LTAT.02.004 Machine Learning II

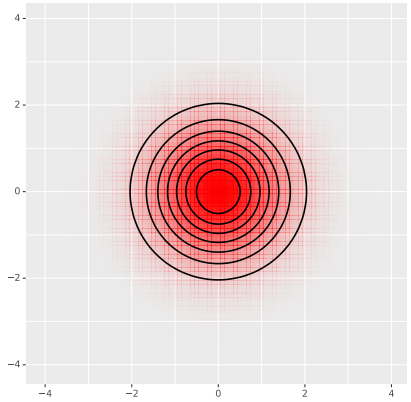
# Affine data projections

## based on normal distribution

Sven Laur
University of Tartu

# Multivariate normal distribution

# White Gaussian noise



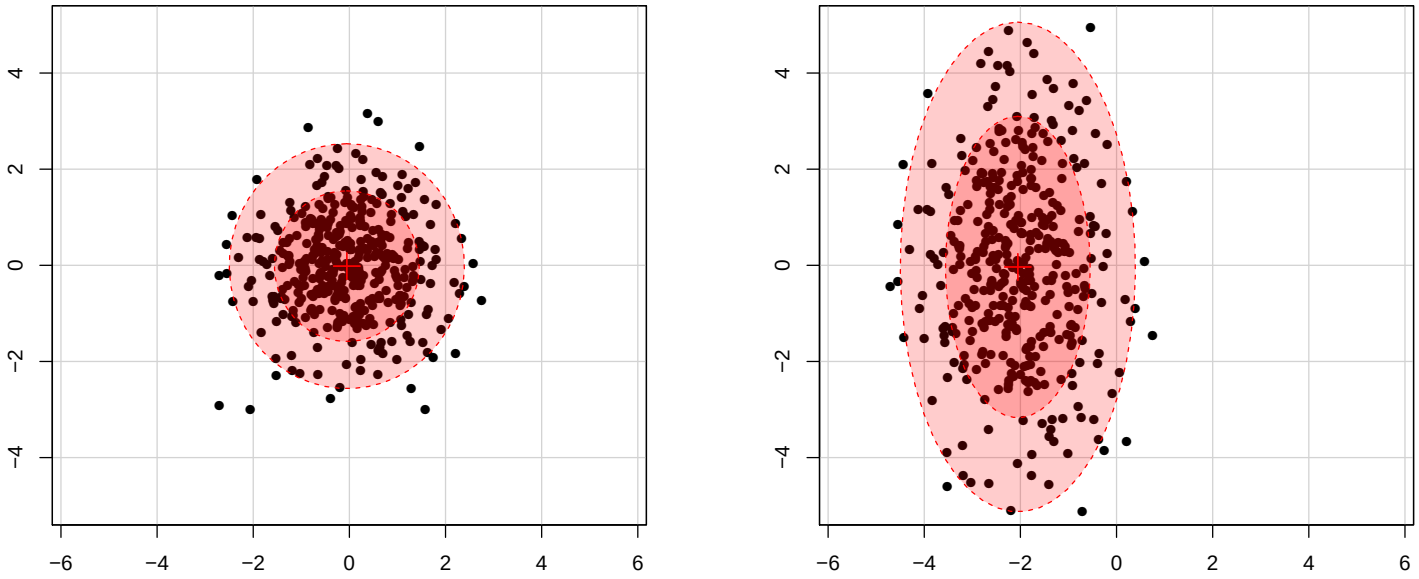

**Definition.** A random vector $X_1, \ldots, X_n$ is a standard normal random vector if all of its components are independent and and $X_i \sim \mathcal{N}(0, 1)$.

▷ The density can be computed based on independence:

$$p(x_1, \ldots, x_n) = p(x_1) \cdots p(x_n) = \frac{1}{(2\pi)^{n/2}} \cdot \exp\left( -\frac{x_1^2 + \cdots + x_n^2}{2} \right) .$$

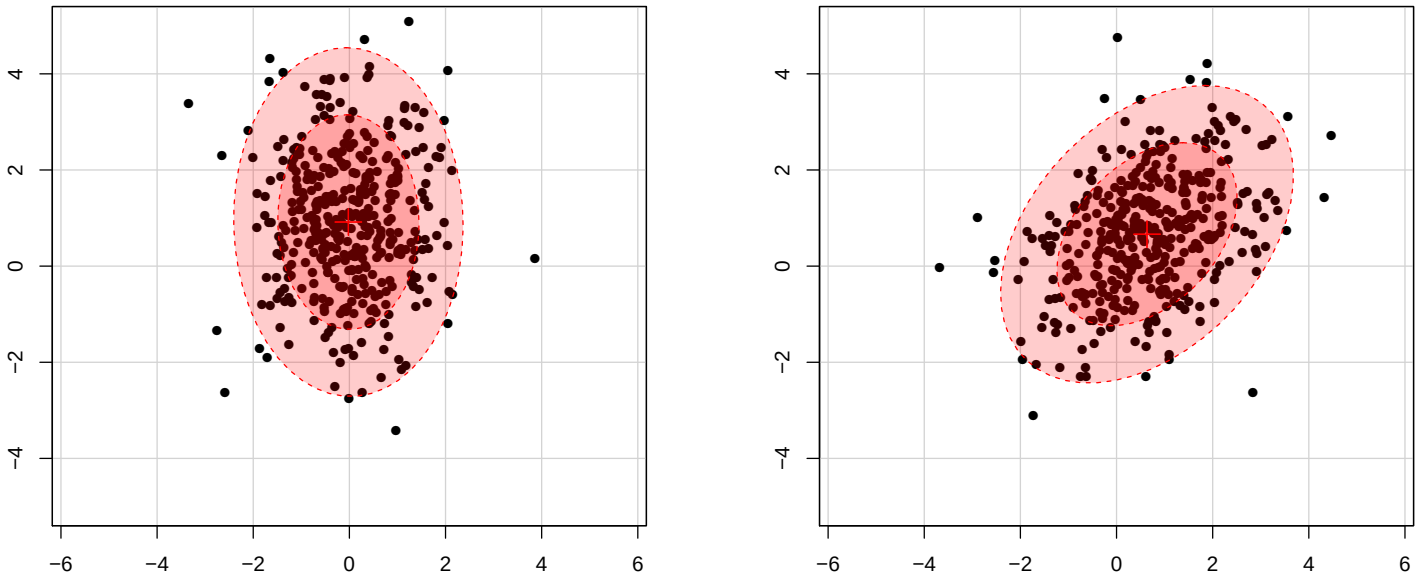
# Scaling and shifting

By shifting and scaling the source distribution $\mathcal{N}(\mathbf{0}, I)$ we can obtain some other instances of multivariate normal distribution.



# Necessity of rotations

As the choice of coordinate axis is sometimes arbitrary, there must be other ways to form a normal distribution – rotations of coordinate axis.




Any affine transformation can be expressed as scaling, rotating and shifting.

# Affine transformations

Let $\boldsymbol{x}$ be standard normal random vector and let $\boldsymbol{y}$ be obtained the scaling, translation and rotation of the coordinate plane.

Then we can express $\boldsymbol{x}$ and $\boldsymbol{y}$ in terms of an affine transformation

$$\begin{aligned} \boldsymbol{y} &= A\boldsymbol{x} + \boldsymbol{\mu} \ , \\ \boldsymbol{x} &= A^{-1}(\boldsymbol{y} - \boldsymbol{\mu}) \ . \end{aligned}$$

**Observation.** Affine transformations are closed with respect to composition, i.e., applying two affine transformations yields a new affine transformation.

**Remark.** Not all affine transformations are invertible.

# What is density in 2D?

Recall that density assigns probability to small enough regions $\mathcal{R}$:

$$\Pr_{x_1^*, x_2^*} \begin{bmatrix} x_1 \leq x_1^* \leq x_1 + \Delta x_1 \\ x_2 \leq x_2^* \leq x_2 + \Delta x_2 \end{bmatrix} = p(x_1, x_2) \cdot \underbrace{\Delta x_1 \Delta x_2}_{S} + \varepsilon$$

where $\varepsilon = o(\Delta x_1 \cdot \Delta x_2)$ in the process $\Delta x_1 \to 0$ and $\Delta x_2 \to 0$.

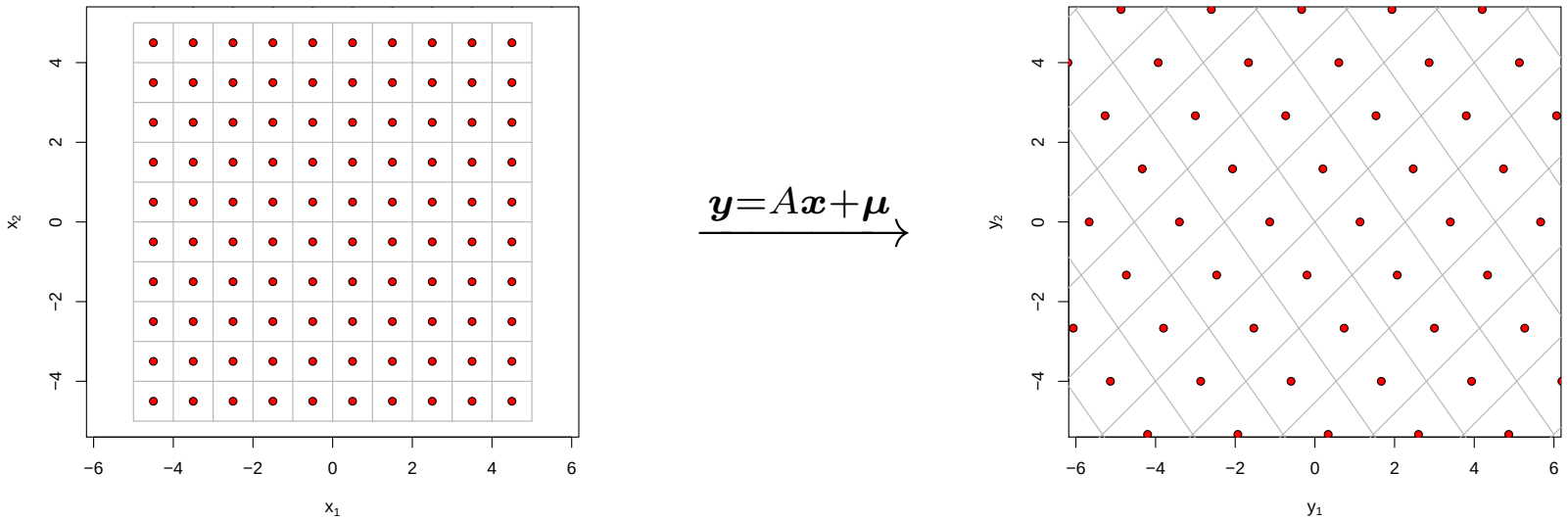
**Remark.** Regions $\mathcal{R}$ do not have to be rectangular as long as:

- ▷ The area $S(\mathcal{R})$ of a region can be computed.
- ▷ Probability can be assigned to the region $\mathcal{R}$ and its scalings.

Then $\varepsilon = o(S)$ when we rescale the region $\mathcal{R}$ around the point $(x_1, x_2)$.

# Density recalibration

Any affine transformation changes a square grid into parallelograms.




As a result, the area of the regions is different on the left and on the right:

$$p(x_1, x_2) \cdot S_1 \approx q(y_1, y_2) \cdot S_2 \qquad \Longrightarrow \qquad q(y_1, y_2) = \frac{S_1}{S_2} \cdot p(x_1, x_2)$$

Fortunately, the ratio between areas are constant over the entire plane!

# Density of two-variate normal distribution

The density of $(x_1, x_2)$ pairs can be computed based on independence:

$$p(x_1, x_2) = p(x_1) \cdot p(x_2) = \frac{1}{2\pi} \cdot \exp\left(-\frac{x_1^2 + x_2^2}{2}\right) \ .$$

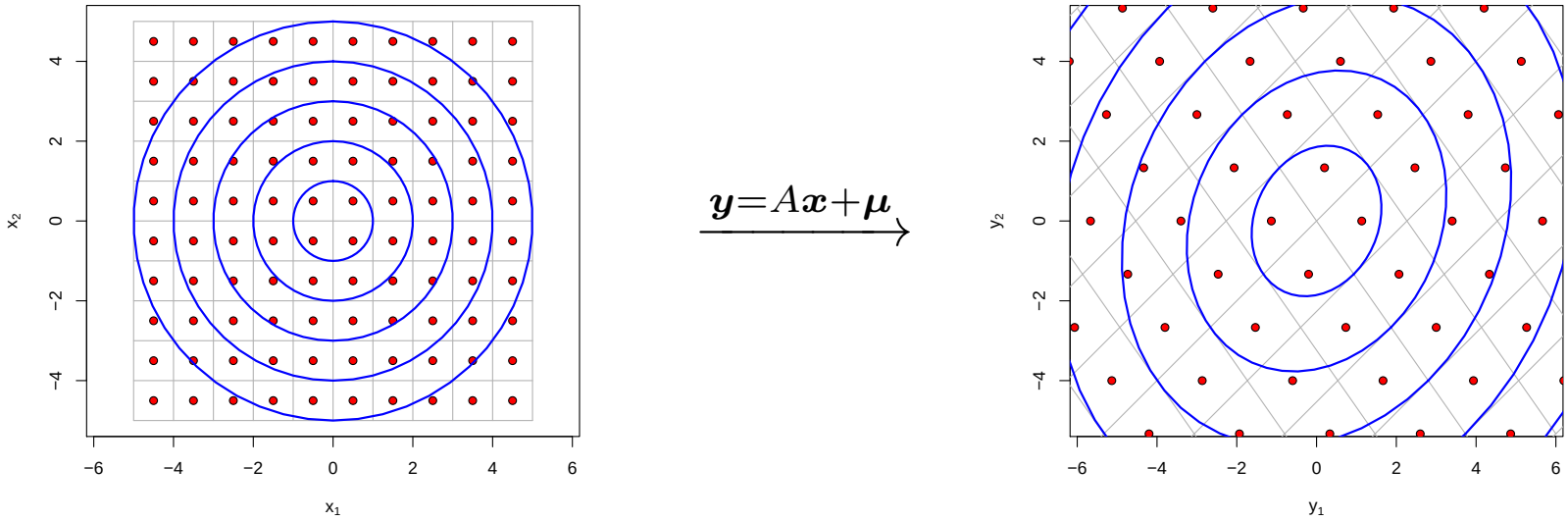
To estimate density $q(y_1, y_2)$, we must find the corresponding $(x_1, x_2)$:

$$\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{\mu} \quad \Leftrightarrow \quad \boldsymbol{x} = A^{-1}(\boldsymbol{y} - \boldsymbol{\mu}) \ .$$

Thus we get

$$\begin{aligned} q(y_1, y_2) &= \frac{S_1}{S_2} \cdot \frac{1}{2\pi} \cdot \exp\left(-\frac{(\boldsymbol{y} - \boldsymbol{\mu})^T A^{-T} A^{-1} (\boldsymbol{y} - \boldsymbol{\mu})}{2}\right) \\ &= \frac{1}{\sqrt{\det(\Sigma)}} \cdot \frac{1}{2\pi} \cdot \exp\left(-\frac{(\boldsymbol{y} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{y} - \boldsymbol{\mu})}{2}\right) \ . \end{aligned}$$

# Illustrative example




- ▷ Affine transformation changes the square grid into parallelograms.
- ▷ Affine transformation changes circular equiprobability lines into ellipses.
- ▷ The axes of the ellipses may intersect with the sides of parallelograms.

# Generalisation to multivariate case

If observed quantities $\boldsymbol{y}$ are generated by applying the affine transformation

$$\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{\mu} \quad \Leftrightarrow \quad \boldsymbol{x} = A^{-1}(\boldsymbol{y} - \boldsymbol{\mu})$$

to the *independent source signals* $x_1, \ldots, x_n \sim \mathcal{N}(0, 1)$, then the resulting distribution is *a multivariate normal distribution* with the density:

$$p(\boldsymbol{y}) = \frac{1}{(2\pi)^{n/2}} \cdot \frac{1}{\sqrt{\det(\Sigma)}} \cdot \exp\left(-\frac{(\boldsymbol{y} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{y} - \boldsymbol{\mu})}{2}\right)$$

where $\Sigma^{-1} = A^{-T} A^{-1}$ is *a positively definite symmetric matrix*.

# Important properties of normal distributions

# Closeness under marginalisation

Let $\boldsymbol{x}_{\mathcal{I}} = (x_i)_{i \in \mathcal{I}}$ be a subvector determined by the coordinate set $\mathcal{I}$. Then $\boldsymbol{x}_{\mathcal{I}}$ is distributed according to a multivariate normal distribution as long as the vector $\boldsymbol{x}$ comes form a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$.

▷ Moment matching gives the parameters of the resulting distribution

$$\mathbf{E}(\boldsymbol{x}_{\mathcal{I}}) = \mathbf{E}(\boldsymbol{x})_{\mathcal{I}} = \boldsymbol{\mu}_{\mathcal{I}}$$
$$\mathbf{Cov}(\boldsymbol{x}_{\mathcal{I}}) = \mathbf{Cov}(\boldsymbol{x})_{\mathcal{I} \times \mathcal{I}} = \Sigma[\mathcal{I}, \mathcal{I}]$$

# Closeness under linear combinations

Linear combination $y = \boldsymbol{\alpha}_1^T \boldsymbol{x}_1 + \boldsymbol{\alpha}_2^T \boldsymbol{x}_2$ of independent multivariate normal distributions $\boldsymbol{x}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$ and $\boldsymbol{x}_2 \sim \mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$ is also a multivariate normal distribution.

▷ Moment matching gives the parameters of the resulting distribution

$$
\begin{aligned}
\mathbf{E}(y) &= \boldsymbol{\alpha}_1^T \mathbf{E}(\boldsymbol{x}_1) + \boldsymbol{\alpha}_2^T \mathbf{E}(\boldsymbol{x}_2) = \boldsymbol{\alpha}_1^T \boldsymbol{\mu}_1 + \boldsymbol{\alpha}_2^T \boldsymbol{\mu}_2 \\
\mathbf{Var}(y) &= \mathbf{Cov}(\boldsymbol{\alpha}_1^T \boldsymbol{x}_1) + \mathbf{Cov}(\boldsymbol{\alpha}_2^T \boldsymbol{x}_2) \\
&= \boldsymbol{\alpha}_1^T \mathbf{Cov}(\boldsymbol{x}_1) \boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2^T \mathbf{Cov}(\boldsymbol{x}_2) \boldsymbol{\alpha}_2 \\
&= \boldsymbol{\alpha}_1^T \Sigma_1 \boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2^T \Sigma_2 \boldsymbol{\alpha}_2
\end{aligned}
$$

▷ Closeness under linear combinations holds also for matrix combinations.

# Closeness under conditioning

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be related random variables. Let $\boldsymbol{x}|\boldsymbol{y}_*$ denote the conditional distribution of $\boldsymbol{x}$ given that a random variable $\boldsymbol{y}$ has a fixed value $\boldsymbol{y}_*$. Then $\boldsymbol{x}|\boldsymbol{y}_*$ is distributed according to a multivariate normal distribution provided that $(\boldsymbol{x}, \boldsymbol{y})$ comes form a multivariate normal distribution $\mathcal{N}((\boldsymbol{\mu}_i), (\Sigma_{ij}))$

▷ Moment matching gives the parameters of the resulting distribution

$$\mathbf{E}(\boldsymbol{x}|\boldsymbol{y}_*) = \boldsymbol{\mu}_1 + \Sigma_{1,2}\Sigma_{2,2}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_2)$$

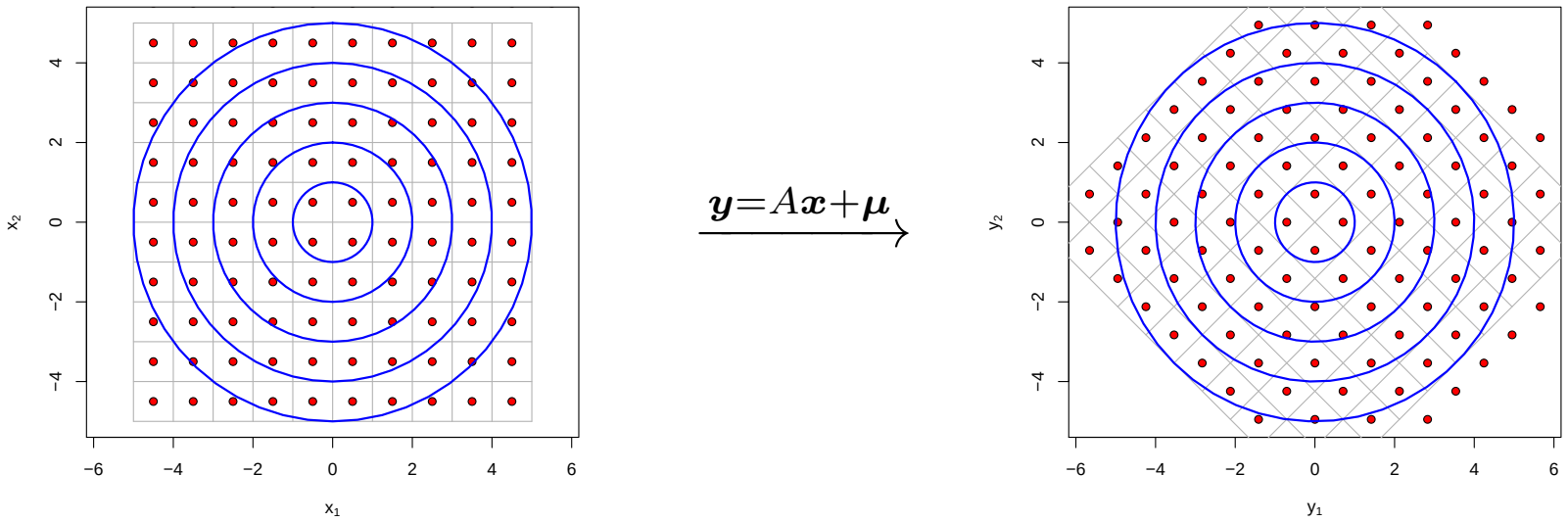
$$\mathbf{Cov}(\boldsymbol{x}|\boldsymbol{y}_*) = \Sigma_{1,1} - \Sigma_{1,2}\Sigma_{2,2}^{-1}\Sigma_{2,1}$$

# Principal component analysis

# Distribution reconstruction task

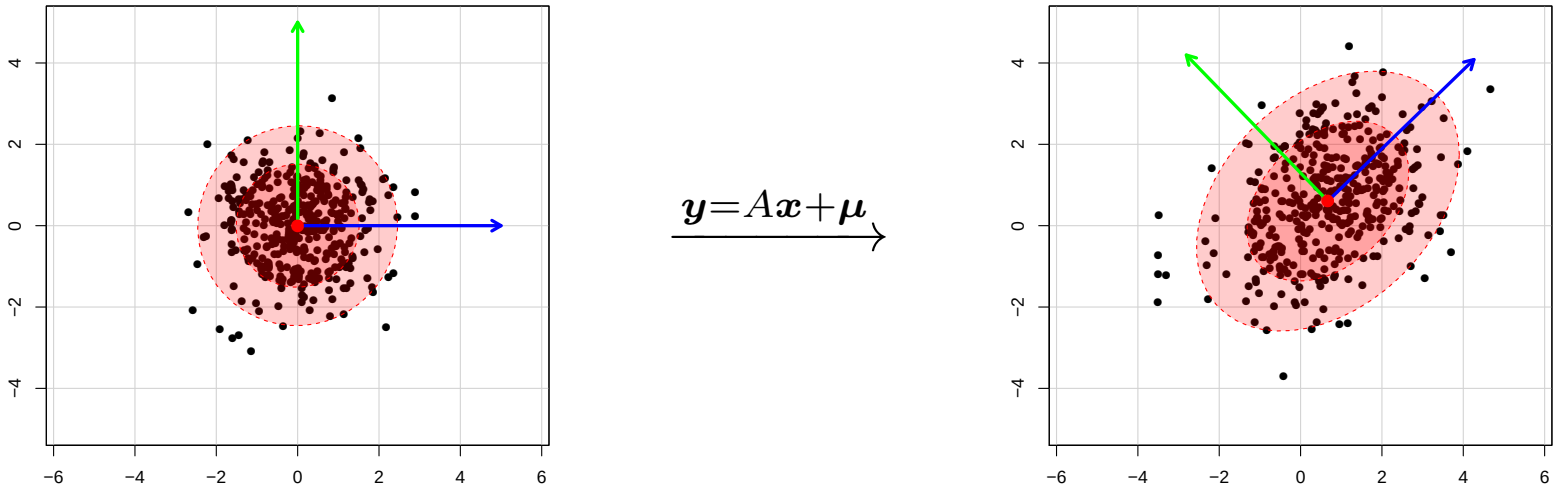
**Original goal.** Given the set of observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ determine the affine transformation $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{\mu}$ and original source signals $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$.

**Impossibility result.** The matrix $A$ can be recovered *only* up to rotations.



# Simplified distribution reconstruction task

**Achievable goal.** Given the set of observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ determine the affine transformation by fixing the centre and axis of the ellipsoid.




- ▷ We need to find the origin and semi-axes $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ of the ellipsoid.
- ▷ Unit vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n$ are mapped to semi-axes $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ of ellipsoid.

# Variance for a fixed direction

**Fact.** Ortogonal projection onto a unit vector $\boldsymbol{w}$ is given by scalar product.

**Question.** What is the direction $\boldsymbol{w}$ that maximises the variance for ellipsoid?

$$\mathbf{Var}(\boldsymbol{w}^T \operatorname{diag}(\boldsymbol{a})\boldsymbol{x}) = \mathbf{Var}\left(\sum_{i=1}^{n} w_i a_i x_i\right) = \sum_{i=1}^{n} w_i^2 a_i^2 \ .$$

The variance is maximised in the direction of the longest ellipse axis $a_1$.

**Question.** How is the center of the ellipsoid and mean values connected?

$$\mathbf{E}(A\boldsymbol{x} + \boldsymbol{\mu}) = \mathbf{E}(A\boldsymbol{x}) + \mathbf{E}(\boldsymbol{\mu}) = \boldsymbol{\mu} \ .$$

# Principal component analysis

- Compute the average value of the observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$:

$$\hat{\boldsymbol{\mu}} \leftarrow \frac{\boldsymbol{y}_1 + \cdots + \boldsymbol{y}_m}{m} \ .$$

- Centre the data by substituting $\hat{\boldsymbol{\mu}}$:

$$\boldsymbol{y}_i \leftarrow \boldsymbol{y}_i - \hat{\boldsymbol{\mu}}, \qquad i \in \{1, \ldots, m\} \ .$$

- Find the unit direction $\boldsymbol{w}_1$ that has *a maximal empirical* variance:

$$F(\boldsymbol{w}) = \mathbf{Var}(\boldsymbol{w}^T\boldsymbol{y}_1, \ldots, \boldsymbol{w}^T\boldsymbol{y}_n) = \frac{(\boldsymbol{w}^T\boldsymbol{y}_1)^2 + \cdots + (\boldsymbol{w}^T\boldsymbol{y}_m)^2}{m} \ .$$

- Find unit directions $\boldsymbol{w}_i$ orthogonal to previous directions that maximise the empirical variance of the corresponding the projection onto $\boldsymbol{w}_i$.

# Covariance matrix and optimisation goal

We can use matrix algebra to simplify the variance estimate

$$F(\boldsymbol{w}) = \frac{1}{m} \cdot \left( \boldsymbol{w}^T \boldsymbol{y}_1 \boldsymbol{y}_1^T \boldsymbol{w} + \cdots + \boldsymbol{w}^T \boldsymbol{y}_m \boldsymbol{y}_m^T \boldsymbol{w} \right)$$
$$= \boldsymbol{w}^T \left( \frac{\boldsymbol{y}_1 \boldsymbol{y}_1^T + \cdots + \boldsymbol{y}_m \boldsymbol{y}_m^T}{m} \right) \boldsymbol{w}$$

The $n \times n$ matrix in the middle is known as a *covariance matrix* $\Sigma$.

Due to the restriction $\|\boldsymbol{w}\|_2^2 = \boldsymbol{w}^T \boldsymbol{w} = 1$, we have to use Lagrange' trick:

$$F_*(\boldsymbol{w}) = \boldsymbol{w}^T \Sigma \boldsymbol{w} - 2\lambda \boldsymbol{w}^T \boldsymbol{w} \qquad \Rightarrow \qquad \frac{\partial F_*(\boldsymbol{w})}{\partial \boldsymbol{w}} = 2\Sigma \boldsymbol{w} - 2\lambda \boldsymbol{w} = \mathbf{0}.$$

# Principal components as eigenvectors

The $F_*(\boldsymbol{w})$ is maximised only if the direction $\boldsymbol{w}$ is an *eigenvector* of $\Sigma$:

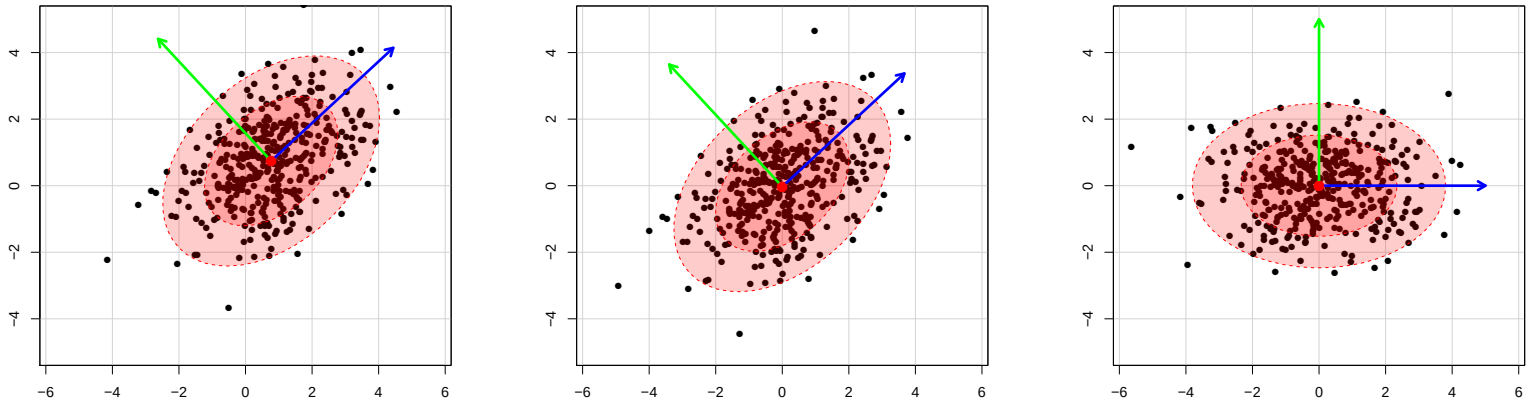
$$\Sigma \boldsymbol{w} = \lambda \boldsymbol{w} \qquad \Rightarrow \qquad \boldsymbol{w}^T \Sigma \boldsymbol{w} = \boldsymbol{w}^T \lambda \boldsymbol{w} = \lambda \ .$$

**Fact.** If $n \times n$ matrix is symmetric and positively definite then there exists $n$ orthogonal eigenvectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n$ with *eigenvalues* $\lambda_1 \geq \ldots \geq \lambda_n > 0$.

**Corollary.** Principal components corresponding to observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ are the eigenvectors of the covariance matrix $\Sigma$.

# Principal component analysis as a rotation

Reconstruction of the source signal can be viewed as a *translation* followed by a *rotation* to orientate the ellipsoid wrt coordinate axis.



As vectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n$ are orthogonal, the rotation can be done through computing projections (read scalar products):

$$\hat{\boldsymbol{x}}_i = (\boldsymbol{w}_1 || \cdots || \boldsymbol{w}_n)^T (\boldsymbol{y}_i - \hat{\boldsymbol{\mu}}_0) = W(\boldsymbol{y}_i - \hat{\boldsymbol{\mu}}) \quad .$$

# Maximum likelihood estimate

The algorithm formulated above was based on *ad hoc* reasoning:

- ▷ Empirical estimates for the mean and variance are not precise!

Theoretically correct way to handle the problem is

- ▷ obtain the maximum likelihood estimate on the model parameters,
- ▷ determine the translation and rotation based on the model parameters.

What are the model parameters?

- ▷ Parameters of the density formula $\Sigma$ and $\boldsymbol{\mu}$.
- ▷ Parameters of the affine transformation $A$ and $\boldsymbol{\mu}$.

# Likelihood function under iid assumption

If all observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ are independent then

$$p[\boldsymbol{y}_i, \ldots, \boldsymbol{y}_m | \Sigma, \boldsymbol{\mu}] = \prod_{i=1}^{m} p[\boldsymbol{y}_i | \Sigma, \boldsymbol{\mu}]$$

where

$$p[\boldsymbol{y}_i | \Sigma, \boldsymbol{\mu}] = \frac{1}{(2\pi)^{n/2}} \cdot \frac{1}{\sqrt{\det(\Sigma)}} \cdot \exp\left( -\frac{(\boldsymbol{y}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu})}{2} \right)$$

The *log-likelihood* of the data $\ln p[\boldsymbol{y}_i, \ldots, \boldsymbol{y}_m | \Sigma, \boldsymbol{\mu}]$ can be expressed

$$\mathcal{L}(\Sigma, \boldsymbol{\mu}) = const + \frac{m}{2} \cdot \ln \det(\Sigma^{-1}) - \sum_{i=1}^{m} \frac{(\boldsymbol{y}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu})}{2}$$

Now we have to find the arrangement $(\Sigma, \boldsymbol{\mu})$ that maximises $\mathcal{L}(\Sigma, \boldsymbol{\mu})$.

# Gradients of the log-likelihood function

Gradient with respect to the shift $\boldsymbol{\mu}$:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = -\sum_{i=1}^{m} \frac{\partial}{\partial \boldsymbol{\mu}} \frac{(\boldsymbol{y}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu})}{2} = -\sum_{i=1}^{m} \frac{\Sigma^{-1}(\boldsymbol{y}_i - \boldsymbol{\mu})}{2} \cdot (-1)$$

Gradient with respect to the inverse matrix $\Sigma^{-1}$:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial (\Sigma^{-1})} &= \frac{m}{2} \cdot \frac{\partial}{\partial (\Sigma^{-1})} \ln \det(\Sigma^{-1}) - \sum_{i=1}^{m} \frac{\partial}{\partial (\Sigma^{-1})} \frac{(\boldsymbol{y}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu})}{2} \\ &= \frac{m}{2} \cdot \Sigma^T - \sum_{i=1}^{m} \frac{(\boldsymbol{y}_i - \boldsymbol{\mu})^T (\boldsymbol{y}_i - \boldsymbol{\mu})}{2}\end{aligned}$$

As $\Sigma$ is symmetric and $\Sigma^{-1}$ exists we can derive closed form solutions.

# Maximum likelihood estimates for parameters

The shift must be the mean of all observations

$$\boldsymbol{\mu} = \frac{1}{m} \cdot \sum_{i=1}^{m} \boldsymbol{y}_i \ .$$

The covariance matrix

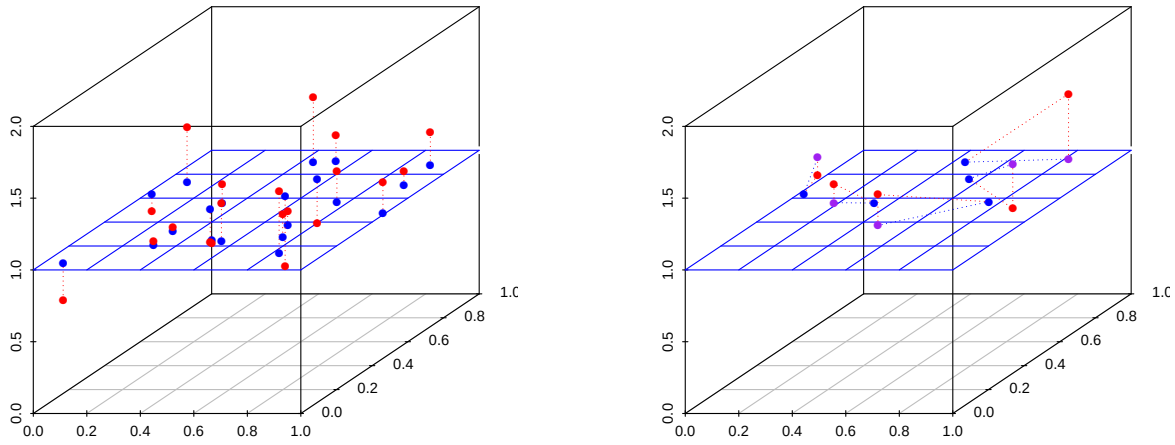
$$\Sigma = \frac{1}{m} \cdot \sum_{i=1}^{m} (\boldsymbol{y}_i - \boldsymbol{\mu})^T (\boldsymbol{y}_i - \boldsymbol{\mu})$$

**Correctness of PCA.** As ML estimates are exactly the same we used in principal component analysis, the method is theoretically justified!

# Principal component analysis
# Alternative formalisations

# Dimensionality reduction

What if the actual data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ lies in a lower-dimensional plane and the observation $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ are obtained by random shifts?




The shifts can be either orthogonal to the plane or just random. The first model is easier to analyse while the second is more plausible.

# Maximum likelihood estimate

Let $\mathcal{H}$ be the plane. Assume that the random shifts $\varepsilon_i$ are orthogonal to the plane and have a normal distribution $\mathcal{N}(0, \sigma I)$. Then

$$p[\boldsymbol{y}_i|\mathcal{H}, \sigma] = const \cdot \exp\left(-\frac{d_i^2}{2\sigma^2}\right)$$

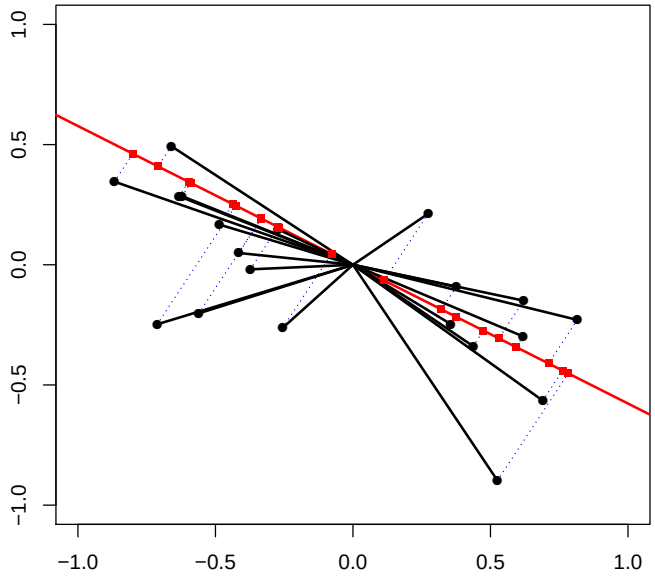
where $d_i$ is the distance between the plane $\mathcal{H}$ and the point $\boldsymbol{y}_i$. Thus

$$p[\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m|\mathcal{H}, \sigma] = const \cdot \exp\left(-\sum_{i=1}^{m} \frac{d_i^2}{2\sigma^2}\right)$$

and the maximum likelihood estimate of the plane minimises sum of the distance squares. Corresponding estimates of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ are projections of $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ to the plane $\mathcal{H}$.

# Another characterisation of PCA

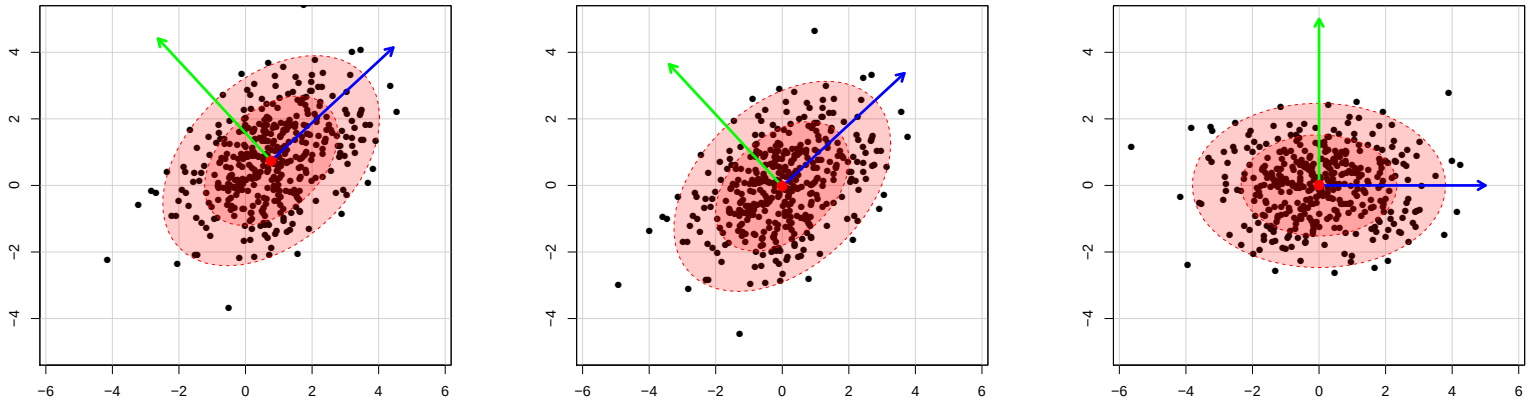
**Fact.** If the data is centred then PCA chooses the direction $\boldsymbol{w}_1$ such that the sum of squares of the projections $\boldsymbol{w}_1^T \boldsymbol{y}_i$ is maximal.




**Corollary.** PCA chooses directions $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n$ such that the sum of distance squares from the hyperplane formed by $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_k$ is minimal.

# PCA as a dimensionality reduction tool

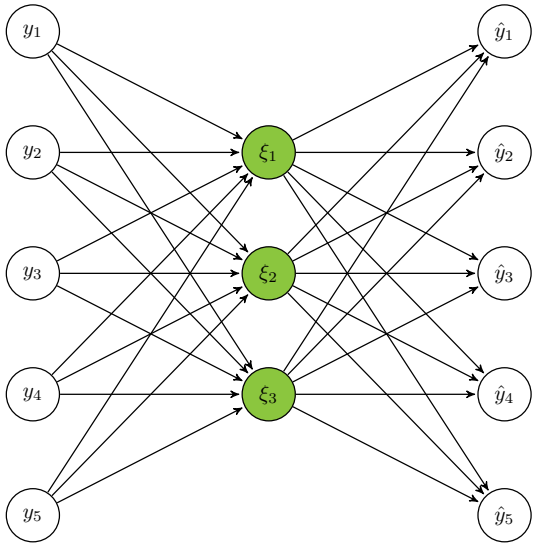
**Corollary.** PCA rotates the data such way that first $k$ coordinates of the rotated data correspond to maximum likelihood reconstructions of original vectors corrupted with white Gaussian noise $\mathcal{N}(0, \sigma I)$.



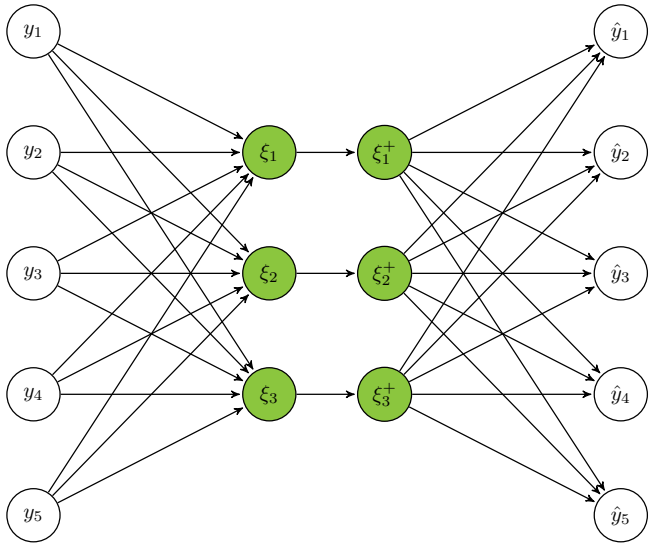
Alternatively, we can view the last components of the source signal $\boldsymbol{x}$ as the uninformative noise. The overall noise component should be small.

# Connection to autoencoders

Linear autonecoder




RELU autoencoder

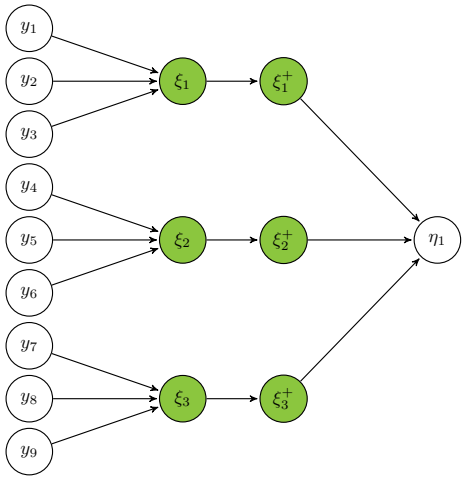



Fix mean square error as optimisation target.

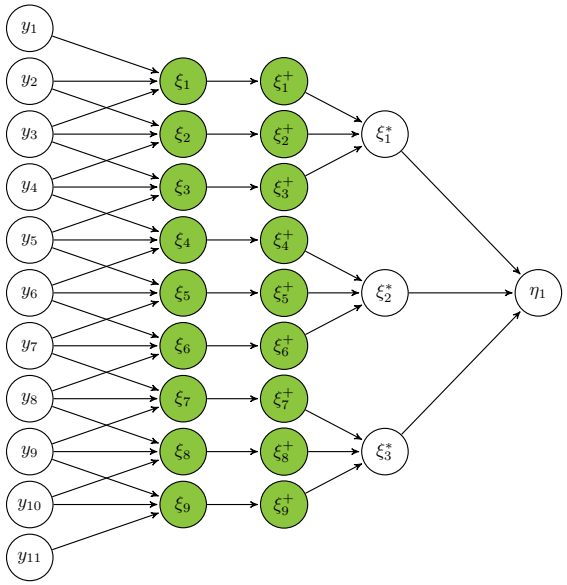
- ▷ Linear-autoencoder is a reformulation of PCA.
- ▷ RELU-autoencoder is a reformulation of non-negative matrix factorisation.

# Connection to convolutional neural networks

Hierarchical NMF




Convolutional network




- ▷ Hierarchical NMF applies the same transformation on patches.
- ▷ Convolutional layer applies the same transformation to sliding windows.
- ▷ Then it picks the strongest response among shifts of the transformation.

# Linear discriminant analysis

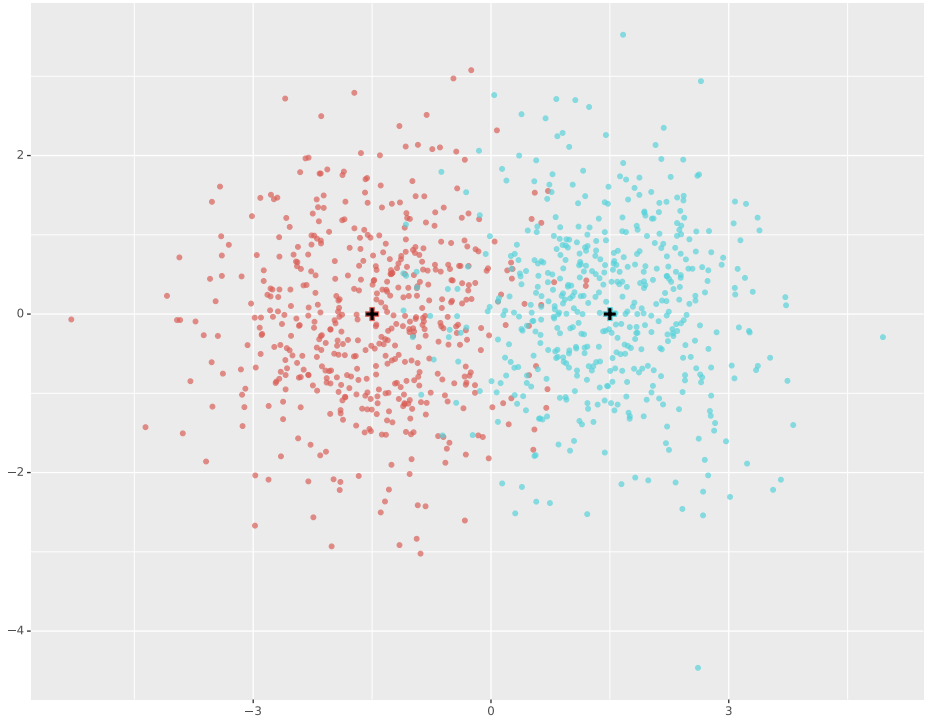
# Underlying assumptions and inference task

**Original goal.** Given a set of observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^M$ together with class labels $z_1, \ldots, z_n \in \{1, \ldots, \ell\}$ find a linear projection $\pi : \mathbb{R}^m \to \mathbb{R}^k$ so that individual classes are maximally separated.

**Assumptions.**

- ▷ There are $\ell$ different classes.
- ▷ All observations $\boldsymbol{x}_i$ are independently sampled.
- ▷ Observations $\boldsymbol{x}_i$ with the same class label $z_i$ come from $\mathcal{N}(\boldsymbol{\mu}_j, \Sigma)$.
- ▷ The covariance matrix $\Sigma$ is shared between different distributions.

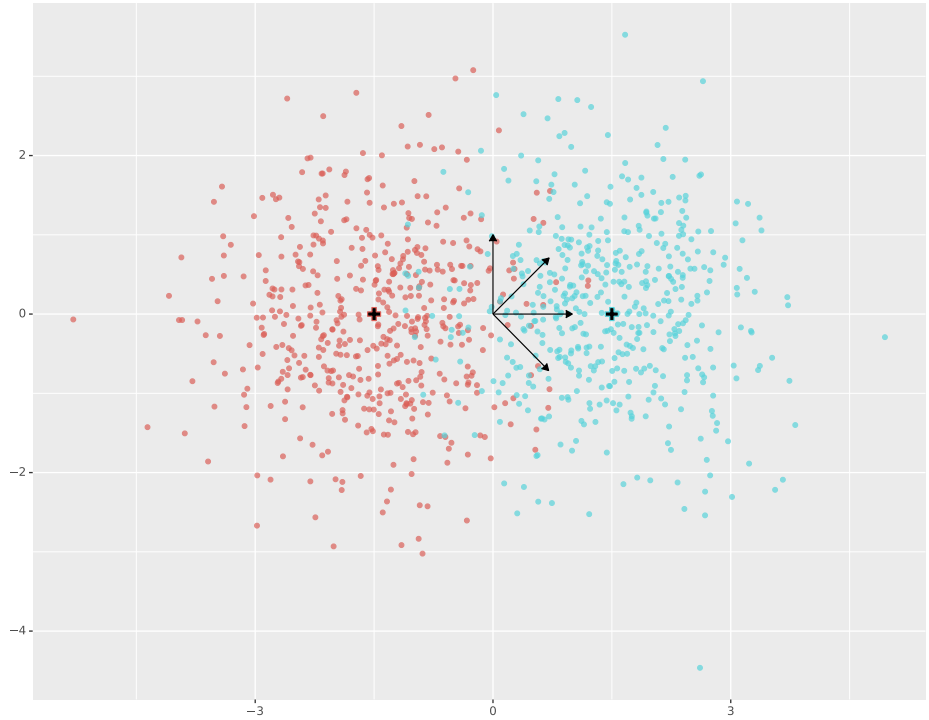
# LDA for spherical normal distributions




We assume that the covariance matrix $\Sigma$ is identity matrix:

- ▷ All vector components have unit variance.
- ▷ Different vector components are independent.

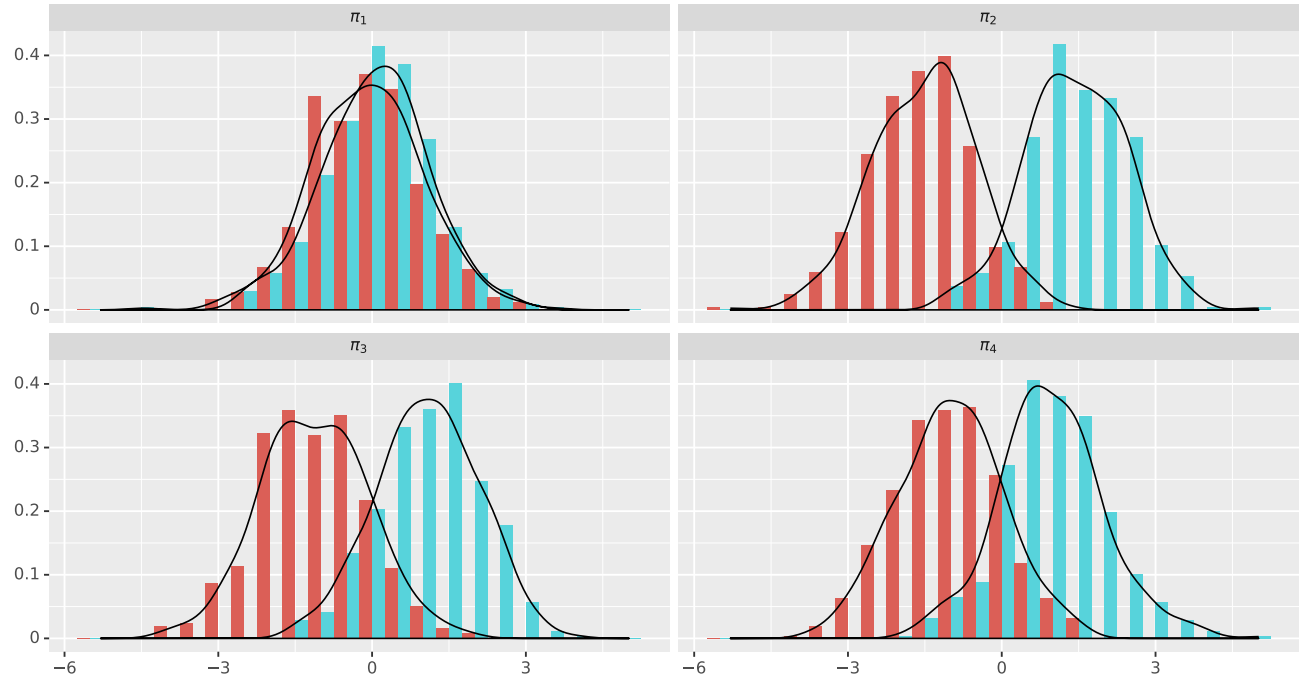
# Projections to one-dimensional subspace




A projection to one-dimensional space is determined by a vector $\boldsymbol{w}$:

- ▷ To get orthogonal projection the length of $\boldsymbol{w}$ must be one.
- ▷ This can be forced by the constraint $\boldsymbol{w}^T\boldsymbol{w} = 1$.

# Projections lead to different separation




We need a need a measure for assessing the goodness of separation:

- ▷ We can use Bayesian factors from statistics.
- ▷ We can use signal-to-noise ratio from signal-processing.

# Choice between alternative hypotheses

- ▷ **Hypothesis $\mathcal{H}_0$.** Projections $y_i, \ldots, y_n$ come from $\mathcal{N}(\bar{y}, 1)$.
- ▷ **Hypothesis $\mathcal{H}_1$.** Projection $y_i$ with label $z_i$ comes from a $\mathcal{N}(\bar{y}_{z_i}, 1)$.

Hypotheses lead to following probability assignments

$$p[y_i|\mathcal{H}_0] = \frac{1}{\sqrt{2\pi}} \cdot \exp\left(-\frac{1}{2}(y_i - \bar{y})^2\right)$$
$$p[y_i|\mathcal{H}_1] = \frac{1}{\sqrt{2\pi}} \cdot \exp\left(-\frac{1}{2}(y_i - \bar{y}_{z_i})^2\right)$$

If we have not preference then the corresponding Bayes factor is

$$\frac{\Pr[\mathcal{H}_1|y_1, \ldots, y_n]}{\Pr[\mathcal{H}_0|y_1, \ldots, y_n]} = \exp\left(\frac{1}{2} \cdot \sum_{i=1}^{n}(y_i - \bar{y})^2 - \frac{1}{2} \cdot \sum_{i=1}^{n}(y_i - \bar{y}_{z_i})^2\right)$$

# The corresponding optimisation task

Given a set of observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^M$ together with class labels $z_1, \ldots, z_n \in \{1, \ldots, \ell\}$ find a vector $\boldsymbol{w}$ with unit length that maximises:

$$F = \sum_{i=1}^{n} (y_i - \bar{y})^2 - \sum_{i=1}^{n} (y_i - \bar{y}_{z_i})^2$$

where $\mathcal{I}_j = \{i : z_i = j\}$ is the index set and $\bar{y}$ and $\bar{y}_j$ are cluster means:

$$\bar{y} = \frac{1}{n} \cdot \sum_{i=1}^{n} y_i$$

$$\bar{y}_j = \frac{1}{|\mathcal{I}_j|} \cdot \sum_{i \in \mathcal{I}_j} y_j$$

# Consequences of variance decomposition

Given a set of observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^M$ together with class labels $z_1, \ldots, z_n \in \{1, \ldots, \ell\}$ find a vector $\boldsymbol{w}$ with unit length that maximises:

$$F = \sum_{i=1}^{n} (\bar{y}_{z_i} - \bar{y})^2$$

Proof. The result follows directly form the variance decomposition

$$\sum_{i=1}^{n} (y_i - \bar{y})^2 = \sum_{i=1}^{n} (y_i - \bar{y}_{z_i})^2 + \sum_{i=1}^{n} (\bar{y}_{z_i} - \bar{y})^2$$

# Matrix magic

Let us define centres in the original data

$$\boldsymbol{\mu} = \frac{1}{n} \cdot \sum_{i=1}^{n} \boldsymbol{x}_i \qquad\qquad \boldsymbol{\mu}_j = \frac{1}{|\mathcal{I}_j|} \cdot \sum_{i \in \mathcal{I}_j} \boldsymbol{x}_j$$

Then we can express

$$\begin{aligned} F = \sum_{i=1}^{n} (\bar{y}_{z_i} - \bar{y})^2 \;\; &= \sum_{i=1}^{n} (\boldsymbol{w}^T \boldsymbol{\mu}_{z_i} - \boldsymbol{w}^T \boldsymbol{\mu})(\boldsymbol{w}^T \boldsymbol{\mu}_{z_i} - \boldsymbol{w}^T \boldsymbol{\mu})^T \\ &= \boldsymbol{w}^T \left( \sum_{i=1}^{n} (\boldsymbol{\mu}_{z_i} - \boldsymbol{\mu})(\boldsymbol{\mu}_{z_i} - \boldsymbol{\mu})^T \right) \boldsymbol{w} \end{aligned}$$

# Corresponding eigenvector task

Find a vector $\boldsymbol{w}$ with unit length that maximises

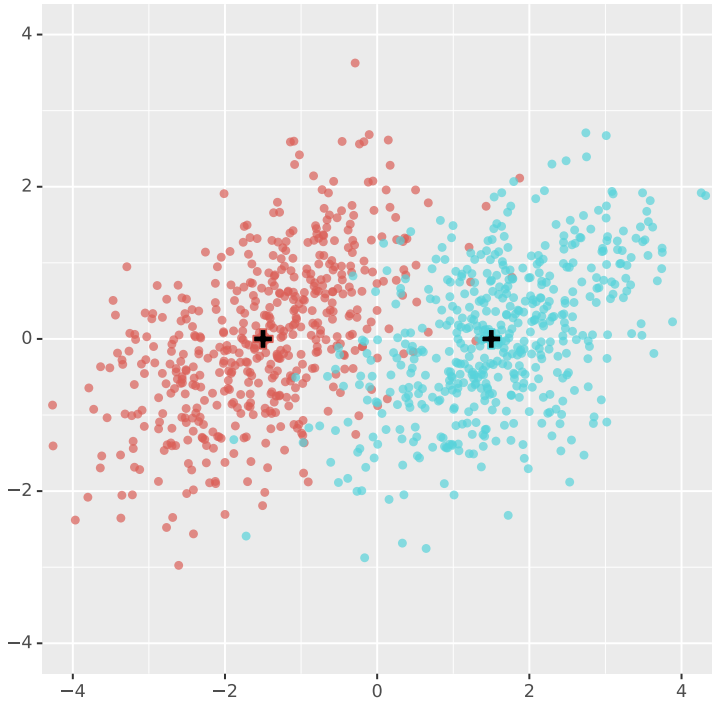
$$F = \boldsymbol{w}^T S_B \boldsymbol{w}$$

where $S_B$ is the between class scatter matrix;

$$S_B = \sum_{i=1}^{n} (\boldsymbol{\mu}_{z_i} - \boldsymbol{\mu})(\boldsymbol{\mu}_{z_i} - \boldsymbol{\mu})^T \ .$$

**Consequence.** The function $F$ is maximised by the eigenvector $\boldsymbol{w}$ of $S_B$ with the highest eigenvalue $\lambda_1$.

# LDA for a normal distribution with any shape




- ▷ As we know cluster labels we can remove the effect of $\boldsymbol{\mu}_1, \ldots \boldsymbol{\mu}_\ell$.
- ▷ After that we can do affine transformation that set the covariance to $I$.
- ▷ We know how to solve the task in the transformed space.

# Data whitening transformation

A linear transformation $\boldsymbol{x}^* = A\boldsymbol{x}$ leads to a unit covariance $I$ if

$$A\Sigma_W A^T = I$$

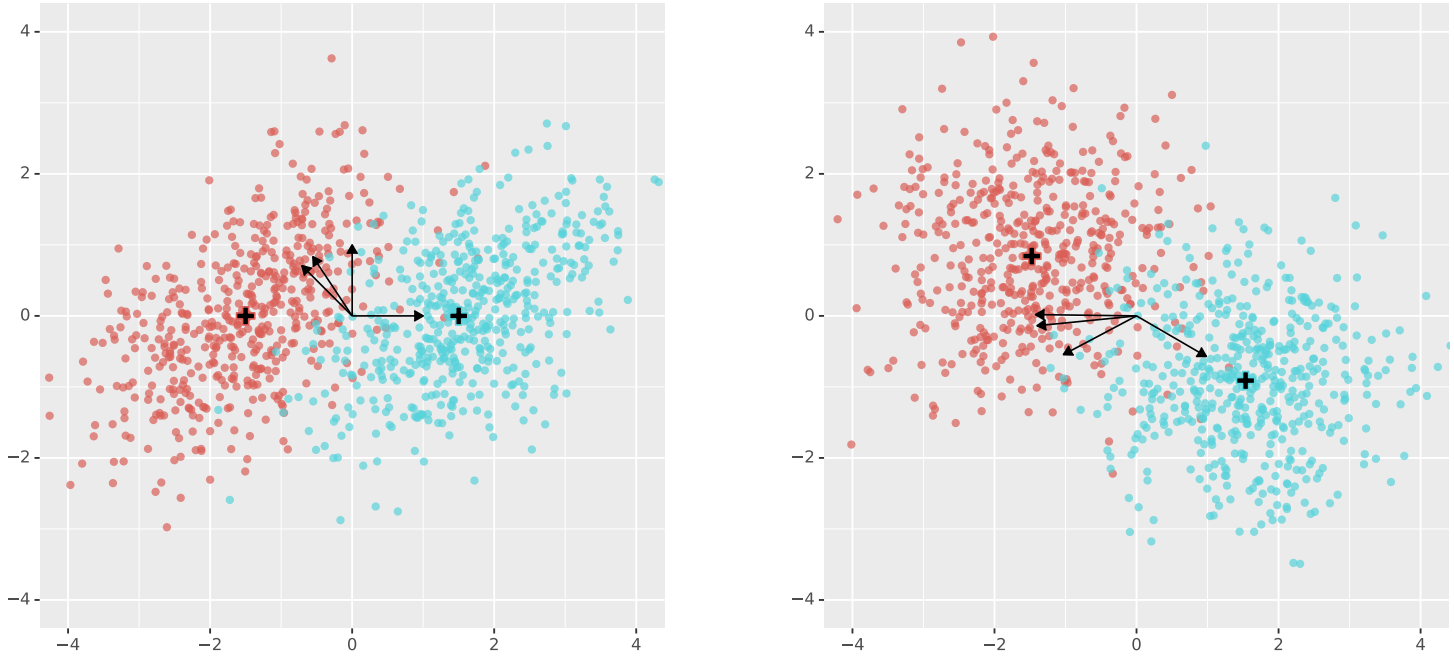
where $\Sigma_W$ is within class covariance matrix:

$$\Sigma_W = \frac{1}{n} \cdot \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\mu}_{z_i})(\boldsymbol{x}_i - \boldsymbol{\mu}_{z_i})^T$$

Let $W$ be the matrix where column vectors $\boldsymbol{w}_i$ are orthonormal eigenvectors with eigenvalues $\lambda_1, \ldots, \lambda_n$. Then we can express

$$A = \mathrm{diag}(\lambda_1^{-1/2}, \ldots, \lambda_n^{-1/2}) W^T \quad .$$

# The effects of data whitening




- ▷ Data whitening alters probing directions: $\boldsymbol{w}^* = A\boldsymbol{w}$.
- ▷ Data whitening alters between class scatter: $S_B^* = AS_BA^T$.
- ▷ Maximisation task in original terms: $\sum_{i=1}^{k} \boldsymbol{w}_i^T \Sigma_W^{-T} S_B \Sigma_W^{-1} \boldsymbol{w}_i \to \max$
- ▷ Othogonality constraints in original terms: $\boldsymbol{w}_i^T \Sigma_W^{-1} \boldsymbol{w}_j = \delta_{ij}$.

# Numerical stabilisation

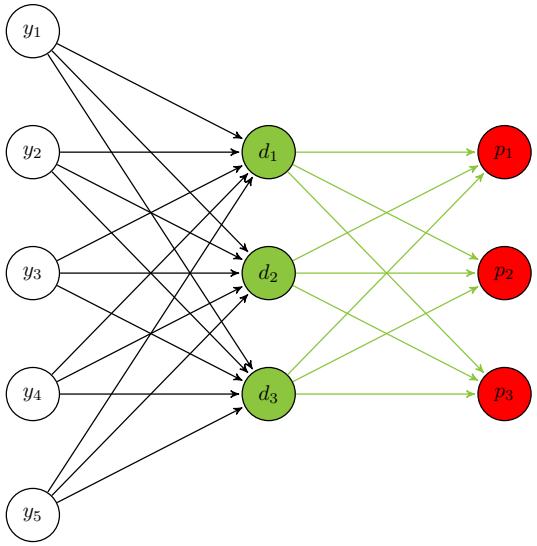
Whitening matrix $\Sigma_W$ can be non-invertible and it can also depend heavily on the perturbations of original datapoints. Ridge stabilisation

$$\Sigma_W^* = \Sigma_W + \rho I$$

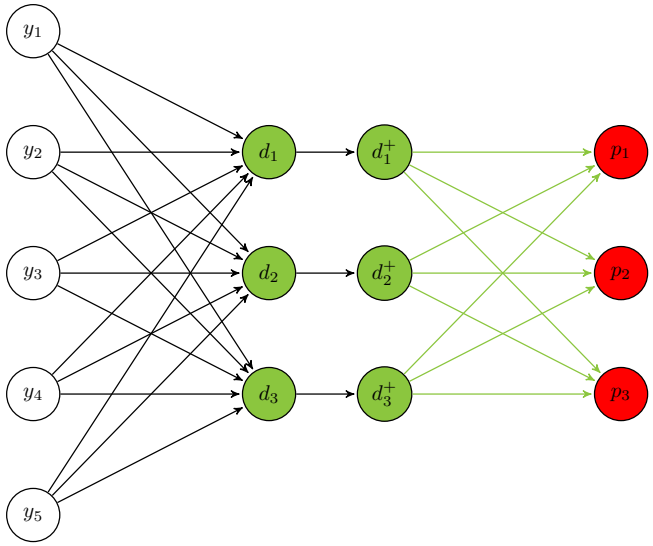
for small value $\rho > 0$ makes linear discriminant analysis more stable.

# Connection to neural networks

Linear + SoftMax




RELU + SoftMax




- ▷ LDA is equivalent to a linear layer followed by a soft-max layer.
- ▷ Neural networks usually use RELU nodes instead of pure linear nodes.
- ▷ This again introduces non-negativity constraint to features.

# Reconstruction vs discrimination

- ▷ PCA and LDA reduce dimensionality.
- ▷ PCA preserves recoverability of the original data.
- ▷ LDA preserves distinguishability between different classes.

Sometimes discriminative are overly selective:

- ▷ Decision is made based on minute details of the data
- ▷ Predictions are not robust against malicious perturbations.

We can quantify the balance between reconstruction vs discrimination.

- ▷ We can measure how much variation LDA projection explains.
- ▷ We can measure robustness against input perturbations.
- ▷ Same measures are applicable for other models such as neural networks.

# Going beyond basics

# Going beyond PCA and LDA

Weighted Principal Component Analysis:

- ▷ Sometimes data contains potential outliers.
- ▷ Sometimes we can assign reliability scores to the data points.

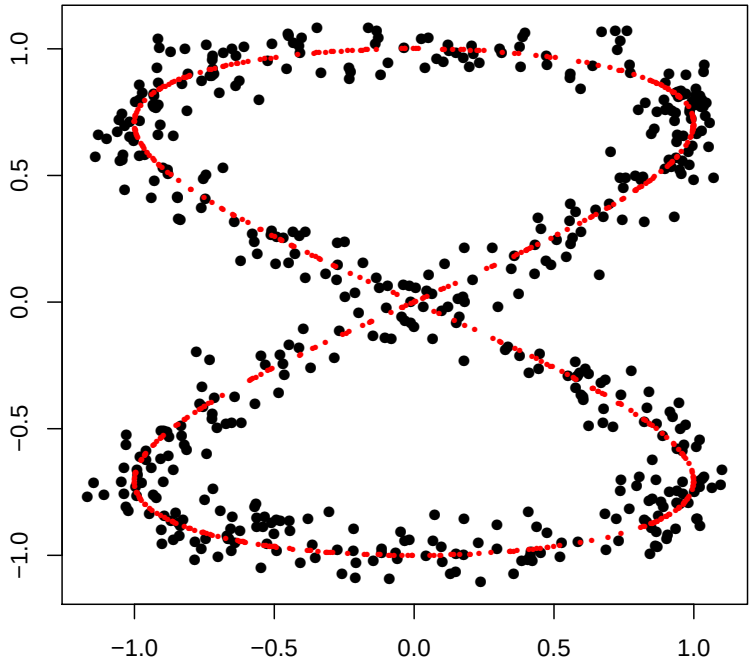
Principal curves and manifolds

- ▷ The original data might be on a low dimensional manifold.
- ▷ The observed data is corrupted by additive white gaussian noise.
- ▷ The task is to reconstruct the manifold and ML estimate for the data.

Independent Component Analysis

- ▷ What if the source components are non-gaussian?
- ▷ Then the reconstruction is possible up to scaling!

# Principal curves and manifolds

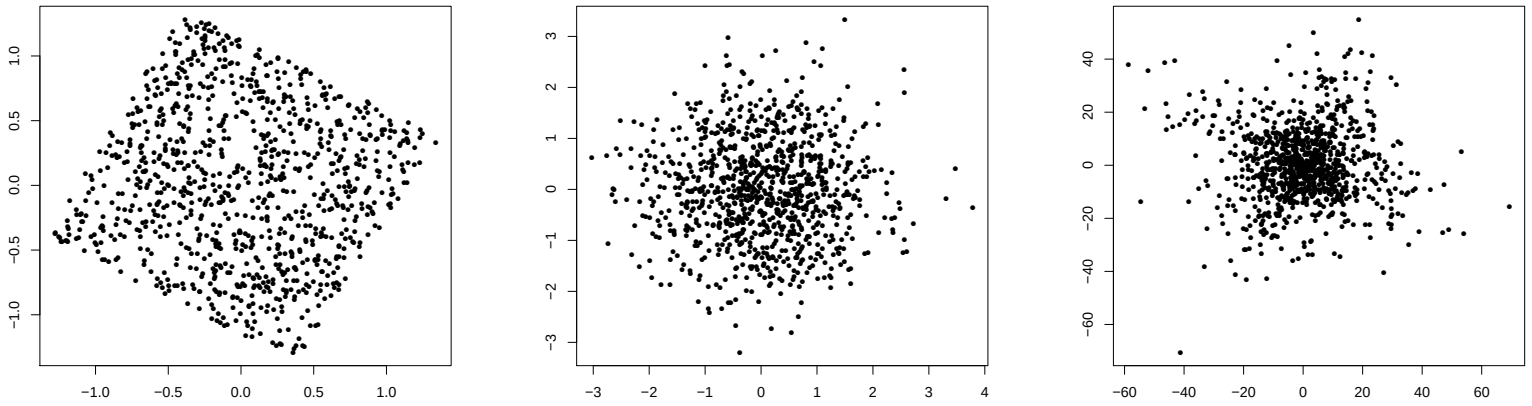



Reconstruction of the underlying curve is much more difficult.

- ▷ We must fix a curve parametrisation
- ▷ The task is different form regression since we have only outputs.

# Independent Component Analysis

Assume that the components of the source data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ are independent but an unknown affine transformation $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{\mu}$ disturbs observations.




It is possible to recover the translation and rotation only if independent components are sufficiently different form the normal distribution.