Moderne Methoden der Datenanalyse
– Ereignisklassifikation –

Roger Wolf

25. Juni 2020

Inhalt der Vorlesung

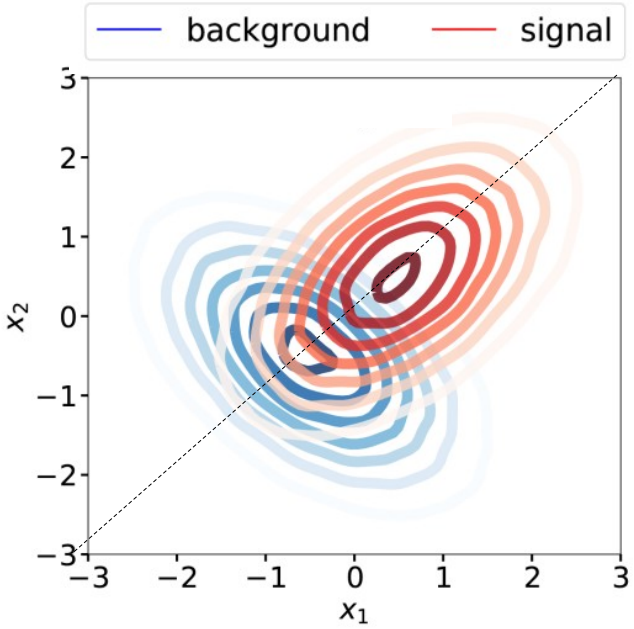
- Ereignisklassifikation.
- Klassifikationsmerkmale, Teststatistik.
- Lineare Diskriminanzanalyse.
- Neuronale Netzwerke:
  - Perceptron.
  - Multilayer Perceptron.
  - Übergang zu reellen Zahlen.
  - Komplizierte Grenzflächen.

Ereignisklassifikation

- Zuordnung eines Stichprobenelements (im folgenden auch *sample* oder Ereignis genannt) zu einer oder mehreren Klassen.
  - Handelt es sich um zwei Klassen („Signal“ und „Untergrund“) spricht man von binärer Klassifikation.
  - Heutzutage werden Klassifikationsprobleme jedoch auch oft als Multiklassifikations-probleme formuliert.
  - Beim Übergang zu unendlich vielen Klassen handelt es sich um ein Regressionspro-blem.

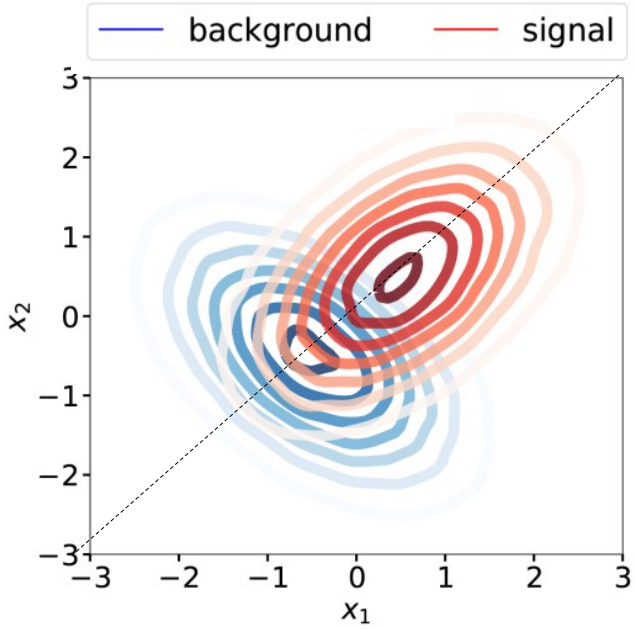
Ereignisklassifikation

- Zuordnung eines Stichprobenelements (im folgenden auch *sample* oder Ereignis genannt) zu einer oder mehreren Klassen.
  - Handelt es sich um zwei Klassen („Signal“ und „Untergrund“) spricht man von binärer Klassifikation.
  - Heutzutage werden Klassifikationsprobleme jedoch auch oft als Multiklassifikations-probleme formuliert.
  - Beim Übergang zu unendlich vielen Klassen handelt es sich um ein Regressionspro-blem.
- Spezielles Problem eines Hypothesentests:
  - In einem solchen Fall bester Diskriminator?



Ereignisklassifikation

- Zuordnung eines Stichprobenelements (im folgenden auch *sample* oder Ereignis genannt) zu einer oder mehreren Klassen.
  - Handelt es sich um zwei Klassen („Signal“ und „Untergrund“) spricht man von binärer Klassifikation.
  - Heutzutage werden Klassifikationsprobleme jedoch auch oft als Multiklassifikations-probleme formuliert.
  - Beim Übergang zu unendlich vielen Klassen handelt es sich um ein Regressionspro-blem.
- Spezielles Problem eines Hypothesentests:
  - In einem solchen Fall bester Diskriminator? – Likeli-hoodquotient, mit 2d Wahrscheinlichkeitsdichten für „Signal“ und „Untergund“.



Klassifikationsmerkmale

- Die Klassifikation erfolgt auf Grundlage mehrerer Merkmale (*features*), die zu einem Merkmaksvektor (*feature vector*) zusammengefasst werden können.

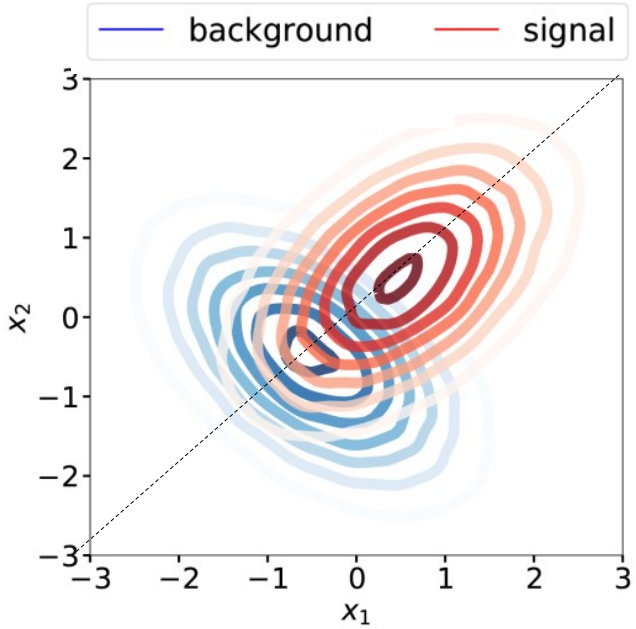
- Im Allgemeinen sind die Klassen in keiner Variablen eindeutig unterscheidbar. Eher trägt die Gesamtheit aller Merkmale zur Unterscheidung der Klassen bei.

Klassifikationsmerkmale

- Die Klassifikation erfolgt auf Grundlage mehrerer Merkmale (*features*), die zu einem Merkmaksvektor (*feature vector*) zusammengefasst werden können.

- Im Allgemeinen sind die Klassen in keiner Variablen eindeutig unterscheidbar. Eher trägt die Gesamtheit aller Merkmale zur Unterscheidung der Klassen bei.

  - In diesem Beispiel umfasst der Merkmalsvektor zwei Merkmale x_1 und x_2.



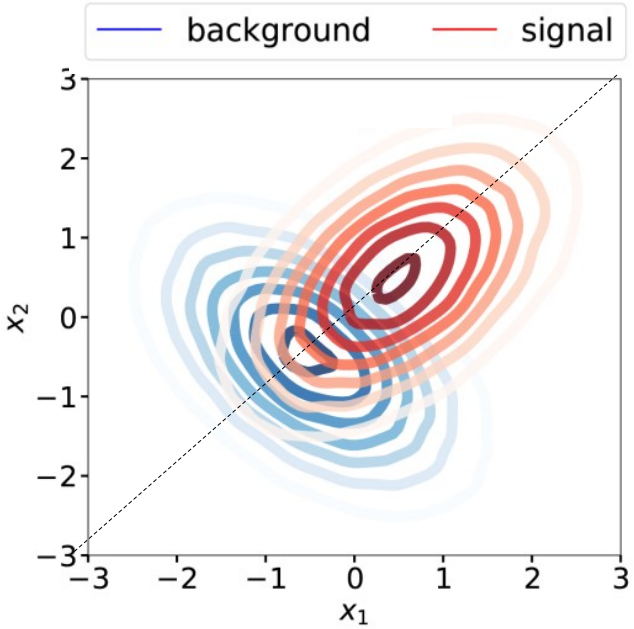
Klassifikationsmerkmale

- Die Klassifikation erfolgt auf Grundlage mehrerer Merkmale (*features*), die zu einem Merkmaksvektor (*feature vector*) zusammengefasst werden können.

- Im Allgemeinen sind die Klassen in keiner Variablen eindeutig unterscheidbar. Eher trägt die Gesamtheit aller Merkmale zur Unterscheidung der Klassen bei.

  - In diesem Beispiel umfasst der Merkmalsvektor zwei Merkmale x_1 und x_2.

  - In realen Beispielen der Teilchenphysik umfasst ein solcher Vektor $\mathcal{O}(10-50)$ Merkmale.



Klassifikationsmerkmale

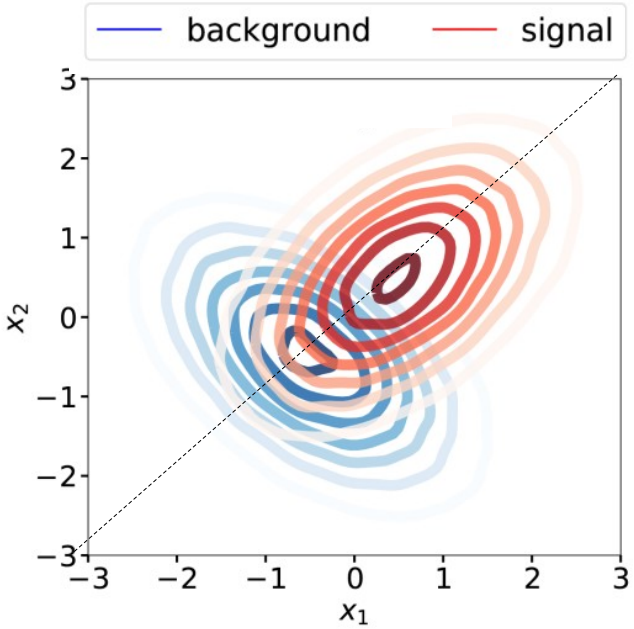
- Die Klassifikation erfolgt auf Grundlage mehrerer Merkmale (*features*), die zu einem Merkmaksvektor (*feature vector*) zusammengefasst werden können.

- Im Allgemeinen sind die Klassen in keiner Variablen eindeutig unterscheidbar. Eher trägt die Gesamtheit aller Merkmale zur Unterscheidung der Klassen bei.

- In diesem Beispiel umfasst der Merkmalsvektor zwei Merkmale x_1 und x_2.

- In realen Beispielen der Teilchenphysik umfasst ein solcher Vektor $\mathcal{O}(10-50)$ Merkmale.

- In realen Beispielen des *machine learning*, z.B. in der Bilderkennung kann der Merkmalsvektor einige 100k Elemente umfassen.

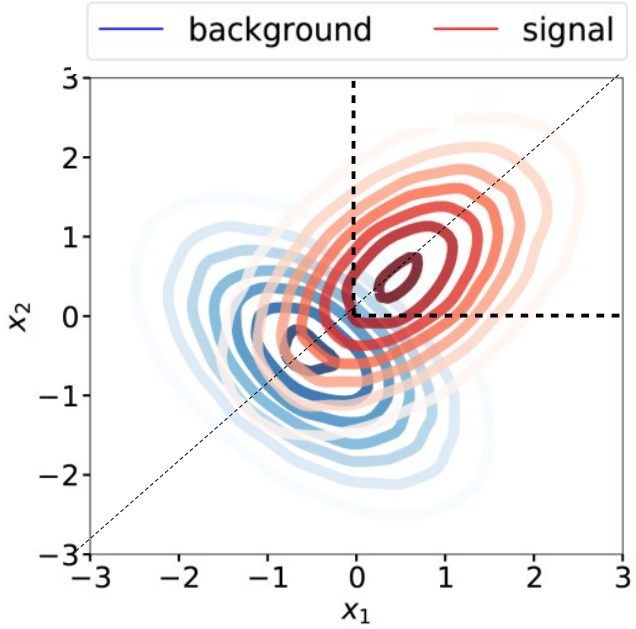


Teststatistik zur Klassifikation

- In solchen Fällen lassen sich die Wahrscheinlichkeitsdichten nicht mehr abbilden (→ Fluch der Dimension).

- Die Zuordnung zu einer gegebenen Klasse erfolgt dann nach praktischen Gesichtspunkten auf Grundlage einer (ein oder mehrdimensionalen) Teststatistik, die mehr oder weniger komplex ausgeprägt sein kann:

  - Selektionsschnitte („cuts“) auf einzelne Variablen (→ hier 2d Teststatistik).



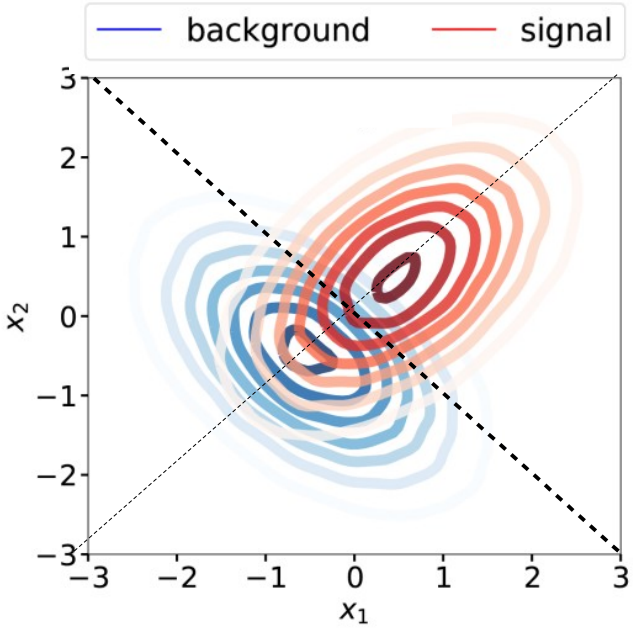
Teststatistik zur Klassifikation

- In solchen Fällen lassen sich die Wahrscheinlichkeitsdichten nicht mehr abbilden (→ Fluch der Dimension).

- Die Zuordnung zu einer gegebenen Klasse erfolgt dann nach praktischen Gesichtspunkten auf Grundlage einer (ein oder mehrdimensionalen) Teststatistik, die mehr oder weniger komplex ausgeprägt sein kann:

  - Selektionsschnitte („cuts“) auf einzelne Variablen (→ hier 2d Teststatistik).

  - Linearkombinationen von Schnitten auf einzelne Variablen (→ 1d Lineare Diskriminanten).



Teststatistik zur Klassifikation

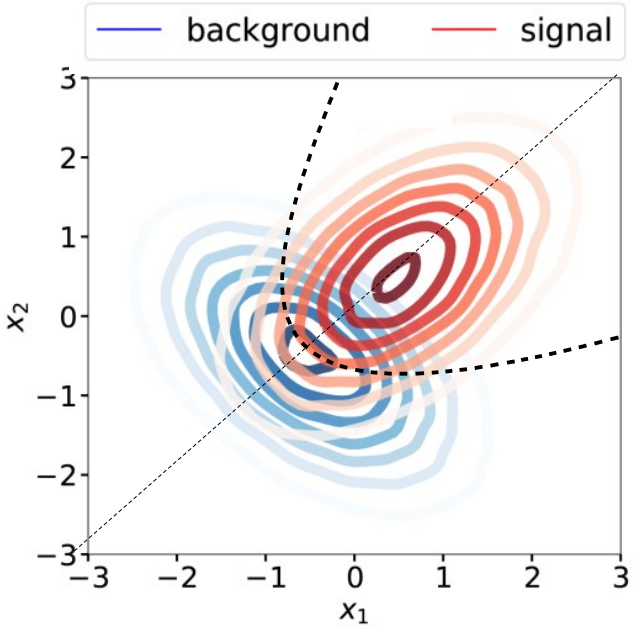
- In solchen Fällen lassen sich die Wahrscheinlichkeitsdichten nicht mehr abbilden (→ Fluch der Dimension).

- Die Zuordnung zu einer gegebenen Klasse erfolgt dann nach praktischen Gesichtspunkten auf Grundlage einer (ein oder mehrdimensionalen) Teststatistik, die mehr oder weniger komplex ausgeprägt sein kann:

  - Selektionsschnitte („cuts“) auf einzelne Variablen (→ hier 2d Teststatistik).

  - Linearkombinationen von Schnitten auf einzelne Variablen (→ 1d Lineare Diskriminanten).

  - Nichtlineare Diskriminanten (→ Maschinelles Lernen, Neuronale Netze (NN) und *Boosted Decision Trees* (BDT) in 1d oder höherdimensional).



Lineare Diskriminanzanalyse

- Liegen zwei Ereignisklassen so vor wie in Bild (1) angezeigt, ist eine Klassifikation mit Hilfe von Selektionsschnitten unzureichend:

- Es ist jedoch möglich, eine geeignete Linearkombination im Ereignisraum zu finden mit der die Klassifikation erfolgreicher erfolgen kann. Siehe Bild (2).

Bild (1)

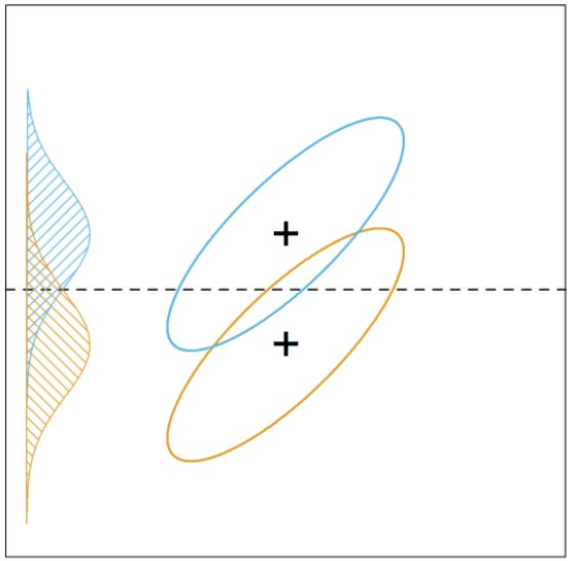
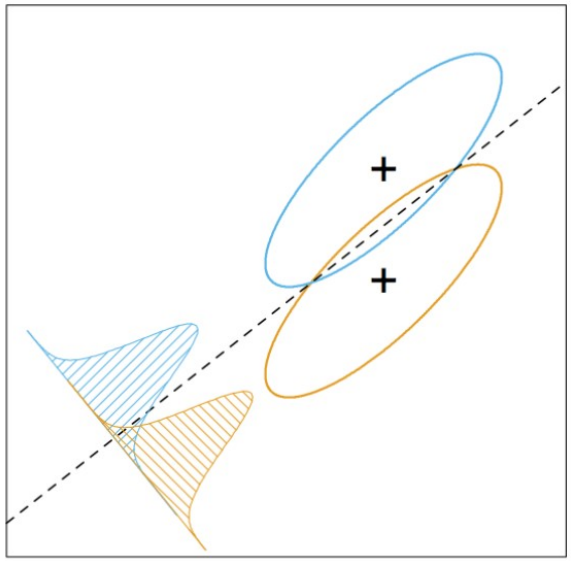


Bild (2)



- **Aufgabe**: bestimme eine Teststatistik mit bestmöglicher Trennung zwischen Signal und Untergrund.

Fisher Diskriminante

- Diese kann durch die Fisher Diskriminante erreicht werden:

$$t(\vec{x}) = \sum_{i=1}^{n} a_i x_i$$

- Die Koeffizienten $\{a_i\}$ können aus den folgenden Randbedingungen bestimmt werden:
  - Die Differenz der Mittelwerte $\mu_s = \overline{x}(\{x_i\}, \{a_i\}|\mathrm{signal})$ und $\mu_b = \overline{x}(\{x_i\}, \{a_i\}|\mathrm{BG})$ soll maximal werden.
  - Die Streuung der Mittelwerte σ_s und σ_b soll minimal werden.
  - Maximiere den Koeffizienten:

$$J(\vec{a}) = \frac{(\mu_s - \mu_b)^2}{\sigma_s^2 + \sigma_b^2}$$

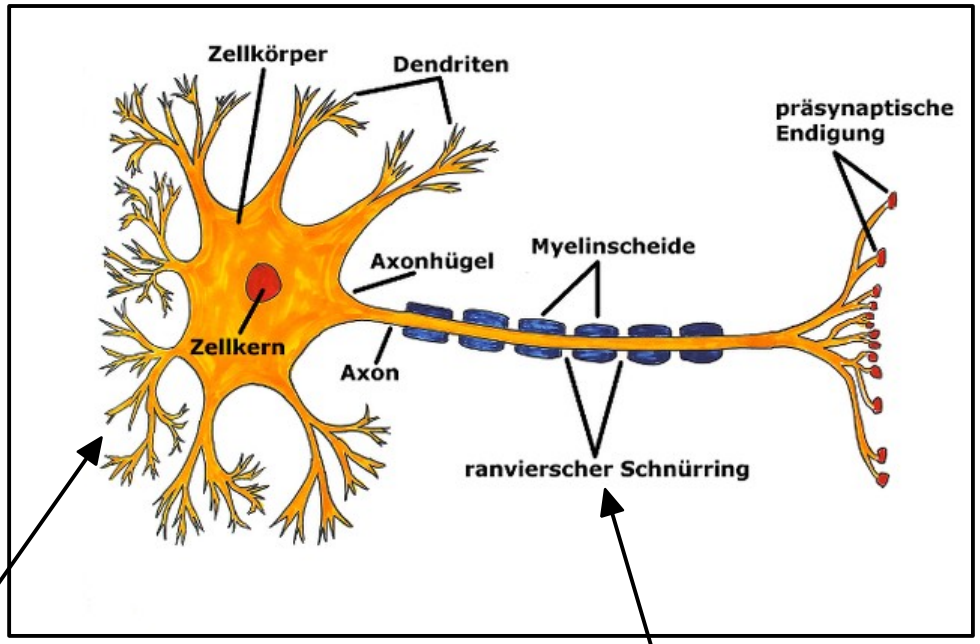
    d.h. bestimme die Koeffizienten $\{a_i\}$ so, dass $\vec{\nabla}_{\vec{a}} J(\vec{a}) = 0$.

- Für lineare Probleme kann man zeigen, dass die Fisher Diskriminante zum Likelihoodquotienten äquivalent ist.

Ereignisklassifikation mit Hilfe neuronaler Netze

- Der (interessanten!) Histogrie nach gründen neuronale Netwerke (NN) auf der neurobiologischen Theorie des (menschlichen) Lernens:

Schematische Darstellung einer Nervenzelle

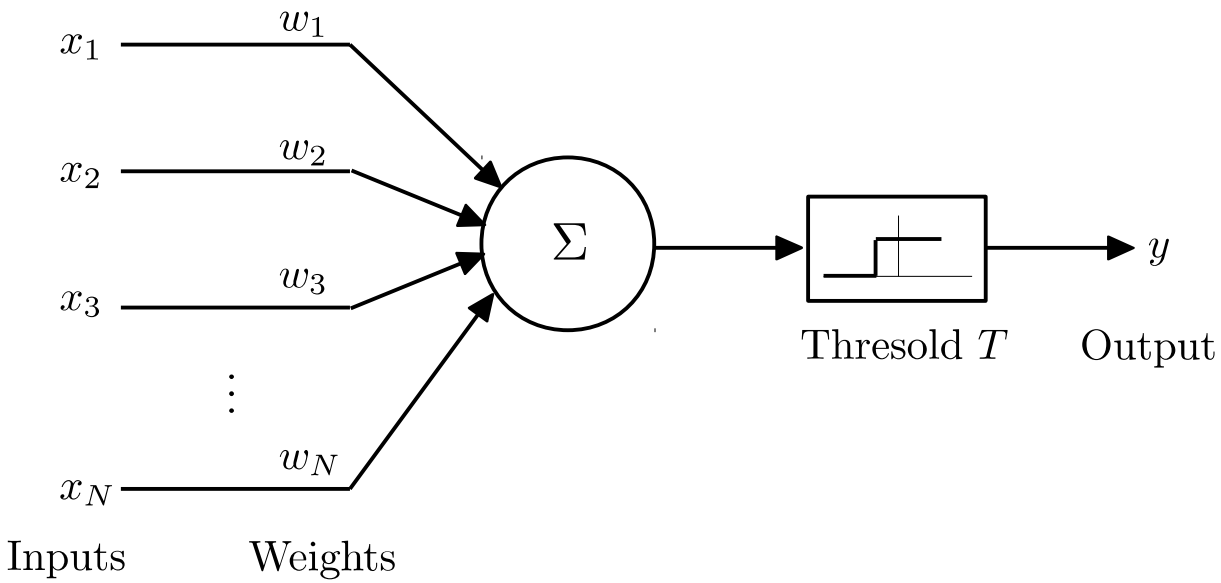


Viele u.U. kleine Signale kommen von hier

Wenn Summe einkommender Signale eine Schwelle überschreitet „feuert“ die Zelle ein eigenes Signal entlang des Axons.

Das Perceptron

- Das mathematische Modell hierzu (nach einigen Irrungen und Wirrungen) stammt von Frank Rosenblatt (11.07.1928 – 11.07.1971):

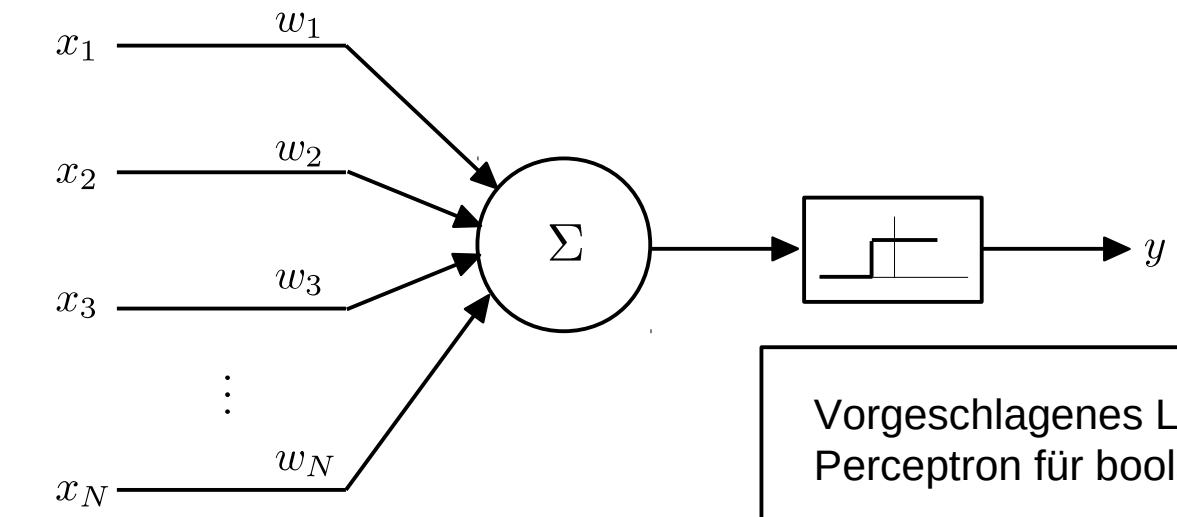


$$y = \begin{cases} 1 & \text{if } \sum_i^N w_i x_i - T > 0 \\ 0 & \text{else} \end{cases}$$

Logik: „feure“ wenn einkommende Signale eine Schwelle überschreiten.

Das Perceptron

- Das mathematische Modell hierzu (nach einigen Irrungen und Wirrungen) stammt von Frank Rosenblatt (11.07.1928 – 11.07.1971):



$$y = \begin{cases} 1 & \text{if } \sum_i^N w_i x_i - T > 0 \\ 0 & \text{else} \end{cases}$$

Vorgeschlagenes Lernmodell für das Perceptron für boolsche Probleme:

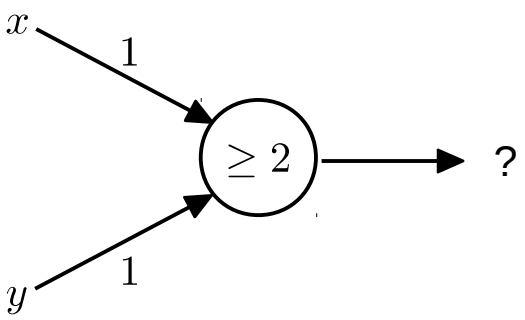
$$\vec{w}' = \vec{w} + \eta \left(d(\vec{x}) - y(\vec{x}) \right)$$

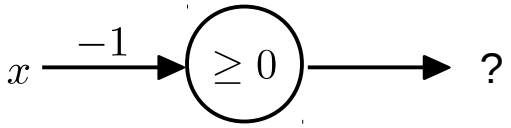
Aktualisiere $\vec{w}$ immer, wenn das Perceptron mit seinem output falsch liegt.

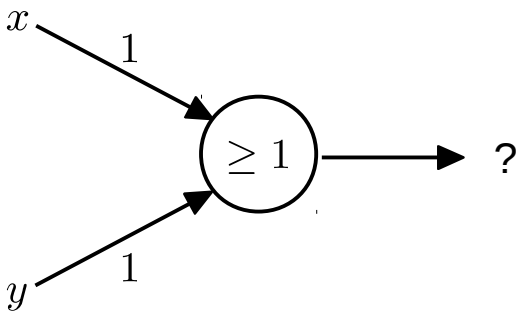
Rosenblatt konnte zeigen, dass dieses Modell für solche Probleme auf die richtige Lösung konvergiert.

Logische Operationen

- Adaptiert man die Gewichte und Schwellen, kann das Perceptron jedes logische Gatter abbilden:



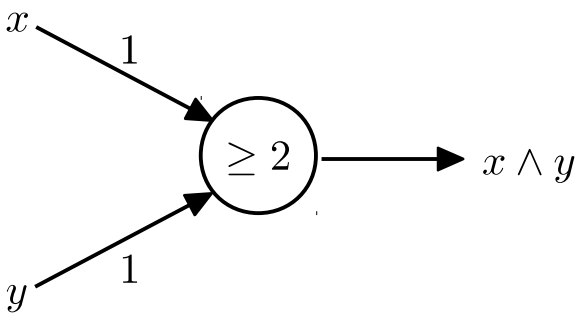


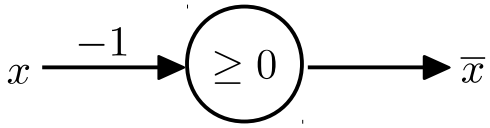


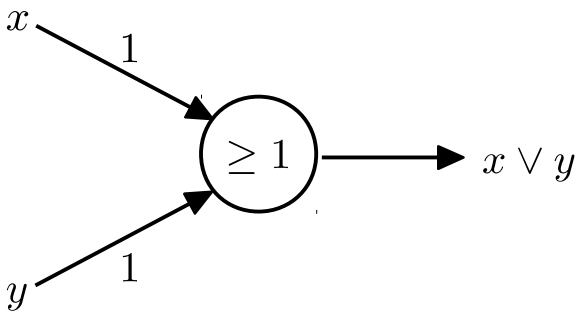
NB: Werte auf den Pfeilen entsprechen Gewichten $w_{x,y}$, Werte in den Kreisen entsprechen Schwellen T.

Logische Operationen

- Adaptiert man die Gewichte und Schwellen, kann das Perceptron jedes logische Gatter abbilden:



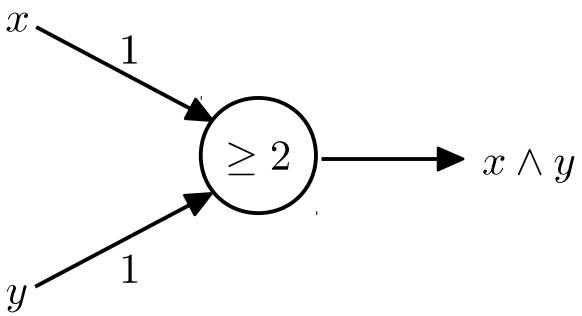


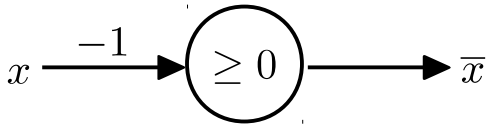


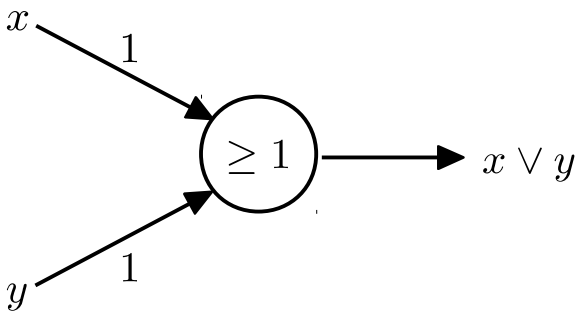
NB: Werte auf den Pfeilen entsprechen Gewichten $w_{x,y}$, Werte in den Kreisen entsprechen Schwellen T.

Logische Operationen

- Adaptiert man die Gewichte und Schwellen, kann das Perceptron jedes logische Gatter abbilden:

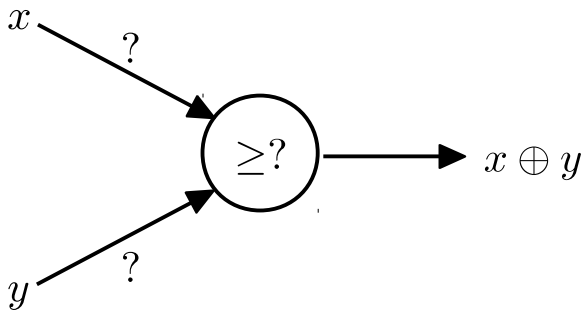






NB: Werte auf den Pfeilen entsprechen Gewichten $w_{x,y}$, Werte in den Kreisen entsprechen Schwellen T.

- Alle, bis auf eines: „XOR“

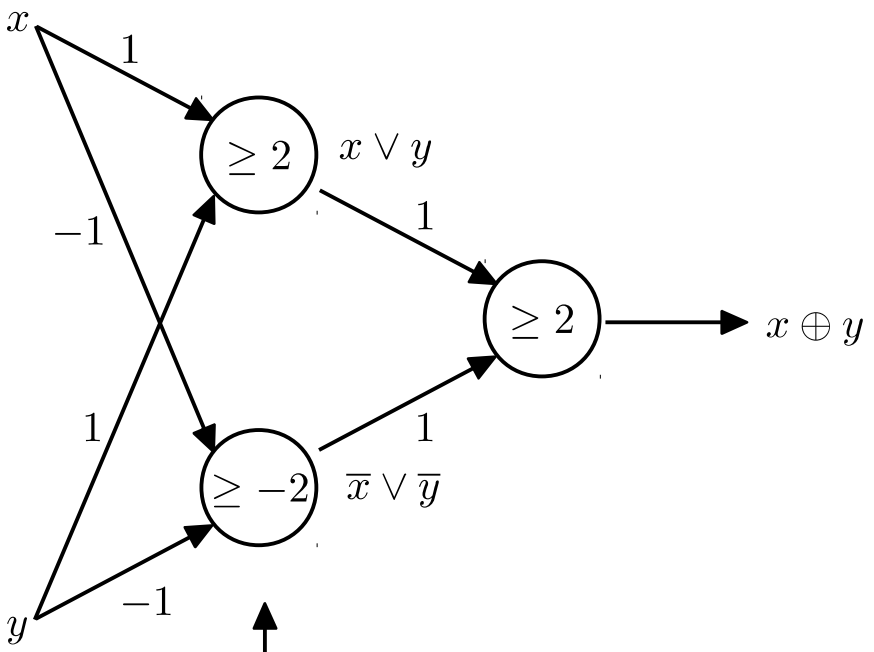


Diskutiert in Marvin Minsky, Seymour Papert „Perceptrons: An Introduction to Computational Geometry“, 1968 (review here).

- D.h. ein einzelnes Perceptron ist keine universelle Recheneinheit.

Lösung des XOR Problems für das Perceptron

- Die Lösung liegt darin mehrere Perceptrons zu kombinieren:



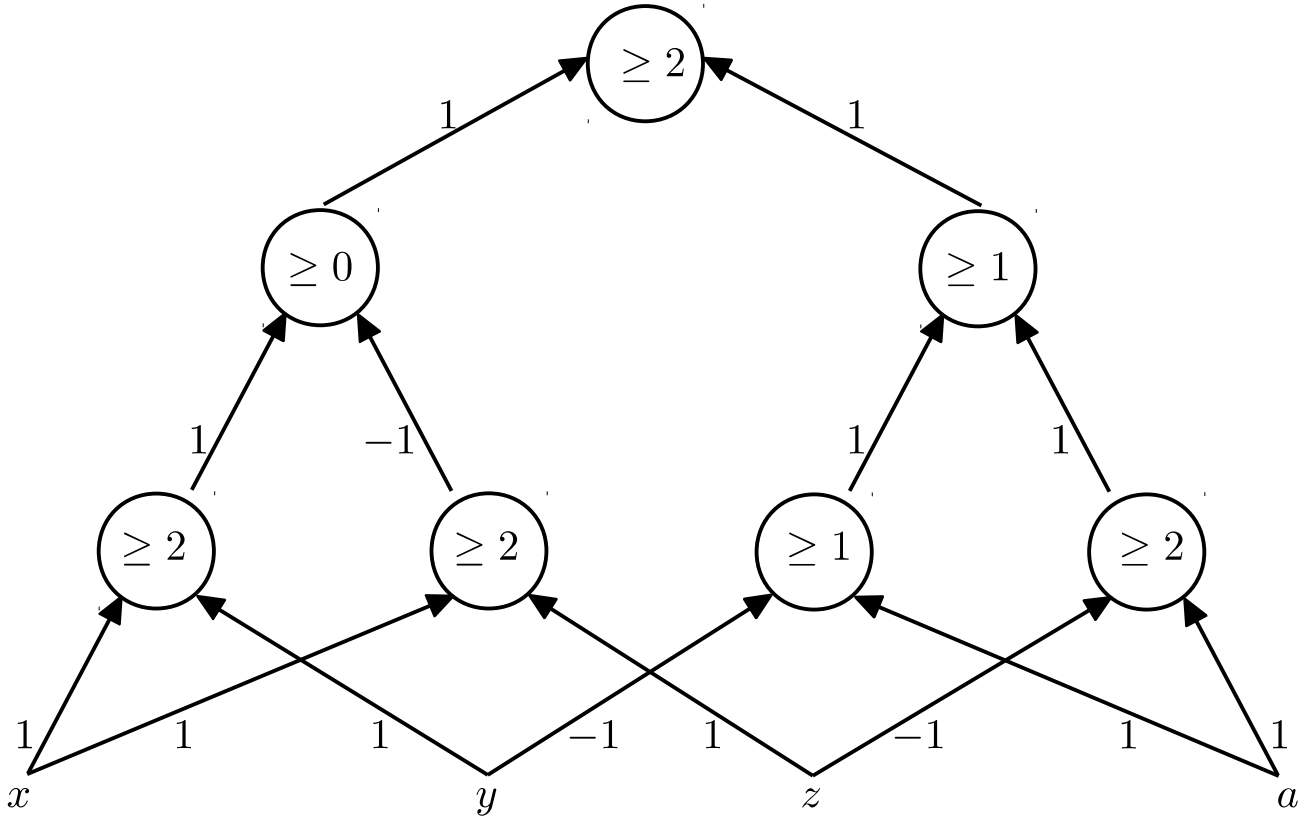
Diskutiert in Marvin Minsky, Seymour Papert „Perceptrons: An Introduction to Computational Geometry“, 1968 (Review hier).

Die erste Lage ist eine „*hidden layer*“

Das Multilayer Perceptron

- Wenn kombiniert kann ein mehrlagiges Perceptron beliebig komplizierte boolsche Operationen abbilden:

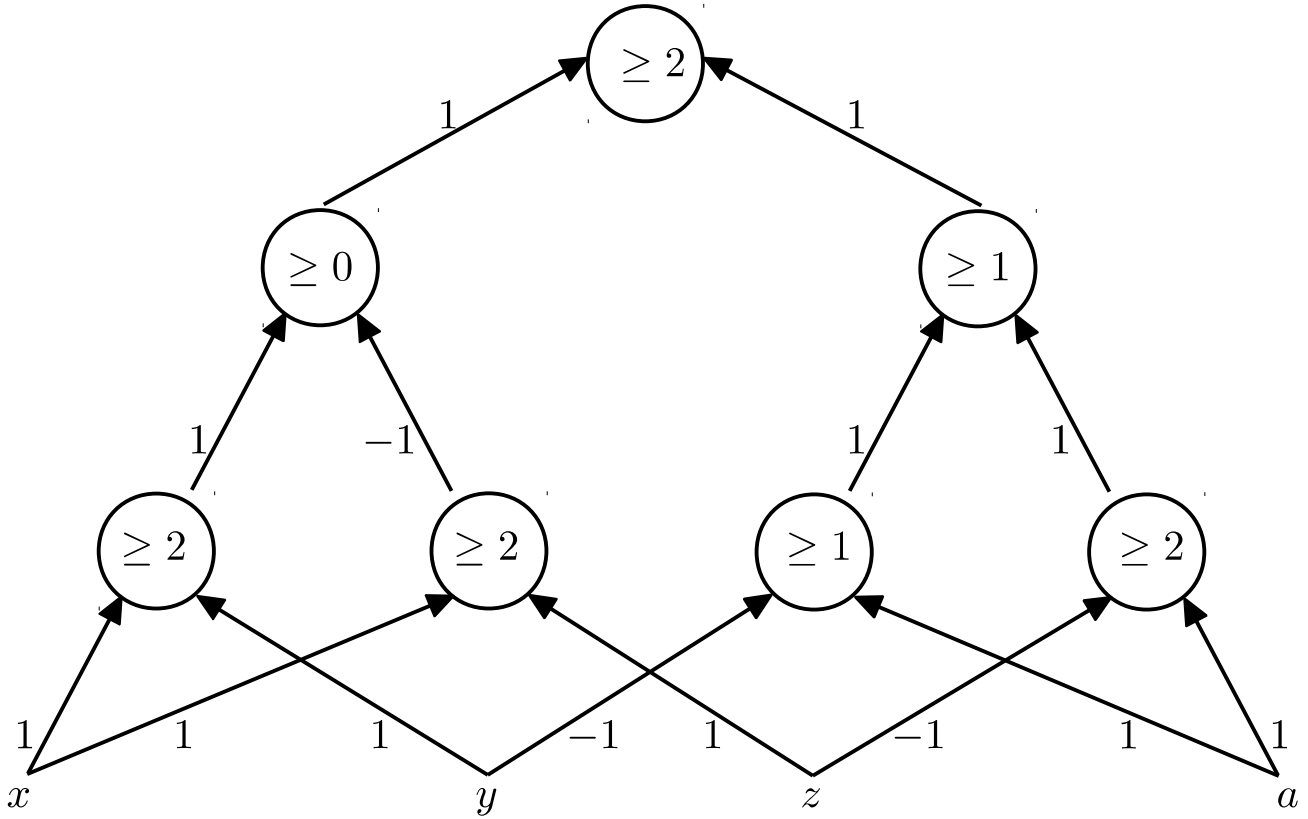
  Welche logische
  Verknüpfung wäre das hier?



Das Multilayer Perceptron

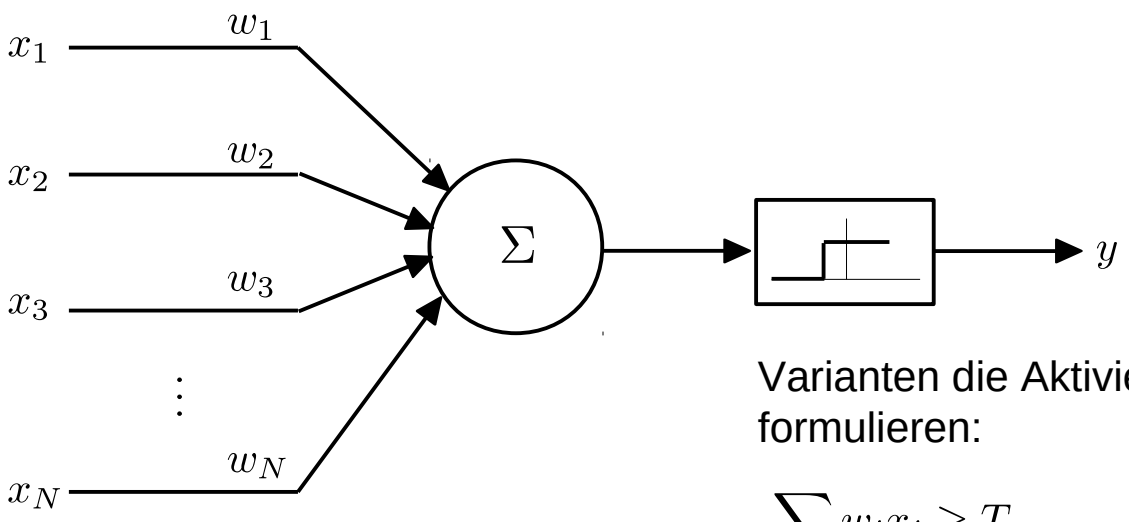
- Wenn kombiniert kann ein mehrlagiges Perceptron beliebig komplizierte boolsche Operationen abbilden:

Welche logische Verknüpfung wäre das hier? $\Big((a \wedge \overline{x} \wedge z) \vee (a \wedge \overline{y})\Big) \wedge \Big((x \wedge y) \vee \overline{(x \wedge z)}\Big)$



Übergang von boolscher Logik zu reellen Zahlen

- Wir wollen nicht nur logische Oprationen ausführen. Der Übergang zu reellen Zahlen erfolgt wie folgt:



Varianten die Aktivierungslogik zu formulieren:

$$\sum_i w_i x_i \geq T\,,$$

$$\sum_i w_i x_i - T \geq 0\,,$$

$$\theta\left(\sum_i w_i x_i - T\right) \quad \text{(Heavyside Funktion)}$$

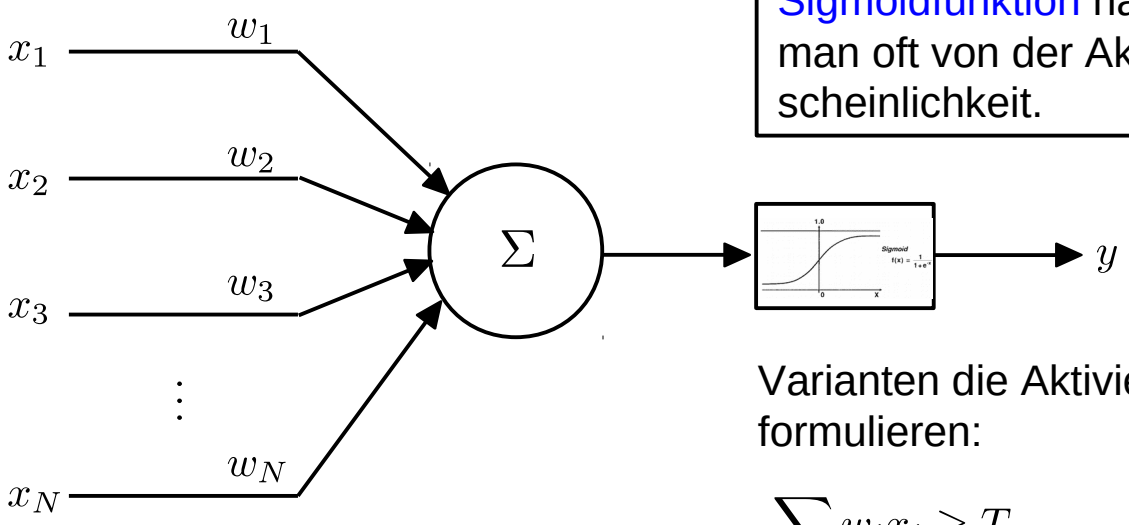
$x_1 \ldots x_N \in \mathbb{R}$

$w_1 \ldots x_N \in \mathbb{R}$

Einheit feuert, wenn $\sum_i w_i x_i \geq T$

Übergang von boolscher Logik zu reellen Zahlen

- Wir wollen nicht nur logische Oprationen ausführen. Der Übergang zu reellen Zahlen erfolgt wie folgt:



Wenn es sich bei $\theta(\cdot)$ z.B. um eine Sigmoidfunktion handelt spricht man oft von der Aktivierungswahrscheinlichkeit.

Varianten die Aktivierungslogik zu formulieren:

$$\sum_i w_i x_i \geq T\,,$$

$$\sum_i w_i x_i - T \geq 0\,,$$

$$\theta\left(\sum_i w_i x_i - T\right) \quad \text{(Heavyside Funktion)}$$

$x_1 \ldots x_N \in \mathbb{R}$

$w_1 \ldots x_N \in \mathbb{R}$

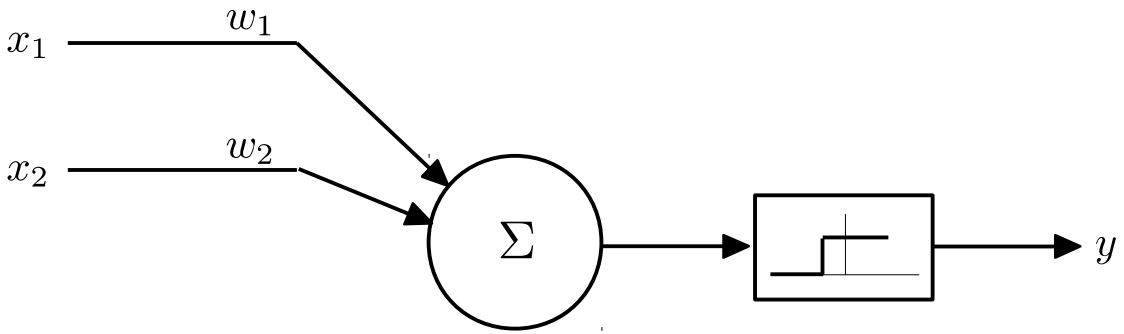
Einheit feuert, wenn $\sum_i w_i x_i \geq T$

$\theta(\cdot)$ könnte jede reelwertige Funktion sein

Übergang von boolscher Logik zu reellen Zahlen

- Wir kehren kurzzeitig und der Anschaulichkeit halber zur boolschen Aktivierung zurück.

- Betrachten wir zwei reellwertige Eingabewerte (*inputs*) x_1 und x_2. Die Einheit „feuert“ oberhalb einer bestimmten Schwelle T. Welcher Grenzfläche entspricht dies im von x_1 und x_2 aufgespannten Raum?

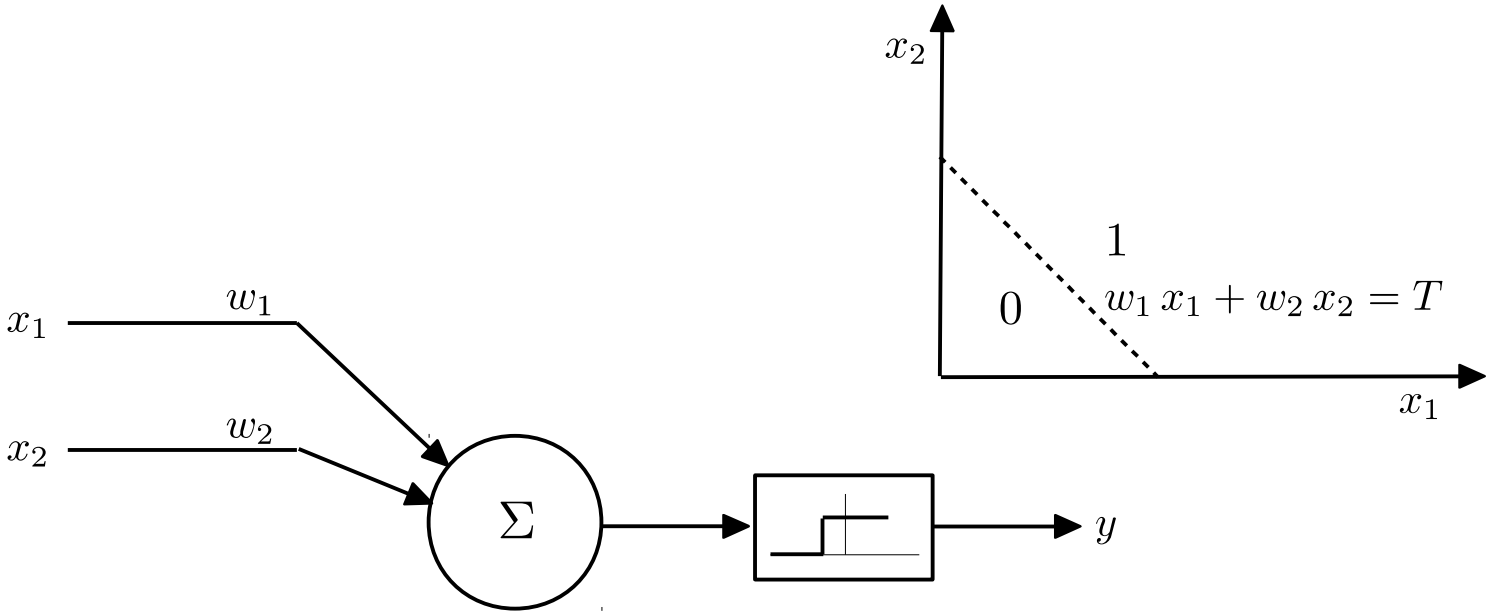


$$y = \begin{cases} 1 & \text{if } \sum_i^2 w_i x_i - T > 0 \\ 0 & \text{else} \end{cases}$$

Übergang von boolscher Logik zu reellen Zahlen

- Wir kehren kurzzeitig und der Anschaulichkeit halber zur boolschen Aktivierung zurück.

- Betrachten wir zwei reellwertige Eingabewerte (*inputs*) x_1 und x_2. Die Einheit „feuert“ oberhalb einer bestimmten Schwelle T. Welcher Grenzfläche entspricht dies im von x_1 und x_2 aufgespannten Raum?

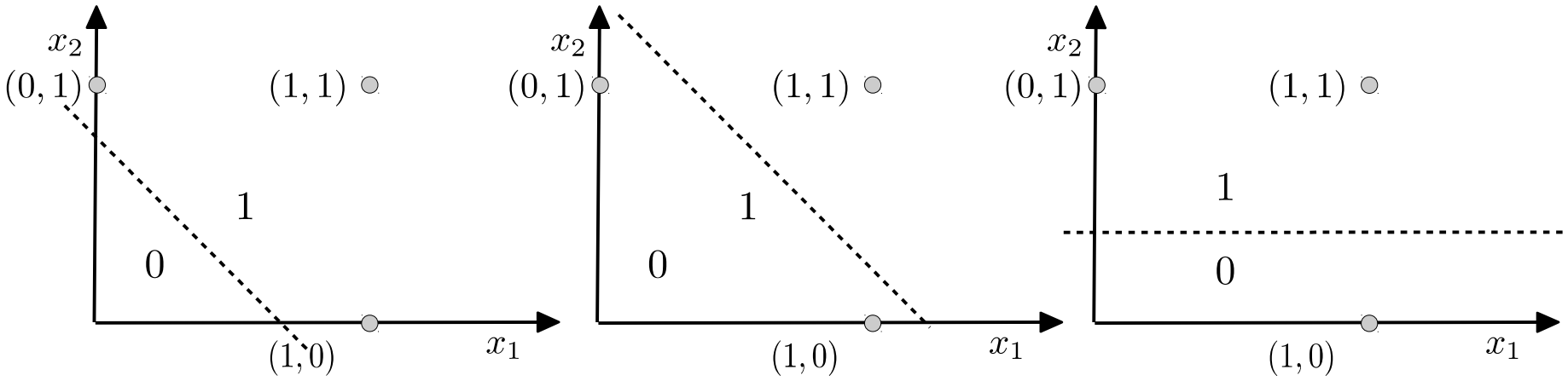


$$y = \begin{cases} 1 & \text{if } \sum_i^2 w_i x_i - T > 0 \\ 0 & \text{else} \end{cases}$$

Das Perceptron übernimmt hier die Aufgabe der linearen Klassifizierung.

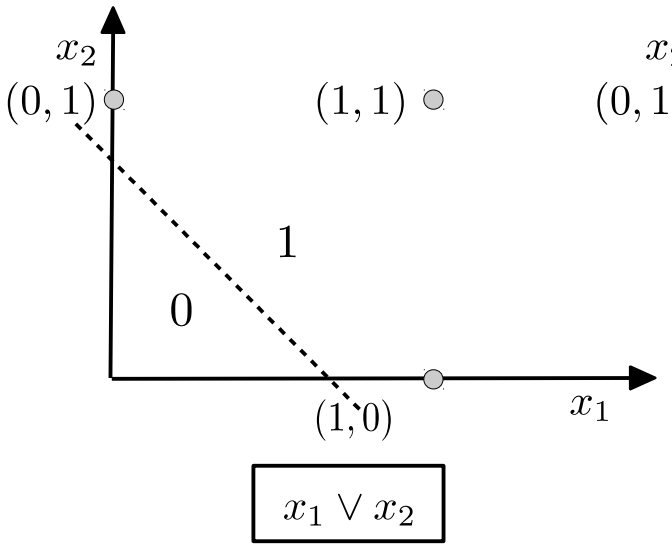
Boolsche Logik – revisited –

- Dieser Logik gemäß, welche boolschen Funktionen sind im folgenden abgebildet:

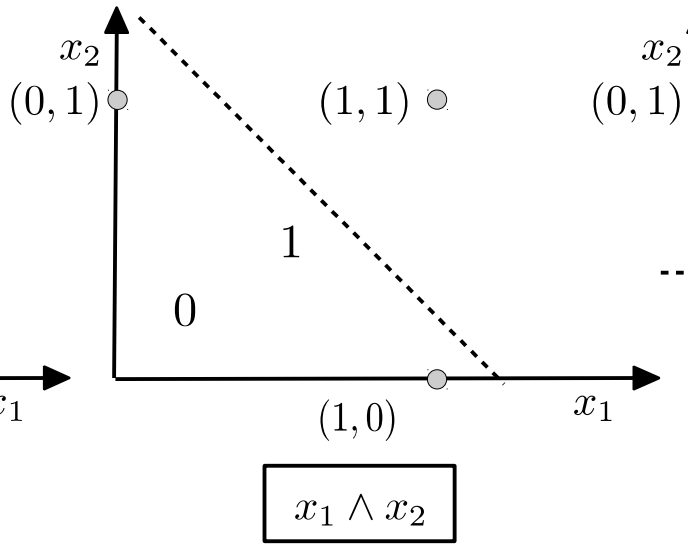


Boolsche Logik – revisited –

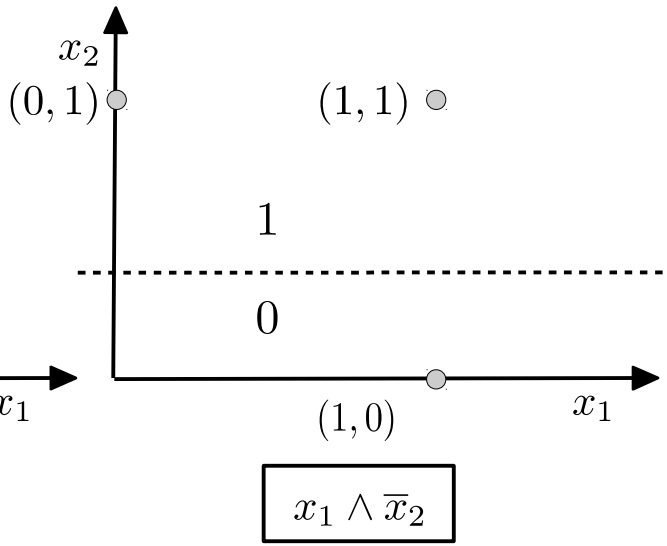
- Dieser Logik gemäß, welche boolschen Funktionen sind im folgenden abgebildet:



$x_1 \vee x_2$



$x_1 \wedge x_2$

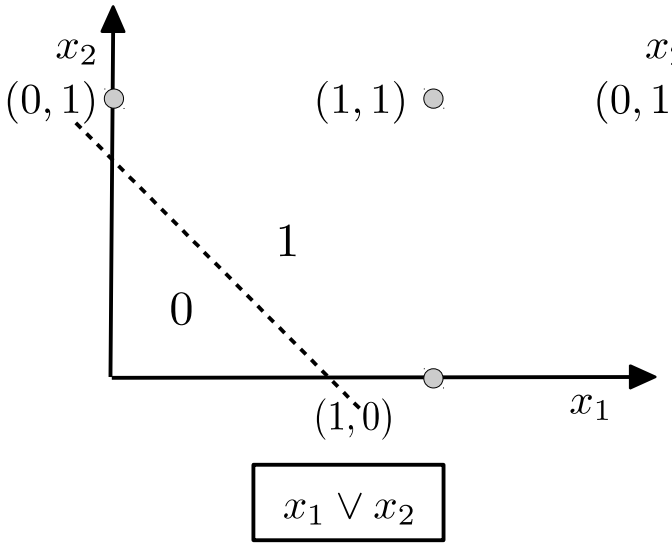


$x_1 \wedge \overline{x}_2$

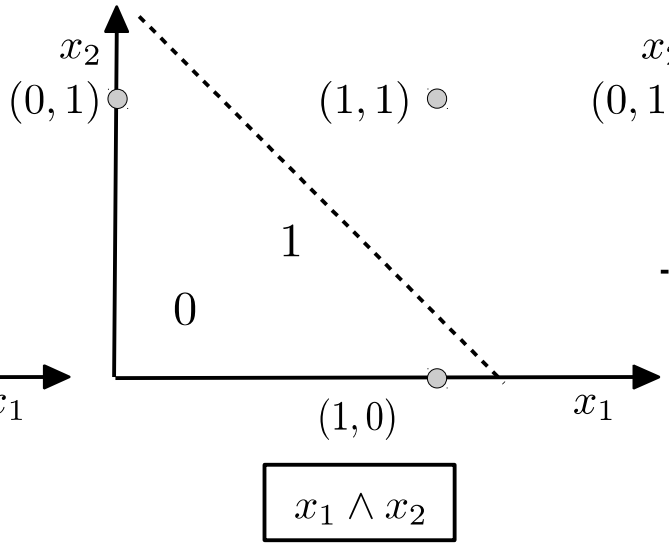
- Warum lässt sich mit diesen Recheneinheiten kein „XOR“ darstellen?

Boolsche Logik – revisited –

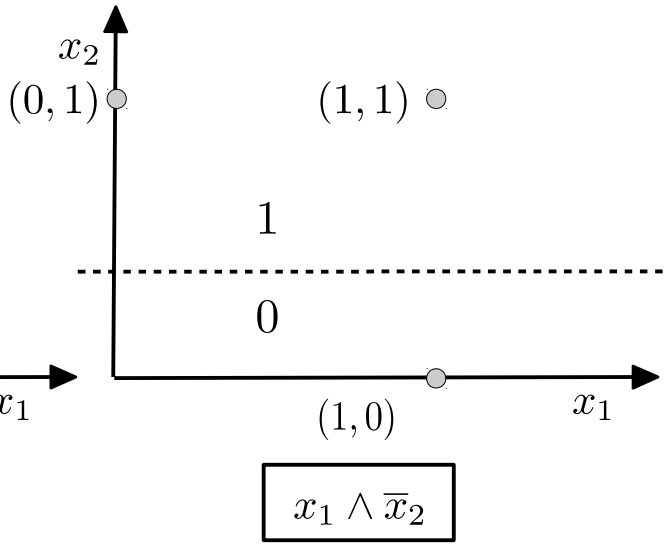
- Dieser Logik gemäß, welche boolschen Funktionen sind im folgenden abgebildet:



$x_1 \vee x_2$



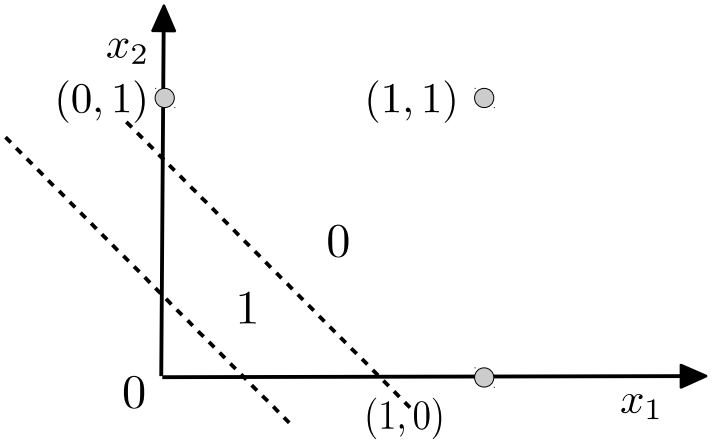
$x_1 \wedge x_2$



$x_1 \wedge \overline{x}_2$

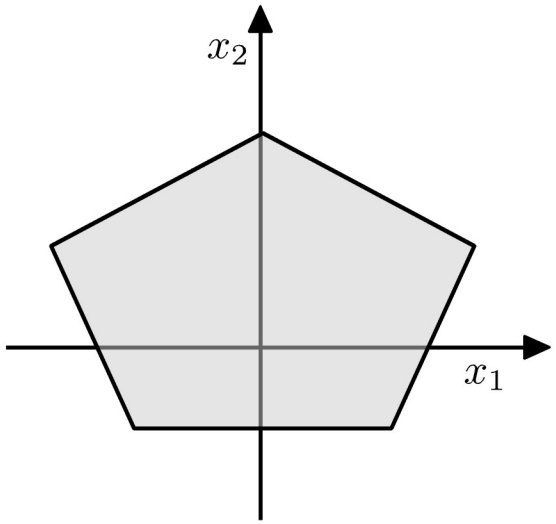
- Warum lässt sich mit diesen Recheneinheiten kein „XOR“ darstellen?

- Ein „XOR“ erfordert eine solche Darstellung. Das ist mit einem einfachen boolschen Perceptron nicht möglich, weil es eben nur <u>einer</u> Linie im $x_1 x_2$ Raum entspricht.



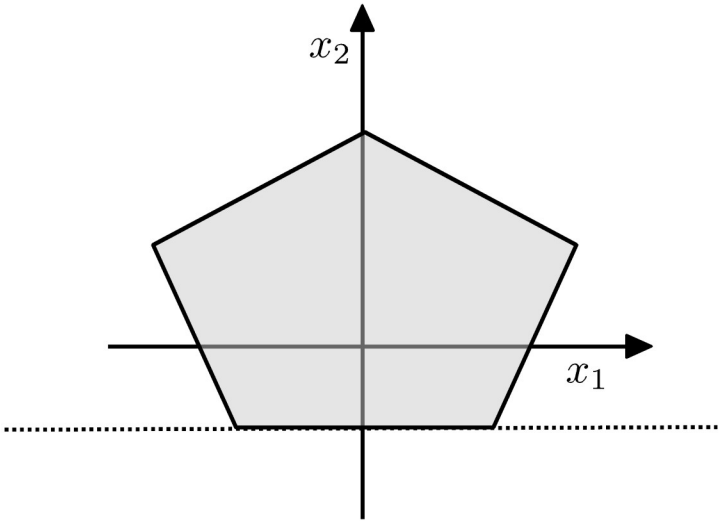
Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

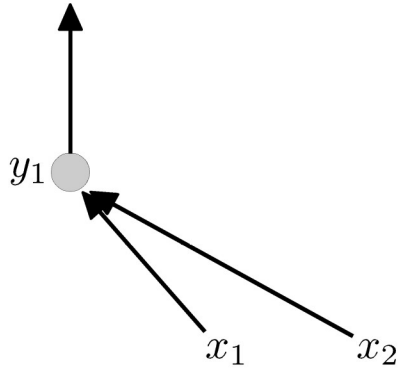
- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?



Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

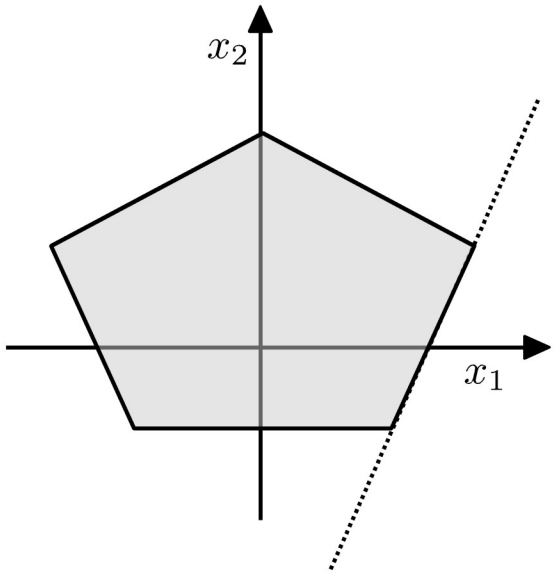
- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?

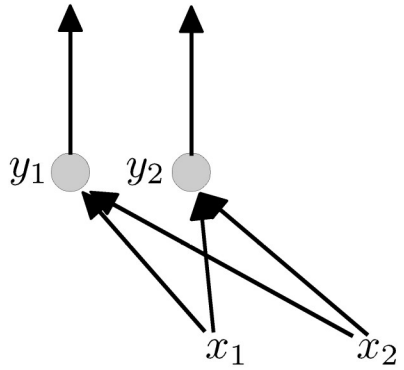




Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

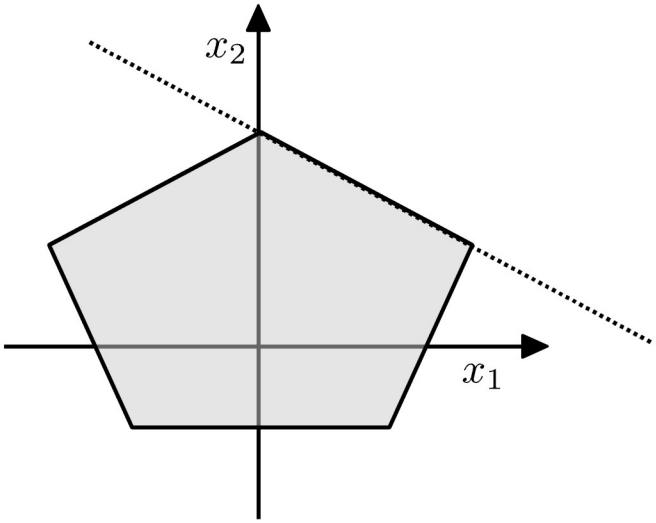
- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?

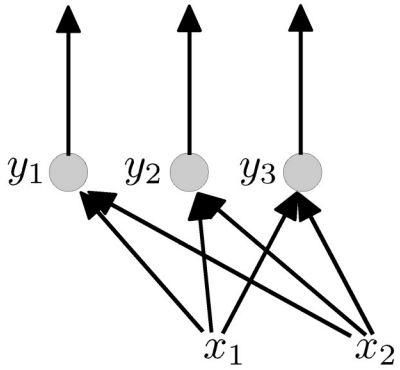




Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

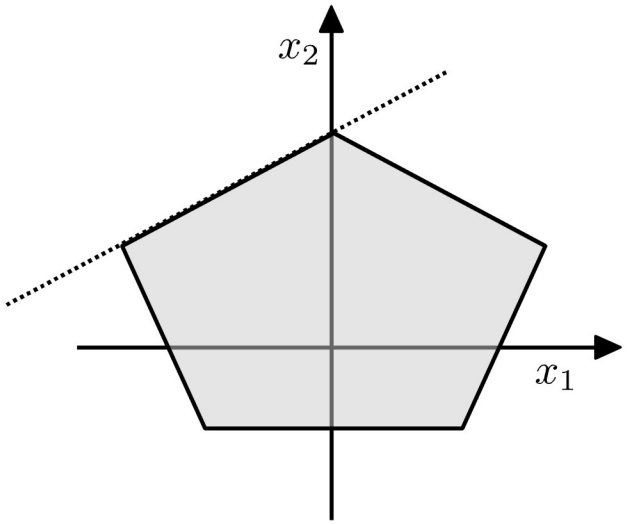
- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?

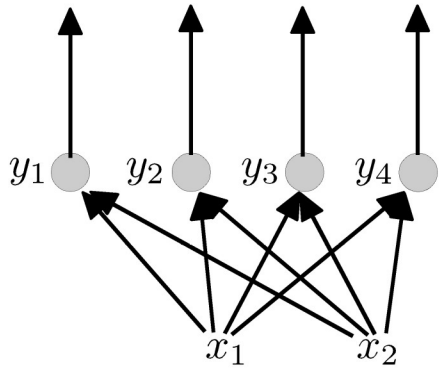




Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

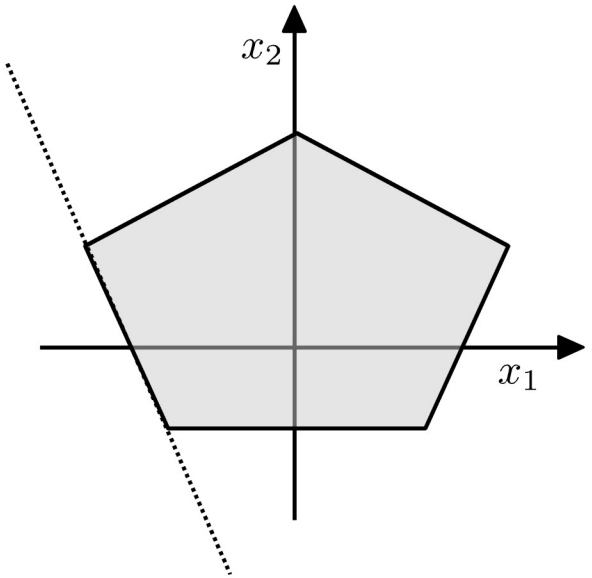
- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?

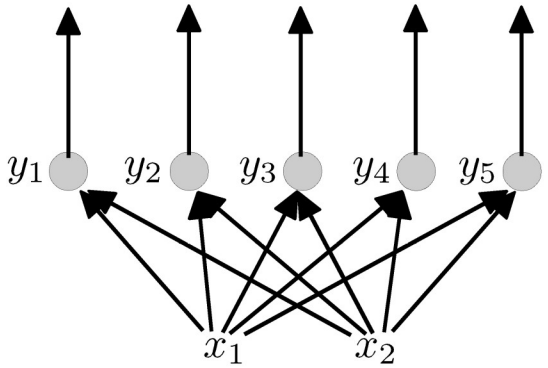




Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

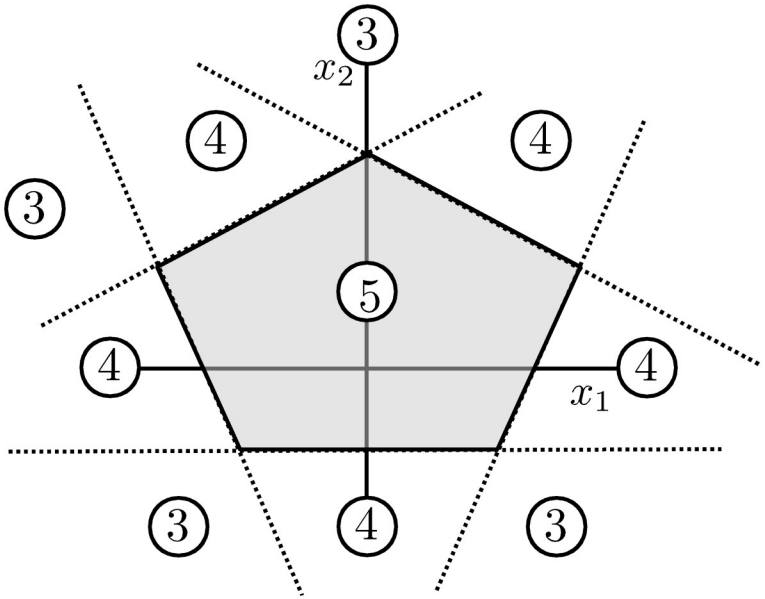
- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?

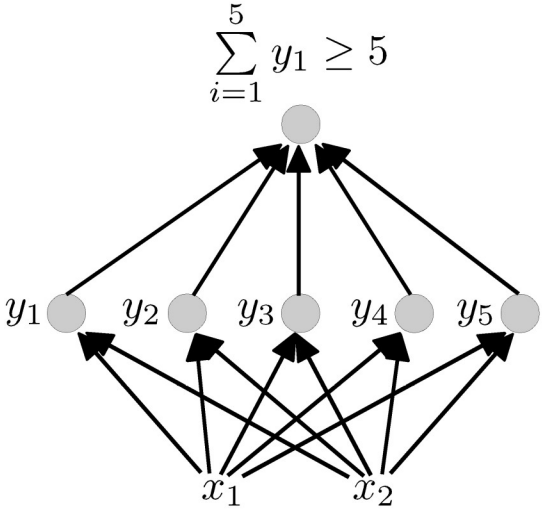




Komplizierte Grenzflächen mit Hilfe boolscher Perceptrons

- Wie würden Sie diese Figur mit Hilfe boolscher Perceptrons abbilden?

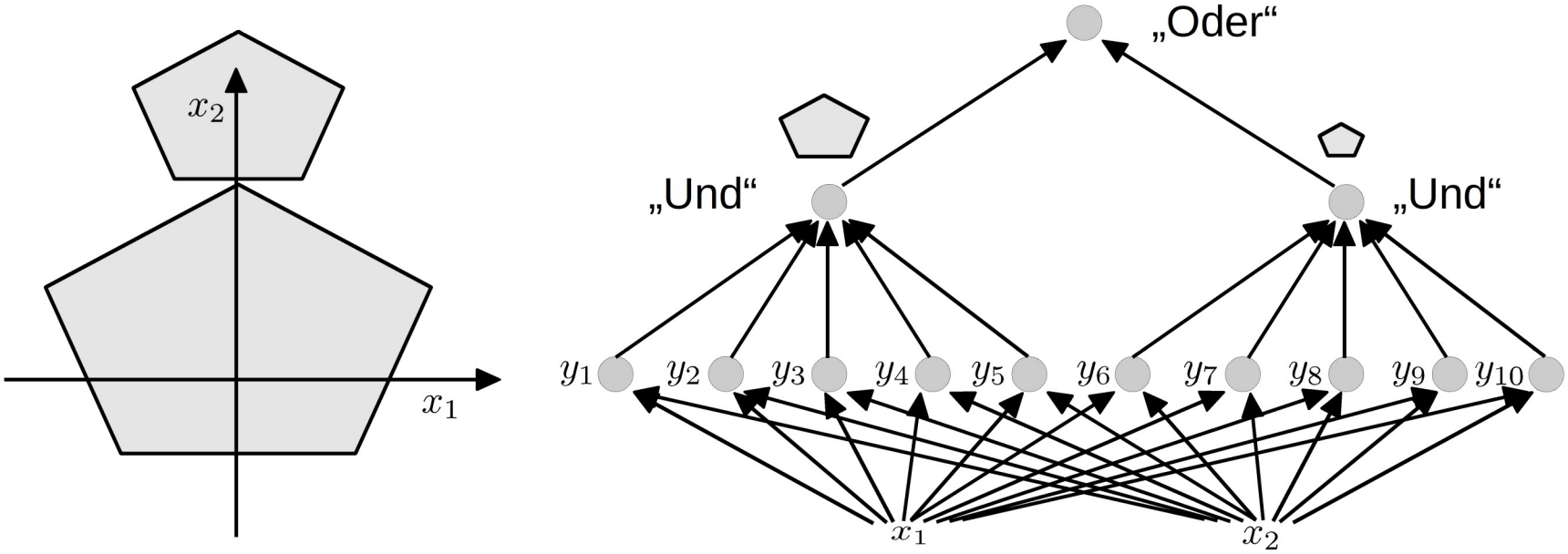




- Normiere den output jeder Recheneinheit y_i auf 1 und addiere.
- Wähle als Schwelle 5.

Noch komplizierte Grenzflächen

- Diese Abbildung würde in unserem Beispiel eine dritte Lage erfordern:

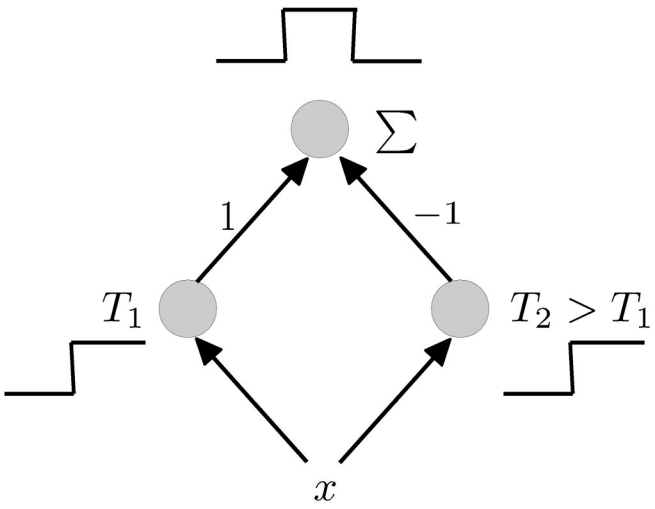


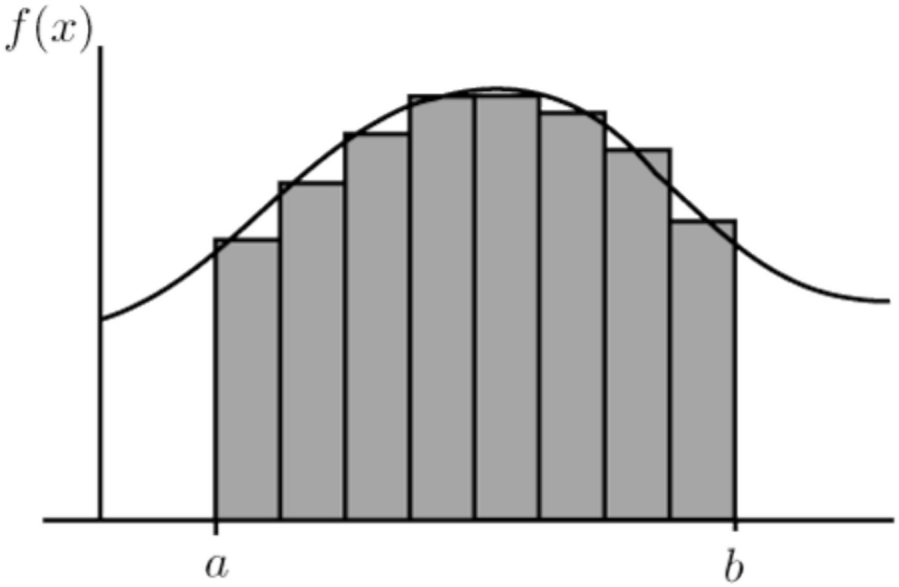
- Da jede beliebige Grenzfläche durch Polygone approximiert werden kann ist es so möglich mit einem hinreichend komplizierten Netzwerk jede beliebige Grenzfläche in einem Hyper-raum zu approximieren.

- Damit ist ein multilayer Perceptron in der Lage jede Form von Klassifikation durchzuführen, weil es jede Form in einem Hyperraum beliebig gut approximieren kann.

Beispiel: Approximation einer beliebigen Funktionen

- Die folgende Einheit demonstriert das Prinzip einer Stufenfunktion:





- Mit einer Gruppe solcher Recheneinheiten ist es möglich jede beliebige Funktion zu approximieren.

Backup