Chapter ML:II

II. Machine Learning Basics

- Regression
- Concept Learning: Search in Hypothesis Space
- Concept Learning: Search in Version Space
- Measuring Performance

Regression

Classification versus Regression

X is a p-dimensional feature space or input space. Example:

Customer 1	
house owner	yes
income (p.a.)	51 000 EUR
repayment (p.m.)	1 000 EUR
credit period	7 years
SCHUFA entry	no
age	37
married	yes
...	

...

Customer n	
house owner	no
income (p.a.)	55 000 EUR
repayment (p.m.)	1 200 EUR
credit period	8 years
SCHUFA entry	no
age	?
married	yes
...	

Classification:

- $C = \{-1, 1\}$ is a set of classes. Similarly: $C = \{0, 1\}$, $C = \{\text{no}, \text{yes}\}$
- $D = \{(\mathbf{x}_1, c(\mathbf{x}_1)), \ldots, (\mathbf{x}_n, c(\mathbf{x}_n))\} \subseteq X \times C$ is a set of examples.
- $c(\mathbf{x}_i)$ is the ground truth for the creditworthiness class, $\mathbf{x}_i \in X$.

Regression

Classification versus Regression

X is a p-dimensional feature space or input space. Example:

Customer 1	
house owner	yes
income (p.a.)	51 000 EUR
repayment (p.m.)	1 000 EUR
credit period	7 years
SCHUFA entry	no
age	37
married	yes
...	

...

Customer n	
house owner	no
income (p.a.)	55 000 EUR
repayment (p.m.)	1 200 EUR
credit period	8 years
SCHUFA entry	no
age	?
married	yes
...	

Classification:

- $C = \{-1, 1\}$ is a set of classes. Similarly: $C = \{0, 1\}$, $C = \{\text{no}, \text{yes}\}$
- $D = \{(\mathbf{x}_1, c(\mathbf{x}_1)), \ldots, (\mathbf{x}_n, c(\mathbf{x}_n))\} \subseteq X \times C$ is a set of examples.
- $c(\mathbf{x}_i)$ is the ground truth for the creditworthiness class, $\mathbf{x}_i \in X$.

Regression:

- $Y \subseteq \mathbf{R}$ is the output space.
- $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \subseteq X \times Y$ is a set of examples.
- y_i is the ground truth for the credit line value, $\mathbf{x}_i \in X$.

Regression

The Linear Regression Model

- Given $\mathbf{x}$, predict a real-valued output under a linear model function:

$$y(\mathbf{x}) = w_0 + \sum_{j=1}^{p} w_j \cdot x_j$$

- Vector notation with $x_0 = 1$ and $\mathbf{w} = (w_0, w_1, \ldots, w_p)^T$:

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$$

- Given $\mathbf{x}_1, \ldots, \mathbf{x}_n$, assess goodness of fit as residual sum of squares:

$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^{n} (y_i - y(\mathbf{x}_i))^2 = \sum_{i=1}^{n} (y_i - \mathbf{w}^T\mathbf{x}_i)^2 \qquad (1)$$

Regression

The Linear Regression Model

- Given $\mathbf{x}$, predict a real-valued output under a linear model function:

$$y(\mathbf{x}) = w_0 + \sum_{j=1}^{p} w_j \cdot x_j$$

- Vector notation with $x_0 = 1$ and $\mathbf{w} = (w_0, w_1, \ldots, w_p)^T$:

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$$

- Given $\mathbf{x}_1, \ldots, \mathbf{x}_n$, assess goodness of fit as residual sum of squares:

$$\text{RSS}(\mathbf{w}) = \sum_{i=1}^{n} (y_i - y(\mathbf{x}_i))^2 = \sum_{i=1}^{n} (y_i - \mathbf{w}^T\mathbf{x}_i)^2 \qquad (1)$$

- Estimate $\mathbf{w}$ by minimizing the residual sum of squares:

$$\hat{\mathbf{w}} = \underset{\mathbf{w} \in \mathbf{R}^{p+1}}{\text{argmin}}\ \text{RSS}(\mathbf{w}) \qquad (2)$$

Remarks:

- A *residual* is the difference between an observed value y_i and the estimated value $y(\mathbf{x}_i)$ of the model function.

- The residual sum of squares, RSS, is the sum of squares of the residuals. It is also known as the sum of squared residuals, SSR, or the sum of squared errors of prediction, SSE.

- The RSS term quantifies the regression error—or similarly, the goodness of fit—in the form of a single value.

- RSS provides several numerical and theoretical advantages, but it is not the only possibility to assess the goodness of fit (= error) between observed values and the model function. Alternative approaches for quantifying the error include absolute residual values or a polynomial in the residual values.

- The error computation is also called loss computation, cost computation, or generally, performance computation. Similarly, for the right-hand side of Equation (1) the following names are used: error function, loss function, cost function, or generally, performance term. Measures that quantify this kind of performance are called *effectiveness* measures. This must not be confused with *efficiency* measures, which quantify the computational effort or runtime performance of a method.

Remarks (continued) :

- From a statistical viewpoint, $\mathbf{x} = x_1, \ldots, x_p$ and y represent *random variables* (vectorial and scalar respectively). Each feature vector, $\mathbf{x}_i$, and outcome, y_i, is the result of a random experiment and hence is governed by a—usually unknown—probability distribution.

- The distributions of the random variables y_i and $(y_i - y(\mathbf{x}_i))$ are identical.

- Equation (2): Estimating $\mathbf{w}$ by RSS minimization is based on the following assumptions:

  1. The random variables y_i are statistically independent. Actually, the conditional independence of the y_i under $\mathbf{x}_i$ is sufficient.

  2. The means $E(y_i)$ lie on a straight line, known as the true (population) regression line: $E(y_i) = \mathbf{w}^{*T}\mathbf{x}_i$. I.e., the relation between the observed $(\mathbf{x}, y) \in X \times Y$ can be completely explained by a linear model function.

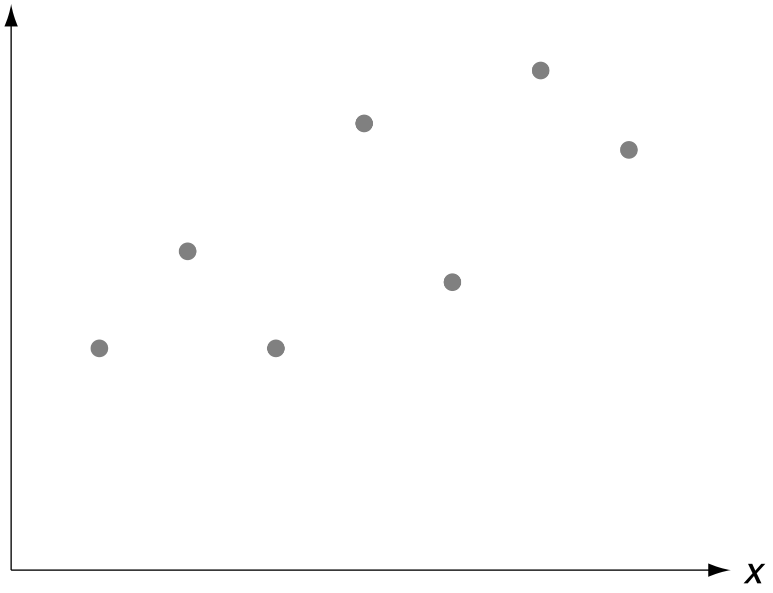
  3. The probability distributions $P(y_i \mid \mathbf{x}_i)$ have the same variance.

  The three assumptions are called the *weak set* (of assumptions). Along with a fourth assumption about the distribution shape of y_i they become the *strong set* of assumptions.

- In order to avoid cluttered notation, we won't use different symbols to distinguish random variables from ordinary variables. I.e., if $\mathbf{x}, x, y$ denote a (vectorial or scalar) random variable this fact will become clear from the context.

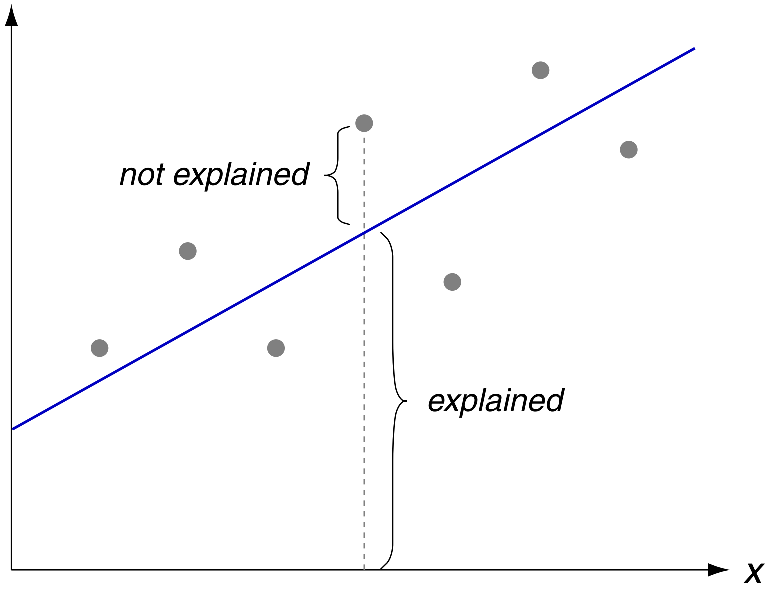
Regression

One-Dimensional Feature Space



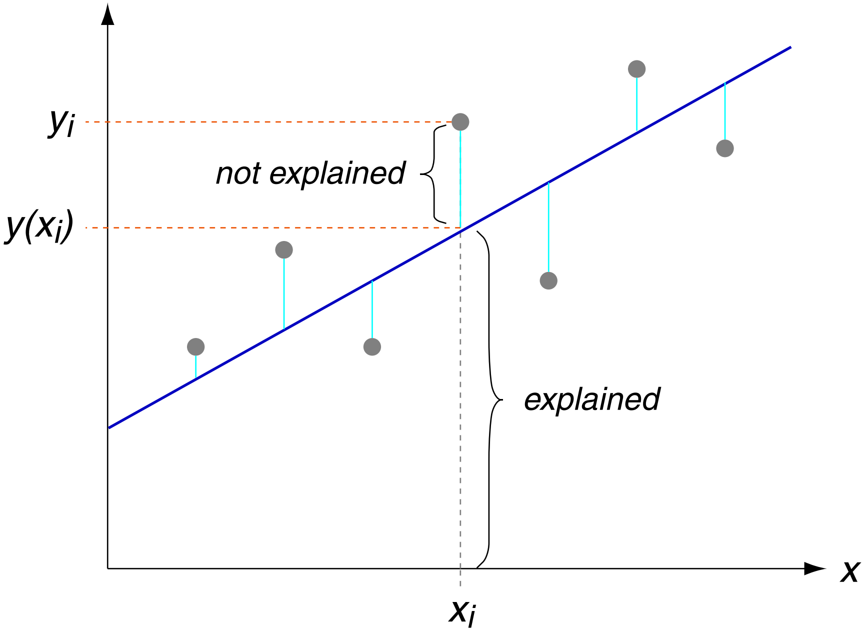
Regression

One-Dimensional Feature Space (continued)



Regression

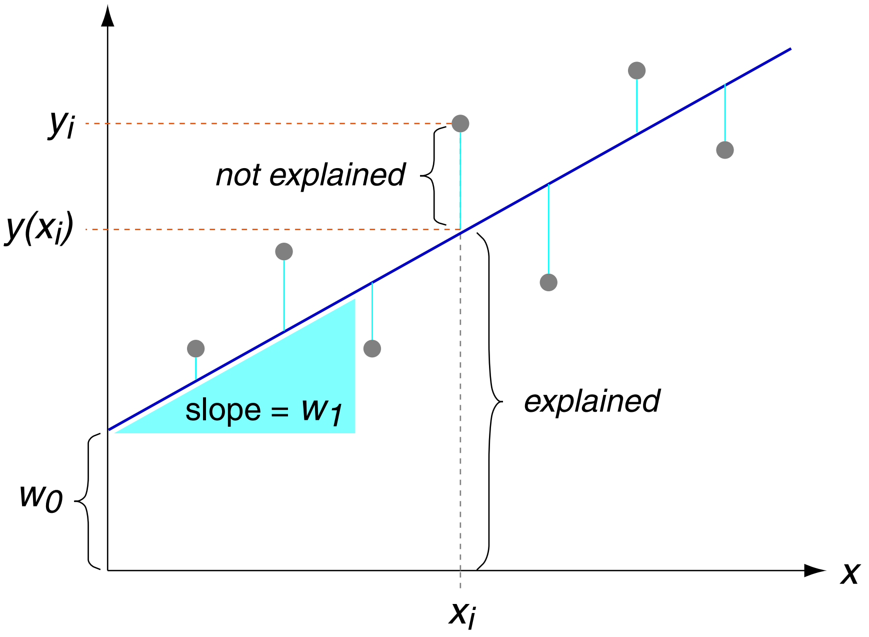
One-Dimensional Feature Space (continued)



$$\text{RSS} = \sum_{i=1}^{n} (y_i - y(x_i))^2$$

Regression

One-Dimensional Feature Space (continued)



$$y(x) = w_0 + w_1 \cdot x, \qquad \text{RSS}(w_0, w_1) = \sum_{i=1}^{n} (y_i - w_0 - w_1 \cdot x_i)^2$$

Regression

One-Dimensional Feature Space (continued) [higher-dimensional]

Minimize RSS(w_0, w_1) by a direct method:

1. $$\frac{\partial}{\partial w_0} \sum_{i=1}^{n} (y_i - w_0 - w_1 \cdot x_i)^2 \; = \; 0$$

$$\leadsto \; \ldots \; \leadsto \quad w_0 \; = \; \frac{1}{n} \sum_{i=1}^{n} y_i \; - \; \frac{w_1}{n} \sum_{i=1}^{n} x_i \; = \; \bar{y} - w_1 \cdot \bar{x}$$

Regression

One-Dimensional Feature Space (continued) [higher-dimensional]

Minimize RSS(w_0, w_1) by a direct method:

1. $$\frac{\partial}{\partial w_0} \sum_{i=1}^{n} (y_i - w_0 - w_1 \cdot x_i)^2 \;=\; 0$$

$$\leadsto \;\ldots\; \leadsto \quad w_0 \;=\; \frac{1}{n}\sum_{i=1}^{n} y_i \;-\; \frac{w_1}{n}\sum_{i=1}^{n} x_i \;=\; \bar{y} - w_1 \cdot \bar{x}$$

2. $$\frac{\partial}{\partial w_1} \sum_{i=1}^{n} (y_i - w_0 - w_1 \cdot x_i)^2 \;=\; 0$$

$$\leadsto \;\ldots\; \leadsto \quad w_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Regression

One-Dimensional Feature Space (continued) [higher-dimensional]

Minimize RSS(w_0, w_1) by a direct method:

1. $$\frac{\partial}{\partial w_0} \sum_{i=1}^{n} (y_i - w_0 - w_1 \cdot x_i)^2 \; = \; 0$$

   $$\rightsquigarrow \; \ldots \; \rightsquigarrow \quad \hat{w}_0 \; = \; \frac{1}{n} \sum_{i=1}^{n} y_i \; - \; \frac{w_1}{n} \sum_{i=1}^{n} x_i \; = \; \bar{y} - \hat{w}_1 \cdot \bar{x}$$

2. $$\frac{\partial}{\partial w_1} \sum_{i=1}^{n} (y_i - w_0 - w_1 \cdot x_i)^2 \; = \; 0$$

   $$\rightsquigarrow \; \ldots \; \rightsquigarrow \quad \hat{w}_1 \equiv w_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Regression

Higher-Dimensional Feature Space

- Recall Equation (1) :

  $$\text{RSS}(\mathbf{w}) = \sum_{\mathbf{x}_i \in D} (y(\mathbf{x}_i) - \mathbf{w}^T \mathbf{x}_i)^2$$

- Let $\mathbf{X}$ denote the $n \times (p+1)$ matrix,
  where row i is the extended input vector $(1 \;\; \mathbf{x}_i^T)$, $\mathbf{x}_i \in D$.

  Let $\mathbf{y}$ denote the n-vector of outputs in the training set D.

Regression

Higher-Dimensional Feature Space

- Recall Equation (1) :

$$\text{RSS}(\mathbf{w}) = \sum_{\mathbf{x}_i \in D} (y(\mathbf{x}_i) - \mathbf{w}^T\mathbf{x}_i)^2$$

- Let $\mathbf{X}$ denote the $n \times (p+1)$ matrix, where row i is the extended input vector $(1 \;\; \mathbf{x}_i^T)$, $\mathbf{x}_i \in D$.

  Let $\mathbf{y}$ denote the n-vector of outputs in the training set D.

$\rightsquigarrow$ $\text{RSS}(\mathbf{w}) = (\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w})$

RSS$(\mathbf{w})$ is a quadratic function in $p+1$ parameters.

Regression

Higher-Dimensional Feature Space (continued) [one-dimensional]

Minimize RSS$(\mathbf{w})$ by a direct method:

$$\frac{\partial\,\text{RSS}}{\partial\mathbf{w}} = -2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) = 0, \qquad \frac{\partial^2\,\text{RSS}}{\partial\mathbf{w}\partial\mathbf{w}^T} = -2\mathbf{X}^T\mathbf{X}$$ [Wikipedia 1, 2, 3]

$$\begin{aligned} & \mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) &=& \; 0 \\ \Leftrightarrow \quad & \mathbf{X}^T\mathbf{X}\mathbf{w} &=& \; \mathbf{X}^T\mathbf{y} \\ \leadsto \quad & \mathbf{w} &=& \; (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\,\mathbf{y} \end{aligned}$$

Regression

Higher-Dimensional Feature Space (continued) [one-dimensional]

Minimize RSS$(\mathbf{w})$ by a direct method:

$$\frac{\partial\,\text{RSS}}{\partial \mathbf{w}} = -2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) = 0, \qquad \frac{\partial^2\,\text{RSS}}{\partial \mathbf{w}\partial \mathbf{w}^T} = -2\mathbf{X}^T\mathbf{X}$$

[Wikipedia 1, 2, 3]

$$\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) = 0$$

$$\Leftrightarrow \quad \mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

Normal equations.

$$\leadsto \quad \mathbf{w} = \underbrace{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T}_{\text{Pseudoinverse of } \mathbf{X}}\,\mathbf{y}$$

If $\mathbf{X}$ has full column rank $p+1$.

[Wikipedia]

Regression

Higher-Dimensional Feature Space (continued) [one-dimensional]

Minimize $\text{RSS}(\mathbf{w})$ by a direct method:

$$\frac{\partial\,\text{RSS}}{\partial\mathbf{w}} = -2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) = 0, \qquad \frac{\partial^2\,\text{RSS}}{\partial\mathbf{w}\partial\mathbf{w}^T} = -2\mathbf{X}^T\mathbf{X}$$ [Wikipedia 1, 2, 3]

$$\mathbf{X}^T(\mathbf{y}-\mathbf{X}\mathbf{w}) = 0$$

$$\Leftrightarrow \quad \mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

Normal equations.

$$\leadsto \quad \hat{\mathbf{w}} \equiv \mathbf{w} = \underbrace{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T}_{\text{Pseudoinverse of } \mathbf{X}}\ \mathbf{y}$$

If $\mathbf{X}$ has full column rank $p+1$.

Pseudoinverse of $\mathbf{X}$ [Wikipedia]

$$\hat{y}(\mathbf{x}_i) = \mathbf{x}_i^T\,\hat{\mathbf{w}}$$

Regression function with least squares estimator $\hat{\mathbf{w}}$.

$$\hat{\mathbf{y}} = \mathbf{X}\,\hat{\mathbf{w}}$$

The n-vector of fitted values at the training input.

$$= \mathbf{X}\,(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

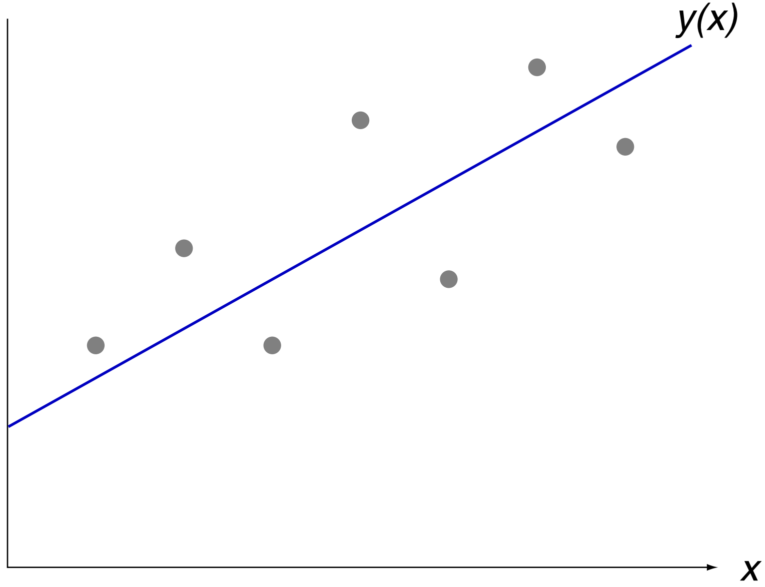
Remarks:

- A curve fitting (or regression) method that is based on the minimization of squared residuals is called a *method of least squares*.
- Various approaches for operationalizing the method of least squares have been devised, in particular for the case of linear model functions. From a numerical viewpoint one can distinguish iterative methods, such as the LMS algorithm, and direct methods, such as solving the normal equations via computing the pseudoinverse.
- More on direct methods. While solving the normal equations is usually fast, it suffers from several deficits: it is numerically unstable and requires singularity handling. Numerically more stable and more accurate methods are based on the QR decomposition and the singular value decomposition, SVD.
- QR decomposition can deal with problems of up to 10^4 variables, provided a dense problem structure. For significantly larger problems (additional 1-2 orders of magnitudes) as well as for sparse matrices iterative solvers are the choice. Even larger, dense problems may be tackled with Artificial Neural Networks.

Regression

Linear Regression for Classification (illustrated for $p = 1$)

Regression learns a real-valued function given as $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$.

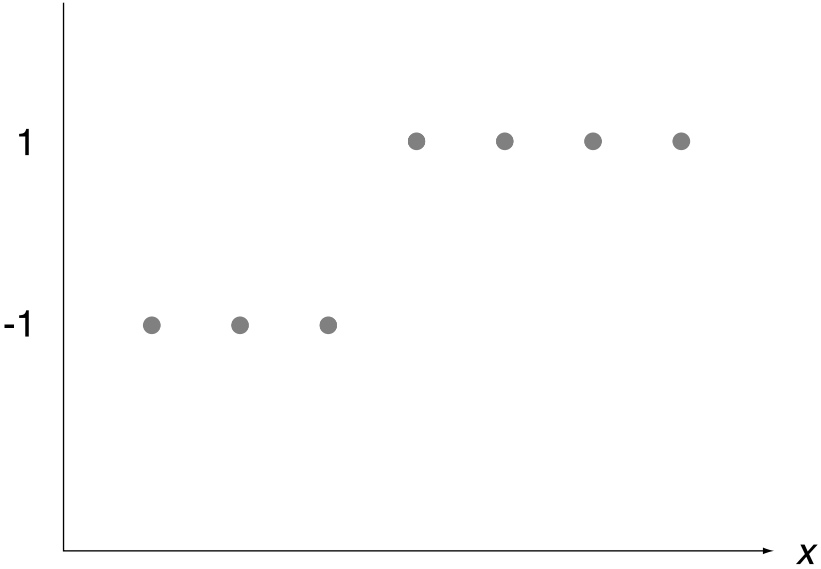


$$y(x) = (w_0 \;\; w_1) \begin{pmatrix} 1 \\ x \end{pmatrix}$$

Regression

Linear Regression for Classification (illustrated for $p = 1$)

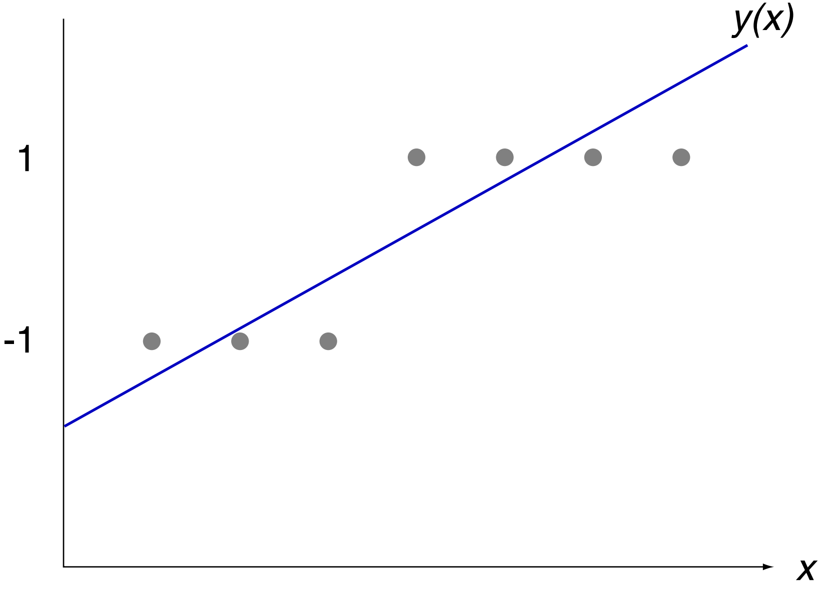
Binary-valued (± 1) functions are also real-valued.



Regression

Linear Regression for Classification (illustrated for $p = 1$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \approx y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.

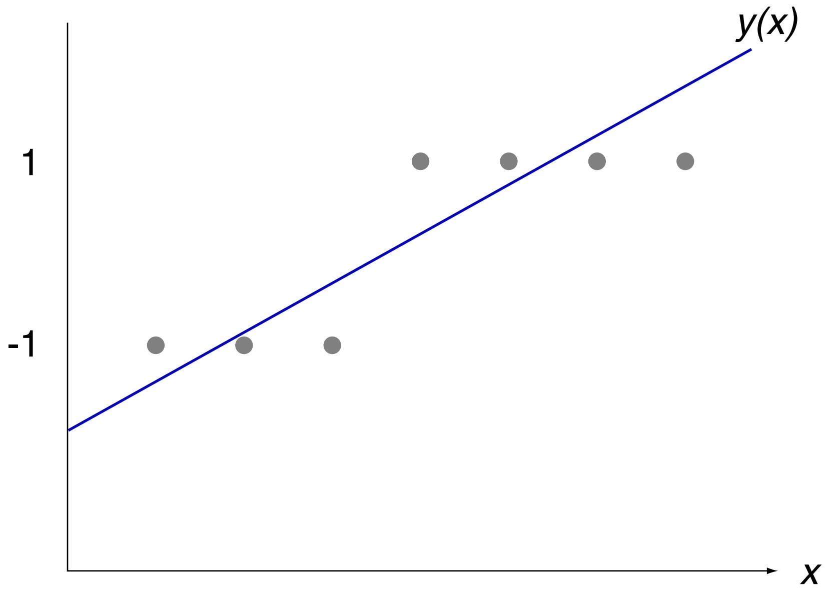


$$y(x) = (w_0 \;\; w_1)\begin{pmatrix}1\\ x\end{pmatrix}$$

Regression

Linear Regression for Classification (illustrated for $p = 1$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.



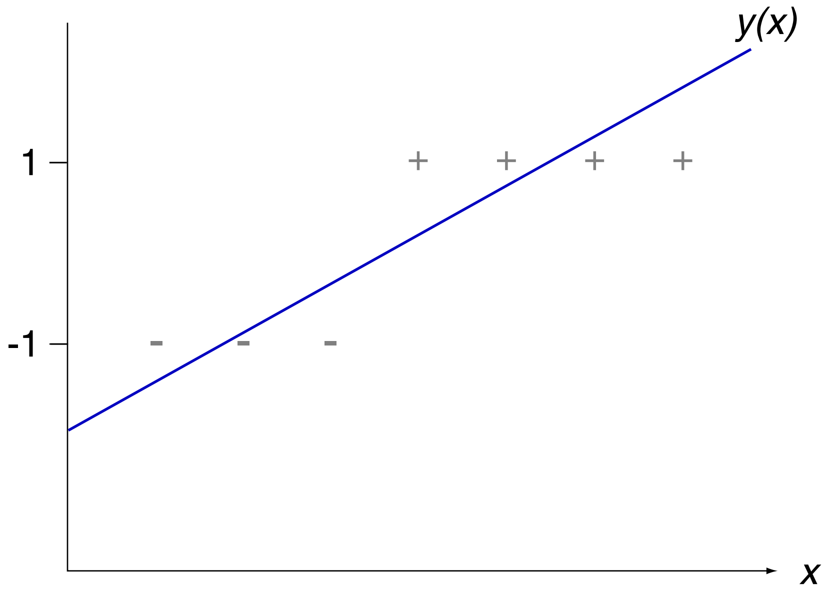
The function "sign$(\mathbf{w}^T\mathbf{x}_i)$" is likely to agree with $y_i = \pm 1$.

- ❑ Regression: $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$
- ❑ Classification: $y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x})$

Regression

Linear Regression for Classification (illustrated for $p = 1$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.



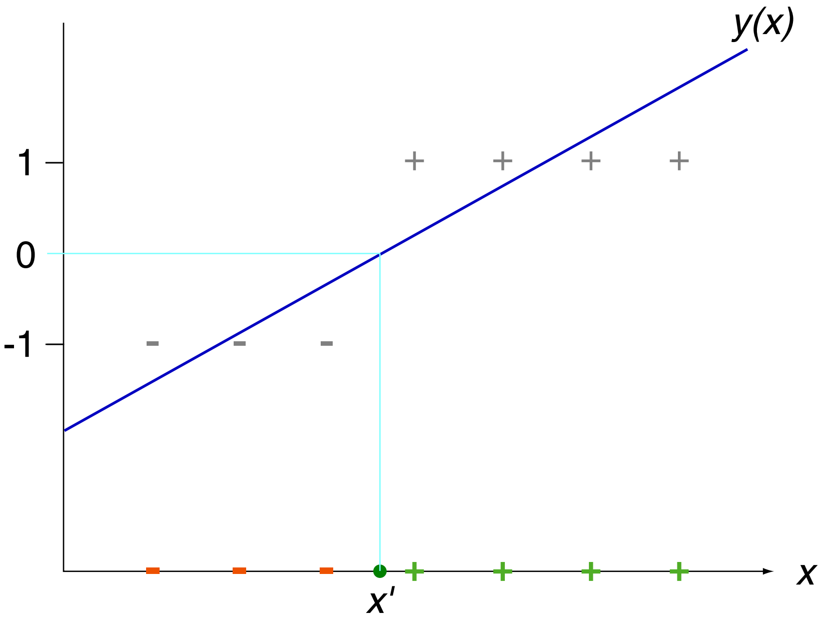
The function "sign$(\mathbf{w}^T\mathbf{x}_i)$" is likely to agree with $y_i = \pm 1$.

- ❑ Regression: $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$
- ❑ Classification: $y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x})$

Regression

Linear Regression for Classification (illustrated for $p = 1$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.



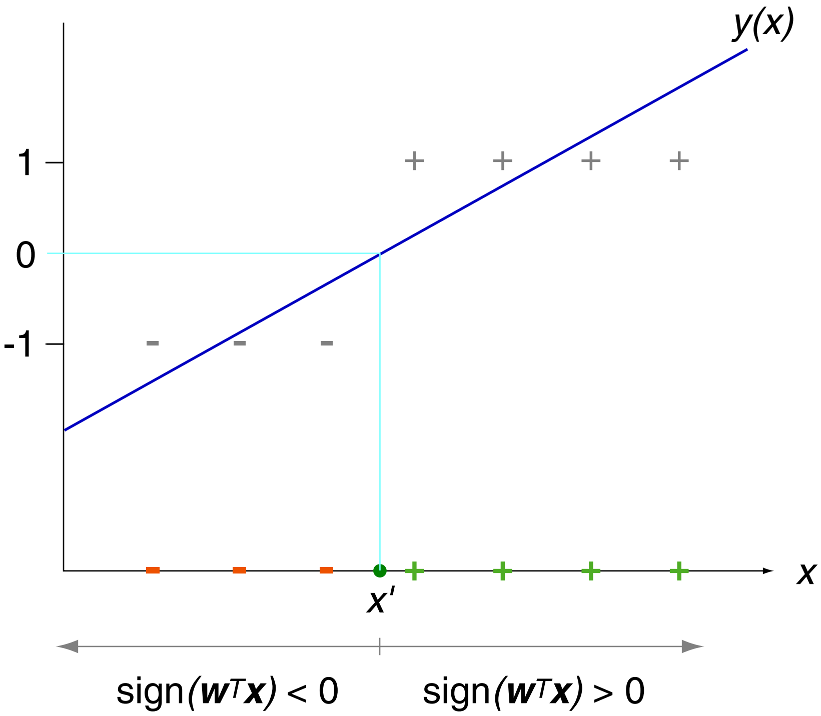
The function "sign$(\mathbf{w}^T\mathbf{x}_i)$" is likely to agree with $y_i = \pm 1$.

- ❑ Regression: $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$
- ❑ Classification: $y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x})$

Regression

Linear Regression for Classification (illustrated for $p = 1$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.

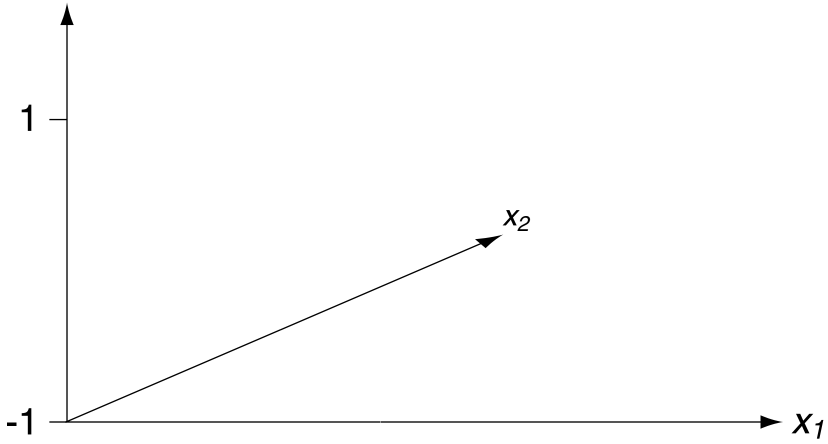


❑ The discrimination point, •, is defined by $w_0 + w_1 \cdot x' = 0$.

❑ For $p = 2$ we are given a discrimination *line*.

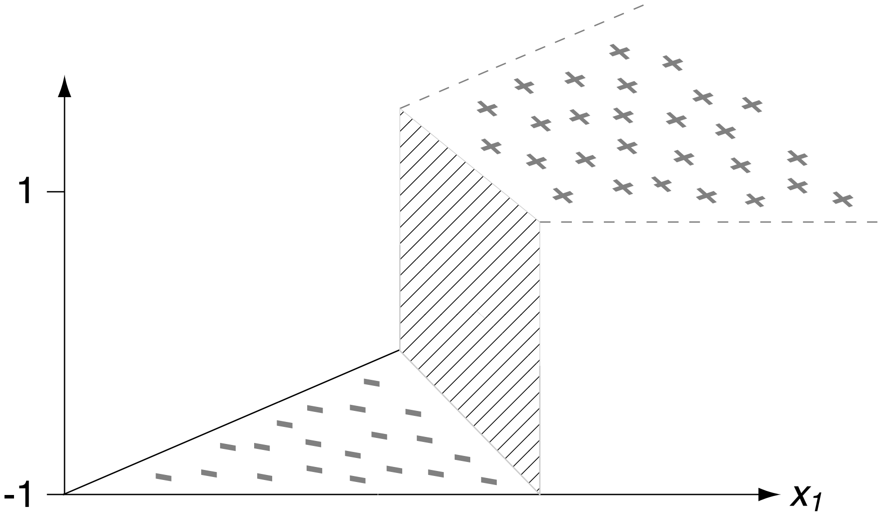
Regression

Linear Regression for Classification (illustrated for $p = 2$)



Regression

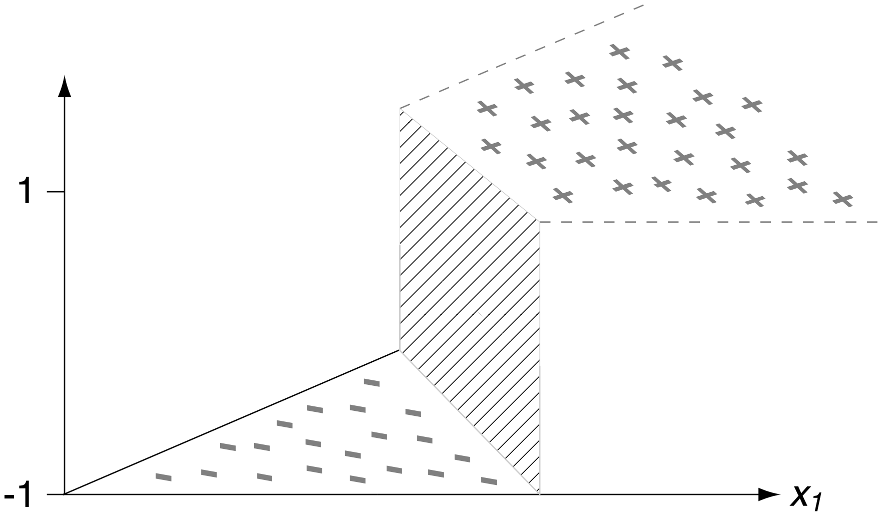
Linear Regression for Classification (illustrated for $p = 2$)



Regression

Linear Regression for Classification (illustrated for $p = 2$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \approx y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.

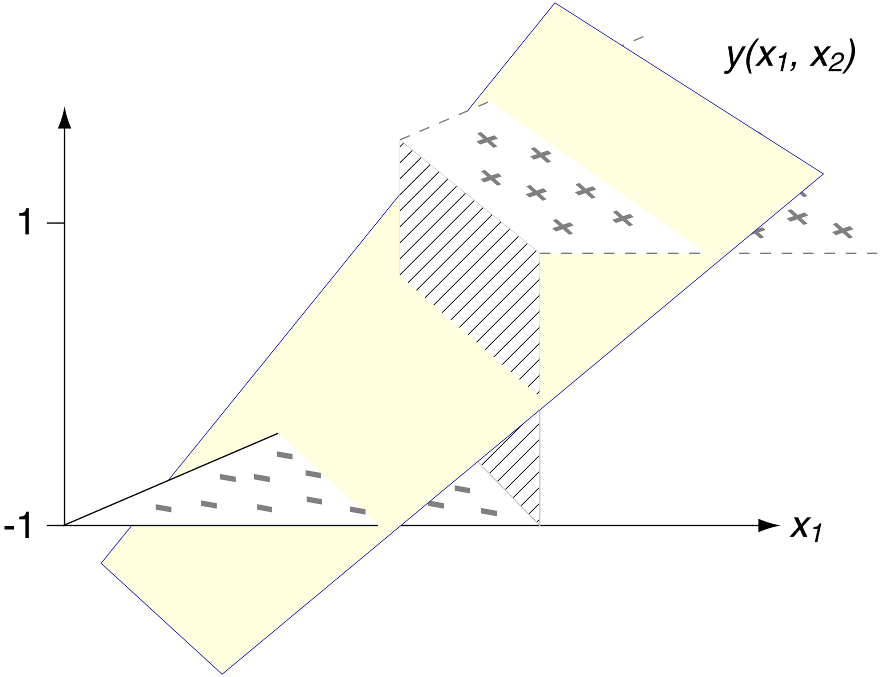


$$y(x) = (w_0 \;\; w_1 \;\; w_2)\begin{pmatrix} 1 \\ x_1 \\ x_2 \end{pmatrix}$$

Regression

Linear Regression for Classification (illustrated for $p = 2$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.



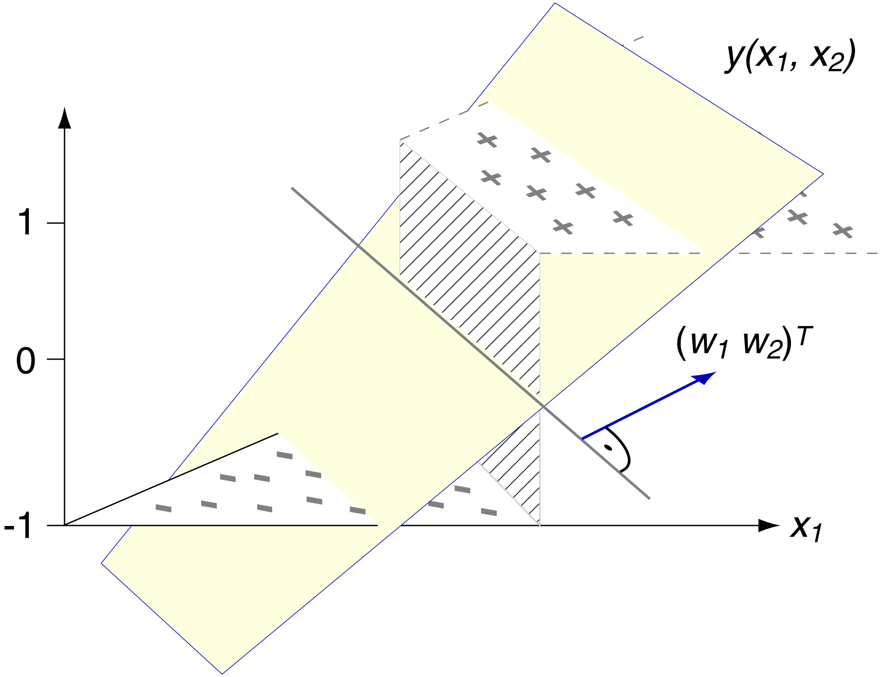
The function "sign$(\mathbf{w}^T\mathbf{x}_i)$" is likely to agree with $y_i = \pm 1$.

- ❑ Regression: $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$
- ❑ Classification: $y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x})$

Regression

Linear Regression for Classification (illustrated for $p = 2$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.



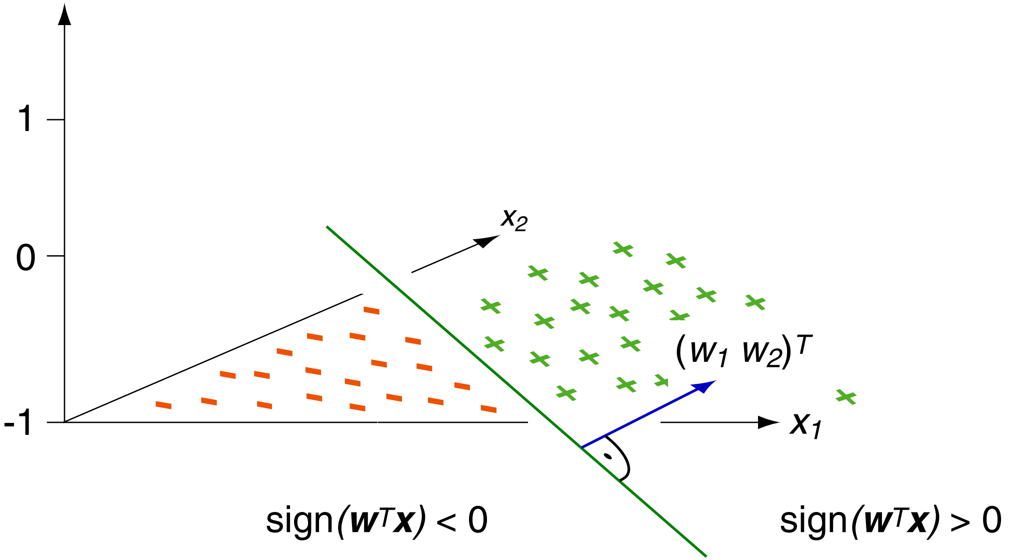
The function "sign$(\mathbf{w}^T\mathbf{x}_i)$" is likely to agree with $y_i = \pm 1$.

- Regression: $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$
- Classification: $y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x})$

Regression

Linear Regression for Classification (illustrated for $p = 2$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \;\approx\; y(\mathbf{x}_i) = \mathbf{w}^T\mathbf{x}_i$.



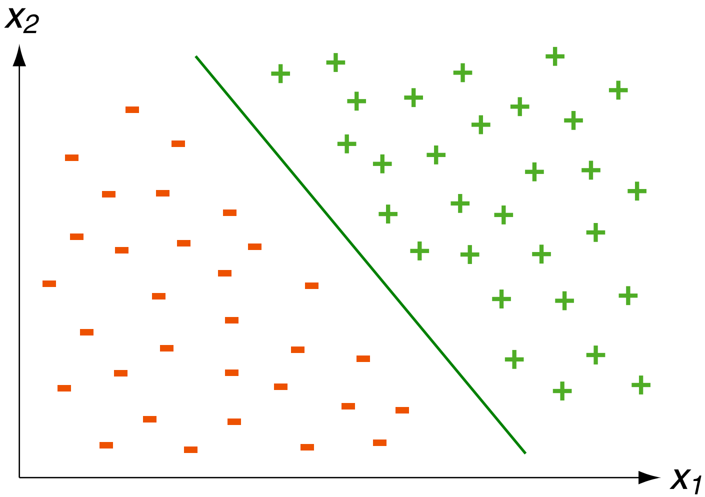
1
0
-1
x_2
x_1
$(w_1\ w_2)^T$
sign$(\boldsymbol{w}^T\boldsymbol{x}) < 0$
sign$(\boldsymbol{w}^T\boldsymbol{x}) > 0$

- The discrimination line, —, is defined by $w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 = 0$.
- For $p = 3$ $(p > 3)$ we are given a discriminating *(hyper)plane*.

Regression

Linear Regression for Classification (illustrated for $p = 2$)

Use linear regression to learn $\mathbf{w}$ from D, where $y_i = \pm 1 \approx y(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i$.



- The discrimination line, —, is defined by $w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 = 0$.
- For $p = 3$ ($p > 3$) we are given a discriminating *(hyper)plane*.

Regression

The Linear Model Function: Variants

The components (variables, random variables) of the input vector $\mathbf{x} = (x_1, \ldots, x_p)$ can come from different sources [Hastie et al. 2001] :

1. quantitative inputs
2. transformations of quantitative inputs, such as $\log x_j$, $\sqrt{x_j}$
3. basis expansions, such as $x_j = (x_1)^j$
4. encoding of a qualitative variable g, $g \in \{1, \ldots, p\}$, as $x_j = I(g = j)$
5. interactions between variables, such as $x_3 = x_1 \cdot x_2$

Regression

The Linear Model Function: Variants

The components (variables, random variables) of the input vector $\mathbf{x} = (x_1, \ldots, x_p)$ can come from different sources [Hastie et al. 2001] :

1. quantitative inputs
2. transformations of quantitative inputs, such as log x_j, $\sqrt{x_j}$
3. basis expansions, such as $x_j = (x_1)^j$
4. encoding of a qualitative variable g, $g \in \{1, \ldots, p\}$, as $x_j = I(g = j)$
5. interactions between variables, such as $x_3 = x_1 \cdot x_2$

No matter the source of the x_j, the model is still linear in the parameters $\mathbf{w}$:

$$y(\mathbf{x}) = w_0 + \sum_{j=1}^{p} w_j \cdot \phi_j(x_j)$$

Regression

The Linear Model Function: Variants

The components (variables, random variables) of the input vector $\mathbf{x} = (x_1, \ldots, x_p)$ can come from different sources [Hastie et al. 2001] :

1. quantitative inputs
2. transformations of quantitative inputs, such as log x_j, $\sqrt{x_j}$
3. basis expansions, such as $x_j = (x_1)^j$
4. encoding of a qualitative variable g, $g \in \{1, \ldots, p\}$, as $x_j = I(g = j)$
5. interactions between variables, such as $x_3 = x_1 \cdot x_2$

No matter the source of the x_j, the model is still linear in the parameters $\mathbf{w}$:

$$y(\mathbf{x}) = w_0 + \sum_{j=1}^{p} w_j \cdot \phi_j(x_j)$$

- linear in the parameters: constant w_j and additive combination

Regression

The Linear Model Function: Variants

The components (variables, random variables) of the input vector $\mathbf{x} = (x_1, \ldots, x_p)$ can come from different sources [Hastie et al. 2001] :

1. quantitative inputs
2. transformations of quantitative inputs, such as $\log x_j$, $\sqrt{x_j}$
3. basis expansions, such as $x_j = (x_1)^j$
4. encoding of a qualitative variable g, $g \in \{1, \ldots, p\}$, as $x_j = I(g = j)$
5. interactions between variables, such as $x_3 = x_1 \cdot x_2$

No matter the source of the x_j, the model is still linear in the parameters $\mathbf{w}$:

$$y(\mathbf{x}) = w_0 + \sum_{j=1}^{p} w_j \cdot \phi_j(x_j)$$

- ❑ linear in the parameters: constant w_j and additive combination
- ❑ basis functions: input variables (space) become(s) feature variables (space)

Regression

The Linear Model Function: Properties of the Solution

Theorem 1 (Gauss-Markov)

Let $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ be a set of examples to be fitted with a linear model function as $y(\mathbf{x}) = \mathbf{x}^T\mathbf{w}$. Within the class of linear unbiased estimators for $\mathbf{w}$, the least squares estimator $\hat{\mathbf{w}}$ has minimum variance, i.e., is most efficient.

Regression

The Linear Model Function: Properties of the Solution

Theorem 1 (Gauss-Markov)

Let $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ be a set of examples to be fitted with a linear model function as $y(\mathbf{x}) = \mathbf{x}^T\mathbf{w}$. Within the class of linear unbiased estimators for $\mathbf{w}$, the least squares estimator $\hat{\mathbf{w}}$ has minimum variance, i.e., is most efficient.

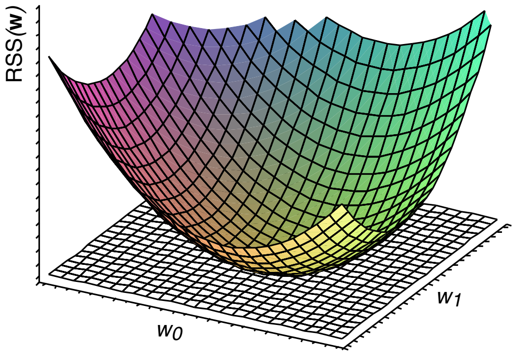
Related followup issues:

- mean and variance of $\hat{\mathbf{w}}$
- proof of the Gauss-Markov theorem
- weak set and strong set of assumptions
- efficiency and consistency of unbiased estimators
- rank deficiencies, where the feature number p exceeds $|D| = n$
- relation of mean least squares and the maximum likelihood principle

Regression

Methods of Least Squares: Iterative versus Direct Methods

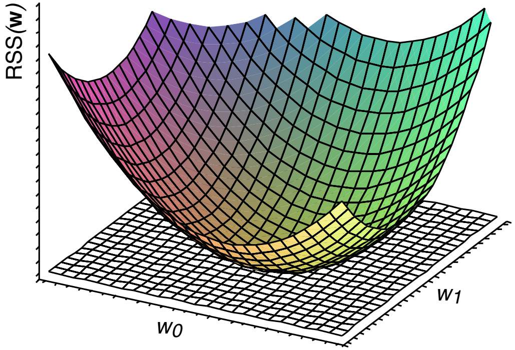
$$\underset{\mathbf{w}}{\text{argmin}}\ \text{RSS}(\mathbf{w}), \quad \text{with}\ \text{RSS}(\mathbf{w}) = \sum_{i=1}^{n}(y_i - \mathbf{w}^T\mathbf{x}_i)^2$$



Regression

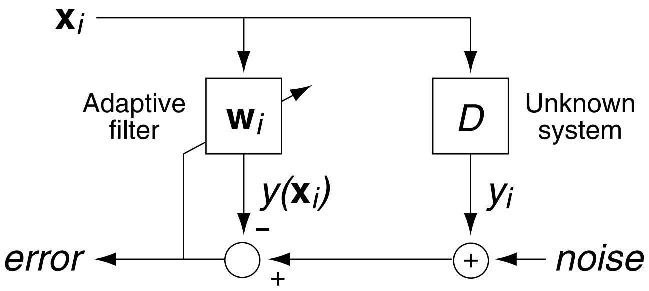
Methods of Least Squares: Iterative versus Direct Methods

$$\underset{\mathbf{w}}{\text{argmin}}\ \text{RSS}(\mathbf{w}), \quad \text{with}\ \text{RSS}(\mathbf{w}) = \sum_{i=1}^{n} (y_i - \mathbf{w}^T\mathbf{x}_i)^2$$



LMS algorithm:

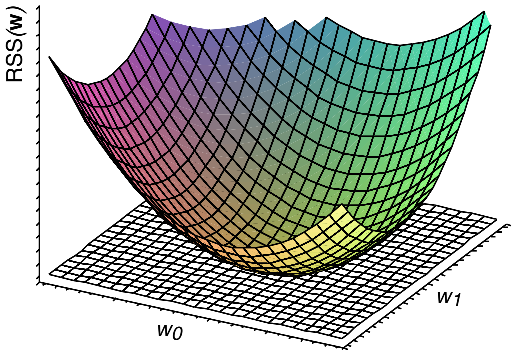
- applicable as online algorithm
- robust algorithm structure
- unsatisfactory convergence
- allows stochastic sampling



Regression

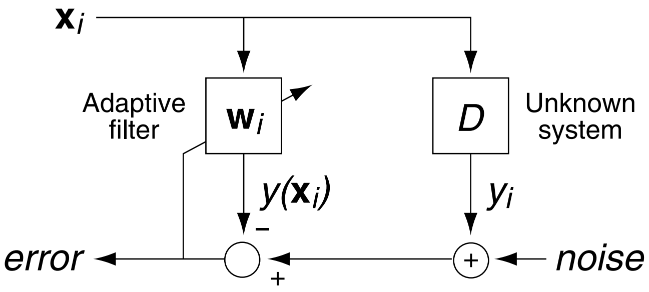
Methods of Least Squares: Iterative versus Direct Methods

$$\underset{\mathbf{w}}{\text{argmin}}\ \text{RSS}(\mathbf{w}), \quad \text{with}\ \text{RSS}(\mathbf{w}) = \sum_{i=1}^{n} (y_i - \mathbf{w}^T\mathbf{x}_i)^2$$



LMS algorithm:

- ❑ applicable as online algorithm
- ❑ robust algorithm structure
- ❑ unsatisfactory convergence
- ❑ allows stochastic sampling



Normal equations:

- ❑ needs complete data
- ❑ numerically unstable
- ❑ requires singularity handling
- ❑ hardly applicable to big data

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

Remarks:

- Recall the definitions for residual and method of least squares.
- The principle of RSS minimization is orthogonal to (= independent of) the type of the model function y, i.e., independent of its dimensionality as well as its linearity or nonlinearity.
- To fit the parameters $\mathbf{w}$ of a (one-dimensional, multi-dimensional, linear, nonlinear) model function y, both the LMS algorithm and direct methods exploit information about the derivative of the RSS term with respect to $\mathbf{w}$. I.e., even if *classification* and not regression is the goal, the distance to the decision boundary (and not the zero-one-loss) is computed, since the latter is not differentiable.
- For a linear model function y, RSS$(\mathbf{w})$ is a convex function and hence a single, global optimum exists.
- Forthcoming: A main goal of machine learning approaches is to avoid overfitting. Overfitting in turn is caused by an inadequate (too high) model function complexity—or, similarly, by insufficient data. A means to reduce the model function complexity is *regularization*.
- Regularization will introduce additional constraints for the model function y or the parameter vector $\mathbf{w}$, respectively. With regularization the minimization expression (2) will consist of two summands: a performance term such as the RSS term, and a penalizing term such as a norm. As before, the first term captures the model function's goodness depending on $\mathbf{w}$, whereas the second term restricts the absolute values of the model function's parameters $\mathbf{w}$.