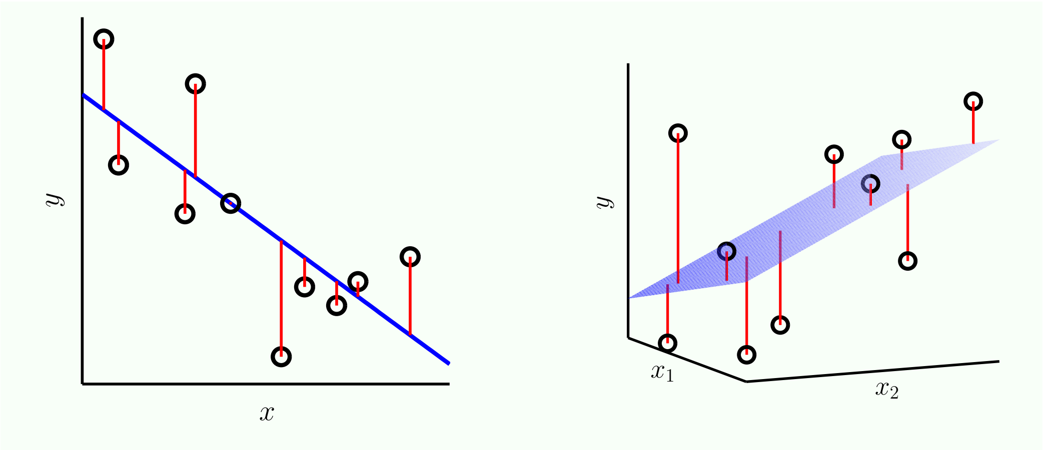
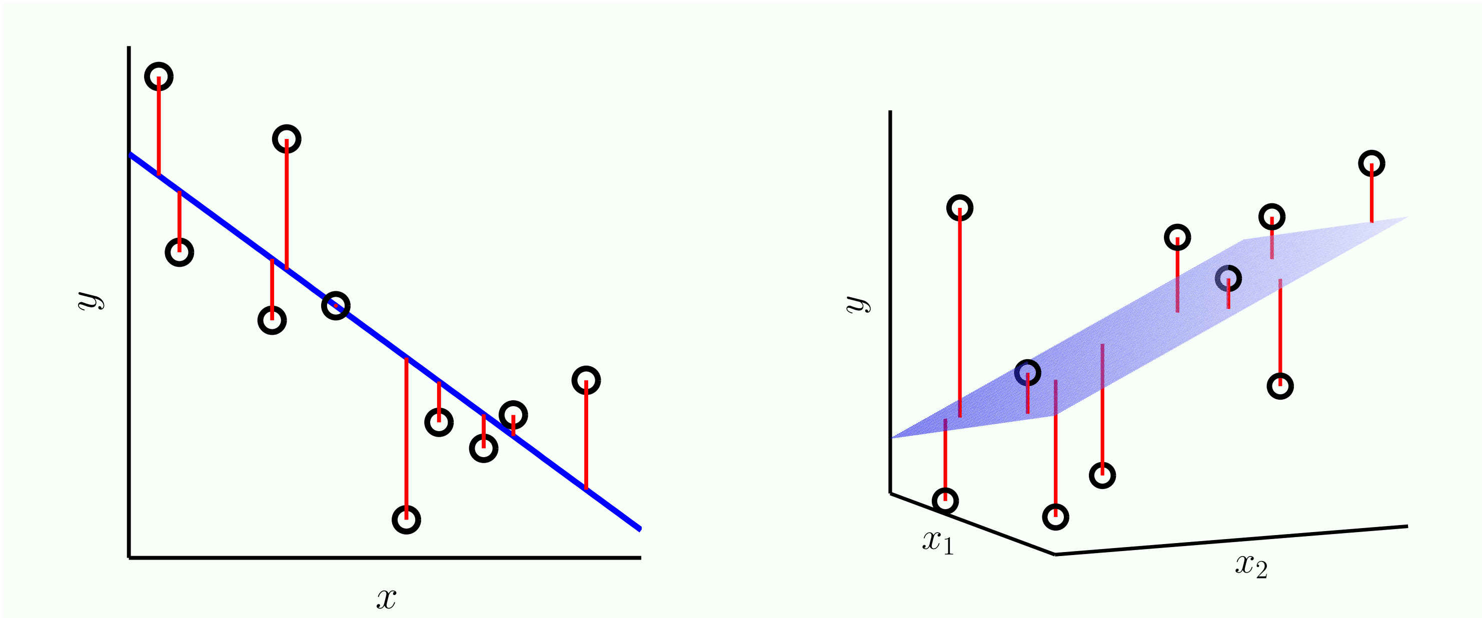
# Linear models: Linear regression

## Chapter 3.2



# Least squares linear regression




$$E_{\text{in}}(h) = \frac{1}{N}\sum_{i=1}^{N}(h(\mathbf{x}_i) - y_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\intercal \mathbf{x} - y_i)^2$$

Note: there is no explicit bias, so introduce it via an additional feature

# Least squares linear regression

Goal: the predicted values be as close as possible to the labels.

$$E_{\text{in}}(h) = \frac{1}{N}\sum_{i=1}^{N}(h(\mathbf{x}_i) - y_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\intercal\mathbf{x} - y_i)^2$$

Training: Find **w** that minimize this cost function

The discrepancy between predictions and labels is measured using a loss function. Here we used the squared-error loss:

$$L(y, \hat{y}) = (y - \hat{y})^2$$

# Expressing $E_{in}$ in matrix form

$$\mathrm{X} = \underbrace{\begin{bmatrix} \text{—}\mathbf{x}_1\text{—} \\ \text{—}\mathbf{x}_2\text{—} \\ \vdots \\ \text{—}\mathbf{x}_N\text{—} \end{bmatrix}}_{\text{data matrix, } N \times (d+1)} \qquad \mathbf{y} = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{\text{target vector}} \qquad \underbrace{\hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix} = \begin{bmatrix} \mathbf{w}^{\mathrm{T}}\mathbf{x}_1 \\ \mathbf{w}^{\mathrm{T}}\mathbf{x}_2 \\ \vdots \\ \mathbf{w}^{\mathrm{T}}\mathbf{x}_N \end{bmatrix} = \mathrm{X}\mathbf{w}}_{\text{in-sample predictions}}$$

$$\begin{aligned} E_{\text{in}}(\mathbf{w}) &= \frac{1}{N}\sum_{n=1}^{N}(\hat{y}_n - y_n)^2 \\ &= \tfrac{1}{N}\|\,\hat{\mathbf{y}} - \mathbf{y}\,\|_2^2 \\ &= \tfrac{1}{N}\|\,\mathrm{X}\mathbf{w} - \mathbf{y}\,\|_2^2 \\ &= \tfrac{1}{N}(\mathbf{w}^{\mathrm{T}}\mathrm{X}^{\mathrm{T}}\mathrm{X}\mathbf{w} - 2\mathbf{w}^{\mathrm{T}}\mathrm{X}^{\mathrm{T}}\mathbf{y} + \mathbf{y}^{\mathrm{T}}\mathbf{y}) \end{aligned}$$

# Gradients and vector differentiation

The gradient of a scalar function f(**w**) denoted by $\nabla f(\mathbf{w})$ is the vector $\left( \frac{\partial f(\mathbf{w})}{\partial w_1}, \ldots, \frac{\partial f(\mathbf{w})}{\partial w_d} \right)^{\mathsf{T}}$

We will also denote it as $\frac{\partial f(\mathbf{w})}{\partial \mathbf{w}}$

The requirement that the gradient be a column vector implies:

$$\frac{\partial}{\partial \mathbf{w}}(\mathbf{w}^{\mathsf{T}}\mathbf{x}) = \frac{\partial}{\partial \mathbf{w}}(\mathbf{x}^{\mathsf{T}}\mathbf{w}) = \mathbf{x}$$

Recall that a necessary (and not sufficient) condition for an extremum of a function f(**w**) is:

$$\frac{\partial f(\mathbf{w})}{\partial \mathbf{w}} = 0$$

# Solving for the weight vector

Let's do some algebra before taking the derivative:

$$\begin{aligned}(\mathbf{y} - \mathbf{X}\mathbf{w})^\intercal(\mathbf{y} - \mathbf{X}\mathbf{w}) &= (\mathbf{y}^\intercal - (\mathbf{X}\mathbf{w})^\intercal)(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= (\mathbf{y}^\intercal\mathbf{y} - \mathbf{w}^\intercal\mathbf{X}^\intercal)(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \mathbf{y}^\intercal\mathbf{y} - \mathbf{y}^\intercal\mathbf{X}\mathbf{w} - \mathbf{w}^\intercal\mathbf{X}^\intercal\mathbf{y} + \mathbf{w}^\intercal\mathbf{X}^\intercal\mathbf{X}\mathbf{w}\end{aligned}$$

$$\begin{aligned}\frac{\partial}{\partial \mathbf{w}}(\mathbf{y} - \mathbf{X}\mathbf{w})^\intercal(\mathbf{y} - \mathbf{X}\mathbf{w}) &= -(\mathbf{y}^\intercal\mathbf{X})^\intercal - \mathbf{X}^\intercal + \mathbf{X}^\intercal\mathbf{X}\mathbf{w} + (\mathbf{w}^\intercal\mathbf{X}^\intercal\mathbf{X})^\intercal \\ &= -2\mathbf{X}^\intercal\mathbf{y} + 2\mathbf{X}^\intercal\mathbf{X}\mathbf{w} = 0\end{aligned}$$

$$\frac{\partial}{\partial \mathbf{w}}(\mathbf{w}^\intercal\mathbf{x}) = \frac{\partial}{\partial \mathbf{w}}(\mathbf{x}^\intercal\mathbf{w}) = \mathbf{x}$$

Now we get that **w** satisfies: $\mathbf{X}^\intercal\mathbf{X}\mathbf{w} = \mathbf{X}^\intercal\mathbf{y}$

and $\mathbf{w} = (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{y}$

pseudo-inverse

# Probability theory digression

Random variable: the outcome of a random process

Examples: the possible outcomes of rolling a die: {1,2,3,4,5,6}

The expected value of a random variable X:

$$\mathbb{E}(X) = \sum_x x P(x)$$

(for a continuous variable replace sum with integral)

The empirical estimate for the expectation:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

# Probability theory digression

The spread of a distribution around the expected value is its variance, defined by:

$$\mathbb{E}\left[(X - \mathbb{E}(X))^2\right] = \mathbb{E}\left[X^2\right] - \mathbb{E}\left[X\right]^2$$

The sample variance is:

$$\sigma_X^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

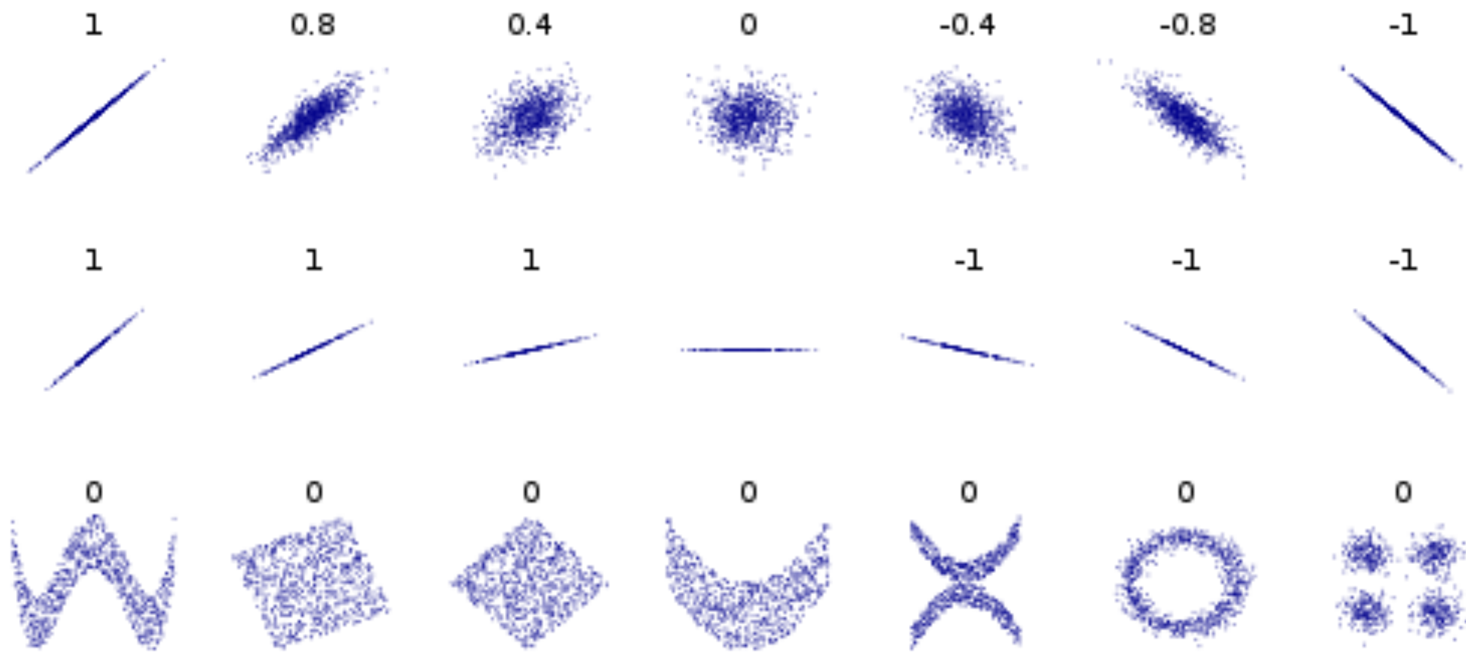
The covariance between two variables X and Y:

$$\mathbb{E}\left[(X - \mathbb{E}(X))\,(Y - \mathbb{E}(Y))\right] = \mathbb{E}\left[X \cdot Y\right] - \mathbb{E}\left[X\right]\mathbb{E}\left[Y\right]$$

The sample covariance:

$$\sigma_{XY}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = \frac{1}{n}\sum_{i=1}^{n} x_i y_i - \bar{x}\bar{y}$$

# Correlation between variables




The Pearson correlation between two variables is defined as:

$$r_{XY} = \frac{\sigma_{XY}}{\sigma_X \sigma_Y}$$

It varies between -1 and 1

Figure from http://en.wikipedia.org/wiki/Correlation_and_dependence

# More insight into the solution

Let's compare the general solution

$$\mathbf{w} = (\mathbf{X}^\intercal \mathbf{X})^{-1} \mathbf{X}^\intercal \mathbf{y}$$

With the solution to the one dimensional case (assume data is centered, i.e. has zero-mean):

$$w = \frac{\sigma_{XY}}{\sigma^2_{XX}}$$

Intuition: if X and Y are weakly correlated, the slope will be small

# More insight into the solution

Let's compare the general solution

$$\mathbf{w} = (\mathbf{X}^\intercal \mathbf{X})^{-1} \mathbf{X}^\intercal \mathbf{y}$$

With the solution to the one dimensional case:

$$w = \frac{\sigma_{XY}}{\sigma_{XX}^2}$$

And the special case of two dimensions:

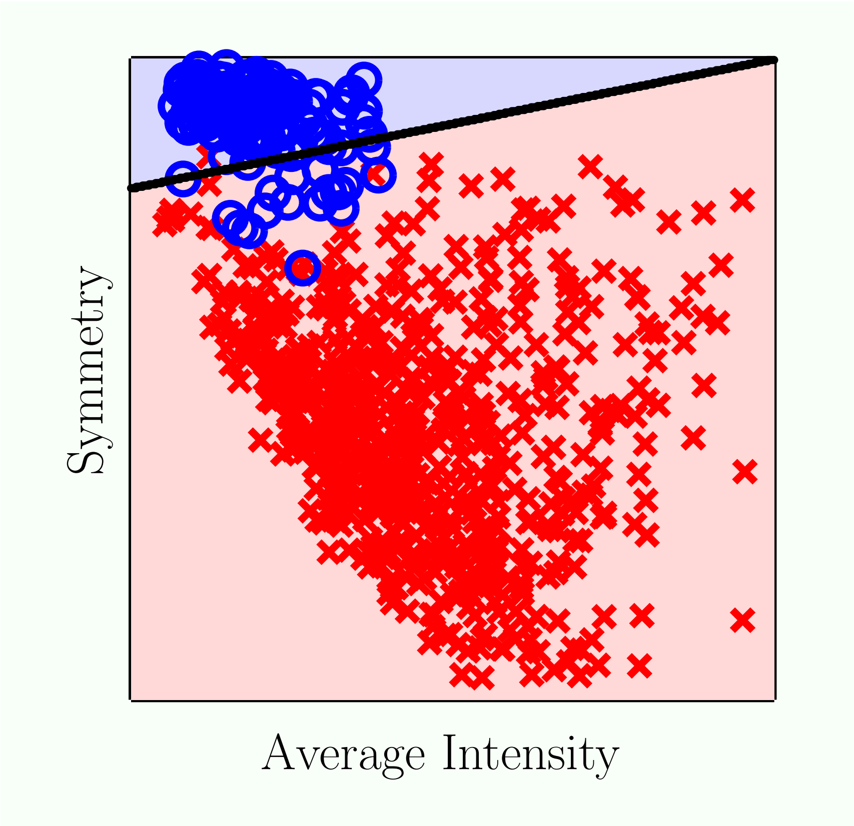
$$\mathbf{w} = \frac{1}{(\sigma_{11}\sigma_{22} - \sigma_{12}^2)} \begin{bmatrix} \sigma_{22}\sigma_{1y} - \sigma_{12}\sigma_{2y} \\ \sigma_{11}\sigma_{2y} - \sigma_{12}\sigma_{1y} \end{bmatrix}$$

What do we observe when $x_1$ and $x_2$ are uncorrelated?

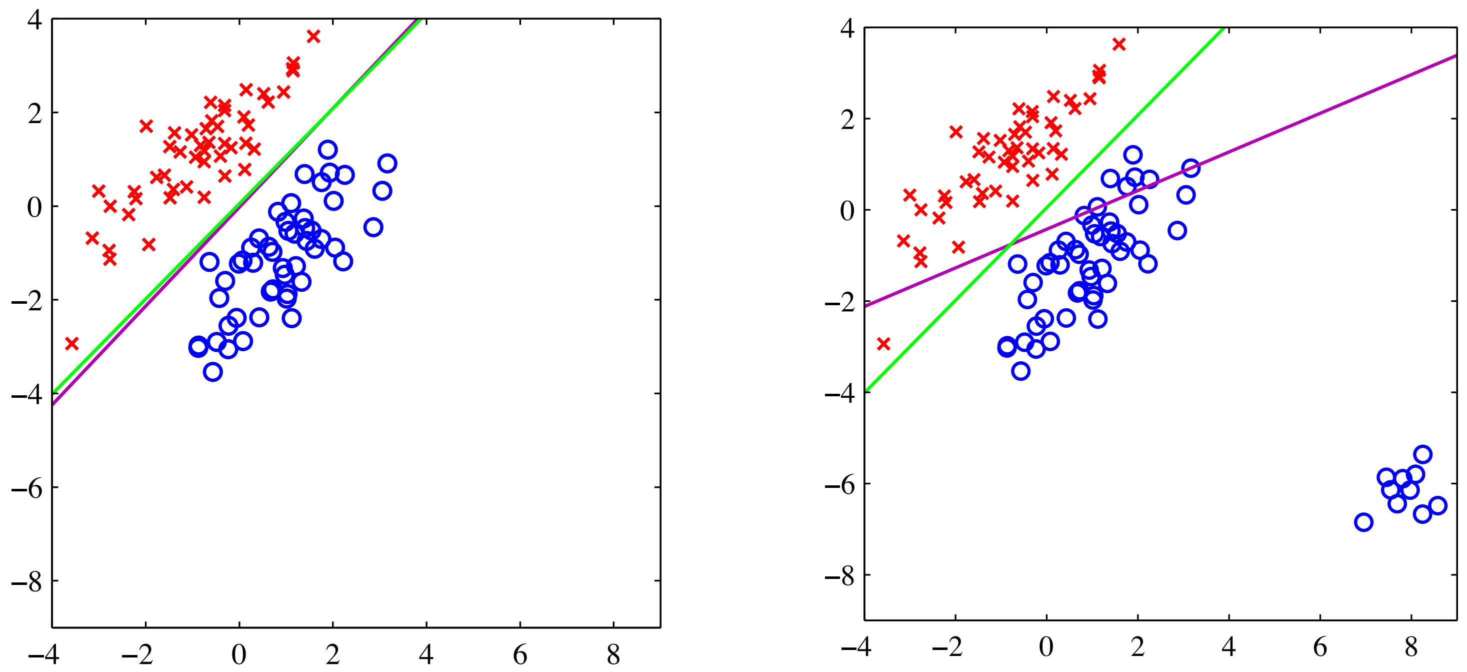
Also notice that $w_1$ may be nonzero even if $x_1$ is uncorrelated with the target variable.

# Linear regression for classification

You can use linear regression for binary classification problems.



# Sensitivity to outliers




Magenta: solution from least-squares
Green: logistic regression

# Do I have to invert that matrix?

In order to compute **w** you don't necessarily need to do it as:

$$\mathbf{w} = (\mathbf{X}^\intercal \mathbf{X})^{-1} \mathbf{X}^\intercal \mathbf{y}$$

Instead, you can solve for **w** as in:

$$\mathbf{X}^\intercal \mathbf{X} \mathbf{w} = \mathbf{X}^\intercal \mathbf{y}$$

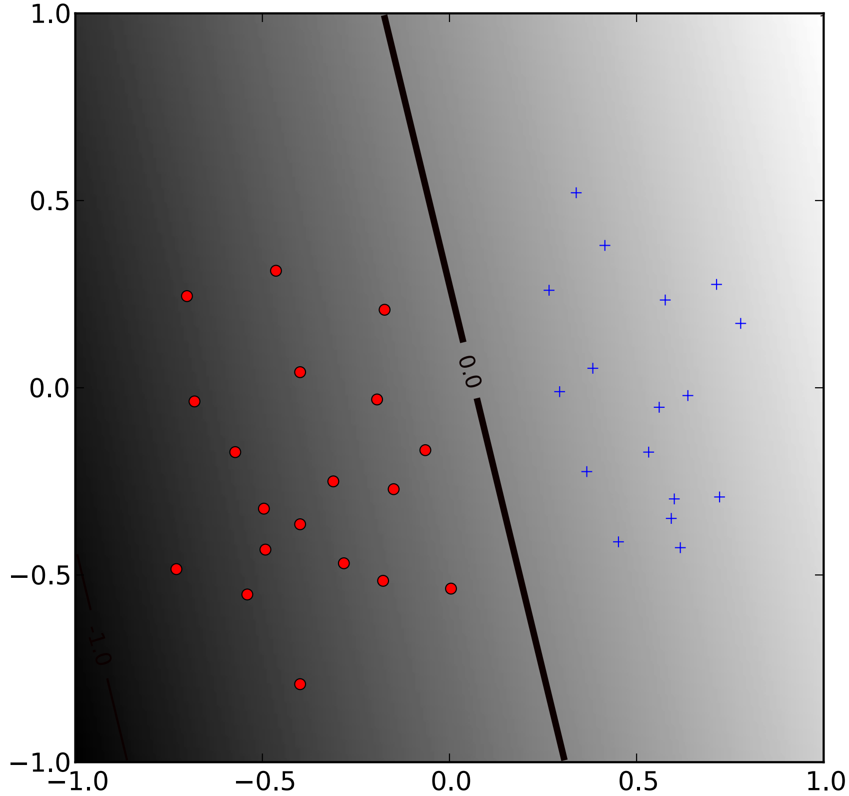
And, in python

```
import numpy as np
w = np.dot(np.linalg.inv(np.dot(X.T, X)), np.dot(X.T,y))
```

or, using the faster and more numerically stable solve function:

```
import numpy as np
w = np.linalg.solve(np.dot(X.T,X), np.dot(X.T, y))
```

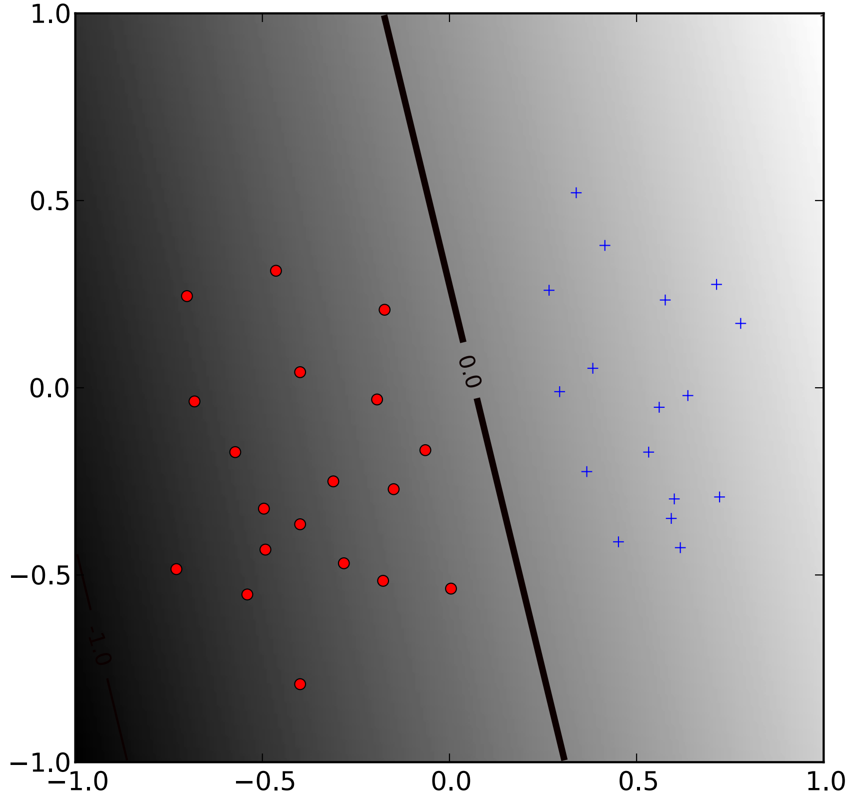
# Interpreting the weight vector




Which component of the weight vector is larger?

Which variable is more relevant for the classification task?

# Interpreting the weight vector




It is common practice to use the magnitude of weight vector components as an indicator of the importance of a feature.

Caveat: data needs to be normalized!

# Interpreting the weight vector

The weight vector for the "heart" dataset:

array([-0.07006162,  0.15838763,  0.28357296,  0.20753778,
0.23265869, -0.08271229,  0.08011837, -0.3363789 ,
0.11753745,  0.25560924, 0.09984765,  0.40073063,
0.23961789])

In the case of a binary classification problem, what is the relevance of the sign of $w_i$?

# Generalization

What can we say about $E_{out}$ having minimized $E_{in}$?

$$\mathbb{E}[E_{\text{out}}(h)] = \mathbb{E}[E_{\text{in}}(h)] + O\left(\frac{d}{N}\right)$$

See section 3.2.2 and exercise 3.4 for details.

# Measuring regression accuracy

Root Mean Square Error (RMSE):

$$\mathrm{RMSE}(h) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(h(\mathbf{x}_i) - y_i\right)^2}$$

Compute the RMSE on a test set

Another common measure of error is the Mean Absolute Deviation (MAD):

$$MAD(h) = \frac{1}{N}\sum_{i=1}^{N}|y_i - h(\mathbf{x}_i)|$$