

Regressionsergebnisse interpretieren

- **Angepasstes R-Quadrat:** Mass für die Modellgüte (0 bis 1, höher = bessere Anpassung).
- **Koeffizienten:** Beschreiben den Einfluss der unabhängigen Variablen; konstantes Glied = Y-Achsenabschnitt.
- **Standardfehler:** Gibt Genauigkeit der Schätzungen an (niedriger = präziser).
- **p-Wert:** Zeigt Signifikanz (unter 0,05 = signifikant).
- **Konfidenzintervall:** Bereich, in dem Koeffizienten mit 95 % Wahrscheinlichkeit liegen.
- **Methode (OLS):** Minimiert Abweichungen durch „Least Squares“.

lm.predict() sagt den Wert der abhängigen Variablen (y) basierend auf dem angepassten linearen Modell voraus.

Unsupervised Learning (Classical ML)

Die Maschine versucht ohne vorgegebene Labels oder Anleitung eigenständig Muster in den Daten (z. B. in Tierfotos) zu erkennen und versucht, Unterschiede oder Gemeinsamkeiten zu finden.

Das ist langsamer, als wenn man Supervised Learning betreibt.

Clustering (K-means, Mean-Shift, DBSCAN)

teilt Objekte basierend auf **unbekannten** Merkmalen; die Maschine wählt dabei selbst die beste Methode.

Verwendung: Bildkompression; Erkennung abnormales Verhalten; Marktsegmentierung

Clustering ist also Klassifizierung ohne vordefinierte Klassen.

Es werden zuerst Merkmale **gesucht** und nach diesen dann zusammengefasst.



Beispiel: Anomalien in Userverhalten feststellen.

Die Maschine weiss zuerst nicht, was «normales» Verhalten ist, dann aber aufgrund aller Daten und Verhalten der ganzen User kann die Maschine analysieren was als üblich gilt und die «abnormalen» aussortieren. Vielleicht ist es ein Bot. -> Muss dann von Menschen gecheckt werden.

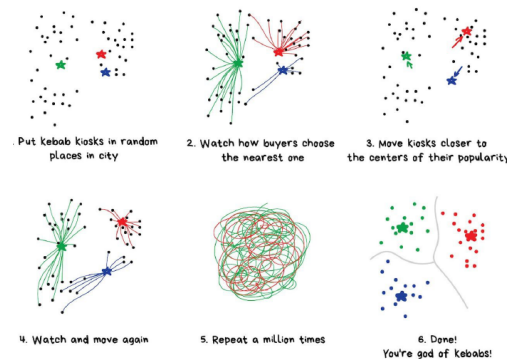
K-Means-Clustering

Klassifiziert Daten, ohne zuvor mit labeled Daten trainiert zu sein. Wird in Suchmaschinen, Astronomie, Computer Vision und Kundenprofilierung benutzt.

Schritt für Schritt:

1. K-Wert setzen (Anzahl der Klassen, Centroids) und irgendwo in die Landschaft setzen
2. Jeden Punkt im Koordinatensystem dem nahegelegensten Centroid zuweisen.
3. Centroid zentralisieren, indem man den Durchschnitt der zugewiesenen Punkte berechnet.
4. Schritt 2 und 3 fortlaufend wiederholen, bis der Centroid sich nicht mehr bewegt.

PUT KEBAB KIOSKS IN THE OPTIMAL WAY
(also illustrating the K-means method)



Es wird ein grosser Wert daraufgelegt, dass du die richtige Menge an Centroids / Clusters wählst.

WCSS (Within Cluster Sum of Squares)

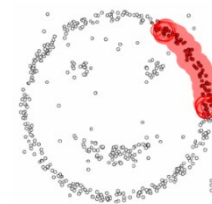
misst die Summe der quadrierten Abstände aller Datenpunkte in einem Cluster zu dessen Zentrum und zeigt, wie gut die Punkte innerhalb der Cluster gruppiert sind.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

in der Praxis sind Cluster oft nicht kreisförmig; sie können **komplex geformt, verschachtelt** sein und eine unbekannte Anzahl haben, was z. B. bei der Suche nach ähnlichen Mineralien auf einer Karte für Geologen eine Herausforderung darstellt. Die Definierung der Anzahl von Centroids wäre sehr zeitaufwendig.

Beispiel DBSCAN:

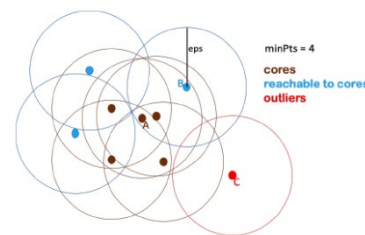
Menschen auf einem Marktplatz. Finde drei nahe beieinanderstehende Personen und lass sie sich an den Händen halten. Diese greifen dann nach weiteren Nachbarn, die sie erreichen können, und so weiter, bis niemand mehr hinzugefügt werden kann. Das ergibt das erste Cluster. Wiederhole den Prozess, bis alle Personen gruppiert sind.



DBSCAN benötigt **keine** feste Clusterform, sondern identifiziert dichte Regionen.

Parameter in DBSCAN

- **eps:** Der maximale Abstand zwischen zwei Proben, damit eine als mit der anderen verbunden betrachtet wird. Der Abstand kann durch jede Art von Distanzfunktion definiert werden, zum Beispiel die „euklidische Distanz“.
- **Core Samples:** Die Proben, die sich in Bereichen hoher Dichte befinden.
- **minPts:** Die minimale Anzahl von Proben in einer Nachbarschaft, damit ein Punkt als Kernpunkt betrachtet wird.



Ein Punkt ist ein Corepoint, wenn mindestens 4 (was wir hier als minPts bezeichnet haben) Punkte (selbst miteingeschlossen) innerhalb der eps liegen.

Vorteile von DBSCAN

- Erfordert nicht, dass die Anzahl der Cluster im Voraus festgelegt wird.
- Ist nicht empfindlich gegenüber Ausreissern.
- Die Cluster können beliebige Formen annehmen, was DBSCAN robust gegenüber verschiedenen Datentypen macht.

Ein Nachteil von DBSCAN ist, dass es bei stark variierenden Dichten zwischen den Clustern schwierig wird und der Parameter **eps** ohne Domain-Wissen schwer festzulegen ist.

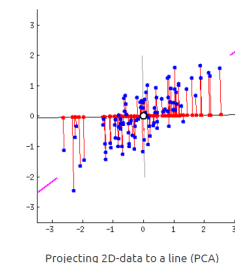
Dimensionality Reduction (Generalization)

fügt spezifische Merkmale zu allgemeineren, höherwertigen Merkmalen zusammen.

Use Case Generalization (PCA, SVD, LDa, LSA...)

personalisierte Vorschläge machen, wie z.B. ansprechende Visualisierungen erstellen, ähnliche Dokumente finden, gefälschte Bilder erkennen und Risikomanagementstrategien empfehlen.

Auch verwendet, um **wichtiges in riesigen Datenmengen** zu finden.

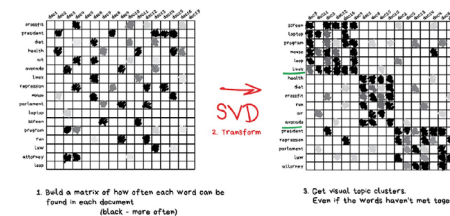


Beispiel Shepherds:

Ein Wort: **Abstraktion**

Es ist oft praktischer, Merkmale zu abstrahieren, indem man beispielsweise Hunde mit dreieckigen Ohren, langen Nasen und grossen Schwänzen zu einer nützlichen Abstraktion wie „Shepherds“ zusammenfasst, da dies die Erklärung vereinfacht, das Modell schneller lernt, weniger überanpasst und weniger Merkmale benötigt. **SVD**

SEPARATE DOCUMENTS BY TOPIC



Topic Modeling with Latent Semantic Analysis

Es geht darum, dass man mit LSA, Themen aus Texten extrahieren kann. Es gibt so viel unstrukturierten Text auf dieser Welt. Um das alles zu analysieren, gibt es diesen automatisierten Weg.

Definition Topics

Ein Topic ist ein Überbegriff für eine Sammlung von Wörtern, die eng miteinander verbunden sind.

LSA-Ablauf

1. Convert raw text into a document-term matrix

Bevor Themen aus Dokumenten abgeleitet werden können, muss der Text in eine Dokument-Term-Matrix umgewandelt werden. Dies geschieht häufig mit dem Bag of Words oder dem TF-IDF-Algorithmus.

2. Implement truncated singular value decomposition

Diese Operation ist der Schlüssel zur Ermittlung von Themen aus einer gegebenen Sammlung von Dokumenten.

Mathematisch:

$$A_{term} = U_{term} S_{term} V_{term}^T$$

A: Die ursprüngliche Document-Word-Matrix, die die Häufigkeit der Wörter (Term) in den Dokumenten darstellt, mit Dimensionen $n * m$ (n = Anzahl der Dokumente, m = Anzahl der Wörter).

U: document-topic Matrix: Zeigt die Stärke der Beziehung zwischen jedem Dokument und den abgeleiteten Themen, mit Dimensionen $n * r$ (n = Dokumente, r = Themen).

S: Single Value Matrix: Eine diagonale Matrix, die die „Bedeutung“ oder Stärke jedes Themas innerhalb der Dokumentensammlung angibt, mit Dimensionen $r * r$ (r = Anzahl der Themen).

V: word-topic Matrix: Zeigt die Stärke der Beziehung zwischen jedem Wort und den abgeleiteten Themen, mit Dimensionen $m * r$ (m = Wörter, r = Themen).

3. Encode the words/documents with the derived topics

Durch die SVD-Zerlegung wird die Dokument-Wort-Matrix in zwei kleineren Matrizen; der Dokument-Themen-Matrix (U) und der Wort-Themen-Matrix (V) umgewandelt, wodurch sich die Zugehörigkeit von Wörtern zu Themen und Dokumenten zu Themen bestimmen lässt.

LSA-Limitationen

Auch wenn es schnell übergreifende Themen eines Dokuments aufdecken kann, hat es seine Nachteile. **Informationsverlust** ist bei der Verwendung von LSA **unvermeidbar**.

Es kann zudem Wörter mit **mehreren Bedeutungen nicht erkennen** und diese anhand ihres Gebrauchs im Text unterscheiden.

LSA fehlt es an **Interpretierbarkeit**, dass heisst, dass die abgeleiteten Themen und Beziehungen oft sehr verständlich oder erklärbar sind.

Case Study

Sagt eigentlich, dass man bevor man das LSA macht, dass man alle Wörter in kleinbuchstaben umwandeln soll.

Association rule learning (Apriori, Euclat, FP-Growth)

Der Algorithmus sucht im Bestellstrom nach Mustern, indem er häufige Bestellungen oder Produkte identifiziert, die gemeinsam auftreten, um Assoziationsregeln zu entdecken und meistens dann auch vorzuschlagen (z.B. „Kunden, die Produkt A kaufen, kaufen auch oft Produkt B“).

- **Verkäufe und Rabatte prognostizieren**
- **Analyse der gemeinsam gekauften Waren**
- **Platzierung der Produkte in den Regalen**
- **Analyse des Surfverhaltens im Internet**

Beispiel im Supermarkt:

Ein Mensch geht in den Supermarkt, um Bier zu kaufen, warum platziert man nicht auf dem Weg vom Bier zur Kasse noch Nüsse, denn viele Menschen kaufen beides zusammen.

Apriori

Vorhin erklärt im Beispiel im Supermarkt, wird Association Rule Learning, hier die Methode Apriori verwendet um versteckte Beziehungen zwischen verschiedenen Produkten / Sachen zu identifizieren.

Apriori Konzept (Bottom-Up-Approach)

fängt mit den kleinsten Elementen (einzelnen Items) an und arbeitet sich Schritt für Schritt zu grösseren Kombinationen hin.

Alle nicht-leeren Teilmengen eines frequenten Itemsets sind ebenfalls frequent.

Apriori sucht zuerst nach einzelnen Items, die oft in Transaktionen vorkommen (z. B. Brot, Milch). Das nennt man frequente Itemsets.

Support

Der Anteil der Transaktionen, die ein bestimmtes Item enthalten.

$$support(I) = \frac{\text{Number of transactions containing } I}{\text{Total number of transactions}}$$

Confidence

Gibt an, wie oft Items in Y in Transaktionen mit X vorkommen.

$$confidence(X \rightarrow Y) = \frac{\text{Number of transactions containing } X \text{ and } Y}{\text{Number of transactions containing } X}$$

Frequent Item Set

Ein Itemset, dessen Support einen definierten Mindestwert (minSup) erreicht oder überschreitet.

Nutze k-1-Itemsets, um k-Itemsets zu generieren:

Baue grössere Itemsets schrittweise aus kleineren frequenten Itemsets auf.

Erzeuge C[k], indem L[k-1] mit L[k-1] kombiniert wird: Kombiniere frequenten Itemsets der Grösse k-1, um Kandidaten-Itemsets der Grösse k zu erstellen.

Prüne C[k] durch Subset-Tests: Entferne Kandidaten aus C[k] C[k] C[k], deren Teilmengen nicht frequent sind, basierend auf dem Apriori-Prinzip.

Generiere L[k], indem die Itemsets in C[k]

gefiltert werden: Extrahiere die Itemsets aus C[k], die die Mindestunterstützung (minSup) erfüllen.

Apriori Nachteile / Shortcomings

Apriori ist einfach zu verstehen und umzusetzen, wird aber bei grossen oder komplexen Datensätzen ineffizient.

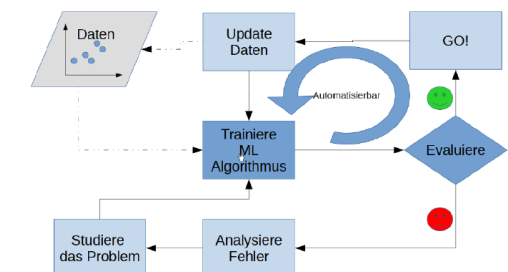
- Hoher Rechenaufwand
- Viele Datenbankzugriffe (Ganze Datenbank für jede Iteration)

Verbesserung von Apriori

Es gibt viele zusätzliche Techniken, die angewendet werden können, um die Effizienz des Apriori-Algorithmus zu verbessern. Einige davon sind:

- **Hashing:** Reduzierung der Datenbankdurchläufe durch Hash-Tabellen.
- **Transaktionsreduktion:** Entfernen von Transaktionen, die keine frequenten Itemsets enthalten, um sie bei weiteren Berechnungen zu ignorieren.
- **Partitionierung:** Ein Itemset, das möglicherweise frequent ist, muss in mindestens einer Partition frequent sein.
- **Dynamisches Zählen von Itemsets:** Verringert die Anzahl der Durchläufe über die Daten.
- **Sampling:** Verwendung von zufälligen Stichproben, um die Datenmenge zu verkleinern und die Berechnungen zu beschleunigen.

Reinforcement Learning (Q-Learning, SARSA, DQN, A3C, Generic algorithm)



«Einen Roboter in ein Labyrinth schmeissen und ihn selbst den Ausgang finden lassen»

Reinforcement Learning wird in Fällen eingesetzt, in denen das Problem gar nicht mit Daten zu tun hat, sondern mit einer Umgebung, in der man lebt.

Use Cases:

- Selbstfahrende Autos
- Roboter-Staubsauger
- Spiele
- Automatisiertes Trading
- Verwaltung von Unternehmensressourcen

Reinforcement Learning kann nicht alle möglichen Situationen voraussagen, sein Ziel ist es, Fehler zu mindern. Überleben ist ein Grundsatz und eine Inspirationsidee dieser Methode.

Model-Based RL

Die KI lernt ein Modell der Umgebung oder hat bereits Zugriff darauf, und um **zukünftige Zustände** und Belohnungen zu simulieren.

Sie plant Aktionen basierend auf diesen Simulationen (z. B. mithilfe von dynamischer Programmierung oder Suchalgorithmen).

Model-Free RL

Die KI benötigt kein Modell der Umgebung. Er lernt direkt durch Interaktionen mit der Umgebung.

Sie führt Aktionen aus und sammelt **Belohnungen** aus der Umgebung und das Milliardenfach. Es merkt sich welche **Aktionen langfristig** zu den höchsten Belohnungen führen, ohne die Umgebung explizit zu modellieren.

FP Growth (Improved Apriori)

effizienter Algorithmus, der zur Suche nach frequenten Itemsets in grossen Datensätzen verwendet wird, ohne dass Kandidaten-Itemsets wie im Apriori-Algorithmus explizit generiert werden müssen

- Der Algorithmus erstellt einen FP-Baum (Frequent Pattern Tree), eine kompakte Datenstruktur.

- Dabei werden häufige Items (basierend auf Mindest-Support) in einer bestimmten Reihenfolge gespeichert, um Redundanz zu minimieren.

Ensemble Methods (Random Forest, Gradient Boostin)

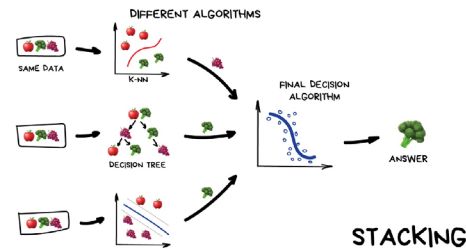
Viele Bäume, die lernen, ihre Fehler gegenseitig zu beheben.

Use Cases:

- Suchsysteme
- Computer Vision
- Objekterkennung

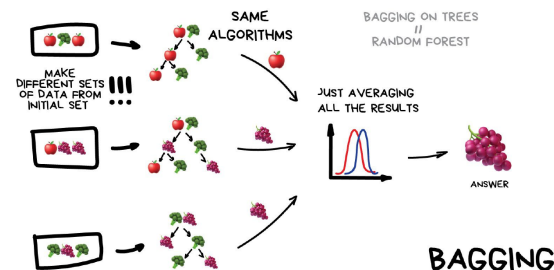
Geht nach dem Prinzip, dass man ineffiziente Algorithmen verwendet und sie dann so darauf abstimmt, dass sie sich selbst korrigieren. Die Qualität des Systems wird besser sein als jeder individuelle Algorithmus.

Stacking -> Verschiedene parallellaufende Algorithmen als Eingabe benutzen, um eine endgültige Antwort zu erhalten. Oftmals eine Regression eine gute Wahl.



Bagging (Bootstrap Aggregating)

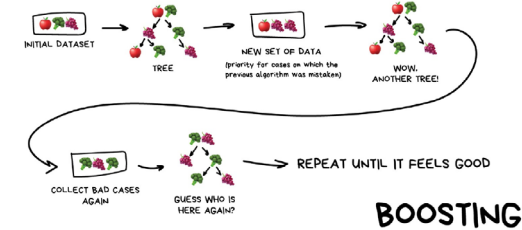
Die bewährteste Methode von Bagging ist wohl der Random Forest, was einfach Bagging mit Decision Trees macht. Immer andere Teilmengen der Daten trainieren und dann zum Mittelwert kombinieren.



Bei einer iPhone Kamera die gelben Boxen sind das Produkt von Bagging.

Boosting (Tools: CatBoost, LightGBM, XGBoost)

Boosting ist eine Methode, bei der mehrere einfache Modelle nacheinander trainiert werden, und jedes Modell lernt aus den Fehlern des vorherigen, um am Ende ein besseres Gesamtergebnis zu bekommen. Wird 1 für 1 trainiert.



Hohe Präzision von Klassifizierung. Nachteil: Es geschieht schrittweise.

Deep Learning und Neuronale Netzwerke

«Wenn man zu viele IT-Ressourcen hat, aber nicht weiss was mit ihnen anfangen soll.»

Berühmte Architekturen

- Perceptron
- Convolutional Network (CNN)
- Recurrent Networks (RNN)
- Autoencoders

Use Cases:

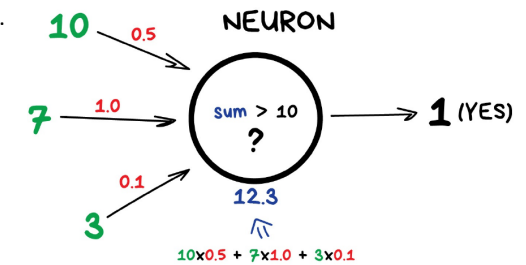
- Alle oben genannten Methoden ersetzen
- Objekterkennung in Fotos oder Videos
- Spracherkennung und Voice Generation
- Bildverarbeitung
- Maschinenübersetzung

Neuronales Netzwerk

Ein neuronales Netzwerk ist im Wesentlichen eine Ansammlung von Neuronen und Verbindungen zwischen ihnen. Ein Neuron ist eine Funktion mit mehreren Eingängen und einem Ausgang.

Seine Aufgabe besteht darin, alle Zahlen von seinen Eingängen zu verarbeiten, eine Funktion darauf anzuwenden und das Ergebnis an den Ausgang zu senden.

Die Neuronen eines Netzwerks sind schichtweise organisiert, wobei Daten nur von den Eingängen zur Ausgabe fließen, ohne dass Verbindungen innerhalb einer Schicht bestehen.



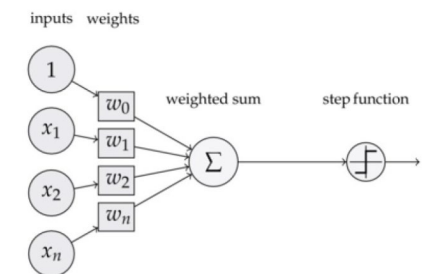
Neuron: Eine Funktion mit mehreren Eingaben und einer Ausgabe.

Verbindungen: Kanäle zwischen Neuronen.

Perceptron

Ein Netzwerk mit mehreren Schichten, bei dem jede Neuron Ebene vollständig mit jeder anderen verbunden ist, wird als Perceptron (MLP, Multi-Layer Perceptron) bezeichnet und gilt als die einfachste Architektur.

Ein Neuron – Graphische vs. Mathematische Darstellung



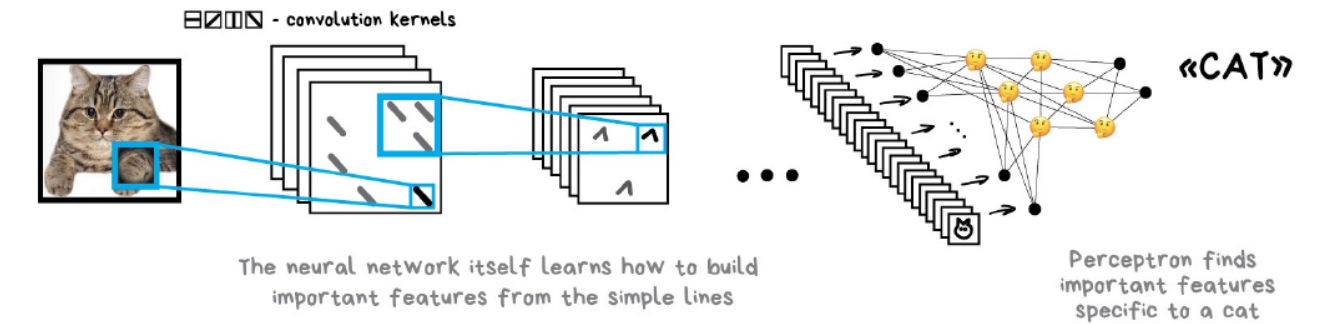
Hebb'sche Lernregel

Wenn zwei Neuronen sehr aktiv sind, also ein starkes Ausgangssignal haben, verstärkt sich auch die Verbindung zwischen den beiden Neuronen.

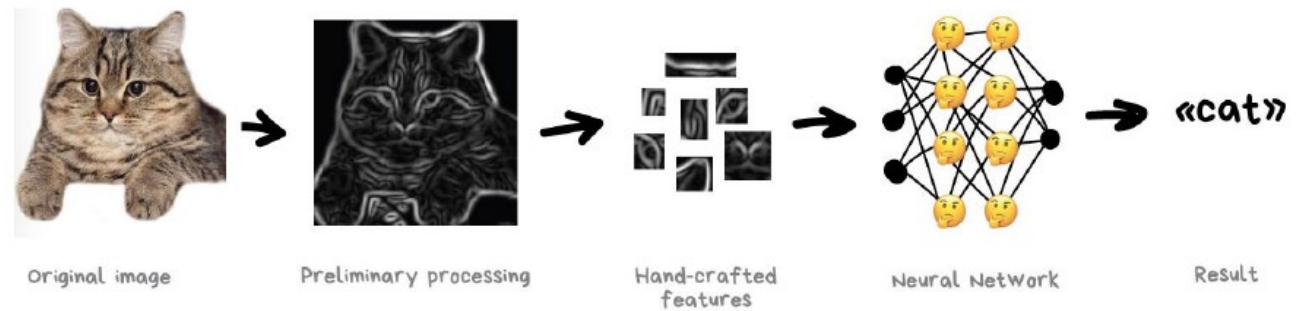
Backpropagation: 1. Gewichte werden mit Zufallswerten initialisiert. Für jedes Trainingsbeispiel: I. Komplettes Netz von den Eingängen zu den Ausgängen berechnen. II. Abweichung zwischen Soll- und Ist Ausgangswerten ermitteln und Fehler rückwärts durch das Netz propagieren. Man sagt jedem Neuron, was er falsch gemacht hat und wie er es richtig machen sollte. Zusätzliche verborgene Schichten bringen ab und zu keinen nennenswerten Fortschritt.

Convolutional Neural Networks (CNN):

Benutzt für: Suche von Objekten auf Fotos und Videos, Gesichtserkennung, Style Transfer, Bildgenerierung, Effekterstellung. Alles, was mit Bild und Video zu tun hat. **Problem bei Bildern:** Extraktion von Features. Bilder müssen manuell gelabelt werden, wo z.B. Ohren sind. **Genannt:** Handcrafting Features. **Lösung:** Die Maschine muss die Features selbst lernen (genannt Convolution, was eine Ebene eines NN abbildet, weil jedes Neuron als jede Funktion fungieren kann). Wenn wir das Netzwerk mit vielen Katzenfotos füttern, werden die Gewichte für die gesehenen Kombinationen grösser. **Ausgabe:** Perceptron mit den am meisten aktivierten Kombinationen. **Schönheit:** Das Neuronale Netzwerk sucht die Features von selbst.



<https://i.vas3k.ru/7wj.jpg>



Recurrent Neural Networks (RNN):

Benutzt für: Sequenzielle Daten wie Stimme, Text oder Musik. Moderne Sprachausgabe spricht nicht Buchstabe für Buchstabe aus, sondern ganze Sätze auf einmal. **Funktionsweise:** Man hat Text als Eingabe, und erwartet Ton als Ausgabe. Audio wird generiert, mit Original verglichen und die Fehler werden korrigiert. **Erinnerung an die vorherigen Töne:** Man fügt jedem Neuron Speicher hinzu und benutzt ihn als zusätzliche Eingabe beim nächsten Durchlauf. Problem: Wenn man alle vorherigen Resultate kennt, wäre das Netzwerk zu gross. **Lösung:** Begrenzung des Neuronspeichers. **Bessere Lösung:** Spezielle Zellen (wie RAM), wo die Neuronen schreiben, lesen und löschen können. Long- and short-term memory cells (LSTM). Das Netzwerk wird trainiert, sowohl die Gewichte als auch die Reminder zu verwenden.

