Regression

Regression

General Idea of Regression

- Method of least squares

Linear Regression

- An illustrative example

Polynomial Regression

- Generalization to polynomial functional relationships

Multivariate Regression

- Generalization to more than one function argument

Logistic Regression

- Generalization to non-polynomial functional relationships
- An illustrative example

Summary

Regression

Also known as: **Method of Least Squares** (Carl Friedrich Gauß)

Given:
- A data set of data tuples (one or more input values and one output value).
- A hypothesis about the functional relationship between output and input values.

Desired:
- A parameterization of the conjectured function that minimizes the sum of squared errors ("best fit").

Depending on

the hypothesis about the functional relationship and

the number of arguments to the conjectured function

different types of regression are distinguished.

Reminder: Function Optimization

Task: Find values $\vec{x} = (x_1, \ldots, x_m)$ such that $f(\vec{x}) = f(x_1, \ldots, x_m)$ is optimal.

Often feasible approach:

A necessary condition for a (local) optimum (maximum or minimum) is that the partial derivatives w.r.t. the parameters vanish (Pierre Fermat).

Therefore: (Try to) solve the equation system that results from setting all partial derivatives w.r.t. the parameters equal to zero.

Example task: Minimize $\quad f(x, y) = x^2 + y^2 + xy - 4x - 5y.$

Solution procedure:

Take the partial derivatives of the objective function and set them to zero:

$$\frac{\partial f}{\partial x} = 2x + y - 4 = 0, \qquad \frac{\partial f}{\partial y} = 2y + x - 5 = 0.$$

Solve the resulting (here: linear) equation system: $\qquad x = 1, \quad y = 2.$

Linear Regression

Given: data set $((x_1, y_1), \ldots, (x_n, y_n))$ of n data tuples

Conjecture: the functional relationship is linear, i.e., $y = g(x) = a + bx$.

Approach: Minimize the sum of squared errors, i.e.

$$F(a, b) = \sum_{i=1}^{n} (g(x_i) - y_i)^2 = \sum_{i=1}^{n} (a + bx_i - y_i)^2.$$

Necessary conditions for a minimum:

$$\frac{\partial F}{\partial a} = \sum_{i=1}^{n} 2(a + bx_i - y_i) = 0 \qquad \text{and}$$

$$\frac{\partial F}{\partial b} = \sum_{i=1}^{n} 2(a + bx_i - y_i)x_i = 0$$

Linear Regression

Result of necessary conditions: System of so-called **normal equations**, i.e.

$$na + \left(\sum_{i=1}^{n} x_i\right) b = \sum_{i=1}^{n} y_i,$$

$$\left(\sum_{i=1}^{n} x_i\right) a + \left(\sum_{i=1}^{n} x_i^2\right) b = \sum_{i=1}^{n} x_i y_i.$$

Two linear equations for two unknowns a and b.

System can be solved with standard methods from linear algebra.

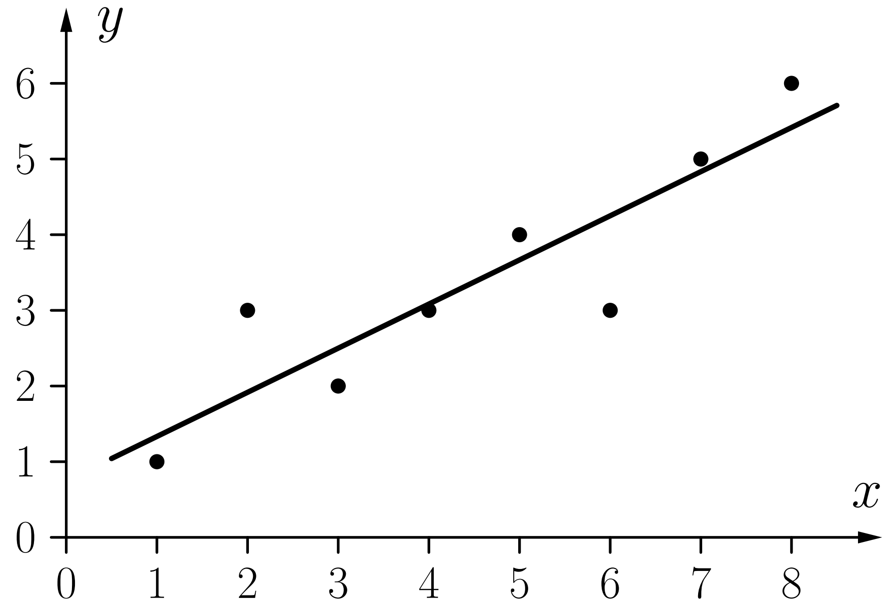
Solution is unique unless all x-values are identical.

The resulting line is called a **regression line**.

Linear Regression: Example

x	1	2	3	4	5	6	7	8
y	1	3	2	3	4	3	5	6

$$y = \frac{3}{4} + \frac{7}{12}x.$$



Least Squares and Maximum Likelihood

A regression line can be interpreted as a **maximum likelihood estimator:**

Assumption: The data generation process can be described well by the model

$$y = a + bx + \xi,$$

where ξ is normally distributed with mean 0 and (unknown) variance σ^2 (σ^2 independent of x, i.e. same dispersion of y for all x).

As a consequence we have

$$f(y \mid x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{(y-(a+bx))^2}{2\sigma^2}\right).$$

With this expression we can set up the **likelihood function**

$$\begin{aligned} & L((x_1, y_1), \ldots (x_n, y_n); a, b, \sigma^2) \\ & \quad = \prod_{i=1}^{n} f(x_i) f(y_i \mid x_i) \quad = \quad \prod_{i=1}^{n} f(x_i) \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{(y_i-(a+bx_i))^2}{2\sigma^2}\right). \end{aligned}$$

Least Squares and Maximum Likelihood

To simplify taking the derivatives, we compute the natural logarithm:

$$\begin{aligned}
&\ln L((x_1, y_1), \dots (x_n, y_n); a, b, \sigma^2) \\
&= \ln \prod_{i=1}^{n} f(x_i) \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{(y_i - (a + bx_i))^2}{2\sigma^2}\right) \\
&= \sum_{i=1}^{n} \ln f(x_i) + \sum_{i=1}^{n} \ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - (a + bx_i))^2
\end{aligned}$$

From this expression it becomes clear that (provided $f(x)$ is independent of a, b, and σ^2) maximizing the likelihood function is equivalent to minimizing

$$F(a, b) = \sum_{i=1}^{n} (y_i - (a + bx_i))^2.$$

Interpreting the method of least squares as a maximum likelihood estimator works also for the generalizations to polynomials and multilinear functions discussed next.

Generalization to polynomials

$$y = p(x) = a_0 + a_1 x + \ldots + a_m x^m$$

Approach: Minimize the sum of squared errors, i.e.

$$F(a_0, a_1, \ldots, a_m) \;=\; \sum_{i=1}^{n} (p(x_i) - y_i)^2 \;=\; \sum_{i=1}^{n} (a_0 + a_1 x_i + \ldots + a_m x_i^m - y_i)^2$$

Necessary conditions for a minimum: All partial derivatives vanish, i.e.

$$\frac{\partial F}{\partial a_0} = 0, \quad \frac{\partial F}{\partial a_1} = 0, \quad \ldots, \quad \frac{\partial F}{\partial a_m} = 0.$$

System of normal equations for polynomials

$$
\begin{aligned}
na_0 + \left(\sum_{i=1}^n x_i\right) a_1 + \ldots + \left(\sum_{i=1}^n x_i^m\right) a_m &= \sum_{i=1}^n y_i \\
\left(\sum_{i=1}^n x_i\right) a_0 + \left(\sum_{i=1}^n x_i^2\right) a_1 + \ldots + \left(\sum_{i=1}^n x_i^{m+1}\right) a_m &= \sum_{i=1}^n x_i y_i \\
&\vdots \\
\left(\sum_{i=1}^n x_i^m\right) a_0 + \left(\sum_{i=1}^n x_i^{m+1}\right) a_1 + \ldots + \left(\sum_{i=1}^n x_i^{2m}\right) a_m &= \sum_{i=1}^n x_i^m y_i,
\end{aligned}
$$

$m+1$ linear equations for $m+1$ unknowns $a_0, \ldots, a_m$.

System can be solved with standard methods from linear algebra.

Solution is unique unless the points lie exactly on a polynomial of lower degree.

Generalization to more than one argument

$$z = f(x, y) = a + bx + cy$$

Approach: Minimize the sum of squared errors, i.e.

$$F(a, b, c) \;=\; \sum_{i=1}^{n} (f(x_i, y_i) - z_i)^2 \;=\; \sum_{i=1}^{n} (a + bx_i + cy_i - z_i)^2$$

Necessary conditions for a minimum: All partial derivatives vanish, i.e.

$$\begin{aligned}
\frac{\partial F}{\partial a} &= \sum_{i=1}^{n} 2(a + bx_i + cy_i - z_i) &= 0, \\
\frac{\partial F}{\partial b} &= \sum_{i=1}^{n} 2(a + bx_i + cy_i - z_i)x_i &= 0, \\
\frac{\partial F}{\partial c} &= \sum_{i=1}^{n} 2(a + bx_i + cy_i - z_i)y_i &= 0.
\end{aligned}$$

Multilinear Regression

System of normal equations for several arguments

$$
\begin{aligned}
na + \left(\sum_{i=1}^{n} x_i\right) b + \left(\sum_{i=1}^{n} y_i\right) c &= \sum_{i=1}^{n} z_i \\
\left(\sum_{i=1}^{n} x_i\right) a + \left(\sum_{i=1}^{n} x_i^2\right) b + \left(\sum_{i=1}^{n} x_i y_i\right) c &= \sum_{i=1}^{n} z_i x_i \\
\left(\sum_{i=1}^{n} y_i\right) a + \left(\sum_{i=1}^{n} x_i y_i\right) b + \left(\sum_{i=1}^{n} y_i^2\right) c &= \sum_{i=1}^{n} z_i y_i
\end{aligned}
$$

3 linear equations for 3 unknowns a, b, and c.

System can be solved with standard methods from linear algebra.

Solution is unique unless all data points lie on a straight line.

Multilinear Regression

General multilinear case:

$$\vec{y} = f(\vec{x}_1, \ldots, \vec{x}_m) = a_0 + \sum_{k=1}^{m} a_k \vec{x}_k$$

Approach: Minimize the sum of squared errors, i.e.

$$F(\vec{a}) = (\mathbf{X}\vec{a} - \vec{y})^\top (\mathbf{X}\vec{a} - \vec{y}),$$

where

$$\mathbf{X} = \begin{pmatrix} 1 & x_{11} & \ldots & x_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \ldots & x_{nm} \end{pmatrix}, \qquad \vec{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \qquad \text{and} \qquad \vec{a} = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{pmatrix}$$

Necessary condition for a minimum:

$$\nabla_{\vec{a}} F(\vec{a}) = \nabla_{\vec{a}} (\mathbf{X}\vec{a} - \vec{y})^\top (\mathbf{X}\vec{a} - \vec{y}) = \vec{0}$$

Multilinear Regression

$\nabla_{\vec{a}} \, F(\vec{a})$ may easily be computed by remembering that the differential operator

$$\nabla_{\vec{a}} = \left(\frac{\partial}{\partial a_0}, \ldots, \frac{\partial}{\partial a_m} \right)$$

behaves formally like a vector that is "multiplied" to the sum of squared errors.

Alternatively, one may write out the differentiation componentwise.

Reminder: Vector Derivatives

What is the derivative of $\vec{x}^\top \vec{x}$ w. r. t. $\vec{x}$?

$$\nabla_{\vec{x}} \vec{x}^\top \vec{x} = \left(\frac{\partial \vec{x}^\top \vec{x}}{\partial x_1}, \cdots, \frac{\partial \vec{x}^\top \vec{x}}{\partial x_m} \right)$$

We get: $k = 1, \ldots, m$

$$\begin{aligned}
\frac{\partial \vec{x}^\top \vec{x}}{\partial x_k} &= \frac{\partial}{\partial x_k} \sum_{i=1}^{m} x_i x_i \\
&= \frac{\partial}{\partial x_k} \left(x_1^2 + \cdots + x_k^2 + \cdots + x_m^2 \right) \\
&= \frac{\partial}{\partial x_k} x_1^2 + \cdots + \frac{\partial}{\partial x_k} x_k^2 + \cdots + \frac{\partial}{\partial x_k} x_m^2 \\
&= 2x_k
\end{aligned}$$

Therefore we get:

$$\nabla_{\vec{x}} \vec{x}^\top \vec{x} = (2x_1, \ldots, 2x_k, \ldots, 2x_m) = 2\vec{x}$$

Multilinear Regression

With the former method we obtain for the derivative:

$$\begin{aligned}
&\nabla_{\vec{a}} \, (\mathbf{X}\vec{a} - \vec{y})^\top (\mathbf{X}\vec{a} - \vec{y}) \\
&\quad = \quad (\nabla_{\vec{a}} \, (\mathbf{X}\vec{a} - \vec{y}))^\top \, (\mathbf{X}\vec{a} - \vec{y}) + ((\mathbf{X}\vec{a} - \vec{y})^\top \, (\nabla_{\vec{a}} \, (\mathbf{X}\vec{a} - \vec{y})))^\top \\
&\quad = \quad (\nabla_{\vec{a}} \, (\mathbf{X}\vec{a} - \vec{y}))^\top \, (\mathbf{X}\vec{a} - \vec{y}) + (\nabla_{\vec{a}} \, (\mathbf{X}\vec{a} - \vec{y}))^\top \, (\mathbf{X}\vec{a} - \vec{y}) \\
&\quad = \quad 2\mathbf{X}^\top (\mathbf{X}\vec{a} - \vec{y}) \\
&\quad = \quad 2\mathbf{X}^\top \mathbf{X}\vec{a} - 2\mathbf{X}^\top \vec{y} \quad = \quad \vec{0}
\end{aligned}$$

Multilinear Regression

Necessary condition for a minimum therefore:

$$\begin{aligned} \nabla_{\vec{a}} F(\vec{a}) &= \nabla_{\vec{a}} (\mathbf{X}\vec{a} - \vec{y})^\top (\mathbf{X}\vec{a} - \vec{y}) \\ &= 2\mathbf{X}^\top \mathbf{X}\vec{a} - 2\mathbf{X}^\top \vec{y} \overset{!}{=} \vec{0} \end{aligned}$$

As a consequence we get the system of **normal equations**:

$$\mathbf{X}^\top \mathbf{X}\vec{a} = \mathbf{X}^\top \vec{y}$$

This system has a unique solution if $\mathbf{X}^\top \mathbf{X}$ is not singular. Then we have

$$\vec{a} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \vec{y}.$$

$(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top$ is called the (Moore–Penrose) **pseudoinverse** of the matrix $\mathbf{X}$.

With the matrix-vector representation of the regression problem an extension to **multipolynomial regression** is straighforward:
Simply add the desired products of powers to the matrix $\mathbf{X}$.

Generalization to non-polynomial functions

Idea: Find transformation to linear/polynomial case.

Simple example: The function $y = ax^b$

can be transformed into $\ln y = \ln a + b \cdot \ln x$.

Special case: **logistic function**

$$y = \frac{Y}{1 + e^{a+bx}} \quad \Leftrightarrow \quad \frac{1}{y} = \frac{1 + e^{a+bx}}{Y} \quad \Leftrightarrow \quad \frac{Y - y}{y} = e^{a+bx}.$$

Result: Apply so-called **Logit Transformation**

$$\ln\left(\frac{Y - y}{y}\right) = a + bx.$$

Logistic Regression: Example

x	1	2	3	4	5
y	0.4	1.0	3.0	5.0	5.6

Transform the data with

$$z = \ln\left(\frac{Y - y}{y}\right), \qquad Y = 6.$$

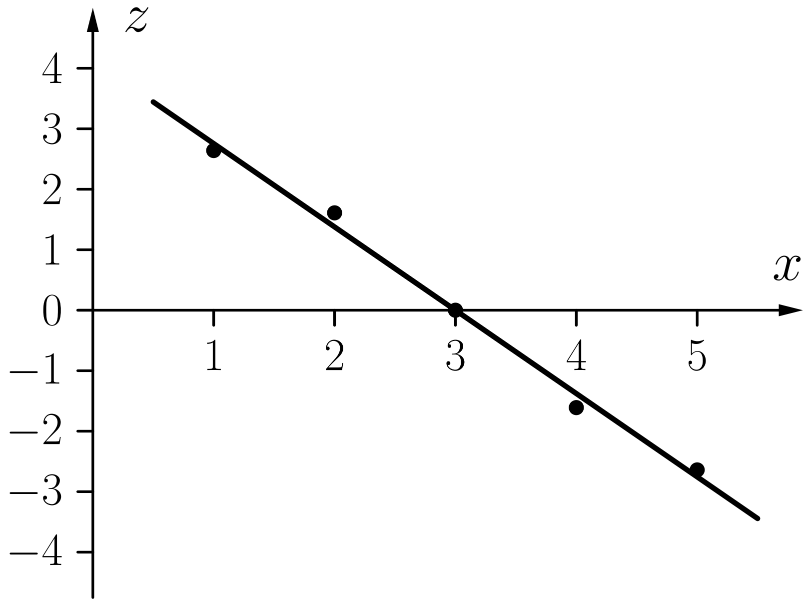
The transformed data points are

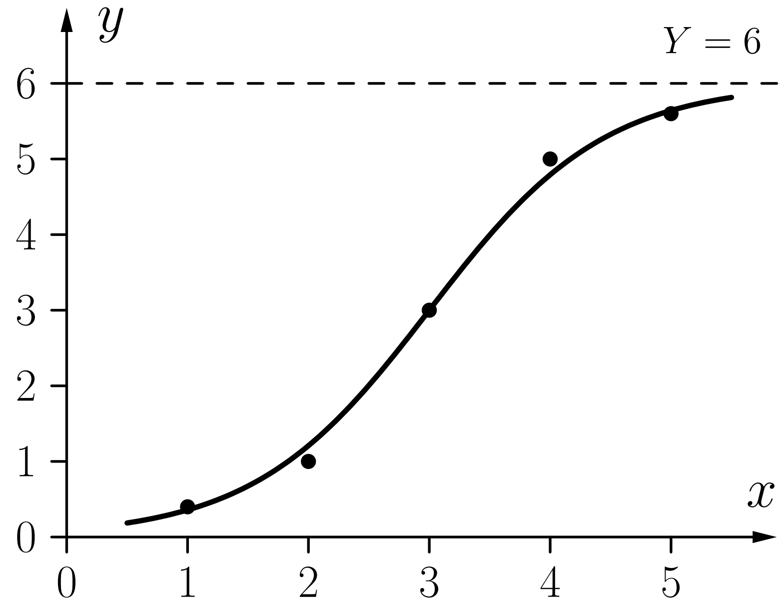
x	1	2	3	4	5
z	2.64	1.61	0.00	-1.61	-2.64

The resulting regression line is

$$z \approx -1.3775x + 4.133.$$

Logistic Regression: Example





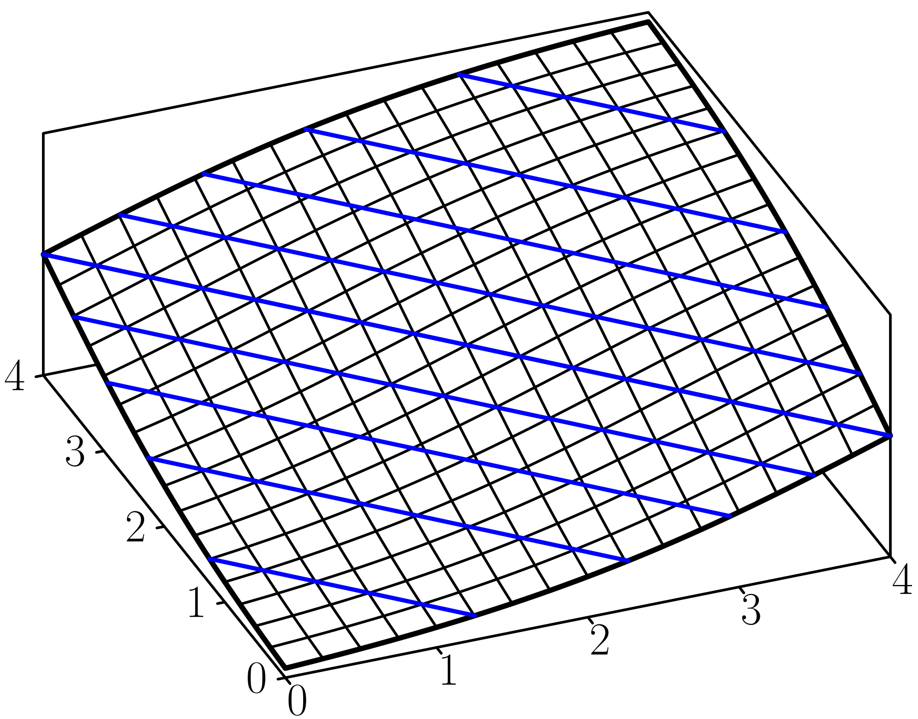
Attention: The sum of squared errors is minimized only in the space the transformation maps to, not in the original space.

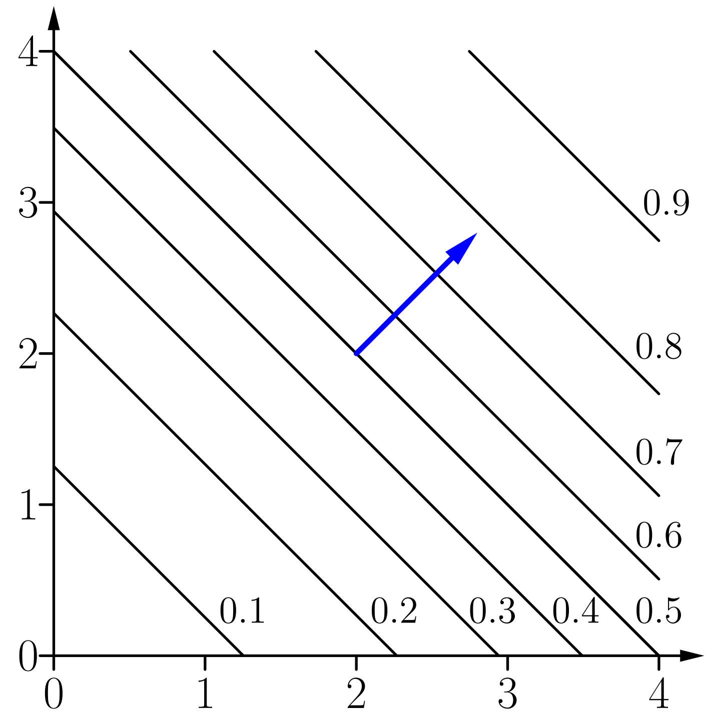
Nevertheless this approach usually leads to very good results.
The result may be improved by a gradient descent in the original space.

Logistic Regression: Two-dimensional Example

Example logistic function for two arguments x_1 and x_2:

$$y = \frac{1}{1 + \exp(4 - x_1 - x_2)} = \frac{1}{1 + \exp\left(4 - (1,1)(x_1, x_2)^\top\right)}$$





Logistic Regression: Two Class Problems

Let C be a class attribute, $\text{dom}(C) = \{c_1, c_2\}$, and $\vec{X}$ an m-dim. random vector.

Let $P(C = c_1 \mid \vec{X} = \vec{x}) = p(\vec{x})$ and $P(C = c_2 \mid \vec{X} = \vec{x}) = 1 - p(\vec{x})$.

Given: A set of data points $\mathbf{X} = \{\vec{x}_1, \ldots, \vec{x}_n\}$ (realizations of $\vec{X}$), each of which belongs to one of the two classes c_1 and c_2.

Desired: A simple description of the function $p(\vec{x})$.

Approach: Describe p by a logistic function:

$$p(\vec{x}) = \frac{1}{1 + e^{a_0 + \vec{a}\vec{x}}} = \frac{1}{1 + \exp\left(a_0 + \sum_{i=1}^{m} a_i x_i\right)}$$

Apply logit transformation to $p(x)$:

$$\ln\left(\frac{1 - p(\vec{x})}{p(\vec{x})}\right) = a_0 + \vec{a}\vec{x} = a_0 + \sum_{i=1}^{m} a_i x_i$$

The values $p(\vec{x}_i)$ may be obtained by kernel estimation.

Kernel Estimation

Idea: Define an "influence function" (kernel), which describes how strongly a data point influences the probability estimate for neighboring points.

Common choice for the kernel function: **Gaussian function**

$$K(\vec{x}, \vec{y}) \quad = \quad \frac{1}{(2\pi\sigma^2)^{\frac{m}{2}}} \exp\left(-\frac{(\vec{x}-\vec{y})^\top(\vec{x}-\vec{y})}{2\sigma^2}\right)$$

Kernel estimate of probability density given a data set $\mathcal{X} = \{\vec{x}_1, \ldots, \vec{x}_n\}$:

$$\hat{f}(\vec{x}) \quad = \quad \frac{1}{n}\sum_{i=1}^{n} K(\vec{x}, \vec{x}_i).$$

Kernel estimation applied to a two class problem:

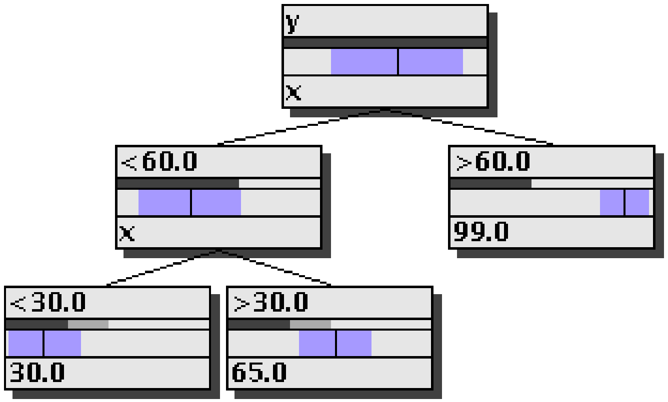
$$\hat{p}(\vec{x}) \quad = \quad \frac{\sum_{i=1}^{n} c(\vec{x}_i) K(\vec{x}, \vec{x}_i)}{\sum_{i=1}^{n} K(\vec{x}, \vec{x}_i)}.$$

(It is $c(\vec{x}_i) = 1$ if x_i belongs to class c_1 and $c(\vec{x}_i) = 0$ otherwise.)

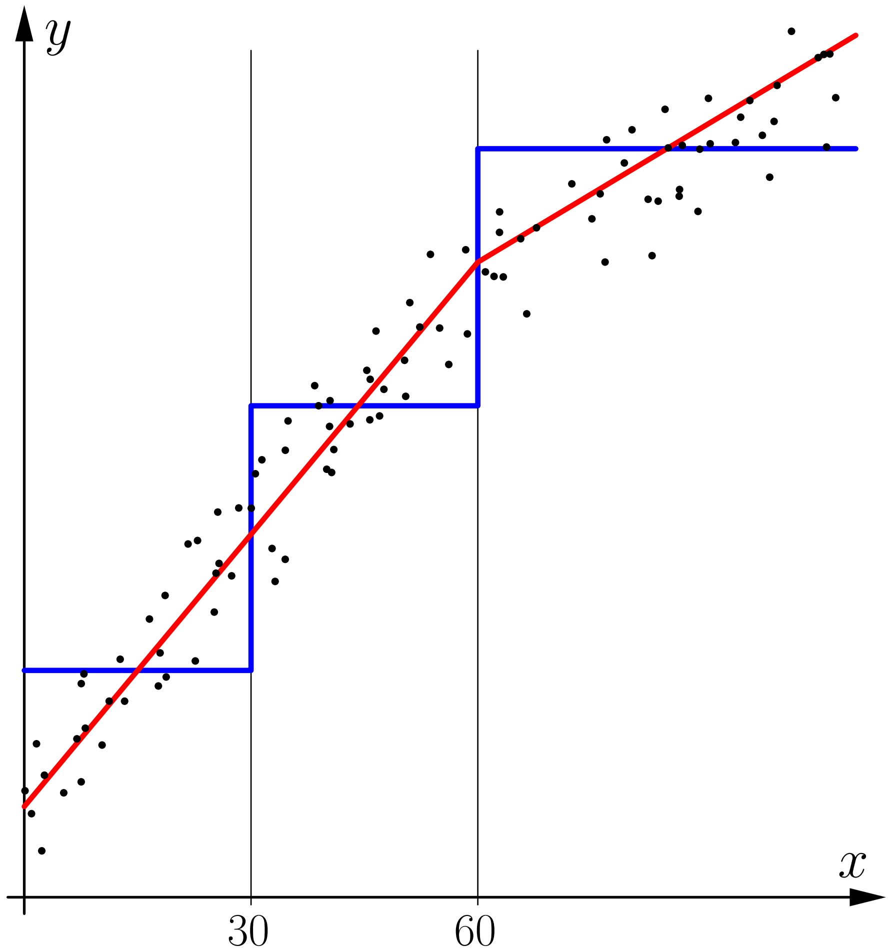
Regression Trees

Target variable is not a class,
but a numeric quantity.

Simple regression trees:
predict constant values in leaves.
(blue lines)

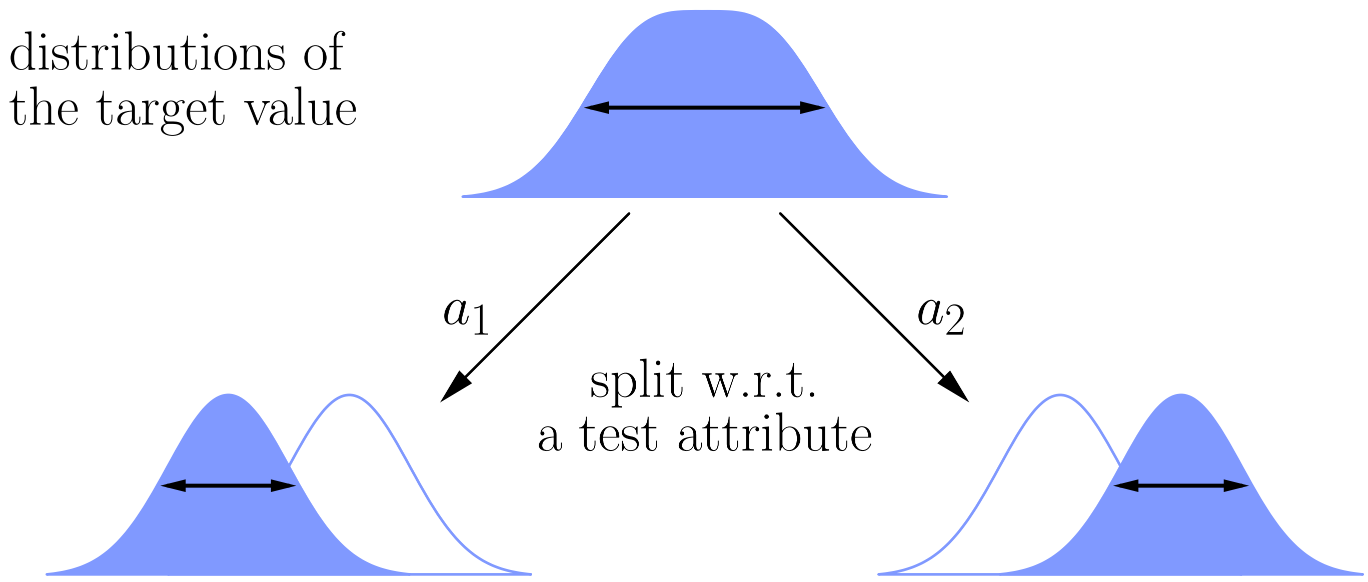


More complex regression trees:
predict linear functions in leaves.
(red line)



x: input variable, y: target variable

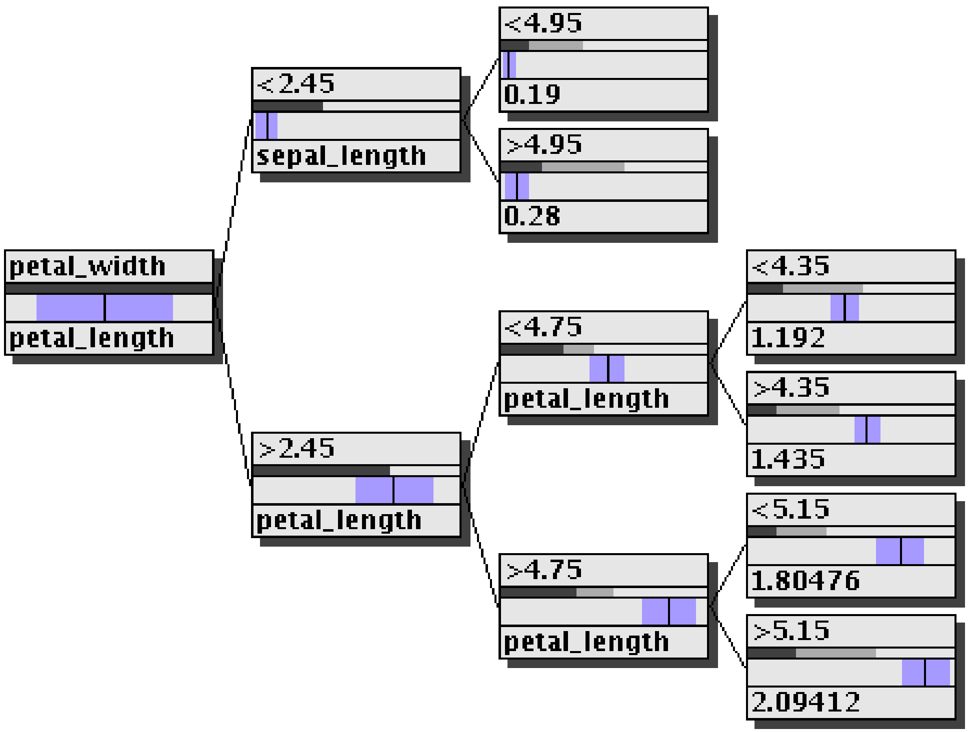
Regression Trees: Attribute Selection

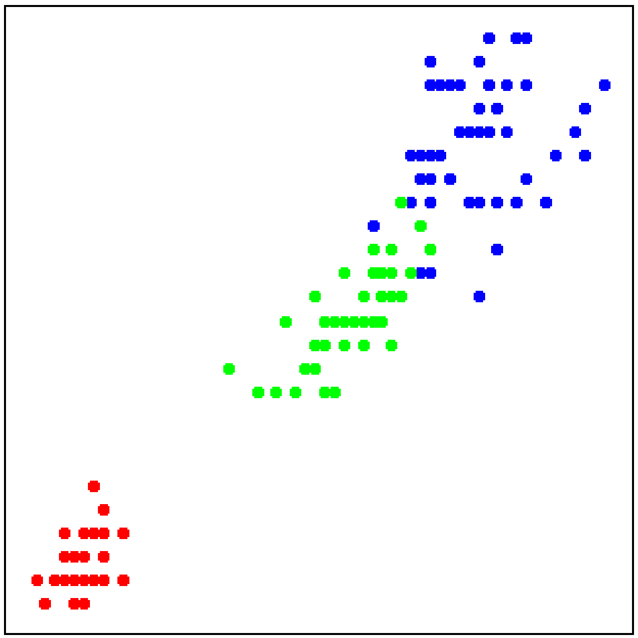


The variance / standard deviation is compared to the variance / standard deviation in the branches.

The attribute that yields the highest reduction is selected.

A regression tree for the Iris data (petal width)
(induced with reduction of sum of squared errors)





Summary Regression

Minimize the Sum of Squared Errors

- Write the sum of squared errors
  as a function of the parameters to be determined.

Exploit Necessary Conditions for a Minimum

- Partial derivatives w.r.t. the parameters to determine must vanish.

Solve the System of Normal Equations

- The best fit parameters are the solution of the system of normal equations.

Non-polynomial Regression Functions

- Find a transformation to the multipolynomial case.
- Logistic regression can be used to solve two class classification problems.