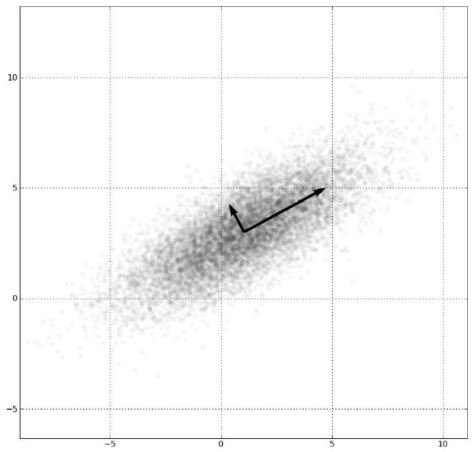
TRABAJO DE CURSO MNC

20/21

Análisis de Componentes Principales

1. Introducción

La técnica denominada PCA (Principal Component Analysis) es empleada en áreas como la Visión por Computador o la Minería de Datos para simplificar la representación de datos masivos y obtener una descripción más compacta de los mismos. Dado un conjunto de puntos (instancias o muestras), se desea obtener las coordenadas de esos puntos en un sistema de representación de nuevas coordenadas (dimensiones o características) que están centrada en el conjunto de datos y rotadas en un alineamiento a las direcciones principales del agrupamiento de los datos.



Estas direcciones se pueden obtener como los autovectores correspondientes a los autovalores más grandes de la matriz de correlación.

2. Selección del conjunto de datos

Cada grupo deberá seleccionar un conjunto de puntos con el que trabajar distinto, elegido de las bases de datos disponibles en el repositorio de Machine Learning de la Universidad de California en Irvine (UCI): https://archive.ics.uci.edu/ml/datasets

Deberá escogerse una base de datos con al menos cinco características numéricas de tipo real y con más de 200 instancias.

Anotarla al crear el grupo para el trabajo de curso, a fin de evitar duplicidades.

3. Solución con Matlab/Octave

Deberán resolverse las siguientes fases:

1. Extraer del fichero de datos las características de tipo real. Se generará una matriz X de m filas (instancias) por n columnas (dimensiones)

2. Centrar los datos restando la media de cada componente, generando una matriz XC
3. Calcular los autovalores y los autovectores de la matriz de covarianza Z = (XC'*XC)/m
4. Representar los datos y los autovalores principales
5. ¿Qué ocurre al multiplicar los datos por la matriz de autovectores?

Se recomienda probar primero el código con un conjunto de datos sintético para verificar que funciona, antes de intentar resolver el problema elegido. Pueden emplear el siguiente código para generar un conjunto de datos rotado y trasladado.

```
m = 500; n = 2;
A = randn(m,n);
% deformación por un factor de 3
A(:,2) = 3*A(:,2);
% matriz de rotación
phi = 45;
cose = cosd(phi); sen = sind(phi);
R = [cose -sen; sen cose];
% rotación y traslacion al punto (10,10)
B = A*R + 10;
```

6. Otras soluciones [OPCIONAL]

Estudiar e implementar soluciones con librerías numéricas y posibles paralelizaciones del trabajo.

7. Evaluación

La evaluación del trabajo, entendiendo que la práctica sea correcta, se guiará por la siguiente escala:

- Solución únicamente con Matlab/Octave: 5
- BLAS/LAPACK: 8
- MPI: 9
- OpenMP: 9
- MPI+OpenMP: 10
- CUDA: 10

8. Entrega y defensa

El código desarrollado se entregará junto con una memoria en PDF con una extensión máxima de 5 páginas para la implementación básica y de 7 páginas si se ha realizado algún apartado opcional (los posibles anexos de código y figuras no cuentan para ese límite). Obligatoriamente deberá mostrarse una gráfica con los datos iniciales y otra con el resultado final. El plazo límite será el 26 de enero.

El trabajo deberá defenderse de forma telepresencial el día 29 de enero, incluyendo una demostración de ejecución.