What is AI

- John McCarthy created the concept of AI
- A machine or a computer program that can solve problems that are usually solved by humans with our natural intelligence.
    - **What is intelligence**?
        - Generalization learning – learning that enables the learner to preform better in situations not previously encountered
        - Reasoning – Draw conclusions appropriate to the situation in hand
        - Problem solving – Given such and such data, find x
        - Perception – analyzing a scanned environment and analyzing features and relationships between objects
            - Self-driving cars
        - Language understanding – understanding language by following syntax and other rules, similar to a human

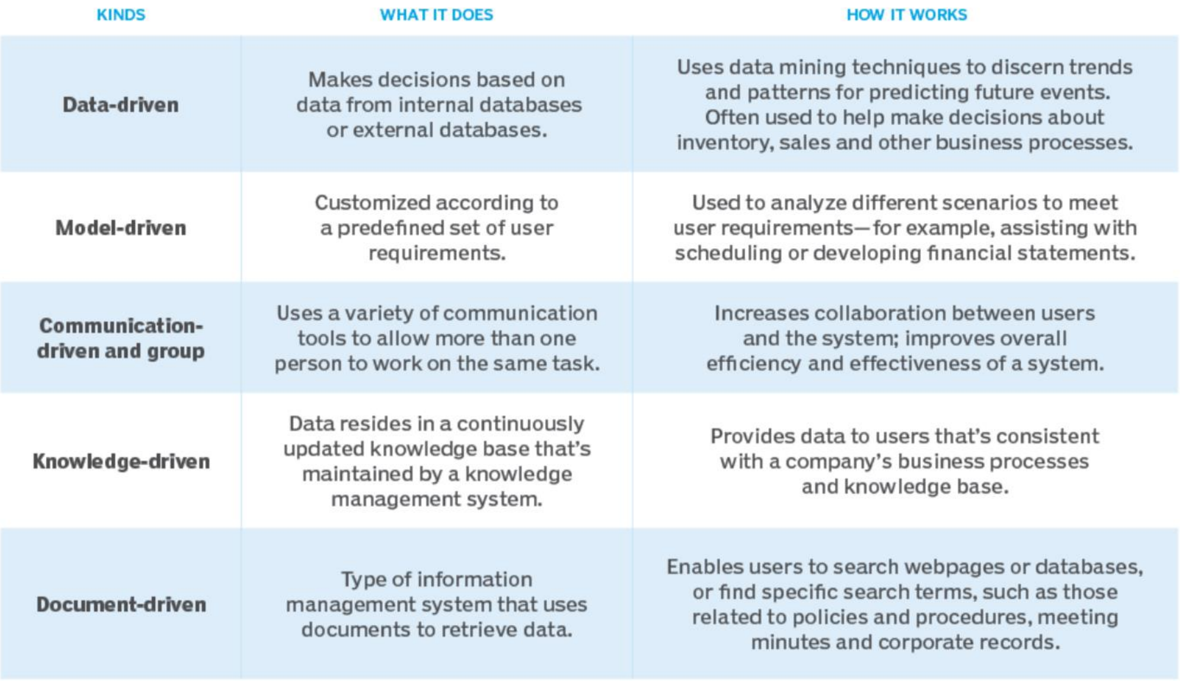
        - Turing test – If a person is unable to distinguish if he is conversing with a computer or person, than it is said that the computer is able to "think".
- **Strong AI** – simulating the human brain by building systems that think and, in the process, give us an insight into how the brain works.
    - **No way near this stage**
- **Weak AI –** system that behaves like a human but doesn't give us an insight to how the brain works
    - Chess playing ai
- Middle ground – Deep learning

Decision-Support system

Decision-Support system – Help draw conclusions, not domain dependent, more general than many other systems. Has the ability to process data, information, and knowledge. Tackle large-scale, time-consuming, complex problems. Make decision process and outcome more reliable. Generate new evidence in decision / in assumptions. Encourage explorations and discovery -> revel new approaches of thinking + Support planning and scheduling.

- **Intelligent system** – umbrella term for all systems making decisions
- **Three levels:**
    - Human-powered, **Passive**, aids the process of decision making, BUT cannot make explicit decisions, nor provide suggestions or solutions.
    - Fully computerized (automatic), **Active**, more sophisticated, can draw conclusions, make decisions, and provide suggestions or solutions.
    - Semi-automatic, **Complex** with cooperation, the decision maker modify, complete, or refine the decision or suggestions provided by the system. OR vise-versa.
- **Categories:**
    - **Communication-driven DSS** – helps conduct meetings
    - **Data-driven DSS –** access and query a database or data warehouse to seek specific answers

  - **Document-driven DSS** – searches, retrieves and manages search web pages and documents on a specific set of keywords or search terms
  - **Knowledge-driven DSS** – provides specialized problem-solving expertise stored as facts, rules, or similar structures.
  - **Model-driven DSS** – access and manipulate statistical, financial, optimization, and simulates models. Model-driven DSS assist in analyzing situations.



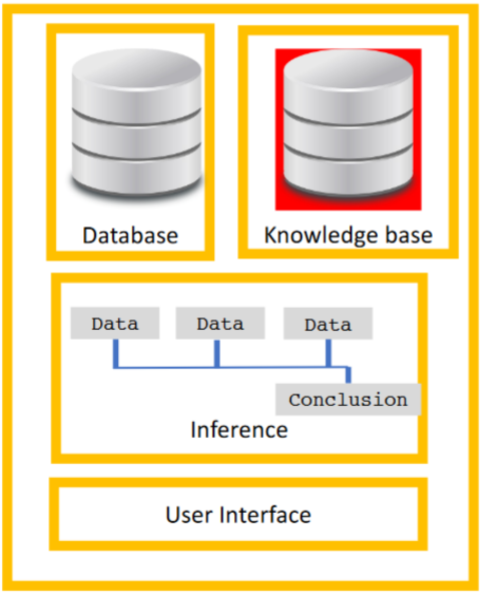
KINDS	WHAT IT DOES	HOW IT WORKS
Data-driven	Makes decisions based on data from internal databases or external databases.	Uses data mining techniques to discern trends and patterns for predicting future events. Often used to help make decisions about inventory, sales and other business processes.
Model-driven	Customized according to a predefined set of user requirements.	Used to analyze different scenarios to meet user requirements—for example, assisting with scheduling or developing financial statements.
Communication-driven and group	Uses a variety of communication tools to allow more than one person to work on the same task.	Increases collaboration between users and the system; improves overall efficiency and effectiveness of a system.
Knowledge-driven	Data resides in a continuously updated knowledge base that's maintained by a knowledge management system.	Provides data to users that's consistent with a company's business processes and knowledge base.
Document-driven	Type of information management system that uses documents to retrieve data.	Enables users to search webpages or databases, or find specific search terms, such as those related to policies and procedures, meeting minutes and corporate records.

- Good for all sort of **decision-making** systems and tools
  - Drawing conclusions
  - Finding the best alternative
  - Planning and scheduling
- **Area of use**:
  - Healthcare
  - Offices and personal homes
  - Self-driving cars and other types of vehicles
  - Robotic Process Automation
- **Contents:**
  - **Storage**
    - A database stored as facts
    - Expands during execution
    - Data comes from user or external computer systems
  - **Search mechanism**
    - Searching for data that are needed for a decision
    - Find data in, internal databases and registers, online, external databases and documents
    - Example: Web crawlers
  - **User interface**
    - Question-and-Answer interface: DSS asks question and user answers

        - Graphical interface: User to give visual commands to DSS
- **Business Intelligence** – a collective term for: skills techniques, applications, process and methods, that support an organization / company to understand its business and the surrounded world.

Expert system – System that employs human knowledge in a computer to solve problems that ordinarily require human expertise within a domain. – Imitate reasoning process – Simulate the decision-making abilities of a human expert.

- Refers to the type of task the system is trying to assist with – to **replace** or aid human expert in a complex task.
- **Contents:**
    - Database -> Facts
        - An organized collection of data. Contains pre-stored data (facts) and satisfied conclusions and advices, expands during execution. Data also comes from drawn conclusions, users and/or external computer systems
    - Knowledge base -> Knowledge representations
        - Used to find facts in the database or ask users about missing facts
        - If – than – else
        - Object-oriented
    - Interface mechanism / interface engine -> Reasoning strategies
        - Interpret the contents of the knowledge base, find conclusions for the content.
        - Based on the logic rule **Modus ponens**: if P implies Q, and P is true, than Q is true.
        - Two main reasoning methods: Forward chaining, Backward chaining
    - User interface
        - Question-and-Answer interface
        - Graphical interface



- **Shells –** everything except the knowledge base

Knowledge-based system – Also working within a domain, but the systems is structured in modules and the knowledge base can be exchanged. – Broad term refers to many kinds of systems.

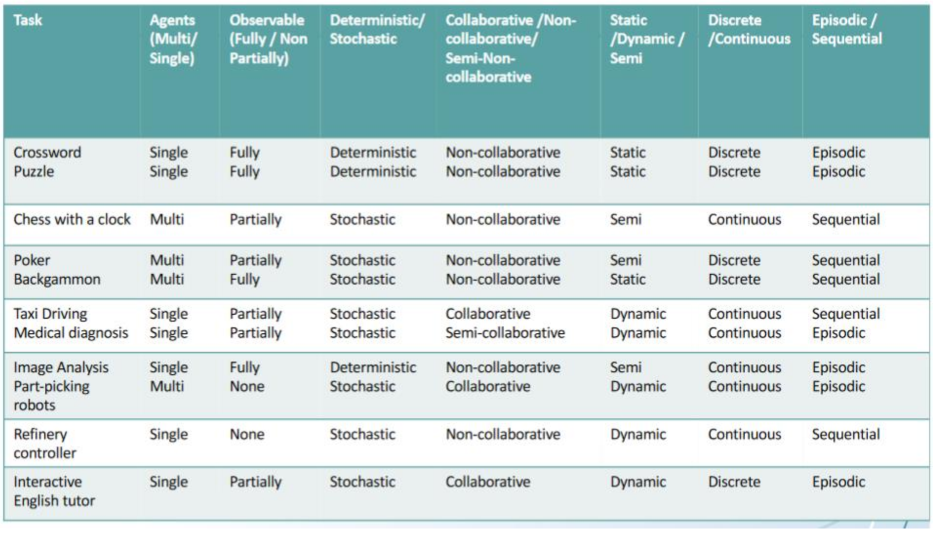
- Refers to the **architecture** of the system – that it represents knowledge explicitly (rather than procedural code).
- **Contents:**
  - Database -> Facts
  - Knowledge base -> Knowledge representations
  - Interface mechanism / interface engine -> Reasoning strategies
  - User interface
- **Shells –** everything except the knowledge base

Multi Agent System – Loosely coupled network of (software or intelligent) agents that interact to solve problems

- A meta-agent for controlling other agents
- **Decentralized –** distributes computational resources, provides capabilities across networks of interconnected agents
    - Prevents **Single point of failure** – if fails, the systems stops executing
- **Advantages**
    - Enhances overall system performance
        - Computational efficiency
        - Coordinates information from distributed sources
        - Allow interconnection and interoperation
- **Disadvantages**
    - Can be slow, especially physical agents
    - Can take time to program, if each agent has own task
    - Communication between agents may be:
        - Asynchronized
        - Carry out task in wrong order
        - Can destroy for each other
- **Terminology**
    - **Performance measure –** the criteria that *determines the successfulness* of an agent
    - **Percept –** Agents *perceptual inputs* at a given instance
    - **Precept sequence –** *history of all percepts* that an agent has perceived until a given date
    - **Behavior of Agent –** The *actions* that agent performs after any given sequence of percepts
    - **A simple agent program (agent function) –** maps every possible percepts sequence to a possible action (that can be performed)
    - **Actuators –** Parts used for acting (on something), legs
- **What kind of agents exists in an agent system**
    - **Software agent**
        - Is situated in an environment, *capable of autonomous actions*
        - A single unit / computer program performing simple tasks – has a repertoire of actions
        - No *intelligence* included (complex strategies)
        - Self-working
        - May interact with people, may get a task from another computer
    - **Intelligent agent –** Gather information or carry out tasks / services *without any interference*
        - An autonomous entity with abilities such as
            - Reactive – can perceive the environment and react to changes and act accordingly
            - Proactive – goal-directive behavior, knows what is next and take initiative
            - Social – capable of interacting with other agents

      - Can learn
    - **Common** intelligent agents
      - **Simple reflex agent –** act only on the current percept. Only succeeds in FULLY observable environments
        - Vacuum cleaner
      - **Model-based reflex agent –** act on the model of "how the world works", use stored structures which describes the part of the world which cannot be seen. Can handle PARTIALLY observable environments
      - **Goal-based agents –** use "goal" information which describes situations that are desirable. Expand the capabilities of model-based agents. Can handle NON-observable environments
  - **Meta-agent –** agents with meta-reasoning ability
    - Monitor/control software/intelligent agents
    - Support providing the fastest, or best, way under given circumstances
    - Handle a vast number of "low-level" agents in a network/graph
  - **Negotiation agents –** negotiates in a particular setting, commonly used for auctions (bid bots) and negotiation for business-to-business (B2B) or business-to-costumer (B2C)
  - **Rational agents**
    - Reasonable, sensible, and have a good sense of judgement
    - Concerned with expected actions and results depending upon what the agent has perceived
    - Performing actions with the aim of obtaining useful outcome
    - Maximize its performance with built-in knowledge base
  - **Beliefs Desire Intentions (BDI) agents** – Intelligent rational agents & A software model/architecture
    - Concepts to solve a particular problem:
      - B – Beliefs about the world
      - D – desires the agent *would like* to accomplish goals
      - I – intentions the agent *has chosen* to do.
    - Planner – an agent with decision-making capabilities (agent system % decision support system)
  - **Mobile agents** (robots)
    - move around in environment
    - capable of performing, appropriately, in the new environment
    - Intelligent agents that observe via sensor or other external means
  - **Autonomous agents** (self-driving cars)
  - **Interface agents, user agents, information agents**
- **Environments –** agents must be programed to handle their environment
  - **Single/Mulit-agent** – Agents handle task individually, number of agents working together to solve a task.
  - **Fully observable, non-observable, partially observable** (also called – accessible – inaccessible)

        - **Fully observable** – obtain complete, accurate, up-to-date information about the environment state
        - **Non-observable –** do not know anything about the environment
        - **Partially observable –** partially know environment
- **Environment cont.**
    - **Deterministic vs. Stochastic**
        - **Deterministic –** Every action has a single guaranteed effect, next state is completely determined by the current state
        - **Stochastic –** multiple, unpredictable outcomes
    - **Collaborative environment –** carry out task together, related agents can be monitored and manipulated.
    - **Non-collaborative –** carry out task, individually
    - **Semi-collaborative –** collaborative environment, with some own work
    - **Static vs. dynamic**
        - **Static –** the environment is assumed to remain unchanged
        - **Dynamic –** can change beyond an agents' control, other processes are operating in the MAS
    - **Discrete vs. Continuous**
        - **Discrete –** fixed and finite number of states and finite number of actions that can be performed and percept in the environment (snake game)
        - **Continuous –** Infinite number of actions, and the environment keep changing (football)
    - **Episodic vs. Sequential** (non-episodic)
        - **Episodic –** one episode at the time, each state is independent. Only consider the task at hand, decides best action, do not think ahead.
        - **Sequential –** next state is dependent on current action. Current action can change all of the future states of the environment.



Task	Agents (Multi/ Single)	Observable (Fully / Non Partially)	Deterministic/ Stochastic	Collaborative /Non-collaborative/ Semi-Non-collaborative	Static /Dynamic / Semi	Discrete /Continuous	Episodic / Sequential
Crossword Puzzle	Single Single	Fully Fully	Deterministic Deterministic	Non-collaborative Non-collaborative	Static Static	Discrete Discrete	Episodic Episodic
Chess with a clock	Multi	Partially	Stochastic	Non-collaborative	Semi	Continuous	Sequential
Poker Backgammon	Multi Multi	Partially Fully	Stochastic Stochastic	Non-collaborative Non-collaborative	Semi Static	Discrete Discrete	Sequential Sequential
Taxi Driving Medical diagnosis	Single Single	Partially Partially	Stochastic Stochastic	Collaborative Semi-collaborative	Dynamic Dynamic	Continuous Continuous	Sequential Episodic
Image Analysis Part-picking robots	Single Multi	Fully None	Deterministic Stochastic	Non-collaborative Collaborative	Semi Dynamic	Continuous Continuous	Episodic Episodic
Refinery controller	Single	None	Stochastic	Non-collaborative	Dynamic	Continuous	Sequential
Interactive English tutor	Single	Partially	Stochastic	Collaborative	Dynamic	Discrete	Episodic

- **Hierarchical organization**
    - Individual agents are organized in a tree-like structure
    - Agents' height in the tree have a more global view
    - Interactions do not take place across the tree

    - Data produced travel up the tree, control travel down the tree
- **Coalition**
    - Agents are group according to their goals
    - The organizational structure is typical flat, although there may be a “leading agent”
- **Congregations** (menigheter)
    - Groups of individual agents banded together into a typically flat organization to drive additional benefits
    - Congregations are formed among agents with similar or complementary characteristics
    - Potentially heterogeneous purposes behind each grouping, in comparison to typically homogeneous coalitions
    - Individual agents do not necessarily have a single or fixed goal
- **Team-based organization**
    - Number of cooperative agents that have “agreed” to work together toward a common goal
    - Teams attempt to maximize the utility of the team, rather than that of the individual members
    - Individual actions are consistent with and supportive of the teams goal
- **Swarm intelligence**
    - **Examples:**
        - Boids
        - Ant colony problem
    - **Utility-based agents**
    - **Learning agents**
    - **user agents**
    - **interface agents**
    - **information agents**
-
- **Area of use**
    - Simulations (Covid)
    - Experiments (population)
    - Collaborations (robots)
    - Negotiations

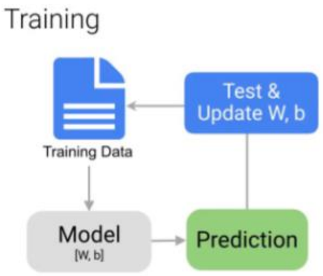
Machine learning - Is about automatically finding patterns and/or recognizing patterns in data sets. Learning without being explicitly programmed

- Machine learning uses algorithms for achieving data matching in a data set -> choose the best algorithm for the data set. Learn patterns, Predict result. Take lots of data and feed it into the system, get results in a form of cluster, class, rules or decision trees. The more data the better prediction.
- Problems to consider; skewed, not enough, too scattered, missing data
- **Terminology**
    - Building – model of data
    - Trimming – adjusting to the data in the data set
    - Label – one output/The known output (supervised learning)
    - Parameterized – array of values
    - Classification – identifying which of a set an observation belongs to
    - **Underfitting –** too small data set, no relationships between input data and target data
    - **Overfitting –** The training data do not adequately and accurately represent the entire data set, the training data may not match unseen data.
- **Classifier –** a function or procedure
    - Implemented by classification algorithm
    - Maps input data to a category
    - Takes data as input, analyzes the data and use the label as output
- **Create a classifier** – train = find pattern in examples
    - with **supervised learning**
    - provide domain knowledge
- **Clustering –** It is basically a collection of objects on the basis of similarity and dissimilarity between them.
    - with **unsupervised learning**
- **training methods – learning types**
    - **supervised learning –** Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.
    - **unsupervised learning –** Find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.
    - **Semi-supervised learning –** Represents the intermediate ground between Supervised and Unsupervised learning algorithms. It uses the combination of labeled and unlabeled datasets during the training period.
    - **Reinforcement learning –** Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty

- Classification algorithms – used as classifiers
    - **Support vector machines (SVM)** – linear/non-linear – used for classification (which set any data should belong to) and regression analysis problems (estimating the relationships of independent variables)
        - The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a **hyperplane**.
    - **Kernel methods –** for pattern analysis (find and study general types of relations)
        - mapping the input space of the data to a higher dimensional feature space, in which simple linear models can be trained, resulting in efficient, low bias low variance models.
    - **K-nearest neighbors (KNN) –** Stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm
    - **Decision tree –** generate rules for prediction modeling, map item to target value. Simple IF-THEN statements
    - **Neural Network**
    - **Deep learning**
    - **Support vector clustering –** improves the SVM by introducing clustering algorithms
        - **Connectivity models – Hierarchical clustering –** build nested cluster by merging or splitting these clusters. This hierarchy of clusters is represented as a tree or dendrogram
        - **Centroid models – K-means clustering –** represents each cluster by a single mean vector
            - Artificial neural network
- **ETHICS**
    - Privacy
- **Area of use**
    - Stock prices predictor
    - Customer predictor
    - Housing price prediction
    - Data mining
    - Beat humans in games
    - Translate between languages
    - Block spam

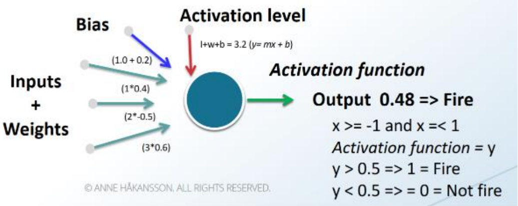
Neural Network - NN is a series of nodes or neurons that are connected with each other. The connection is used to transmit signals with different weights. Multiple layers handle complex problems

- Subset of machine learning
- **Perceptron –** a single neuron that can be used for two-class classification problems and provides the foundation for later developing much larger networks
  - Algorithm
  - Processor
  - Linear classifier
- **Perceptron algorithm –** receives input signals from examples of training data that we weight and combined in a linear equation (this equation is called the activation)
  - This way the perceptron is a classification algorithm for problems with two classes (0 and 1) where a linear equation can be used to separate the two classes.
- **Artificial neural network (ANN) –** is an interconnected group of nodes (perceptrons). Connections between neurons are weighted -> by modify these weight the ANN can perform complex classification task
- **Terminology**
  - **Weights (W) –** a constant between -1, 1
    - Weights are randomly initialized and distributed in the network
    - Training, means, applying training data + *adjusting the weights* until the output is correct according to the label -> training data is run multiple times through the network



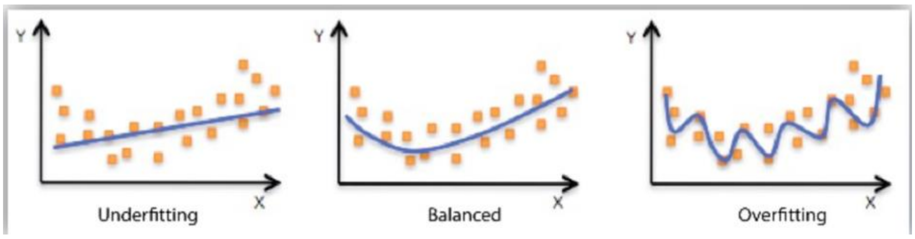
A training step

  - **Threshold/Activation level –** Decides if a neuron will become active. Neuron receives a number of inputs + weights. BIAS support setting the "breakpoint". Calculate the value -> activation level of the neuron **(Linear polynomial). Activation function** is applied to the inputs (e.g., Logistic Sigmoid)
  - **Bias (b) –** used for triggering the **activation level** (threshold), together with the sum of the input perceptrons. Makes adjustment within the neuron. Bias is learnable – has own weights.
    Purpose of the bias term is to shift the position of the curve left or right to delay or accelerate the activation of a node.
  - **Activation function –** calculation inside each neuron using a function, commonly non-linear (Logistic Sigmoid)
    - **Logistic Sigmoid –** range [0,1]
    - **Hyperbolic tangent –** range [-1,1]
    - **Heaviside (unit) step –** either 0 or 1
    - **ReLU – Rectified linear units –** linear scaling [0,1]



  - *How well we are doing –* compare desire value with actual value
  - **Epoch –** Every time the data set is run through – each pass of entire data set – Carefully change weights for every epoch, train slowly
  - **Batch size –** the number of samples that will pass through the network at one time

    - **Cost function –** determines how well a model works for a given data set. Calculate the difference between the expected value and predicted value and represents it as a single real number
        - **Regression Cost function**
            - **Mean Squared Error (MSE)** – adjust the weights of the neural network to minimize the mean squared error on training set
                - Error = (Actual output – predicted output)^2. Mean of all Error during training.
        - **Binary classification –** predicts for 0 (dog) or 1 (cat)
        - **Multi-class classification –** target values range from 0 to 1,3…, n-classes
    - **Optimization algorithm**
        - **Gradient descent –** minimize the cost function
- **Common neural networks**
    - **Feedforward Artificial Neural Network (FFANN) –** supervised learning algorithm
        - Peceptrons are arranged in layers and fully connected. Information flows form input to output
    - **Multilayer Perceptron Neural Network (MLPNN) –** supervised
    - **ANN with Backpropagation –** supervised learning
        - Fastest technique to find weights and bias values
        - **Learning rate –** moderating how much can be changed in one step
        - **Momentum –** resistance to change
    - **Genetic algorithms –** supervised learning
    - **Hebbian learning law –** unsupervised learning
    - **Self-organizing map** – unsupervised learning
    - **Reinforcement learning** – unsupervised learning, intelligent agents
        - ANN is part of the overall algorithm
        - Applied to multi-dimensional non-linear problems (vehicle routing)
        - Agent tries to maximize the total amount of reward it receives while interacting with a complex and uncertain environment
        - **Policy** (behaving)**, reward** (goal)**, value function** (good in long run)**, model** (environment)
    - **Deep neural network –** more hidden layers, more than tree total layers
    - **Deep learning –** see bellow
- **Normalization** – Standardizing the inputs can make training faster and reduce the chance of getting stuck in local optimization. Make data set symmetric
- **Backpropagation –** usually for unsupervised learning
    - **Forward pass –** figure out the outputs and total net weights
    - **Backward pass –** update each of the weights in the network. To cause the actual output to be closer the target output
    - **Process –** feed inputs to the network then compare the output with the correct value. This error value is used to change the weights. At each layer of the network, the error is *backpropagated* by using the weights to compute the amount of error assigned to each neuron in the network. The error is reduced by modifying the weights so the activation of the neuron is correct.



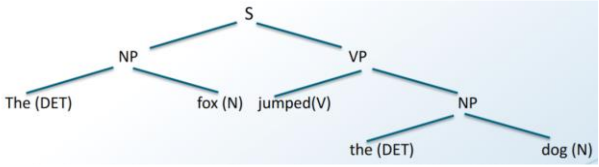
- **Underfitting –** When the model performs poorly on the training data. The model is unable to capture the relationship between the input examples and the target values
  - low variance and high bias means underfitting
    - **High bias** – try getting additional features, you are generalizing the datasets
- **Overfitting –** When the model is memorizing the data set it has seen and is unable to generalize to unseen examples. When the model performs well on training data but does not perform well on the evaluation data.
  - High variance and low bias means overfitting
    - **High variance –** get more training examples
- **Different data set**
  - **Training data (80%) –** train the model from this data, the system makes prediction on this model. Used in the process of updating the weights
  - **Validation data (20%) –** avoid overfitting, validate the model during training. DOES NOT update the weight
  - **Test data –** after the training, similar to deploying the model in the field
- **Area of use**
  - Read handwriting
  - Predict which team will win a game
  - Mine social media sentiment (a view or opinion that is held or expressed)
  - Images
  - Robotics, including directing manipulators

Deep learning

- **Networks**
  - **Convolutional neural networks (CNN) –** detect features in images
    - Type of feed forward network
  - **Recurrent neural network (RNN) –** neurons are fed information form the previous layer and from themselves from the previous pass
    - best used for sequential data
  - **Long-Sort term memory (LTSM) –** each neuron has a memory cell and three gates: input, output and forget
- Area of use
  - Image clarification
  - Visual tracking system
  - Face detection system
  - Handwritten characters

Natural language processing – Build software that analyzes, understands and generates human languages, naturally

- **Concepts:**
  - **Morphology** – study of words, how they are formed, and relationships to other words, analyzes the structure of words
    - **Stem** – the smallest meaningful unit for a word (untouchable, untouchable – s)
    - **Root** or a **root word –** does not have a prefix or suffix (Touch, un – touch – able – s)
      - **Prefix –** in front of the word (Un-)
      - **Suffix –** at the end of the word (-able -s)
    - **Lemmatization -** process of grouping together the inflected forms of a word so they can be analyzed as a single item.
      - **Inflicted forms –** a word is modified to express different grammatical categories.
  - **Phonology** – Study of pattern in sounds in languages
  - **Lexicology** – Study of words, their nature and meaning, words elements, relations between words including semantic relations (vocabulary)
  - **Parsing** – Process of analyzing string of symbols. Breaking down a text into its component parts of speech with an explanation of the form, function and syntactic relationship of each part
    - Parser – a program *works out* the grammatical structure of sentences
  - **Generation** – process of producing meaningful phrases (groups of words) from an internal representation (knowledge-base, data-base)
  - **Text Realization** – Provide sentence structure form *sentence generation*. Putting together multiple sentences forming texts
  - **Semantic** – needed to ascertain the meaning of the sentence, meaning of words, how to combine words into meaningful phrases and sentences.
  - **Pragmatic** – consideration of the context of the sentence utterance. Representing background knowledge and reasoning.
  - **Tokenization** – is a way of separating a piece of text into smaller units = tokens
- **IBM Watson (Natural Language Classifier) –** learns how to understand different phrasings expressed in natural language
- **PoS tagging (Part-of-Speech tagging) (grammatical tagging)** - process which refers to categorizing words in a text (corpus) in correspondence with a particular part of speech, depending on the definition of the word and its context. (S, NP, VB, DT)
- **Context-Free Grammar -** Context free grammar is a formal grammar which is used to generate all possible strings in a given formal language.
    - Sentences (S) -> noun-phrase, verb-phrase
    - Noun-phrase (NP) -> determiner, noun
    - Verb-phrase (VB) -> verb, noun-phrase

- 

- **Backus-Naur Form (BNF) –** Is an notation for context-free grammar that describe the syntax of languages used in computing, with four components: Start symbol, terminal symbol, nonterminal symbols, and rewrite rules.
- **Parse tree (derivation tree) (concrete syntax tree) –** ordered rooted tree that represents the syntactic structure of a string according to some context free grammar. Generated for sentences in natural languages.
- **Bag-of-words –** Simple method for feature extraction

- **Area of use**
  - Text generation
  - Topic modeling, collecting topics for documents
  - Chatbot
  - Healthcare
    - Improving the diagnostic process
    - Finding cures for diseases – finding new applications for existing drugs or to identify new patients
  - Digital therapeutic platform