**Sequential task envoirment (epsodicness)** – Behöver minne av tidigare actions för att bestämma nästa.

**Touring test** – Om förhöraren inte kan avgöra om det är en människa eller maskin, kan en maskin var intelligent.

**Complete algortihm –** addresserar all möjliga inputs och missar ingen.

We evaluate the algorithms by their:

– Completeness (do they explore all possibilities)

– Optimality (do they find the solution with minimum path cost)

– Time complexity (number of nodes expanded during search)

– Space complexity (maximum number of nodes in memory)

**Iterative deepening (DFS)** – är itne complete. Men om state space är finite och inte har loopar, så kan vi hitta en lösning. Behöver inte vara Optimal!

**A\*** - f(n) = g(n) + h(n)

**BFS** – är en kö struktur (FIFO).

**DFS –** har en stack struktur (LIFO). Har O(bd) i minnet. Inkomplett. Inte optimal. Linjär space, expo i tid.

**SVM** – Hittar **optimal** solutions. Maximerar marginalen mellan support vektorerna. Quadratic progamming problem som är lätt att lösa.

**Support vector** are the data points that lie **closest** to the decision surface (hyperplane). De är de svåraste datapunkterna att klassifisera. De har direkt bärande på den optimala platsen av decision surface.

Är en inductive learning algortihm.

**Inductive learning** – observation -> conclusion

Deductive learning – conclusion -> observation

**The large margin classifer** – maximerar marginalen runt den separerande hyperplanet. För att maximera distansen mellan planet vill vi minimera ||w||.

**Local beam search** – heuristic sök algortihm that explores a graph by expanding the most promising node in a limited set. Det är en optimering av best-first serach. Använder BFS för bygga ett sökträd. Är en greedy algorithm.

**Hill-climbing** -

**Local search** – använder väldigt lite minne. Keep track of single current state. Move only to neighboring states och ignorerar paths.

**Resilient propgataion (RPROP)** – är en algortihm som kan träna neurala nätverk. Snabbare än back propgation och den behöver inga ”free parameters values”. Mer komplex algortihm att implementera. Bygger på Backpropgation för att beräkna gradient that is needed in the calculation of the weights.

Det är en ”first order algortihm”

**Backpropagation (**Gradient descent) – används inom ANN för att beräkna gradient för vikterna.

**CNF**  - Conjuctive normal form (boolean logic)

En formel är på CNF om det är en sammanslutning av en eller flera klausuler.

En klausul är en disjunkion l1 v l1 … v l1 av en eller flera bokstäver

**Propositonal logic** – It deals with propostions (which can be true or false) and argument flow.

Negation – NOT ¬

Conjuction – AND (A ∧B)

Disjunction – OR (A v B)

Implication – implies (A => B)

p => q är lika med Not P eller Q.

**Information gain –** the amount of information gained about a random variable from observering another random variable.

**Bayesian network** – probablistic diracted acylic graphical model.

Bayes theorem:

**Genetic algorithm** – är en search heuristic som är inspirerad utav theory of natural evolution. Med mutation, crossover och selektion. Det är en variant of **stochastic beam search.**

Börjar med en population. Rankar med en fitness funktion. Väljer sedan en selektion, gör en crossover och sedan muterar. De med bäst fitness blir det nya population.

**Supervised learning** -

**Machine learning** –

**Alpha-beta pruning** –

max node n if α(n) > β higher in the tree

min node n if β(n) < α higher in the tree

**Early stopping** - algoritmen slutar generera noder när det inte finns någon bra egenskap att dela på, snarare än att gå till alla svårigheter att generera noder och sedan beskära dem bort.

**Perceptron** – är en algortihm för supervised learning av binära klassificerare. En binär klassificerare kan bestämma om input, representerad av en vektor av siffror tillhör vissa specifika klasser.

Perceptrons lärning fungerar kanon med 2 klasser.

Multilayer perceptron om man vill ha fler.

**Inductive bias** (learning bias) – av en inlärningsalgortihm är den uppsättning antaganden som eleven använder för att förutsäga utdata givet inmantingar som den inte stött på. Ett set av assumptions som tillåter machine learnings algortihmen att generalizera.

**Hypotes** – Any hypothesis that correctly classifes all the training examples is said to be **consistent**.

1. Träningsdatan may be noisy so that there is no consistent hypothesis at all.

2. The real target function kan vara utanför hypotesen och måste approximeras.

3. En rot-elev som helt enkelt matar ut y för varje x sådan hx, yi tillhör D är kosekvent men misslyckas med att klassificera någon x inte i D.

**Consistent hypothesis**: a concept c that evaluates to T on all positive examples and to F on all negatives. (Alltså ger alltid rätt?).

Biased in the context that you are speaking means, that your model overfits the training data and can not generalize well. It means your model performs very well on your training data, but can not do well on cross-validation and test data. It is customary to say that biased learners memorize the training data which is really true. Biased learners don't *learn* the data, they *fit* the data. For understanding the other usages of *bias* take a look at this [question](https://datascience.stackexchange.com/q/18123/28175).

**Unsupervised learning**

**Uniformed search** – kan ses som f(n) = g(n) där g(n) är path cost.

**Best first search** – f(n) = h(n)

A\* - f(n) = g(n) + h(n)

January 2018

1. Short-term actions can have long term consequences
2. Branching factor is finite
3. A cost function returns the estimated cost from start node to current node, while the heuristic functions returns the estimated cost from current node to goal
4. BFS är implementerade som en FIFO.
5. Average branching factor för 8-puzzle is 3. (En ruta ledig)
6. DFS är inte optimal.
7. Skillnaden på alpha-beta pruned standard, är dubbelt om det funkar.
8. They are typically zero for msot datapoints. (alpha för det mesta är 0. För varje non-zero indikerar på att det är en support vektor (Langrage mutliplaer).
9. Using the sign of the gradient
10. Minimizes the margin by maximizing w.
11. x V y
12. x => y
13. Negation and disjunction
14. A directed acyclic graph with conditional probability distrubtions in node.
15. Information gain
16. Classification accuracy caluclated on training set(?)
17. vet itne
18. vet inte
19. A variant of stochastic beam search
20. (1+2+3+4+5+6)/6 = 3.5

May 2017

1. max node n if α(n) > β higher in the tree
2. Kopiera ett träningsset 10 gånger?
3. Finds the solution that has the lowest path cost among all solutons
4. Twice
5. An observer cannot tell the difference between our agent and other human being
6. New more fit candidate replace less-fit candidate in the population
7. Inte optimal
8. En stopping/removal method
9. Learning rule is a sound and complete method for a Perceptron to learn correctly classify any two-class problem.
10. SVM
11. Ingen aning
12. Set of assumptions that allows machine learning algorithm to generalize.
13. Vet inte
14. Bayesian networks
15. CNF är a conjuction of clauses, where a clause is a disjunction of literals.
16. Vet inte
17. Classification är discrimitive och regression generative
18. Den ger correct value to each test ?
19. Vet inte
20. First-order inference? Förmodligen inte

April 2017

1. Keeps only a small number of nodes in memory
2. A\* = g(n) + h(n) -> A = g(n), borde vara uniform cost search