



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Facultad de Ciencias Naturales y Matemáticas
Ingeniería en Estadística Informática

“RECONOCIMIENTO FACIAL USANDO MODELOS MIXTOS”

T E S I S D E G R A D O

Previo a la obtención del Título de:

INGENIERO EN ESTADÍSTICA INFORMÁTICA

Presentado por:

Kevin Santiago Flores Zambrano

Guayaquil – Ecuador

2014

AGRADECIMIENTO

Agradezco a Dios, al cosmos por permitirme culminar una etapa más de mi vida; a mis padres Lourdes Zambrano Vera y Santiago Flores Game por brindarme su apoyo para seguir adelante; a mis hermanas Carla, Ginger y Pamela Flores; a mi director de tesis PhD. Francisco Vera Alcívar y a mis amigos y colegas estadísticos Rebeca López y Marlon Elizalde por ser sujetos de estudio en este trabajo.

Kevin Santiago Flores Zambrano

DEDICATORIA

A mis padres, hermanas, a aquellos amigos que estuvieron siempre apoyándome y a la Escuela Superior Politécnica del Litoral por permitir formarme en tan grande institución.

Kevin Santiago Flores Zambrano

TRIBUNAL DE SUSTENTACIÓN

Francisco Vera Alcívar, Ph.D.

DIRECTOR DE TESIS

Vanessa Salazar, Máster

PRESIDENTE

Elkin Angulo Ramírez, MIM

VOCAL

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de este Trabajo Final de Graduación, me corresponde exclusivamente; y el patrimonio intelectual del mismo a la Escuela Superior Politécnica del Litoral"

Kevin Santiago Flores Zambrano

RESUMEN

Este trabajo tiene como objetivo aplicar un modelo estadístico en reconocimiento facial entre dos individuos y evaluar su efectividad. También se quiere mostrar cómo la inclusión de efectos aleatorios disminuye la variabilidad del error de un modelo, debido a que la variabilidad total estará contenida mayormente en otras fuentes de variación que tienen una interpretación más física (como rotaciones o traslaciones en nuestro caso de reconocimiento facial) que el error aleatorio (perturbaciones). Estas nuevas fuentes de variación son justamente los efectos aleatorios. Este trabajo está formado por una introducción, cuatro capítulos y un apartado de anexos.

En la Introducción de este trabajo se menciona la evolución del reconocimiento facial en la historia y algunos de los métodos más usados. También se hablará brevemente de los modelos lineales mixtos y cómo podrían aplicarse en reconocimiento facial. En el Capítulo I se describirá el Marco teórico, donde se repasarán brevemente los conocimientos previos que se requieren para poder comprender completamente este trabajo. Se señala cuáles serán los individuos de estudio y la manera en que sus fotografías se convertirán en datos para el modelo. Se indica también la metodología y los pasos a seguir para hacer reconocimiento facial con estas fotos. Al final se hace el planteamiento del problema, se formula el objetivo general y los objetivos específicos de este trabajo y las hipótesis de hallazgos que se esperan.

El Capítulo II contiene el desarrollo del modelo que se aplicará, concretamente más adelante se verá que se trata del modelo de Procrustes de efectos aleatorios. Aquí se desarrollará este modelo paso por paso desde su definición hasta su aplicación en nuestro problema. Se incluyen todas las expresiones matemáticas necesarias. Las demostraciones que sean muy extensas estarán incluidas al final en la sección de Anexos. Se muestra al final cómo la prueba de pertenencia desarrollado previamente en el marco teórico es aplicado al modelo de Procrustes y es lo que permite realizar reconocimiento facial entre dos individuos.

En el Capítulo III se incluye el desarrollo computacional del modelo en el software estadístico R. Aquí se encuentran los resultados de la consola del software en cada paso así como su respectiva interpretación. Se estudian varios enfoques, que dependen tanto del número de puntos de referencia seleccionados como del uso de registración de imágenes en los individuos o no. No obstante, no se incluye el script que incluye la programación completa el cual se encuentra en los Anexos al final del documento.

El Capítulo IV contiene la evaluación del desempeño (efectividad) del modelo de Procrustes de efectos aleatorios en reconocimiento facial. Se incluye al inicio la deducción intuitiva de un indicador formulado para este efecto. Se comparará este indicador entre todos los enfoques realizados al ajustar el modelo, comparando la efectividad del mismo para distintas cantidades de puntos de referencia y uso o no de registración de imágenes.

Al final del presente trabajo se presentan las Conclusiones y Recomendaciones. Las conclusiones son realizadas a partir de los objetivos específicos e hipótesis de hallazgos y las recomendaciones son sugerencias que podrían mejorar el desempeño del modelo de Procrustes de efectos aleatorios en reconocimiento facial, que son surgidas en alguna instancia de este trabajo. Finalmente, está la sección de Anexos, que como se dijo comprenden las demostraciones y programación en el software R, así como también las imágenes de los individuos de estudio de este trabajo.

ÍNDICE GENERAL

AGRADECIMIENTO	ii
DEDICATORIA	iii
TRIBUNAL DE SUSTENTACIÓN	iv
DECLARACIÓN EXPRESA	v
RESUMEN.....	vi
ÍNDICE DE ILUSTRACIONES.....	xiii
ÍNDICE DE GRÁFICOS	xiv
ÍNDICE DE TABLAS	xvii
INTRODUCCIÓN	xix
CAPÍTULO 1	2
1. Marco teórico, metodología, fuente de datos y objetivos	2
1.1 Introducción	2
1.2 Marco teórico	3
1.2.1 El producto Kronecker	3
1.2.2 La Distribución Normal Multivariada.....	4
1.2.3 La prueba de pertenencia	8
1.2.4 El modelo lineal mixto	10
1.2.5 Mínimos cuadrados para la varianza (VLS).....	16

1.2.6	Registración de imágenes.....	17
1.3	Fuentes de datos y metodología	21
1.4	Planteamiento del problema.....	23
1.5	Objetivos	24
1.5.1	Objetivo General	24
1.5.2	Objetivos Específicos.....	24
1.6	Hipótesis de Hallazgos	24
CAPÍTULO 2		26
2.	El Modelo de Procrustes de efectos aleatorios.....	26
2.1	Introducción	26
2.2	El modelo de Procrustes de efectos aleatorios	27
2.3	Método VLS para el modelo de Procrustes.....	31
2.4	Estimación de la varianza de los efectos aleatorios por máxima verosimilitud	33
2.5	Reconocimiento facial con la prueba de pertenencia.....	36
CAPÍTULO 3		39
3.	Programación del modelo de Procrustes de efectos aleatorios	39
3.1	Introducción	39
3.2	Conversión de las imágenes en datos.....	39
3.3	Ajuste del modelo de Procrustes con tres puntos de referencia.....	42

3.4	Uso de la prueba de pertenencia y reconocimiento facial.....	46
3.5	Aplicación del modelo Procrustes en el segundo individuo	47
3.6	El modelo de Procrustes con cinco puntos de referencia.....	50
3.6.1	Ajustando el modelo en el individuo 1	53
3.6.2	Ajustando el modelo en el individuo 2	55
3.7	El modelo de Procrustes con siete puntos de referencia.	58
3.7.1	Ajustando el modelo en el individuo 1	61
3.7.2	Ajustando el modelo en el individuo 2	63
3.8	El modelo de Procrustes con registraci3n de im3genes	65
3.8.1	El modelo Procrustes con tres puntos de referencia y registraci3n de im3genes.....	69
3.8.2	El modelo Procrustes con cinco puntos de referencia y registraci3n de im3genes.....	75
3.8.3	El modelo Procrustes con siete puntos de referencia y registraci3n de im3genes.....	81
CAPÍTULO 4		88
4.	Evaluaci3n del modelo de Procrustes	88
4.1	Introducci3n	88
4.2	Evaluaci3n del modelo con tres puntos de referencia.....	90

4.2.1 Sin registraci3n de im3genes.....	91
4.2.2 Con registraci3n de im3genes	93
4.3 Evaluaci3n del modelo con cinco puntos de referencia.....	94
4.3.1 Sin registraci3n de im3genes.....	94
4.3.2 Con registraci3n de im3genes	95
4.4 Evaluaci3n del modelo con siete puntos de referencia	96
4.4.1 Sin registraci3n de im3genes.....	97
4.4.2 Con registraci3n de im3genes	98
4.5 Evoluci3n de la efectividad del modelo.....	99
5. CONCLUSIONES	101
6. RECOMENDACIONES	104
ANEXOS	106
I. Demostraci3n de la varianza del vector u_i	107
II. Demostraci3n del estimador VLS para el modelo de Procrustes.	108
III. Script con la codificaci3n en R	117
IV. Im3genes utilizadas.....	126
Referencias Bibliogr3ficas	132

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Distribución normal estándar	5
Ilustración 2. Registración de imágenes.....	19
Ilustración 3. Imagen rotada en distintos ángulos	20
Ilustración 4. Algunas de las imágenes que son parte de la fuente de datos.....	22
Ilustración 5. Normalización de las imágenes para el modelo Procrustes: Ejemplo con tres puntos de referencia.....	26
Ilustración 6. Uso de la función locator para señalar un punto en la imagen	40
Ilustración 7. Los cinco puntos de referencia para el modelo Procrustes	51
Ilustración 8. Los siete puntos de referencia para el modelo de Procrustes.....	59
Ilustración 9. Componentes RGB de una imagen del individuo uno.....	66
Ilustración 10. Registración de imágenes en el individuo 1.....	68

ÍNDICE DE GRÁFICOS

Gráfico 1. Gráfico de tres puntos de referencia correspondiente al individuo uno.....	41
Gráfico 2. Gráfico de tres puntos de referencia correspondiente al individuo dos	42
Gráfico 3. Estimación del polígono promedio para el individuo uno	43
Gráfico 4. Estimación del polígono promedio para el individuo dos	48
Gráfico 5. Gráfico de puntos correspondiente al individuo uno con cinco puntos de referencia.....	52
Gráfico 6. Gráfico de puntos correspondiente al individuo dos con cinco puntos de referencia.....	52
Gráfico 7. Estimación del polígono medio para el individuo uno con cinco puntos de referencia.....	53
Gráfico 8. Estimación del polígono promedio desconocido para el individuo dos con cinco puntos de referencia.....	56
Gráfico 9. Gráfico de puntos correspondiente al individuo uno con siete puntos de referencia.....	60
Gráfico 10. Gráfico de puntos correspondiente al individuo dos con siete puntos de referencia.....	60
Gráfico 11. Estimación del polígono medio desconocido para el individuo uno con siete puntos de referencia	61
Gráfico 12. Estimación del polígono medio desconocido para el individuo dos con siete puntos de referencia	63

Gráfico 13. Gráfico con tres puntos correspondiente al individuo uno con registraci3n	70
Gráfico 14. Gráfico con tres puntos correspondiente al individuo dos con registraci3n	70
Gráfico 15. Estimaci3n del polígono promedio para el individuo 1 con tres puntos y registraci3n	71
Gráfico 16. Estimaci3n del polígono promedio para el individuo dos con tres puntos y registraci3n	73
Gráfico 17. Gráfico con cinco puntos correspondiente al individuo uno con registraci3n	76
Gráfico 18. Gráfico con cinco puntos correspondiente al individuo dos con registraci3n	76
Gráfico 19. Estimaci3n del polígono promedio para el individuo uno con cinco puntos y registraci3n	77
Gráfico 20. Estimaci3n del polígono promedio para el individuo 2 con cinco puntos y registraci3n	79
Gráfico 21. Gráfico con siete puntos correspondiente al individuo uno con registraci3n	82
Gráfico 22. Gráfico con siete puntos correspondiente al individuo dos con registraci3n	82
Gráfico 23. Estimaci3n del polígono promedio para el individuo uno con siete puntos y registraci3n	83

Gráfico 24. Estimación del polígono promedio para el individuo dos con siete puntos y registraci3n.....	85
Gráfico 25. Evoluci3n de la efectividad promedio del modelo de Procrustes de efectos aleatorios	99

ÍNDICE DE TABLAS

Tabla 1. Matriz de error para evaluar el modelo luego de ajustarlo en el m -ésimo individuo	89
Tabla 2. Matriz de error del modelo Procrustes con tres puntos de referencia en el primer individuo sin registraci3n	91
Tabla 3. Matriz de error del modelo Procrustes con tres puntos de referencia en el segundo individuo sin registraci3n.....	92
Tabla 4. Matriz de error del modelo Procrustes con tres puntos de referencia en el primer individuo con registraci3n	93
Tabla 5. Matriz de error del modelo Procrustes con tres puntos de referencia en el segundo individuo con registraci3n	93
Tabla 6. Matriz de error del modelo Procrustes con cinco puntos de referencia en el primer individuo sin registraci3n	94
Tabla 7. Matriz de error del modelo Procrustes con cinco puntos de referencia en el segundo individuo sin registraci3n.....	95
Tabla 8. Matriz de error del modelo Procrustes con cinco puntos de referencia en el primer individuo con registraci3n	95
Tabla 9. Matriz de error del modelo Procrustes con cinco puntos de referencia en el segundo individuo con registraci3n	96
Tabla 10. Matriz de error del modelo Procrustes con siete puntos de referencia en el primer individuo sin registraci3n	97

Tabla 11. Matriz de error del modelo Procrustes con siete puntos de referencia en el segundo individuo sin registraci3n.....	97
Tabla 12. Matriz de error del modelo Procrustes con siete puntos de referencia en el primer individuo con registraci3n	98
Tabla 13. Matriz de error del modelo Procrustes con siete puntos de referencia en el segundo individuo con registraci3n	98

INTRODUCCIÓN

En muchas aplicaciones de diversas áreas el reconocimiento de individuos es parte de vital importancia sea por razones de seguridad, privacidad, etc. Existen varias alternativas para lograrlo, como por ejemplo el uso de huellas dactilares, o el clásico método de usuario y contraseña. No obstante existen métodos más sofisticados enfocados en el rostro de los individuos. Los primeros avances en reconocimiento facial datan de inicios de la década de los 60. El uso de patrones es un pilar fundamental en esta área actualmente y con los avances computacionales cada vez mayores, es posible crear elaboradas técnicas automatizadas para este cometido (1). Los primeros métodos consistían en sencillos modelos geométricos pero ahora se habla de toda un vasto campo que incluye sofisticadas técnicas matemáticas, tanto para fines de autenticación como verificación de individuos (2).

El reconocimiento facial requiere complejas técnicas visuales para la identificación de los rostros. Este proceso puede ser resumido de manera sencilla en tres pasos: adquisición, normalización y reconocimiento. La adquisición se refiere a la forma en que son recibidos los rostros a través de imágenes; la normalización comprende el conjunto de operaciones mediante las cuales las imágenes son llevadas a formatos conocidos de datos usando cierta lógica común para todas como por ejemplo usar coordenadas, polígonos, etc. Finalmente la etapa de reconocimiento consiste en aplicar un determinado método, modelo, algoritmo, etc que reciba los datos normalizados y realice la tarea de reconocimiento propiamente dicha (1).

Históricamente, el desarrollo del reconocimiento facial ha sido tan prioritario como el de la inteligencia artificial tanto por sus importancias prácticas como por el conocimiento teórico que genera. Los sistemas de reconocimiento están inspirados en la forma en la que realizan esta actividad los humanos, sean estos dactilares, faciales, retinales, etc. No obstante en los sistemas faciales es en donde se profundiza más en investigación debido a que es el menos invasivo a la privacidad de las personas (3). Hay dos enfoques principales en los sistemas de reconocimiento facial, que son el *enfoque geométrico* y el *enfoque pictórico*.

El enfoque geométrico utiliza la configuración espacial de las características de los rostros. Esto significa que dichas características principales como por ejemplo los ojos, la nariz y la boca se expresan en función de parámetros geométricos como distancia o ángulo entre las características. Por otro lado, el enfoque pictórico consiste en plantillas donde se resaltan las características principales del rostro y su contorno y el reconocimiento se efectúa en vistas frontales de los rostros (2). Bajo el enfoque geométrico se soporta el modelo mixto que se planteará en este trabajo. Existen técnicas que combinan ambos enfoques como los métodos basados en plantillas deformables que son aplicables en situaciones donde varía la posición del rostro y la expresión anímica.

El reconocimiento facial desde sus orígenes ha estado fuertemente ligado a las redes neuronales (4). El más famoso de los primeros sistemas que usaban estas redes es el modelo de Kohonen que consistía en una red neuronal simple que permitía reconocer rostros en imágenes alineadas y normalizadas. El tipo de red que él empleó se basaba

en aproximación de los vectores propios de la matriz de varianzas y covarianzas de las imágenes de un determinado rostro, estos vectores propios son conocidos en inglés como *eigenfaces* (5). Posteriormente surgieron más métodos tanto de enfoque geométrico como de enfoque pictórico, surgió el método de Componentes Principales y otros que combinaban las redes neuronales con sistemas expertos (sistemas informáticos con un grado de inteligencia artificial). El método basado en Análisis de Discriminante también se difundió ampliamente al igual que el algoritmo *Elastic Bunch Graph Matching* abreviado como EBGM (2).

Análisis de componentes principales (PCA)

Esta técnica fue desarrollada en 1988 por Kirby y Sirovich (6). En este método las imágenes deben tener el mismo tamaño y las características notables deben estar alineadas entre las imágenes. Luego el siguiente paso consiste en reducir la dimensión de los datos y conseguir la estructura de dimensión más pequeña de patrones faciales más efectiva. Esta reducción de dimensiones elimina información no relevante y descompone precisamente las estructuras de las caras en componentes ortogonales (no correlacionados) que coinciden con las *eigenfaces* (caras propias) mencionadas anteriormente. Cada imagen es descrita como una combinación lineal de las caras propias. Para comparar una nueva cara con las ya almacenadas se mide la distancia entre los respectivos vectores propios; es decir, se compara la nueva cara con las caras propias. Este método tiene la desventaja de que se necesitan vistas frontales completas de las caras pero tiene la ventaja de que se necesitan pocas imágenes de un individuo para poder identificarlo (7).

Análisis de discriminante lineal (LDA)

Es un método estadístico que clasifica muestras de clases desconocidas basándose en muestras de entrenamiento que pertenecen a clases conocidas, y lo hace maximizando la varianza entre clases (individuos) y minimizando su contraparte intracase (intra-individuo). Cuando existe una alta dimensionalidad en las muestras y estas no son abundantes, esta técnica no se ve muy afectada por el hecho de que hayan pocas muestras de entrenamiento disponibles comparadas con la alta dimensionalidad de las mismas (8).

Algoritmo EBGM: Elastic Bunch Graph Matching

Esta técnica se sustenta en el hecho de que las imágenes de rostros reales tienen muchas características no lineales como por ejemplo la cantidad de luminosidad, posición de la cabeza o expresión facial (9). Dada una imagen de un rostro cualquiera, la *transformación de Gabor* se usa para proyectar el rostro en un grid (cuadrícula) elástico. El comportamiento de la imagen en un pixel cualquiera se obtiene como la convolución de la imagen con el filtro de Gabor; de esta manera se detectan formas en el rostro y se extraen las características notables (nodos de Gabor). El reconocimiento facial se basa en la similitud en la respuesta del filtro de Gabor en cada nodo de Gabor de la imagen de entrada y las que se usan como muestra de entrenamiento (10).

Las técnicas explicadas anteriormente son las pioneras en el campo del reconocimiento facial y son usadas actualmente pero eso no significa que sean las únicas, existen muchos otros métodos. Se pueden aplicar los *modelos lineales mixtos* para realizar esta tarea, lo cual es la razón de ser del presente trabajo.

Los modelos lineales mixtos (LME, por sus siglas en inglés) pertenecen a una clase de modelos llamada modelos sujeto-específicos, en donde cada individuo descrito por el modelo posee parámetros en el modelo específicos a él (11). En este caso cada individuo estaría representado por un conjunto de rostros. Los modelos lineales mixtos sirven para describir relaciones entre variables en datos agrupados. La(s) variable(s) de respuesta es (son) considerada(s) como una función lineal de la(s) variable(s) independiente(s). Los coeficientes de la función lineal pueden representar *efectos fijos* o *efectos aleatorios*. Los efectos fijos son factores que influyen en la variable de respuesta y que pueden ser fijados durante un experimento, los efectos aleatorios son factores que también influyen en la variable de respuesta pero vienen dados por variables aleatorias. Los *modelos mixtos* son aquellos que incluyen tanto efectos fijos como efectos aleatorios (12). En el marco teórico se dirá todo lo necesario sobre los modelos mixtos y los conocimientos convenientes para comprender este trabajo.

CAPÍTULO 1

1. Marco teórico, metodología, fuente de datos y objetivos

1.1 Introducción

Detallados los precedentes de la tarea de reconocimiento facial en la sección de Introducción, en este capítulo se procederá a describir el marco teórico, es decir el conjunto de conocimientos que son necesarios para poder entender este trabajo. También se detallará la metodología que se utilizará, la cual consiste en el conjunto de pasos que se deberán realizar para llegar al objetivo final. Se muestra también cual será la fuente de datos a usar y qué formato deberá tener para su debido análisis, así como demás pormenores que se consideren necesarios. Al final del capítulo se planteará el problema, los objetivos de este trabajo y las hipótesis de los hallazgos que se esperan.

1.2 Marco teórico

1.2.1 El producto Kronecker

Sean dos matrices $A \in M_{m \times n}$ y $B \in M_{p \times q}$, entonces el **producto Kronecker** de A con B , denotado por $A \otimes B$, es una matriz de orden $mp \times nq$ que se define como:

$$A \otimes B = \begin{bmatrix} A_{11}B & A_{12}B & \dots & A_{1n}B \\ A_{21}B & A_{22}B & \dots & A_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1}B & A_{m2}B & \dots & A_{mn}B \end{bmatrix}$$

Es decir el producto Kronecker entre dos matrices A y B consiste en multiplicar cada uno de los elementos de la matriz A por la matriz B .

Algunas propiedades que cumple el producto Kronecker son las siguientes:

$A \otimes B \neq B \otimes A$ en general	(Anticonmutativa)
$A \otimes (B + C) = A \otimes B + A \otimes C$	(Distributiva)
$(A \otimes B) \otimes C = A \otimes (B \otimes C)$	(Asociativa)
$a_1 A \otimes a_2 B = a_1 a_2 (A \otimes B)$	(Linealidad)
$(A \otimes B)(C \otimes D) = AC \otimes BD$	(Producto mixto)
$(A \otimes B)^T = A^T \otimes B^T$	(Transposición del prod. Kronecker)
$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$	(Inverso del prod. Kronecker)
$tr(A \otimes B) = tr(A)tr(B)$	(Traza del prod. Kronecker)
$rang(A \otimes B) = rang(A)rang(B)$	(Rango del prod. Kronecker)
$det(A \otimes B) = (det(A)^{rang(B)})(det(B)^{rang(A)})$	(Determinante del prod. Kronecker)

1.2.2 La Distribución Normal Multivariada

Muchas pruebas y técnicas estadísticas univariadas donde se requieren hacer suposiciones con respecto a una variable o parámetro de interés (estadística paramétrica) tienen en la distribución normal la piedra angular requerida para poder obtener conclusiones y realizar inferencias. Cuando ya no es una, sino varias las variables en cuestión y requieren ser analizadas de manera conjunta, entonces surge la necesidad de definir una distribución multivariada que sirva de soporte a las inferencias que se quieran realizar. Es por ello que surge la *distribución normal multivariada*.

La distribución normal multivariada tiene varias ventajas como:

- La distribución puede ser descrita completamente en términos solo de medias, varianzas y covarianzas.
- Los gráficos bidimensionales muestran tendencia lineal.
- Si no hay correlación entre las variables, estas son independientes.
- Las funciones lineales de variables con distribución normal tienen también distribución normal.
- Y la más importante es que incluso cuando la verdadera distribución de un conjunto de variables no sea normal multivariada, ésta es una aproximación muy buena a la hora de estimar vectores de medias debido al *Teorema del Límite Central*.

I. La distribución normal univariada

Se dice que una variable aleatoria X tiene distribución normal con media μ y varianza σ^2 si su función de densidad de probabilidad viene dada por:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad -\infty < x < \infty \quad (1.1)$$

Entonces decimos que la variable X tiene distribución normal con media μ y varianza σ^2 , lo cual se denota por $X \sim N(\mu, \sigma^2)$. La forma de esta distribución es la de la denominada campana de Gauss. Cuando $\mu = 0$ y $\sigma^2 = 1$ se denomina distribución normal estándar. Para estos valores, la distribución normal luce así:

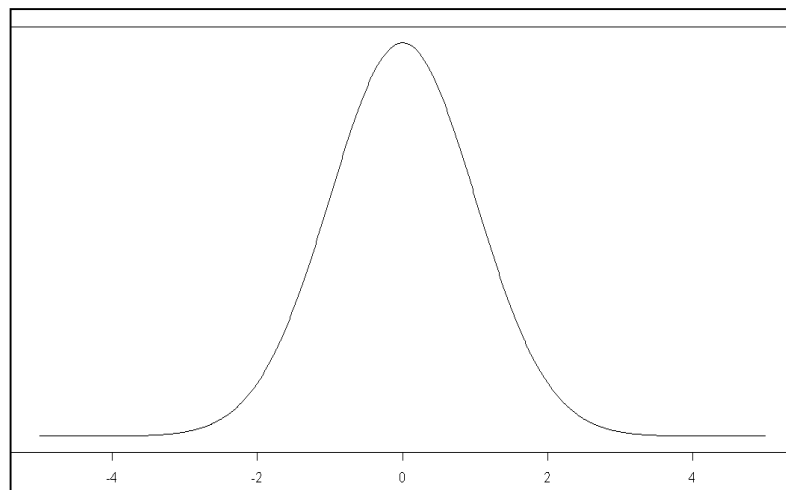


Ilustración 1. Distribución normal estándar

II. La distribución normal multivariada

Sea \mathbf{X} un vector aleatorio (un vector cuyas componentes son variables aleatorias) con distribución normal multivariada con vector de medias $\boldsymbol{\mu}$ y matriz de varianzas y covarianzas $\boldsymbol{\Sigma}$, entonces la densidad de probabilidad de \mathbf{X} viene dada por:

$$f(\mathbf{x}) = \frac{1}{(\sqrt{2\pi})^p \sqrt{|\boldsymbol{\Sigma}|}} e^{-(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})/2} \quad (1.2)$$

Donde p es el número de variables (componentes) del vector \mathbf{X} . Decimos entonces que $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Es fácil notar que es una generalización de la distribución normal univariada (13). El término del numerador en el exponente se conoce como la distancia de Mahalanobis o distancia cuadrática generalizada entre \mathbf{x} y $\boldsymbol{\mu}$.

$$\Delta^2 = (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \quad (1.3)$$

Para el caso de una variable ($p=1$) esta distancia es equivalente a $(x - \mu)(\sigma^2)^{-1}(x - \mu)$ que es el exponente de la distribución normal univariada. La varianza, en el caso unidimensional mide el grado de dispersión de x respecto de su media μ , en el caso multivariado quien proporciona esta información es $|\boldsymbol{\Sigma}|$: mientras este determinante sea mayor, mayor será la dispersión y viceversa. Para el caso bivariado, esta distribución luce como una campana de Gauss en tres dimensiones.

III. Estimación de los parámetros de la distribución normal multivariada con el método de máxima verosimilitud

Cuando se supone distribución normal multivariada en un conjunto de datos, hay que proceder a estimar los parámetros de la misma, i.e. vector de medias y matriz de varianzas y covarianzas. Hay varias técnicas disponibles, siendo el *Método de*

Máxima Verosimilitud uno de los más ampliamente extendidos. Este método consiste en: conocidos los vectores de observaciones de n individuos $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$ se hallan a partir de ellos los valores de $\boldsymbol{\mu}$ y de $\boldsymbol{\Sigma}$ que maximicen el valor de la densidad conjunta de los \mathbf{X} 's, llamada la *función de verosimilitud*. Estos vectores constituyen una muestra aleatoria, por lo cual son independientes y la densidad conjunta de los mismos es por tanto el producto de las densidades de cada uno, se tiene:

$$\begin{aligned} L(\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \prod_{i=1}^n f(\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ &= \prod_{i=1}^n \frac{1}{(\sqrt{2\pi})^p \sqrt{|\boldsymbol{\Sigma}|}} e^{-(\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) / 2} \\ &= \frac{1}{(\sqrt{2\pi})^{np} \sqrt{|\boldsymbol{\Sigma}|^n}} e^{-\sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) / 2} \end{aligned}$$

Cuando el número de individuos es muy grande, se dificulta la optimización de esta función. Matemáticamente, es equivalente maximizar esta función o su logaritmo, dado por:

$$\begin{aligned} l(\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \ln L(\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ &= -\frac{np}{2} \ln(2\pi) - \frac{n}{2} \ln(|\boldsymbol{\Sigma}|) - \sum_{i=1}^n \frac{(\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu})}{2} \end{aligned}$$

Se puede probar que los estimadores de máxima verosimilitud al optimizar esta función son:

$$\hat{\boldsymbol{\mu}} = \bar{\mathbf{x}}$$

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \quad (1.4)$$

1.2.3 La prueba de pertenencia

La prueba de pertenencia permite responder a una pregunta que más adelante se formulará: dados N individuos idénticamente distribuidos y un nuevo individuo (el individuo $N+1$), ¿Podría pertenecer este nuevo individuo a la misma población?, es decir: ¿tiene este nuevo individuo la misma distribución que los primeros y con los mismos parámetros? Un resultado conocido de estadística básica dice que *la suma de N variables aleatorias normales estándar elevadas al cuadrado es igual a una variable aleatoria ji-cuadrado con N grados de libertad*, es decir:

$$z_1^2 + z_2^2 + \dots + z_N^2 \sim \chi_{(N)}^2, \quad z_i \sim N(0,1) \quad (1.5)$$

Si se define el vector aleatorio $\mathbf{z} = (z_1, z_2, \dots, z_n)^T$ entonces:

$$\mathbf{z}^T \mathbf{z} \sim \chi_{(N)}^2 \quad (1.6)$$

Para el caso multivariado, sean N individuos $\mathbf{X}_i, i = 1, 2, \dots, N$ a los que se les ha medido p características y que tienen distribución normal multivariada con media $\boldsymbol{\mu}$ y matriz de varianzas y covarianzas $\boldsymbol{\Sigma}$. En este caso se cumple que:

$$(\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) \sim \chi_{(p)}^2 \quad (1.7)$$

Lo que es verificable haciendo $\mathbf{z} = \boldsymbol{\Sigma}^{-1/2} (\mathbf{x}_i - \boldsymbol{\mu})$ en (1.6) y sabiendo también que $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}^T$ (14). Esto es también válido si se reemplaza $\boldsymbol{\mu}$ por su estimador $\bar{\mathbf{x}}$, cuya distribución viene dada por:

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^N n_i \sim \mathcal{N}\left(\boldsymbol{\mu}, \frac{1}{N} \boldsymbol{\Sigma}\right) \quad (1.8)$$

Suponiendo que llega un $(N+1)$ -ésimo individuo, para responder a la pregunta planteada al inicio de esta sección se realizará el supuesto de que este individuo tiene distribución normal multivariada con los parámetros siguientes:

$$\mathbf{X}_{N+1} \sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}) \quad (1.9)$$

La pertenencia de este nuevo individuo a la misma población que los primeros, se reduce a contrastar las hipótesis:

$$H_0: \boldsymbol{\mu} = \boldsymbol{\mu}_* \text{ vs } H_A: \boldsymbol{\mu} \neq \boldsymbol{\mu}_* \quad (1.10)$$

Si la hipótesis nula fuese verdadera, entonces $\mathbf{X}_{N+1} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Por tanto:

$$(\bar{\mathbf{x}} - \mathbf{X}_{N+1}) \sim \mathcal{N}\left(\mathbf{0}, \left(1 + \frac{1}{N}\right) \boldsymbol{\Sigma}\right) \quad (1.11)$$

Aplicando el resultado (1.7), se tiene:

$$(\bar{\mathbf{x}} - \mathbf{X}_{N+1})^T \left[\left(1 + \frac{1}{N}\right) \boldsymbol{\Sigma} \right]^{-1} (\bar{\mathbf{x}} - \mathbf{X}_{N+1}) \sim \chi_{(p)}^2 \quad (1.12)$$

Con lo cual el estadístico de prueba ep es:

$$ep = \left(\frac{N}{N+1} \right) (\bar{\mathbf{x}} - \mathbf{x}_{N+1})^T \boldsymbol{\Sigma}^{-1} (\bar{\mathbf{x}} - \mathbf{x}_{N+1}) \sim \chi_{(p)}^2 \quad (1.13)$$

Por tanto, se dice con $(1 - \alpha) * 100\%$ de confianza que el individuo $N+1$ **no** pertenece a la misma población que los N primeros si se cumple que:

$$ep > \chi_{(p, \alpha)}^2$$

Donde $\chi_{(p,\alpha)}^2$ es el percentil $1 - \alpha$ de la distribución ji-cuadrado con p grados de libertad. O de manera equivalente, si la probabilidad de que la correspondiente variable ji-cuadrado sea mayor que este estadístico es menor que α (esta probabilidad se llama valor P), es decir:

$$P(\chi_{(p)}^2 \geq ep) < \alpha \quad (1.14)$$

Si esta probabilidad es mayor o igual que α entonces se dice que no hay evidencia de que el individuo $N+1$ pertenezca a una población distinta, o de manera equivalente si $ep \leq \chi_{(p,\alpha)}^2$. Si se usa un estimador de Σ , como la matriz de varianzas y covarianzas de la muestra (matriz de covarianzas empírica) o el estimador de una matriz de covarianzas estructurada originada de un modelo lineal mixto, entonces (1.13) se cumple de manera aproximada con un tamaño de muestra que no sea pequeño y por tanto las conclusiones con el valor P dadas por (1.14) pierden potencia.

1.2.4 El modelo lineal mixto

1. Introducción

Antes de definir el modelo lineal general, se planteará un ejemplo introductorio (15). Consideremos dos variables cuyo uso es clásico en estadística: peso y estatura, las cuales son medidas en *sujetos* que pertenecen a una determinada *familia*, es decir cada fila de la matriz de datos tiene la forma (F_k, P_k, E_k) donde F_k denota a qué familia pertenece la k-ésima persona, P_k es su respectivo peso y E_k es su estatura. Sin pérdida de generalidad, supongamos que los miembros de una misma familia tienen estaturas y pesos similares.

Es de esperarse una relación positiva entre las variables Peso y Estatura. Una forma de expresar esta relación es mediante un modelo lineal sencillo como el siguiente:

$$P_k = \alpha + \beta E_k + \varepsilon_k \quad (1.15)$$

Los coeficientes α y β podrían ser estimados usando el método de mínimos cuadrados. No obstante, se estaría ignorando la correlación intrafamiliar, es decir la existente entre miembros de una misma familia (causada por similares hábitos alimenticios, deportivos, o por razones genéticas, etcétera); y el hacerlo proporciona estimaciones menos eficientes de los coeficientes α y β del modelo y por tanto afecta la relación entre ambas variables. Para mejorar estas estimaciones se debe tomar en cuenta la correlación intrafamiliar (o *intracluster*), es decir hay que dar una estructura a la correlación total, que la incluya. El objetivo será entonces el plantear un modelo lineal que tome en cuenta estas correlaciones intraclusters. El siguiente paso será estimar el mencionado modelo bajo el supuesto de normalidad.

Una manera de hacer esto es con el uso de dos subíndices (i, j) en lugar de uno, para denotar al j -ésimo miembro de la i -ésima familia (cluster). Denotando por n_i al número de elementos que hay en el i -ésimo cluster, y los vectores $\mathbf{P}_i = (P_1, P_2, \dots, P_{n_i})^T$ y $\mathbf{E}_i = (E_1, E_2, \dots, E_{n_i})^T$ como el peso y la altura de los individuos del i -ésimo cluster. El número total de individuos es $N_T = \sum_{i=1}^N n_i$ donde N denota el número total de clusters. La correlación intracluster se puede suponer que viene dada por una constante positiva ρ para miembros dentro de una misma familia, i.e: $cor(W_{ij}, W_{ik}) = \rho$ para $j \neq k$ y para $i=1, 2, \dots, N$. A diferencia del modelo anterior, se

puede suponer que el intercepto no es un parámetro sino que es aleatorio, aunque es más realista el considerar que cada cluster tiene su propio intercepto $a_i = \alpha + b_i$ donde b_i es una variable aleatoria o **efecto aleatorio** con varianza desconocida y constante σ_d^2 y α es el intercepto común. Con las definiciones anteriores de \mathbf{P}_i y \mathbf{E}_i podríamos escribir el modelo como $\mathbf{P}_i = a_i + \beta \mathbf{E}_i + \boldsymbol{\varepsilon}_i$, suponga que $\boldsymbol{\varepsilon}_i$ es el vector de errores con promedio cero, correlación cero entre sus componentes y varianza σ^2 . Reemplazando en el modelo propuesto inicialmente y los nuevos vectores definidos tenemos:

$$\mathbf{P}_i = \alpha + \beta \mathbf{E}_i + b_i \mathbf{Z}_i + \boldsymbol{\varepsilon}_i \quad (1.16)$$

Donde $\mathbf{Z}_i = \mathbf{1}_i = (1, 1, \dots, 1)^T$ es un vector columna de dimensión n_i . Se puede probar que la matriz de varianzas y covarianzas de \mathbf{P}_i tiene la forma $\sigma^2 \mathbf{V}_i$ donde \mathbf{V}_i es una matriz de orden $n_i \times n_i$ definida por:

$$\mathbf{V}_i = \begin{bmatrix} 1 + \frac{\sigma_d^2}{\sigma^2} & \dots & \frac{\sigma_d^2}{\sigma^2} \\ \vdots & \ddots & \vdots \\ \frac{\sigma_d^2}{\sigma^2} & \dots & 1 + \frac{\sigma_d^2}{\sigma^2} \end{bmatrix}$$

El punto clave de los modelos mixtos es que se pueden obtener estimaciones más eficientes de los parámetros, si se le da una estructura a la matriz de varianzas y covarianzas debido a la correlación intracluster. Esta estructura generalmente es de la forma $\boldsymbol{\Sigma}_i = \sigma^2 \mathbf{V}_i$. Uno de los métodos más usados para hallarla es el de mínimos cuadrados generalizado (GLS), el cual minimiza la suma de cuadrados ponderada dada por (1.17)

$$\sum_{i=1}^{n_i} (\mathbf{P}_i - \alpha - \beta \mathbf{E}_i)^T \mathbf{V}_i^{-1} (\mathbf{P}_i - \alpha - \beta \mathbf{E}_i) \quad (1.17)$$

El ratio σ_d^2/σ^2 puede ser estimado usando el método de máxima verosimilitud suponiendo distribución normal.

II. Definición

Con el preámbulo anterior ya se puede definir el Modelo Lineal Mixto (LME) (15). Sean N_T observaciones agrupadas en N clusters de tamaño n_i para $i = 1, 2, \dots, N$ en las que se ha medido una variable de respuesta Y y m variables explicativas X , las cuales son conocidas por lo que las llamaremos **efectos fijos**. Suponiendo que la variabilidad de Y es explicada por estas m variables explicativas pero también por k **efectos aleatorios** que son variables aleatorias no observables con distribución normal y varianza constante. Entonces el modelo lineal mixto, como lo definieron Laird y Ware (1982) viene dado por:

$$\mathbf{y}_i = \mathbf{X}_i \boldsymbol{\beta} + \mathbf{Z}_i \mathbf{b}_i + \boldsymbol{\varepsilon}_i \quad i = 1, 2, \dots, N \quad (1.18)$$

Donde:

- \mathbf{y}_i es un vector de orden $n_i \times 1$ con la variable Y medida en el i -ésimo grupo o cluster.
- \mathbf{X}_i es una matriz de orden $n_i \times m$ que contiene las m variables de efecto fijo medidas en el i -ésimo grupo o cluster.
- $\boldsymbol{\beta}$ es un vector de orden $m \times 1$ que contiene los coeficientes o parámetros poblacionales de efectos fijos.

- \mathbf{Z}_i es una matriz de orden $n_i \times k$ llamada matriz de diseño de efectos aleatorios.
- $\boldsymbol{\varepsilon}_i$ es un vector de orden $n_i \times 1$ de errores aleatorios; sus componentes son independientes, tienen promedio cero y varianza (intracluster) σ^2 .
- \mathbf{b}_i es un vector de orden $k \times 1$ que contiene los efectos aleatorios que tienen promedio cero y matriz de varianzas y covarianzas $\mathbf{D}_* = \sigma^2 \mathbf{D}$. Se supone también que cada uno de estos vectores son independientes.

Esta definición del modelo comprende N ecuaciones distintas, una para cada cluster. Se puede escribir el modelo en una forma más general, “apilando” estas N ecuaciones en una sola. Así:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{b} + \boldsymbol{\varepsilon} \quad i = 1, 2, \dots, N \quad (1.19)$$

En donde se tiene:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \end{bmatrix} \in M_{N_T \times 1} \quad \mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_N \end{bmatrix} \in M_{N_T \times m}$$

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{Z}_N \end{bmatrix} \in M_{N_T \times Nk} \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_N \end{bmatrix} \in M_{Nk \times 1}$$

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \boldsymbol{\varepsilon}_1 \\ \boldsymbol{\varepsilon}_2 \\ \vdots \\ \boldsymbol{\varepsilon}_N \end{bmatrix} \in M_{N_T \times 1} \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{bmatrix} \in M_{m \times 1}$$

Se pueden agrupar las fuentes de variación no controlables, es decir el error y los efectos aleatorios, en un solo término de error $\boldsymbol{\eta} = \mathbf{Z}\mathbf{b} + \boldsymbol{\varepsilon}$, donde $\boldsymbol{\eta}$ viene dado por:

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \\ \vdots \\ \boldsymbol{\eta}_N \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varepsilon}_1 + \mathbf{Z}_1 \mathbf{b}_1 \\ \boldsymbol{\varepsilon}_2 + \mathbf{Z}_2 \mathbf{b}_2 \\ \vdots \\ \boldsymbol{\varepsilon}_N + \mathbf{Z}_N \mathbf{b}_N \end{bmatrix} \quad (1.20)$$

Realizando los siguientes supuestos:

$$\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{n_i}) \quad (1.21)$$

$$\mathbf{b}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{D})$$

Se puede verificar también que $\text{var}(\mathbf{b}) = \sigma^2 (\mathbf{I}_N \otimes \mathbf{D})$. Con lo que el modelo también podría ser escrito de la siguiente manera:

$$\mathbf{y}_i \sim \mathcal{N}(\mathbf{X}_i \boldsymbol{\beta}, \sigma^2 (\mathbf{I}_{n_i} + \mathbf{Z}_i \mathbf{D} \mathbf{Z}_i^T)) \quad (1.22)$$

Y sabiendo que $E(\boldsymbol{\eta}) = \mathbf{0}$ se puede comprobar que la varianza de $\boldsymbol{\eta}$ viene dada por:

$$V(\boldsymbol{\eta}) = \mathbf{V} = \sigma^2 \begin{bmatrix} \mathbf{I}_{n_1} + \mathbf{Z}_1 \mathbf{D} \mathbf{Z}_1^T & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_2} + \mathbf{Z}_2 \mathbf{D} \mathbf{Z}_2^T & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{I}_{n_N} + \mathbf{Z}_N \mathbf{D} \mathbf{Z}_N^T \end{bmatrix}$$

III. Función de verosimilitud del modelo lineal mixto.

Como \mathbf{y}_i tiene distribución normal multivariada como se menciona en (1.22), su función de verosimilitud, quitando el término constante, sería la siguiente:

$$l(\sigma^2, \boldsymbol{\beta}, \mathbf{D}) = -\frac{1}{2} \left\{ N_T \ln \sigma^2 + \sum_{i=1}^N \left[\ln |\mathbf{I} + \mathbf{Z}_i \mathbf{D} \mathbf{Z}_i^T| + \frac{1}{\sigma^2} (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta})^T (\mathbf{I} + \mathbf{Z}_i \mathbf{D} \mathbf{Z}_i^T)^{-1} (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta}) \right] \right\} \quad (1.23)$$

Esta función será de utilidad posteriormente ya que se estimarán los parámetros del modelo mixto para reconocimiento facial usando el método de máxima verosimilitud.

1.2.5 Mínimos cuadrados para la varianza (VLS)

Se usará el método de *Variance Least Squares* (VLS) o *Mínimos Cuadrados para la Varianza* para estimar los parámetros σ^2 y \mathbf{D}_* en el modelo lineal mixto (LME) definido en (1.18) (15). Los estimadores obtenidos por este método pueden ser utilizados como valores iniciales en métodos computacionales para otro método de estimación, como el método de máxima verosimilitud. VLS es muy útil tanto para modelos mixtos lineales como no lineales.

El estimador VLS surge intuitivamente del método de mínimos cuadrados ordinarios. Sea $\hat{\boldsymbol{\beta}}_{OLS}$ el estimador de mínimos cuadrados ordinarios de $\boldsymbol{\beta}$ en el modelo (1.18), lo que se tratará de conseguir es que este estimador OLS coincida con el verdadero vector de parámetros. Suponga que $\hat{\boldsymbol{e}}_i = \mathbf{y}_i - \mathbf{X}_i \hat{\boldsymbol{\beta}}_{OLS}$ representa el vector de error del individuo (cluster) i , el cual es la estimación de $\boldsymbol{\eta}_i$ definido en (1.20). La matriz de covarianzas empírica puede ser obtenida como $\hat{\boldsymbol{e}}_i \hat{\boldsymbol{e}}_i^T$. Nuestro objetivo será encontrar estimadores para σ^2 y \mathbf{D}_* .

Es un resultado conocido que, si \mathbf{A} y \mathbf{B} son matrices de dimensión $m \times n$ entonces $tr(\mathbf{A}\mathbf{B}^T) = tr(\mathbf{A}^T\mathbf{B}) = \sum_{i,j}(a_{ij}b_{ij})$. Si $\mathbf{A} = \mathbf{B}$ y además es simétrica, entonces $tr(\mathbf{A}^2) = \sum_i(a_{ii}^2)$. Si hacemos que $\mathbf{A} = \mathbf{X} - \mathbf{Y}$ donde \mathbf{X} e \mathbf{Y} son matrices simétricas, entonces $tr(\mathbf{A}^2) = tr[(\mathbf{X} - \mathbf{Y})^2] = \sum_i[(x_{ii} - y_{ii})^2]$ lo cual es una medida de la distancia entre dos matrices. Si tomamos $\mathbf{X} = \hat{\boldsymbol{e}}_i \hat{\boldsymbol{e}}_i^T$ como la matriz de covarianzas empírica y a $\mathbf{Y} = \sigma^2(\mathbf{I}_{n_i} + \mathbf{Z}_i \mathbf{D} \mathbf{Z}_i^T) = \sigma^2 \mathbf{I}_{n_i} + \mathbf{Z}_i \mathbf{D}_* \mathbf{Z}_i^T$ como la matriz de covarianzas del modelo, entonces se pueden estimar σ^2 y \mathbf{D}_* minimizando la

distancia entre ellas, es decir *minimizando la suma de las trazas del cuadrado de la diferencia matricial entre la matriz de covarianzas empírica y la teórica para cada individuo*. El estimador resultante se denomina *estimador VLS*. Resumidamente, el estimador VLS de σ^2 y \mathbf{D}_* del modelo (1.18) minimiza la siguiente suma cuadrática de trazas:

$$\sum_{i=1}^N \text{tr} \left[(\hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T - \sigma^2 \mathbf{I}_{n_i} - \mathbf{Z}_i \mathbf{D}_* \mathbf{Z}_i^T)^2 \right] \quad (1.24)$$

Se calculan entonces las respectivas derivadas con respecto a σ^2 y \mathbf{D}_* y se igualan a cero. Se puede utilizar también algún algoritmo numérico de optimización implementado computacionalmente. De cualquier manera, no procederemos a hallar una solución general para estos estimadores, se lo hará mediante software para nuestro caso particular de reconocimiento facial.

1.2.6 Registración de imágenes

Una imagen en escala de grises de tamaño $p \times q$ píxeles, tiene una matriz asociada de la misma dimensión. Cada elemento a_{ij} de esta matriz representa un píxel en la respectiva ubicación en la imagen. El valor de este elemento es un número entre cero y 255 que contiene la coloración del píxel, donde cero equivale a completamente negro y 255 a completamente blanco. Para el caso de las imágenes en escala de colores, uno de los estándares más usados es el RGB (Red Green Blue), según el cual cualquier color puede representarse como una combinación de los colores rojo, azul y verde; por tanto, una imagen en escala de colores tiene tres matrices asociadas: una correspondiente al color rojo, otra para el azul y otra para el verde. Cada píxel de la

imagen posee tres elementos asociados en la respectiva ubicación en cada matriz, los cuales contienen el valor requerido de los tres colores mencionados en una escala de cero a 255, cuya combinación corresponde a la coloración del pixel. Dicho esto se puede hablar de *registración de imágenes*.

La registración de imágenes es un proceso que se realiza antes de hacer comparación estadística entre imágenes. Consiste en colocar un conjunto de imágenes en la misma escala, es decir que estén debidamente alineadas y rotadas. Cuando se quiere hallar la diferencia entre dos imágenes (que es lo que busca el modelo de Procrustes), una alternativa es hacer la diferencia pixel a pixel, el inconveniente es que usualmente un pixel en una imagen no corresponde exactamente al mismo pixel en la otra imagen, por lo cual surge la necesidad de que las imágenes estén alineadas antes de hacer la diferencia pixel por pixel. A continuación se presenta un ejemplo de dos imágenes de un reloj que aparecen ya registradas (alineadas), las cuales fueron tomadas con segundos de diferencia (15):

Imagen de origen

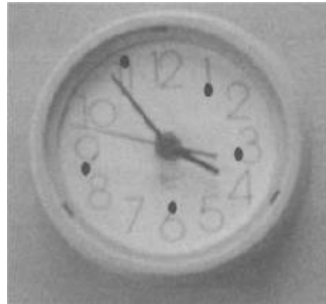
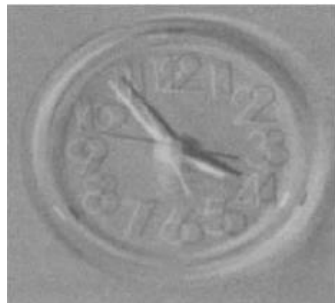


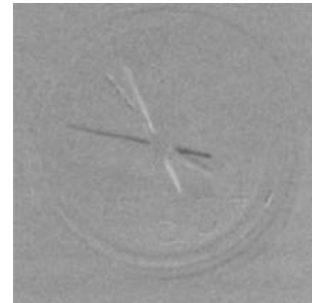
Imagen de destino



Diferencia antes de registraci3n



Diferencia luego de la registraci3n

**Ilustraci3n 2.** Registraci3n de im3genes

Las rotaciones tambi3n est3n permitidas, puesto que la imagen objetivo podr3a ser la imagen de origen girada un determinado 3ngulo θ , como se muestra en la siguiente ilustraci3n:

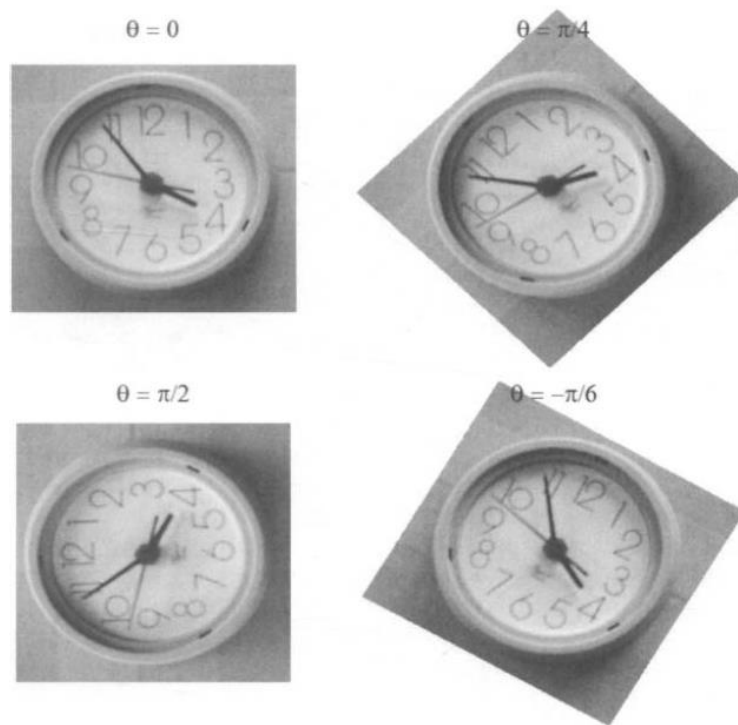


Ilustración 3. Imagen rotada en distintos ángulos

Existen diversos métodos para hacer registración de imágenes los cuales pueden ser clasificados como lineales o no lineales. Un método muy común para la registración entre dos imágenes (corregistración) es la registración afín (en inglés, *affine registration*), el cual es un método de registración lineal. Sean $\mathbf{M}_i = \{M_i(p, q); p = 1, 2, \dots, P_i; q = 1, 2, \dots, Q_i; i = 1, 2\}$ dos imágenes, donde \mathbf{M}_1 es llamada imagen de origen y \mathbf{M}_2 es llamada imagen de destino. Nos interesa hallar una transformación lineal (afín) tal que la suma cuadrática (1.25) sea mínima.

$$S = \frac{1}{|\mathcal{M}|} \sum_{(p,q) \in \mathcal{M}} [M_1(p, q) - v M_2(\beta_1 + \beta_2 p + \beta_3 q, \beta_4 + \beta_5 p + \beta_6 q) - \mu]^2 \quad (1.25)$$

Esta expresión es conocida a veces como media cuadrática del error (MCE) o función de costo. Ocho parámetros deben ser hallados: el cambio de intensidad (μ), el parámetro de escala o de intensidad (ν) y el vector de parámetros $\boldsymbol{\beta} = (\beta_1, \dots, \beta_6)^T$. Se puede suponer que la varianza en (11.25) es la misma que en ambas imágenes, o usar mínimos cuadrados ponderados. Los parámetros β_2 y β_6 controlan el zoom o alejamiento; β_3 y β_5 controlan la rotación; β_1 y β_4 controlan la traslación. Si de antemano se sabe que se pueden obviar las traslaciones y solo permitir rotación y reescalamiento, entonces $\beta_2 = \beta_6$ y $\beta_3 = -\beta_5$ (15). Se puede demostrar que minimizar la expresión (1.25) es equivalente a maximizar el *coeficiente de correlación* entre dos imágenes. No es objetivo de este trabajo el estudio detallado de este y otros métodos de registración de imágenes, pero este método ha sido descrito aquí de una manera muy breve por ser el que se usará más adelante.

1.3 Fuentes de datos y metodología

La fuente de datos será un conjunto de 16 fotografías pertenecientes a 2 individuos, ocho de cada uno de ellos. Todas son de tamaño 640×480 píxeles. Los individuos corresponderán a los *clusters* del modelo que se formulará y son estudiantes de la Escuela Superior Politécnica del Litoral. Algunas de las fotos de ellos se pueden apreciar en la Ilustración 4. Todas las fotos se encuentran en el Anexo 4.



Ilustración 4. Algunas de las imágenes que son parte de la fuente de datos

Estas fotos serán datos de entrada para el modelo de efectos aleatorios que se planteará. La herramienta informática a usar será el software RStudio (16). La tarea de efectuar reconocimiento facial con fotos y el modelo mencionado se puede resumir en cuatro pasos:

1. Conversión de las fotos en datos para el modelo mixto: Las fotos serán importadas al software mencionado con ayuda de una librería que se detallará más adelante. Este paso también comprende la manera en que se convertirán las imágenes en vectores aleatorios.
2. Desarrollo del modelo lineal mixto: una vez que se conoce la forma que tendrán los vectores que servirán de entrada para el modelo mixto, se procederá a desarrollar el mismo con todo el conocimiento teórico que se necesita el cual ya ha sido bosquejado en la sección 1.2 correspondiente al marco teórico. Se puede resumir esta etapa en:

- I. Estimación del polígono promedio desconocido (método de los momentos)
 - II. Estimación de la varianza del error y de los efectos aleatorios en dos pasos: VLS primero y después el método de máxima verosimilitud.
 - III. Aplicación de la prueba de pertenencia en el modelo Procrustes de efectos aleatorios para efectuar reconocimiento facial.
3. Fase de programación: el modelo lineal mixto desarrollado se programará en el software RStudio. Toda la programación requerida se puede encontrar en el Anexo III.
 4. Conclusiones: Se realizarán conclusiones acerca del modelo y su funcionamiento con los datos usados, su efectividad, su comportamiento y demás puntos que se consideren relevantes. Además se recomendarán posibles alternativas en caso de surgirlas en una determinada instancia del desarrollo del modelo.

1.4 Planteamiento del problema

El problema consiste en realizar distinción entre individuos a través de reconocimiento facial, mediante el modelo de Procrustes de efectos aleatorios. Dadas N fotos de un individuo y una $(N+1)$ -ésima foto, ajustar dicho modelo en las N primeras estimando sus parámetros, los cuales permitirán calcular un estadístico de prueba usado para contrastar la hipótesis nula: la $(N+1)$ -ésima foto pertenece al mismo individuo, versus la hipótesis alternativa de que pertenece a un individuo distinto.

1.5 Objetivos

1.5.1 Objetivo General

Realizar reconocimiento facial entre individuos usando el modelo de Procrustes de efectos aleatorios y evaluar su efectividad, usando varios enfoques definidos por la cantidad de puntos de referencia que se usan y el uso de registración de imágenes.

1.5.2 Objetivos Específicos

- I. Hallar una expresión matemática general que permita estimar la varianza del error y de los efectos aleatorios en el modelo de Procrustes con el método de VLS.
- II. Aplicar la prueba de pertenencia general en el modelo de Procrustes para realizar reconocimiento facial.
- III. Hallar un indicador que permita medir la efectividad del modelo de Procrustes de efectos aleatorios en reconocimiento facial.
- IV. Determinar si la registración de imágenes mejora la efectividad del modelo de Procrustes al hacer reconocimiento facial.
- V. Determinar si la cantidad de puntos de referencia que se tomen en los individuos influye en la efectividad del modelo de Procrustes de efectos aleatorios al hacer reconocimiento facial.

1.6 Hipótesis de Hallazgos

Las hipótesis que se plantearán y que se pretende comprobar con esta investigación son:

- I. Los efectos aleatorios en el modelo de Procrustes captan buena parte de la variabilidad total de las fotografías. Es decir, las estimaciones de las varianzas de los efectos aleatorios son significativamente más grandes que la del error.
- II. La efectividad del modelo de Procrustes de efectos aleatorios en reconocimiento facial es buena. Por “buena” se llamará a porcentajes de efectividad del 70% o más.
- III. Conforme aumenta la cantidad de puntos de referencia usados, aumenta también la efectividad del modelo de Procrustes de efectos aleatorios.
- IV. Realizar registración de imágenes antes de ajustar el modelo de Procrustes de efectos aleatorios mejora la efectividad del mismo.

CAPÍTULO 2

2. El Modelo de Procrustes de efectos aleatorios

2.1 Introducción

El modelo de Procrustes permite tratar el problema de analizar rostros como un problema de análisis de pares ordenados (x,y) que definen polígonos y se ven afectados por efectos aleatorios (15). Se tomarán n puntos (características notables) en cada imagen de un individuo. Cada punto tiene coordenadas (x,y) , que serán apiladas en un vector. Si n es igual a tres podrían ser los ojos y la punta de la nariz (Ilustración 5). El modelo será formulado de manera general para n puntos.

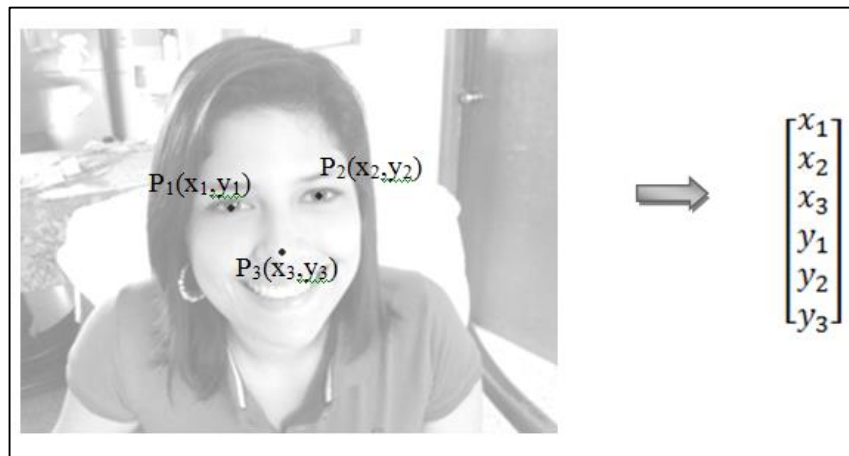


Ilustración 5. Normalización de las imágenes para el modelo Procrustes:
Ejemplo con tres puntos de referencia

2.2 El modelo de Procrustes de efectos aleatorios

Sean N polígonos definidos por n vértices con coordenadas $\mathbf{u}_{ij} = (x_{ij}, y_{ij})^T \in R^2$ donde $i = 1, 2, \dots, N$ y $j = 1, 2, \dots, n$. Suponiendo la existencia de un *polígono promedio* y que cada vértice j de cada polígono i proviene de una *rotación, reescalamiento y desplazamiento aleatorios* (es decir, son efectos aleatorios) del correspondiente vértice de este polígono promedio (15), se formula el modelo:

$$\mathbf{u}_{ij} = r_i \mathbf{R}(\theta) \boldsymbol{\mu}_j + \mathbf{t}_i + \boldsymbol{\varepsilon}_{ij} \quad (2.1)$$

Donde r_i representa el coeficiente de reescalamiento y:

$$\mathbf{R}(\theta) = \begin{pmatrix} \cos \theta_i & -\text{sen } \theta_i \\ \text{sen } \theta_i & \cos \theta_i \end{pmatrix}$$

Es la matriz que representa la rotación de las coordenadas de los vértices del polígono promedio desconocido que llamaremos $\boldsymbol{\mu}$. Y además:

- $\mathbf{u}_{ij} = (x_{ij}, y_{ij})^T$: Es el vector de coordenadas del j -ésimo punto en el i -ésimo polígono
- $\boldsymbol{\mu}_j = (\mu_{jx}, \mu_{jy})^T$: Es el vector de coordenadas del j -ésimo punto del polígono promedio desconocido
- $\mathbf{t}_i = (t_{i1}, t_{i2})^T$: Efectos aleatorios que representan desplazamiento del polígono promedio desconocido
- $\boldsymbol{\varepsilon}_{ij}$: Error aleatorio que representa las perturbaciones o errores producidos al señalar las características notables, en el caso de que se haga manualmente como en este trabajo, representa el *error de pulso*.

El ángulo θ es una variable aleatoria que toma valores entre cero y 2π ; se puede transformar el primer término del modelo (2.1) en lineal definiendo las siguientes variables aleatorias:

$$\begin{aligned} h_i &= r_i \cos \theta_i \\ g_i &= -r_i \sin \theta_i \end{aligned} \quad (2.2)$$

El modelo de efectos aleatorios es, entonces:

$$\mathbf{u}_{ij} = \begin{pmatrix} h_i & g_i \\ -g_i & h_i \end{pmatrix} \boldsymbol{\mu}_j + \mathbf{t}_i + \boldsymbol{\varepsilon}_{ij} \quad (2.3)$$

De manera obvia, el valor esperado de cada polígono debe coincidir con el polígono promedio desconocido, por ello se supondrá que el ángulo de rotación es cero en promedio, y el coeficiente de reescalamiento es uno en promedio; por tanto h_i sería aproximadamente uno en promedio y g_i sería aproximadamente cero en promedio. Además, es de esperarse que el desplazamiento promedio de los polígonos con respecto del polígono promedio sea nulo, por tanto se supondrá que el valor esperado de \mathbf{t}_i es el vector $\mathbf{0} = (0,0)^T$. No hay efectos fijos. Suponiendo además distribución normal para h_i, g_i, \mathbf{t}_i y $\boldsymbol{\varepsilon}_{ij}$; para estos últimos también varianza constante, promedio cero y correlación nula. Resumiendo esto, se tiene:

$$\begin{aligned} \begin{bmatrix} h_i \\ g_i \end{bmatrix} &\sim \mathcal{N} \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \sigma^2 \begin{bmatrix} \sigma_h^2 & \sigma_{hg} \\ \sigma_{hg} & \sigma_g^2 \end{bmatrix} \right) \\ \mathbf{t}_i &\sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \sigma^2 \begin{bmatrix} \tau_1^2 & \tau_{12} \\ \tau_{12} & \tau_2^2 \end{bmatrix} \right) \\ \boldsymbol{\varepsilon}_{ij} &\sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \sigma^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) \end{aligned} \quad (2.4)$$

Suponiendo también que estos tres vectores no están correlacionados entre sí, es fácil verificar en (2.3) que $E(\mathbf{u}_{ij}) = \boldsymbol{\mu}_j$. Se probará que la varianza de \mathbf{u}_{ij} puede ser expresada en la forma $\sigma^2 V(\boldsymbol{\mu}_j)$, donde $V(\boldsymbol{\mu}_j)$ depende solo de las varianzas de los efectos aleatorios. Es decir, la matriz de varianzas y covarianzas está estructurada, así:

$$\begin{aligned}
Var(\mathbf{u}_{ij}) &= Var \left[\begin{pmatrix} h_i & g_i \\ -g_i & h_i \end{pmatrix} \boldsymbol{\mu}_j + \mathbf{t}_i + \boldsymbol{\varepsilon}_{ij} \right] \\
&= Var \left[h_i \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \boldsymbol{\mu}_j + g_i \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \boldsymbol{\mu}_j + \mathbf{t}_i + \boldsymbol{\varepsilon}_{ij} \right] \\
&= Var(h_i \mathbf{I}_2 \boldsymbol{\mu}_j) + Var(g_i \mathbf{L} \boldsymbol{\mu}_j) + Cov(h_i, g_i) [\mathbf{I}_2 \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \mathbf{L}^T + \mathbf{L} \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \mathbf{I}_2^T] \\
&\quad + Var(\mathbf{t}_i) + Var(\boldsymbol{\varepsilon}_{ij}) \\
&= \sigma^2 [\sigma_h^2 \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T + \sigma_g^2 \mathbf{L} \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \mathbf{L}^T + \sigma_{hg} (\boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \mathbf{L}^T + \mathbf{L} \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T) + \mathbf{T} + \mathbf{I}_2]
\end{aligned}$$

Siendo:

$$\mathbf{L} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \qquad \mathbf{T} = \begin{bmatrix} \tau_1^2 & \tau_{12} \\ \tau_{12} & \tau_2^2 \end{bmatrix}$$

Por tanto:

$$Var(\mathbf{u}_{ij}) = \sigma^2 V(\boldsymbol{\mu}_j) \tag{2.5}$$

Donde $V(\boldsymbol{\mu}_j)$ es la matriz $\sigma_h^2 \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T + \sigma_g^2 \mathbf{L} \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \mathbf{L}^T + \sigma_{hg} (\boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \mathbf{L}^T + \mathbf{L} \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T) + \mathbf{T} + \mathbf{I}_2$. Esto es válido para las coordenadas de un determinado vértice de un polígono como podrá notarse. Se generalizará el modelo (2.3) para que incluya todos los vértices de un polígono i . Definiendo el vector \mathbf{u}_i que contiene todas las coordenadas de los n puntos del i -ésimo polígono y el vector $\boldsymbol{\varepsilon}_i$ con los respectivos errores

aleatorios. Si se denota por $\boldsymbol{\mu}$ al vector de orden $(2n) \times 1$ que contiene las coordenadas respectivas del polígono promedio desconocido, se tiene:

$$\mathbf{u}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \\ y_{i1} \\ y_{i2} \\ \vdots \\ y_{in} \end{bmatrix} \quad \boldsymbol{\varepsilon}_i = \begin{bmatrix} \varepsilon_{i1} \\ \varepsilon_{i2} \\ \vdots \\ \varepsilon_{in} \\ \varepsilon_{i1} \\ \varepsilon_{i2} \\ \vdots \\ \varepsilon_{in} \end{bmatrix} \quad \boldsymbol{\mu} = \begin{bmatrix} \mu_{1x} \\ \mu_{2x} \\ \vdots \\ \mu_{nx} \\ \mu_{1y} \\ \mu_{2y} \\ \vdots \\ \mu_{ny} \end{bmatrix}$$

Con lo cual, mediante una generalización del modelo (2.3), es plausible demostrar que:

$$\mathbf{u}_i = h_i \mathbf{I}_{2n} \boldsymbol{\mu} + g_i (\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu} + \mathbf{t}_i \otimes \mathbf{1}_n + \boldsymbol{\varepsilon}_i \quad (2.6)$$

Donde $\mathbf{1}_n = [1 \ 1 \ \dots \ 1]^T$ es un vector de n componentes todas iguales a la unidad. Conociendo ya las distribuciones de \mathbf{u}_{ij} , h_i , g_i , \mathbf{t}_i y $\boldsymbol{\varepsilon}_{ij}$ y siguiendo un procedimiento similar que el usado para (2.5), se puede verificar también que $\mathbf{u}_i \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 V(\boldsymbol{\mu}))$.

El vector $\boldsymbol{\mu}$ puede ser fácilmente estimado con el método de los momentos, como el promedio de los vectores \mathbf{u}_i :

$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^N \mathbf{u}_i \quad (2.7)$$

Hallando $\sigma^2 V(\boldsymbol{\mu})$, es decir la varianza de \mathbf{u}_i :

$$\text{Var}(\mathbf{u}_i) = \text{Var}(h_i\boldsymbol{\mu} + g_i(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu} + \mathbf{t}_i\otimes\mathbf{1} + \boldsymbol{\varepsilon}_i)$$

Se puede probar que $\text{Var}(\mathbf{X}\otimes\mathbf{a}) = \text{Var}(\mathbf{X})\otimes(\mathbf{a}\mathbf{a}^T)$, con lo cual se puede demostrar que:

$$\text{Var}(\mathbf{u}_i) = \sigma^2\mathbf{V}(\boldsymbol{\mu}) \quad (2.8)$$

Donde:

$$\begin{aligned} \mathbf{V}(\boldsymbol{\mu}) = & \sigma_h^2\boldsymbol{\mu}\boldsymbol{\mu}^T + \sigma_g^2[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}][(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T \\ & + \sigma_{hg}[\boldsymbol{\mu}[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T + [(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]\boldsymbol{\mu}^T] \\ & + \tau_1^2(\mathbf{A}_1\otimes\mathbf{1}\mathbf{1}^T) + \tau_2^2(\mathbf{A}_2\otimes\mathbf{1}\mathbf{1}^T) + \tau_{12}(\mathbf{A}_{12}\otimes\mathbf{1}\mathbf{1}^T) + \mathbf{I}_{2n} \end{aligned} \quad (2.9)$$

Con:

$$\begin{aligned} \mathbf{L} &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} & \mathbf{A}_2 &= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ \mathbf{A}_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} & \mathbf{A}_{12} &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{aligned}$$

La demostración completa se encuentra en el anexo I.

Para estimar la varianza (2.8) se requiere que se estimen las varianzas y covarianzas de los efectos aleatorios y del error, i.e: $\sigma^2, \sigma_h^2, \sigma_g^2, \sigma_{hg}, \tau_1^2, \tau_2^2, \tau_{12}$; para lo cual se usará el método de máxima verosimilitud computacionalmente, los valores iniciales que se requieren serán estimados con el método de VLS en la siguiente sección.

2.3 Método VLS para el modelo de Procrustes

Como se describió en la sección 1.2.5, el método de VLS sirve para estimar las varianzas de los efectos aleatorios en un modelo lineal; se aplicará este método para

estimar las varianzas de h_i, g_i, t_i y de los errores aleatorios. Acorde con la definición de VLS, tenemos que hallar Σ de tal forma que se minimice la expresión:

$$\sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i - \Sigma)^2 \quad (2.10)$$

La varianza de \mathbf{u}_i está dada en (2.8) con lo cual nuestro problema es:

$$\min \sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i - \sigma^2 \mathbf{V}(\boldsymbol{\mu}))^2 \quad (2.11)$$

Donde $\mathbf{V}(\boldsymbol{\mu})$ viene dada por (2.9). Definiendo el vector $\boldsymbol{\theta} = (\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6, \gamma_7)^T$ mediante la reparametrización:

$$\begin{aligned} \gamma_1 &= \sigma^2 \sigma_h^2 \\ \gamma_2 &= \sigma^2 \sigma_g^2 \\ \gamma_3 &= \sigma^2 \sigma_{hg} \\ \gamma_4 &= \sigma^2 \tau_1^2 \\ \gamma_5 &= \sigma^2 \tau_2^2 \\ \gamma_6 &= \sigma^2 \tau_{12} \\ \gamma_7 &= \sigma^2 \end{aligned} \quad (2.12)$$

Se demuestra en el anexo II que la expresión (2.11) con los nuevos parámetros (2.12) es equivalente a una forma cuadrática en función de los mismos, con lo cual el problema se traduce en hallar el vector $\boldsymbol{\theta}$ que minimice la expresión:

$$N\boldsymbol{\theta}^T \mathbf{M} \boldsymbol{\theta} - \mathbf{m}^T \boldsymbol{\theta} \quad (2.13)$$

Sujeto a las restricciones:

$$\begin{aligned} \gamma_k &\geq 0 \quad , k = \{1,2,4,5,7\} \\ \gamma_{k-2}\gamma_{k-1} - \gamma_k^2 &\geq 0 \quad , k = \{3,6\} \end{aligned} \quad (2.14)$$

El problema de minimizar formas cuadráticas sujetas a restricciones lineales tiene solución analítica, sin embargo cuando se agregan restricciones cuadráticas como en este caso, el nivel de dificultad de hallar alguna aumenta significativamente. Para tal efecto, se usarán métodos de optimización computacionales especiales para formas cuadráticas que se describirán en el siguiente capítulo. Dicho esto, el siguiente paso sería hallar los parámetros originales a partir de las componentes del vector obtenido θ mediante las ecuaciones (2.12). El resultado son estimadores VLS dados por el vector:

$$\mathbf{w} = \begin{bmatrix} \tilde{\sigma}^2 \\ \tilde{\sigma}_h^2 \\ \tilde{\sigma}_g^2 \\ \tilde{\sigma}_{hg} \\ \tilde{\tau}_1^2 \\ \tilde{\tau}_2^2 \\ \tilde{\tau}_{12} \end{bmatrix}$$

2.4 Estimación de la varianza de los efectos aleatorios por máxima verosimilitud

En el apartado III de la sección 1.2.2 se vio el uso del método de máxima verosimilitud para estimar los parámetros de media y varianza de la distribución

normal multivariada. Se usará el mismo método para estimar la matriz de varianzas y covarianzas del modelo (2.6). El vector media ya viene dado por (2.7).

Se sabe que $\mathbf{u}_i \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{V}(\boldsymbol{\mu}))$, entonces acorde a la ecuación (1.2) la función de densidad de \mathbf{u}_i viene dada por:

$$f(\mathbf{u}_i) = \frac{1}{(\sqrt{2\pi} \sigma^2)^{2n} \sqrt{|\mathbf{V}(\boldsymbol{\mu})|}} e^{-(\mathbf{u}_i - \boldsymbol{\mu})^T [\mathbf{V}(\boldsymbol{\mu})]^{-1} (\mathbf{u}_i - \boldsymbol{\mu}) / 2 \sigma^2} \quad (2.15)$$

Porque $|\sigma^2 \mathbf{V}(\boldsymbol{\mu})| = (\sigma^2)^{2n} |\mathbf{V}(\boldsymbol{\mu})|$. Se sabe que la matriz de varianzas y covarianzas de la distribución normal multivariada es simétrica y definida positiva, por lo tanto podemos aplicar la descomposición de Cholesky, la cual ayuda a realizar cálculos numéricos de una manera más precisa, a la vez que también permite simplificar la expresión (2.15). Se tratará entonces de hallar una matriz triangular superior \mathbf{R} tal que:

$$\mathbf{R}^T \mathbf{R} = \mathbf{V}(\boldsymbol{\mu}) \quad (2.16)$$

De donde:

$$\mathbf{R}^{-1} \mathbf{R}^{-T} = [\mathbf{V}(\boldsymbol{\mu})]^{-1} \quad (2.17)$$

Mediante las propiedades de los determinantes y logaritmos:

$$|\mathbf{V}(\boldsymbol{\mu})| = |\mathbf{R}^T| |\mathbf{R}| = |\mathbf{R}|^2 = \prod_{j=1}^{2n} r_{jj} \quad (2.18)$$

$$\ln |\mathbf{V}(\boldsymbol{\mu})| = \sum_{j=1}^{2n} \ln r_{jj}$$

La función de verosimilitud L de los vectores \mathbf{u}_i al suponer independencia en los mismos es el producto de las densidades de probabilidades. Maximizar esta función es equivalente a maximizar su logaritmo l , se tiene entonces:

$$L = \prod_{i=1}^N f(\mathbf{u}_i)$$

$$l = \ln L = \ln \prod_{i=1}^N f(\mathbf{u}_i) = \sum_{i=1}^N \ln f(\mathbf{u}_i)$$

Reemplazando (2.17) y (2.18) en (2.15) y luego en l y eliminando también el término constante que no afecta la maximización, se tiene:

$$l = \sum_{i=1}^N \left[-n \ln \sigma^2 - \sum_{j=1}^{2n} [\ln r_{jj}] - \frac{1}{2\sigma^2} \mathbf{r}_i^T \mathbf{r}_i \right] \quad (2.19)$$

Con:

$$\mathbf{r}_i = \mathbf{R}^{-T}(\mathbf{u}_i - \boldsymbol{\mu}) \quad (2.20)$$

La optimización de la función l se realizará computacionalmente con métodos numéricos determinados que se describirán en el siguiente capítulo, como ya se dijo los valores iniciales requeridos por estos métodos numéricos están dados por el vector \mathbf{w} descrito al final de la sección anterior; con lo que se hallarán los estimadores definitivos de $\sigma^2, \sigma_h^2, \sigma_g^2, \sigma_{hg}, \tau_1^2, \tau_2^2, \tau_{12}$. El resultado de este procedimiento será un vector \mathbf{v} dado por:

$$\mathbf{v} = \begin{bmatrix} \hat{\sigma}^2 \\ \hat{\sigma}_h^2 \\ \hat{\sigma}_g^2 \\ \hat{\sigma}_{hg} \\ \hat{t}_1^2 \\ \hat{t}_2^2 \\ \hat{t}_{12} \end{bmatrix}$$

Reemplazando los parámetros de la ecuación (2.9) por los correspondientes estimadores aquí obtenidos en las componentes dos a siete del vector \mathbf{v} , obtenemos a $\hat{\mathbf{V}}(\boldsymbol{\mu})$, el respectivo estimador de $\mathbf{V}(\boldsymbol{\mu})$. Por tanto la estimación de la varianza de \mathbf{u}_i es $\hat{\sigma}^2 \hat{\mathbf{V}}(\boldsymbol{\mu})$.

2.5 Reconocimiento facial con la prueba de pertenencia

Recapitulando el problema: se quiere desarrollar un modelo lineal que sea capaz de distinguir a un individuo de otro distinto, para lo cual hay N fotos del individuo y una $(N+1)$ -ésima foto que posiblemente pertenezca a él mismo o no; en cada foto se señalaron n características notables (puntos) cuyas coordenadas son los datos de entrada del modelo lineal. El resultado es el modelo (2.6); el promedio de dicho modelo se estimó con el método de los momentos mientras que la varianza se estimó en dos pasos: VLS y luego máxima verosimilitud. El problema de distinción entre sujetos (reconocimiento facial) se lo convirtió a un contraste de hipótesis, en la cual el promedio y varianza estimados sirven para calcular el estadístico de prueba y posteriormente el valor P de la prueba. El contraste de hipótesis justamente vendrá dado por la *prueba de pertenencia* el cual se describió en el marco teórico en la sección 1.2.3.

Las N fotos del primer individuo dadas por $\mathbf{u}_i; i = 1, 2, \dots, N$ tienen distribución normal multivariada como se mencionó anteriormente, i.e.: $\mathbf{u}_i \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{V}(\boldsymbol{\mu}))$, suponiendo que llega una $(N+1)$ -ésima foto cuyo vector respectivo es \mathbf{u}_{N+1} , el cual posee también distribución normal multivariada con parámetros distintos, i.e.: $\mathbf{u}_{N+1} \sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma})$. Nos interesa responder la pregunta ¿pertenece esta nueva foto al mismo individuo? Como se mencionó en el marco teórico, esto es equivalente a contrastar las hipótesis: $H_0: \boldsymbol{\mu} = \boldsymbol{\mu}_*$ vs $H_A: \boldsymbol{\mu} \neq \boldsymbol{\mu}_*$. Si la hipótesis nula fuese verdadera, se debe suponer además que la varianza de la $(N+1)$ -ésima foto es la misma que para las N primeras (ya que estamos suponiendo que pertenece al mismo individuo), en otras palabras bajo la hipótesis nula se cumple que $\mathbf{u}_{N+1} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{V}(\boldsymbol{\mu}))$. Entonces, el estadístico de prueba dado por la ecuación (1.13) para este caso y con los estimadores que ya se hallaron, es:

$$ep = \left(\frac{N}{N+1} \right) (\hat{\boldsymbol{\mu}} - \mathbf{u}_{N+1})^T \frac{1}{\hat{\sigma}^2} [\hat{\mathbf{V}}(\boldsymbol{\mu})]^{-1} (\hat{\boldsymbol{\mu}} - \mathbf{u}_{N+1}) \sim \chi_{(2n)}^2 \quad (2.21)$$

Nótese que no se están usando los verdaderos parámetros $\boldsymbol{\mu}$ y $\sigma^2 \mathbf{V}(\boldsymbol{\mu})$; si bien (2.21) se cumple cuando usamos $\hat{\boldsymbol{\mu}} = \bar{\mathbf{x}}$ en lugar de $\boldsymbol{\mu}$, al usar un estimador de $\sigma^2 \mathbf{V}(\boldsymbol{\mu})$ en lugar de la propia $\sigma^2 \mathbf{V}(\boldsymbol{\mu})$ el contraste de hipótesis pierde algo de potencia. Si α es el nivel de significancia, entonces se dice con $(1 - \alpha) * 100\%$ de confianza que la foto $N+1$ **no** pertenece al individuo al cual pertenecen las N primeras (en otras palabras, esta foto pertenece a un individuo distinto), si se cumple que:

$$ep > \chi_{(2n, \alpha)}^2$$

Donde $\chi_{(2n,\alpha)}^2$ es el percentil $1 - \alpha$ de la distribución ji-cuadrado con $2n$ grados de libertad. O en otras palabras, si la probabilidad de que la correspondiente variable ji-cuadrado sea mayor que este estadístico es menor que α (valor P), es decir:

$$P(\chi_{(2n)}^2 \geq ep) < \alpha \quad (2.22)$$

Si esta probabilidad es mayor o igual que α entonces decimos que no hay evidencia estadística de que la $(N+1)$ -ésima foto pertenezca a otro individuo (es decir si $ep \leq \chi_{(2n,\alpha)}^2$).

Algunos autores llaman región de incertidumbre a la comprendida entre 0.05 y 0.1, si el valor P se encuentra en esta región entonces diríamos que no hay suficiente evidencia estadística para rechazar o no rechazar H_0 . Según este enfoque se rechaza H_0 si P es menor a 0.05 y no se la rechaza si es mayor a 0.1; no obstante, se realizarán las conclusiones en la manera descrita en el párrafo anterior.

CAPÍTULO 3

3. Programación del modelo de Procrustes de efectos aleatorios

3.1 Introducción

En este capítulo se programará el modelo lineal mixto desarrollado en el capítulo anterior en RStudio, que es una interfaz gráfica del ya mencionado software estadístico R (16). Serán dos individuos los sujetos de estudio. Comenzaremos definiendo los valores de N y n como ocho y tres respectivamente, es decir habrá ocho fotos por individuo y tres puntos de referencia que se tomarán en cada foto. Los tres puntos serán el iris de los ojos y la punta de la nariz. El código usado está en el anexo III. También se hace mención de las librerías de R utilizadas a lo largo del proceso. Luego se aumentará la cantidad de puntos de referencia a cinco y siete.

3.2 Conversión de las imágenes en datos

En la sección 2.1 se describió la manera en que las imágenes se convierten en vectores que serán los datos de entrada para el modelo mixto, para lo cual se hizo uso de la función `locator` la cual viene incluida por defecto en R en la librería `base`, y

funciona colocando un sistema de coordenadas en la parte inferior izquierda de cada imagen y asumiendo como positivas las direcciones hacia arriba y hacia la derecha; el resultado son tres puntos con coordenadas X e Y que serán almacenadas en un vector con seis componentes, como se mostró en la figura 3. La importación de las imágenes a R se realiza gracias al paquete `EImage` (17).

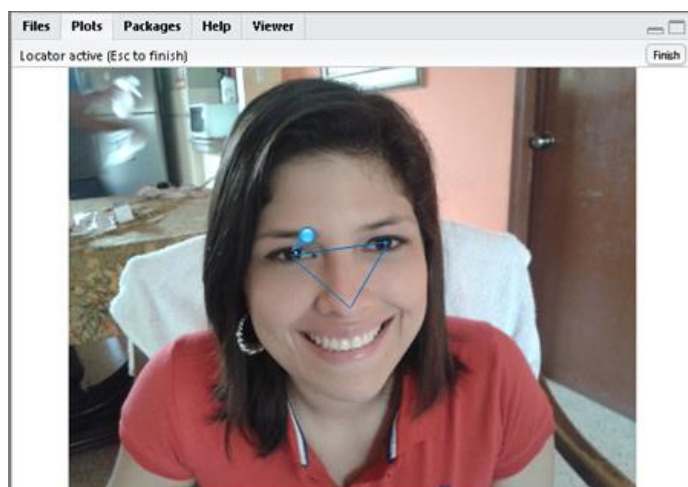


Ilustración 6. Uso de la función `locator` para señalar un punto en la imagen

El resultado de aplicar esta operación en las ocho imágenes de ambos individuos (clusters) es un objeto de tipo lista. Mediante una función, la cual se detalla en el anexo III; se convertirá esta lista en una matriz de dimensión $2n \times N = 6 \times 8$; habrá una matriz para cada individuo. Las columnas serán vectores que representen una determinada foto del individuo, las tres primeras filas representarán las abscisas de cada punto y las tres últimas las ordenadas, como se mostró en la sección 2.1 en la Ilustración 3. El resultado son dos matrices, una para el individuo 1 y otra para el individuo 2, que se muestran a continuación redondeadas a cuatro decimales.

```

> round(datos_Rebeca,4)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
x1 0.3472 0.4377 0.3958 0.3494 0.4024 0.3826 0.4068 0.3958 0.4333
x2 0.5238 0.5923 0.5812 0.5371 0.5967 0.5636 0.5592 0.5724 0.5790
x3 0.4488 0.5172 0.4841 0.4510 0.5018 0.4775 0.4775 0.4510 0.5128
y1 0.5684 0.6511 0.7172 0.6787 0.5187 0.5518 0.6897 0.6070 0.6566
y2 0.6180 0.6787 0.7172 0.7007 0.5132 0.5739 0.7007 0.5739 0.6787
y3 0.3258 0.5022 0.4746 0.4691 0.3423 0.3147 0.4857 0.3202 0.4746
> round(datos_Marlon,4)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
x1 0.3936 0.3958 0.3649 0.3450 0.3318 0.3759 0.3848 0.3914 0.3870
x2 0.5349 0.5283 0.5128 0.4907 0.4819 0.5349 0.5437 0.5283 0.5194
x3 0.4709 0.4709 0.4422 0.4179 0.4113 0.4709 0.4731 0.4664 0.4598
y1 0.7172 0.6456 0.4250 0.5849 0.6897 0.6456 0.6015 0.4746 0.4526
y2 0.7007 0.6621 0.4471 0.5904 0.6897 0.6731 0.6235 0.4801 0.4912
y3 0.5132 0.4416 0.2155 0.3699 0.4967 0.3864 0.3258 0.2541 0.2982

```

Nótese que hay ocho observaciones (fotos) por cada individuo, por lo cual el estadístico (2.21) tiene distribución ji-cuadrado no de manera exacta, sino aproximada debido a esta razón y al hecho de usar un estimador de la matriz $V(\mu)$. Las coordenadas de los ocho triángulos que se originan para cada individuo se muestran en los siguientes gráficos.

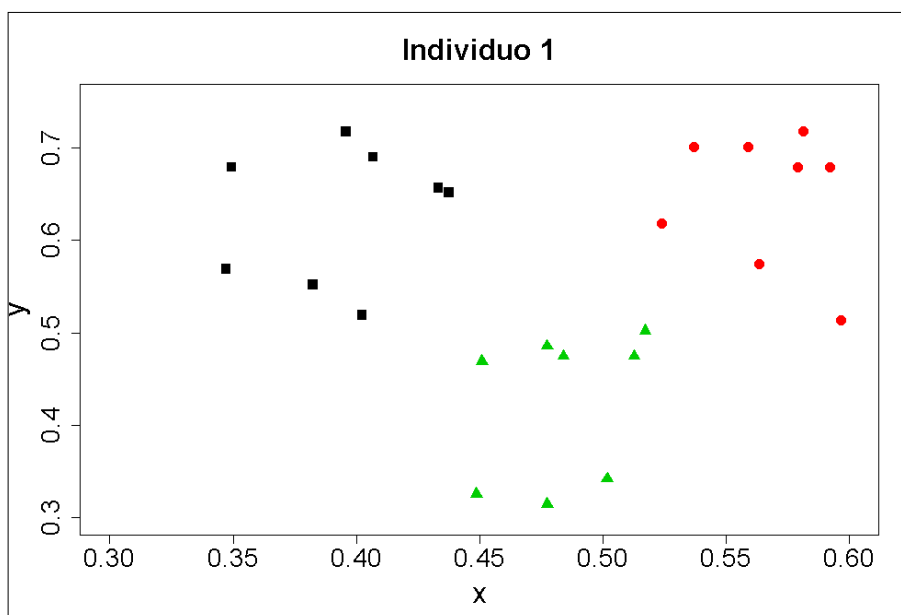


Gráfico 1. Gráfico de tres puntos de referencia correspondiente al individuo uno

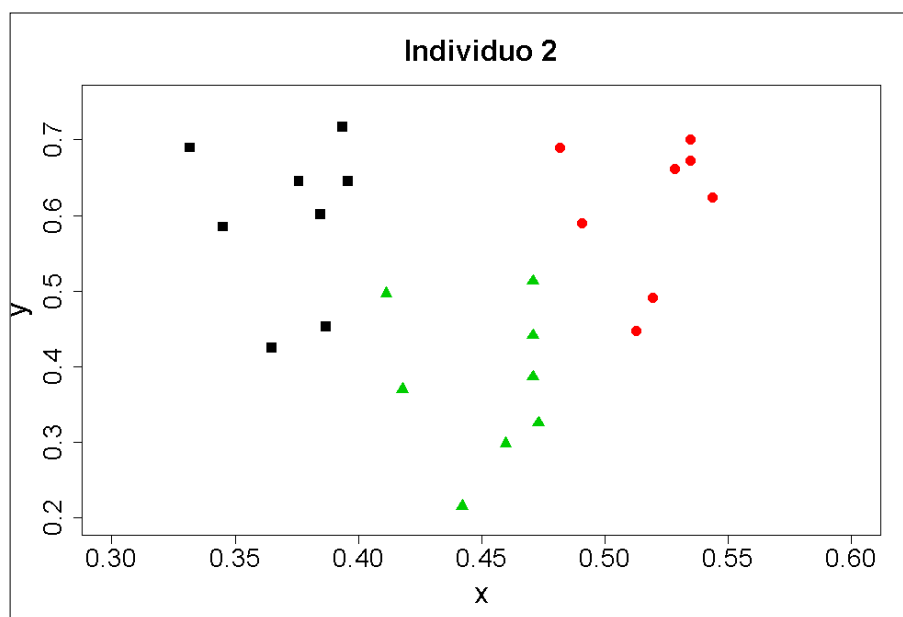


Gráfico 2. Gráfico de tres puntos de referencia correspondiente al individuo dos

Cada color representa un vértice distinto del triángulo, se aprecia una dispersión un poco elevada entre vértices correspondientes. Los cuadrados negros corresponden al ojo derecho, los círculos rojos corresponden al ojo izquierdo y los triángulos verdes corresponden a la punta de la nariz. Cada columna de las matrices es un vector u_i al que ajustaremos el modelo (2.6) por lo cual hay que estimar sus parámetros de media y varianza dados por (2.7) y (2.8). Hay que decidir ahora si se ajustará el modelo para las fotos del individuo uno y se compararán las del individuo dos o viceversa. Ambos enfoques serán considerados.

3.3 Ajuste del modelo de Procrustes con tres puntos de referencia

Se ajustará el modelo (2.6) al individuo uno primeramente, se procederá entonces a estimar los parámetros de media y varianza. La media será el promedio de los

vectores \mathbf{u}_i , es decir el promedio por filas para la matriz de datos de este individuo.

El vector promedio obtenido es:

```
> round(mu,4)
  x1    x2    x3    y1    y2    y3
0.3944 0.5666 0.4838 0.6290 0.6476 0.4236
```

Este vector contiene las coordenadas del estimador del polígono promedio desconocido, el cual luce de la siguiente manera con respecto de los polígonos correspondientes a las fotos del individuo uno:

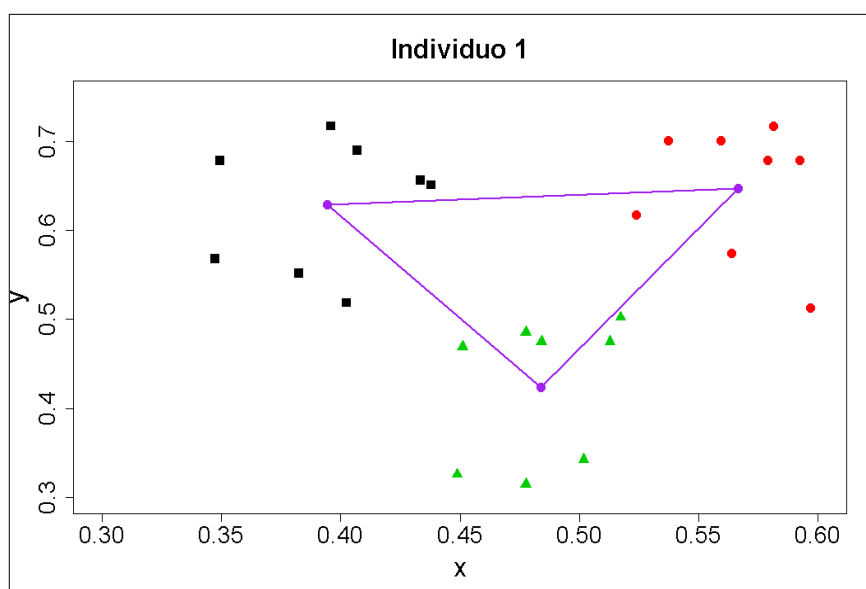


Gráfico 3. Estimación del polígono promedio para el individuo uno

La varianza será estimada en dos pasos como se dijo antes: VLS y luego máxima verosimilitud. En ambos casos se requiere de optimización sujeta a restricciones, las

restricciones vienen dadas por (2.14), que son propiedades matemáticas de la varianza y covarianza de variables aleatorias detalladas en el anexo II.

Para hallar los estimadores VLS se ha escrito una función que se encuentra en el anexo III, en la cual se hace uso de la función `constrOptim.nl` que se incluye en la librería `alabama` (18), la misma que se usa para realizar la optimización requerida de la sumatoria de las trazas cuadráticas que comprende VLS. El método usado por esta función es un método de barreras, concretamente el del Lagrangiano Aumentado (Augmented Lagrangian Adaptive Barrier Minimization Algorithm: ALABMA por sus siglas en inglés). El resultado es un vector con siete estimadores que representan las estimaciones de las varianzas de los efectos aleatorios $\mathbf{w} = [\tilde{\sigma}^2, \tilde{\sigma}_h^2, \tilde{\sigma}_g^2, \tilde{\sigma}_{hg}, \tilde{\tau}_1^2, \tilde{\tau}_2^2, \tilde{\tau}_{12}^2]^T$. El resultado a cuatro decimales es:

```
> round(vp,4)
[1] 0.0003 0.3045 2.8698 -0.9349 5.2801 17.5231 4.9937
```

Para la siguiente fase que corresponde a la estimación de la varianza de los efectos aleatorios por máxima verosimilitud se usará también la función `constrOptim.nl` del paquete `alabama` (18). Los valores iniciales para esta etapa son los que se muestran arriba hallados con VLS. Previamente se crean las matrices de constantes requeridas que se mostraron en (2.9). El valor de la función logaritmo de la verosimilitud en este punto inicial es:

```
> logver(datos_Rebeca,mu,vp)
[1] 147.4429
```

Luego de maximizar la función de verosimilitud el resultado obtenido es un vector de la forma $\mathbf{v} = [\hat{\sigma}^2, \hat{\sigma}_h^2, \hat{\sigma}_g^2, \hat{\sigma}_{hg}, \hat{t}_1^2, \hat{t}_2^2, \hat{t}_{12}^2]^T$ que maximiza dicha función, para el caso de este individuo este vector es:

```
> round(resp$par,4)
[1] 0.0002 0.3045 2.8698 -0.9349 5.2801 17.5231 4.9937
```

El cual es un resultado que difiere obtenido antes con VLS en la primera componente. La varianza del error (componente 1) es pequeña comparada con la varianza de los efectos aleatorios, lo cual es uno de los objetivos del modelo de Procrustes. Las componentes dos y tres de este vector son más grandes que la primera, y reflejan una variación más alta en rotación y reescalamiento respecto del polígono promedio desconocido que el ocasionado por señalar las coordenadas manualmente; a su vez estas variaciones tienen correlación negativa (componente 4). La dispersión del desplazamiento de las fotos con respecto a la foto que contiene al polígono promedio desconocido en sentido horizontal (componente 5) es bastante más pequeño que el correspondiente desplazamiento en sentido vertical (componente 6) y además tienen correlación positiva (componente 7). El valor óptimo de la función logaritmo de la verosimilitud en este punto óptimo es:

```
> logver(datos_Rebeca,mu,resp$par)
[1] 147.7481
```

El cual difiere muy poco del valor en el punto inicial, lo cual indica que el método VLS proporcionó estimadores muy buenos.

3.4 Uso de la prueba de pertenencia y reconocimiento facial

Con los estimadores ya obtenidos de la media y la varianza, hay que calcular el respectivo estadístico de prueba dado por la ecuación (2.21). Como será calculado para el individuo 1, se comparará cada polígono de las fotos del segundo individuo con la estimación del polígono promedio desconocido del individuo 1. En la mencionada ecuación (2.21), cada vector \mathbf{u}_{N+1} representa una columna de la matriz de datos del individuo 2. Habrá que calcular entonces ocho estadísticos de prueba, los cuales tienen distribución ji-cuadrado con 6 grados de libertad. Trabajando con un nivel de significancia $\alpha=0.05$ vamos a calcular el valor P dado por la probabilidad (2.22). Para una determinada foto, el valor P puede estar en cualquiera de los dos casos siguientes:

- Si es menor que 0.05, entonces esta foto no tiene la misma distribución que las del individuo uno. Por tanto pertenece a un individuo distinto.
- Si es mayor o igual que 0.05 ocurre lo contrario, se concluye que no existe evidencia de que esta foto tenga una distribución distinta a las del individuo uno, por lo que decimos que pertenece a este mismo individuo.

A priori, de manera obvia sabemos que ambos individuos que se comparan son distintos, por tanto lo ideal es que el valor P para cada foto del individuo dos sea menor que 0.05; pero en la práctica ningún modelo de reconocimiento facial tiene 100% de efectividad. El objetivo será entonces, evaluar cuán bueno es el modelo. Luego también se compararán las fotos del primer individuo consigo mismo.

El vector de ocho componentes que contiene el valor P para cada foto del individuo dos es:

```
> round(valorp_Marlon,4)
[1] 0.0602 0.1353 0.0237 0.1278 0.0335 0.0265 0.0039 0.0065
```

Cinco componentes de este vector que son la tres, cinco, seis, siete y ocho tienen valor menor que 0.05, por lo cual mediante estas fotos se ha concluido que no pertenecen al individuo uno. Mediante las fotos uno, dos y cuatro el modelo se equivoca y concluye que éstas pertenecen al mismo individuo uno. Ahora calculando el valor P pero en las mismas fotos del primer individuo:

```
> round(valorp_Rebeca2,4)
[1] 0.0204 0.0411 0.1266 0.1270 0.0258 0.1188 0.1110 0.1346
```

El modelo de efectos aleatorios ha concluido correctamente que estas fotos pertenecen al mismo individuo uno en cinco de ocho ocasiones. En las tres restantes ocasiones se equivocó. De manera similar se analizará el comportamiento del modelo esta vez ajustándolo en el individuo dos y comparándolo con el individuo uno.

3.5 Aplicación del modelo Procrustes en el segundo individuo

Nuevamente, como en el caso del individuo uno, sabiendo que los vectores \mathbf{u}_i siguen el modelo (2.6) y tienen distribución normal multivariada, procedemos a estimar en primer lugar los parámetros de la misma, para después calcular el estadístico de prueba y el valor P. El proceso es idéntico al anterior por lo cual no se incurrirá en muchos detalles.

El vector promedio que representa la estimación del polígono promedio desconocido es:

```
> round(mu2,4)
  x1    x2    x3    y1    y2    y3
0.3723 0.5183 0.4521 0.5953 0.6097 0.3809
```

El cual, graficado con los demás puntos de las fotos respectivas, luce de la siguiente manera:

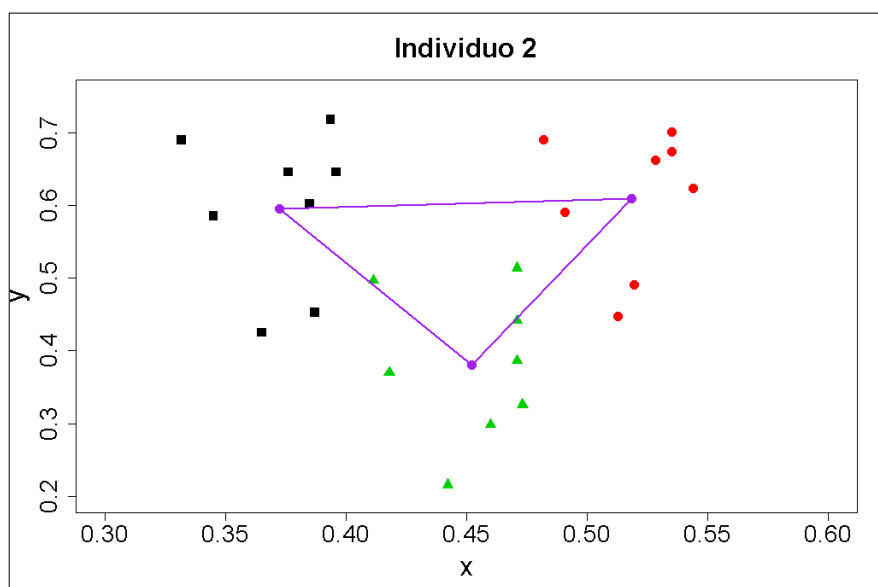


Gráfico 4. Estimación del polígono promedio para el individuo dos

La estimación de la varianza de los efectos aleatorios mediante VLS es el vector dado por:

```
> round(vp2,4)
[1] 0.0003 19.6506 2.3382 -6.7784 2.0133 16.5449 -2.5075
```

El valor de la función logaritmo de la verosimilitud en este punto es:

```
> logver(datos_Marlon,mu2,vp2)
[1] 142.1933
```

El siguiente paso que comprende la maximización de esta función, proporciona el vector de estimación de varianza de efectos aleatorios, el cual es:

```
> round(vp_opt2,4)
[1] 0.0002 19.6506 2.3382 -6.7784 2.0133 16.5449 -2.5075
```

Es un vector muy similar al obtenido inicialmente, difieren solamente en la primera componente. La varianza del error es bien pequeña si se la compara con la varianza de los efectos aleatorios; por otro, la rotación y el reescalamiento del polígono promedio tienen varianzas muy distintas y además tienen correlación negativa. La dispersión del desplazamiento horizontal respecto del polígono promedio es mucho más pequeña que la del desplazamiento vertical y tienen correlación negativa. El valor óptimo de la función en este punto no difiere mucho del que tenía en el punto inicial:

```
> logver(datos_Marlon,mu2,resp2$par)
[1] 145.9458
```

Ahora con estas estimaciones se procede a calcular el estadístico de prueba para cada fotografía del individuo uno, las cuales se compararán con las de este individuo. El vector que contiene los valores P correspondientes es el siguiente:

```
> round(valorp_Rebeca,4)
[1] 0.0232 0.0002 0.1161 0.1226 0.0020 0.1256 0.1231 0.0288
```

Con las fotos uno, dos, cinco y ocho que tienen un valor P menor que 0.05, el modelo concluye correctamente que estas no pertenecen al individuo dos; no es el caso de las fotos tres, cuatro, seis y siete ya que con las mismas concluye exactamente lo opuesto, es decir que pertenecen al individuo dos cuando en realidad son del primero. Calculando el valor P en las mismas fotos del segundo individuo:

```
> round(valorp_Marlon2,4)
[1] 0.0374 0.1139 0.0906 0.0460 0.1036 0.1057 0.0306 0.0495
```

El modelo ha concluido correctamente que estas fotos pertenecen al mismo individuo dos en solo cuatro de ocho ocasiones, en las restantes cuatro se equivoca.

Hasta ahora se ha analizado el comportamiento del modelo de efectos aleatorios teniendo en cuenta tres puntos notables (los ojos y la punta de la nariz), pero ¿qué ocurriría si se toman cinco puntos notables como referencia, o siete? Este será el tema abordado en las secciones siguientes.

3.6 El modelo de Procrustes con cinco puntos de referencia.

En este nuevo enfoque se observará el comportamiento del modelo de efectos aleatorios con cinco características notables que definirán cinco puntos de referencia. Los puntos que se considerarán son el iris de ambos ojos, los extremos izquierdo y derecho de la boca y el punto más inferior de la misma, de la siguiente manera:

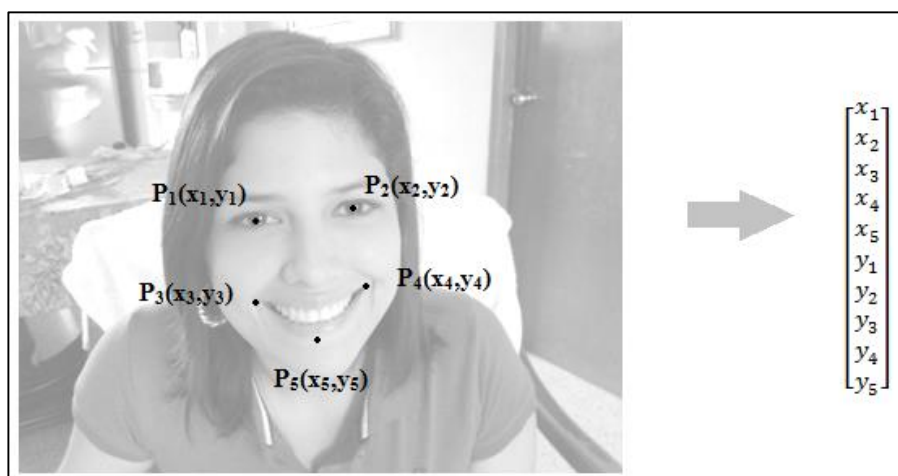


Ilustración 7. Los cinco puntos de referencia para el modelo Procrustes

Nótese que al haber cinco puntos, hay diez variables aleatorias normales por lo cual las estimaciones de los parámetros y los valores P que se obtengan pierden confiabilidad al ser obtenidos con ocho observaciones. Al igual que con el caso anterior se ajustará el modelo tanto en el individuo uno como en el dos para observar el comportamiento del modelo. Las matrices de coordenadas de datos obtenidas son:

```
> round(datos_Rebeca,4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3472 0.4355 0.3958 0.3494 0.4068 0.3803 0.4046 0.4311
x2 0.5216 0.5857 0.5790 0.5327 0.5945 0.5614 0.5570 0.5746
x3 0.3605 0.4554 0.3715 0.3472 0.4245 0.4090 0.4223 0.4267
x4 0.5327 0.5790 0.5790 0.5393 0.5680 0.5548 0.5393 0.5746
x5 0.4532 0.5238 0.4797 0.4510 0.5040 0.4885 0.4819 0.5084
y1 0.5684 0.6511 0.7117 0.6787 0.5132 0.5353 0.6897 0.6511
y2 0.6180 0.6731 0.7117 0.6952 0.5132 0.5573 0.7007 0.6731
y3 0.2045 0.3202 0.3368 0.3478 0.0997 0.0887 0.3258 0.3478
y4 0.2706 0.3533 0.3368 0.3588 0.1217 0.1217 0.3313 0.3699
y5 0.0556 0.2596 0.1493 0.1714 0.0170 0.0060 0.2486 0.1659
> round(datos_Marlon,4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3936 0.3936 0.3605 0.3406 0.3318 0.3737 0.3826 0.3870
x2 0.5327 0.5305 0.5062 0.4863 0.4819 0.5327 0.5459 0.5172
x3 0.3914 0.4201 0.3583 0.3539 0.3516 0.4068 0.4157 0.3958
x4 0.5327 0.5283 0.5128 0.4907 0.4731 0.5261 0.5349 0.5261
x5 0.4598 0.4753 0.4400 0.4245 0.4201 0.4664 0.4753 0.4664
y1 0.7172 0.6401 0.4360 0.5904 0.6842 0.6290 0.6015 0.4581
y2 0.7007 0.6621 0.4360 0.5959 0.6897 0.6621 0.6125 0.4912
y3 0.4030 0.2927 0.1273 0.2486 0.3478 0.2265 0.1714 0.1714
y4 0.3864 0.3092 0.1217 0.2596 0.3699 0.2541 0.1769 0.1934
y5 0.2816 0.2265 0.0115 0.1603 0.2651 0.1603 0.0887 0.0556
```

Los polígonos que se forman en este caso son pentágonos cuyos vértices lucen de la siguiente manera:

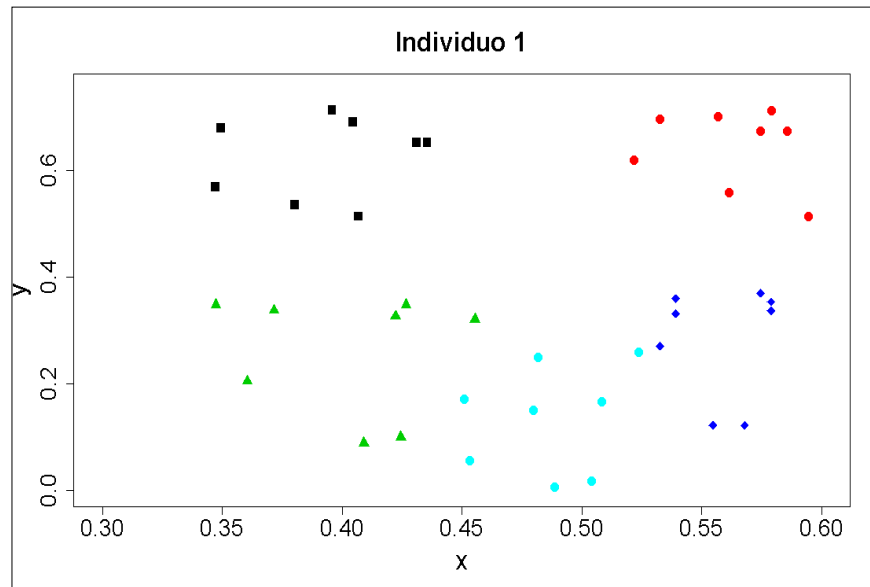


Gráfico 5. Gráfico de puntos correspondiente al individuo uno con cinco puntos de referencia

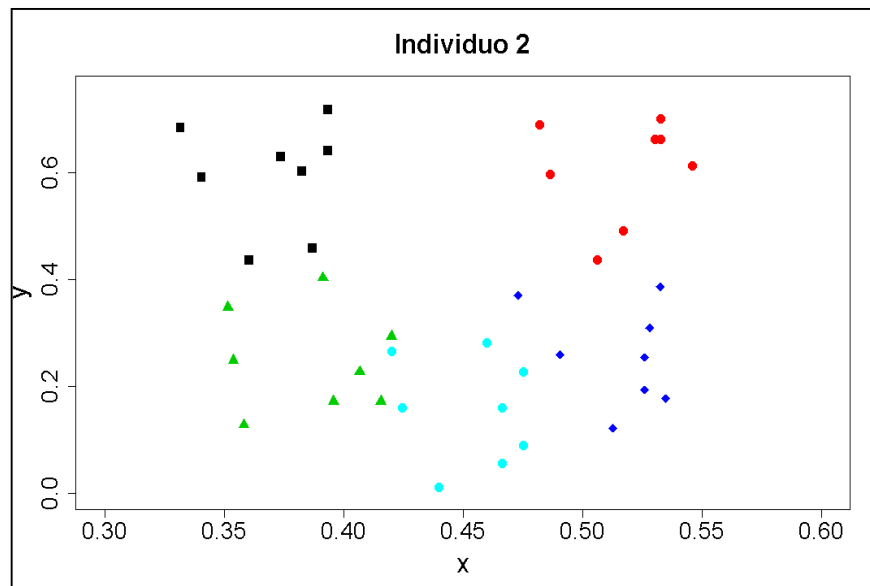


Gráfico 6. Gráfico de puntos correspondiente al individuo dos con cinco puntos de referencia

Cada punto de un color determinado representa un mismo punto notable en las distintas fotografías; aunque no se distingue de manera clara el patrón pentagonal, es debido a la variación de expresiones faciales de la cara. Los cuadrados negros y círculos rojos corresponden a los ojos, los triángulos verdes y rombos azules corresponden a los extremos laterales de la boca y los círculos celestes representan el punto más bajo de la misma.

3.6.1 Ajustando el modelo en el individuo 1

La media es el promedio por filas, lo cual da un vector de diez localidades que es:

```
> round(mu,4)
  x1    x2    x3    x4    x5    y1    y2    y3    y4    y5
0.3939 0.5633 0.4021 0.5583 0.4863 0.6249 0.6428 0.2589 0.2830 0.1342
```

El cual representa las coordenadas del polígono promedio desconocido que luce así:

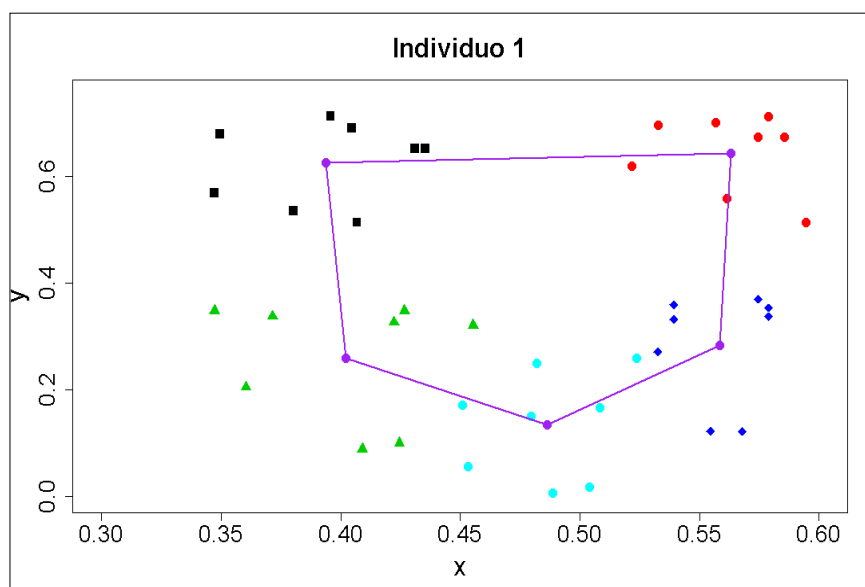


Gráfico 7. Estimación del polígono medio para el individuo uno con cinco puntos de referencia

La varianza estimada de los efectos aleatorios mediante VLS está dada por el vector:

```
> round(vp,4)
[1] 0.0006 0.2788 0.6344 0.4206 0.6397 10.8580 0.3014
```

Y el respectivo valor de la función logaritmo de la verosimilitud en este punto es:

```
> logver(datos_Rebeca,mu,vp)
[1] 235.6137
```

El estimador de máxima verosimilitud de la varianza es:

```
> round(vp_opt,4)
[1] 0.0005 0.1832 0.2422 -0.2107 0.8266 10.9863 0.2234
```

La varianza del error es pequeña con respecto a la varianza de los efectos aleatorios, la rotación y el reescalamiento del polígono promedio tienen varianzas similares y además tienen correlación negativa. La dispersión del desplazamiento horizontal respecto del polígono promedio es bastante más pequeña que la del desplazamiento vertical y tienen covarianza positiva. El valor óptimo de la función logaritmo de la verosimilitud en este punto es:

```
> logver(datos_Rebeca,mu,resp$par)
[1] 237.7163
```

Luego se calculan los estadísticos de prueba y los valores P comparando con las fotos del individuo dos, dados por:

```
> round(valorp_Marlon,4)
[1] 0.0495 0.0821 0.0270 0.0649 0.0193 0.0970 0.0698 0.0271
```

Mediante las fotos uno, tres, cinco y ocho el modelo concluye que estas pertenecen a alguien distinto del individuo uno; con las restantes cuatro fotos el modelo concluye que pertenecen al mismo individuo uno, es decir se equivoca estas cuatro ocasiones.

Calculando los valores P en las mismas fotos del individuo uno se tiene:

```
> round(valorp_Rebeca2,4)
[1] 0.0976 0.0324 0.0940 0.0967 0.0977 0.0450 0.0958 0.0923
```

El modelo ha concluido correctamente que las fotos pertenecen al mismo individuo uno en seis ocasiones y ha concluido que pertenecen a alguien distinto en dos ocasiones, es decir se ha equivocado dos veces, con las fotografías dos y seis.

3.6.2 Ajustando el modelo en el individuo 2

Al igual que antes, lo primero que se hace estimar los parámetros media y varianza.

El vector estimador del polígono promedio desconocido es:

```
> round(mu2,4)
  x1    x2    x3    x4    x5    y1    y2    y3    y4    y5
0.3704 0.5167 0.3867 0.5156 0.4535 0.5946 0.6063 0.2486 0.2589 0.1562
```

Este polígono promedio luce de la siguiente manera:

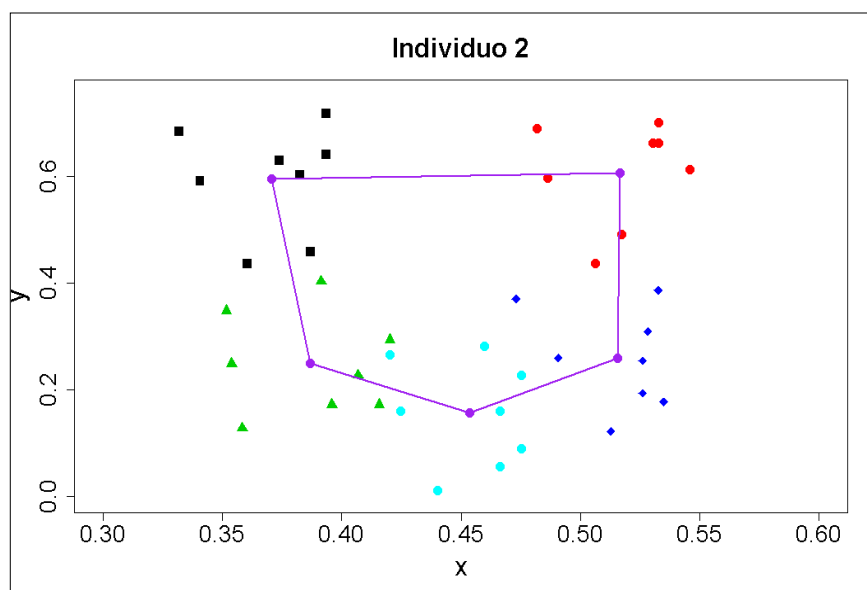


Gráfico 8. Estimación del polígono promedio desconocido para el individuo dos con cinco puntos de referencia

La varianza estimada de los efectos aleatorios mediante VLS está dada por el vector:

```
> round(vp2,4)
[1] 0.0004 5.7850 0.0261 0.3878 0.0641 20.9174 -1.1583
```

Y el respectivo valor de la función logaritmo de la verosimilitud en este punto es:

```
> logver(datos_Marlon,mu2, vp2)
[1] 256.3299
```

El estimador de la varianza que maximiza la función de verosimilitud es:

```
> round(vp_opt2,4)
[1] 0.0002 6.9062 0.0185 -0.3554 0.2562 21.4704 -0.8871
```

La varianza del error es pequeña con respecto a la varianza de los efectos aleatorios. Por otro lado, la rotación y el reescalamiento del polígono promedio tienen varianzas muy distintas y tienen correlación negativa. La dispersión del desplazamiento horizontal respecto del polígono promedio es mucho más pequeña que la del desplazamiento vertical y tienen correlación negativa también. El valor óptimo de la función en este punto es ligeramente mayor que el anterior:

```
> logver(datos_Marlon,mu2,resp2$par)
[1] 260.7144
```

Al igual que con el individuo anterior, hay un aumento más grande en la función logaritmo de verosimilitud que con tres puntos de referencia, ocasionado porque el estimador de la varianza de los efectos aleatorios de máxima verosimilitud ya no difiere en solo una componente del estimador VLS.

Los valores P obtenidos al comparar el modelo ajustado en el individuo uno con las fotos del individuo dos, son:

```
> round(valorp_Rebeca,4)
[1] 0.0000 0.0000 0.0000 0.0001 0.0332 0.0001 0.0925 0.0000
```

Observando estos valores, mediante todas las fotos excepto la siete que tienen valor P mayor que 0.05, el modelo concluye que efectivamente pertenecen a un individuo distinto; con la foto siete del primer individuo concluye exactamente lo contrario. Nótese que con respecto al caso del modelo con tres puntos de referencia ajustado en el individuo dos, ha aumentado la cantidad de aciertos significativamente ya que ahora acertó en siete de ocho ocasiones. Se puede hablar entonces de una mejoría en

el modelo al usar cinco puntos de referencia en lugar de tres, para el caso del individuo dos.

Ahora bien, comparando las mismas fotos del segundo individuo se obtienen los siguientes valores P:

```
> round(valorp_Marlon2,4)
[1] 0.0392 0.0937 0.0914 0.0219 0.0828 0.0674 0.0035 0.0303
```

El modelo de Procrustes ha acertado en cuatro de ocho ocasiones al momento de comparar las fotos del individuo consigo mismo concluyendo que son fotos de él mismo, en los otros casos donde el valor P es menor que 0.05 se ha equivocado. Ahora conviene analizar ¿qué pasaría con siete puntos de referencia?

3.7 El modelo de Procrustes con siete puntos de referencia.

En esta sección se observará el comportamiento del modelo de Procrustes de efectos aleatorios con siete características notables que definirán siete puntos de referencia. Estos puntos son el iris de los ojos, los hoyos de la nariz, los extremos izquierdo y derecho de la boca y el punto más inferior en la misma, así:

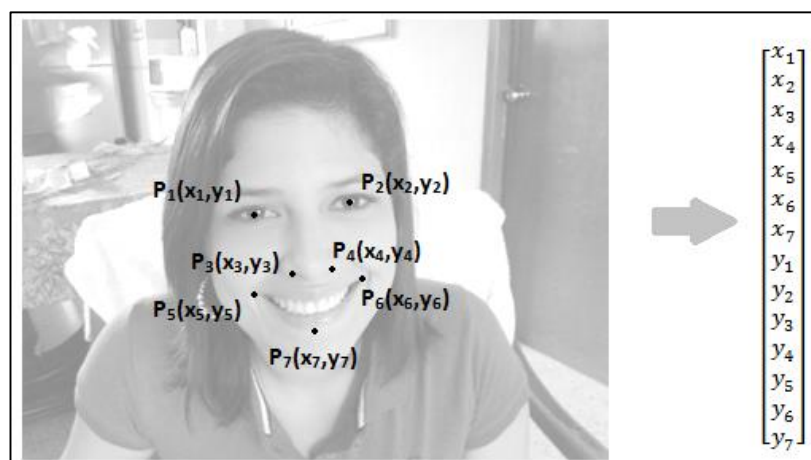


Ilustración 8. Los siete puntos de referencia para el modelo de Procrustes

Las estimaciones y las conclusiones con los valores P pierden potencia porque hay menos observaciones que variables (ocho versus catorce). Las matrices de datos son:

```
> round(datos_Rebeca, 4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3494 0.4333 0.3936 0.3539 0.4046 0.3848 0.4002 0.4333
x2 0.5172 0.5835 0.5835 0.5349 0.5945 0.5592 0.5592 0.5746
x3 0.4090 0.4974 0.4576 0.4223 0.4753 0.4510 0.4576 0.4907
x4 0.4841 0.5437 0.5216 0.4819 0.5305 0.5062 0.5084 0.5371
x5 0.3539 0.4576 0.3759 0.3406 0.4223 0.4068 0.4113 0.4311
x6 0.5415 0.5790 0.5835 0.5459 0.5724 0.5459 0.5393 0.5812
x7 0.4554 0.5194 0.4775 0.4488 0.5040 0.4819 0.4775 0.5040
y1 0.5684 0.6511 0.7172 0.6897 0.5187 0.5463 0.7007 0.6621
y2 0.6180 0.6731 0.7117 0.6952 0.5132 0.5629 0.6952 0.6676
y3 0.3037 0.4746 0.4636 0.4250 0.2982 0.2761 0.4526 0.4416
y4 0.3258 0.4746 0.4581 0.4416 0.2872 0.2816 0.4636 0.4471
y5 0.2155 0.3147 0.3313 0.3423 0.1107 0.0997 0.3313 0.3478
y6 0.2816 0.3533 0.3423 0.3588 0.1217 0.1217 0.3368 0.3699
y7 0.0611 0.2596 0.1548 0.1659 0.0170 -0.0051 0.2486 0.1603

> round(datos_Marlon, 4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3936 0.4002 0.3605 0.3428 0.3296 0.3737 0.3826 0.3870
x2 0.5327 0.5305 0.5062 0.4863 0.4797 0.5327 0.5481 0.5150
x3 0.4400 0.4488 0.4068 0.3914 0.3781 0.4311 0.4377 0.4289
x4 0.5040 0.4951 0.4753 0.4598 0.4422 0.4974 0.5062 0.4885
x5 0.3826 0.4223 0.3605 0.3539 0.3539 0.4046 0.4090 0.3914
x6 0.5371 0.5349 0.5106 0.4907 0.4819 0.5327 0.5393 0.5238
x7 0.4620 0.4753 0.4355 0.4245 0.4179 0.4709 0.4709 0.4687
y1 0.7117 0.6456 0.4250 0.5849 0.6952 0.6401 0.5959 0.4636
y2 0.7062 0.6731 0.4416 0.5959 0.6952 0.6621 0.6125 0.4801
y3 0.5077 0.4360 0.2100 0.3644 0.4691 0.3754 0.2982 0.2927
y4 0.5022 0.4305 0.2155 0.3533 0.4746 0.3864 0.3092 0.3037
y5 0.4030 0.2872 0.1107 0.2596 0.3478 0.2431 0.1659 0.1603
y6 0.3809 0.3037 0.1273 0.2596 0.3754 0.2596 0.1769 0.1989
y7 0.2761 0.2210 0.0115 0.1548 0.2651 0.1659 0.0831 0.0446
```

En cada foto se forma un heptágono irregular donde los vértices lucen como en los Gráficos 9 y 10.

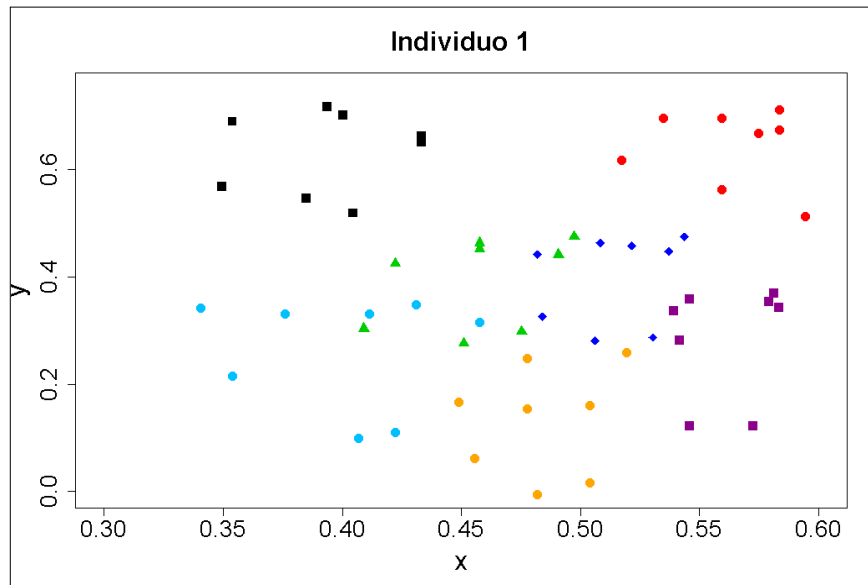


Gráfico 9. Gráfico de puntos correspondiente al individuo uno con siete puntos de referencia

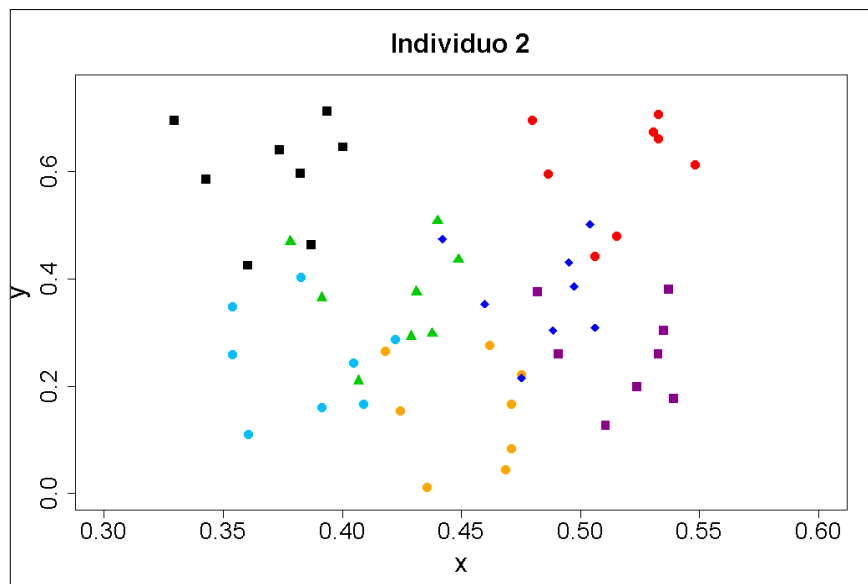


Gráfico 10. Gráfico de puntos correspondiente al individuo dos con siete puntos de referencia

Cada punto de un color determinado representa un mismo punto notable en las distintas fotografías, el patrón heptagonal no es tan distinguible debido a la elevada dispersión. Los cuadrados negros y los círculos rojos corresponden a los ojos, los triángulos verdes y los rombos azules corresponden a los agujeros de la nariz, los círculos celestes y los cuadrados lila corresponden a los extremos laterales de la boca y los círculos naranjas representan el punto más bajo de la misma.

3.7.1 Ajustando el modelo en el individuo 1

El vector estimador de la media para el primer individuo es:

```
> round(mu, 3)
  x1  x2  x3  x4  x5  x6  x7  y1  y2  y3  y4  y5  y6  y7
0.394 0.563 0.458 0.514 0.400 0.561 0.484 0.632 0.642 0.392 0.397 0.262 0.286 0.133
```

Con lo cual el estimador del polígono promedio desconocido luce así

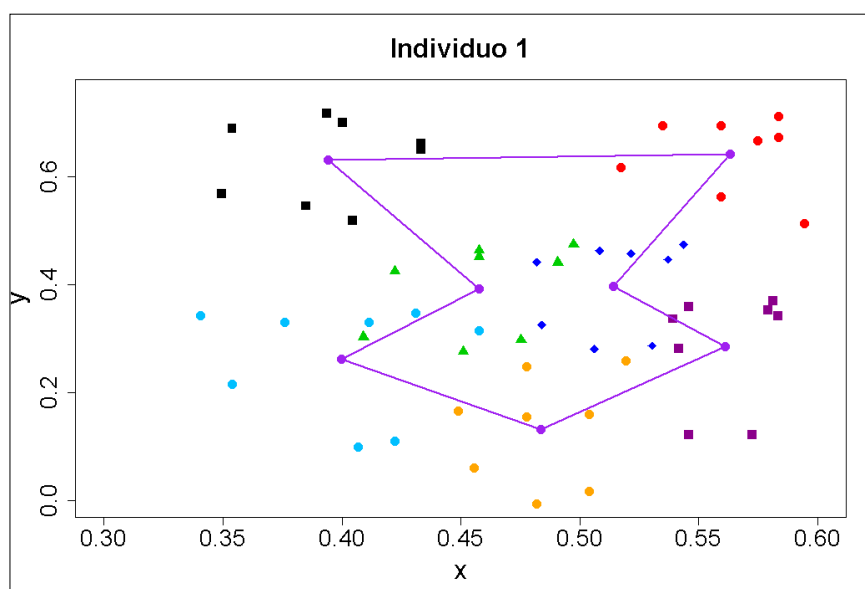


Gráfico 11. Estimación del polígono medio desconocido para el individuo uno con siete puntos de referencia

La varianza estimada de los efectos aleatorios mediante VLS está dada por el vector:

```
> round(vp,4)
[1] 0.0005 0.2492 5.0772 -1.1249 0.9990 11.8705 1.4329
```

Y el respectivo valor de la función logaritmo de la verosimilitud en este punto es:

```
> logver(datos_Rebeca,mu,vp)
[1] 349.1693
```

Al maximizar esta función para obtener el estimador de máxima verosimilitud de la varianza de los efectos aleatorios, se obtiene:

```
> round(vp_opt,4)
[1] 0.0004 0.2492 5.0772 -1.1249 0.9990 11.8705 1.4329
```

El cual es un vector que difiere del anterior solo en la primera componente. La varianza del error es pequeña con respecto a la varianza de los efectos aleatorios; la rotación y el reescalamiento del polígono promedio tienen varianzas dispares y tienen correlación negativa. La dispersión del desplazamiento horizontal respecto del polígono promedio es más pequeña que la del desplazamiento vertical y tienen correlación positiva. El valor óptimo de la función en este punto es:

```
> logver(datos_Rebeca,mu,resp$par)
[1] 351.1117
```

Ahora se comparan las fotos del individuo dos y se calculan sus valores P, dados por:

```
> round(valorp_Marlon,4)
[1] 0.0501 0.0732 0.0329 0.0628 0.0328 0.0690 0.0453 0.0178
```

Mediante las fotos tres, cinco, siete y ocho el modelo concluye correctamente que pertenecen a un individuo distinto; con las cuatro restantes concluye lo contrario.

Comparando las mismas fotos del individuo uno se tiene los siguientes valores P:

```
> round(valorp_Rebeca2,4)
[1] 0.0639 0.0133 0.0734 0.0796 0.0766 0.0309 0.0704 0.0801
```

Al evaluar las mismas fotos del individuo uno, el modelo concluye que se trata de fotografías de él mismo en seis ocasiones y en dos se equivoca (fotos dos y seis).

3.7.2 Ajustando el modelo en el individuo 2

Para este caso, el vector estimador del polígono promedio desconocido es:

```
> round(mu2,3)
  x1  x2  x3  x4  x5  x6  x7  y1  y2  y3  y4  y5  y6  y7
0.371 0.516 0.420 0.484 0.385 0.519 0.453 0.595 0.608 0.369 0.372 0.247 0.260 0.153
```

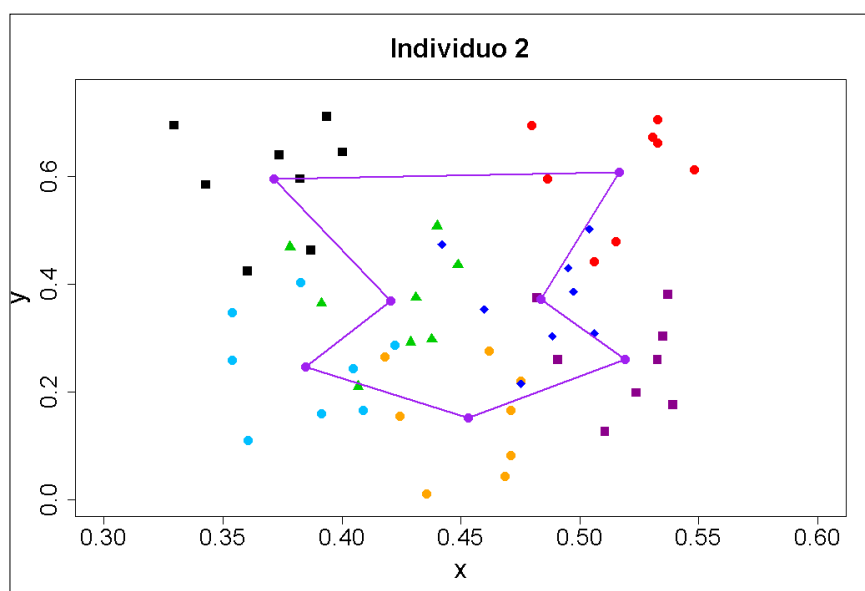


Gráfico 12. Estimación del polígono medio desconocido para el individuo dos con siete puntos de referencia

El Gráfico 12 muestra cómo luce el estimador del polígono promedio desconocido para el individuo dos. La varianza estimada de los efectos aleatorios mediante VLS está dada por el vector:

```
> round(vp2,4)
[1] 0.0002 17.8360 0.0131 -0.4660 0.1816 34.6215 -2.5073
```

Y el respectivo valor de la función logaritmo de la verosimilitud en este punto es:

```
> logver(datos_Marlon,mu2,vp2)
[1] 385.6236
```

El estimador de la varianza que maximiza la función de verosimilitud es:

```
> round(vp_opt2,4)
[1] 0.0001 19.8627 2.6022 -6.7553 1.4978 39.5474 -0.2873
```

La varianza del error es pequeña en relación a la varianza de los efectos aleatorios, la rotación y el reescalamiento del polígono promedio tienen varianzas muy diferentes y tienen correlación negativa. La dispersión del desplazamiento horizontal respecto del polígono promedio es bastante más pequeña que la del desplazamiento vertical y tienen correlación negativa también. El valor óptimo de la función en este punto es un poco mayor que en el valor inicial:

```
> logver(datos_Marlon,mu2,resp2$par)
[1] 395.7069
```

Al calcular los respectivos valores P cuando se comparan con este modelo las del individuo dos, se tiene:

```
> round(valorp_Rebeca,4)
[1] 0.0000 0.0000 0.0000 0.0000 0.0002 0.0000 0.0659 0.0000
```

Para cada foto del otro individuo se ha hallado un valor P menor que 0.05, con excepción de la foto número siete, lo cual indica que el modelo ha concluido correctamente que estas fotos pertenecen a un individuo distinto en siete ocasiones. En la mencionada foto siete se concluye lo contrario. Como se ha hecho anteriormente, se comparará las fotografías del mismo individuo dos con el modelo ajustado en sí mismo, obteniendo los siguientes valores P:

```
> round(valorp_Marlon2,4)
[1] 0.0153 0.0696 0.0800 0.0147 0.0459 0.0779 0.0003 0.0001
```

El modelo ha concluido que estas fotos pertenecen al mismo individuo dos en solo tres de ocho ocasiones, en las fotografías dos, tres y seis; en las restantes se equivoca y concluye lo contrario.

3.8 El modelo de Procrustes con registración de imágenes

Todo lo realizado en las secciones anteriores en este capítulo se ha realizado usando las imágenes tal cual fueron tomadas. Ahora, en esta sección se hará una registración de imágenes antes de ajustar el modelo de Procrustes. El método que se usará será la registración afín, que fue explicada brevemente en el primer capítulo correspondiente al Marco Teórico. La registración se hará “dentro” de cada individuo, se tomará la primera imagen de cada uno como la imagen de origen y las restantes ocho serán

imágenes de destino, es decir las fotografías dos a ocho de cada individuo serán *corregistradas* con la primera, y además de manera independiente cada una.

Como se mencionó antes, una imagen a color tiene tres matrices asociadas: una al color rojo, otra al color verde y otra al color azul; estas tres imágenes están en escala de grises. Un ejemplo de esta descomposición en tres imágenes correspondiente a la primera foto del individuo uno se muestra en la siguiente ilustración.

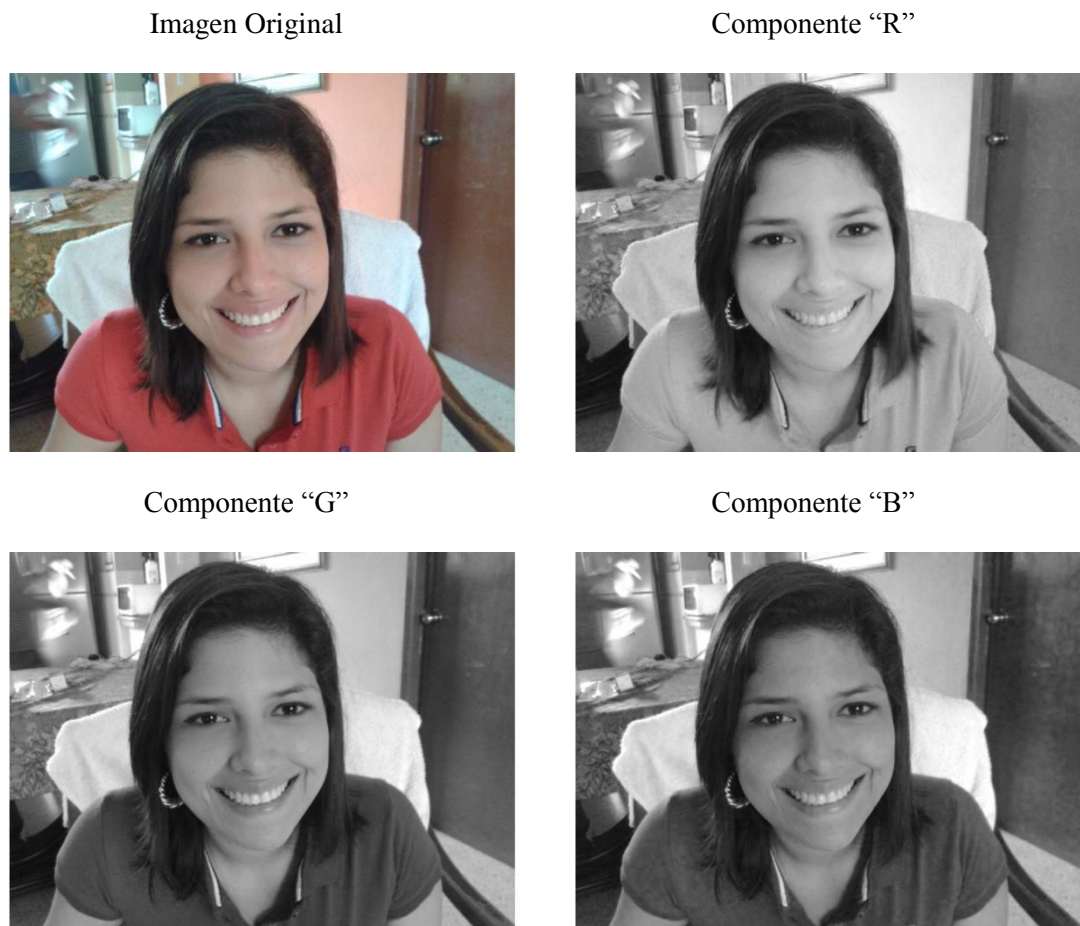


Ilustración 9. Componentes RGB de una imagen del individuo uno

Como se puede observar, el rostro del individuo es perfectamente reconocible en cualquiera de las tres componentes de la imagen, por tanto la registración podría realizarse (y simplificarse) usando cualquiera de ellas como si se tratara de una imagen en escala de grises. Se usará la componente “R” de cada imagen en nuestro caso. El resultado para cada individuo serán ocho nuevas imágenes que corresponden a las imágenes de destino debidamente alineadas con su respectiva primera imagen, como se muestra a continuación con dos imágenes de ejemplo del individuo uno.

Imagen 1 (origen)



Imagen 4 original

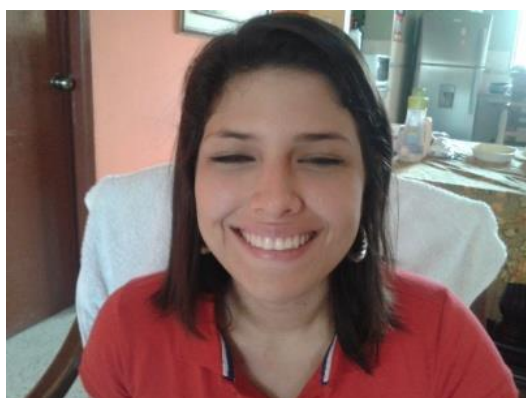


Imagen 4 corregistrada

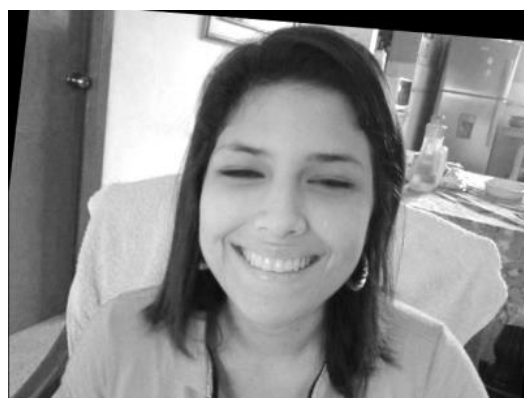


Imagen 10 original

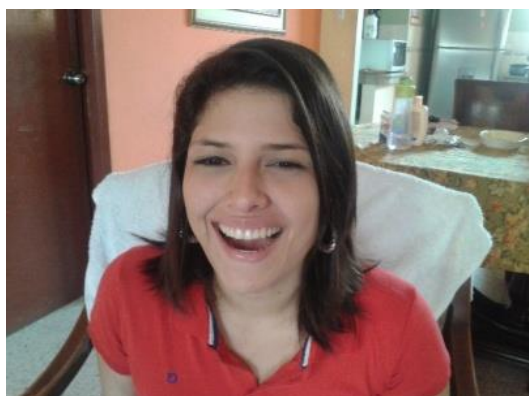


Imagen 10 corregistrada



Ilustración 10. Registración de imágenes en el individuo 1

La librería de R con la que se hizo la registración de imágenes es `RNiftyReg` (19) y la respectiva función que realiza registración lineal es `niftyreg.linear`. Esta librería está orientada principalmente hacia imágenes en formato NIFTI, pero funciona también con imágenes en formato JPEG. Luego, las imágenes corregistradas se guardan en el disco duro en formato NIFTI y fueron convertidas después en formato PNG, el cual es admitido por la librería `EBImage`, que se usa nuevamente para importarlas a R como si se tratara de las imágenes originales. Finalmente, la función `locator` permite señalar las características notables con un determinado número de puntos de referencia de la misma manera en que se explicó al inicio de este capítulo en la sección 3.2.

La registración de imágenes tiene por objetivo alinear las imágenes de cada individuo, por tanto, se espera que al hacer los respectivos gráficos de las coordenadas de los puntos notables, estos luzcan menos dispersos que antes. En otras

palabras, esperamos que la varianza, tanto la del error como la de los efectos aleatorios, disminuya. El proceso será similar al anterior: ajustar nuevamente el modelo de Procrustes para cada individuo con tres, cinco y siete puntos de referencia y comparar las fotos del otro individuo y de sí mismo.

3.8.1 El modelo Procrustes con tres puntos de referencia y registración de imágenes

Al igual que en el caso anterior donde no hubo registración de imágenes, después de usar la función `locator` y arreglar los datos obtenemos una matriz de seis filas y ocho columnas para cada individuo, donde están dispuestas las coordenadas de cada vértice de la manera descrita en la sección 2.1 pero esta vez correspondientes a las imágenes registradas. Se tiene:

```
> round(datos_Rebeca,4)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.4023 0.3981 0.4023 0.4043 0.3898 0.4023 0.4023 0.3857
x2 0.5679 0.5596 0.5762 0.5741 0.5638 0.5658 0.5493 0.5306
x3 0.4706 0.4665 0.4768 0.4768 0.4747 0.4623 0.4665 0.4458
y1 0.6235 0.7117 0.6235 0.6235 0.5298 0.5904 0.6787 0.6401
y2 0.5739 0.6676 0.5849 0.5849 0.5463 0.5573 0.6401 0.6180
y3 0.3202 0.5022 0.3699 0.3588 0.3313 0.3313 0.4305 0.4085
> round(datos_Marlon,4)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3961 0.4043 0.3898 0.4043 0.3961 0.3836 0.3795 0.4106
x2 0.5286 0.5369 0.5265 0.5369 0.5369 0.5348 0.5265 0.5348
x3 0.4582 0.4685 0.4520 0.4685 0.4685 0.4582 0.4582 0.4644
y1 0.7062 0.6676 0.4801 0.6456 0.7117 0.6180 0.5794 0.7117
y2 0.7228 0.6566 0.4801 0.6401 0.7393 0.6070 0.5684 0.6952
y3 0.5187 0.4360 0.2541 0.4030 0.5022 0.3588 0.2761 0.5243
```

La gráfica correspondiente a cada una de estas matrices es:

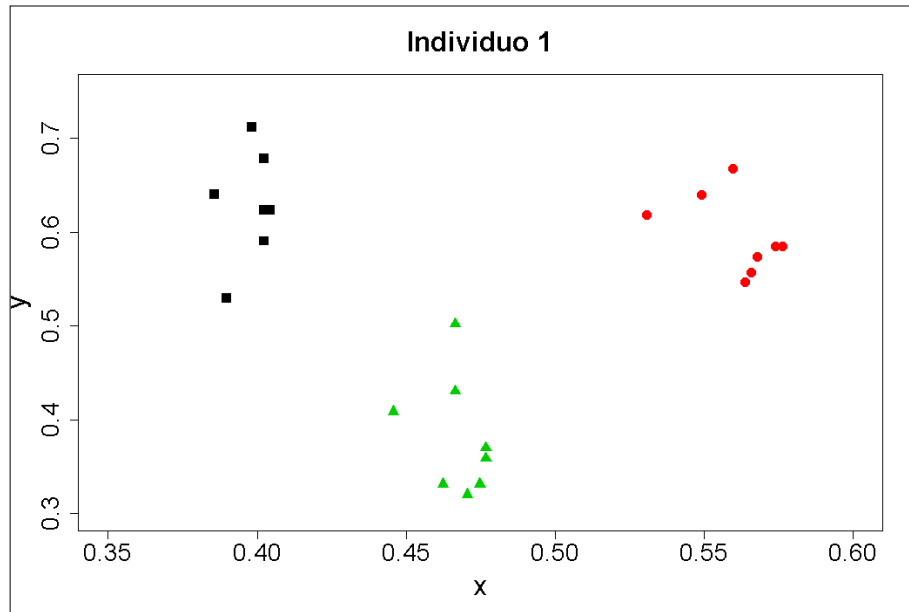


Gráfico 13. Gráfico con tres puntos correspondiente al individuo uno con registraci3n

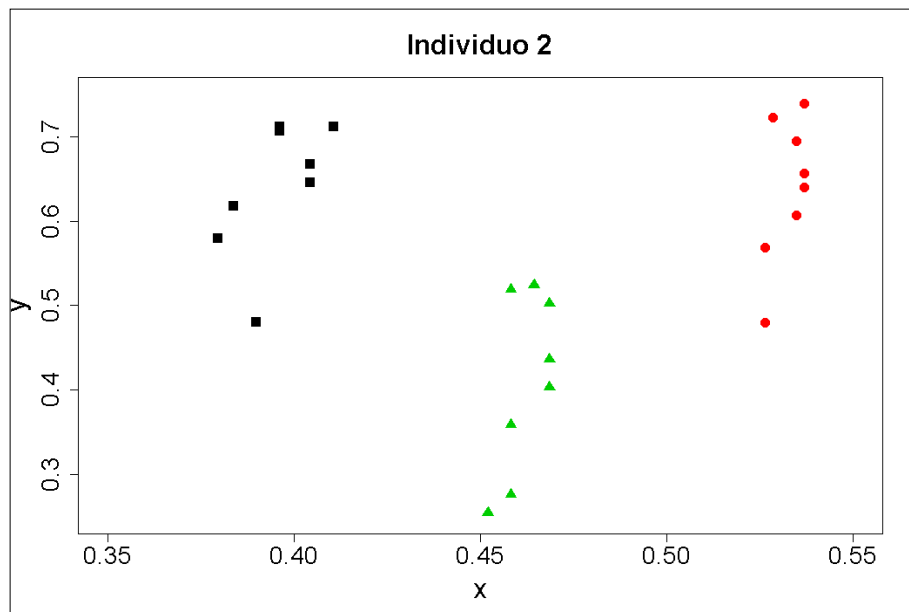


Gráfico 14. Gráfico con tres puntos correspondiente al individuo dos con registraci3n

Comparando estos gráficos con los respectivos Gráficos 1 y 2 cuando no había registración de imágenes, es fácil apreciar que la dispersión de los puntos se ha reducido. Ahora se ajustará el modelo en cada individuo.

1) En el individuo uno

La estimación del polígono promedio y su respectiva representación son:

```
> round(mu, 4)
  x1    x2    x3    y1    y2    y3
0.3984 0.5609 0.4675 0.6276 0.5966 0.3816
```

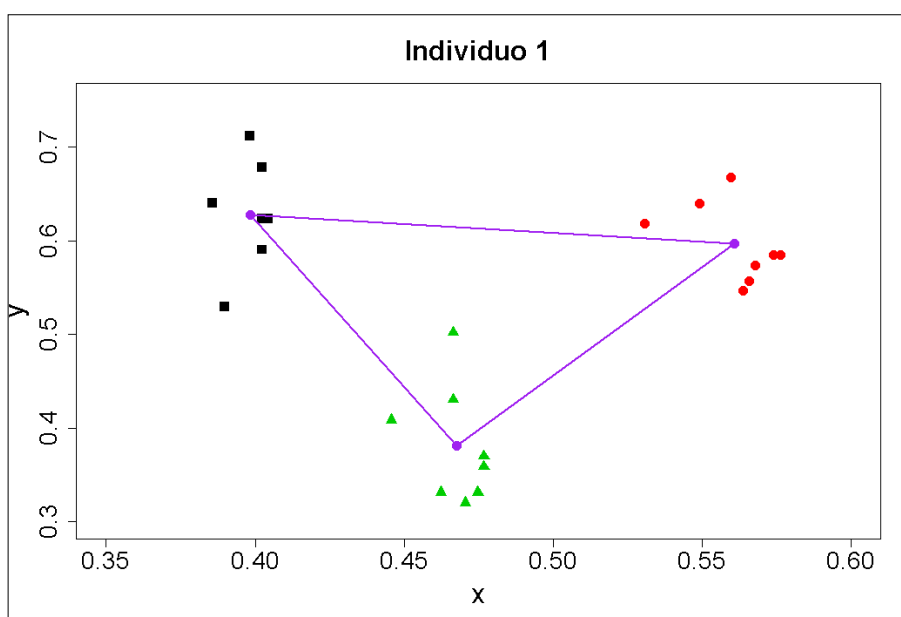


Gráfico 15. Estimación del polígono promedio para el individuo 1 con tres puntos y registración

Los estimadores VLS de la varianza de los efectos aleatorios vienen dados por:

```
> round(vp, 4)
[1] 0.0004 2.2199 0.0014 0.0148 0.0776 5.4825 -0.6524
```

En este punto, la función logaritmo de la verosimilitud tiene el valor de:

```
> logver(datos_Rebeca,mu,vp)
[1] 160.0619
```

Los estimadores de máxima verosimilitud de la varianza de los efectos aleatorios son:

```
> round(vp_opt,4)
[1] 0.0002 2.2204 0.0000 -0.0086 0.1108 5.5101 -0.6404
```

El nuevo valor de la función logaritmo de verosimilitud es:

```
> logver(datos_Rebeca,mu,resp$par)
[1] 165.7183
```

Con lo cual se procede a calcular los estadísticos de prueba al comparar con las fotos del individuo dos, cuyos respectivos valores P son los siguientes:

```
> round(valorp_Marlon,4)
[1] 0.0005 0.1185 0.0027 0.1285 0.0009 0.1240 0.0002 0.0004
```

El modelo de Procrustes de efectos aleatorios ha concluido correctamente que las fotos pertenecen a un individuo distinto en cinco de ocho ocasiones, con las fotografías uno, tres, cinco, siete y ocho. Con las restantes fotos ha concluido lo contrario. Comparando las mismas fotografías del primer individuo con el modelo que se ajustó en sí mismo se tienen los siguientes valores P:

```
> round(valorp_Rebeca2,4)
[1] 0.0612 0.0136 0.0308 0.0540 0.0348 0.1011 0.1131 0.1310
```

El modelo ha concluido correctamente que estas fotos pertenecen al mismo individuo uno en cinco de ocho ocasiones, mediante las fotos uno, cuatro, seis, siete y ocho. Con las restantes fotografías concluye que no se trata del mismo individuo uno.

II) En el individuo dos

La estimación del polígono promedio y su respectiva representación son:

```
> round(mu2,4)
  x1    x2    x3    y1    y2    y3
0.3955 0.5327 0.4621 0.6401 0.6387 0.4092
```

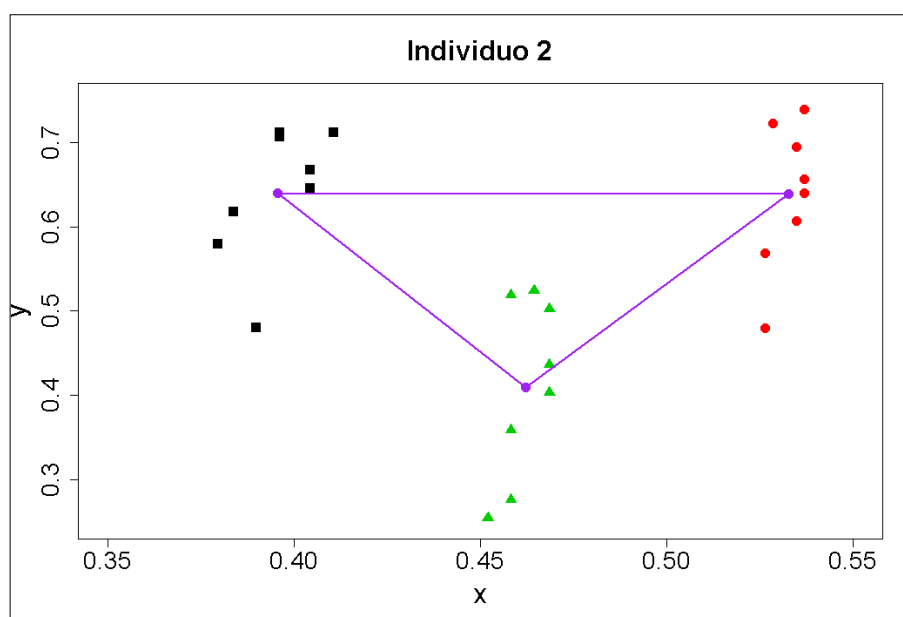


Gráfico 16. Estimación del polígono promedio para el individuo dos con tres puntos y registración

Los estimadores VLS de la varianza de los efectos aleatorios vienen dados por:

```
> round(vp2,4)
[1] 0.0001 0.3877 0.9839 -0.6176 4.5506 86.4104 5.8314
```

En este punto, la función logaritmo de la verosimilitud tiene el valor de:

```
> logver(datos_Marlon,mu2,vp2)
[1] 147.038
```

Los estimadores de máxima verosimilitud de la varianza de los efectos aleatorios son:

```
> round(vp_opt2,4)
[1] 0.0001 0.3877 0.9839 -0.6176 4.5506 86.4104 5.8314
```

El nuevo valor de la función logaritmo de verosimilitud es:

```
> logver(datos_Marlon,mu2,resp2$par)
[1] 152.5633
```

A pesar de que los estimadores VLS y de máxima verosimilitud coinciden, hay un incremento en la función de máxima verosimilitud, esto puede deberse al redondeo a cuatro decimales. Ahora se calcularán los estadísticos de prueba al comparar con las fotos del individuo uno, cuyos respectivos valores P son los siguientes:

```
> round(valorp_Rebeca,4)
[1] 0.0009 0.0054 0.0459 0.0461 0.0606 0.0721 0.1294 0.1110
```

El modelo de Procrustes de efectos aleatorios ha concluido correctamente estas fotos pertenecen a un individuo distinto en cuatro de ocho ocasiones; con las fotografías uno, dos, tres y cuatro. Con las restantes fotos ha concluido que pertenecen al mismo individuo dos. Comparando las mismas fotografías del segundo individuo con el modelo que se ajustó se tienen los siguientes valores P:

```
> round(valorp_Marlon2,4)
[1] 0.0997 0.0123 0.0900 0.0148 0.1328 0.1304 0.0010 0.0178
```


El modelo ha concluido correctamente que estas fotos pertenecen al mismo individuo dos en cuatro de ocho ocasiones, mediante las fotos uno, tres, cinco y seis. Con las restantes fotografías se concluye que no se trata del mismo individuo dos.

3.8.2 El modelo Procrustes con cinco puntos de referencia y registraci3n de im3genes

Para este caso, se ajustar3 el modelo de Procrustes con los mismos cinco puntos de referencia que se usaron antes, pero esta vez obtenidos de las im3genes registradas.

Las dos matrices de coordenadas obtenidas son:

```
> round(datos_Rebeca,4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.4064 0.3981 0.4002 0.4023 0.3878 0.4023 0.4023 0.3857
x2 0.5679 0.5617 0.5679 0.5762 0.5679 0.5638 0.5534 0.5244
x3 0.3836 0.3940 0.3691 0.3774 0.3898 0.3754 0.3981 0.3774
x4 0.5596 0.5182 0.5679 0.5679 0.5472 0.5224 0.5182 0.5265
x5 0.4582 0.4437 0.4602 0.4582 0.4644 0.4375 0.4520 0.4437
y1 0.6180 0.7007 0.6290 0.6290 0.5243 0.6015 0.6676 0.6401
y2 0.5573 0.6842 0.5849 0.5684 0.5298 0.5463 0.6345 0.6070
y3 0.2927 0.3533 0.2596 0.2816 0.1273 0.1603 0.2927 0.3147
y4 0.2155 0.3092 0.2045 0.2265 0.1162 0.1217 0.2541 0.2872
y5 0.0611 0.2431 0.0170 0.0611 0.0225 0.0280 0.1879 0.0942
> round(datos_Marlon,4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3919 0.4043 0.3857 0.4002 0.3981 0.3836 0.3774 0.4043
x2 0.5244 0.5348 0.5286 0.5327 0.5348 0.5306 0.5265 0.5327
x3 0.3898 0.4126 0.3836 0.4023 0.4023 0.3919 0.4002 0.4002
x4 0.5244 0.5182 0.5327 0.5244 0.5244 0.5162 0.5203 0.5286
x5 0.4540 0.4665 0.4520 0.4623 0.4623 0.4540 0.4582 0.4602
y1 0.7007 0.6621 0.4801 0.6345 0.7117 0.6070 0.5629 0.7062
y2 0.7172 0.6511 0.4912 0.6456 0.7338 0.6125 0.5573 0.7007
y3 0.3974 0.2706 0.1659 0.2927 0.3644 0.2210 0.1493 0.3974
y4 0.3919 0.2706 0.1603 0.2927 0.3754 0.2155 0.1493 0.3974
y5 0.2706 0.2100 0.0446 0.1934 0.2816 0.1273 0.0666 0.2651
```

La gr3fica correspondiente a cada una de estas matrices es:

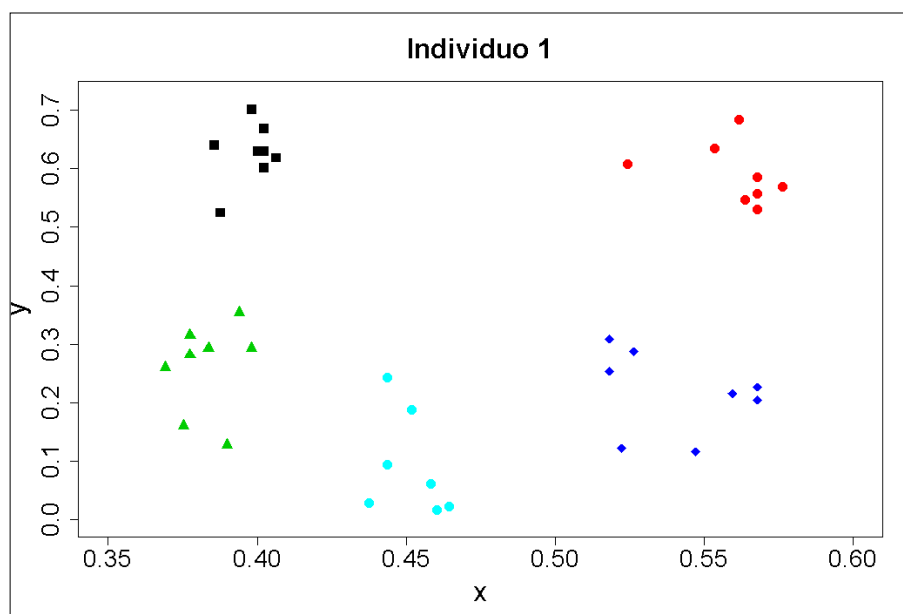


Gráfico 17. Gráfico con cinco puntos correspondiente al individuo uno con registraci3n

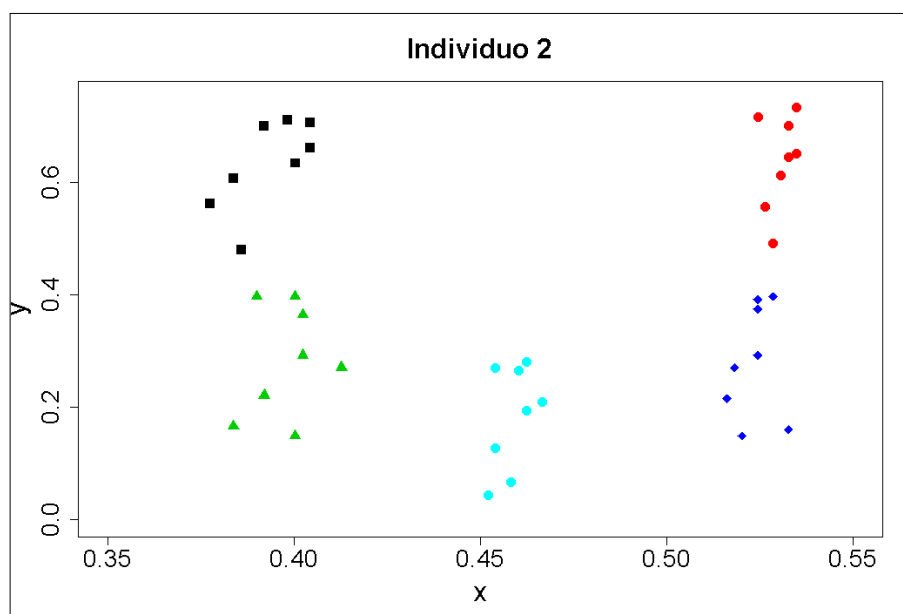


Gráfico 18. Gráfico con cinco puntos correspondiente al individuo dos con registraci3n

Comparando estos gráficos con los respectivos gráficos 5 y 6 que fueron obtenidos sin corregir las imágenes, se puede notar que la dispersión de los puntos se ha reducido. Ahora se procederá a ajustar el modelo en cada individuo.

I) En el individuo uno

La estimación del polígono promedio y su respectiva representación son:

```
> round(mu,4)
      x1      x2      x3      x4      x5      y1      y2      y3      y4      y5
0.3981 0.5604 0.3831 0.5410 0.4522 0.6263 0.5890 0.2603 0.2169 0.0894
```

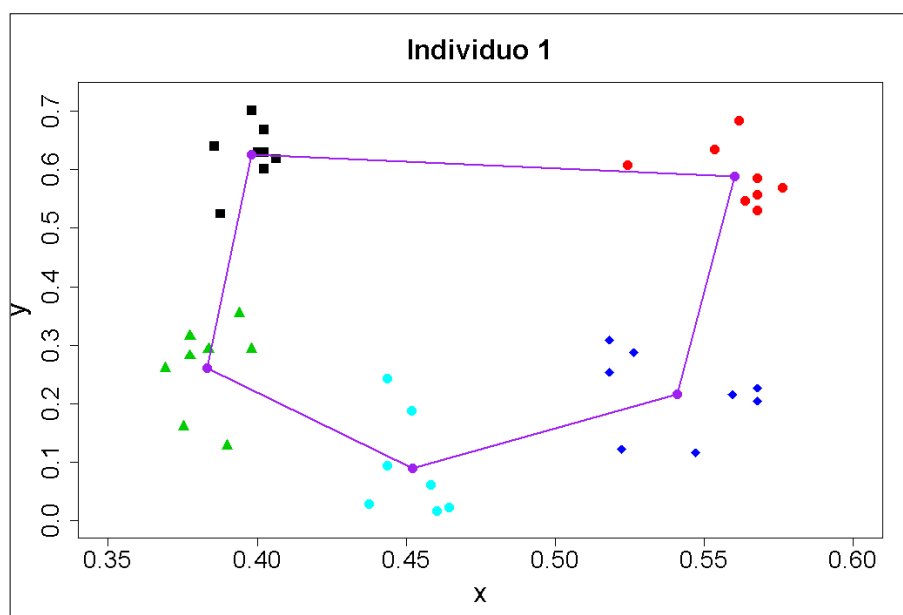


Gráfico 19. Estimación del polígono promedio para el individuo uno con cinco puntos y registración

Los estimadores VLS de la varianza de los efectos aleatorios vienen dados por:

```
> round(vp,4)
[1] 0.0006 0.2021 1.1486 0.4819 0.0265 5.7955 0.3390
```

En este punto, la función logaritmo de la verosimilitud tiene el valor de:

```
> logver(datos_Rebeca,mu,vp)
[1] 249.9199
```

Los estimadores de máxima verosimilitud de la varianza de los efectos aleatorios son:

```
> round(vp_opt,4)
[1] 0.0005 0.2026 1.1478 0.4802 0.0232 5.7957 0.3383
```

El nuevo valor de la función logaritmo de verosimilitud es:

```
> logver(datos_Rebeca,mu,resp$par)
[1] 250.465
```

Con lo cual se puede calcular los estadísticos de prueba al comparar con las fotos del individuo dos, cuyos respectivos valores P son los siguientes:

```
> round(valorp_Marlon,4)
[1] 0.0069 0.0184 0.0362 0.0505 0.0067 0.0931 0.0516 0.0068
```

El modelo de Procrustes de efectos aleatorios ha concluido correctamente que las fotos pertenecen a un individuo distinto en cinco de ocho ocasiones; con las fotografías uno, dos, tres, cinco y ocho. Con las restantes fotos ha concluido lo opuesto. Comparando las mismas fotografías del primer individuo con el modelo que se ajustó en sí mismo se tienen los siguientes valores P:

```
> round(valorp_Rebeca2,4)
[1] 0.0782 0.0719 0.0947 0.0630 0.0809 0.0851 0.0976 0.0976
```

El modelo ha concluido correctamente que estas fotos pertenecen al mismo individuo uno en las ocho ocasiones, lo cual sin duda es un resultado sorprendente.

II) En el individuo dos

La estimación del polígono promedio y su respectiva representación son:

```
> round(mu2, 4)
  x1    x2    x3    x4    x5    y1    y2    y3    y4    y5
0.3932 0.5306 0.3979 0.5237 0.4587 0.6332 0.6387 0.2823 0.2816 0.1824
```

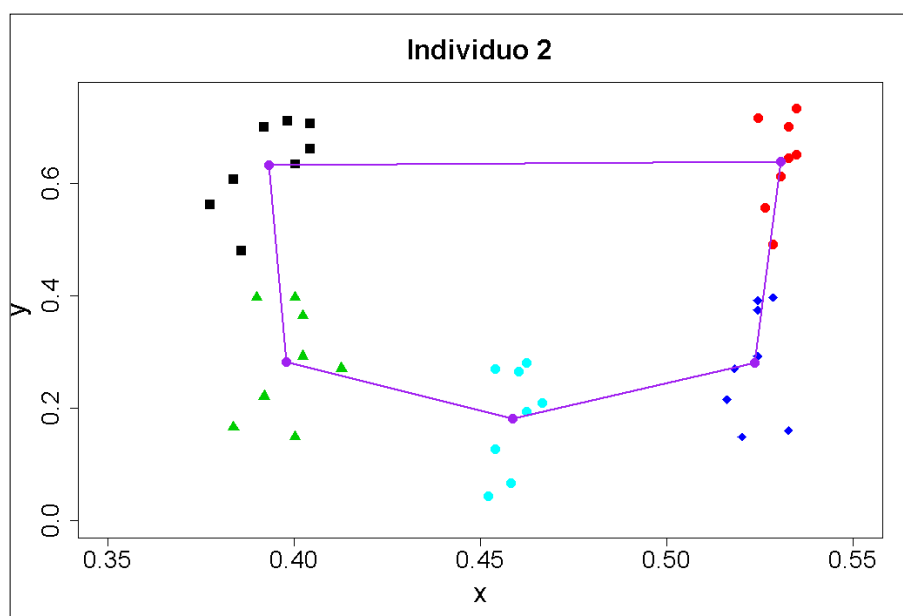


Gráfico 20. Estimación del polígono promedio para el individuo 2 con cinco puntos y registración

Los estimadores VLS de la varianza de los efectos aleatorios vienen dados por:

```
> round(vp2, 4)
[1] 0.0003 0.0499 2.1973 0.3310 0.3716 24.9753 2.1055
```

En este punto, la función logaritmo de la verosimilitud tiene el valor de:

```
> logver(datos_Marlon,mu2,vp2)
[1] 272.9309
```

Los estimadores de máxima verosimilitud de la varianza de los efectos aleatorios son:

```
> round(vp_opt2,4)
[1] 0.0002 0.1446 2.1647 0.0736 0.1807 24.9852 2.0844
```

El nuevo valor de la función logaritmo de verosimilitud es:

```
> logver(datos_Marlon,mu2,resp2$par)
[1] 277.2363
```

Ahora se calcularán los estadísticos de prueba al comparar con las fotos del individuo uno, cuyos respectivos valores P son los siguientes:

```
> round(valorp_Rebeca,4)
[1] 0.0000 0.0681 0.0000 0.0000 0.0010 0.0000 0.0775 0.0000
```

El modelo ha concluido correctamente estas fotos pertenecen a un individuo distinto en seis de ocho ocasiones; con las fotografías uno, tres, cuatro, cinco, seis y ocho. Con las restantes fotos concluye lo contrario. Ahora se compararán las mismas fotografías del segundo individuo con el modelo que fue ajustado en sí mismo, para lo cual se tienen los siguientes valores P:

```
> round(valorp_Marlon2,4)
[1] 0.0862 0.0967 0.0905 0.0001 0.0232 0.0971 0.0144 0.0570
```

El modelo ha concluido correctamente que estas fotos pertenecen al mismo individuo dos en cinco de ocho ocasiones, mediante las fotos uno, dos, tres, seis y ocho. Con las restantes fotografías concluye que no se trata del mismo individuo dos.

3.8.3 El modelo Procrustes con siete puntos de referencia y registraci3n de im3genes

Se ajustar3 el modelo de Procrustes con los siete puntos de referencia que se usaron antes, pero esta con las im3genes corregistradas. Las matrices de coordenadas que se obtienen al se3alar los puntos notables son:

```
> round(datos_Rebeca,4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.4043 0.3961 0.4023 0.4064 0.3898 0.4023 0.4023 0.3857
x2 0.5638 0.5617 0.5721 0.5824 0.5783 0.5638 0.5472 0.5244
x3 0.4354 0.4395 0.4478 0.4416 0.4499 0.4416 0.4458 0.4209
x4 0.5017 0.4872 0.5058 0.5037 0.5017 0.4872 0.4913 0.4685
x5 0.3816 0.3898 0.3733 0.3816 0.3981 0.3816 0.4064 0.3774
x6 0.5596 0.5203 0.5658 0.5658 0.5410 0.5203 0.5182 0.5327
x7 0.4520 0.4458 0.4540 0.4561 0.4582 0.4416 0.4520 0.4458
y1 0.6235 0.7172 0.6125 0.6290 0.5243 0.6015 0.6731 0.6345
y2 0.5684 0.6621 0.5904 0.5684 0.5353 0.5518 0.6401 0.6125
y3 0.3202 0.4912 0.3423 0.3423 0.2982 0.2982 0.4140 0.3974
y4 0.2982 0.4801 0.3258 0.3258 0.3092 0.2872 0.4030 0.3864
y5 0.2927 0.3478 0.2431 0.2872 0.1328 0.1548 0.2927 0.3313
y6 0.2265 0.3202 0.1989 0.2210 0.1273 0.0942 0.2541 0.2816
y7 0.0831 0.2486 0.0115 0.0776 0.0335 0.0446 0.1934 0.0831
> round(datos_Marlon,4)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
x1 0.3919 0.4043 0.3898 0.4002 0.3981 0.3836 0.3774 0.4064
x2 0.5244 0.5348 0.5224 0.5348 0.5389 0.5348 0.5244 0.5327
x3 0.4209 0.4437 0.4230 0.4375 0.4375 0.4313 0.4271 0.4375
x4 0.4851 0.4954 0.4851 0.4954 0.4913 0.4934 0.4892 0.4913
x5 0.3857 0.4126 0.3857 0.4043 0.4023 0.3940 0.4023 0.3981
x6 0.5327 0.5224 0.5265 0.5244 0.5203 0.5203 0.5203 0.5286
x7 0.4540 0.4665 0.4520 0.4623 0.4644 0.4561 0.4582 0.4602
y1 0.7007 0.6566 0.4857 0.6345 0.7062 0.6125 0.5684 0.7117
y2 0.7062 0.6566 0.4801 0.6456 0.7393 0.6015 0.5629 0.7007
y3 0.5022 0.4250 0.2541 0.3974 0.4912 0.3533 0.2872 0.5243
y4 0.5187 0.4140 0.2431 0.3974 0.4912 0.3478 0.2761 0.5132
y5 0.3809 0.2706 0.1714 0.2872 0.3588 0.2210 0.1493 0.3974
y6 0.4085 0.2816 0.1603 0.2982 0.3699 0.2210 0.1493 0.3919
y7 0.2816 0.2100 0.0390 0.2100 0.2761 0.1383 0.0611 0.2651
```

La gráfica correspondiente a cada una de estas matrices es:

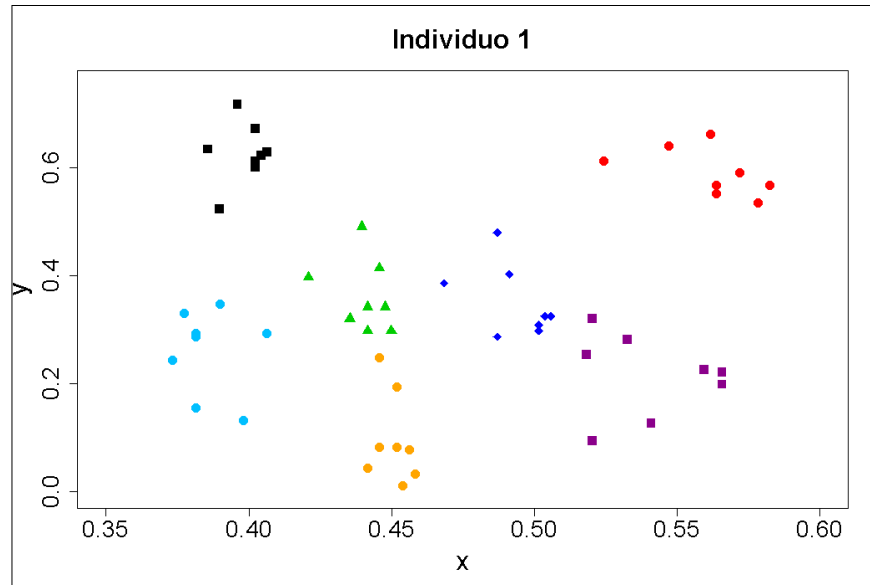


Gráfico 21. Gráfico con siete puntos correspondiente al individuo uno con registraci3n

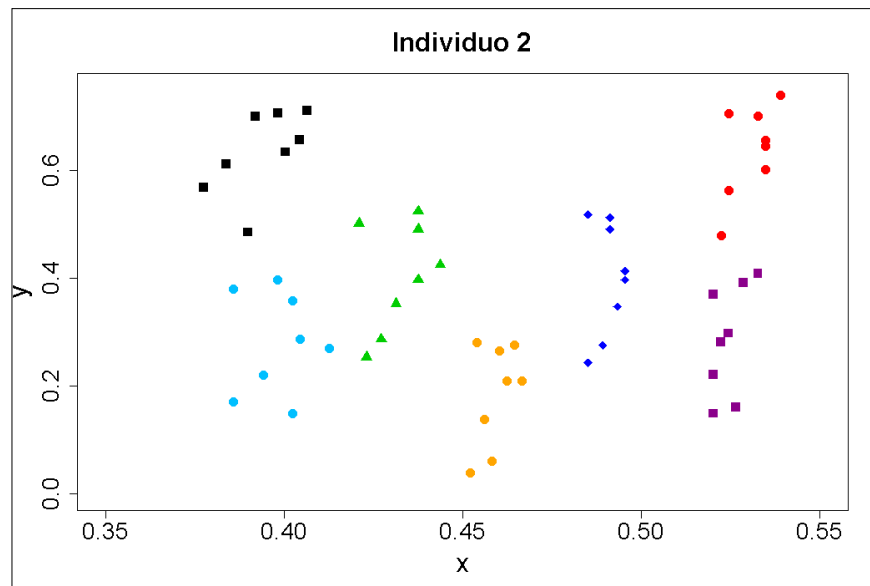


Gráfico 22. Gráfico con siete puntos correspondiente al individuo dos con registraci3n

Comparando estos gráficos con los respectivos Gráficos 9 y 10 donde no se hizo registración de imágenes, se puede ver que la dispersión de los puntos se ha reducido.

Ahora se ajustará el modelo en cada individuo.

I) En el individuo uno

La estimación del polígono promedio y su respectiva representación son:

```
> round(mu, 3)
  x1    x2    x3    x4    x5    x6    x7    y1    y2    y3    y4    y5    y6    y7
0.399 0.562 0.440 0.493 0.386 0.540 0.451 0.627 0.591 0.363 0.352 0.260 0.215 0.097
```

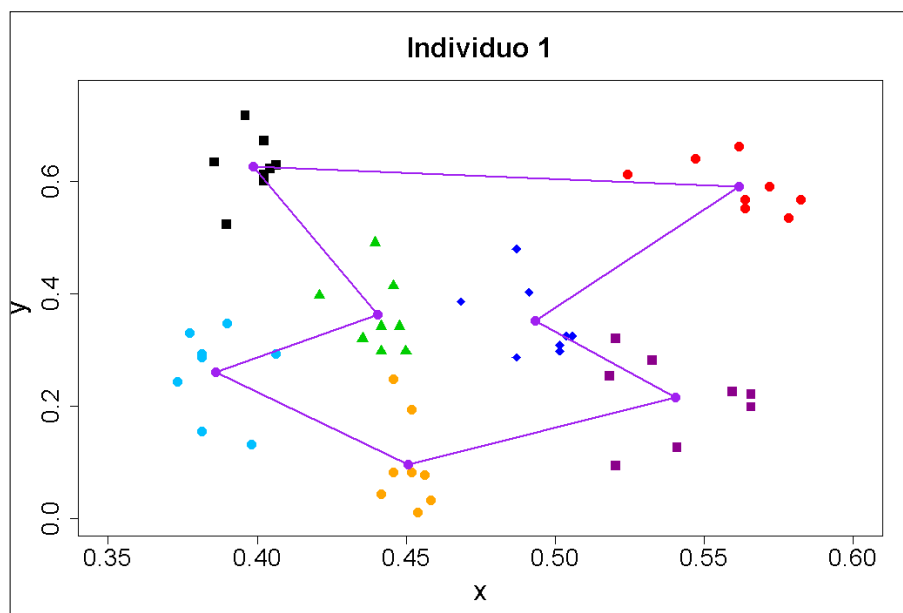


Gráfico 23. Estimación del polígono promedio para el individuo uno con siete puntos y registración

Los estimadores VLS de la varianza de los efectos aleatorios vienen dados por:

```
> round(vp, 4)
[1] 0.0004 0.6207 4.2911 0.7118 0.0365 8.7921 0.5668
```

En este punto, la función logaritmo de la verosimilitud tiene el valor de:

```
> logver(datos_Rebeca,mu,vp)
[1] 353.5863
```

Los estimadores de máxima verosimilitud de la varianza de los efectos aleatorios son:

```
> round(vp_opt,4)
[1] 0.0004 0.6207 4.2911 0.7118 0.0365 8.7921 0.5668
```

El nuevo valor de la función logaritmo de verosimilitud es:

```
> logver(datos_Rebeca,mu,resp$par)
[1] 354.7882
```

Con cuatro cifras significativas, el estimador VLS en este caso particular ha coincidido con el de máxima verosimilitud, si bien la función tuvo un pequeño incremento. Ahora se calculan los estadísticos de prueba al comparar con las fotos del individuo dos, cuyos respectivos valores P son los siguientes:

```
> round(valorp_Marlon,4)
[1] 0.0064 0.0457 0.0663 0.0399 0.0324 0.0674 0.0737 0.0288
```

El modelo de Procrustes de efectos aleatorios ha concluido correctamente que estas fotografías pertenecen a un individuo distinto en cinco de ocho ocasiones, con las fotografías uno, dos, cuatro, cinco y ocho. Con las restantes fotos se ha equivocado concluyendo lo contrario. Ahora, comparando las mismas fotografías del primer individuo con el modelo que se ajustó en sí mismo se tienen los siguientes valores P:

```
> round(valorp_Rebeca2,4)
[1] 0.0798 0.0758 0.0782 0.0309 0.0704 0.0516 0.0585 0.0767
```

El modelo ha concluido correctamente que estas fotografías pertenecen al mismo individuo uno en siete de las ocho ocasiones, solamente se equivocó concluyendo lo contrario al momento de comparar la foto cuatro.

II) En el individuo dos

La estimación del polígono promedio y su respectiva representación son:

```
> round(mu2, 3)
  x1    x2    x3    x4    x5    x6    x7    y1    y2    y3    y4    y5    y6    y7
0.394 0.531 0.432 0.491 0.398 0.524 0.459 0.635 0.637 0.404 0.400 0.280 0.285 0.185
```

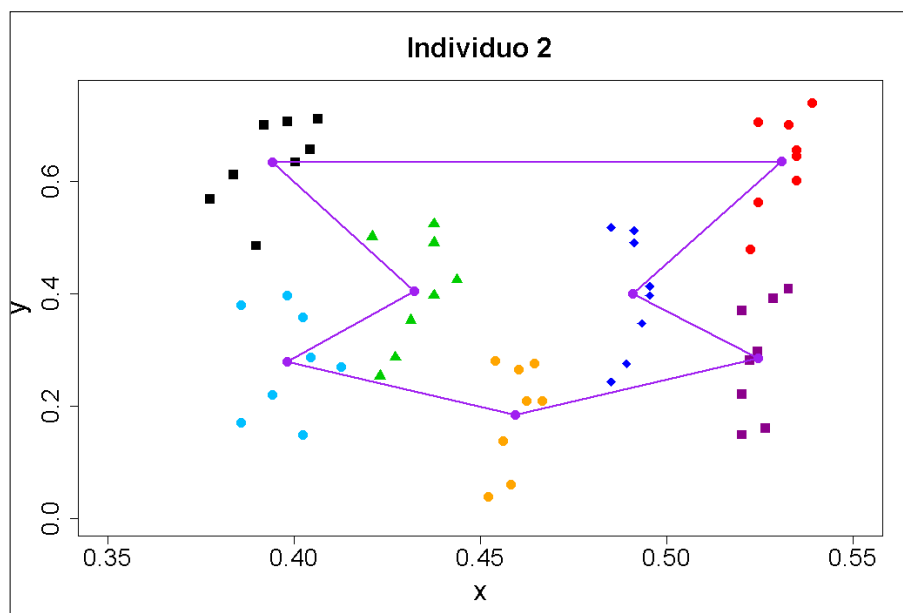


Gráfico 24. Estimación del polígono promedio para el individuo dos con siete puntos y registración

Los estimadores VLS de la varianza de los efectos aleatorios vienen dados por:

```
> round(vp2, 4)
[1] 0.0002 0.4988 3.6480 0.6113 0.0876 30.1850 1.6257
```

En este punto, la función logaritmo de la verosimilitud tiene el valor de:

```
> logver(datos_Marlon,mu2,vp2)
[1] 393.9874
```

Los estimadores de máxima verosimilitud de la varianza de los efectos aleatorios son:

```
> round(vp_opt2,4)
[1] 0.0001 0.8464 1.3407 -1.0653 0.0915 30.6026 1.4108
```

El nuevo valor de la función logaritmo de verosimilitud es:

```
> logver(datos_Marlon,mu2,resp2$par)
[1] 407.3276
```

Se ve que hay un incremento significativo en la función de verosimilitud. Ahora se calcularán los estadísticos de prueba al comparar las fotos del individuo uno, cuyos respectivos valores P son los siguientes:

```
> round(valorp_Rebeca,4)
[1] 0.0000 0.0033 0.0000 0.0000 0.0001 0.0000 0.0486 0.0000
```

El modelo de Procrustes de efectos aleatorios ha concluido correctamente que estas fotos pertenecen a un individuo distinto en las ocho ocasiones. No se ha equivocado en ninguna ocasión. Comparando ahora las mismas fotografías del segundo individuo con el modelo que se ajustó en sí mismo se tienen los siguientes valores P:

```
> round(valorp_Marlon2,4)
[1] 0.0303 0.0530 0.0580 0.0113 0.0208 0.0574 0.0025 0.0265
```

El modelo ha concluido correctamente que estas fotos pertenecen al mismo individuo dos en solo tres de ocho ocasiones, mediante las fotos dos, tres y seis. Con las restantes fotografías se concluye que no se trata del mismo individuo dos.

CAPÍTULO 4

4. Evaluación del modelo de Procrustes

4.1 Introducción

Se han ajustado tres modelos a ambos individuos con distinto número de puntos de referencia y bosquejado algunas conclusiones. En este capítulo se analizará el comportamiento de un modelo con respecto a otro de una manera más detallada y un poco más formal.

Recapitulando, sean dos individuos con N fotos (observaciones) cada uno a los que se le ajustó el modelo (2.6), usando n puntos de referencia. El modelo se ajusta a un individuo cualquiera usando sus N observaciones y con este modelo se compara una $(N+1)$ -ésima observación mediante la prueba de pertenencia, entonces para esta nueva foto que posiblemente pertenezca a otro individuo, puede ocurrir cualquiera de los dos casos siguientes:

- I. Que el modelo concluya que se trata de un individuo distinto (valor $P < 0.05$).
- II. Que el modelo concluya que se trata del mismo individuo (valor $P \geq 0.05$).

La efectividad del modelo vendría dada por la proporción de aciertos, es decir: si se ajusta el modelo usando las fotos de un individuo y se comparan N fotos de *otro* individuo y el modelo concluye que estas pertenecen a un individuo distinto ($P < 0.05$) en a ocasiones, entonces su efectividad será a/N ; si en cambio se comparan N fotos del *mismo* individuo donde se ajustó el modelo y el modelo concluye que se trata del mismo individuo ($P \geq 0.05$) en b ocasiones, entonces su efectividad será b/N . Con lo cual se puede armar una *matriz de confusión* o *matriz de error* para cada m -ésimo individuo de estudio donde se ajuste el modelo, donde en las filas se colocan los n -ésimos individuos cuyas fotos se compararon con las del m -ésimo individuo en cuestión y en las columnas se colocan los resultados de esta comparación surgidos del modelo, es decir si se trata del mismo m -ésimo individuo o no ($m=1,2$ y $n=1,2$); cada celda vendrá dada por a_{mn}/N , se tomará la primera columna para cuando se concluya que se trata del mismo m -ésimo individuo, i.e. los elementos a_{11} y a_{21} ; como se muestra en la siguiente tabla:

Tabla 1. Matriz de error para evaluar el modelo luego de ajustarlo en el m -ésimo individuo

Individuo a comparar	El modelo concluye:	
	Mismo individuo m	Individuo distinto
Individuo 1	a_{11}/N	a_{12}/N
Individuo 2	a_{21}/N	a_{22}/N

Autor: Kevin Flores

Nótese que la suma de las filas es igual a la unidad. Se colorearán además las celdas de tal manera que en color verde se incluya el porcentaje de aciertos, es decir cuando el modelo concluye que se trata del mismo m -ésimo individuo, y en color rojo quede

el porcentaje de equivocación. Se elaborará esta matriz para cada individuo y con cada enfoque que se hizo: tres, cinco y siete puntos de referencia, tanto con registración de imágenes como sin ella.

A su vez a partir de esta matriz se define la efectividad promedio del modelo ajustado en el m -ésimo individuo a la que se llamará w_m , mediante la expresión:

$$w_m = \sum_{k=1}^2 \frac{a_{k1}}{2N} ; m = 1,2 \quad (4.1)$$

El promedio de estos indicadores para cada individuo es la *efectividad promedio* del modelo, que se denotará por w .

$$w = \frac{w_1 + w_2}{2} \quad (4.2)$$

Este indicador servirá para decir qué tan bueno es el modelo y posteriormente ver la evolución de su comportamiento.

4.2 Evaluación del modelo con tres puntos de referencia

En la sección 3.4 se vio el ajuste del modelo Procrustes con tres puntos de referencia en el individuo uno. Análogamente, en la sección 3.5 se analizó el ajuste del modelo de Procrustes con tres puntos de referencia en el individuo dos. Adicionalmente, en la sección 3.8.1 se lo ajustó nuevamente pero haciendo registración de imágenes. Las conclusiones realizadas en cada caso con los valores P obtenidos de la prueba de pertenencia permiten elaborar las matrices respectivas.

4.2.1 Sin registraci3n de im3genes

Cuando se ajust3 el modelo de Procrustes en el primer individuo y se compararon las fotograf3as del individuo dos, se obtuvieron los siguientes valores P:

```
> round(valorp_Marlon,4)
[1] 0.0602 0.1353 0.0237 0.1278 0.0335 0.0265 0.0039 0.0065
```

El modelo concluy3 que no se trataba del individuo uno en cinco de ocho ocasiones (valor $P < 0.05$), es decir su efectividad fue igual a $5/8 = 62.5\%$ y por tanto su porcentaje de error fue de 37.5% .

Cuando se compararon las fotos del mismo individuo uno, se obtuvo los valores P:

```
> round(valorp_Rebeca2,4)
[1] 0.0204 0.0411 0.1266 0.1270 0.0258 0.1188 0.1110 0.1346
```

El modelo concluy3 que se trataba del mismo individuo uno en cinco de ocho ocasiones (valor $P \geq 0.05$), es decir su efectividad fue igual a $5/8 = 62.5\%$ y por tanto su porcentaje de error fue de 37.5% .

Por tanto, la matriz correspondiente a la evaluaci3n del modelo cuando se ajust3 en el individuo uno es la siguiente:

Tabla 2. Matriz de error del modelo Procrustes con tres puntos de referencia en el primer individuo sin registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo uno	Individuo distinto
Individuo 1	62,50%	37,50%
Individuo 2	37,50%	62,50%

Autor: Kevin Flores

De manera similar, cuando se ajustó el modelo de Procrustes en el individuo dos y se compararon las fotografías del individuo uno, se obtuvo los siguientes valores P:

```
> round(valorp_Rebeca,4)
[1] 0.0232 0.0002 0.1161 0.1226 0.0020 0.1256 0.1231 0.0288
```

El modelo concluyó que no se trataba del individuo dos en cuatro de ocho ocasiones (valor $P < 0.05$), es decir su efectividad fue igual a $4/8 = 50\%$ y por tanto su porcentaje de error fue de 50%. Cuando se compararon las fotos del mismo individuo dos, se obtuvo los valores P:

```
> round(valorp_Marlon2,4)
[1] 0.0374 0.1139 0.0906 0.0460 0.1036 0.1057 0.0306 0.0495
```

El modelo concluyó que se trataba del mismo individuo dos en cuatro de ocho ocasiones (valor $P \geq 0.05$), es decir su efectividad fue igual a $4/8 = 50\%$ y por tanto su porcentaje de error fue de 50%. Entonces, la matriz correspondiente al ajuste en el segundo individuo es:

Tabla 3. Matriz de error del modelo Procrustes con tres puntos de referencia en el segundo individuo sin registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo dos	Individuo distinto
Individuo 1	50,00%	50,00%
Individuo 2	50,00%	50,00%

Autor: Kevin Flores

La efectividad promedio del modelo es entonces:

$$\frac{62.5\% + 62.5\% + 50\% + 50\%}{4} = 56.25\%$$

El cual es un valor que está por encima del 50% pero no es muy bueno, ya que se dice que un modelo es “bueno” si el indicador que mide su desempeño tiene un valor mayor o igual al 70%. De aquí en adelante solo se presentarán las matrices correspondientes, el lector podrá referirse a las respectivas secciones anteriores para su verificación con los respectivos valores P.

4.2.2 Con registraci3n de im3genes

Con registraci3n de im3genes del primer individuo, la matriz obtenida a partir del modelo de Procrustes es:

Tabla 4. Matriz de error del modelo Procrustes con tres puntos de referencia en el primer individuo con registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo uno	Individuo distinto
Individuo 1	62,50%	37,50%
Individuo 2	37,50%	62,50%

Autor: Kevin Flores

Al ajustar el modelo de Procrustes en el individuo dos con registraci3n de im3genes tenemos:

Tabla 5. Matriz de error del modelo Procrustes con tres puntos de referencia en el segundo individuo con registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo dos	Individuo distinto
Individuo 1	50,00%	50,00%
Individuo 2	50,00%	50,00%

Autor: Kevin Flores

La efectividad promedio del modelo es entonces:

$$\frac{62.5\% + 62.5\% + 50\% + 50\%}{4} = 56.25\%$$

En este caso, la efectividad es la misma que cuando no se realizó registración en las imágenes. El objetivo de hacer registración es que las varianzas de los efectos aleatorios disminuyan debido a que las imágenes ya están alineadas y así mejorar las estimaciones y por ende la efectividad promedio del modelo, aunque en este caso particular no hay una mejoría.

4.3 Evaluación del modelo con cinco puntos de referencia

Las matrices que se obtendrán serán elaboradas a partir de las conclusiones que se bosquejaron con los valores P al final de las secciones 3.6.1 y 3.6.2 para el caso de no registración de imágenes; y al final de los apartados I) y II) de la sección 3.8.2 para el caso en que se usó registración de imágenes.

4.3.1 Sin registración de imágenes

La matriz correspondiente a la evaluación del modelo cuando se ajustó en el individuo uno es la siguiente:

Tabla 6. Matriz de error del modelo Procrustes con cinco puntos de referencia en el primer individuo sin registración

Individuo a comparar	El modelo concluye:	
	Mismo individuo uno	Individuo distinto
Individuo 1	75,00%	25,00%
Individuo 2	50,00%	50,00%

Autor: Kevin Flores

La matriz correspondiente al ajuste en el segundo individuo es:

Tabla 7. Matriz de error del modelo Procrustes con cinco puntos de referencia en el segundo individuo sin registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo dos	Individuo distinto
Individuo 1	12,50%	87,50%
Individuo 2	50,00%	50,00%

Autor: Kevin Flores

La efectividad promedio del modelo es entonces:

$$\frac{75\% + 50\% + 87.5\% + 50\%}{4} = 65.63\%$$

La efectividad promedio del modelo ha aumentado con respecto al ajuste con tres puntos de referencia sin registraci3n, desde 56.25% a 65.63%.

4.3.2 Con registraci3n de im3genes

Con registraci3n de im3genes del primer individuo, la matriz obtenida a partir del modelo de Procrustes es:

Tabla 8. Matriz de error del modelo Procrustes con cinco puntos de referencia en el primer individuo con registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo uno	Individuo distinto
Individuo 1	100,00%	0,00%
Individuo 2	37,50%	62,50%

Autor: Kevin Flores

En el caso del ajuste del modelo de Procrustes en el individuo dos con registraci3n de im3genes tenemos:

Tabla 9. Matriz de error del modelo Procrustes con cinco puntos de referencia en el segundo individuo con registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo dos	Individuo distinto
Individuo 1	25,00%	75,00%
Individuo 2	62,50%	37,50%

Autor: Kevin Flores

La efectividad promedio del modelo es entonces:

$$\frac{100\% + 62.5\% + 75\% + 62.5\%}{4} = 75.00\%$$

La efectividad promedio aument3n tanto con respecto al caso cuando se ajust3n el modelo con cinco puntos pero no se hizo registraci3n de im3genes (de 65.63% a 75%), como con respecto al caso cuando se ajust3n con tres puntos de referencia y con registraci3n (56.25% a 75%). Se puede decir entonces que el usar cinco puntos de referencia es mejor que usar tres, independientemente de si se hace registraci3n o no.

4.4 Evaluaci3n del modelo con siete puntos de referencia

Las matrices que se obtendr3n ser3n elaboradas a partir de las conclusiones que se bosquejaron con los valores P al final de las secciones 3.7.1 y 3.7.2 para el caso de no registraci3n de im3genes; y al final de los apartados I) y II) de la secci3n 3.8.3 para el caso en que se us3n registraci3n de im3genes.

4.4.1 Sin registraci3n de im3genes

La matriz correspondiente a la evaluaci3n del modelo cuando se ajust3 en el individuo uno es la siguiente:

Tabla 10. Matriz de error del modelo Procrustes con siete puntos de referencia en el primer individuo sin registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo uno	Individuo distinto
Individuo 1	75,00%	25,00%
Individuo 2	50,00%	50,00%

Autor: Kevin Flores

La matriz correspondiente al ajuste en el segundo individuo es:

Tabla 11. Matriz de error del modelo Procrustes con siete puntos de referencia en el segundo individuo sin registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo dos	Individuo distinto
Individuo 1	12,50%	87,50%
Individuo 2	37,50%	62,50%

Autor: Kevin Flores

La efectividad promedio del modelo es entonces:

$$\frac{75\% + 50\% + 87.5\% + 37.5\%}{4} = 62.50\%$$

La efectividad promedio del modelo se ha reducido un poco con respecto al ajuste con cinco puntos de referencia sin registraci3n, desde 65.63% a 62.50%.

4.4.2 Con registraci3n de im3genes

Con registraci3n de im3genes del primer individuo, la matriz obtenida a partir del modelo de Procrustes es:

Tabla 12. Matriz de error del modelo Procrustes con siete puntos de referencia en el primer individuo con registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo uno	Individuo distinto
Individuo 1	87,50%	12,50%
Individuo 2	37,50%	62,50%

Autor: Kevin Flores

En el caso del ajuste del modelo de Procrustes en el individuo dos con registraci3n de im3genes tenemos:

Tabla 13. Matriz de error del modelo Procrustes con siete puntos de referencia en el segundo individuo con registraci3n

Individuo a comparar	El modelo concluye:	
	Mismo individuo dos	Individuo distinto
Individuo 1	0,00%	100,00%
Individuo 2	37,50%	62,50%

Autor: Kevin Flores

La efectividad promedio del modelo es entonces:

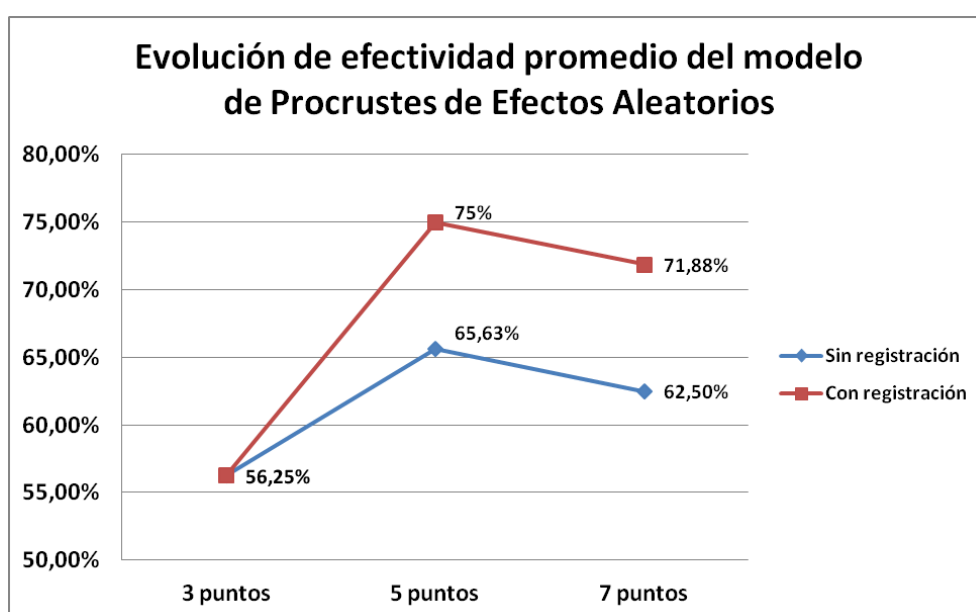
$$\frac{87.5\% + 62.5\% + 100\% + 37.5\%}{4} = 71.88\%$$

La efectividad promedio ha aumentado con respecto al caso cuando se ajust3 el modelo con siete puntos pero no se hizo registraci3n de im3genes (de 62.50% a

71.88%), pero se ha reducido con respecto al caso cuando se ajustó con cinco puntos de referencia y con registraci3n (75% a 71.88%).

4.5 Evoluci3n de la efectividad del modelo

Lo analizado anteriormente, adem1s de ser examinado desde el punto de vista de la matriz de error, puede ser planteado de una manera gr1fica. Realizando un gr1fico con la evoluci3n de la efectividad del modelo en todos los casos analizados, se tiene:



Gr1fico 25. Evoluci3n de la efectividad promedio del modelo de Procrustes de efectos aleatorios

En este gr1fico se puede observar que para el caso de tres puntos de referencia, la registraci3n de im1genes no influye en la efectividad promedio. Cuando se usan cinco puntos de referencia o siete, es mejor realizar previamente una registraci3n de im1genes antes de ajustar el modelo de Procrustes de efectos aleatorios. Sin embargo,

con nuestros individuos de estudio, es mejor usar solo cinco puntos de referencia y no siete ya que la efectividad promedio disminuye. El mejor desempeño del modelo de Procrustes de efectos aleatorios para reconocimiento facial con los individuos de estudio aquí utilizados ocurre cuando se toman cinco puntos de referencia y se realiza previamente una registración de imágenes.

5. CONCLUSIONES

1. El modelo de Procrustes de efectos aleatorios aplicado en reconocimiento facial tiene un rendimiento bueno considerando la relativa sencillez del mismo. La efectividad promedio del modelo en ningún caso que se analizó fue menor al 50%, y además usando registración de imágenes se superó la barrera del 70%.
2. Se halló una expresión general para el estimador VLS de las varianzas de los efectos aleatorios en el modelo de Procrustes, cumpliéndose así uno de los objetivos específicos. Esta expresión corresponde a la forma cuadrática (2.13) del capítulo dos cuya optimización está sujeta a las restricciones (2.14).
3. El método de VLS proporcionó estimaciones muy buenas de las varianzas del error y de los efectos aleatorios. Hubo casos en que la diferencia entre los estimadores VLS de varianza de los efectos aleatorios y los respectivos estimadores de máxima verosimilitud, era nula y sólo había diferencia al estimar la varianza del error. Esta situación es conveniente debido a que, según la complejidad de un determinado modelo lineal mixto, puede ser laborioso hallar la función de verosimilitud y quizás resulte más sencillo minimizar la distancia entre la matriz de covarianzas teórica y la empírica, que es lo que se realiza en el método de VLS.

4. El modelo de Procrustes de efectos aleatorios se ha ajustado muy bien en los individuos analizados, esto se lo pudo observar en el momento en que se estimaron los vectores que contienen las varianzas de los efectos aleatorios y del error. La varianza del error (primera componente) es muy pequeña en relación a la varianza de cualquiera de los efectos aleatorios; lo cual es muy bueno porque significa que los efectos aleatorios tal cual se definieron han capturado buena parte de la variabilidad total de las observaciones. Si se hubiera ajustado un modelo lineal general, sin efectos aleatorios, toda esta variabilidad hubiese sido incluida en la varianza del error.

5. La efectividad promedio del modelo aumentó al incrementar la cantidad de puntos de referencia de tres a cinco, sin embargo disminuyó ligeramente al usar siete puntos de referencia, concretamente un 5% cuando no se hizo registración de imágenes y un 4.3% cuando sí la hubo. Aunque en ambos casos, usando tanto cinco como siete puntos de referencia la efectividad promedio se mantuvo por encima del 60% sin importar si hubo registración o no. Podríamos decir entonces que el modelo de Procrustes de efectos aleatorios aplicado en reconocimiento facial tiene una efectividad promedio aceptable cuando se usan muchos puntos de referencia, y podría aumentar incluso si se usaran más puntos como por ejemplo diez aumentando desde luego el número de observaciones.

6. La registraci3n de im3genes caus3 un aumento en la efectividad promedio del modelo de Procrustes de efectos aleatorios, a excepci3n del caso de tres puntos de referencia donde se mantuvo en el mismo 56.25%. Esto puede estar relacionado al hecho de que estos tres puntos de referencia eran los ojos y la punta de la nariz, los cuales son fijos en las personas sin importar la expresi3n facial. Cuando se usaron cinco o siete puntos de referencia se tomaron puntos notables en la boca, la cual s3 cambia con la expresi3n facial del individuo y la registraci3n de im3genes aument3 la efectividad del modelo en un 14% y 15%, respectivamente. Se puede decir entonces que hacer registraci3n de las im3genes de una persona antes de ajustar el modelo de Procrustes de efectos aleatorios mejora la efectividad del mismo.

6. RECOMENDACIONES

1. Aumentar la cantidad de puntos de referencia si se quiere mayor efectividad en el modelo. Además, tratar de que estos puntos incluyan no solo puntos fijos sino también puntos que no lo sean en las distintas expresiones faciales que se analicen, sino que sean “móviles” y varíen con las mismas.

2. Se debe cuidar ciertos parámetros para que las fotografías que se usen como datos de entrada estén tomadas en similares condiciones, como por ejemplo que la cámara esté aproximadamente a la misma distancia y los rostros estén todos en posición frontal, con el fin de reducir la variabilidad de los efectos aleatorios. De momento no es mandatorio las mismas condiciones de iluminación, color de fondo, uso de flash, etc ya que la identificación de los puntos de referencia se hizo de manera manual.

3. Aumentar la cantidad de observaciones (fotografías) de cada individuo de estudio para que mejoren las estimaciones de los parámetros del modelo y por tanto la efectividad del mismo. Esta recomendación es especialmente mandatoria si se usan muchos puntos de referencia, debido a que estadísticamente las estimaciones son mejores cuando hay más observaciones que variables. En particular, la distribución del estadístico de prueba de la prueba de pertenencia será más cercana a una ji-

cuadrado conforme aumenta el número de observaciones, sin importar que no se use la matriz de varianzas y covarianzas real $V(\boldsymbol{\mu})$ sino su estimación.

4. Como se dijo antes, es mejor hacer primero una registración de imágenes. En este trabajo solo se usó el método de registración afín, el cual es del tipo lineal. No obstante, se podrían realizar otros métodos no lineales, los cuales podrían mejorar aún más la efectividad promedio del modelo. Dos de estos métodos son el de registración *rígida* y *deformable* y la registración *estocástica*.

5. Se podría obtener una mayor efectividad en el modelo de Procrustes de efectos aleatorios al momento de hacer reconocimiento facial, si los puntos de referencia son obtenidos de manera automática a partir de las fotografías. Para tal efecto, los *filtros de Gabor* que se mencionaron muy brevemente en la introducción son útiles. No obstante, esta labor podría ser incluso más laboriosa que la tarea de reconocimiento facial propiamente dicha, por lo que requeriría un trabajo de investigación aparte.

6. Se pueden arreglar las distribuciones de r_i y θ_i para disminuir el número de parámetros a estimar, lo cual ayuda mucho cuando hay pocas observaciones. Por ejemplo, cuando r_i^2 es exponencial y θ_i es uniforme entre $[0, 2\pi]$ se puede probar que $\text{corr}(h_i, g_i) = \text{corr}(r_i \cos \theta_i, r_i \sin \theta_i) = 0$ con lo cual tendríamos $\sigma_{hg} = 0$.

ANEXOS

I. Demostración de la varianza del vector \mathbf{u}_i

Demostrar que:

$$\text{Var}(\mathbf{u}_i) = \sigma^2 \mathbf{V}(\boldsymbol{\mu}) \quad (2.8)$$

Donde:

$$\begin{aligned} \mathbf{V}(\boldsymbol{\mu}) &= \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T + \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\ &\quad + \sigma_{hg} [\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T] \\ &\quad + \tau_1^2 (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) + \tau_2^2 (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) + \tau_{12} (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) + \mathbf{I}_{2n} \end{aligned} \quad (2.9)$$

Con:

$$\begin{aligned} \mathbf{L} &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} & \mathbf{A}_2 &= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ \mathbf{A}_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} & \mathbf{A}_{12} &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{aligned}$$

Prueba:

$$\begin{aligned} \text{Var}(\mathbf{u}_i) &= \text{Var}(h_i \boldsymbol{\mu} + g_i (\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu} + \mathbf{t}_i \otimes \mathbf{1} + \boldsymbol{\varepsilon}_i) \\ &= \text{Var}(h_i \boldsymbol{\mu}) + \text{Var}[g_i (\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] + \text{Cov}(h_i \boldsymbol{\mu}, g_i (\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}) + \text{Var}(\mathbf{t}_i \otimes \mathbf{1}) \\ &\quad + \text{Var}(\boldsymbol{\varepsilon}_i) \\ &= \sigma^2 \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T + \sigma^2 \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\ &\quad + \sigma^2 \sigma_{hg} [\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T] + \sigma^2 (\mathbf{T} \otimes \mathbf{1} \mathbf{1}^T) + \sigma^2 \mathbf{I}_{2n} \end{aligned}$$

Como:

$$\begin{aligned}
\mathbf{T} &= \begin{bmatrix} \tau_1^2 & \tau_{12} \\ \tau_{12} & \tau_2^2 \end{bmatrix} = \begin{bmatrix} \tau_1^2 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \tau_2^2 \end{bmatrix} + \begin{bmatrix} 0 & \tau_{12} \\ \tau_{12} & 0 \end{bmatrix} \\
&= \tau_1^2 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \tau_2^2 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} + \tau_{12} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\
&= \tau_1^2 \mathbf{A}_1 + \tau_2^2 \mathbf{A}_2 + \tau_{12} \mathbf{A}_{12}
\end{aligned}$$

Entonces:

$$\begin{aligned}
\text{Var}(\mathbf{u}_i) &= \sigma^2 \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T + \sigma^2 \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
&\quad + \sigma^2 \sigma_{hg} [\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T] + \sigma^2 (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \sigma^2 (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) + \sigma^2 (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) + \sigma^2 \mathbf{I}_{2n} \\
&= \sigma^2 \mathbf{V}(\boldsymbol{\mu})
\end{aligned}$$

II. Demostración del estimador VLS para el modelo de Procrustes.

Demostrar que la expresión:

$$\min \sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i - \sigma^2 \mathbf{V}(\boldsymbol{\mu}))^2 \quad (2.11)$$

Donde $\mathbf{V}(\boldsymbol{\mu})$ viene dada por la ecuación (2.9) del capítulo 2, es equivalente a minimizar la forma cuadrática:

$$N \boldsymbol{\theta}^T \mathbf{M} \boldsymbol{\theta} - \mathbf{m}^T \boldsymbol{\theta} \quad (2.13)$$

Sujeto a las restricciones:

$$\begin{aligned}\gamma_k &\geq 0, \quad k = \{1,2,4,5,7\} \\ \gamma_{k-2}\gamma_{k-1} - \gamma_k^2 &\geq 0, \quad k = \{3,6\}\end{aligned}\tag{2.14}$$

Donde el vector $\boldsymbol{\theta} = (\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6, \gamma_7)^T$ representa la reparametrización:

$$\begin{aligned}\gamma_1 &= \sigma^2 \sigma_h^2 \\ \gamma_2 &= \sigma^2 \sigma_g^2 \\ \gamma_3 &= \sigma^2 \sigma_{hg} \\ \gamma_4 &= \sigma^2 \tau_1^2 \\ \gamma_5 &= \sigma^2 \tau_2^2 \\ \gamma_6 &= \sigma^2 \tau_{12} \\ \gamma_7 &= \sigma^2\end{aligned}\tag{2.12}$$

Prueba:

Tenemos:

$$\begin{aligned}\sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i - \sigma^2 \mathbf{V}(\boldsymbol{\mu}))^2 &= \sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i - 2\sigma^2 \hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i \mathbf{V}(\boldsymbol{\mu}) + \sigma^4 \mathbf{V}(\boldsymbol{\mu}) \mathbf{V}(\boldsymbol{\mu})) \\ &= \sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i)^2 - 2\sigma^2 \sum_{i=1}^N \text{tr}(\hat{\mathbf{e}}_i^T \mathbf{V}(\boldsymbol{\mu}) \hat{\mathbf{e}}_i) + \sigma^4 \sum_{i=1}^N \text{tr}(\mathbf{V}(\boldsymbol{\mu}) \mathbf{V}(\boldsymbol{\mu})) \\ &= \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i)^2 - 2\sigma^2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \mathbf{V}(\boldsymbol{\mu}) \hat{\mathbf{e}}_i) + \sigma^4 \sum_{i=1}^N \text{tr}(\mathbf{V}(\boldsymbol{\mu}) \mathbf{V}(\boldsymbol{\mu}))\end{aligned}$$

Es fácil probar que:

$$\begin{aligned}
\hat{\mathbf{e}}_i^T \mathbf{V}(\boldsymbol{\mu}) \hat{\mathbf{e}}_i &= \sigma_h^2 \hat{\mathbf{e}}_i^T \boldsymbol{\mu} \boldsymbol{\mu}^T \hat{\mathbf{e}}_i + \sigma_g^2 \hat{\mathbf{e}}_i^T [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \hat{\mathbf{e}}_i \\
&\quad + \sigma_{hg} \hat{\mathbf{e}}_i^T (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \hat{\mathbf{e}}_i \\
&\quad + \tau_1^2 \hat{\mathbf{e}}_i^T (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \hat{\mathbf{e}}_i + \tau_2^2 \hat{\mathbf{e}}_i^T (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) \hat{\mathbf{e}}_i \\
&\quad + \tau_{12} \hat{\mathbf{e}}_i^T (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) \hat{\mathbf{e}}_i + \hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i
\end{aligned}$$

Con lo cual el término del centro es equivalente a:

$$2\sigma^2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \mathbf{V}(\boldsymbol{\mu}) \hat{\mathbf{e}}_i) = \mathbf{m}^T \boldsymbol{\theta}$$

Con:

$$\boldsymbol{\theta} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \gamma_4 \\ \gamma_5 \\ \gamma_6 \\ \gamma_7 \end{pmatrix} \qquad \mathbf{m} = \begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \\ m_6 \\ m_7 \end{pmatrix}$$

$$m_1 = 2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \boldsymbol{\mu})^2$$

$$m_2 = 2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}])^2$$

$$m_3 = 4 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \boldsymbol{\mu}) (\hat{\mathbf{e}}_i^T [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}])$$

$$m_4 = 2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \hat{\mathbf{e}}_i)$$

$$m_5 = 2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) \hat{\mathbf{e}}_i)$$

$$m_6 = 2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) \hat{\mathbf{e}}_i)$$

$$m_7 = 2 \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i)$$

Antes de expresar el último término en función de los nuevos parámetros, podemos verificar primero que:

$$\begin{aligned}
V(\boldsymbol{\mu})V(\boldsymbol{\mu}) &= \sigma_h^4 \boldsymbol{\mu} \boldsymbol{\mu}^T \boldsymbol{\mu} \boldsymbol{\mu}^T + \sigma_g^2 \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
&\quad + \sigma_{hg} \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) + \tau_1^2 \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \tau_2^2 \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) + \tau_{12} \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \sigma_h^2 \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \boldsymbol{\mu} \boldsymbol{\mu}^T \\
&\quad + \sigma_g^4 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
&\quad + \sigma_{hg} \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \\
&\quad + \tau_1^2 \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \tau_2^2 \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \tau_{12} \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \sigma_h^2 \sigma_{hg} (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \boldsymbol{\mu} \boldsymbol{\mu}^T \\
&\quad + \sigma_g^2 \sigma_{hg} (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
&\quad + \sigma_{hg} (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \\
&\quad \quad + \tau_1^2 \sigma_{hg} (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \tau_2^2 \sigma_{hg} (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) (\mathbf{A}_2 \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \tau_{12} \sigma_{hg} (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) (\mathbf{A}_{12} \otimes \mathbf{1} \mathbf{1}^T) \\
&\quad + \sigma_h^2 \tau_1^2 (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) \boldsymbol{\mu} \boldsymbol{\mu}^T + \sigma_g^2 \tau_1^2 (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
&\quad + \sigma_{hg} \tau_1^2 (\mathbf{A}_1 \otimes \mathbf{1} \mathbf{1}^T) (\boldsymbol{\mu} [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T)
\end{aligned}$$

$$\begin{aligned}
& +\tau_1^4(\mathbf{A}_1 \otimes \mathbf{11}^T)(\mathbf{A}_1 \otimes \mathbf{11}^T) + \tau_2^2 \tau_1^2(\mathbf{A}_1 \otimes \mathbf{11}^T)(\mathbf{A}_2 \otimes \mathbf{11}^T) \\
& \quad + \tau_{12} \tau_1^2(\mathbf{A}_1 \otimes \mathbf{11}^T)(\mathbf{A}_{12} \otimes \mathbf{11}^T) + \sigma_h^2 \tau_2^2(\mathbf{A}_2 \otimes \mathbf{11}^T) \boldsymbol{\mu} \boldsymbol{\mu}^T \\
& \quad + \sigma_g^2 \tau_2^2(\mathbf{A}_2 \otimes \mathbf{11}^T)[(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}][(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
& \quad + \sigma_{hg} \tau_2^2(\mathbf{A}_2 \otimes \mathbf{11}^T)(\boldsymbol{\mu}[(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \\
& +\tau_1^2 \tau_2^2(\mathbf{A}_2 \otimes \mathbf{11}^T)(\mathbf{A}_1 \otimes \mathbf{11}^T) + \tau_2^4(\mathbf{A}_2 \otimes \mathbf{11}^T)(\mathbf{A}_2 \otimes \mathbf{11}^T) \\
& \quad + \tau_{12} \tau_2^2(\mathbf{A}_2 \otimes \mathbf{11}^T)(\mathbf{A}_{12} \otimes \mathbf{11}^T) + \sigma_h^2 \tau_{12}(\mathbf{A}_{12} \otimes \mathbf{11}^T) \boldsymbol{\mu} \boldsymbol{\mu}^T \\
& \quad + \sigma_g^2 \tau_{12}(\mathbf{A}_{12} \otimes \mathbf{11}^T)[(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}][(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
& \quad + \sigma_{hg} \tau_{12}(\mathbf{A}_{12} \otimes \mathbf{11}^T)(\boldsymbol{\mu}[(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \\
& +\tau_1^2 \tau_{12}(\mathbf{A}_{12} \otimes \mathbf{11}^T)(\mathbf{A}_1 \otimes \mathbf{11}^T) + \tau_2^2 \tau_{12}(\mathbf{A}_{12} \otimes \mathbf{11}^T)(\mathbf{A}_2 \otimes \mathbf{11}^T) \\
& \quad + \tau_{12}^2(\mathbf{A}_{12} \otimes \mathbf{11}^T)(\mathbf{A}_{12} \otimes \mathbf{11}^T) + \sigma_h^2 \boldsymbol{\mu} \boldsymbol{\mu}^T + \sigma_g^2 [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}][(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T \\
& \quad + \sigma_{hg}(\boldsymbol{\mu}[(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}]^T + [(\mathbf{L} \otimes \mathbf{I}_n) \boldsymbol{\mu}] \boldsymbol{\mu}^T) \\
& +\tau_1^2(\mathbf{A}_1 \otimes \mathbf{11}^T) + \tau_2^2(\mathbf{A}_2 \otimes \mathbf{11}^T) + \tau_{12}(\mathbf{A}_{12} \otimes \mathbf{11}^T) + \mathbf{I}_{2n}
\end{aligned}$$

Y sabiendo que:

$$\text{tr}(\mathbf{A}_1 \otimes \mathbf{11}^T) = n$$

$$\text{tr}(\mathbf{A}_1 \otimes \mathbf{11}^T)^2 = n^2$$

$$\text{tr}(\mathbf{A}_2 \otimes \mathbf{11}^T) = n$$

$$\text{tr}(\mathbf{A}_2 \otimes \mathbf{11}^T)^2 = n^2$$

$$\text{tr}(\mathbf{A}_{12} \otimes \mathbf{11}^T) = 0$$

$$\text{tr}(\mathbf{A}_{12} \otimes \mathbf{11}^T)^2 = 2n^2$$

Se comprueba que:

$$\begin{aligned}
tr(\mathbf{V}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu})) &= \sigma_h^4(\boldsymbol{\mu}^T\boldsymbol{\mu})^2 + \sigma_g^4[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^2 \\
&+ 2\sigma_{hg}^2\left[(\boldsymbol{\mu}^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}])^2 + \boldsymbol{\mu}^T\boldsymbol{\mu}[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]\right] \\
&+ 2\sigma_h^2\sigma_g^2(\boldsymbol{\mu}^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T)^2 + 4\sigma_h^2\sigma_{hg}\boldsymbol{\mu}^T\boldsymbol{\mu}[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T\boldsymbol{\mu} \\
&+ 4\sigma_g^2\sigma_{hg}\boldsymbol{\mu}^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}][(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] + 2\sigma_h^2\boldsymbol{\mu}^T\boldsymbol{\mu} \\
&+ 2\sigma_g^2[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] + 4\sigma_{hg}\boldsymbol{\mu}^T[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] \\
&+ 2\sigma_h^2\tau_1^2\boldsymbol{\mu}^T(\mathbf{A}_1\otimes\mathbf{1}\mathbf{1}^T)\boldsymbol{\mu} + 2\sigma_h^2\tau_2^2\boldsymbol{\mu}^T(\mathbf{A}_2\otimes\mathbf{1}\mathbf{1}^T)\boldsymbol{\mu} \\
&+ 2\sigma_h^2\tau_{12}\boldsymbol{\mu}^T(\mathbf{A}_{12}\otimes\mathbf{1}\mathbf{1}^T)\boldsymbol{\mu} + 2\sigma_g^2\tau_1^2[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T(\mathbf{A}_1\otimes\mathbf{1}\mathbf{1}^T)[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] \\
&+ 2\sigma_g^2\tau_2^2[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T(\mathbf{A}_2\otimes\mathbf{1}\mathbf{1}^T)[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] \\
&+ 2\sigma_g^2\tau_{12}[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}]^T(\mathbf{A}_{12}\otimes\mathbf{1}\mathbf{1}^T)[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] \\
&+ 4\sigma_{hg}\tau_1^2\boldsymbol{\mu}^T(\mathbf{A}_1\otimes\mathbf{1}\mathbf{1}^T)[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] \\
&+ 4\sigma_{hg}\tau_2^2\boldsymbol{\mu}^T(\mathbf{A}_2\otimes\mathbf{1}\mathbf{1}^T)[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] \\
&+ 4\sigma_{hg}\tau_{12}\boldsymbol{\mu}^T(\mathbf{A}_{12}\otimes\mathbf{1}\mathbf{1}^T)[(\mathbf{L}\otimes\mathbf{I}_n)\boldsymbol{\mu}] + n\tau_1^2 + n\tau_2^2 + n^2\tau_1^4 + n^2\tau_2^4 \\
&+ 2n^2\tau_{12}^2 + 2n
\end{aligned}$$

Con lo cual se puede demostrar que:

$$\sigma^4 \sum_{i=1}^N tr(\mathbf{V}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu})) = N\boldsymbol{\theta}^T \mathbf{M} \boldsymbol{\theta}$$

Donde:

$$\mathbf{M} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} & m_{17} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} & m_{27} \\ m_{31} & m_{32} & m_{33} & m_{34} & m_{35} & m_{36} & m_{37} \\ m_{41} & m_{42} & m_{43} & m_{44} & m_{45} & m_{46} & m_{47} \\ m_{51} & m_{52} & m_{53} & m_{54} & m_{55} & m_{56} & m_{57} \\ m_{61} & m_{62} & m_{63} & m_{64} & m_{65} & m_{66} & m_{67} \\ m_{71} & m_{72} & m_{73} & m_{74} & m_{75} & m_{76} & m_{77} \end{pmatrix}$$

Con:

$$m_{11} = (\boldsymbol{\mu}^T \boldsymbol{\mu})^2$$

$$m_{12} = (\boldsymbol{\mu}^T [(L \otimes I_n) \boldsymbol{\mu}])^2$$

$$m_{13} = 2\boldsymbol{\mu}^T \boldsymbol{\mu} [(L \otimes I_n) \boldsymbol{\mu}]^T \boldsymbol{\mu}$$

$$m_{14} = \boldsymbol{\mu}^T (A_1 \otimes \mathbf{1}\mathbf{1}^T) \boldsymbol{\mu}$$

$$m_{15} = \boldsymbol{\mu}^T (A_2 \otimes \mathbf{1}\mathbf{1}^T) \boldsymbol{\mu}$$

$$m_{16} = \boldsymbol{\mu}^T (A_{12} \otimes \mathbf{1}\mathbf{1}^T) \boldsymbol{\mu}$$

$$m_{17} = \boldsymbol{\mu}^T \boldsymbol{\mu}$$

$$m_{22} = [(L \otimes I_n) \boldsymbol{\mu}]^T [(L \otimes I_n) \boldsymbol{\mu}]^2$$

$$m_{23} = 2\boldsymbol{\mu}^T [(L \otimes I_n) \boldsymbol{\mu}] [(L \otimes I_n) \boldsymbol{\mu}]^T [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{24} = [(L \otimes I_n) \boldsymbol{\mu}]^T (A_1 \otimes \mathbf{1}\mathbf{1}^T) [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{25} = [(L \otimes I_n) \boldsymbol{\mu}]^T (A_2 \otimes \mathbf{1}\mathbf{1}^T) [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{26} = [(L \otimes I_n) \boldsymbol{\mu}]^T (A_{12} \otimes \mathbf{1}\mathbf{1}^T) [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{27} = [(L \otimes I_n) \boldsymbol{\mu}]^T [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{33} = 2 \left[(\boldsymbol{\mu}^T [(L \otimes I_n) \boldsymbol{\mu}])^2 + \boldsymbol{\mu}^T \boldsymbol{\mu} [(L \otimes I_n) \boldsymbol{\mu}]^T [(L \otimes I_n) \boldsymbol{\mu}] \right]$$

$$m_{34} = 2\boldsymbol{\mu}^T (A_1 \otimes \mathbf{1}\mathbf{1}^T) [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{35} = 2\boldsymbol{\mu}^T (A_2 \otimes \mathbf{1}\mathbf{1}^T) [(L \otimes I_n) \boldsymbol{\mu}]$$

$$m_{36} = 2\boldsymbol{\mu}^T (\mathbf{A}_{12} \otimes \mathbf{1}\mathbf{1}^T) [(\mathbf{L} \otimes \mathbf{I}_n)\boldsymbol{\mu}]$$

$$m_{37} = 2\boldsymbol{\mu}^T [(\mathbf{L} \otimes \mathbf{I}_n)\boldsymbol{\mu}]$$

$$m_{44} = n^2$$

$$m_{45} = 0$$

$$m_{46} = 0$$

$$m_{47} = n$$

$$m_{55} = n^2$$

$$m_{56} = 0$$

$$m_{57} = n$$

$$m_{66} = 2n^2$$

$$m_{67} = 0$$

$$m_{77} = 2n$$

Para todos los demás elementos se cumple que $m_{ij} = m_{ji}$

Con lo cual el problema original equivale a hallar $\boldsymbol{\theta}$ tal que:

$$\min(N\boldsymbol{\theta}^T \mathbf{C} \boldsymbol{\theta} - \mathbf{c}^T \boldsymbol{\theta} + \mathbf{d})$$

Donde \mathbf{d} es una constante dada por:

$$\mathbf{d} = \sum_{i=1}^N (\hat{\mathbf{e}}_i^T \hat{\mathbf{e}}_i)^2$$

Restricciones de varianzas y covarianzas en términos de los nuevos parámetros:

$$\sigma_h^2 \geq 0 \rightarrow \sigma^2 \sigma_h^2 \geq 0 \rightarrow \gamma_1 \geq 0$$

$$\sigma_g^2 \geq 0 \rightarrow \sigma^2 \sigma_g^2 \geq 0 \rightarrow \gamma_2 \geq 0$$

$$\tau_1^2 \geq 0 \rightarrow \sigma^2 \tau_1^2 \geq 0 \rightarrow \gamma_4 \geq 0$$

$$\tau_2^2 \geq 0 \rightarrow \sigma^2 \tau_2^2 \geq 0 \rightarrow \gamma_5 \geq 0$$

$$\sigma^2 \geq 0 \rightarrow \gamma_7 \geq 0$$

$$|\sigma_{hg}| \leq \sigma_h \sigma_g \rightarrow \sigma^4 \sigma_{hg}^2 \leq (\sigma^2 \sigma_h^2)(\sigma^2 \sigma_g^2) \rightarrow \gamma_3^2 \leq \gamma_1 \gamma_2$$

$$|\tau_{12}| \leq \tau_1 \tau_2 \rightarrow \sigma^4 \tau_{12}^2 \leq (\sigma^2 \tau_1^2)(\sigma^2 \tau_2^2) \rightarrow \gamma_6^2 \leq \gamma_4 \gamma_5$$

Por lo tanto, el problema original equivale a hallar el vector de parámetros $\boldsymbol{\theta}$ tal que se minimice la forma cuadrática:

$$\min(N\boldsymbol{\theta}^T \mathbf{M} \boldsymbol{\theta} - \mathbf{m}^T \boldsymbol{\theta})$$

Sujeto a las restricciones:

$$\gamma_k \geq 0 \quad , k = \{1,2,4,5,7\}$$

$$\gamma_{k-2} \gamma_{k-1} - \gamma_k^2 \geq 0 \quad , k = \{3,6\}$$

III. Script con la codificación en R

```
#####
####Registración de Imágenes
library(RNiftyReg)
library(oro.nifti)
library(EBImage)

#Lectura Fotos de Marlon
a1=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon1.jpg")[, , 1]
a2=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon2.jpg")[, , 1]
a3=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon3.jpg")[, , 1]
a4=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon4.jpg")[, , 1]
a5=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon5.jpg")[, , 1]
a6=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon6.jpg")[, , 1]
a7=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon7.jpg")[, , 1]
a8=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon8.jpg")[, , 1]
a9=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Marlon9.jpg")[, , 1]
a10=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nueva
s fotos 2/Marlon10.jpg")[, , 1]

#Convirtiendo a formato nifti
#Imágenes objetivo
a=as.nifti(array(c(a2,a3,a4,a5,a6,a7,a8,a9,a10),dim=c(dim(a2),9)))
#Imagen de origen
b=as.nifti(array(c(a1),dim=dim(a1)))
#Visualizacion de imagenes
image(a)
image(b)
#Registracion
z=niftyreg.linear(a,b)
#Visualizacion de imagenes registradas
image(z$image)

#Guardar imagenes registradas
writeNIFTI(b,"Marlon1",gzipped=FALSE)
writeNIFTI(z$image[, , 1],"Marlon2",gzipped=FALSE)
writeNIFTI(z$image[, , 2],"Marlon3",gzipped=FALSE)
writeNIFTI(z$image[, , 3],"Marlon4",gzipped=FALSE)
writeNIFTI(z$image[, , 4],"Marlon5",gzipped=FALSE)
writeNIFTI(z$image[, , 5],"Marlon6",gzipped=FALSE)
writeNIFTI(z$image[, , 6],"Marlon7",gzipped=FALSE)
writeNIFTI(z$image[, , 7],"Marlon8",gzipped=FALSE)
writeNIFTI(z$image[, , 8],"Marlon9",gzipped=FALSE)
```

```

writeNIFTI(z$image[,,9], "Marlon10", gzipped=FALSE)

#Lectura de Fotos Rebeca
c1=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca1.jpg")[,,1]
c2=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca2.jpg")[,,1]
c3=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca3.jpg")[,,1]
c4=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca4.jpg")[,,1]
c5=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca5.jpg")[,,1]
c6=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca6.jpg")[,,1]
c7=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca7.jpg")[,,1]
c8=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca8.jpg")[,,1]
c9=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nuevas
fotos 2/Rebeca9.jpg")[,,1]
c10=readImage("H:/Usuarios/kflores/Documents/Espol/Tesis/Tesis/Nueva
s fotos 2/Rebeca10.jpg")[,,1]

#Imágenes objetivo
c=as.nifti(array(c(c2,c3,c4,c5,c6,c7,c8,c9,c10), dim=c(dim(c2),9)))
#Imágenes de origen
d=as.nifti(array(c(c1), dim=dim(c1)))
#Visualización de imágenes
image(c)
image(d)
#Registración
z2=niftyreg.linear(c,d)
#Visualización de imágenes registradas
image(z2$image)

#Guardar imágenes registradas
writeNIFTI(d, "Rebeca1", gzipped=FALSE)
writeNIFTI(z2$image[,,1], "Rebeca2", gzipped=FALSE)
writeNIFTI(z2$image[,,2], "Rebeca3", gzipped=FALSE)
writeNIFTI(z2$image[,,3], "Rebeca4", gzipped=FALSE)
writeNIFTI(z2$image[,,4], "Rebeca5", gzipped=FALSE)
writeNIFTI(z2$image[,,5], "Rebeca6", gzipped=FALSE)
writeNIFTI(z2$image[,,6], "Rebeca7", gzipped=FALSE)
writeNIFTI(z2$image[,,7], "Rebeca8", gzipped=FALSE)
writeNIFTI(z2$image[,,8], "Rebeca9", gzipped=FALSE)
writeNIFTI(z2$image[,,9], "Rebeca10", gzipped=FALSE)

#####
###Ajuste del modelo Procrustes
library(alabama)
library(EBImage)
n=3 #Numero de puntos de referencia por imagen: 3, 5 o 7

```

```

#Rebeca = Individuo 1
#Marlon = Individuo 2
#####
#Apertura de fotos Rebeca y uso de locator
persona="Rebeca%i.png"
lista=NULL
for(i in 1:10){
  file=sprintf(persona,i)
  foto=readImage(file)
  display(foto,method="raster")
  lista=c(lista,list(locator(n)))
}
saveRDS(lista,"Rebeca.rds")
#saveRDS(lista,"Rebeca_5p.rds") <- Ejecutar esta linea si n=5
#saveRDS(lista,"Rebeca_7p.rds") <- Ejecutar esta linea si n=7

#Fotos de Marlon
persona2="Marlon%i.png"
lista2=NULL
for(i in 1:10){
  file2=sprintf(persona2,i)
  foto2=readImage(file2)
  display(foto2,method="raster")
  lista2=c(lista2,list(locator(n)))
}
saveRDS(lista2,"Marlon.rds")
#saveRDS(lista2,"Marlon_5p.rds") <- Ejecutar esta linea si n=5
#saveRDS(lista2,"Marlon_7p.rds") <- Ejecutar esta linea si n=7

#Apertura de los archivos tipo lista que se originaron si n=3
lista=readRDS("Rebeca.rds")
lista2=readRDS("Marlon.rds")
#####
#Apertura de los archivos tipo lista que se originaron si n=5
lista=readRDS("Rebeca_5p.rds")
lista2=readRDS("Marlon_5p.rds")
#####
#Apertura de los archivos tipo lista que se originaron si n=7
lista=readRDS("Rebeca_7p.rds")
lista2=readRDS("Marlon_7p.rds")
#####
#Quitando foto siete y nueve
lista=lista[-c(7,9)]
lista2=lista2[-c(7,9)]

#Funcion para convertir la lista en matriz
arreglar=function(i){
  res=c(i$x,i$y)
  names(res)=c(paste("x",1:n,sep=""),paste("y",1:n,sep=""))
  res
}
#Aplicando esta funcion en nuestros archivos tipo lista
datos_Rebeca=sapply(lista,arreglar)
datos_Marlon=sapply(lista2,arreglar)

```

```

#Mostrar datos
round(datos_Rebeca,4)
round(datos_Marlon,4)

#Calculando el vector promedio
mu=rowMeans(datos_Rebeca)
mu2=rowMeans(datos_Marlon)

#####Usar esta parte para graficar con tres puntos de referencia
#Graficar a Rebeca
range(sapply(lista,function(a)a$x))
range(sapply(lista,function(a)a$y))
plot(c(.35,.60),c(.3,.75),main="Individuo
1",cex.lab=2,cex.axis=1.75,
      type="n",cex.main=2,xlab="x",ylab="y")
lapply(lista,points,col=1:3,pch=15:17,cex=1.5)
points(mu[4:6]~mu[1:3],col="purple",pch=19,cex=1.5)
lines(x=mu[1:3],y=mu[4:6],col="purple",lwd=2)
segments(x0=mu[1],x1=mu[3],y0=mu[4],y1=mu[6],col="purple",lwd=2)

#Graficar a Marlon
range(sapply(lista2,function(a)a$x))
range(sapply(lista2,function(a)a$y))
plot(c(.35,.55),c(.25,.75),main="Individuo
2",cex.lab=2,cex.axis=1.75,
      type="n",cex.main=2,xlab="x",ylab="y")
lapply(lista2,points,col=1:3,pch=15:17,cex=1.5)
points(mu2[4:6]~mu2[1:3],col="purple",pch=19,cex=1.5)
lines(x=mu2[1:3],y=mu2[4:6],col="purple",lwd=2)
segments(x0=mu2[1],x1=mu2[3],y0=mu2[4],y1=mu2[6],col="purple",lwd=2)
#####

#####Usar esta parte para graficar con cinco puntos de referencia
#Graficar a Rebeca
range(sapply(lista,function(a)a$x))
range(sapply(lista,function(a)a$y))
plot(c(.35,.60),c(0,.72),main="Individuo 1",cex.lab=2,cex.axis=1.75,
      type="n",cex.main=2,xlab="x",ylab="y")
lapply(lista,points,col=1:5,pch=15:19,cex=1.5)
points(mu[6:10]~mu[1:5],col="purple",pch=19,cex=1.5)
segments(x0=mu[1],x1=mu[2],y0=mu[6],y1=mu[7],col="purple",lwd=2)
segments(x0=mu[2],x1=mu[4],y0=mu[7],y1=mu[9],col="purple",lwd=2)
segments(x0=mu[1],x1=mu[3],y0=mu[6],y1=mu[8],col="purple",lwd=2)
segments(x0=mu[4],x1=mu[5],y0=mu[9],y1=mu[10],col="purple",lwd=2)
segments(x0=mu[3],x1=mu[5],y0=mu[8],y1=mu[10],col="purple",lwd=2)

#Graficar a Marlon
range(sapply(lista2,function(a)a$x))
range(sapply(lista2,function(a)a$y))
plot(c(.35,.55),c(0,.75),main="Individuo 2",cex.lab=2,cex.axis=1.75,
      type="n",cex.main=2,xlab="x",ylab="y")
lapply(lista2,points,col=1:5,pch=15:19,cex=1.5)
points(mu2[6:10]~mu2[1:5],col="purple",pch=19,cex=1.5)
segments(x0=mu2[1],x1=mu2[2],y0=mu2[6],y1=mu2[7],col="purple",lwd=2)

```

```

segments(x0=mu2[2],x1=mu2[4],y0=mu2[7],y1=mu2[9],col="purple",lwd=2)
segments(x0=mu2[1],x1=mu2[3],y0=mu2[6],y1=mu2[8],col="purple",lwd=2)
segments(x0=mu2[4],x1=mu2[5],y0=mu2[9],y1=mu2[10],col="purple",lwd=2
)
segments(x0=mu2[3],x1=mu2[5],y0=mu2[8],y1=mu2[10],col="purple",lwd=2
)
#####

#####Usar esta parte para graficar con siete puntos de referencia
#Graficar a Rebeca
range(sapply(lista,function(a)a$x))
range(sapply(lista,function(a)a$y))
plot(c(.35,.60),c(0,.75),main="Individuo 1",cex.lab=2,cex.axis=1.75,
type="n",cex.main=2,xlab="x",ylab="y")
lapply(lista,points,col=c(1:4,"deepskyblue","darkmagenta","orange"),
pch=15:19,cex=1.5)
points(mu[8:14]~mu[1:7],col="purple",pch=19,cex=1.5)
segments(x0=mu[1],x1=mu[2],y0=mu[8],y1=mu[9],col="purple",lwd=2)
segments(x0=mu[2],x1=mu[4],y0=mu[9],y1=mu[11],col="purple",lwd=2)
segments(x0=mu[1],x1=mu[3],y0=mu[8],y1=mu[10],col="purple",lwd=2)
segments(x0=mu[4],x1=mu[6],y0=mu[11],y1=mu[13],col="purple",lwd=2)
segments(x0=mu[3],x1=mu[5],y0=mu[10],y1=mu[12],col="purple",lwd=2)
segments(x0=mu[5],x1=mu[7],y0=mu[12],y1=mu[14],col="purple",lwd=2)
segments(x0=mu[6],x1=mu[7],y0=mu[13],y1=mu[14],col="purple",lwd=2)

#Graficar a Marlon
range(sapply(lista2,function(a)a$x))
range(sapply(lista2,function(a)a$y))
plot(c(.35,.55),c(0,.75),main="Individuo 2",cex.lab=2,cex.axis=1.75,
type="n",cex.main=2,xlab="x",ylab="y")
lapply(lista2,points,col=c(1:4,"deepskyblue","darkmagenta","orange")
,pch=15:19,cex=1.5)
points(mu2[8:14]~mu2[1:7],col="purple",pch=19,cex=1.5)
segments(x0=mu2[1],x1=mu2[2],y0=mu2[8],y1=mu2[9],col="purple",lwd=2)
segments(x0=mu2[2],x1=mu2[4],y0=mu2[9],y1=mu2[11],col="purple",lwd=2
)
segments(x0=mu2[1],x1=mu2[3],y0=mu2[8],y1=mu2[10],col="purple",lwd=2
)
segments(x0=mu2[4],x1=mu2[6],y0=mu2[11],y1=mu2[13],col="purple",lwd=
2)
segments(x0=mu2[3],x1=mu2[5],y0=mu2[10],y1=mu2[12],col="purple",lwd=
2)
segments(x0=mu2[5],x1=mu2[7],y0=mu2[12],y1=mu2[14],col="purple",lwd=
2)
segments(x0=mu2[6],x1=mu2[7],y0=mu2[13],y1=mu2[14],col="purple",lwd=
2)
#####

#Ajuste del modelo de procrustes
#Primero se har□ las funciones requeridas, luego se las usar□on las
fotos
#Matrices requeridas
A=matrix(c(0,-1,1,0),ncol=2) %x% diag(1,n)
B=tcrossprod(rep(1,n))

```

```

C=diag(1,2*n)

#u=c(ulx,...,unx,uly,...,uny)
#mu=c(mulx,...,munx,muly,...,muny)
#vp=c(s2,s2h,s2g,sg,sth,sth2)

#Funcion que calcula los estimadores VLS de varianza de efectos
aleatorios
var.vls=function(datos)
{
  mu=rowMeans(datos)
  A1=matrix(c(1,0,0,0),ncol=2)
  A2=matrix(c(0,0,0,1),ncol=2)
  A12=matrix(c(0,1,1,0),ncol=2)
  A=matrix(c(0,-1,1,0),ncol=2) %x% diag(1,n)
  B=tcrossprod(rep(1,n))
  e=sapply(1:ncol(datos),function(j)(datos[,j]-mu))
  Amu=A%*%mu

  M=matrix(0,ncol=7,nrow=7)
  M[1,1]=(crossprod(mu))^2
  M[1,2]=(crossprod(mu,Amu))^2
  M[1,3]=2*crossprod(Amu,mu)*crossprod(mu)
  M[1,4]=crossprod(mu,(A1%x%B)%*%mu)
  M[1,5]=crossprod(mu,(A2%x%B)%*%mu)
  M[1,6]=crossprod(mu,(A12%x%B)%*%mu)
  M[1,7]=crossprod(mu)
  M[2,2]=(crossprod(Amu))^2
  M[2,3]=2*crossprod(mu,Amu)*crossprod(Amu)
  M[2,4]=crossprod(Amu,(A1%x%B)%*%Amu)
  M[2,5]=crossprod(Amu,(A2%x%B)%*%Amu)
  M[2,6]=crossprod(Amu,(A12%x%B)%*%Amu)
  M[2,7]=crossprod(Amu)
  M[3,3]=2*((crossprod(mu,Amu))^2)+2*(crossprod(Amu)*crossprod(mu))
  M[3,4]=2*crossprod(mu,(A1%x%B)%*%Amu)
  M[3,5]=2*crossprod(mu,(A2%x%B)%*%Amu)
  M[3,6]=2*crossprod(mu,(A12%x%B)%*%Amu)
  M[3,7]=2*crossprod(mu,Amu)
  M[4,4]=n^2
  M[4,5]=0
  M[4,6]=0
  M[4,7]=n
  M[5,5]=n^2
  M[5,6]=0
  M[5,7]=n
  M[6,6]=2*(n^2)
  M[6,7]=0
  M[7,7]=2*n
  M[2,1]=M[1,2]
  M[3,1]=M[1,3]
  M[3,2]=M[2,3]
  M[4,1]=M[1,4]
  M[4,2]=M[2,4]
  M[4,3]=M[3,4]

```



```

M[5,1]=M[1,5]
M[5,2]=M[2,5]
M[5,3]=M[3,5]
M[5,4]=M[4,5]
M[6,1]=M[1,6]
M[6,2]=M[2,6]
M[6,3]=M[3,6]
M[6,4]=M[4,6]
M[6,5]=M[5,6]
M[7,1]=M[1,7]
M[7,2]=M[2,7]
M[7,3]=M[3,7]
M[7,4]=M[4,7]
M[7,5]=M[5,7]
M[7,6]=M[6,7]

b1=2*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j],mu)^2)))
b2=2*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j],Amu)^2)))

b3=4*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j],Amu)*crossprod
(mu,e[,j]))))

b4=2*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j], (A1%x%B)%*%e[,
j]))))

b5=2*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j], (A2%x%B)%*%e[,
j]))))

b6=2*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j], (A12%x%B)%*%e[
,j]))))
b7=2*sum(sapply(1:ncol(e), function(j) (crossprod(e[,j]))))

b=c(b1,b2,b3,b4,b5,b6,b7)
N=ncol(datos)

fn=function(x) {
  N*crossprod(x,M%*%x) - crossprod(b,x)
}

hin = function(x) {
  h=rep(NA,1)
  h[1] <-x[1]
  h[2] <-x[2]
  h[3] <-x[4]
  h[4] <-x[5]
  h[5] <-x[7]
  h[6] <-x[1]*x[2]-(x[3]^2)
  h[7] <-x[4]*x[5]-(x[6]^2)
  h
}

par=c(0.01,0.01,0.0094,0.01,0.01,0.0094,0.01)
ans <- constrOptim.nl(par=par, fn=fn, hin=hin)
v=ans$par

```

```

v2=c(v[7],v[1]/v[7],v[2]/v[7],v[3]/v[7],v[4]/v[7],v[5]/v[7],v[6]/v[7
])
  return(v2)
}

#Funcion que calcula la verosimilitud en unz determinada foto
logverind=function(u,mu,vp)
{
  mumut=tcrossprod(mu)
  tau=matrix(c(vp[5],vp[7],vp[7],vp[6]),ncol=2)
  Amu= A %*% mu
  V=vp[2]*mumut + vp[3]*tcrossprod(Amu) + vp[4]*((mumut%*%A) +
    (A%*%mumut)) + (tau%x%B) + C
  r=chol(V)
  e=backsolve(r,u-mu,transpose=TRUE)
  res=-n*log(vp[1]) - sum(log(diag(r))) - 0.5*crossprod(e)/vp[1]
  unclass(res)
}

#Funcion que calcula la verosimilitud en todas las fotos de un
individuo
logver=function(datos,mu,vp)
{
  l=apply(1:ncol(datos),function(j)logverind(datos[,j],mu,vp))
  sum(l)
}

#Restricciones en los parametros de varianza y covarianza
#que se necesitan en la optimizacion
rest = function(x) {
  h=rep(NA,1)
  h[1] <-x[1]
  h[2] <-x[2]
  h[3] <-x[3]
  h[4] <-x[5]
  h[5] <-x[6]
  h[6] <-x[2]*x[3]-(x[4]^2)
  h[7] <-x[5]*x[6]-(x[7]^2)
  h
}

#####
#####Uso de las funciones con las fotos
###Ajustando el modelo en Rebeca
#Calculo del vector promedio
mu=rowMeans(datos_Rebeca)
#Estimando varianza con VLS
vp=var.vls(datos_Rebeca)
#Logaritmo de la verosimilitud en fotos de Rebeca
logver(datos_Rebeca,mu,vp)
#Funcion a optimizar (negativo de la verosimilitud)
f=function(theta) -logver(datos_Rebeca,mu,theta)
#Optimizacion de la verosimilitud

```

```

resp <- constrOptim.nl(par=vp, fn=f, hin=rest)
round(resp$par, 4)
#Valor de la verosimilitud en el punto optimo
logver(datos_Rebeca,mu, resp$par)
#Comparando valor inicial de la varianza y el optimo
round(vp, 4)
round(resp$par, 4)
#Estadistico de prueba
mumut=tcrossprod(mu)
vp_opt=resp$par
tau=matrix(c(vp_opt[5],vp_opt[7],vp_opt[7],vp_opt[6]),ncol=2)
Amu= A %*% mu
V=(vp_opt[2]*mumut + vp_opt[3]*tcrossprod(Amu) +
vp_opt[4]*((mumut%*%A) +
(A%*%mumut)) + (tau%x%B) + C)*vp_opt[1]
N=ncol(datos_Rebeca)

#####Uso de las funciones con las fotos
###Ajustando el modelo en Marlon
#Calculo del vector promedio
mu2=rowMeans(datos_Marlon)
#Estimando varianza con VLS
vp2=var.vls(datos_Marlon)
#Logaritmo de la verosimilitud en fotos de Marlon
logver(datos_Marlon,mu2,vp2)
#Funcion a optimizar (negativo de la verosimilitud)
f2=function(theta) -logver(datos_Marlon,mu2,theta)
#Optimizacion de la funcion de verosimilitud
resp2 <- constrOptim.nl(par=vp2, fn=f2, hin=rest)
resp2$par
#Valor de la verosimilitud en el punto optimo
logver(datos_Marlon,mu2, resp2$par)
#Comparando valor inicial de la varianza y el optimo
round(vp2, 4)
round(resp2$par, 4)
#Estadistico de prueba
mumut2=tcrossprod(mu2)
vp_opt2=resp2$par
tau2=matrix(c(vp_opt2[5],vp_opt2[7],vp_opt2[7],vp_opt2[6]),ncol=2)
Amu2= A %*% mu2
V2=(vp_opt2[2]*mumut2 + vp_opt2[3]*tcrossprod(Amu2) +
vp_opt2[4]*((mumut2%*%A) +
(A%*%mumut2)) + (tau2%x%B) + C)*vp_opt2[1]
N2=ncol(datos_Marlon)

#Ajustando el modelo en Rebeca (individuo 1)
#Estadistico de prueba para comparar fotos de Marlon con las de
Rebeca
ep_Rebeca=(N/(N+1))*sapply(1:N,function(i) crossprod(mu-
datos_Marlon[,i], solve(V,mu-datos_Marlon[,i])))
#Valor P
valorp_Marlon=dchisq(ep_Rebeca,df=2*n)
round(valorp_Marlon, 4)
#Comparando las fotos de Rebeca consigo misma

```

```

ep_Rebeca2=(N/(N+1))*sapply(1:N,function(i)crossprod(mu-
datos_Rebeca[,i],solve(V,mu-datos_Rebeca[,i])))
valorp_Rebeca2=dchisq(ep_Rebeca2,df=2*n)
round(valorp_Rebeca2,4)

#Ajustando el modelo en Marlon (individuo 2)
#Estadistico de prueba para comparar fotos de Rebeca con las de
Marlon
ep_Marlon=(N/(N+1))*sapply(1:N,function(i)crossprod(mu2-
datos_Rebeca[,i],solve(V2,mu2-datos_Rebeca[,i])))
#Valor P
valorp_Rebeca=dchisq(ep_Marlon,df=2*n)
round(valorp_Rebeca,4)
#Comparando las fotos de Marlon consigo mismo
ep_Marlon2=(N/(N+1))*sapply(1:N,function(i)crossprod(mu2-
datos_Marlon[,i],solve(V2,mu2-datos_Marlon[,i])))
valorp_Marlon2=dchisq(ep_Marlon2,df=2*n)
round(valorp_Marlon2,4)

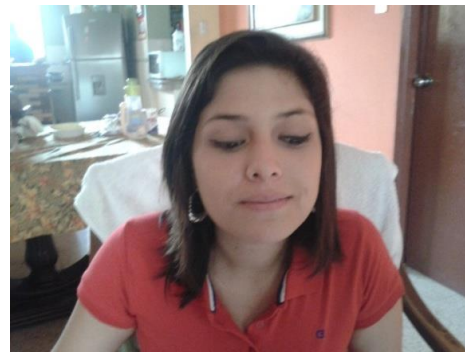
```

IV. Imágenes utilizadas

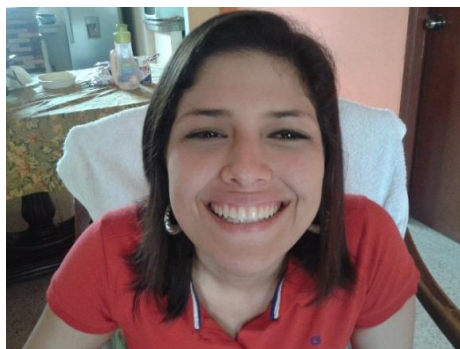
Imágenes sin registrar del individuo uno



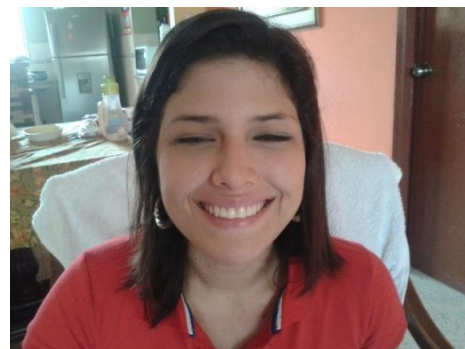
Rebeca1



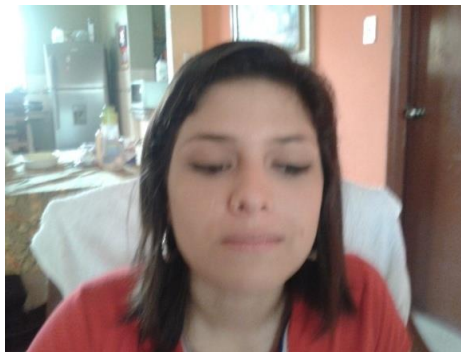
Rebeca2



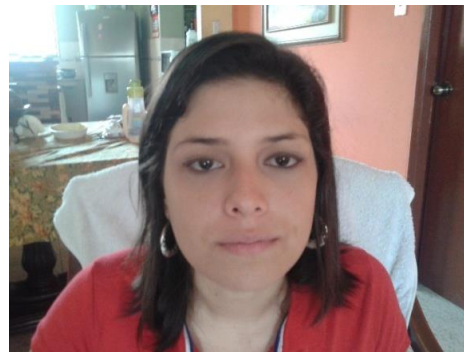
Rebeca3



Rebeca4



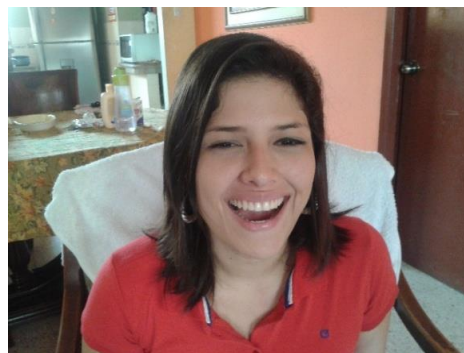
Rebeca5



Rebeca6



Rebeca7

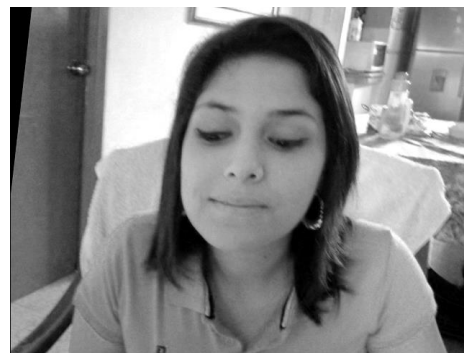


Rebeca8

Imágenes corregistradas del individuo uno



Rebeca1



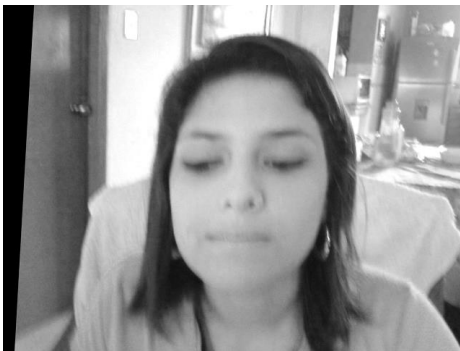
Rebeca2



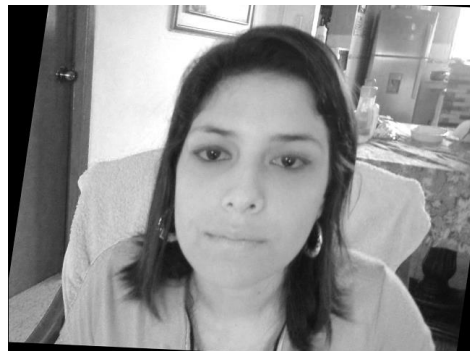
Rebeca3



Rebeca4



Rebeca5



Rebeca6



Rebeca7



Rebeca8

Imágenes sin registrar del individuo dos



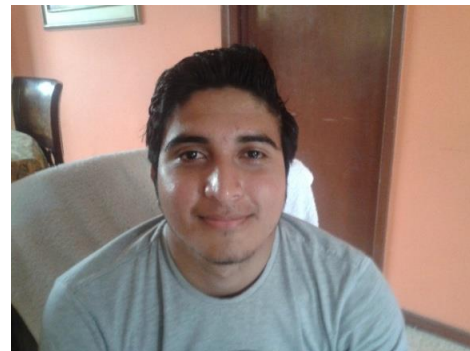
Marlon1



Marlon2



Marlon3



Marlon4



Marlon5



Marlon6



Marlon7



Marlon8

Imágenes corregistradas del individuo dos



Marlon1



Marlon2



Marlon3



Marlon4



Marlon5



Marlon6



Marlon7



Marlon8

Referencias Bibliográficas

1. **Z. Li, Stan y K. Jain, Anil.** *Handbook of face recognition*. New York : Springer, 2004.
2. FBI - Facial Recognition. *sitio web de reconocimiento facial del FBI*. [En línea] [Citado el: 26 de Junio de 2014.] http://www.fbi.gov/about-us/cjis/fingerprints_biometrics/biometric-center-of-excellence/modalities/facial-recognition.
3. History of Face Recognition. *Sitio web de Tanzeem Choudhury* . [En línea] 21 de Enero de 2000. [Citado el: 25 de Junio de 2014.] <http://vismod.media.mit.edu/tech-reports/TR-516/node7.html>.
4. Face Recognition Background. *CNX Org*. [En línea] 17 de Diciembre de 2004. [Citado el: 27 de Junio de 2014.] <http://cnx.org/content/m12535/latest/>.
5. **Tan, Xiaoyang, y otros.** *Face recognition from a single image per person: A survey*. China : Elsevier, 2006.
6. **Modi, Mitul y Macwan, Fedrick.** *Face Detection Approaches: A Survey*. Gujarat, India : s.n., 2014.
7. **Blanco Iturralde, David y Chavez Sanchez, Juan.** *Sistema de reconocimiento facial utilizando el análisis de componentes principales con una red neuronal Backpropagation desarrollada en C# y Matlab*. Quito : s.n., 2012.

8. **Cheng, Li-Fen, y otros.** *A new LDA-based face recognition system which can solve the small sample size problem.* s.l. : Elsevier, 2000.
9. *Face Recognition by Extending Elastic Bunch Graph Matching with Particle Swarm Optimization.* **Senaratne, Rajinda, Halgamuge, Saman y Hsu, Arthur.** Melbourne, Australia : s.n., 2009, Journal of Multimedia, Vol. 4, pág. 204.
10. **Wiskott, Laurenz, y otros.** *Face Recognition by Elastic Bunch Graph Matching.* Los Angeles, EEUU - Bochum, Alemania : CRC press, 1999.
11. **C. Pinheiro, Jose y M. Bates, Douglas.** *Mixed Effects Models with S and S-Plus.* New York : Springer, 2000.
12. **Fox, John.** *Linear Mixed Models.* 2002.
13. **Everitt, Brian y Hothorn, Torsten.** *An Introduction to Applied Multivariate Analysis with R.* New York : Springer, 2011.
14. **C. Rencher, Alvin.** *Methods of Multivariate Analysis.* New York : Wiley, 2002.
15. **Demidenko, Eugene.** *Mixed Models - Theory and Applications.* New Jersey : John Wiley & Sons, Inc., 2004.
16. R: A language and environment for statistical computing. *R Foundation for Statistical Computing.* [En línea] <http://cran.r-project.org/>.
17. **Gregoire, Pau, y otros.** EBImage: Image processing. *R package version 4.4.0.* [En línea] <http://www.bioconductor.org/packages/release/bioc/html/EBImage.html>.

18. **Varadhan, Ravi.** R package version 2011.9-1. *Package alabama*. [En línea] 2012. <http://CRAN.R-project.org/package=alabama>.
19. **Clayden, Jon.** R package version 1.1.2. *Package RNiftyReg*. [En línea] 2013. basado en el código original de Marc Modat y Pankaj Daga. <http://CRAN.R-project.org/package=RNiftyReg>.