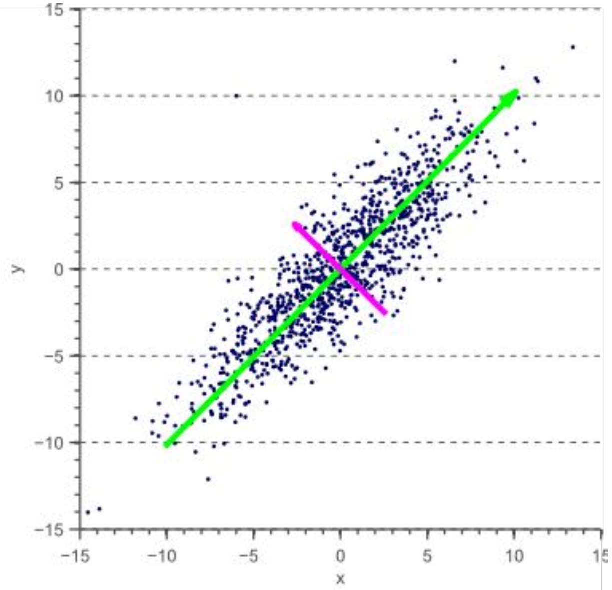
Chapter 5. dimension reduction

Principal component analysis

Dimension reduction method (unsupervised feature extraction)



PCA visualization

(blue dot: origin data, arrow: output vector)

PCA algorithm

1. Data pre-processing
2. create covariance matrix
3. eigen decomposition (covariance matrix)
4. select eigenvector (standard: eigen value)
5. space projection

Data pre-processing

Data mean -> 0

Data variance -> 1

sklearn.preprocessing.StandardScaler

class `sklearn.preprocessing.`**`StandardScaler`**(*copy=True, with_mean=True, with_std=True*) [source]

Standardize features by removing the mean and scaling to unit variance

The standard score of a sample `x` is calculated as:

z = (x - u) / s

where `u` is the mean of the training samples or zero if `with_mean=False`, and `s` is the standard deviation of the training samples or one if `with_std=False`.

```
from sklearn.preprocessing import StandardScaler
data1 = [[-9, 100], [2, 100]]
scaler = StandardScaler()
print(scaler.fit(data1))
print(scaler.transform(data1))
```

```
StandardScaler(copy=True, with_mean=True, with_std=True)
[[-1.  0.]
 [ 1.  0.]]
```

```
from sklearn.preprocessing import StandardScaler
data2 = [[-9, 100], [9, -100]]
scaler = StandardScaler()
print(scaler.fit(data2))
print(scaler.transform(data2))
```

```
StandardScaler(copy=True, with_mean=True, with_std=True)
[[-1.  1.]
 [ 1. -1.]]
```

Create covariance matrix

$$
\begin{array}{cc}
 & y \\
x & \begin{pmatrix} s_{0,0} & \cdots & s_{0,j} \\ \vdots & \ddots & \vdots \\ s_{i,0} & \cdots & s_{i,j} \end{pmatrix}
\end{array}
$$

covariance

– $\mathrm{cov}(x,y) = E[(x-m_x)(y-m_y)]$

Eigen decomposition

A (non-zero) vector $\mathbf{v}$ of dimension N is an **eigenvector** of a square $N \times N$ matrix $\mathbf{A}$ if it satisfies the linear equation

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

The decomposition can be derived from the fundamental property of eigenvectors:

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

$$\mathbf{A}\mathbf{Q} = \mathbf{Q}\mathbf{\Lambda}$$

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1}.$$

$$= \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

$$= \begin{bmatrix} \lambda_1 v_1 & \lambda_2 v_2 & \cdots & \lambda_n v_n \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

$$= \lambda_1 v_1 v_1^T + \lambda_2 v_2 v_2^T + \cdots + \lambda_n v_n v_n^T$$