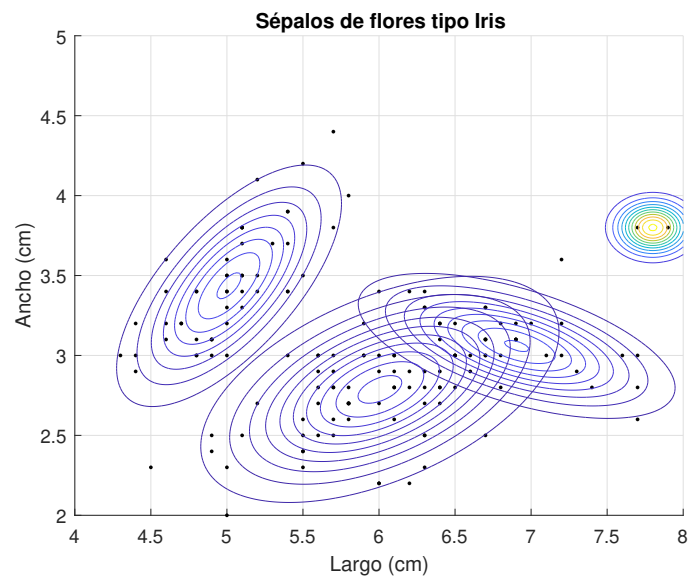


Figura 5: Curvas de nivel de las funciones que sumadas generan la función  $p_2(x)$

## Punto 2

a.

Para realizar este literal el primer paso fue dividir y organizar todos los datos en seis vectores diferentes, uno por cada expresión facial, de esta forma se obtuvieron los vectores *neutral*, *feliz*, *triste*, *furioso*, *disgusto* y *miedo*, cabe mencionar que estos vectores no quedaron del mismo tamaño, esto indica que en la base de datos proporcionada no se encontraban la misma cantidad de datos para todas las expresiones faciales.

El siguiente paso fue escoger aleatoriamente el 70 % de los datos de cada vector, mencionados anteriormente, y guardarlos en un nuevo vector de entrenamiento, asimismo, el 30 % restante se dividió aleatoriamente en dos partes y se guardaron en dos vectores nuevos de validación y prueba. Con todo esto se logró que los grupos de entrenamiento, validación y prueba tuvieran sus datos balanceados, en otras palabras, que tuvieran aproximadamente la misma cantidad de datos de cada una de las expresiones faciales.

b.

Usando solo los datos de entrenamiento se obtuvo la media de procrustes de estos datos la cual se puede observar en la figura 6.

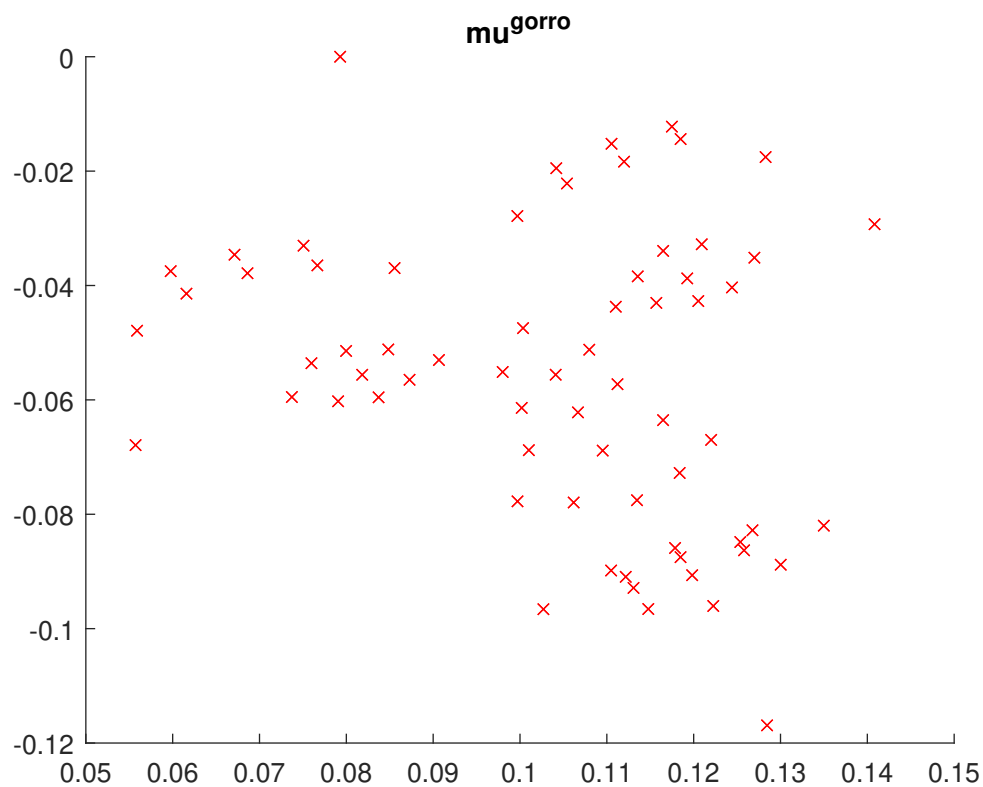


Figura 6: Media de procrustes

Después de obtener la media de procrustes se proyectó cada uno de los landmarks de los distintos grupos a esta media, de esta forma se obtuvieron los nuevos landmarks mostrados en la figura 7.

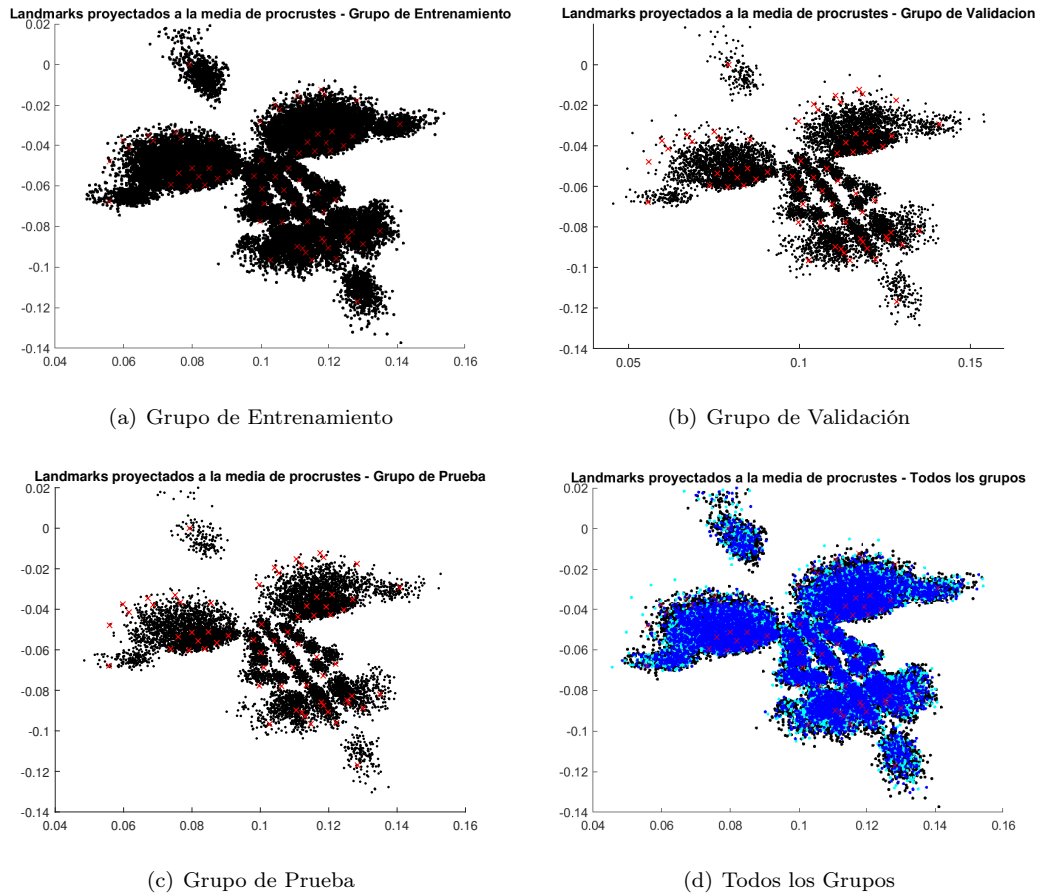


Figura 7: Landmarks proyectados a la media de procrustes

Observando la figura 7 se puede notar que, además de que quedaron bien proyectados a la media de procrustes, los seis tipos de expresiones faciales quedaron bien distribuidos entre los tres grupos, entrenamiento, validación y prueba, debido a que los datos tienen las mismas dimensiones y tienen una estructura bastante similar.

### c.

Por medio de un recorrido de tipo for se generaron los vectores de características de cada uno de los tres grupos de datos, sin embargo, en posteriores literales del presente trabajo se deberá calcular el vector de media y la matriz de covarianza de cada una de las expresiones faciales, por este motivo, además de dividir los vectores de características entre los tres grupos, entrenamiento, validación y prueba, también se separaron de acuerdo a su expresión facial, con esto se obtuvieron 18 vectores de características en total, seis por cada grupo.

d.

Usando únicamente los datos de entrenamiento se estimó el vector de medias y la matriz de covarianza de cada una de las expresiones faciales. El vector de media y la matriz de covarianza se estimaron usando las ecuaciones 3 y 4, respectivamente, donde el subíndice  $c$  corresponde a cada una de las distintas clases,  $m_c$  corresponde al número de datos de la clase  $c$  y  $x_i$  corresponde a un vector de características.

$$\mu_c = \frac{1}{m_c} \cdot \sum_{i=1}^{m_c} x_i \quad (3)$$

$$C_c = \frac{1}{m_c} \cdot \sum_{i=1}^{m_c} (x_i - \mu_c)(x_i - \mu_c)^T \quad (4)$$

Para este punto no se realizó regularización a ninguna matriz de covarianza, debido a que en el literal e. del presente trabajo se deben añadir distintas regularizaciones y determinar con cuál valor de regularización se obtienen menos errores.

Seguidamente, se creó una función en *Matlab* llamada  $pdf_c$  que recibía como parámetro un vector de características, un vector de medias y una matriz de covarianza y retorna el valor del vector de características evaluada en una distribución normal multivariable con los parámetros del vector de medias y la matriz de covarianza suministrados. La ecuación 5 fue la función que se empleó para determinar el valor mencionado anteriormente, donde el subíndice  $c$  corresponde a cada una de las distintas expresiones faciales y  $X$  corresponde al vector de características que se está evaluando.

$$valor = \frac{e^{(-\frac{1}{2} \cdot (X - \mu_c)^T \text{inv}(Q_c) (X - \mu_c))}}{\sqrt{\det(Q_c)} \cdot (2\pi)^{n/2}} \quad (5)$$

e.

Por medio de la función creada en el literal d. del presente trabajo, se tomaron los datos del conjunto *validación* y se empezaron a evaluar, en la función mencionada anteriormente, con la diferencia de que a cada matriz de covarianza se le agregaba una regularización de la forma  $Q_c = C_c + \lambda I$ , donde  $C_c$  corresponde a la matriz de covarianza de cada uno de las distintas expresiones faciales,  $I$  es una matriz identidad con las mismas dimensiones de  $C_c$  y  $\lambda$  corresponde a un valor entre 1 y 50, i, multiplicado por 0.001, es decir,  $\lambda = 0,001 \cdot i$ , con  $i = [1, 50]$ . En la figura 8 se puede observar el número de errores totales contra el valor de  $i$ , que como se acabó de mostrar es directamente proporcional con el valor de  $\lambda$ .

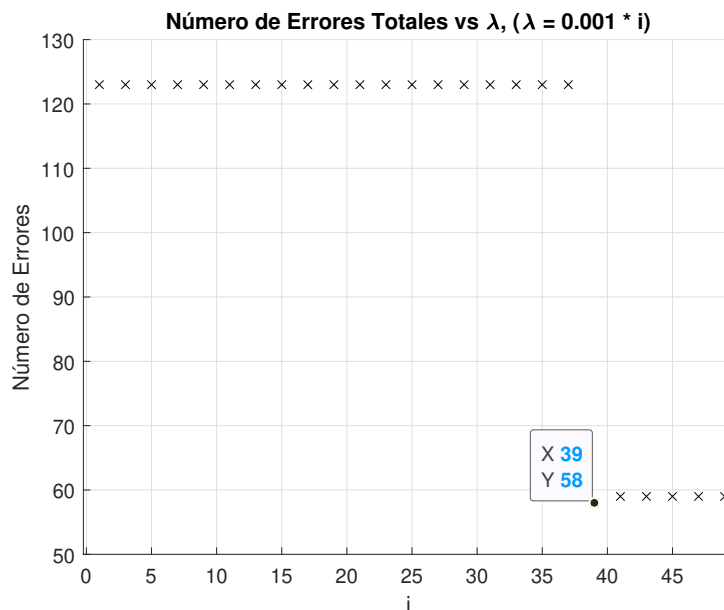


Figura 8: Número de Errores Totales vs  $\lambda$

En la figura 8 se puede observar que el valor de  $i$  que generó menos errores corresponde a  $i = 39$ , que indica un valor de  $\lambda = 0,039$ . Por otro lado, se puede notar que con un  $\lambda$  menor a 0,038 se genera un error constante de 123 valores, esto se debe a que las matrices de covarianza son cercanas a ser singulares y la regularización no es lo suficientemente alta para lograr corregir este problema, sin embargo, con un  $\lambda$  mayor o igual a 0,038 se disminuye sustancialmente el número de errores y se observa una tendencia casi lineal.

f.

Con el valor de  $\lambda$  obtenido anteriormente se llevó a cabo la clasificación de los datos del conjunto *prueba* obteniendo los siguientes resultados, aunque cabe aclarar que estos resultados varían cada vez que se ejecuta todo el código desde cero, esto es debido a que los datos de cada uno de los conjuntos se escogen de manera aleatoria:

- Número total de datos: 148
- Número total de clasificaciones correctas: 101
- Número total de clasificaciones erróneas: 47
- Porcentaje de error total: 31,76 %

Individualmente se obtuvieron los siguientes errores:

- Porcentaje de error expresión neutral: 28,00 %
- Porcentaje de error expresión feliz: 19,23 %
- Porcentaje de error expresión triste: 52,17 %
- Porcentaje de error expresión furioso: 20,83 %
- Porcentaje de error expresión disgusto: 50,00 %
- Porcentaje de error expresión miedo: 23,08 %

En la tabla 1 se puede observar la matriz de confusión obtenida.

Tabla 1: Matriz de Confusión

Clase	Neutral	Feliz	Triste	Furioso	Disgusto	Miedo
Neutral	18	0	4	0	1	2
Feliz	0	21	0	0	4	1
Triste	6	0	11	1	3	2
Furioso	3	0	1	19	1	0
Disgusto	3	0	1	7	12	1
Miedo	0	0	6	0	0	20

En la tabla 1 se puede observar que para cada fila la mayor concentración de datos se encuentra en la diagonal de la matriz esto indica que el algoritmo si logra determinar cual es la expresión facial correcta en la mayoría de los casos, sin embargo, hay expresiones faciales que se lograron confundir, por ejemplo, la expresión facial triste se confundió, en varias ocasiones, con la expresión neutral, otro caso puede ser el de la expresión facial de disgusto que se confundió, también en varias ocasiones, con la expresión furioso. Estas confusiones son debidas a que algunas expresiones son difíciles de interpretar hasta para una persona, por ejemplo, en la figura 9 se pueden observar tres caras que son catalogadas como expresiones tristes, pero, al verlas se puede notar que no es fácil clasificarlas como expresiones tristes, podrían ser caras de disgusto, por ejemplo. Por otro lado, en la figura 10 se pueden observar tres caras que son etiquetadas como caras de disgusto, sin embargo, observándolas es difícil clasificarlas como expresiones de disgusto dado que una persona podría clasificarlas como expresiones de furia. Estos son solo algunos ejemplos, pero, realmente el problema de los clasificadores de expresiones faciales es que son muy sensibles a los cambios, es decir, una persona que tenga las cejas muy pobladas podría estar

presentando una expresión neutral y aún así ser clasificada como una expresión de furia, por otro lado, no todas las personas tienen exactamente el mismo gesto para una expresión facial dada.

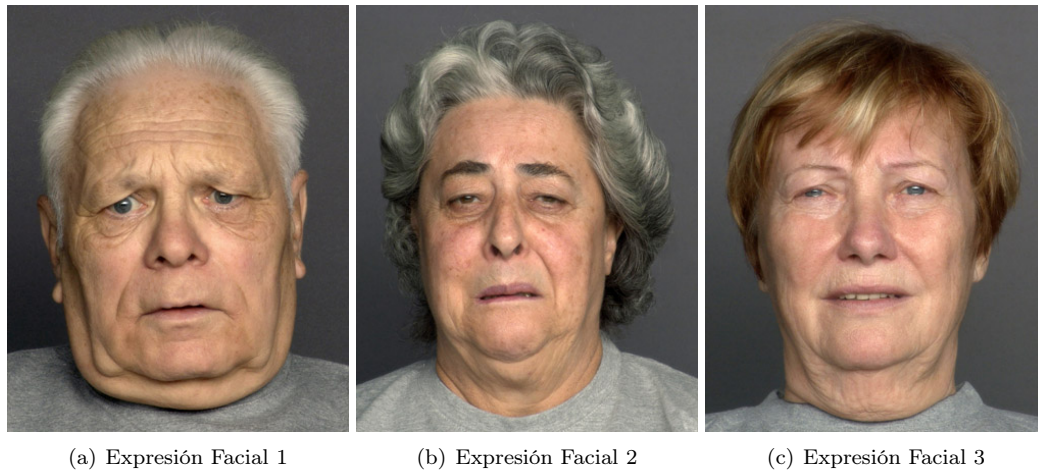


Figura 9: Expresiones Faciales Tristes

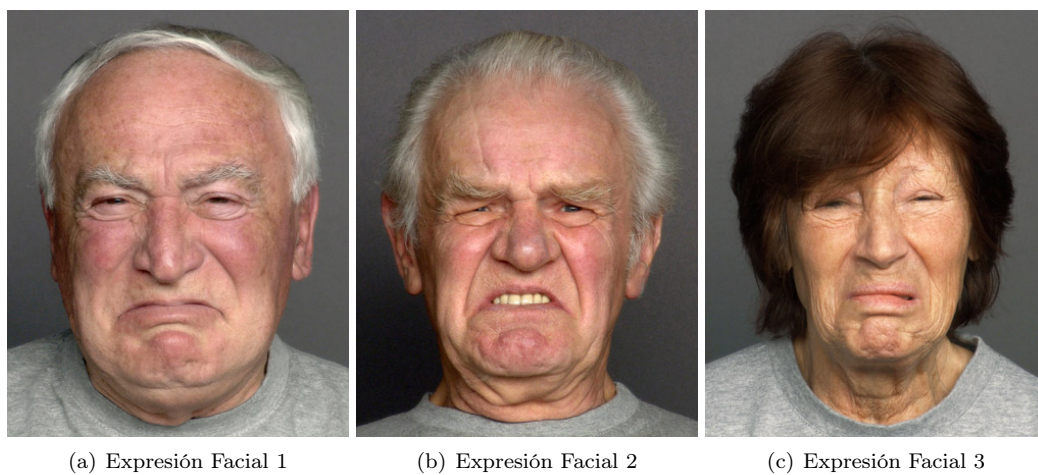


Figura 10: Expresiones Faciales de Disgusto