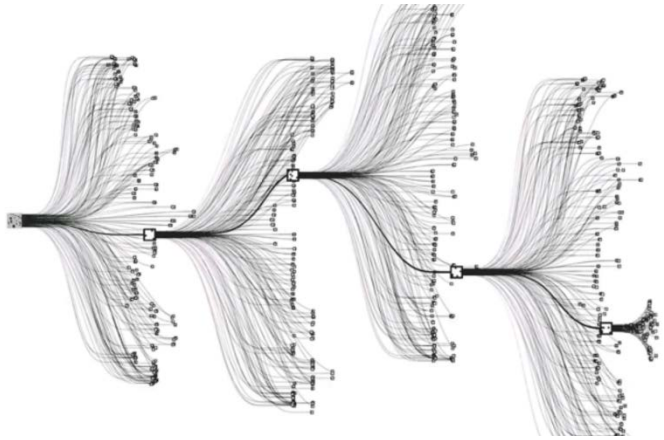
Koneoppiminen: Johdanto, peruskäsitteet ja menetelmät

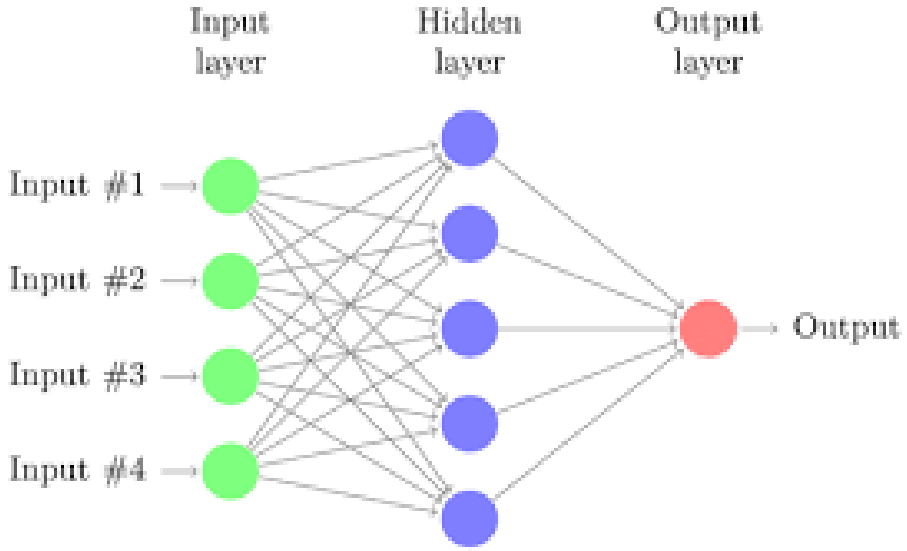
- Tomi Nieminen (FT, dos.)
- Ota yhteyttä: tomi.nieminen@jamk.fi
- Puh. 0445613075

- Maaliskuussa 2016 Googlen tekoäly AlphaGo voitti maailmanmestarin kiinalaisessa Go-strategiapelissä.

- Shakkiin verrattuna Go-pelissä on enemmän pelitilanteita kuin koko maailmankaikkeudessa on atomeita.
- Tekoäly ei tarvinnut pelin oppimisessa lainkaan ihmisen ohjausta. Se loi itse pelitaktiikkansa. Se kuitenkin aloitti opiskelun katsomalla ihmisten pelaamia pelejä (dataa).



- AlphaGo-ohjelman ytimessä on *neuroverkko*, jonka toimintaperiaate on samankaltainen ihmisaivojen kanssa.



- Kokemuksen (datan) kautta neuroverkkoon tallentuu *tietoa* (verkon painokertoimet). Tätä prosessia kutsutaan *oppimiseksi.*
- Kyseessä on siis aidosti oppiva äly, joka kehittyy itsenäisesti paremmaksi harjoittelemassaan tehtävässä.

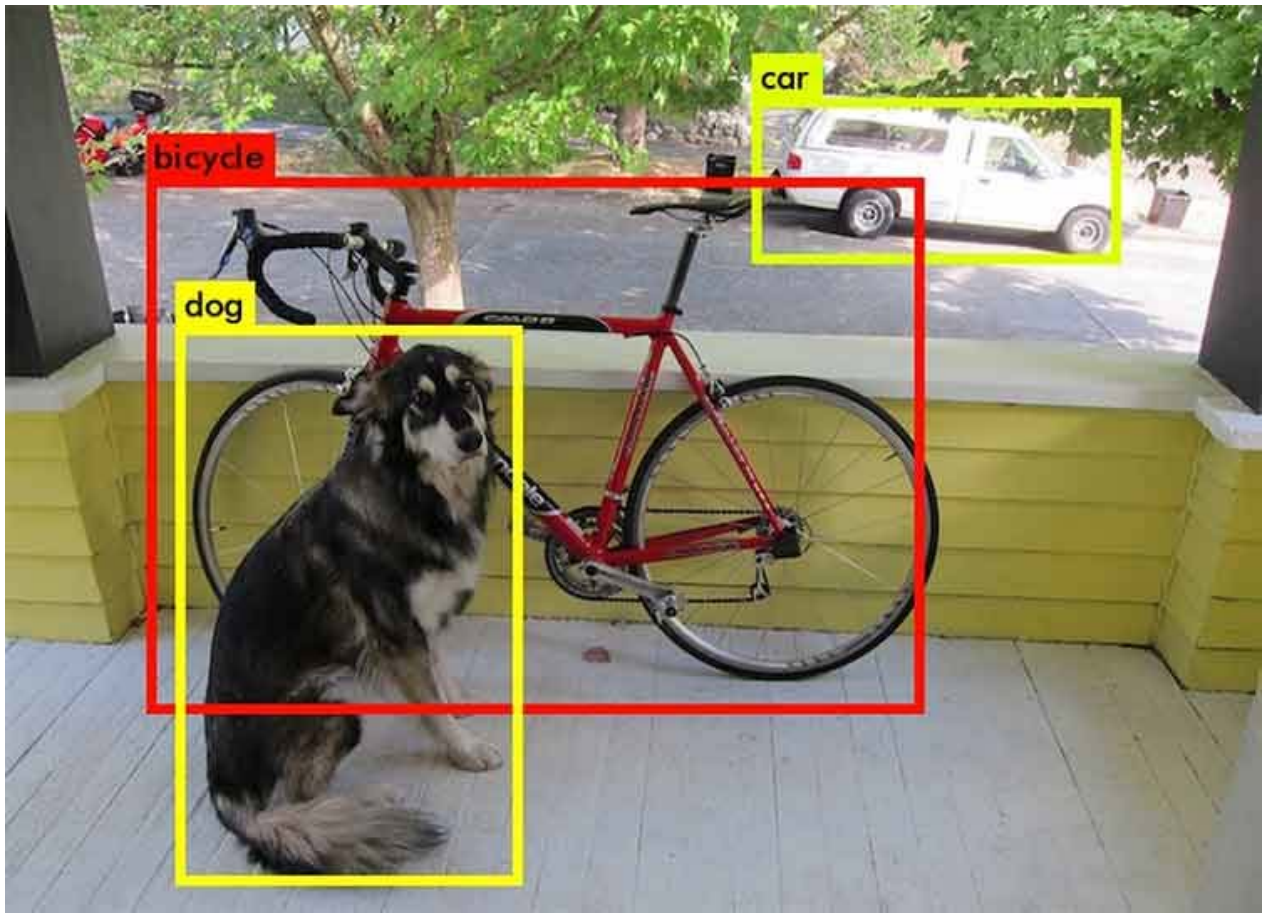
- Sama Googlen neuroverkko oppi nopeasti pelaamaan muitakin pelejä:

- Ihmisen ohjausta ei taaskaan tarvittu! Saamansa Score-pistemäärän tekoäly kuitenkin näki koko ajan harjoitellessaan.

- Tekoäly-käsitteen ympärillä pyörivän hypen takia lähes kaikki laitteet leimataan nykyään ”älykkäiksi”.
- Mutta esimerkiksi tämä kuvassa näkyvä Intelligent Dispenser ei ole oikeasti älykäs. ☺

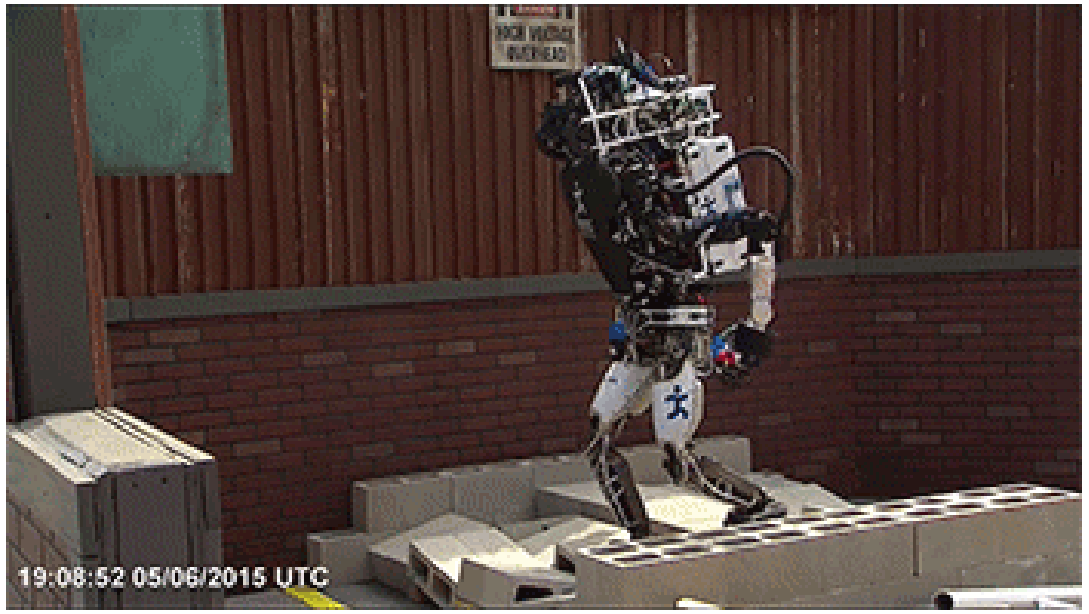
Todellisen elämän sovelluksia

- Kuvan luokittelu
- Puheen tunnistus
- Tekstin luokittelu (esim. positiivinen / negatiinen twiitti)
- Signaalin tunnistus (sähkö, ääni, valo…)
- Yhteys data-analytiikkaan: numeerinen input-data -> johtopäätös

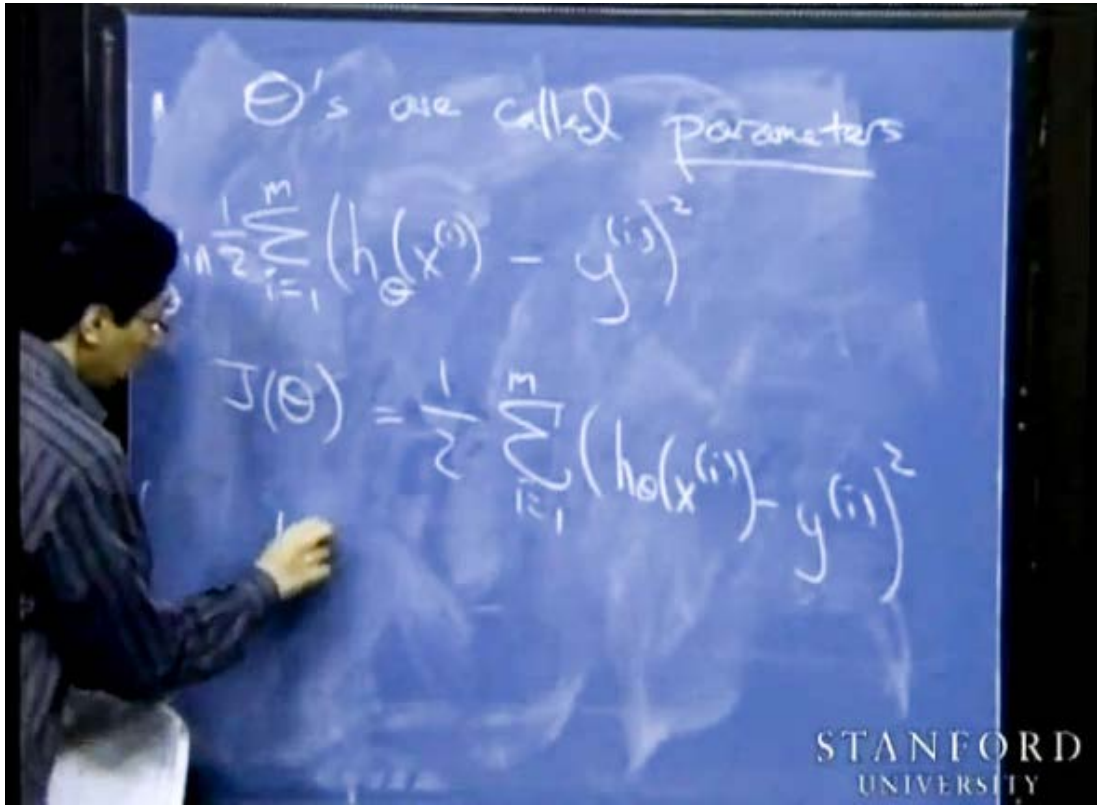


- Robotiikka
  - 2017 Googlen tekoäly opetteli itse kävelemään simuloidussa ympäristössä:
  - https://www.youtube.com/watch?v=gn4nRCC9TwQ

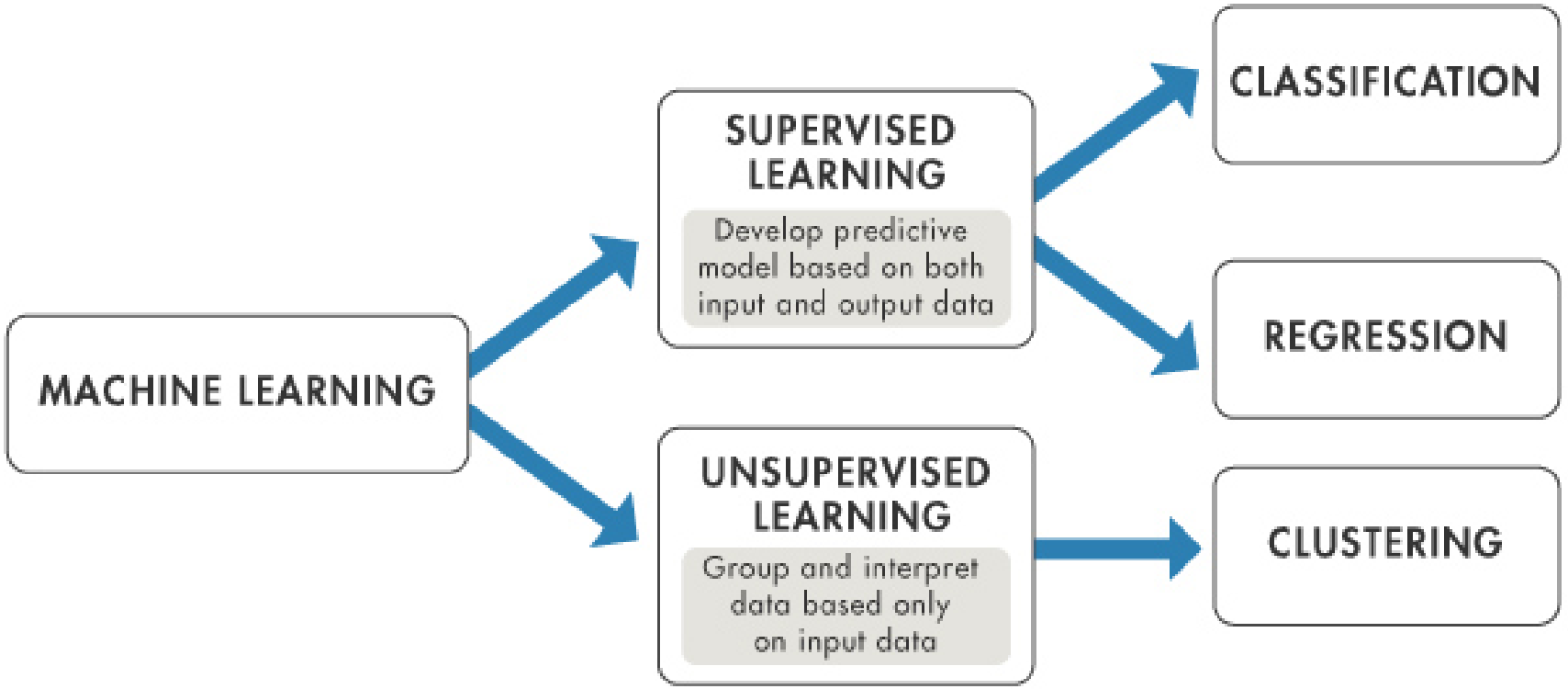
  - Mekatroniikka ei ole kuitenkaan ihan pysynyt kehityksessä mukana:
  - https://www.youtube.com/watch?v=g0TaYhjpOfo



- Koneoppiminen ja tekoäly on erittäin voimakkaasti kasvava tieteenhaara.
- Stanfordin yliopiston suosituin kurssi vuonna 2014: Machine Learning (CS229)
  - 760 osallistujaa, vaikka esitietovaatimuksiin oli kirjattu: ohjelmointi, vektorit ja matriisit sekä todennäköisyyslaskenta.
- *”Artificial intelligence is the new electricity”* – Andrew Ng



Koneoppimisen ongelmanasettelut ja yleisimmät algoritmit



- Logistinen regressio
- Päätöspuut
- Random Forest
- Naive-Bayes
- KNN
- SVM-luokittelu
- Neuroverkot

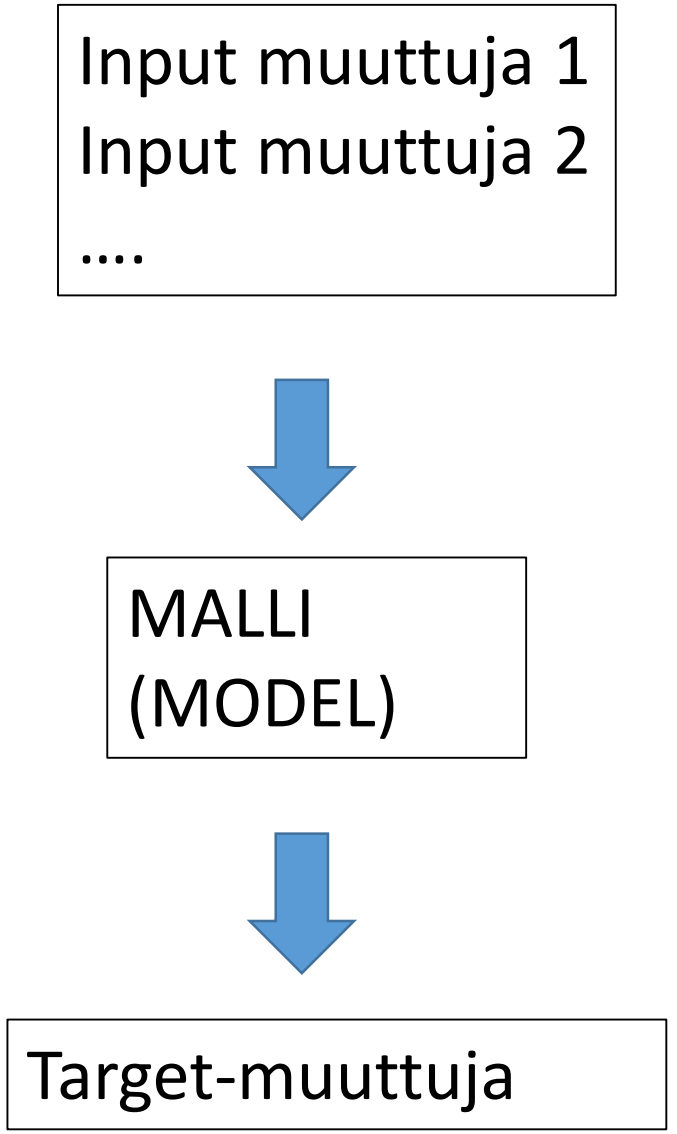
- Lineaarinen regressio
- SVM-regressio
- Random Forest
- Neuroverkot

- K-means klusterointi
- DBSCAN

Regressiossa ennustetaan välimatka-asteikollista (numeerista) Target-muuttujaa, kun taas luokittelussa ennustetaan luokitteluasteikollista Target-muuttujaa.

Mallin (model) käsite

- *Malli* on sääntö, joka liittää yhteen kiinnostuksen kohteena olevan Target-muuttujan sekä syötemuuttujat.
- Mallin parametrit ”sovitetaan” opetusdatan avulla. Tätä kutsutaan myös mallin ”opettamiseksi”.
- Opetettu malli osaa ennustaa Target-muuttujan arvon millä tahansa syötemuuttujien arvoilla. Ennustukseen liittyy toki aina jonkin verran epävarmuutta. Tämä virhemarginaali voidaan kuitenkin arvioida mallin opetusvaiheessa.
- Malli saattaa myös osata kertoa esim. mikä syötemuuttuja vaikuttaa eniten Target-muuttujan arvoon.



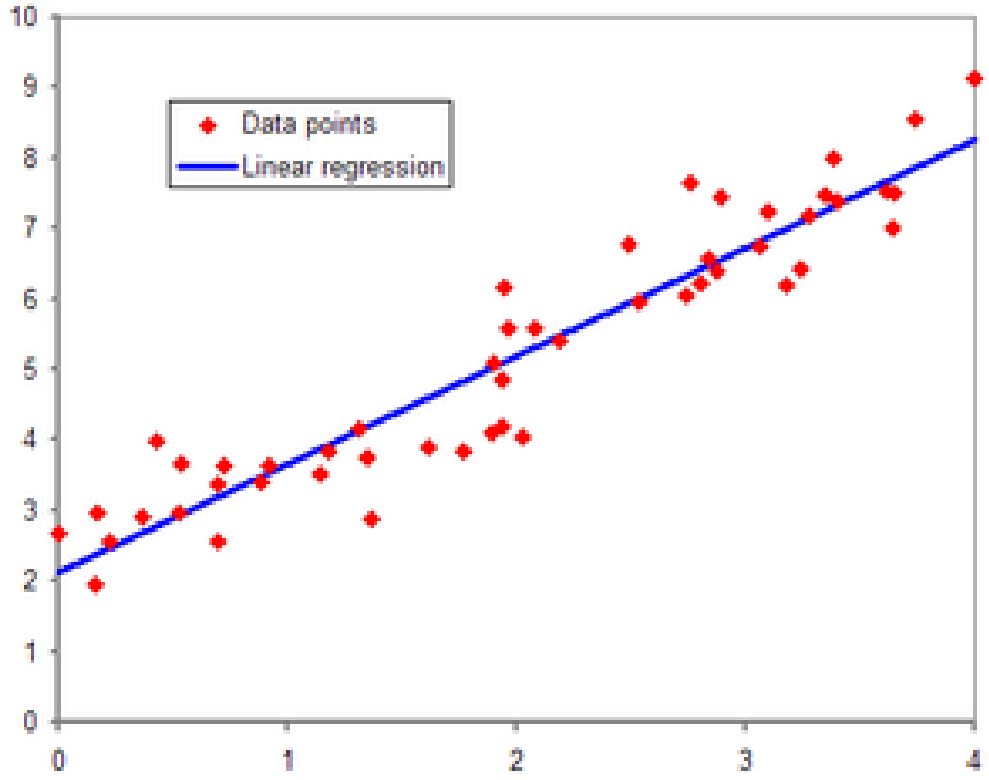
Mallien ominaisuuksia

Lineaarinen regressio

- Perusmenetelmä regressioon
- Nopea sovittaa, ei vaadi paljoa laskentakapasiteettia tai muistia.
- Myös ennuste erittäin nopea suorittaa.
- Mallin perusyhtälö on helppo tulkita, ja se antaa suoraan informaatiota eri input-muuttujien tärkeydestä (kunhan syötemuuttujat on skaalattu samaan mittakaavaan).

$$y = \alpha + \beta x$$

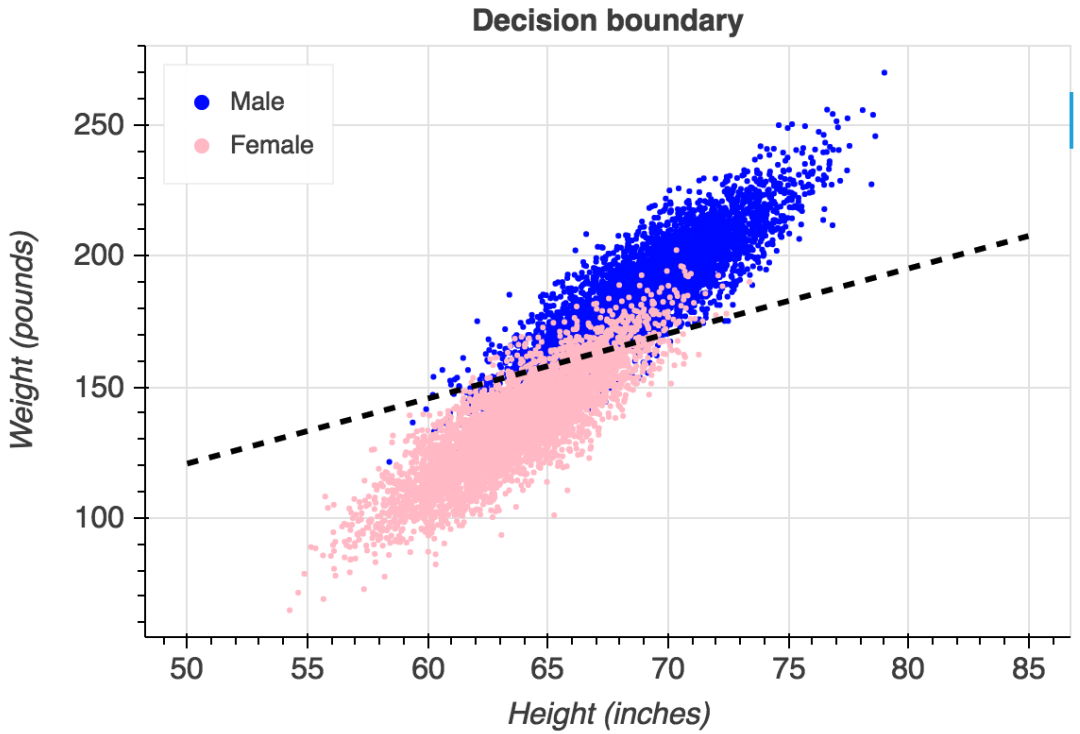
- Huomattavaa on, että yhtälössä α, β ja x voivat olla vektoreita, mikäli input-muuttujia on useita.



Logistinen regressio

- Perusmenetelmä luokitteluun (nimestään huolimatta).
- Nopea sovittaa, ei vaadi paljoa laskentakapasiteettia tai muistia.
- Myös ennuste erittäin nopea suorittaa.
- Malli on dikotominen luokittelija (luokat 0 ja 1), mutta siitä on myös moniluokkainen versio.
- Mallin päätösrajapinta piirreavaruudessa on suora (ks. kuva).
- Mallin perusyhtälö antaa tietoa syötemuuttujien vaikutuksesta target-muuttujaan.

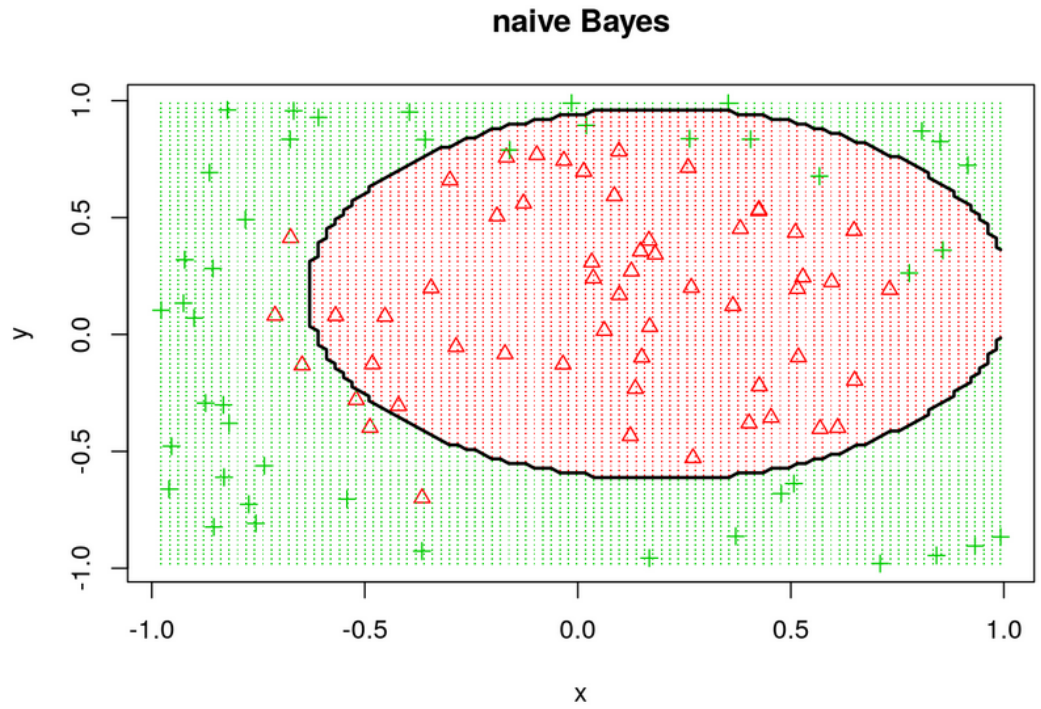
$$\pi(x) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}}{1+e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}}$$



Naive Bayes

- Toinen perusmenetelmä luokitteluun
- Tarkkuus vaihtelee, mutta malli on kevyt ja helppo toteuttaa logistisen regression rinnalla.
- Päätösrajapinta piirreavaruudessa voi olla käyrä, kuten tässä kuvassa.
- Mallin perusyhtälö hieman vaikeatulkintaisempi kuin lineaarisessa ja logistisessa regressiossa.

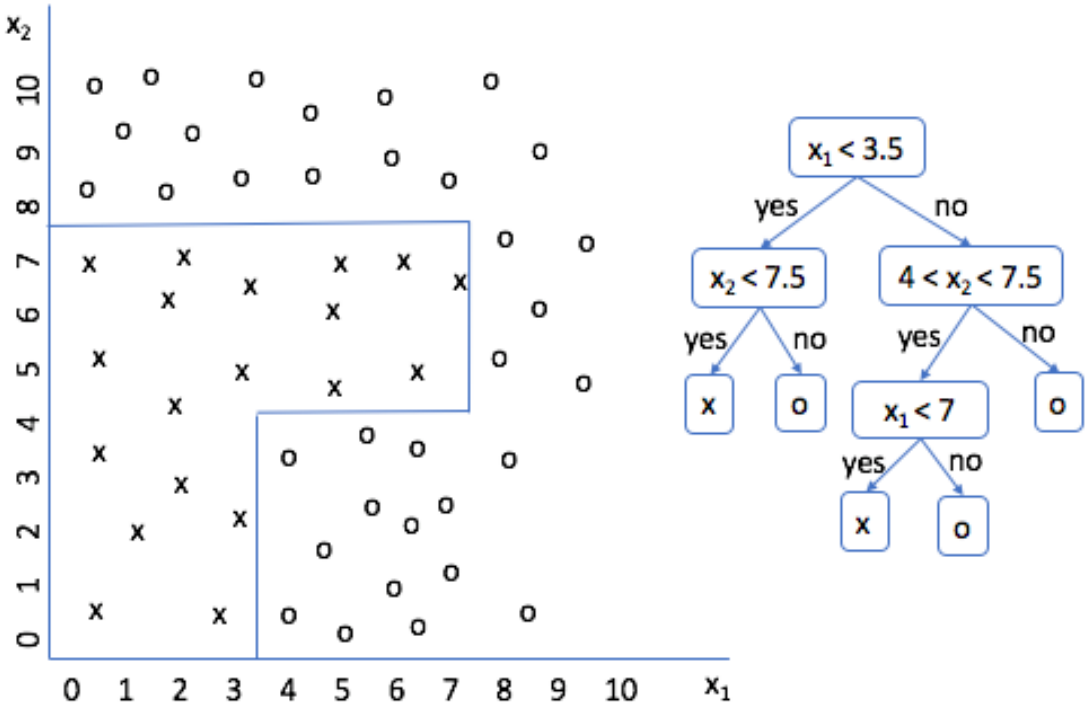
$$\hat{y} = \arg\max_y P(y) \prod_{i=1}^{n} P(x_i \mid y)$$



Päätöspuut ja Random Forest

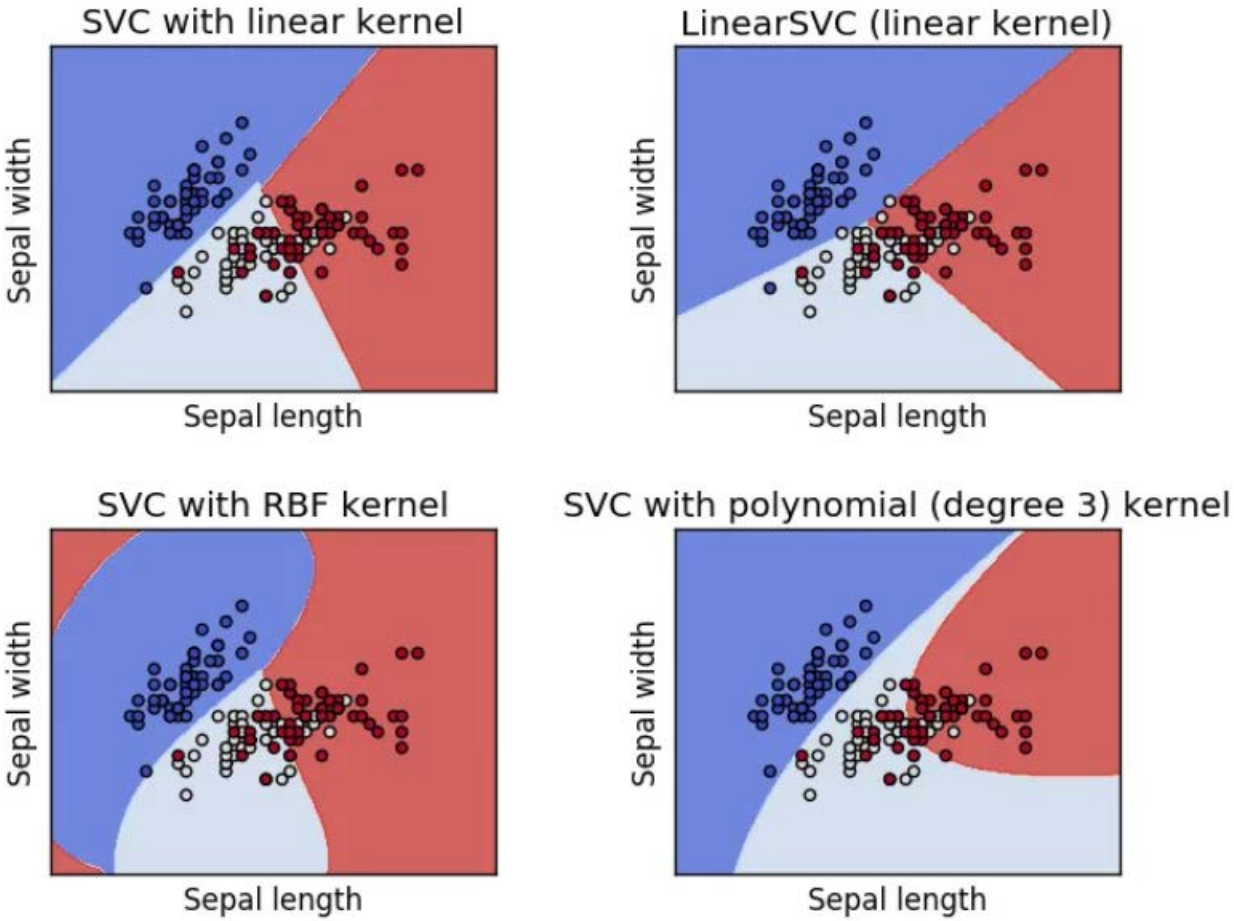
- Päätöspuu on hieman edistyneempi malli luokitteluun.
- Etuna erinomainen havainnollisuus.
- Tarkkuus ja laskentakapasiteetin tarve riippuu puun syvyydestä: syvät puut tarkempia, mutta ne vaativat samalla enemmän laskentakapasiteettia sovitusvaiheessa.
- Vaarana on *ylisovittaminen:* syvä päätöspuu saavuttaa erinomaisen tarkkuuden opetus-datassa (training-data), mutta mallin tarkkuus test-datassa onkin huono.

  Ylisovittaminen on tosin vaarana kaikissa koneoppimismalleissa.
- Random forest –algoritmissa muodostetaan kokoelma päätöspuita. Se on vähemmän altis ylisovittamiselle kuin yksittäinen päätöspuu. Sillä voidaan tehdä myös regressiota.



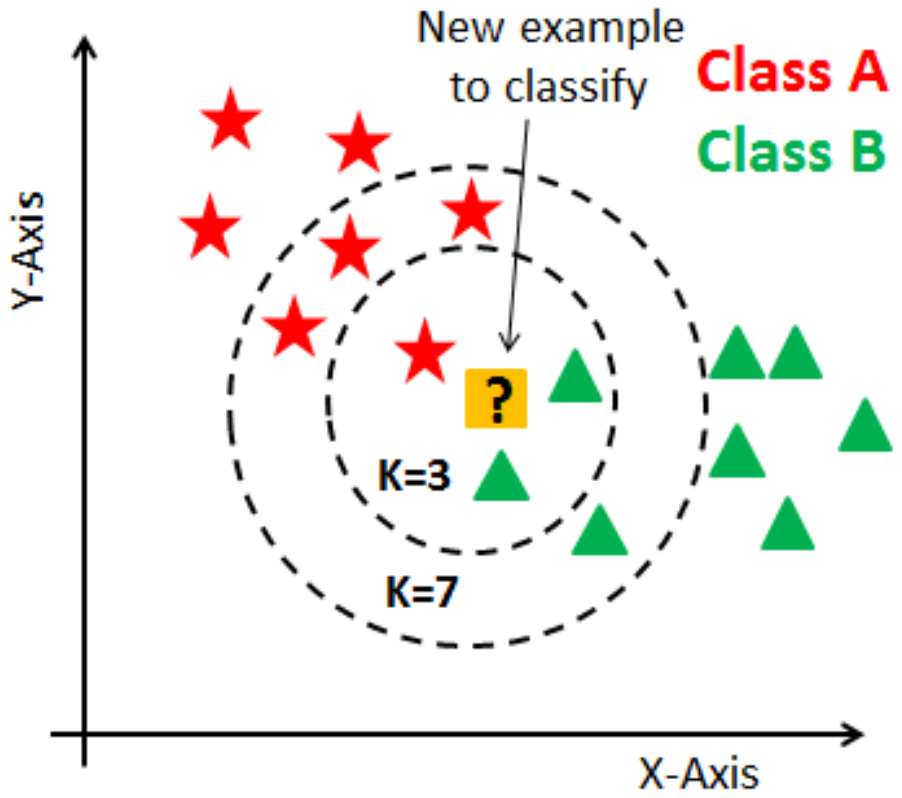
Support Vector Machine

- Edistyneempi menetelmä luokitteluun tai regressioon.
- Sovittaminen vaatii paljon laskentakapasiteettia ja muistia: kompleksisuus $O(n^2)$
- Sovituksen jälkeen ennusteet kuitenkin nopeita.
- Tulos riippuu valituista ”Kernel-funktioista” oheisen kuvan mukaisesti. Kaarevat päätösrajat piirreavaruudessa ovat mahdollisia epälineaarisilla kernel-funktioilla.
- Tyypillisesti saavutetaan parempi tarkkuus kuin Logistisessa regressiossa tai Naive Bayes -luokittelussa, mutta hintana on paljon pidempi laskenta-aika sovituksessa.
- Ei anna helposti tietoa syötemuuttujien keskinäisestä tärkeydestä.



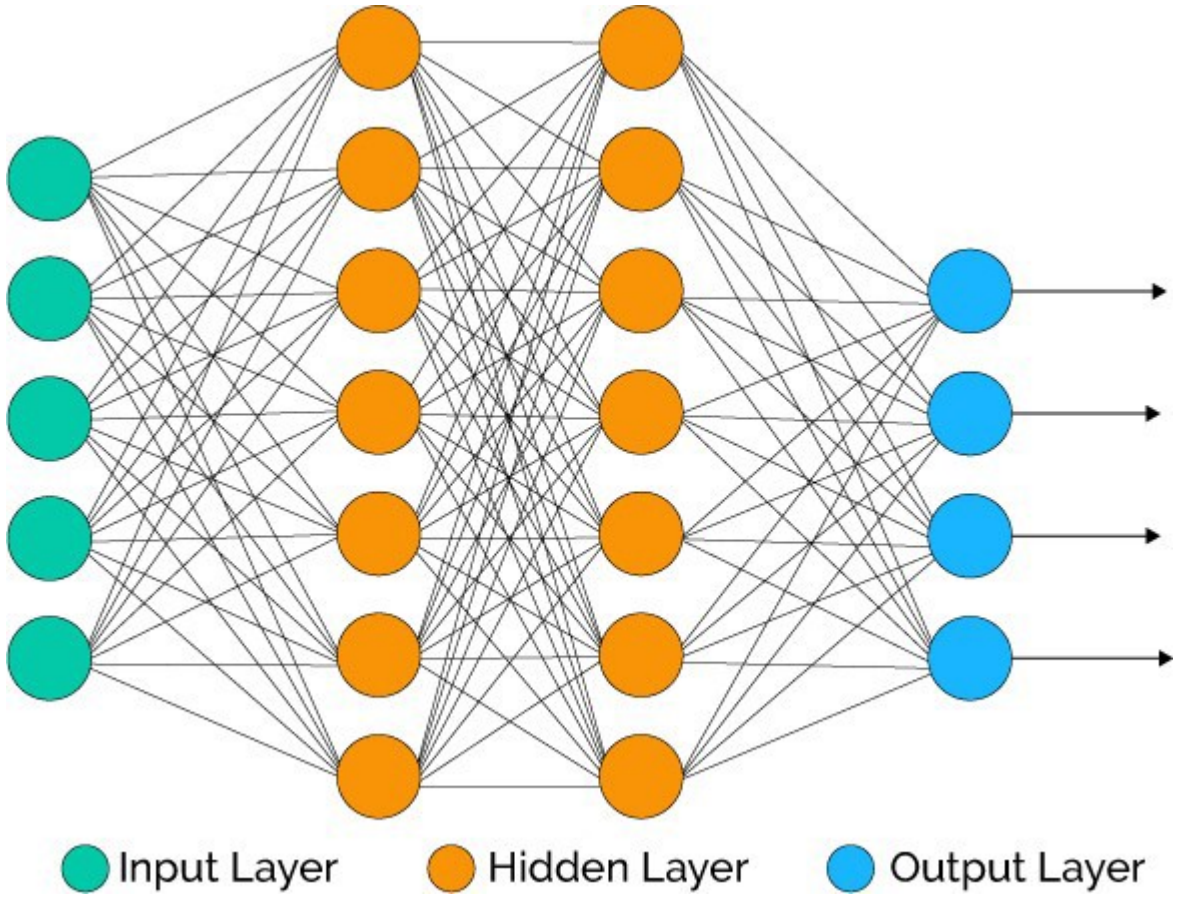
K-nearest neighbors

- Yksinkertainen luokittelumenetelmä.
- Lasketaan tuntemattoman alkion etäisyys kaikkiin training-datan alkioihin piirreavaruudessa. Otetaan lähimmät 3 (tai 5 tai 7) alkiota ja valitaan tulokseksi enemmistön luokka.
- Nopea sovittaa, mutta ennusteet hitaita, koska jokaisen ennustuksen kohdalla täytyy laskea etäisyys kaikkiin alkioihin. Vaatii myös paljon muistia, koska koko training-data täytyy säilyttää muistissa.
- Mallin tarkkuus on yleensä hyvä, mutta yllä olevat laskennalliset vaikeudet rajoittavat tämän mallin soveltamista käytännössä.



Neuroverkot

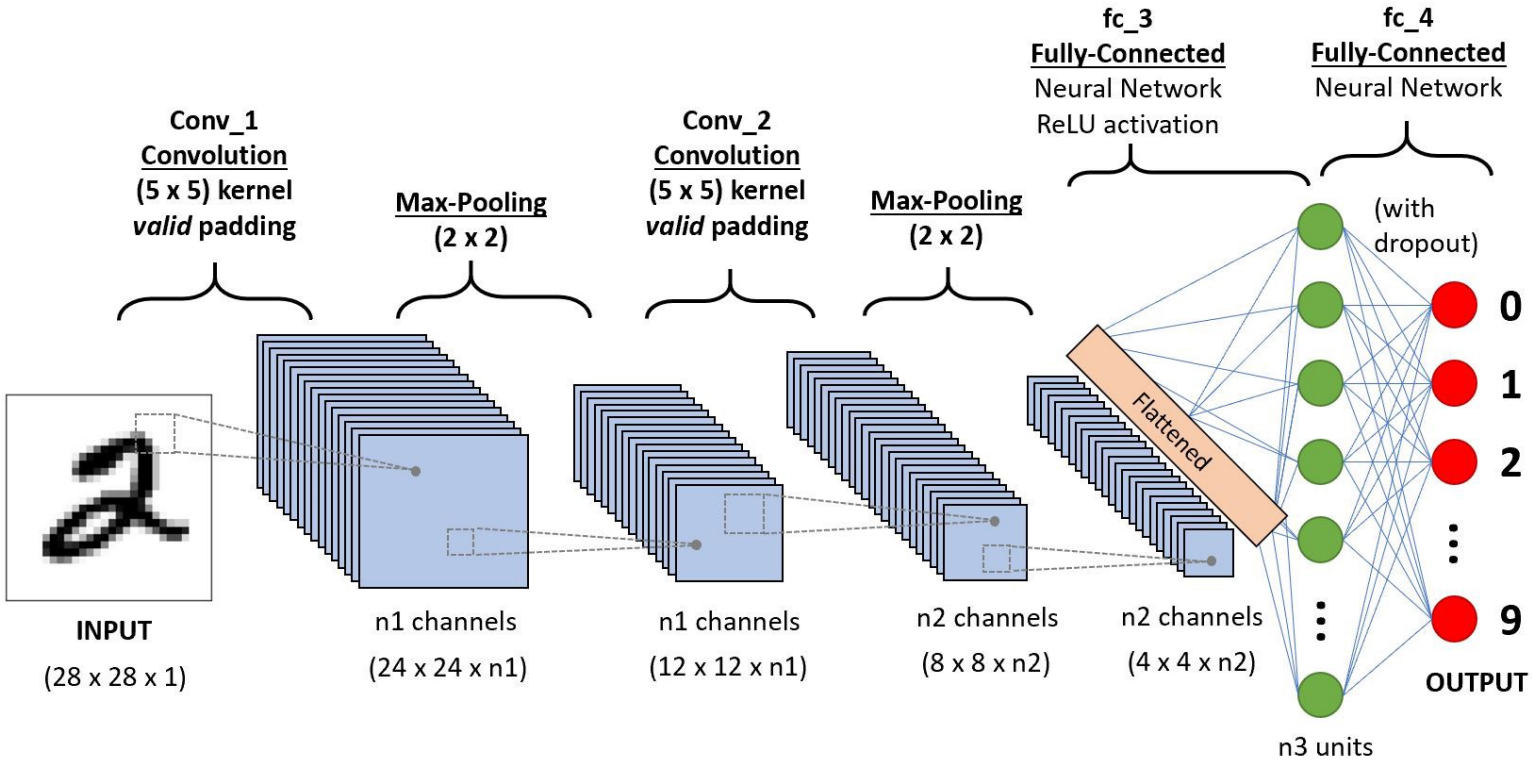
- Voidaan käyttää regressioon tai luokitteluun.
- Kun puhutaan oppivasta tekoälystä, niin se tarkoittaa yleensä neuroverkkoa.
- Neuroverkon oppima tieto varastoituu verkon painokertoimiin (weights).
- Ehkäpä monipuolisin ja hienostunein koneoppimismenetelmä, mutta vaatii eniten asiantuntemusta sovittaa: ylisovitus on aina vaarana.
- Opettamisen jälkeen ennusteet nopeita.
- ”Musta laatikko”: ei anna ollenkaan havainnollista tietoa syötemuuttujien vaikutuksesta targetiin.
- Soveltuu vaativiin tehtäviin: kuvan tunnistus, äänen tai muun signaalin tunnistus, chattibotit, pelien tekoälyt, itsenäisesti ajavat autot, robotin opettaminen kävelemään...



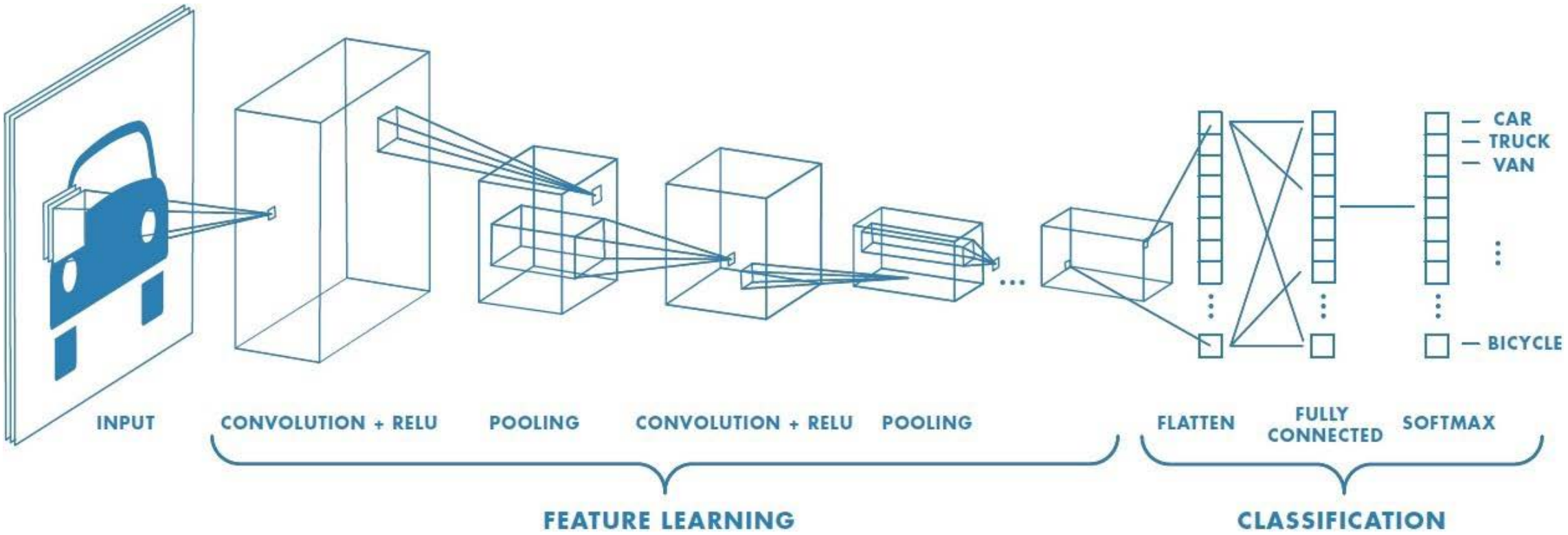
Syvät neuroverkot: CNN, RNN ja LSTM

Konvoluutioneuroverkko (CNN)

- Hienostunein menetelmä kuvantunnistukseen. CNN tunnistaa kuvasta ominaispiirteitä riippumatta siitä, missä kohtaa kuvaa ne sijaitsevat.
- Yksiulotteista konvoluutioverkkoa voidaan käyttää signaalin tunnistukseen: verkko etsii signaalin aikasarjasta ominaispiirteitä riippumatta siitä, millä ajanhetkellä ne tapahtuvat.

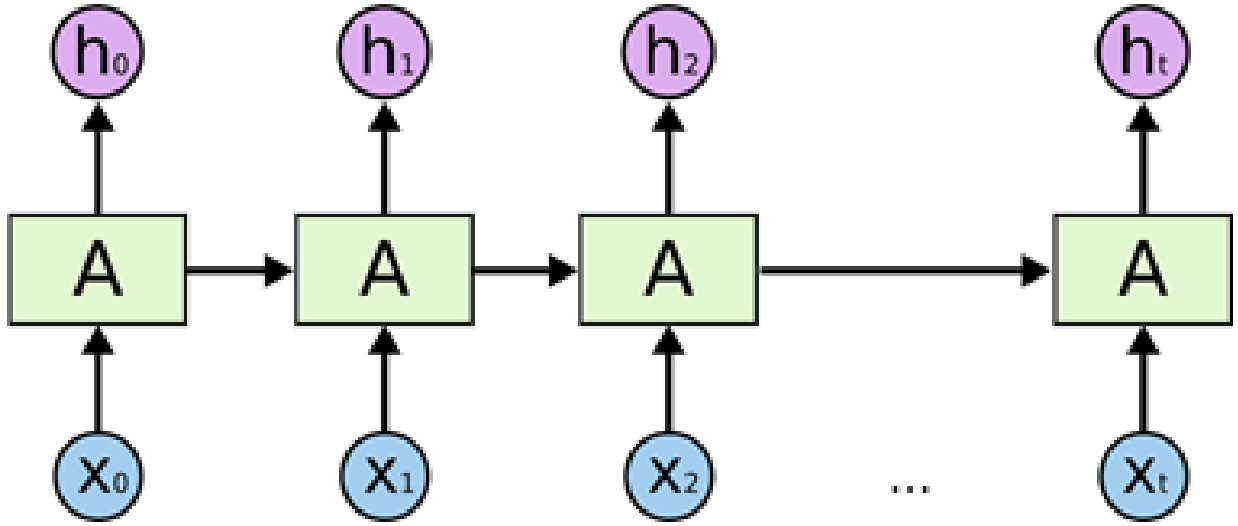


Konvoluutioverkon rakenne



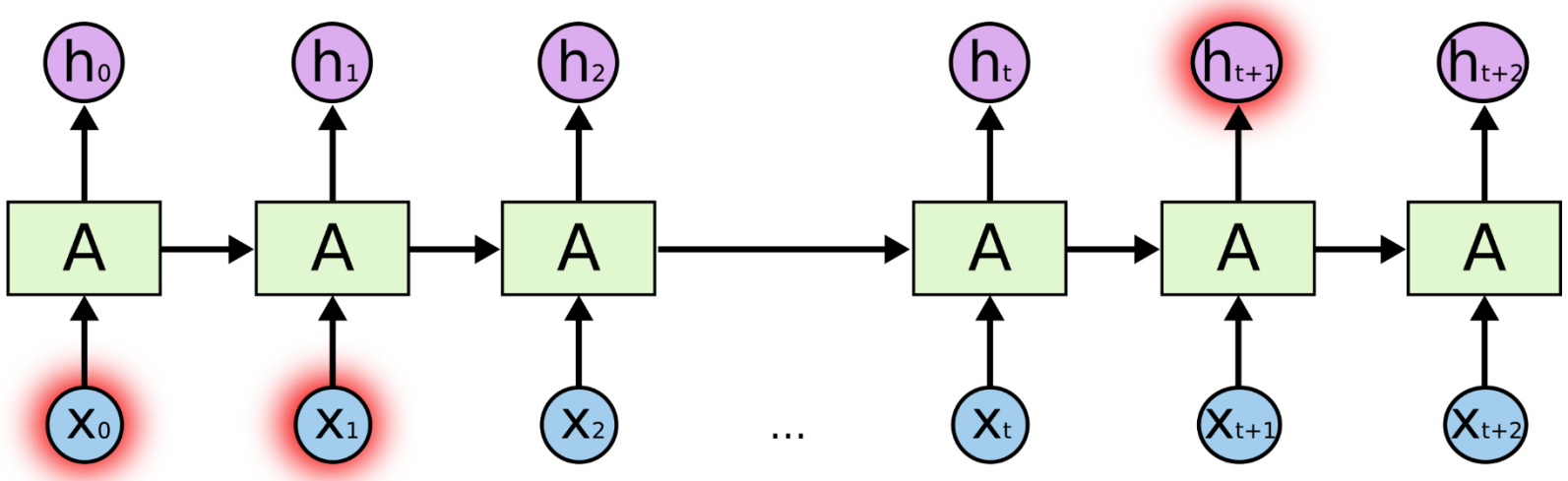
Rekursiivinen neuroverkko (RNN)

- RNN on erikoistunut käsittelemään aikasarjoja ja jonoja.
- RNN koostuu periaatteessa jonosta neuroverkkoja A (tai toisin sanoen luuppi, jossa toistetaan verkkoa A).
- Kukin verkko A tulostaa inputista x_t outputin h_t ihan kuten tavallinenkin neuroverkko. Mutta sen lisäksi RNN siirtää verkon A painokertoimet (eli verkon ”tilan”) tiedoksi jonon seuraavalle neuroverkolle. Tällä tavalla input x0 vaikuttaa välillisesti outputiin h1 vaikka x0 ei olekaan varsinainen syöte siihen neuroverkkoon, joka laskee outputin h1.



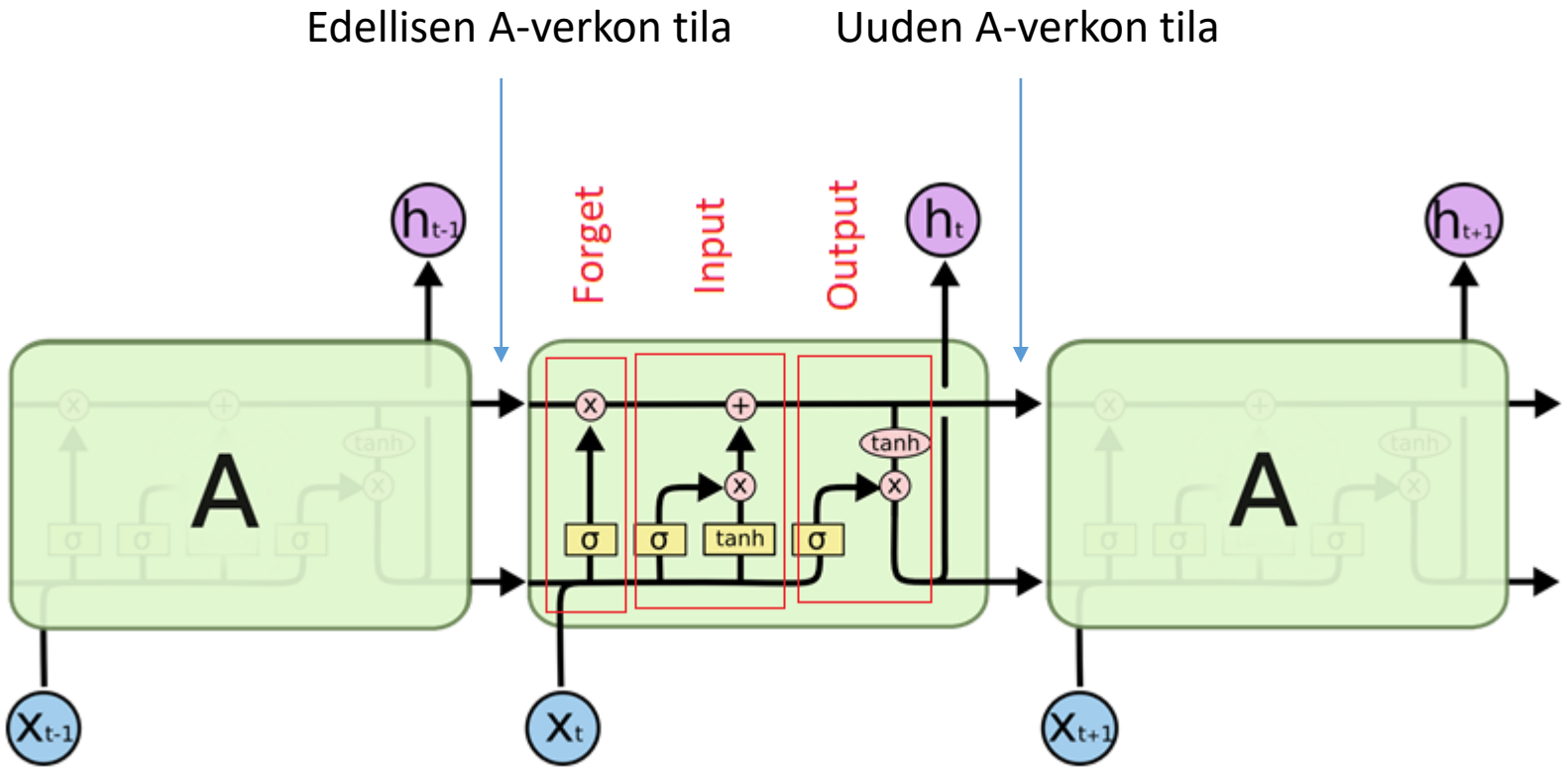
Rekursiivisen neuroverkon ”rako”-ongelma

- RNN:ssä input x0 vaikuttaa merkittävästi outputiin h1, mutta sen vaikutus outputiin h50 on hyvin vähäinen, koska niiden välissä on 50 A-verkkoa. Tätä välimatkaa kutsutaan ”gapiksi” eli ”raoksi”. Inputin x0 vaikutus ehtii siis ikään kuin unohtua matkan varrella.



LSTM-neuroverkko

- LSTM (eli Long Short Term Memory) neuroverkko on erityistapaus RNN-verkosta.
- LSTM ratkaisee rekursiivisen neuroverkon rako-ongelman hallitsemalla kunkin A-verkon tilaa hieman monimutkaisemmin 3 portin avulla: Forget-portti, Input-portti ja Output-portti.
- Forget-portti määrittää kertolaskuoperaation avulla, mikä informaatio edellisen A-verkon tilasta säilytetään ja mikä unohdetaan. Näin LSTM-verkko kykenee muistamaan tärkeät asiat hyvinkin kaukaa jonon alusta.



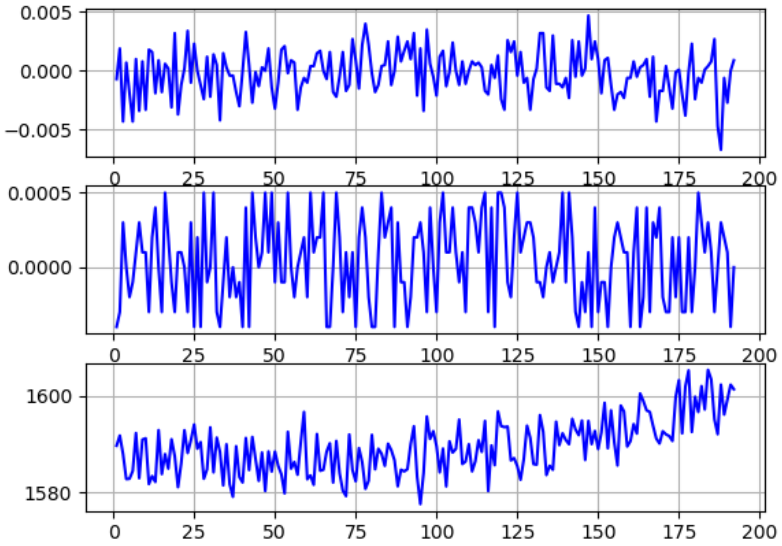
- Input-portti määrittää, mikä informaatio uudesta syötteestä otetaan mukaan A-verkon tilaan.
- Ouput-portti määrittää, mikä informaatio tulostetaan ulos nykyisestä A-verkosta (syötteellä x).

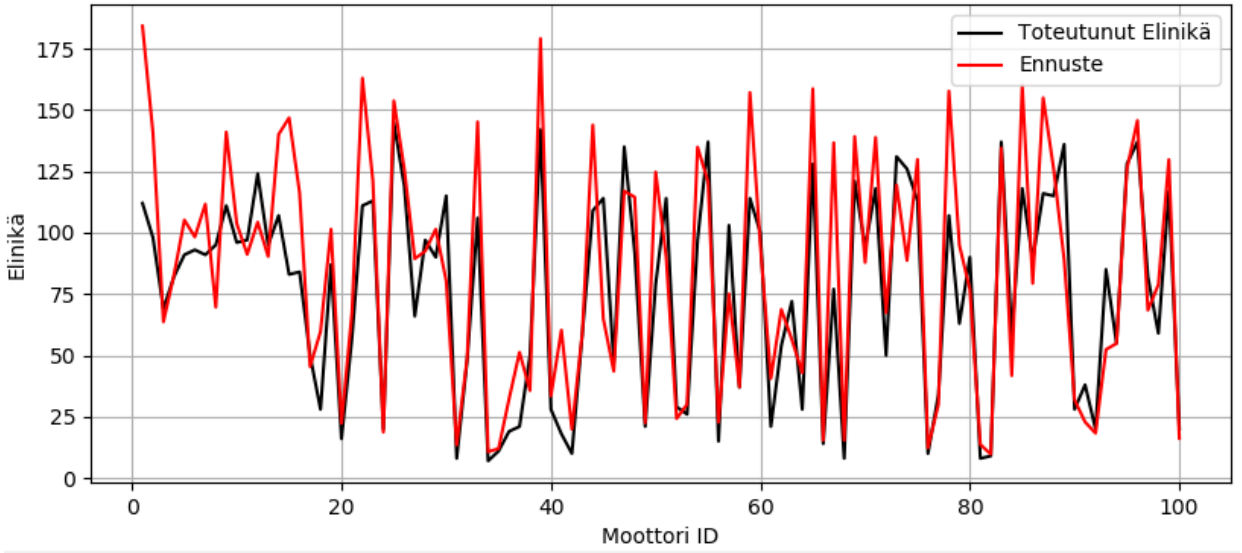
LSTM sovelluksia

- Aikasarjat, joissa historialla on vaikutusta sarjan jatkumiseen.
- Puheen tunnistus (epäselvän sanan arvaamiseen vaikuttaa edelliset sanat)
- Kirjoitetun tekstin tunnistus (epäselvän sanan tai kirjaimen arvaamiseen vaikuttaa edelliset sanat tai kirjaimet)
- Tekstin tuottaminen (Chatbotit)
- Musiikin tuottaminen (sopiva nuotti riippuu edellisistä nuoteista)
- Signaalien tunnistus

Esimerkki: Moottorin vikaantumisen ennustaminen (regressio)

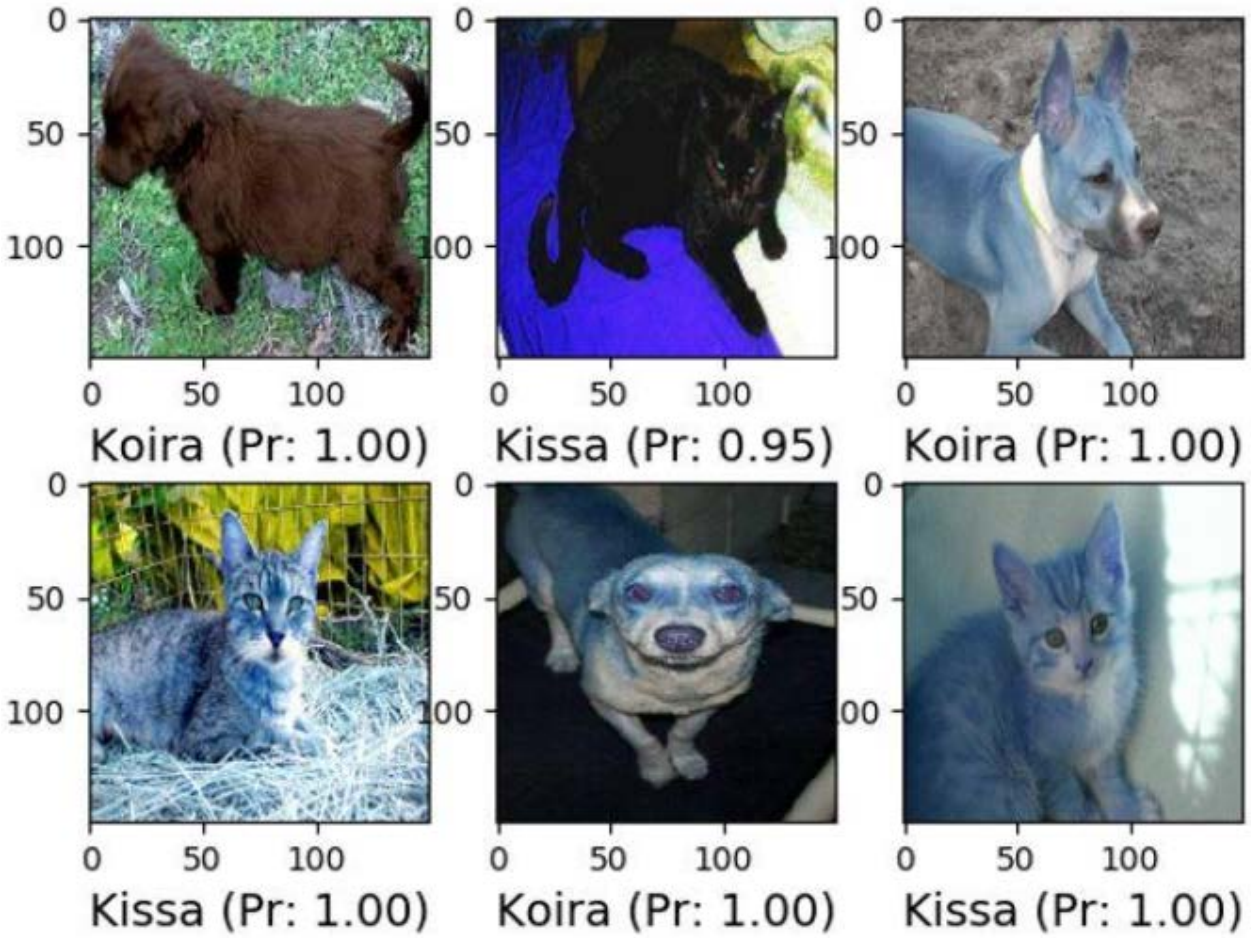
- Input-muuttujina 24 anturin mittausdata.
- Target-muuttuja moottorin jäljellä oleva elinikä.
- Osa antureista ei välttämättä liity vikaantumiseen lainkaan.
- Neuroverkkomalli löytää kuitenkin riippuvuudet ja 100 moottorin test-datassa saadaan erinomainen tarkkuus.





Esimerkki: Kuvantunnistus (luokittelu)

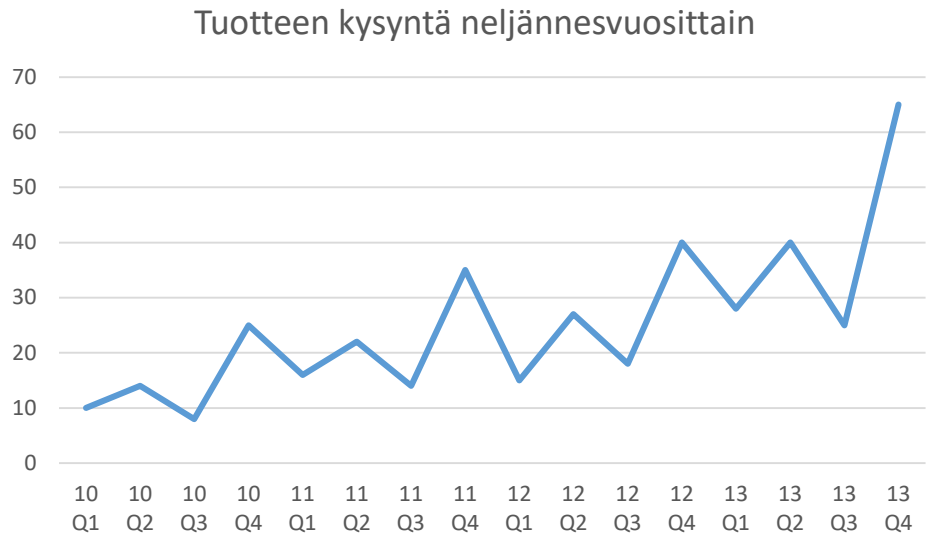
- Mallin Input-muuttujina on kunkin kuvan pikselit (150x150x3 kappaletta).
- Target-muuttuja on luokka ”Kissa” tai ”Koira”.
- Mallityyppinä tässä konvoluutioneuroverkko.
- Tekoälyn voi tietenkin opettaa tunnistamaan ihan mitä tahansa kohteita!



Esimerkki: Kohteen etsintä (luokittelu)

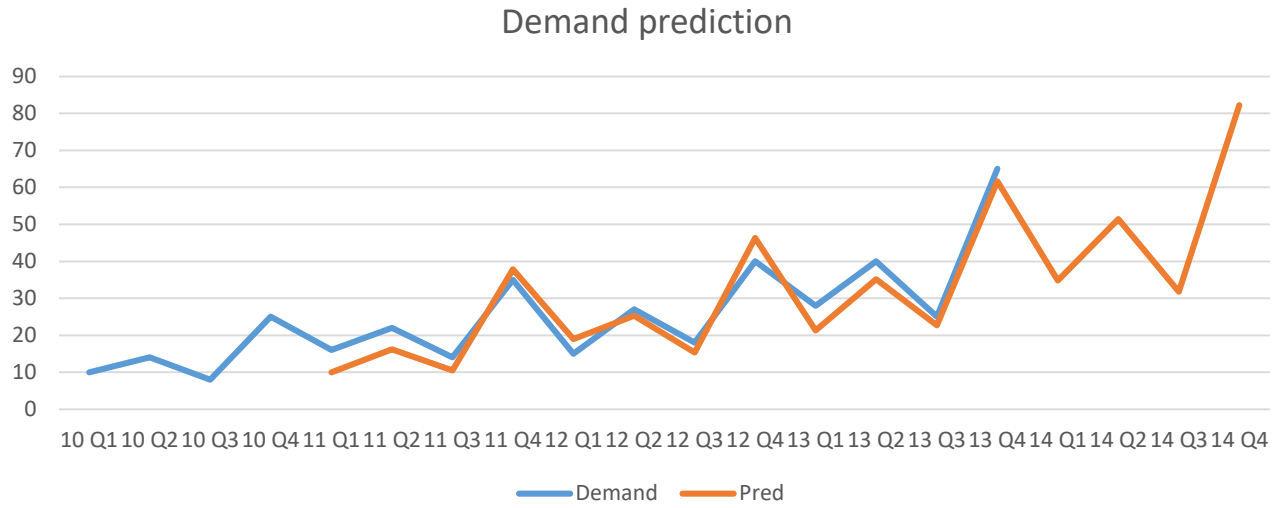
- Tekoäly etsii kuvasta tunnistettavia kohteita.
- Tässä on mukana myös ”object detection” -ominaisuus, joka etsii kuvasta useita kohteita ja tunnistaa ne sitten.
- Huomaa, että malli tunnistaa myös osittain piiloon jääneen auton 80% todennäköisyydellä.
- Mallityyppinä tässä esimerkissä alueellinen konvoluutioneuroverkko.

Esimerkki: Kysynnän ennustaminen (regressio)



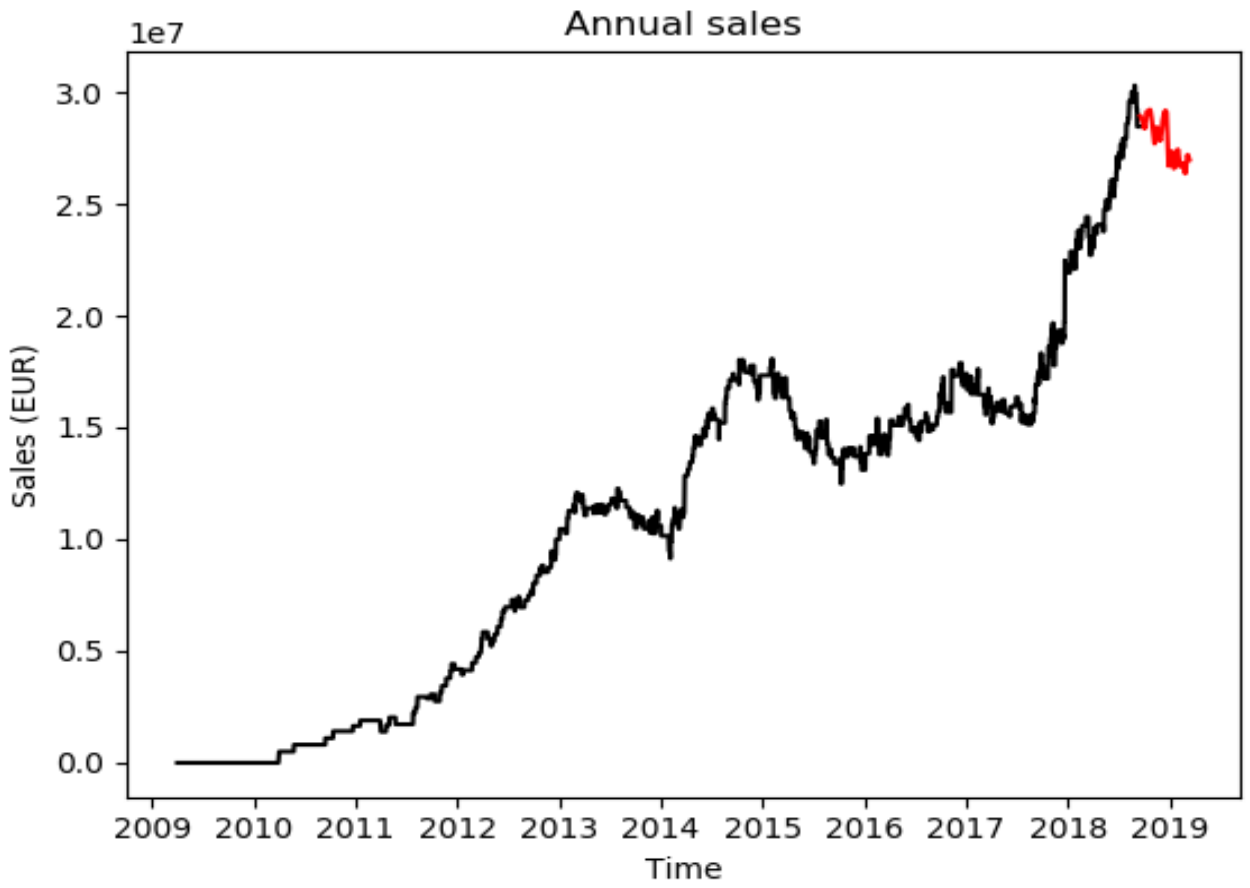
Miten käyrä jatkuu ensi vuonna?

Input-muuttujana pelkkä aikasarjan historia.



LSTM-neuroverkon (Long Short Term Memory) ennuste, joka oppii huomioimaan myös kysynnän kausivaihtelun.

Esimerkki: Myynnin tai liikevaihdon ennustaminen (regressio)

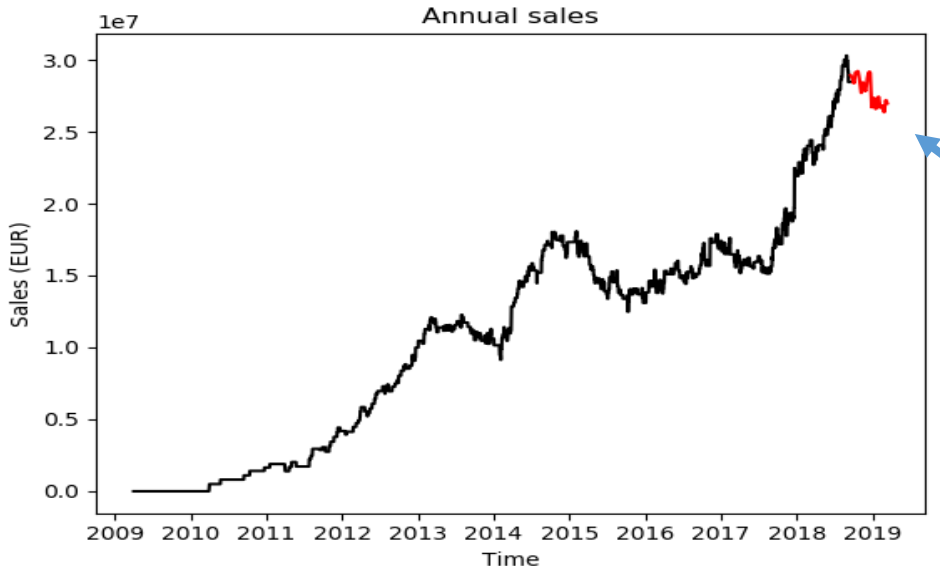


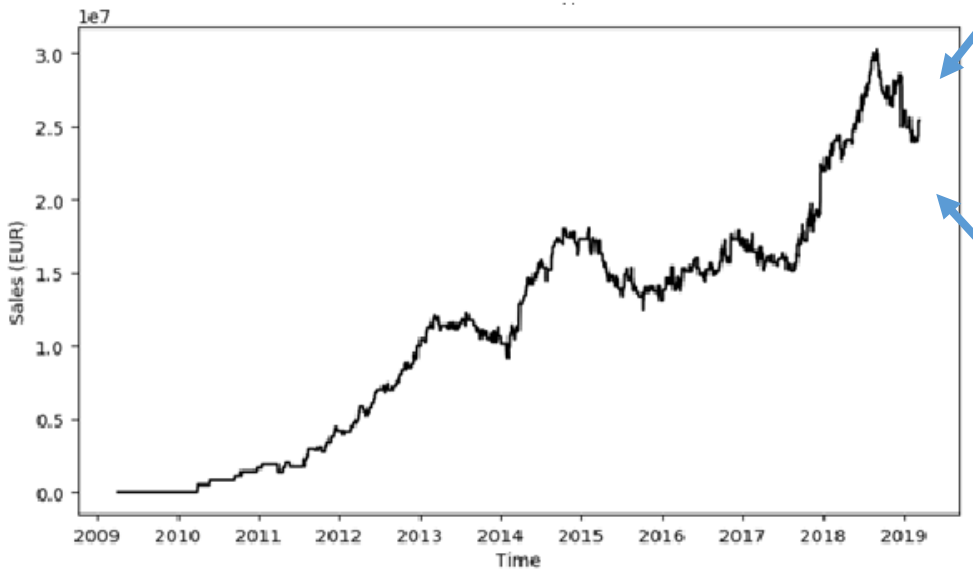
6kk ennuste

Toteutunut myynti

Kyseessä on erään oikean yrityksen myyntikäyrä. Malli saa syötteeksi avoinna olevien tarjousten sijainnin, tuotteen ja voittopotentiaalin sekä toteutuneiden tarjousten/kauppojen koko historian. Se ennustaa näiden perusteella tulevaa myyntiä.

Osuiko ennuste oikeaan?





Malli pystyi ennustamaan myyntikäyrän laskun, vaikka kyseessä on merkittävä muutos käyrän trendiin.

Toteutunut myynti maaliskuussa 2019