

Machine learning: basi e sue applicazioni

Christian Salvatore
Scuola Universitaria Superiore IUSS Pavia

christian.salvatore@iusspavia.it

FEATURE EXTRACTION

Feature Extraction | Aim

Map from the initial space to a space smaller than \mathbb{R}^k , with $k < d$

Operate in spaces of lower dimensionality:

- makes it easier to train machine-learning algorithms (requires less data for training)
- discarding redundant (correlated information) and noisy data also improves performance and makes it more robust

The goal is to discard information that is not relevant or less relevant to the problem of interest

Reducing dimensionality does not mean discarding some dimensions and saving others, but combining sizes appropriately.

Feature Extraction | Main Techniques

Principal Component Analysis (PCA)

Unsupervised transformation that performs a linear mapping of dimensions with the aim of preserving the pattern information as much as possible (also known as Karhunen-Loeve transform)

Feature Extraction | Principal Components Analysis

Dato un training set $\mathbf{x}_i \in \mathbb{R}^d$, $i = 1 \dots n$, siano

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1 \dots n} \mathbf{x}_i$$

il vettore medio $\in \mathbb{R}^d$

$$\Sigma = \frac{1}{n-1} \sum_{i=1 \dots n} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^t$$

la matrice di covarianza $\in \mathbb{R}^{d \times d}$

allora per un dato k ($k < d$, $k < n$, $k > 0$), lo spazio k dimensionale $(S_{\mathbf{x}}, \Phi_k)$

è univocamente definito dal vettore medio e dalla matrice di proiezione $\Phi_k \in \mathbb{R}^{d \times k}$

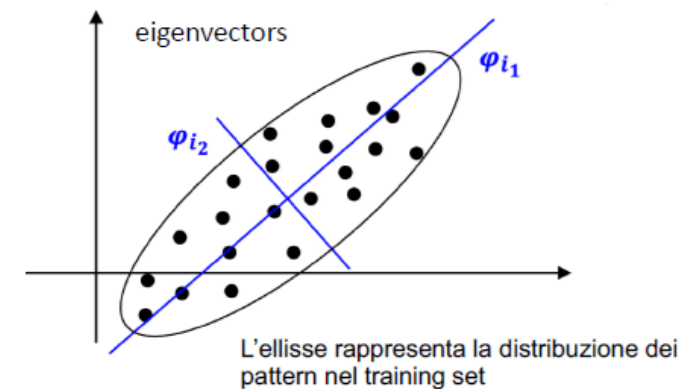
le cui colonne sono costituite dagli autovettori di Σ corrispondenti ai k più grandi autovalori

$$\Phi_k = [\boldsymbol{\varphi}_{i_1}, \boldsymbol{\varphi}_{i_2} \dots \boldsymbol{\varphi}_{i_k}] \text{ con } \lambda_{i_1} \geq \lambda_{i_2} \geq \dots \lambda_{i_k} \geq \dots \lambda_{i_d}$$

$\boldsymbol{\varphi}_{ir}$ autovettore di Σ corrispondente all'autovalore λ_{ir} $r=1 \dots d$

I primi k autovettori sono detti *componenti principali (PC)*

$\boldsymbol{\varphi}_{i_1}$ indica la direzione di maggior varianza nel training set



Feature Extraction | Principal Components Analysis

Variance of one variable X :

$$\text{Var}(X) = \frac{1}{n} \sum_j (\bar{x} - x_j)^2 = \sigma_X^2$$

Covariance of two variables X and Y :

$$\text{Cov}(X, Y) = \frac{1}{n} \sum_j (\bar{x} - x_j)(\bar{y} - y_j) = \sigma_{XY}$$

Covariance matrix of n variables $X_1 \dots X_n$:

$$C = \begin{pmatrix} \sigma_{11}^2 & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22}^2 & \dots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_{nn}^2 \end{pmatrix}$$

PCA diagonalizes the covariance matrix C :

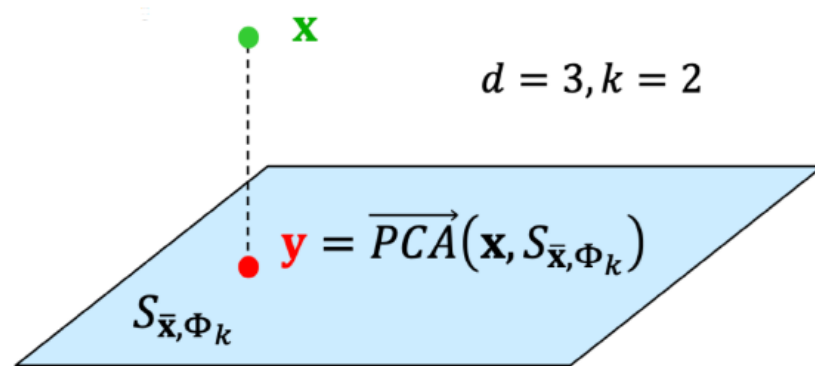
$$C = UDU^T \\ = U \begin{pmatrix} \lambda_1^2 & 0 & \dots & 0 \\ 0 & \lambda_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n^2 \end{pmatrix} U^T$$

U : rotation matrix

D : diagonal matrix

λ_j^2 : eigenvalues (= variance explained by each component)

Feature Extraction | Principal Components Analysis



$$\mathbb{R}_d \rightarrow \mathbb{R}_k$$

Proiezione Una volta determinato lo spazio PCA, la proiezione di un pattern \mathbf{x} su tale spazio è semplicemente la proiezione geometrica del vettore \mathbf{x} sull'iperpiano che definisce lo spazio. In realtà la vera proiezione geometrica è un vettore che ha la stessa dimensionalità del vettore originale mentre in questo contesto indichiamo con proiezione il vettore (ridotto) nello spazio PCA. Matematicamente questa operazione è eseguita come prodotto della matrice di proiezione trasposta per il pattern \mathbf{x} al quale è preventivamente sottratta la media.

$$\overrightarrow{PCA}(\mathbf{x}, S_{\bar{\mathbf{x}}, \Phi_k}) = \Phi_k^t (\mathbf{x} - \bar{\mathbf{x}})$$

$$\mathbb{R}_k \rightarrow \mathbb{R}_d$$

Retroproiezione Dato un vettore \mathbf{y} nello spazio PCA, la sua retro-proiezione verso lo spazio originale si ottiene moltiplicando il vettore per la matrice di proiezione e sommando il vettore medio. Questa trasformazione non sposta spazialmente il vettore, che giace ancora sullo spazio PCA, ma opera un cambiamento di coordinate che ne permette la codifica in termini delle d componenti dello spazio originale.

$$\overleftarrow{PCA}(\mathbf{y}, S_{\bar{\mathbf{x}}, \Phi_k}) = \Phi_k \mathbf{y} + \bar{\mathbf{x}}$$

Feature Extraction | Principal Components Analysis

Se l'obiettivo è quello di scartare informazione inutile e dati correlati mantenendo gran parte del contenuto informativo si può scegliere k nel modo seguente:

Fissata una percentuale t del contenuto informativo che si vuole preservare (es. $t = 95\%$) si sceglie il minimo valore di k per cui la somma dei più grandi k autovalori e' maggiore o uguale a t rispetto alla somma di tutti gli autovalori

Considerando gli autovalori ordinati in ordine decrescente:

$$k = \arg \min_z \left\{ \frac{\sum_{i=1 \dots z} \lambda_i}{\sum_{i=1 \dots d} \lambda_i} \geq t \right\}$$

Poiche' che gli autovalori denotano la varianza lungo i diversi assi, il rapporto nella formula indica la varianza conservata rispetto alla varianza totale

La scelta di k è obbligata ad esempio per la visualizzazione 2D o 3D dei dati ($k=2, 3$)

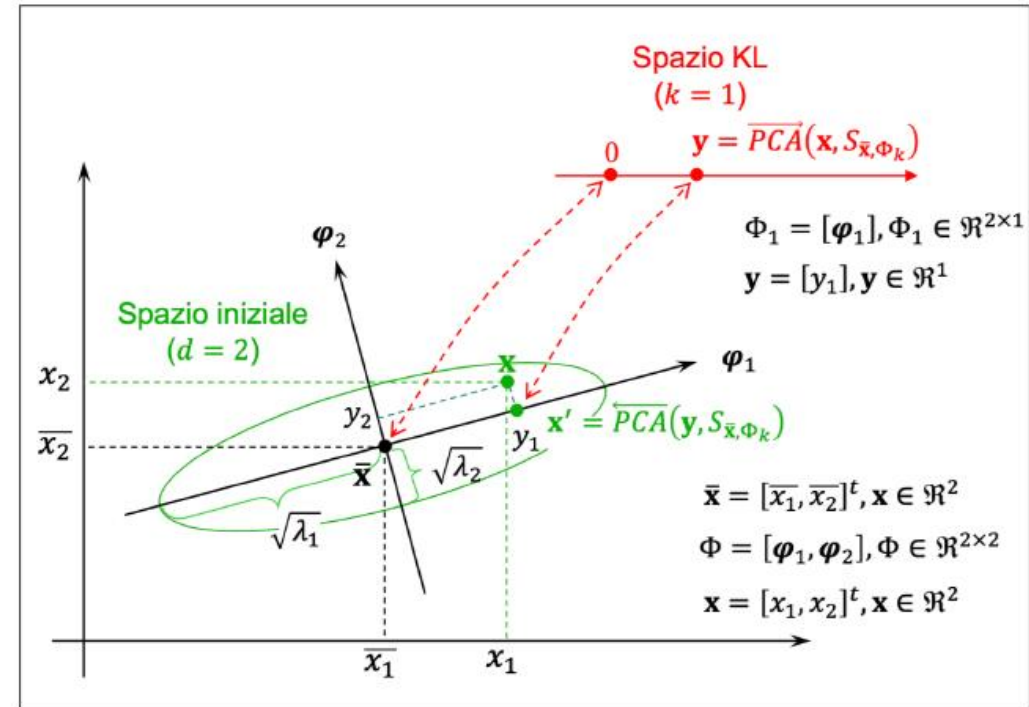
Feature Extraction | Principal Components Analysis

L'ellisse rappresenta la distribuzione dei pattern nel training set

φ_1 e φ_2 sono gli autovettori della matrice di covarianza

Gli autovalori λ_1 e λ_2 sono le varianze della distribuzione lungo gli assi φ_1 e φ_2 .

y_1 e y_2 sono le proiezioni di \mathbf{x} sugli assi φ_1 e φ_2 .

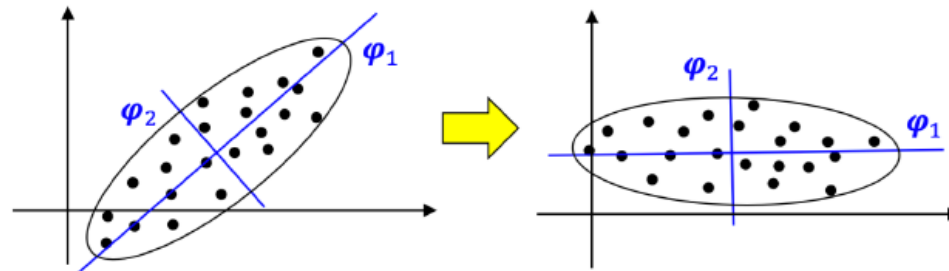


se λ_2 è piccolo, \mathbf{x} può essere approssimato con \mathbf{x}' (retroproiezione di \mathbf{y}) senza perdite significative di informazione

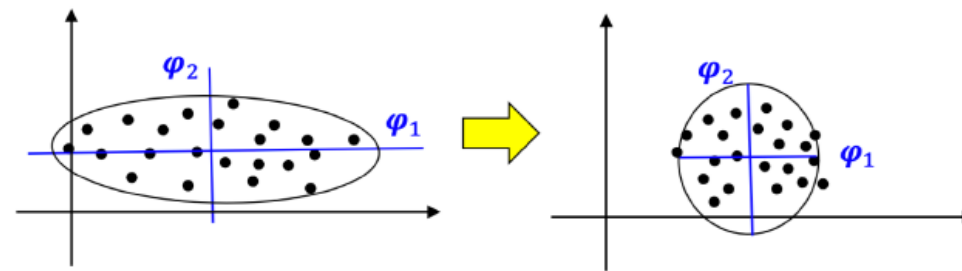
Feature Extraction | Principal Components Analysis

È una tecnica di pre-normalizzazione dei dati, che:

- Rimuove le correlazioni tra le dimensioni, ruotando la nuvola di punti per allineare gli assi di variazione principale dei dati (autovettori) agli assi cartesiani.



- Sfericizza l'ellissoide, uniformando le varianze (denotate dagli autovalori) a 1 lungo tutti gli assi



- Dopo aver proiettato i pattern sullo spazio PCA (definito dai primi k autovettori) è sufficiente dividere ogni dimensione per la radice quadrata dell'autovalore corrispondente (deviazione standard).

La matrice di covarianza dei dati normalizzati è l'identità

Feature Extraction | Principal Components Analysis

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n}$$

$$C = \begin{pmatrix} cov(x, x) & cov(x, y) & cov(x, z) \\ cov(y, x) & cov(y, y) & cov(y, z) \\ cov(z, x) & cov(z, y) & cov(z, z) \end{pmatrix}$$

$$s = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}}$$

Eigenvectors

Eigenvalues

$$s^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}$$

$$var(X) = \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{(n-1)}$$

$$cov(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)}$$

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

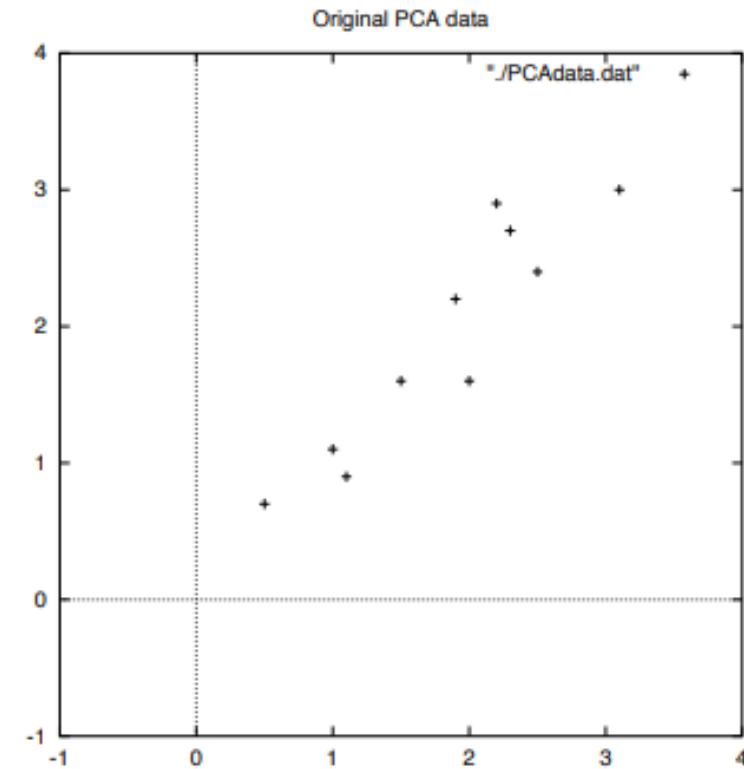
Step 5: Choosing components and forming a feature vector

Step 5: Deriving the new data set

Feature Extraction | Principal Components Analysis

Step 1: Get some data

	x	y
	2.5	2.4
	0.5	0.7
	2.2	2.9
	1.9	2.2
Data =	3.1	3.0
	2.3	2.7
	2	1.6
	1	1.1
	1.5	1.6
	1.1	0.9



Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

	x	y
	2.5	2.4
	0.5	0.7
	2.2	2.9
	1.9	2.2
Data =	3.1	3.0
	2.3	2.7
	2	1.6
	1	1.1
	1.5	1.6
	1.1	0.9

	x	y
	.69	.49
	-1.31	-1.21
	.39	.99
	.09	.29
DataAdjust =	1.29	1.09
	.49	.79
	.19	-.31
	-.81	-.81
	-.31	-.31
	-.71	-1.01

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

	<i>x</i>	<i>y</i>
	.69	.49
	-1.31	-1.21
	.39	.99
	.09	.29
DataAdjust =	1.29	1.09
	.49	.79
	.19	-.31
	-.81	-.81
	-.31	-.31
	-.71	-1.01

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

	<i>x</i>	<i>y</i>
	.69	.49
	-1.31	-1.21
	.39	.99
	.09	.29
DataAdjust =	1.29	1.09
	.49	.79
	.19	-.31
	-.81	-.81
	-.31	-.31
	-.71	-1.01

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

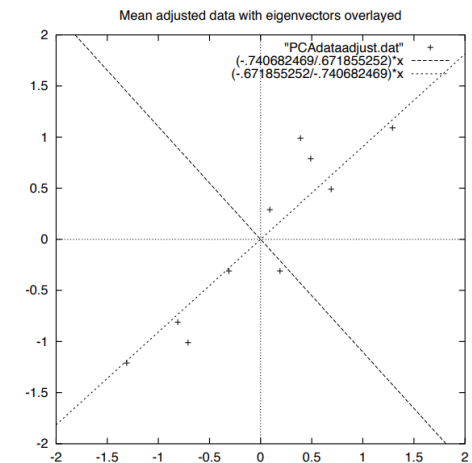


Figure 3.2: A plot of the normalised data (mean subtracted) with the eigenvectors of the covariance matrix overlayed on top.

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

	<i>x</i>	<i>y</i>
	.69	.49
	-1.31	-1.21
	.39	.99
	.09	.29
DataAdjust =	1.29	1.09
	.49	.79
	.19	-.31
	-.81	-.81
	-.31	-.31
	-.71	-1.01

$$\text{eigenvalues} = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$\text{eigenvectors} = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

$$\text{cov} = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

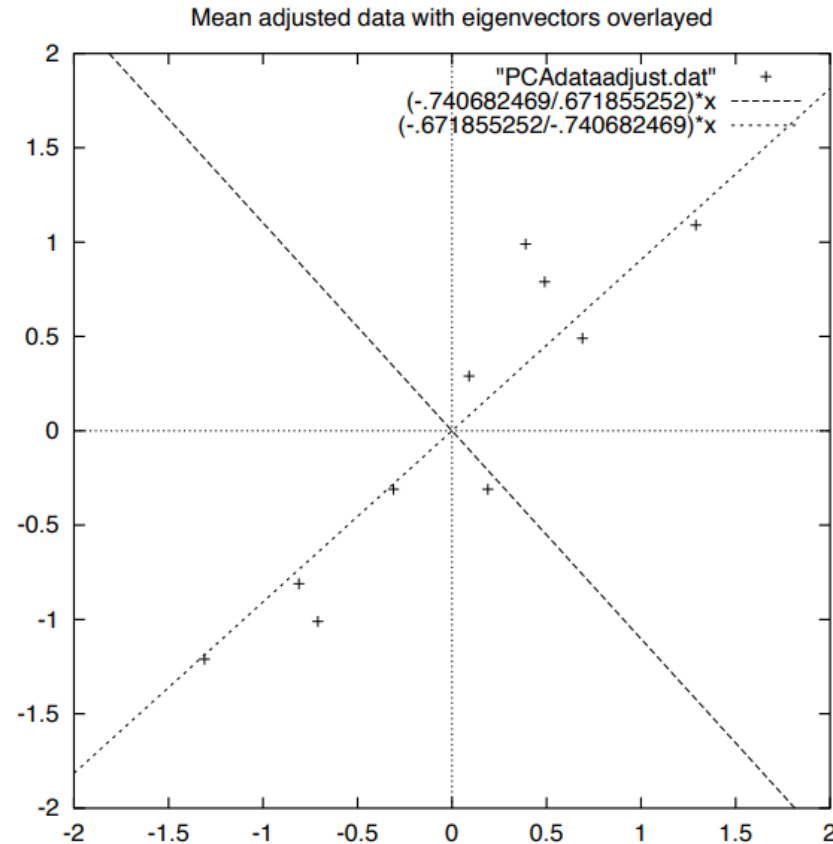


Figure 3.2: A plot of the normalised data (mean subtracted) with the eigenvectors of the covariance matrix overlayed on top.

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step 5: Choosing components and forming a feature vector

	<i>x</i>	<i>y</i>
	.69	.49
	-1.31	-1.21
	.39	.99
	.09	.29
DataAdjust =	1.29	1.09
	.49	.79
	.19	-.31
	-.81	-.81
	-.31	-.31
	-.71	-1.01

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

$$FeatureVector = (eig_1 \ eig_2 \ eig_3 \ \ eig_n)$$

$$\begin{pmatrix} -.677873399 & -.735178656 \\ -.735178656 & .677873399 \end{pmatrix}$$

$$\begin{pmatrix} -.677873399 \\ -.735178656 \end{pmatrix}$$

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step 5: Choosing components and forming a feature vector

Step 5: Deriving the new data set

$$FinalData = RowFeatureVector \times RowDataAdjust,$$

where *RowFeatureVector* is the matrix with the eigenvectors in the columns *transposed* so that the eigenvectors are now in the rows, with the most significant eigenvector at the top, and *RowDataAdjust* is the mean-adjusted data *transposed*, ie. the data items are in each column, with each row holding a separate dimension.

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step 5: Choosing components and forming a feature vector

Step 5: Deriving the new data set

$$FinalData = RowFeatureVector \times RowDataAdjust,$$

where *RowFeatureVector* is the matrix with the eigenvectors in the columns *transposed* so that the eigenvectors are now in the rows, with the most significant eigenvector at the top, and *RowDataAdjust* is the mean-adjusted data *transposed*, ie. the data items are in each column, with each row holding a separate dimension.

Transformed Data=

<i>x</i>	<i>y</i>
-.827970186	-.175115307
1.77758033	.142857227
-.992197494	.384374989
-.274210416	.130417207
-1.67580142	-.209498461
-.912949103	.175282444
.0991094375	-.349824698
1.14457216	.0464172582
.438046137	.0177646297
1.22382056	-.162675287

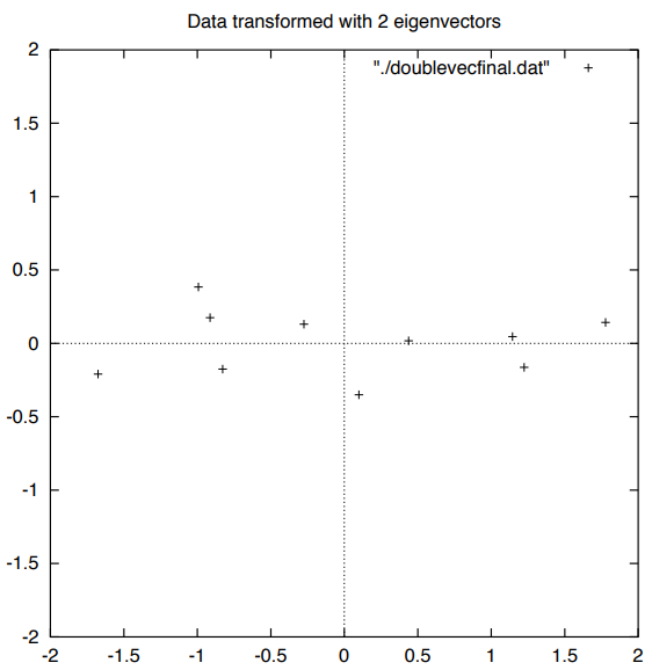


Figure 3.3: The table of data by applying the PCA analysis using both eigenvectors, and a plot of the new data points.

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step 5: Choosing components and forming a feature vector

Step 5: Deriving the new data set

Transformed Data (Single eigenvector)

x
-.827970186
1.77758033
-.992197494
-.274210416
-1.67580142
-.912949103
.0991094375
1.14457216
.438046137
1.22382056

$$FinalData = RowFeatureVector \times RowDataAdjust,$$

where *RowFeatureVector* is the matrix with the eigenvectors in the columns *transposed* so that the eigenvectors are now in the rows, with the most significant eigenvector at the top, and *RowDataAdjust* is the mean-adjusted data *transposed*, ie. the data items are in each column, with each row holding a separate dimension.

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step 5: Choosing components and forming a feature vector

Step 5: Deriving the new data set

Getting the old data back

$$FinalData = RowFeatureVector \times RowDataAdjust$$

$$RowDataAdjust = RowFeatureVector^{-1} \times FinalData$$

$$RowDataAdjust = RowFeatureVector^T \times FinalData$$

only true if the elements of the matrix are all unit eigenvectors

$$RowDataAdjust = RowFeatureVector^T \times FinalData$$

$$RowOriginalData = (RowFeatureVector^T \times FinalData) + OriginalMean$$

Feature Extraction | Principal Components Analysis

Step 1: Get some data

Step 2: Subtract the mean

Step 3: Calculate the covariance matrix

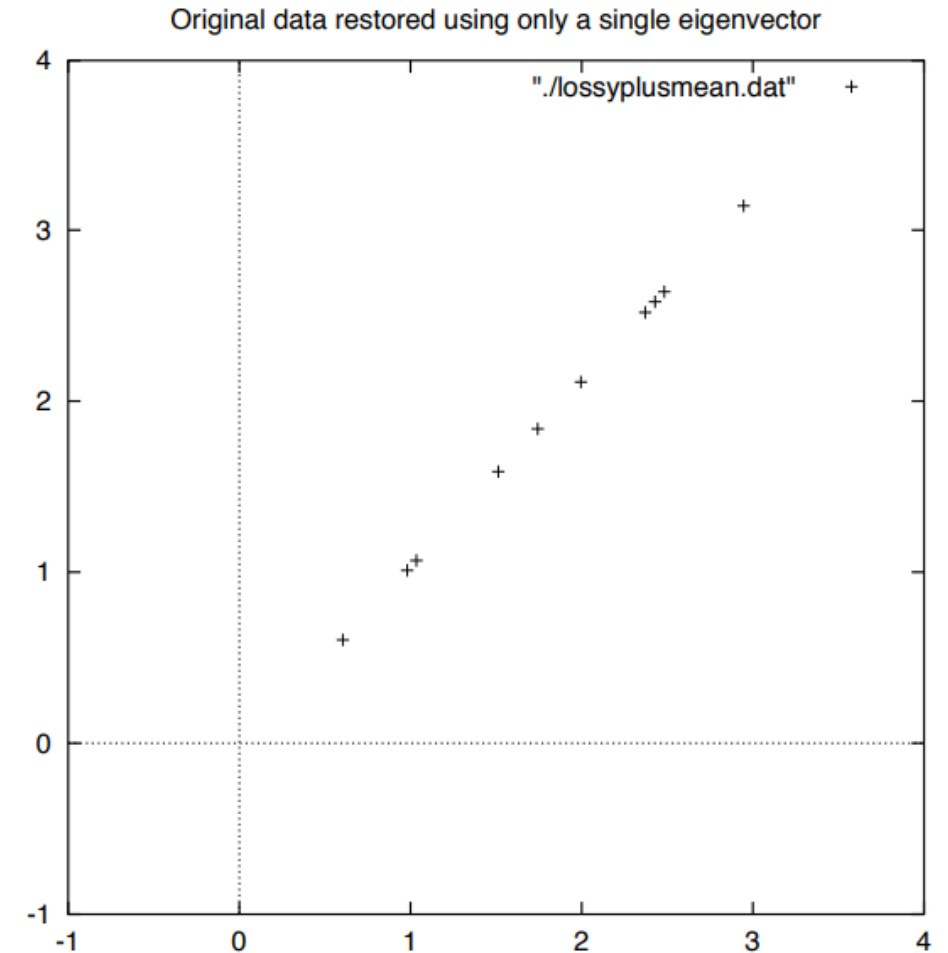
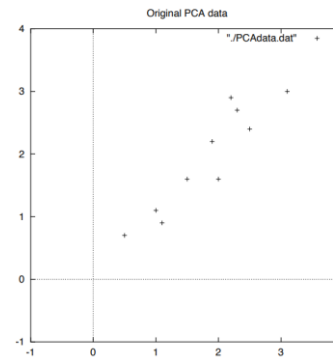
Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step 5: Choosing components and forming a feature vector

Step 5: Deriving the new data set

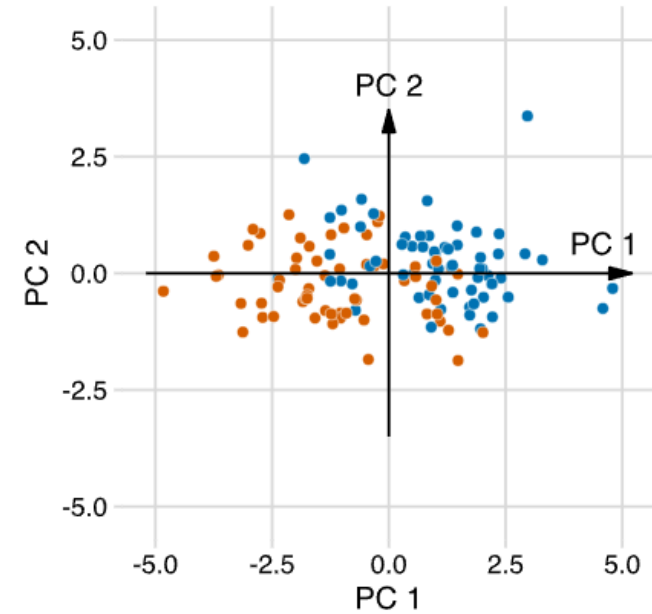
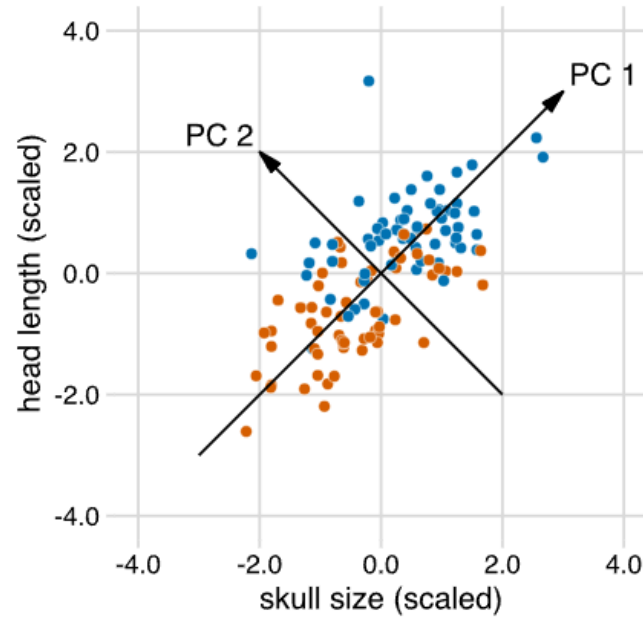
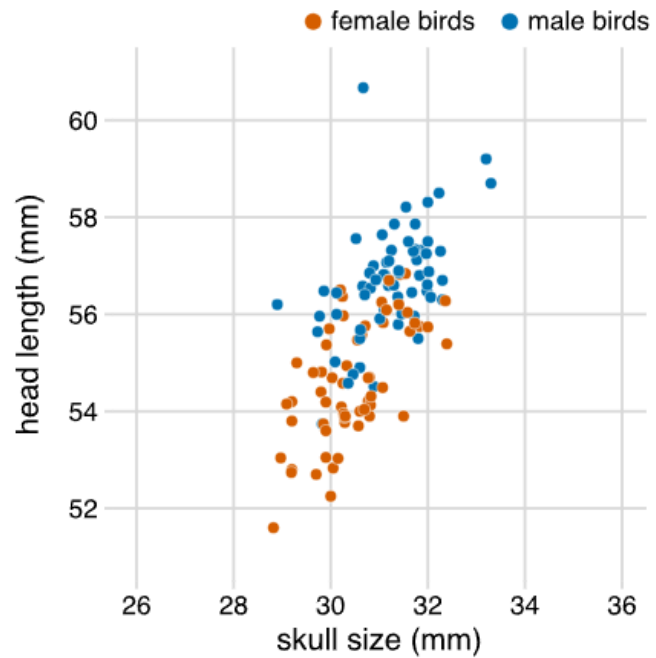
Getting the old data back

$$\text{RowOriginalData} = (\text{RowFeatureVector}^T \times \text{FinalData}) + \text{OriginalMean}$$

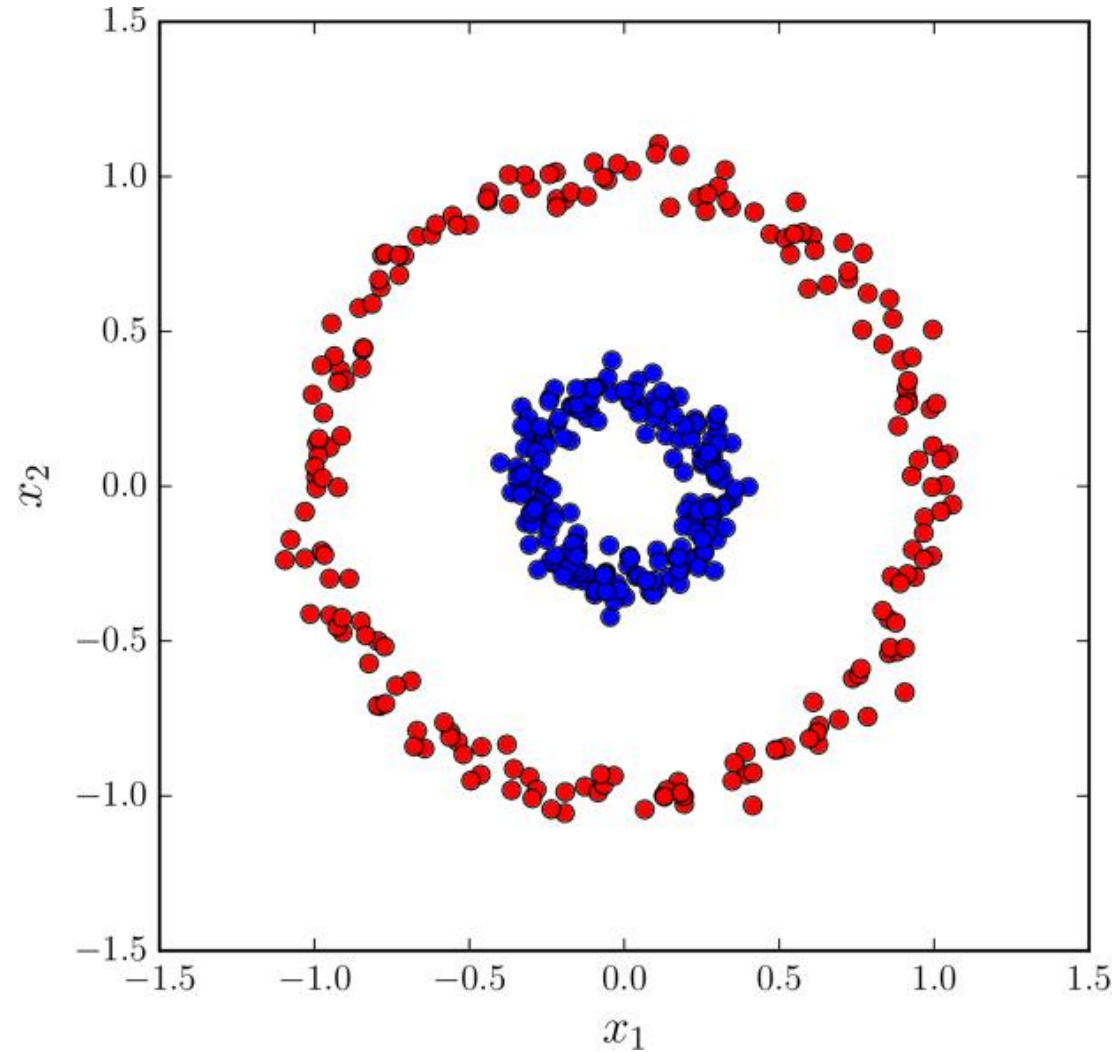


Feature Extraction | Principal Components Analysis

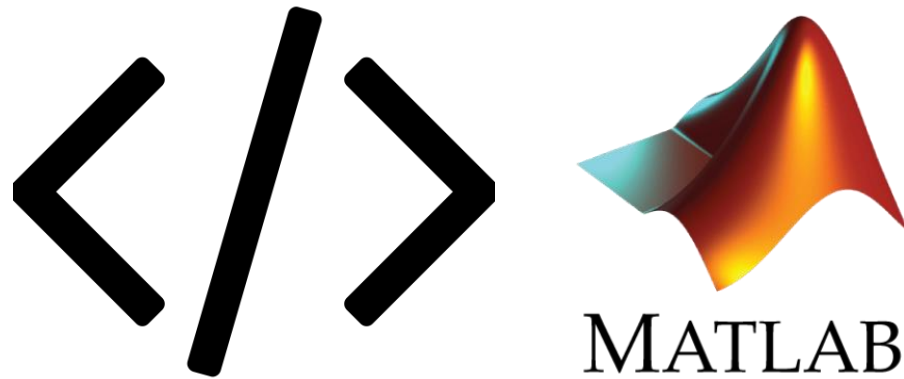
WORK



Feature Extraction | Principal Components Analysis



NOT WORKING



Feature Extraction

- Accuracy improvements.
- Overfitting risk reduction.
- Speed up in training.
- Improved Data Visualization.
- Increase in explainability of our model.

Many other
techniques...

For a visual explanation of PCA, see
<https://setosa.io/ev/principal-component-analysis/>