

# Reducción de dimensión: análisis en componentes principales

Mathieu Kessler

Departamento de Matemática Aplicada y Estadística  
Universidad Politécnica de Cartagena

Cartagena

# El problema de la reducción de dimensión

## Reducción de dimensión

En situaciones donde tenemos muchas variables asociadas a los individuos de un conjunto, buscamos reducir la dimensión del conjunto sin perder demasiada información.

## Reducción de dimensión

En situaciones donde tenemos muchas variables asociadas a los individuos de un conjunto, buscamos reducir la dimensión del conjunto sin perder demasiada información.

Lo hacemos con posiblemente dos objetivos:

- Compresión del conjunto de datos.
- Visualización del conjunto de datos,

# El problema de la reducción de dimensión

Un conjunto con  $k$  variables, y  $n$  individuos: introducimos la matriz

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} \end{pmatrix}.$$

# El problema de la reducción de dimensión

Un conjunto con  $k$  variables, y  $n$  individuos: introducimos la matriz

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} \end{pmatrix}.$$

Los datos forman una nube en un espacio  $k$ -dimensional: cada fila contiene las  $k$  coordenadas del punto asociado a un individuo en el espacio.

# El problema de la reducción de dimensión

*Ejemplo:* Consideremos el conjunto de datos:

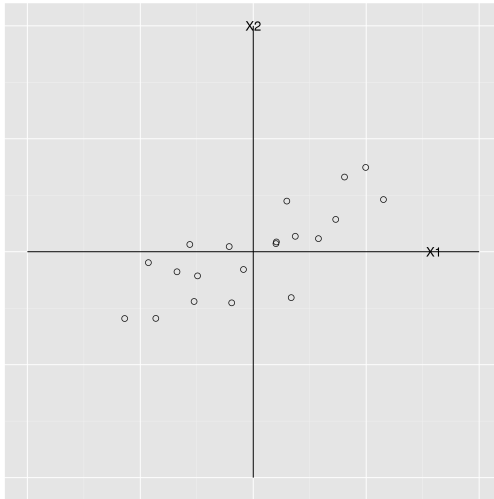
$$X = \begin{pmatrix} 1.360 & 0.705 \\ -2.115 & -0.720 \\ -2.460 & 0.670 \\ \vdots & \vdots \\ 4.000 & 1.775 \\ -0.080 & -0.440 \\ -3.960 & -2.605 \\ -2.270 & -1.860 \end{pmatrix}.$$

# El problema de la reducción de dimensión

*Ejemplo:* Consideremos el conjunto de datos: 20 individuos, 2 variables

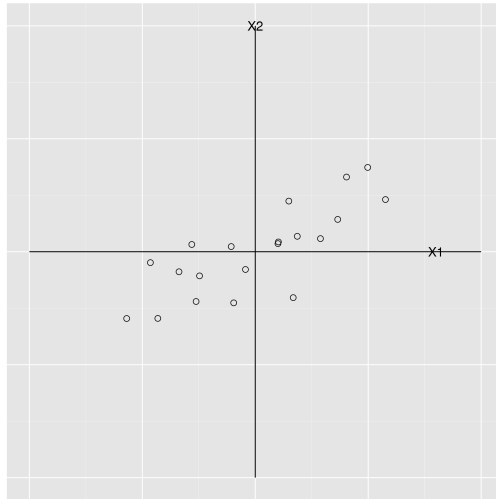
$$X = \begin{pmatrix} 1.360 & 0.705 \\ -2.115 & -0.720 \\ -2.460 & 0.670 \\ \vdots & \vdots \\ 4.000 & 1.775 \\ -0.080 & -0.440 \\ -3.960 & -2.605 \\ -2.270 & -1.860 \end{pmatrix}.$$

# Representación de la nube

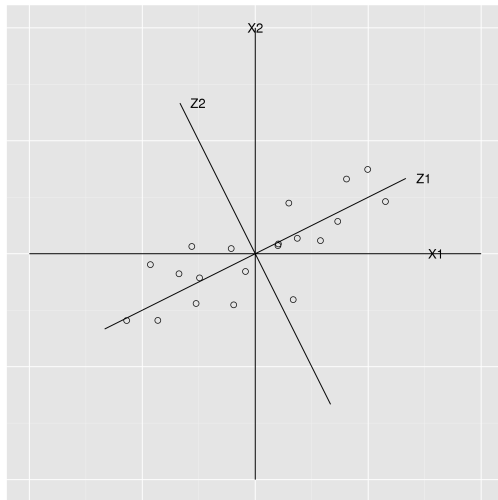




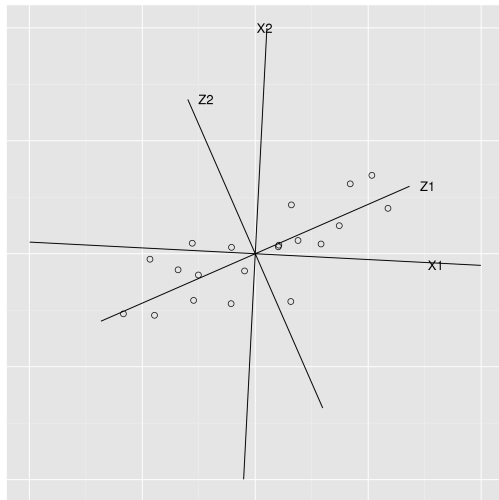
# Cambio de sistema de coordenadas



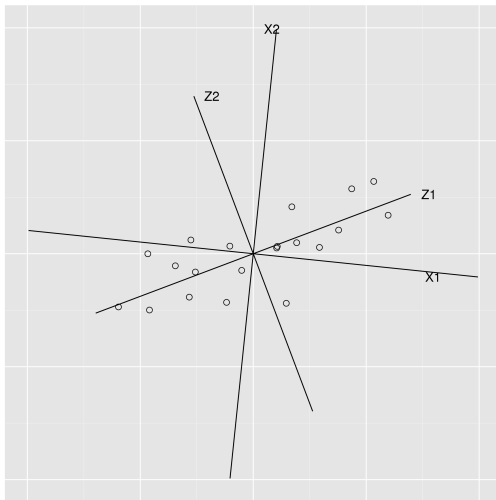
# Cambio de sistema de coordenadas



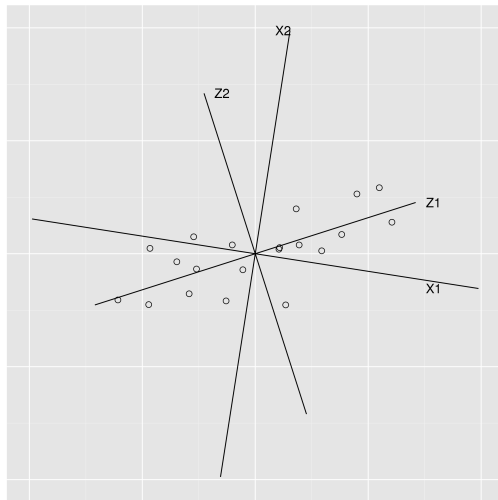
# Cambio de sistema de coordenadas



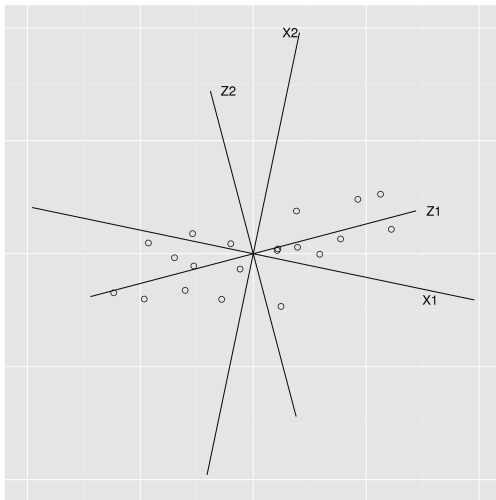
# Cambio de sistema de coordenadas



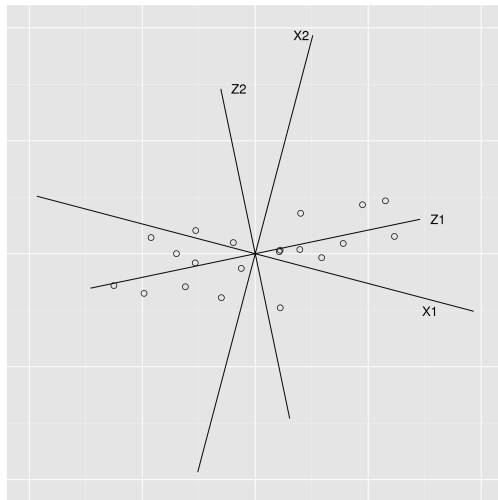
# Cambio de sistema de coordenadas



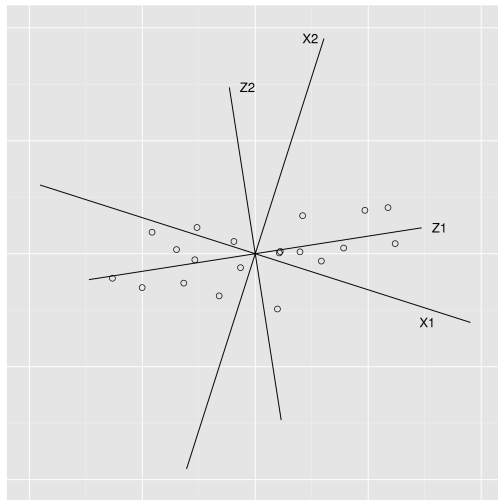
# Cambio de sistema de coordenadas



# Cambio de sistema de coordenadas

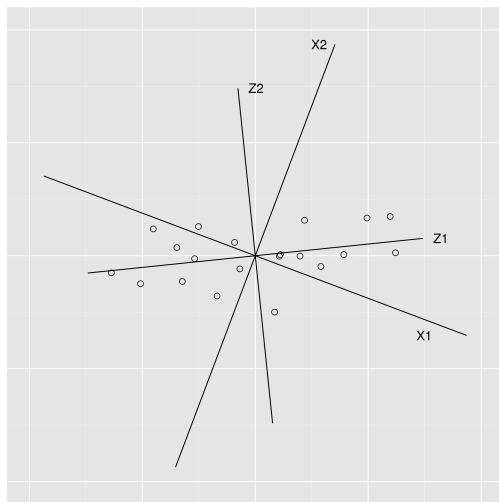


# Cambio de sistema de coordenadas

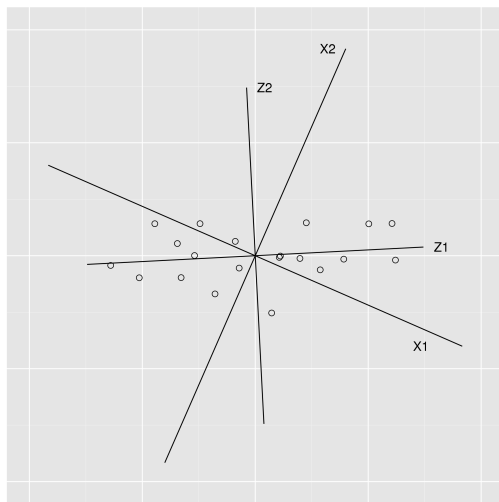




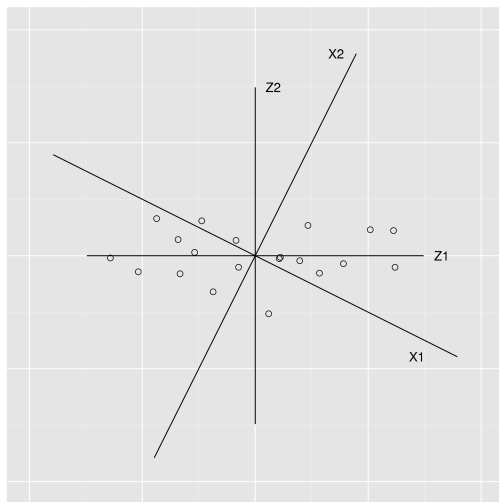
# Cambio de sistema de coordenadas



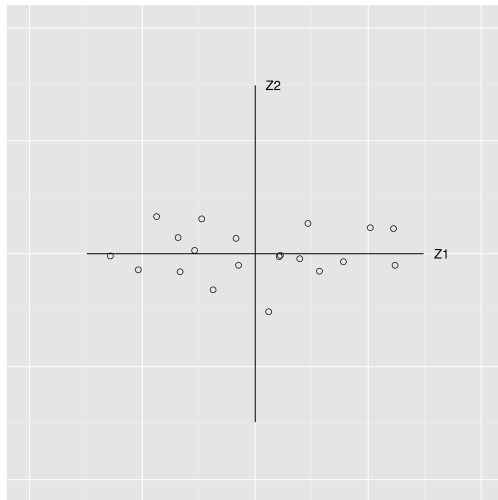
# Cambio de sistema de coordenadas



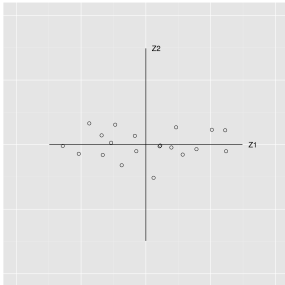
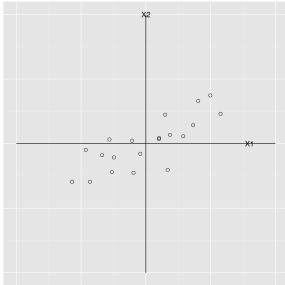
# Cambio de sistema de coordenadas



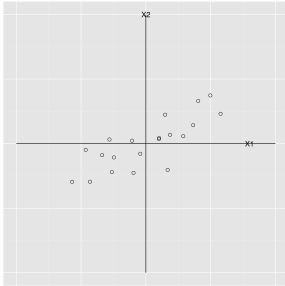
# Cambio de sistema de coordenadas



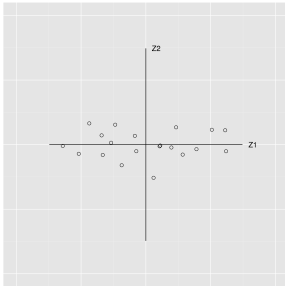
# Qué conseguimos con el cambio



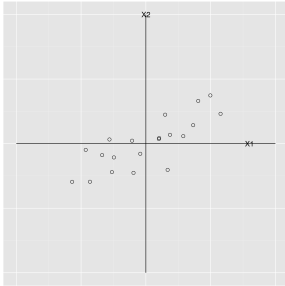
# Qué conseguimos con el cambio



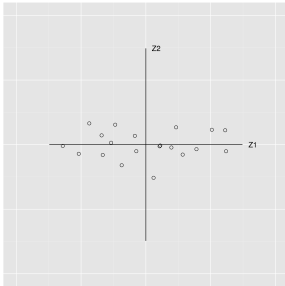
- La descripción es equivalente en los dos sistemas de coordenadas



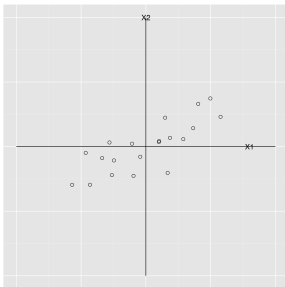
# Qué conseguimos con el cambio



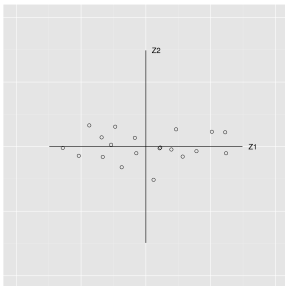
- La descripción es equivalente en los dos sistemas de coordenadas
- Pero la variabilidad de las componentes en los dos sistemas es diferente:



# Qué conseguimos con el cambio

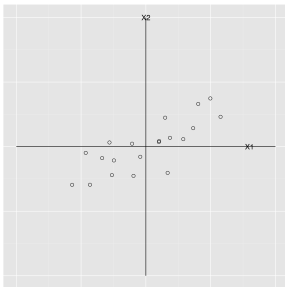


- La descripción es equivalente en los dos sistemas de coordenadas
- Pero la variabilidad de las componentes en los dos sistemas es diferente:
  - $Var(X_1) = 11.2$ ,  $Var(X_2) = 3.84$ .



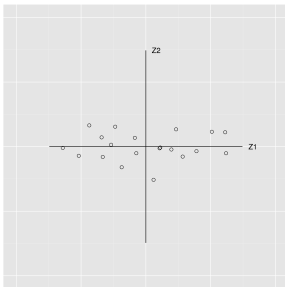


# Qué conseguimos con el cambio

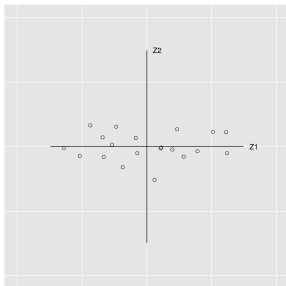
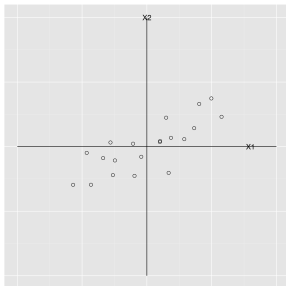


- La descripción es equivalente en los dos sistemas de coordenadas
- Pero la variabilidad de las componentes en los dos sistemas es diferente:

- $Var(X_1) = 11.2, Var(X_2) = 3.84.$
- $Var(Z_1) = 11.13, Var(Z_2) = 0.93.$



# Qué conseguimos con el cambio



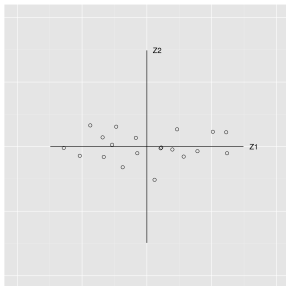
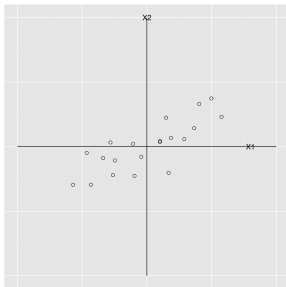
- La descripción es equivalente en los dos sistemas de coordenadas
- Pero la variabilidad de las componentes en los dos sistemas es diferente:

- $Var(X_1) = 11.2$ ,  $Var(X_2) = 3.84$ .

- $Var(Z_1) = 11.13$ ,  $Var(Z_2) = 0.93$ .

En la segunda representación, la variabilidad de  $Z_2$  es pequeña respecto a la variabilidad de  $Z_1$ .

# Qué conseguimos con el cambio



- La descripción es equivalente en los dos sistemas de coordenadas
- Pero la variabilidad de las componentes en los dos sistemas es diferente:

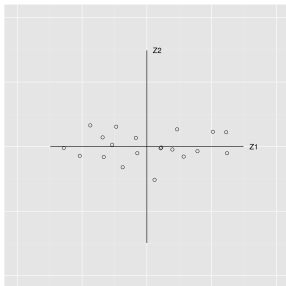
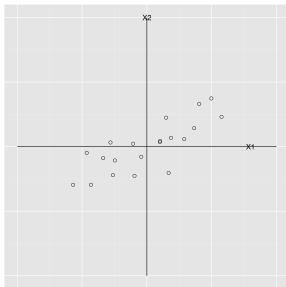
- $Var(X_1) = 11.2$ ,  $Var(X_2) = 3.84$ .

- $Var(Z_1) = 11.13$ ,  $Var(Z_2) = 0.93$ .

En la segunda representación, la variabilidad de  $Z_2$  es pequeña respecto a la variabilidad de  $Z_1$ .

⇒ si tenemos que resumir, nos podemos quedar con  $Z_1$  sólo...

# Qué conseguimos con el cambio



- La descripción es equivalente en los dos sistemas de coordenadas
- Pero la variabilidad de las componentes en los dos sistemas es diferente:

- $Var(X_1) = 11.2$ ,  $Var(X_2) = 3.84$ .

- $Var(Z_1) = 11.13$ ,  $Var(Z_2) = 0.93$ .

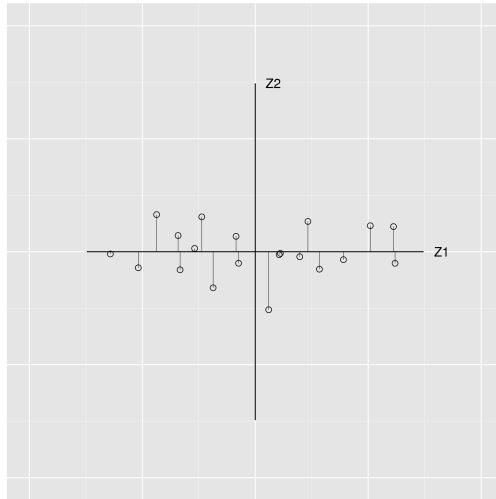
En la segunda representación, la variabilidad de  $Z_2$  es pequeña respecto a la variabilidad de  $Z_1$ .

⇒ si tenemos que resumir, nos podemos quedar con  $Z_1$  sólo...

- $Z_1$  y  $Z_2$  son incorrelados.

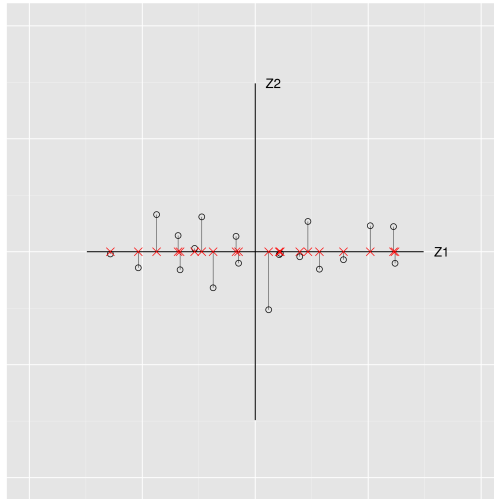
# Reducimos la dimensión

Consideramos la componente  $Z_1$ :



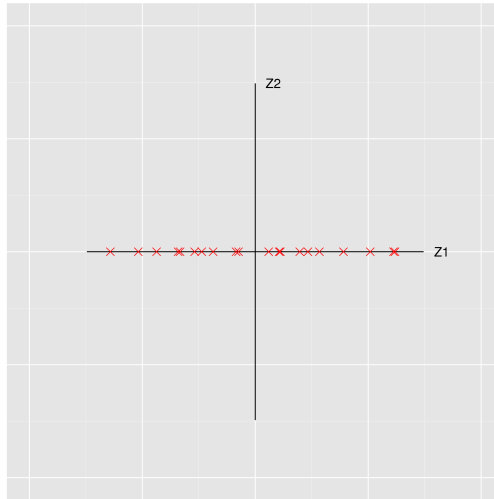
# Reducimos la dimensión

Consideramos la componente  $Z_1$ :



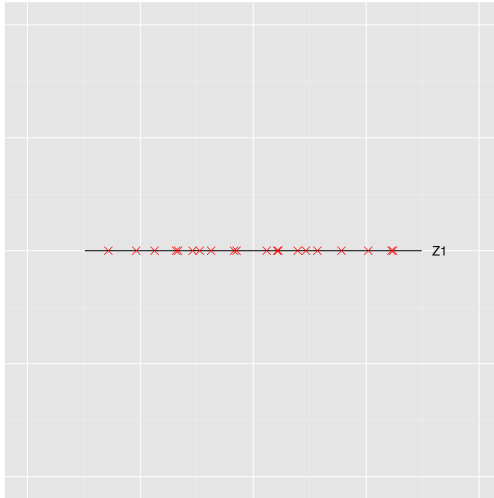
# Reducimos la dimensión

Consideramos la componente Z1:



# Reducimos la dimensión

Consideramos la componente Z1:





# Reducción de dimensión

$$X = \begin{pmatrix} 1.360 & 0.705 \\ -2.115 & -0.720 \\ -2.460 & 0.670 \\ \vdots & \vdots \\ \vdots & \vdots \\ 4.000 & 1.775 \\ -0.080 & -0.440 \\ -3.960 & -2.605 \\ -2.270 & -1.860 \end{pmatrix} \xrightarrow[\text{coordenadas}]{\text{Cambio de}} Z = \begin{pmatrix} 0.947 & 0.114 \\ -2.400 & -0.131 \\ -2.118 & -1.380 \\ \vdots & \vdots \\ \vdots & \vdots \\ 3.492 & 0.315 \\ -0.660 & 0.455 \\ -4.631 & 0.635 \\ -2.976 & 0.714 \end{pmatrix} \xrightarrow[\text{dim.}]{\text{Reducc. de}} \hat{Z} = \begin{pmatrix} 0.947 \\ -2.400 \\ -2.118 \\ \vdots \\ \vdots \\ 3.492 \\ -0.660 \\ -4.631 \\ -2.976 \end{pmatrix}$$

Ejemplo en 3D.

- Sea  $\vec{x}_1, \dots, \vec{x}_k$  una base de  $\mathbb{R}^k$ . Nube de puntos  $k$ -dimensional de  $n$  puntos  $M_1, M_2, \dots, M_n$ .

# Planteamiento general: preliminares

- Sea  $\vec{x}_1, \dots, \vec{x}_k$  una base de  $\mathbb{R}^k$ . Nube de puntos  $k$ -dimensional de  $n$  puntos  $M_1, M_2, \dots, M_n$ .
- $(x_{i1}, x_{i2}, \dots, x_{ik})$  son las coordenadas del punto  $M_i$  en la base  $\vec{x}_1, \dots, \vec{x}_k$ .

# Planteamiento general: preliminares

- Sea  $\vec{x}_1, \dots, \vec{x}_k$  una base de  $\mathbb{R}^k$ . Nube de puntos  $k$ -dimensional de  $n$  puntos  $M_1, M_2, \dots, M_n$ .
- $(x_{i1}, x_{i2}, \dots, x_{ik})$  son las coordenadas del punto  $M_i$  en la base  $\vec{x}_1, \dots, \vec{x}_k$ .
- Consideramos la matriz

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} \end{pmatrix}.$$

- Escogemos otra base  $\vec{u}_1, \dots, \vec{u}_k$  de  $\mathbb{R}^k$ .

# Planteamiento general: preliminares

- Escogemos otra base  $\vec{u}_1, \dots, \vec{u}_k$  de  $\mathbb{R}^k$ .
- Sean  $(z_{i1}, z_{i2}, \dots, z_{ik})$  las coordenadas del punto  $M_i$  en esta nueva base

- Escogemos otra base  $\vec{u}_1, \dots, \vec{u}_k$  de  $\mathbb{R}^k$ .
- Sean  $(z_{i1}, z_{i2}, \dots, z_{ik})$  las coordenadas del punto  $M_i$  en esta nueva base
- Consideramos la matriz

$$Z = \begin{pmatrix} z_{11} & z_{12} & \cdots & z_{1k} \\ z_{21} & z_{22} & \cdots & z_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{nk} \end{pmatrix},$$

de las coordenadas de los puntos de la nube.



# Planteamiento general: preliminares

- Para relacionar  $Z$  con  $X$ , escribimos la matriz de paso de la base base  $\vec{x}_1, \dots, \vec{x}_k$  a la nueva base.

# Planteamiento general: preliminares

- Para relacionar  $Z$  con  $X$ , escribimos la matriz de paso de la base base  $\vec{x}_1, \dots, \vec{x}_k$  a la nueva base.
- La matriz  $U$ , es la matriz  $k \times k$  cuyas columnas contienen las coordenadas de los vectores  $\vec{u}_1, \dots, \vec{u}_k$  en la base inicial, es decir si

$$\begin{aligned}\vec{u}_1 &= u_{11}\vec{x}_1 + u_{21}\vec{x}_2 + \cdots + u_{k1}\vec{x}_k \\ \vec{u}_2 &= u_{12}\vec{x}_1 + u_{22}\vec{x}_2 + \cdots + u_{k2}\vec{x}_k \\ &\vdots \\ \vec{u}_k &= u_{1k}\vec{x}_1 + u_{2k}\vec{x}_2 + \cdots + u_{kk}\vec{x}_k,\end{aligned}$$

■ .... si

$$\begin{aligned}\vec{u}_1 &= u_{11}\vec{x}_1 + u_{21}\vec{x}_2 + \cdots + u_{k1}\vec{x}_k \\ \vec{u}_2 &= u_{12}\vec{x}_1 + u_{22}\vec{x}_2 + \cdots + u_{k2}\vec{x}_k \\ &\vdots \\ \vec{u}_k &= u_{1k}\vec{x}_1 + u_{2k}\vec{x}_2 + \cdots + u_{kk}\vec{x}_k,\end{aligned}$$

la matriz de paso será

$$U = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1k} \\ u_{21} & u_{22} & \cdots & u_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ u_{k1} & u_{k2} & \cdots & u_{kk} \end{pmatrix}.$$

- La relación entre  $Z$ ,  $X$  y  $U$  es

$$X = ZU^T,$$

- La relación entre  $Z$ ,  $X$  y  $U$  es

$$X = ZU^T,$$

- $U$  es invertible , por lo que

$$X(U^T)^{-1} = Z.$$

- La relación entre  $Z$ ,  $X$  y  $U$  es

$$X = ZU^T,$$

- $U$  es invertible, por lo que

$$X(U^T)^{-1} = Z.$$

- Si los vectores  $\vec{u}_1, \dots, \vec{u}_k$  forman una base ortonormal, la matriz  $U$  satisface  $U^T U = Id$ , y por lo tanto  $(U^T)^{-1} = U$ :

$$Z = XU.$$

- Consideremos el conjunto de datos representado por la matriz  $X$ . La matriz de covarianzas de las variables  $X_1, \dots, X_k$  es

$$S_X = \begin{pmatrix} s_{X_1}^2 & s_{X_1 X_2} & \cdots & s_{X_1 X_k} \\ s_{X_1 X_2} & s_{X_2}^2 & \cdots & s_{X_2 X_k} \\ \vdots & \vdots & \vdots & \vdots \\ s_{X_1 X_k} & s_{X_2 X_k} & \cdots & s_{X_k}^2 \end{pmatrix},$$

donde  $s_{X_i}^2$  representa la varianza de la variable  $X_i$  en el conjunto y  $s_{X_i X_j}$  es la covarianza de  $X_i$  y  $X_j$ .

- Consideremos el conjunto de datos representado por la matriz  $X$ . La matriz de covarianzas de las variables  $X_1, \dots, X_k$  es

$$S_X = \begin{pmatrix} s_{X_1}^2 & s_{X_1 X_2} & \cdots & s_{X_1 X_k} \\ s_{X_1 X_2} & s_{X_2}^2 & \cdots & s_{X_2 X_k} \\ \vdots & \vdots & \ddots & \vdots \\ s_{X_1 X_k} & s_{X_2 X_k} & \cdots & s_{X_k}^2 \end{pmatrix},$$

donde  $s_{X_i}^2$  representa la varianza de la variable  $X_i$  en el conjunto y  $s_{X_i X_j}$  es la covarianza de  $X_i$  y  $X_j$ .

- Si  $X_1, \dots, X_k$  tienen cada una media 0:

$$S_X = \frac{1}{n-1} X^T X.$$



- Deducimos, usando la relación entre  $Z$  y  $X$  que

$$S_Z = \frac{1}{n-1} Z^T Z = \frac{1}{n-1} U^{-1} X^T X (U^T)^{-1} = U^{-1} S_X (U^T)^{-1}.$$

- Deducimos, usando la relación entre  $Z$  y  $X$  que

$$S_Z = \frac{1}{n-1} Z^T Z = \frac{1}{n-1} U^{-1} X^T X (U^T)^{-1} = U^{-1} S_X (U^T)^{-1}.$$

- En el caso en que  $U$  es ortogonal,

$$S_Z = U^{-1} S_X U.$$

# Planteamiento general: análisis en componentes principales

- Buscamos el cambio de sistemas de coordenadas que nos permita posiblemente una reducción de dimensión.

# Planteamiento general: análisis en componentes principales

- Buscamos el cambio de sistemas de coordenadas que nos permita posiblemente una reducción de dimensión.
- En ese nuevo sistema, queremos que haya diferencias entre las variabilidad de los componentes: la primera coordenada debe presentar la mayor varianza, la segunda, la segunda mayor varianza, etc...

# Planteamiento general: análisis en componentes principales

- Buscamos el cambio de sistemas de coordenadas que nos permita posiblemente una reducción de dimensión.
- En ese nuevo sistema, queremos que haya diferencias entre las variabilidad de los componentes: la primera coordenada debe presentar la mayor varianza, la segunda, la segunda mayor varianza, etc...
- Teniendo en cuenta

$$S_Z = U^{-1}S_X U,$$

se conseguirá el sistema deseado si  $S_Z$  es una matriz diagonal...

# Planteamiento general: análisis en componentes principales

- Buscamos el cambio de sistemas de coordenadas que nos permita posiblemente una reducción de dimensión.
- En ese nuevo sistema, queremos que haya diferencias entre las variabilidad de los componentes: la primera coordenada debe presentar la mayor varianza, la segunda, la segunda mayor varianza, etc...
- Teniendo en cuenta

$$S_Z = U^{-1}S_XU,$$

se conseguirá el sistema deseado si  $S_Z$  es una matriz diagonal...

- Realizar el análisis en componentes principales, consiste en diagonalizar la matriz  $S_X$ ...

- Al calcular los vectores propios de la matriz  $S_X$ , obtenemos los coeficientes  $u_{ij}$  de la matriz  $U$  de paso y

$$Z_1 = u_{11}X_1 + u_{21}X_2 + \cdots + u_{k1}X_k$$

$$Z_2 = u_{12}X_1 + u_{22}X_2 + \cdots + u_{k2}X_k$$

$$\vdots$$

$$Z_k = u_{1k}X_1 + u_{2k}X_2 + \cdots + u_{kk}X_k.$$

- Al calcular los vectores propios de la matriz  $S_X$ , obtenemos los coeficientes  $u_{ij}$  de la matriz  $U$  de paso y

$$Z_1 = u_{11}X_1 + u_{21}X_2 + \cdots + u_{k1}X_k$$

$$Z_2 = u_{12}X_1 + u_{22}X_2 + \cdots + u_{k2}X_k$$

$$\vdots$$

$$Z_k = u_{1k}X_1 + u_{2k}X_2 + \cdots + u_{kk}X_k.$$

$$1^\circ \text{ vector propio: } \vec{u}_1 = \begin{pmatrix} u_{11} \\ u_{21} \\ \vdots \\ u_{k1} \end{pmatrix}, \dots, k^\circ \text{ vector propio: } \vec{u}_k = \begin{pmatrix} u_{1k} \\ u_{2k} \\ \vdots \\ u_{kk} \end{pmatrix}.$$



- Al calcular los vectores propios de la matriz  $S_X$ , obtenemos los coeficientes  $u_{ij}$  de la matriz  $U$  de paso y

$$\begin{aligned}Z_1 &= u_{11}X_1 + u_{21}X_2 + \cdots + u_{k1}X_k \\Z_2 &= u_{12}X_1 + u_{22}X_2 + \cdots + u_{k2}X_k \\&\vdots \\Z_k &= u_{1k}X_1 + u_{2k}X_2 + \cdots + u_{kk}X_k.\end{aligned}$$

## Componentes principales

Los componentes principales  $Z_1, \dots, Z_k$  se obtienen por lo tanto como combinaciones lineales de las variables originales  $X_1, \dots, X_k$ , cuyos coeficientes se deducen de la expresión de los vectores propios.

# Análisis en componentes principales: ejemplo

- Para el ejemplo 2D que vimos al principio de la clase, tenemos

$$S_X = \begin{pmatrix} 11.238 & 5.194 \\ 5.194 & 3.835 \end{pmatrix},$$

# Análisis en componentes principales: ejemplo

- Para el ejemplo 2D que vimos al principio de la clase, tenemos

$$S_X = \begin{pmatrix} 11.238 & 5.194 \\ 5.194 & 3.835 \end{pmatrix},$$

- Usamos Python para encontrar los valores propios y vectores propios:

# Análisis en componentes principales: ejemplo

- Para el ejemplo 2D que vimos al principio de la clase, tenemos

$$S_X = \begin{pmatrix} 11.238 & 5.194 \\ 5.194 & 3.835 \end{pmatrix},$$

- Usamos Python para encontrar los valores propios y vectores propios:
  - $\lambda_1 \simeq 13.93$  y  $\lambda_2 \simeq 1.16$ .

# Análisis en componentes principales: ejemplo

- Para el ejemplo 2D que vimos al principio de la clase, tenemos

$$S_X = \begin{pmatrix} 11.238 & 5.194 \\ 5.194 & 3.835 \end{pmatrix},$$

- Usamos Python para encontrar los valores propios y vectores propios:

- $\lambda_1 \simeq 13.93$  y  $\lambda_2 \simeq 1.16$ .

- $\vec{u}_1 = \begin{pmatrix} 0.89 \\ 0.46 \end{pmatrix}$  y  $\vec{u}_2 = \begin{pmatrix} -0.46 \\ 0.89 \end{pmatrix}$

# Análisis en componentes principales: ejemplo

- Para el ejemplo 2D que vimos al principio de la clase, tenemos

$$S_X = \begin{pmatrix} 11.238 & 5.194 \\ 5.194 & 3.835 \end{pmatrix},$$

- Usamos Python para encontrar los valores propios y vectores propios:

- $\lambda_1 \simeq 13.93$  y  $\lambda_2 \simeq 1.16$ .

- $\vec{u}_1 = \begin{pmatrix} 0.89 \\ 0.46 \end{pmatrix}$  y  $\vec{u}_2 = \begin{pmatrix} -0.46 \\ 0.89 \end{pmatrix}$

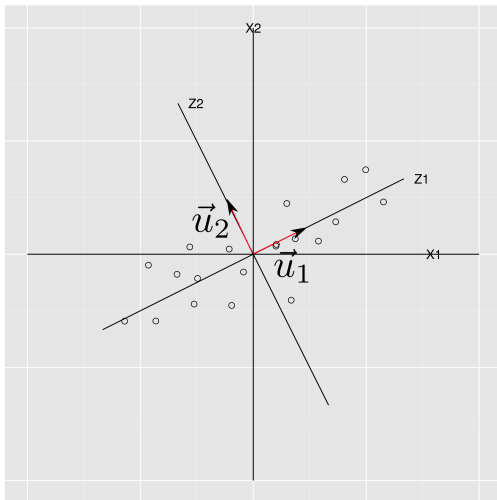
- Los dos componentes principales son

$$Z_1 = 0.89X_1 + 0.46X_2$$

$$Z_2 = -0.46X_1 + 0.89X_2$$

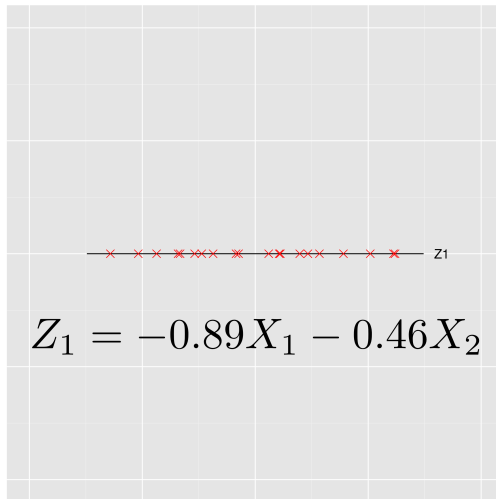
# Cambio de sistema en el ejemplo:

La nueva base:



# Reducimos la dimensión

Cómo calculamos la componente  $Z_1$ :





Por la definición de los componentes principales, la matriz de covarianzas  $S_Z$  es

$$S_Z = \begin{pmatrix} \lambda_1 & 0 & \dots & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ \vdots & \dots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & \dots & \lambda_k \end{pmatrix},$$

Por lo tanto:

- $\lambda_i = \text{var}(Z_i)$

Por la definición de los componentes principales, la matriz de covarianzas  $S_Z$  es

$$S_Z = \begin{pmatrix} \lambda_1 & 0 & \dots & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ \vdots & \dots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & \dots & \lambda_k \end{pmatrix},$$

Por lo tanto:

- $\lambda_i = \text{var}(Z_i)$
- Los componentes principales son incorrelados ( $r_{Z_i Z_j} = 0$ ).

Por teoremas de álgebra lineal:

- 1 Cualquier combinación lineal estandarizada de las variables iniciales, es decir  $a_1X_1 + \cdots + a_kX_k$  con  $a_1^2 + \cdots + a_k^2 = 1$ , presenta una varianza menor o igual que la del primer componente  $Z_1$ , es decir:

$$\text{Var}(a_1X_1 + \cdots + a_kX_k) \leq \lambda_1.$$

Es decir:

Cuando fijamos el primer vector del nuevo sistema como  $\vec{u}_1$ , el vector propio de  $S_X$  asociado a  $\lambda_1$ , maximizamos la varianza de los valores que toma la primera componente en la nube de puntos

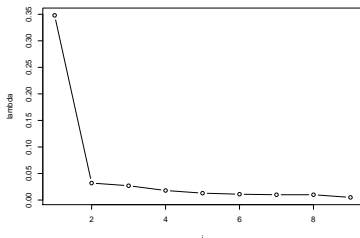
- 2 La variabilidad total se preserva.

$$\begin{aligned} \text{Var}(X_1) + \cdots + \text{Var}(X_k) &= \text{Var}(Z_1) + \cdots + \text{Var}(Z_k) \\ &= \lambda_1 + \cdots + \lambda_k \end{aligned}$$

# ¿Cómo escoger con cuántos componentes nos quedamos?

Nos basamos en la evolución de los valores propios: con dos opciones.

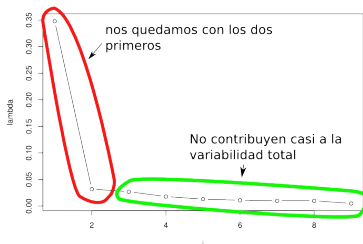
## 1 Diagrama de codo (Scree plot)



# ¿Cómo escoger con cuántos componentes nos quedamos?

Nos basamos en la evolución de los valores propios: con dos opciones.

## 1 Diagrama de codo (Scree plot)



# ¿Cómo escoger con cuántos componentes nos quedamos?

Nos basamos en la evolución de los valores propios: con dos opciones.

## 2 Varianza acumulada explicada. Ejemplo:

Valor propio	% Varianza	% acumulado variabilidad explicada
$\lambda_1$	0.734	0.734
$\lambda_2$	0.068	0.802
$\lambda_3$	0.057	0.859
$\lambda_4$	0.038	0.897
$\lambda_5$	0.027	0.924
$\lambda_6$	0.023	0.947
$\lambda_7$	0.021	0.968
$\lambda_8$	0.021	0.989
$\lambda_9$	0.011	1.000

# ¿Cómo escoger con cuántos componentes nos quedamos?

Nos basamos en la evolución de los valores propios: con dos opciones.

## 2 Varianza acumulada explicada. Ejemplo:

Valor propio	% Varianza	% acumulado variabilidad explicada
$\lambda_1$	0.734	0.734
$\lambda_2$	0.068	0.802
$\lambda_3$	0.057	0.859
$\lambda_4$	0.038	0.897
$\lambda_5$	0.027	0.924
$\lambda_6$	0.023	0.947
$\lambda_7$	0.021	0.968
$\lambda_8$	0.021	0.989
$\lambda_9$	0.011	1.000



# ¿Cómo escoger con cuántos componentes nos quedamos?

Nos basamos en la evolución de los valores propios: con dos opciones.

## 2 Varianza acumulada explicada. Ejemplo:

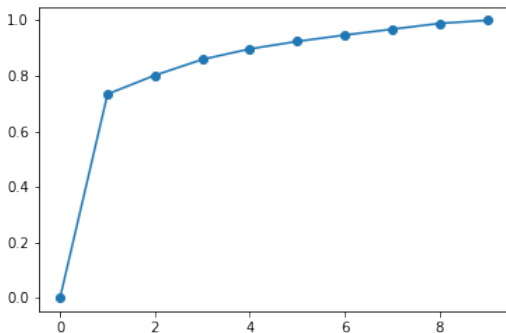
Valor propio	% Varianza	% acumulado variabilidad explicada
$\lambda_1$	0.734	0.734
$\lambda_2$	0.068	0.802
$\lambda_3$	0.057	0.859
$\lambda_4$	0.038	0.897
$\lambda_5$	0.027	0.924
$\lambda_6$	0.023	0.947
$\lambda_7$	0.021	0.968
$\lambda_8$	0.021	0.989
$\lambda_9$	0.011	1.000

Nos quedamos con los lambdas necesarios para alcanzar aprox. 90% varianza.

# ¿Cómo escoger con cuántos componentes nos quedamos?

Nos basamos en la evolución de los valores propios: con dos opciones.

- 2 Varianza acumulada explicada. También se puede representar gráficamente la proporción acumulada de varianza explicada:



Empezamos por cargar los datos del ejemplo:

```
import pandas as pd  
X = pd.read_csv('http://bit.ly/X_ejemplo_acp').values
```

Empezamos por cargar los datos del ejemplo:

```
import pandas as pd  
X = pd.read_csv('http://bit.ly/X_ejemplo_acp').values
```

Usamos la clase PCA del submódulo decomposition.

```
from sklearn.decomposition import PCA
```

Podemos instanciar el transformador y ajustarlo.

```
acp = PCA()  
acp.fit(X) # inplace
```

Podemos ver la expresión de cada componente con el atributo `components_`:

```
print(acp.components_)  
[[-0.88891214 -0.45807773]  
 [ 0.45807773 -0.88891214]]
```

Podemos ver la expresión de cada componente con el atributo `components_`:

```
print(acp.components_)  
[[-0.88891214 -0.45807773]  
 [ 0.45807773 -0.88891214]]
```

Cada componente es una línea de `acp.components_`, por lo tanto, deducimos, redondeando

$$\begin{aligned}Z_1 &= -0.89X_1 - 0.46X_2 \\Z_2 &= 0.46X_1 - 0.89X_2\end{aligned}$$

Además, usando el atributo `explained_variance_` podemos obtener la varianza explicada por cada componente, que coinciden con los auto-valores de  $S_X$ .

Además, usando el atributo `explained_variance_` podemos obtener la varianza explicada por cada componente, que coinciden con los auto-valores de  $S_X$ .

```
print(acp.explained_variance_)  
[13.91446644  1.15890546]
```



Además, usando el atributo `explained_variance_` podemos obtener la varianza explicada por cada componente, que coinciden con los auto-valores de  $S_X$ .

```
print(acp.explained_variance_)  
[13.91446644  1.15890546]
```

Podemos obtener la proporción de varianza explicada con el atributo `explained_variance_ratio_`

```
print(acp.explained_variance_ratio_)  
[0.92311571 0.07688429]
```

Ahora, podemos obtener las coordenadas de cada individuo en el nuevo sistema de coordenadas, es decir los valores de cada individuo en los componentes. Para ello, usamos el método `transform` aplicado a nuestro transformador `acp`

Ahora, podemos obtener las coordenadas de cada individuo en el nuevo sistema de coordenadas, es decir los valores de cada individuo en los componentes. Para ello, usamos el método `transform` aplicado a nuestro transformador `acp`

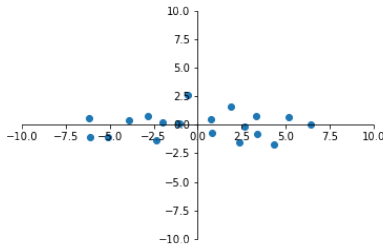
```
Z = acp.transform(X)
```

Ahora, podemos obtener las coordenadas de cada individuo en el nuevo sistema de coordenadas, es decir los valores de cada individuo en los componentes. Para ello, usamos el método `transform` aplicado a nuestro transformador `acp`

```
Z = acp.transform(X)
```

La matriz `Z` contiene las coordenadas de los puntos en el nuevo sistema,

```
fig, ax = plt.subplots()  
ax.scatter(Z[:,0], Z[:,1])
```



Por defecto la clase PCA conserva todos los componentes, pero podríamos especificar directamente el número de componentes que queremos conservar:

```
acp1 = PCA(n_components=1)
acp1.fit(X) # inplace
Z1 = acp1.transform(X)
```

Por defecto la clase PCA conserva todos los componentes, pero podríamos especificar directamente el número de componentes que queremos conservar:

```
acp1 = PCA(n_components=1)
```

```
acp1.fit(X) # inplace
```

```
Z1 = acp1.transform(X)
```

Z1 contiene ahora solo una columna, porque se ha descartado la segunda componente.

Por defecto la clase PCA conserva todos los componentes, pero podríamos especificar directamente el número de componentes que queremos conservar:

```
acp1 = PCA(n_components=1)
```

```
acp1.fit(X) # inplace
```

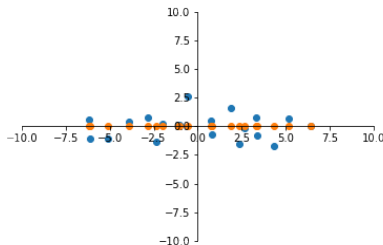
```
Z1 = acp1.transform(X)
```

Z1 contiene ahora solo una columna, porque se ha descartado la segunda componente. Lo representamos junto con Z:

```
fig, ax = plt.subplots()
```

```
ax.scatter(Z[:,0], Z[:,1])
```

```
ax.scatter(Z1, np.zeros(Z.shape[0]))
```



Finalmente, podríamos desde las coordenadas en el nuevo sistema, volver a las coordenadas en el sistema original, usando el método `inverse_transform`

```
X_reconstruido = acp.inverse_transform(Z)
```

```
X_reducido_reconstruido = acp1.inverse_transform(Z1)
```



Finalmente, podríamos desde las coordenadas en el nuevo sistema, volver a las coordenadas en el sistema original, usando el método `inverse_transform`

```
X_reconstruido = acp.inverse_transform(Z)
```

```
X_reducido_reconstruido = acp1.inverse_transform(Z1)
```

Si lo representamos

```
fig, ax = plt.subplots()
```

```
ax.scatter(X_reconstruida[:,0], X_reconstruida[:,1])
```

```
ax.scatter(X_reducida_reconstruida[:,0], X_reducida_reconstruida[:,1])
```

