

Documentación Programa de Morfometría Matricial en R

1. Descripción General

Este programa procesa, analiza y visualiza contornos morfométricos de objetos geométricos. Sus principales funcionalidades son:

- Importación de contornos internos y externos desde archivos `.txt`.
 - Cálculo de métricas morfométricas: longitud de contorno, ángulo, volumen.
 - Normalización y ajuste de contornos al origen cero.
 - Generación de arrays y listas para análisis multivariante.
 - Análisis exploratorio: estadísticos básicos, correlación, covarianza, PCA.
 - Visualización 2D y 3D de contornos y comparaciones entre individuos.
 - Cálculo de distancias entre coordenadas y áreas bajo curvas.
-

2. Estructura del Programa

2.1 `librerias.R`

Propósito: Gestión de dependencias y carga de librerías.

Funciones principales:

- `install_if_missing(pkg)`: Instala una librería si no está presente.
- Carga librerías esenciales: `bezier`, `plotrix`, `scatterplot3d`, `tcltk`, `stats`, `pracma`, `gWidgets2`, `gWidgets2tcltk`, `digest`.

2.2 morfometria.R

Propósito: Procesamiento de contornos y métricas morfométricas.

Funciones clave:

- `IMP()`: Importa archivos de contornos y calcula longitud, ángulo y volumen.
- `AR()`: Genera arrays interpolados con curvas Bezier.
- `NF()`: Normaliza valores de X, Y, CL, Ángulo y Volumen.
- `OR_ce()`, `OR_ci()`: Ajustan contornos al origen cero.
- `IND()`, `GT()`: Identificación de individuos y tablas de datos.
- Visualización: `COMP()`, `PER_1()`, `PER_2()`, `PER_3()`, `P3D()`, `PO()`.

2.3 analisis.R

Propósito: Análisis exploratorio y multivariante.

Funciones clave:

- `sumario()`: Resumen estadístico (mínimo, cuartiles, mediana, media, máximo, sd).
- `COR()`, `COV()`: Matrices de correlación y covarianza.
- `PCA()`: Análisis de componentes principales (prcomp/princomp).
- `COP0()`: Calcula distancias euclidianas entre coordenadas.
- `AREA.FUN()`: Calcula el área bajo curvas con integración trapezoidal.

3. Flujo de Trabajo (Ejemplo Práctico)

1. Importar librerías y dependencias

```
source("librerias.R")
```

2. Importar contornos

```
L1 <- IMP(1, ruta) # Contornos internos
```

```
L2 <- IMP(2, ruta) # Contornos externos
```

3. Normalizar datos

L3 <- NF(L1)

L4 <- NF(L2)

4. Traslación de contornos al origen cero (opcional)

L5 <- OR_ci(L1, L2)

L6 <- OR_ce(L2)

5. Generación de arrays interpolados

A1 <- AR(1, 20, ruta) # 20 puntos arbitrarios

A2 <- AR(2, 20, ruta)

A3 <- NF(A1)

A4 <- NF(A2)

6. Plantilla de individuos

ind <- IND(A1, destination)

7. Análisis exploratorio

S1 <- sumario(L1)

S2 <- sumario(A1)

CO1 <- COR(L1)

CO2 <- COR(A1)

CV1 <- COV(L1)

CV2 <- COV(A1)

8. Distancias entre coordenadas

DC <- COPO(A3)

9. Análisis de componentes principales

ACP1 <- PCA(L1, 1, TRUE)

ACP2 <- PCA(A1, 1, TRUE)

ACP3 <- PCA(CO1, 1, FALSE)

ACP4 <- PCA(CO2, 1, FALSE)

10. Visualización (ejemplos)

AREA.FUN(A3, 1)

COMP(L4, L3, 1, 2)

PER_1(L2, L1, 1, 2, 3, 4)

P3D(L1, 1)

PO(A3, 1, 1)

4. Notas y Recomendaciones

- Todas las funciones manejan **listas** y **arrays 3D**, permitiendo múltiples individuos y variables.
- Las rutas (**ruta** y **destination**) deben definirse correctamente antes de ejecutar el flujo de trabajo.
- La interpolación con Bezier y normalización facilitan comparaciones entre individuos de distinto tamaño.
- Se recomienda revisar las funciones de visualización antes de procesar grandes conjuntos de datos para optimizar rendimiento.