

Matematiska modeller för beslutsfattande hos människor, djur och maskiner

Kalle Åström

Abstract

The science of decision making is studied in many areas of science and engineering, for example within, computer science, automatic control theory, mathematics and neurophysiology. The aim of this text is to provide a very brief overview of some of the models and tools used within mathematical decision making. These models can be studied in terms of an agent that observes the world and takes decisions. The techniques that have been developed, e.g. optimal control, dual control, machine learning, reinforcement learning, have found numerous uses in many applications, where machines take decisions, but they are also an inspiration for the question of understanding biological decision making, and could potentially be important for understanding decision making in political science.

Inledning

Typiskt för modellering och särskilt matematisk modellering är att man behöver göra förenklingar av de fenomen man vill betrakta. Det betyder ofta att en modell bara fångar vissa egenskaper hos det man vill efterlikna. En leksaksmodell av en bil kanske är användbar för att beskriva färg och form, men kan inte användas för att studera t.ex. bränsleförbrukning. Typiskt för modelleringsprocessen är att man, trots att man vet att modellen har brister, genom analys försöker förstå vilka slutsatser som har bäring på det väsentliga eller centrala problemet. Det är då viktigt att man försöker förstå vilka aspekter som är relevanta och vilka som är osäkra eller direkt irrelevanta. Inom maskininlärning pratar man t.ex. också om hur väl en modell generaliserar till nya data.

Kalle Åström är professor vid Matematikcentrum, Lunds universitet.
E-post: kalle@maths.lth.se

Syftet med den här artikeln är att diskutera matematiska modeller för beslutsfattande. Målet är att ge en översikt över modeller för agenter som fattar beslut i en värld av andra agenter. I den här kontexten vill jag kunna vara väldigt generös med vad som kan vara en agent. Jag vill kunna prata om människor och andra djur, och då även väldigt enkla organismer, men också konstruerade maskiner, robotar och datorprogram. Man skulle också kunna se en organisation som en agent. I vår modell tänker vi oss att agenten har möjlighet att få information om sin värld, sin omgivning, genom sinnesintryck. Vi tänker oss också att agenten har en möjlighet att fatta beslut om hur den vill påverka sin omgivning. Centralt för vår modellering är alltså följande fyra begrepp: agent, värld, sinnesintryck och beslut (se figur 1). Vår förenklade modell av människan är alltså att den mottar ett antal sinnesintryck som den får genom syn, hörsel, smak, lukt och känsel, att den har möjlighet att fatta beslut om hur muskler kan styras och på så sätt förflytta sig, att den kan manipulera saker i sin omgivning och att den kan styra ögonen och prata och äta. En intressant fråga är därför hur man konstruerar agenter som självständigt lär sig hur världen är beskaffad och sedan använder sig av sinnesintrycken för att fatta bra beslut. Hur fungerar människor och djur i det här avseendet? Och hur kan vi konstruera algoritmer och robotar som verkar på detta sättet? Inom maskininlärning och artificiell intelligens utvecklas ett antal metoder och verktyg som kan användas i sammanhanget.

Min ambition är här att introducera några centrala begrepp för att se hur dessa kan nyttjas i en analys av adekvat beslutsfattande. Vi går kort igenom övervakat lärande, självkodning, världsmodeller, dual reglering samt förstärkningsinlärning och diskuterar hur de kan användas för att utveckla *automatiserade* metoder för att skapa agenter som både lär sig *och* kan fatta beslut. Några av begreppen och idéerna om hur agenter kan utveckla sina beslutsprocesser automatiskt är gamla, välutvecklade och välkända och används mycket redan idag, medan andra är betydligt nyare. Jag tar därför särskilt upp några begrepp från nyligen publicerade arbeten.

Fixerade beslutsprocesser

För många agenter är processen att gå från sinnesintryck till beslut fixerat, vilket innebär att agenten inte lär sig av sina erfarenheter, även om det kanske sker ett lärande ur ett evolutionärt perspektiv, (Abbott i detta nummer). Ett exempel (Sterling & Laughlin 2015) är *e-coli* bakterien som kan känna av om det finns glucos och laktos i sin omgivning. Baserat på detta kan den ta beslut om hur mycket av tre olika proteiner, som ska tillverkas för att föra in glukos respektive laktos, samt hur mycket galactosidase som ska tillverkas för att bryta ner laktos. Om det finns glukos så tillverkas det protein som behövs för att bryta ner glukos. Om det inte finns glukos, men det finns laktos, så tar

bakterien beslut om att tillverka det proteinet som för in laktos samt det protein som behövs för att bryta ner laktos. För just e-coli bakterien vet man dessutom precis hur beslutsprocessen går till rent kemiskt.

Bakterien e-coli tar dessutom beslut om hur flageller ska användas för att navigera. Beslutsprocessen för rörelsen kan förenklat beskrivas som att varva beteendet att åka rakt fram med en beteendet om att slumpmässig rotera. Om det verkar finnas mer mat här än för en sekund sedan, och du fortfarande rör dig rakt fram, kör i så fall ytterligare lite längre.

Klassificeringsproblemet inom maskininlärning

Hur kan man ta fram en optimal beslutsprocess? Inom maskininlärning är s k övervakat lärande en central idé; man tänker sig att man tränar upp agentens kapacitet att gå från sinnesintryck till beslut genom att presentera ett stort antal exempel på sinnesintryck samt tillhörande korrekt beslut. Man använder sedan matematisk statistik och optimering för att koppla ihop de två för att på så sätt få till stånd en fungerande beslutsgång (se figur 2).

Detta har visat sig vara framgångsrikt för många tekniska tillämpningar. Erfarenheten är att det är svårt att för hand skriva datorprogram som tar rätt beslut givet en viss insignal. Däremot kan det vara möjligt att samla på sig exempel på indata med tillhörande korrekta beslut. Med övervakat lärande tar man sedan fram en beslutsprocess som fungerar. Här kommer några exempel på tekniska lösningar baserat på övervakat lärande.

- Företaget Hövding tillverkar en cykelhjälm med en uppblåsbar krockkudde. Baserat på sinnesintryck tar hjälmen hundratals gånger per sekund beslut om krockkudden ska lösas ut eller inte. Företaget har samlat in en stor mängd exempel på data (normal cykling), då den *inte* ska lösa ut, och ett stort antal iscensatta olyckor, som ger exempel på data när den väl *ska* göra det. Med maskininlärning tar man sedan fram en beslutsprocess som givet data bestämmer när krockkudden ska aktiveras och inte.
- Företaget Cellavision (Simonsson 2018), tillverkar ett automatiskt system för analys av blodprov. Systemet hittar vita och röda blodkroppar och känner t ex igen olika typer av vita blodkroppar; företaget har samlat in en stor mängd data om just vita blodkroppar. För varje bild har ett antal experter angett vad som är rätt klass. Med maskininlärning tar man sedan fram en beslutsprocess som på grundval av bilderna avgör vilken vit blodkropp det rör sig om.
- Vid patologen på sjukhus undersöker man biopsier genom att färga in cellproverna och sedan studera dem i mikroskop. För varje prov ska man ange

på en så kallad Gleasons skala hur irreguljärt cellerna växer. Med maskininlärning kan man konstruera en beslutsprocess som bedömer dessa prover automatiskt (Gummesson 2017).

- I fartkameror finns det system som, förutom att mäta hastigheten på bilarna, också hittar registreringsskylten automatiskt. Ett system för att känna igen vilka bokstäver och siffror som finns på registreringsskylten bygger på klassificering. Man har sedan tidigare ett stort antal bilder av t ex siffror, där man för varje bild har angett korrekt klass.

Inlärningsprocessen är enklast om man har få sinnesintryck och få beslutsmöjligheter. Låt oss ta ett klassiskt exempel hämtat från biologen Ronald Fisher i en artikel från 1936 (Fischer 1936). Där beskrivs ett dataset med 50 mätningar från tre sorters blommor. För varje blomma har man mätt kronbladslängd, kronbladsbredd, foderbladslängd och foderbladsbredd samt noterat vilken art det är. Som illustration studerar vi bara en av egenskaperna (kronbladslängd), som vi betraktar som sinnesintryck; vi tänker oss att vi ska försöka avgöra om det är arten *Iris Setosa* (klass 1) eller arten *Iris Versicolor* (klass 2) det rör sig om. Det finns många olika tillvägagångssätt för att klassificera. Vanligtvis har man någon sorts träningsdata, i det här fallet $T_a = \{(s_1, b_1), \dots, (s_n, b_n)\}$, där s_k är sinnesintryck (kronbladslängd) för exempel nr k och b_k är det korrekta beslutet, antingen klass 1 för *Iris Setosa* eller klass 2 för *Iris Versicolor*. Ett sätt att utföra analysen och komma fram till ett beslut är att studera träningsmaterialet enligt figur 3. Första steget är att skatta sannlikhetsfördelningen $f_x(x|Y = 1)$, dvs hur sannolikt det är att kronbladslängden är x om det är en mätning från en *Iris Setosa*. Överst i figur 3 illustreras just de mätningarna med stjärnor.

Genom att använda en metod som kallas kärnskattning, (Parzen 1962; Rosenblatt 1956), kan man skatta sannlikhetsfördelningen $f_x(x|Y = 1)$ för kronbladslängden för blad hos *Iris Setosa*. I grafen har vi multiplicerat den med sannolikheten $P(Y = 1)$ för att en slumpmässig växt är just *Iris Setosa*. I det här exemplet har vi antagit att $P(Y = 1)$ är 50 %. På motsvarande sätt har vi skattat sannlikhetsfördelningen $f_x(x|Y = 2)$ för kronbladslängden för *Iris Versicolor*. Analysen visar att kronbladslängderna typiskt är lite kortare för *Iris Setosa* jämfört med *Iris Versicolor*. Genom att summera $f_x(x|Y = 1)P(Y = 1)$ och $f_x(x|Y = 2)P(Y = 2)$ får man den totala sannlikhetsfördelningen $f(x)$. Genom att sedan dividera $f_x(x|Y = 1)P(Y = 1)$ med $f_x(x)$ får man, enligt Bayes sats (Bayes 1763), sannolikheten $P(Y = 1|x)$, dvs sannolikheten att det är *Iris Setosa* om kronbladslängden är x . Om denna sannolikhet överskrider 50 % tar man beslutet att det är *Iris Setosa*, annars att det är *Iris Versicolor*. I det här fallet ger analysen alltså att man ska gissa på *Iris Setosa* om kronbladslängden ligger under ett värde som är cirka 5.5.

Vissa klassificeringsmetoder försöker direkt göra en skattning av $P(Y = 1|x)$,

t ex linjär logistisk regression, genom att modellera den som en kombination av en linjär avbildning (som beror på några parametrar w) och en mjuk stegfunktion. Genom att formulera en felfunktion $L(w, T_A)$ som beror på parametrarna w och träningsdata T_A och minimera den, dvs försöka hitta de parametrar w som gör felet L så litet som möjligt, så får man en klassificerare som förhoppningsvis fungerar bra för framtida beslut. Kombinationen av en linjär avbildning och en olinjär klassificering ser vi i flera andra metoder, t ex perceptronen (Rosenblatt 1957) och stödvektormaskiner (Vapnik 1963). Genom att koppla flera perceptronmoduler i ett nätverk kan man konstruera en maskininlärningsalgoritm (ett artificiellt neuralt nätverk), som kan approximera olinjära beslutsytor i rumden. Man kan se det som att man ansätter $P(Y = 1|x)$ som ett nätverk av omväxlande linjära funktioner och mjuka stegfunktioner. En sådan funktion kan ha tusentals variabler och miljontals parametrar. Faltning (Hilbert 1906) är en matematisk operation som enklast beskrivs som ett viktat rullande medelvärde. Om man ersätter de linjära operationerna i ett artificiellt neuralt nätverk med faltningar så får man ett faltningsnätverk (LeCun et al 2015). Analogt med linjär regression ställer man in parametrarna w genom att minimera en felfunktion $L(w, T_A)$ som också beror på träningsdata T_A .

För att använda sig av klassificering med maskininlärning så behöver man tala om vad som är rätt beslut för en stor mängd sinnesintryck. Det är emellertid inte alltid lätt att veta vad som är rätt beslut och det kan vara dyrt och tidskrävande att samla in data, säkerställa att de har rätt kvalitet och att ange rätt beslut/svar. Trots dessa svårigheter används övervakat lärande framgångsrikt i ett stort antal tillämpningar för att få maskiner att nå fram till någorlunda korrekt beslut.

Självkodare

En svårighet med att använda övervakat lärande är att man måste ange rätt beslut för ett stort antal sinnesintryck. Självkodning är namnet på system som från indata (t ex sinnesintryck) försöker återskapa samma sinnesintryck som utsignal. Det här blir ointressant och oanvändbart om man får lov att skicka vidare insignalen direkt till utsignalen. Däremot blir det intressant om man konstruerar systemet med en analys/kodningsdel och en syntes/avkodningsdel. Analysdelen tar sinnesintrycken och returnerar några få tal, som fungerar som en komprimerad representation av sinnesintrycken. Syntesdelen tar den komprimerade representationen och återskapar sinnesintrycken. Både analys- och syntesdelen innehåller många parametrar som "tränas" med hjälp av övervakat lärande. Fördelen här är att varje exempel på sinnesintryck kan användas både som indata och som facit i det övervakade lärandet. Självkodande kan vara en viktig komponent för en agent som försöker förstå världen. Den kodade representationen kan ibland vara enklare att använda för inlärning och tolkning.

Både analys och syntesdelen kan vara användbara. T ex kan syntesdelen användas för att generera nya syntetiska sinnesintryck (eller bilder i det här fallet). Ett intressant exempel på framgångsrik syntetisering av realistiska ansiktsbilder kan man hitta på <https://thispersondoesnotexist.com>. Varje gång man klickar på sidan får man en ny slumpmässig ansiktsbild (Karras 2017).

Världsmodeller

I ovanstående exempel var agentens samspel med omvärlden begränsat. Mer intressant blir det om de beslut agenten tar påverkar vilka sinnesintryck den får i framtiden. Det här blir relevant när man ska styra saker, t ex inom reglerteori. Ett exempel på detta är om man vill ha en agent som håller temperaturen vid 22 grader i ett hus. Agenten får temperaturmätningar som "sinnesintryck" och kan "ta beslut" om hur mycket värme som elementen ska ge från sig. Andra exempel är självkörande bilar och självflygande drönare.

Ett vanligt tillvägagångssätt är att en människa analyserar och kommer fram till en världsmodell V' som kan användas för att förutsäga hur kommande sinnesintryck (temperaturen) påverkas av agentens beslut. Om man har en världsmodell V' , så kan den användas för att ta fram en bra beslutsprocess, som i sin tur leder till kloka beslut baserat på vissa sinnesintryck.

Optimal reglerteknik utvecklades under slutet av 1950-talet. Genom att formulera beslutsproblemet så att man tar de beslut som minimerar ett felmått för ett dynamiskt system över tiden, kan man för vissa typer av problem räkna ut vad som är ett optimalt beslut eller en optimal reglerteknik. En lösning utvecklades i mitten av 1950-talet av Richard Bellman och andra genom att utvidga teorier från 1800-talet framtagna av Hamilton och Jacobi. Metoden för att lösa optimala beslutsproblem genom att lösa den så kallade Bellman-ekvationen blev känd som dynamisk programmering (Bellman 1957). Bellman introducerade också den diskreta, stokastiska modell som kallas Markovska beslutsprocesser; och Ron Howard (1960) utformade i sin tur en iterationsmetod för sådana Markovska beslutsprocesser.

Dual reglering

Men hur ska man bära sig åt, om det inte finns någon bra världsmodell V' ? Inom reglerteori finns det ett forskningsområde som heter dual reglering, och som utvecklades av Alexander Aronovich Feldbaum 1960 (Feldbaum 1960). Idén är att man har ett reglersystem med insignaler (sinnesintryck) och utsignaler (beslut). Utsignalerna påverkar verkligheten och indirekt insignalerna, som illustreras i figur 1. Om man har en tillförlitlig modell av hur verkligheten fungerar så kan man konstruera regleralgoritmen (b beslutsprocessen) så att den gör ett så bra jobb som möjligt, det vill säga blir optimal. Men om man inte vet

hur den verkliga processen fungerar kan man behöva experimentera för att få fram mer information om världen. Med bättre information kan man i ett senare skede styra processen bättre. Om systemets egenskaper är okända från början och kanske också ändras över tid har man två olika mål. Man måste både styra och undersöka, dvs

- använda kunskapen om hur världen fungerar för att uppnå sina mål (*styra*).
- testa och studera världen för att lära sig mer om hur systemet fungerar (*undersöka*).

De två handlingarna kan stå i konflikt med varandra. Om man bara styr, lär man sig mindre om världen och gör kanske misstag senare. Om man bara undersöker uppnår man inte sina mål.

Ett praktiskt exempel är om man ska ut och köra en ny bil en kall vinterdag. Då behöver man kanske prova att gasa, bromsa och svänga lite försiktigt för att kunna köra mer effektivt och säkert senare. Ett annat exempel är när man ska ställa in styrparametrar i en fabrik. Då behöver man också göra lite experiment för att sedan se vilka beslutsprocesser som fungerar bäst. Ett tredje exempel är om man ska bära en bricka med glas. När man lyfter upp brickan gör man det försiktigt och känner efter hur stadigt glaset står. Baserat på denna undersökning så får man mer information. Känner man att glaset står stadigt, vågar man gå raskt, men om det känns ostadigt får man ta det försiktigt. Ett fjärde exempel är inom medicinen. För att förbättra vården och utveckla nya rutiner så gör man ibland studier på ett mindre patientmaterial. Baserat på resultatet på studien så väljer man det som man tror är bäst för framtida patienter.

För enklare system, med ett fåtal in- och utsignaler och enklare modeller, kan man analysera problemet matematiskt och komma fram till optimala strategier. Optimalt kan i det här sammanhanget t ex formuleras som att felet i den egenskap man vill styra blir så litet som möjligt över tiden. Feldbaum visade att den optimala lösningen kan hittas genom dynamisk programmering. Genom att analysera de optimala strategierna kan man se att det mycket riktigt finns tillfällen då man ska undersöka och andra då det är bäst att styra (se t ex Åström & Helmersson 1986).

Förstärkningsinlärning

En besläktad frågeställning finns inom det vi kallar *förstärkningsinlärning* (Sutton & Barto 2011). Även här tänker man sig en agent som tar emot insignaler (dvs sinnesintryck) och skickar ut utsignaler (dvs fattar beslut), så som i figur 1. I likhet med dual reglering har man ett tydligt mål. Inom förstärkningsinlärning är målet formulerat som en belöning som agenten får utifrån.

Förstärkningsinlärning bygger vidare på metoder inom optimal reglerteori, dual reglering, men också från forskning inom psykologin för djurs lärande (Abbott i detta nummer).

Om antalet möjliga beslut är få, antalet möjliga insignaler inte är för många och om systemet, miljön man befinner sig i, går att modellera som en partiellt observerbar Markov-beslutsmodell, POMDP (Åström 1965), så finns det bra algoritmer som löser problemet effektivt. Dessa algoritmer (precis som för dual reglering) löser i princip problemet att både utforska/undersöka systemet och hantera det så att agenten efter ett tag har en ganska bra modell av systemet/världen och därefter kan styra ganska bra. Tack vare utvecklingen inom djupa faltningsnätverk har man kunnat konstruera approximativa lösningar, som även fungerar för lite större problem. Metoderna har varit väldigt framgångsrika, t ex för att konstruera en agent (ett datorprogram) som automatiskt spelar spel som exempelvis schack (Silver et al 2018) och go (Silver et al 2016).

Tillstånd eller minne

Många modeller använder sig av så kallade tillståndsmodeller. Man kan se tillståndet som ett sätt att lagra tidigare händelser i ett minne. För agenten består historien av alla tidigare beslut och sinnesintryck. En agents tillstånd skulle kunna bestå av all sparad historia, men ofta finns det mer effektiva sätt att lagra relevant information i ett tillstånd. Om den interna tillståndsmodellen är tillräckligt informativ räcker denna för att kunna modellera framtiden. För att automatiskt göra bra tillståndsmodeller kan man ibland använda sig av tekniker som självkodning. Man hushållar m a o med resurserna.

Världsmodeller, drömmar och beslutsträning

Det finns alltså flera exempel på att en agent kan skapa sig en modell av sinnesintrycken och dynamiken, t ex med hjälp av dual reglering eller förstärkningsinlärning. Men det finns fortfarande frågetecken om vad det är för målfunktion man ska ha. I dual reglering tänker man sig att det är ett reglerfel som ska minimeras. I förstärkande inlärning tänker man sig att det finns en väldefinierad belöningsfunktion.

Hur är det här kopplat till biologiska varelser och biologiskt beslutsfattande? För biologiska varelser är det inte helt tydligt vad målet är och vi är långt från att ha några som helst rimliga modeller för komplexa biologiska varelser, som människan.

Går det att få liknande resultat som för dual reglering och förstärkningsinlärning men utan externt definierade mål, och istället med internt formulerade mål eller åtminstone med andra principer?

I en färsk artikel (Ha & Schmidhuber 2018) föreslås en arkitektur där man

bygger upp en explicit modell av världen genom maskininlärning. Den består väsentligen av två delar. Den första är en självkodare, som tränas på ett stort antal sinnesintryck. Det gör att man får en mer komprimerad representation. Den andra delen är en dynamisk modell som "tränas upp" på tripplar av data (ett beslut samt sinnesintryck före och efter detta beslut). Den dynamiska modellen tar också fram en tillståndsmodell, som fungerar som ett minne för den dynamiska modellen. De två delarna tränas var för sig med maskininlärning. De är konstruerade så att de kan tränas på data som agenten själv kan ta fram genom att helt enkelt testa lite olika beslut. När de två delarna väl är inkörda, har agenten tillgång till en intern modell av världen som den kan använda för att prova olika scenarier, för att på så sätt utveckla en adekvat beslutsprocess.

En annan princip som tillämpats t ex i en artikel av Klyubin et al (2015) är att basera agentens inlärning på sk egenmakt. I artikeln utgår man från Gibsons princip om att kopplingen mellan sinnesintryck och beslut är centralt för en agents inlärning, men bygger vidare på tanken om att besluten är minst lika viktiga som sinnesintrycken och att det är viktigt att kvantifiera agentens förmåga att kontrollera sin situation. Agentens egenmakt mäts i bitar. Den är noll när agenten inte har någon kontroll över framtida sinnesintryck, men växer poängmässigt ju högre kännbart inflytande den har. Egenmakt kan också tolkas som den mängd information, som agenten potentiellt kan injicera i sin värld via sina beslut och senare fånga via sina sinnesintryck. Det har visat sig vara svårt att beräkna egenmakt, men i artikeln av Karl et al (2017) utvecklas metoder för att åtminstone approximativt göra just detta och för att sedan kunna använda dessa som principer för inlärning. I artikeln studerar författarna flera enkla dynamiska system. I exemplet "den svängande pendeln" har agenten störst egenmakt i det övre läget. Systemet både lär sig världsmodeller och att manövrera till detta övre läge. För ett annat exempel simuleras en tvåbent agent. Här visar det sig att det tillstånd med störst egenmakt råder, när agenten står upp. Egenmakten tillmäts ett högre värde i de tillstånd då agenten har stor handledningsfrihet, dvs när agenten inte är trängd i ett hörn eller befinner sig i en återvändsgränd.

Slutsatser

I det här kapitlet har vi studerat några olika matematiska/statistiska modeller för maskinellt – eller automatiserat – beslutsfattande. Även om kunskaperna, modellerna och metoderna för sådant autonomt lärande och beslutsfattande utvecklats enormt, används de för relativt enkla problem, ofta för spel där reglerna och systemen är relativt begränsade. De börjar dock användas i viss utsträckning även inom robotik, för självkörande fordon och för drönare. Problemet om automatiserat beslutsfattande beror på en mängd olika faktorer (Russel 2019). Även om dagens metoder är otillräckliga för att förstå mänskligt

beslutsfattande och beslutsfattande på en planetär nivå, så tror jag att det är värdefullt med kunskapsutbyte inom beslutsteori över traditionella ämnesgränser. Det skulle vara intressant att se hur metoderna skulle kunna användas inom statsvetenskapen och andra samhällsdiscipliner.¹

Referenser

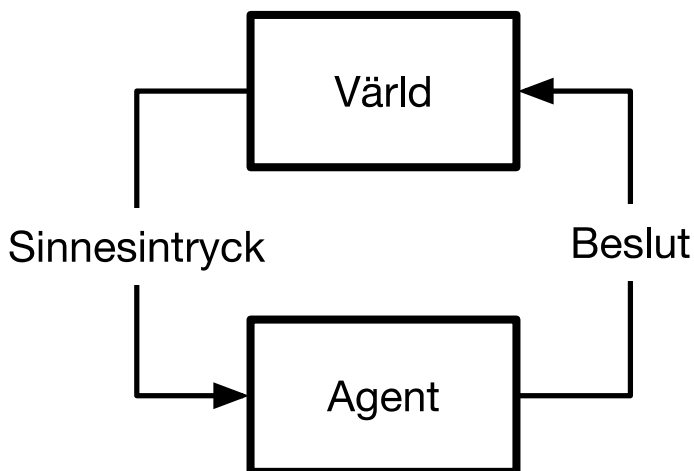
- Bayes, Thomas, 1763. "LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFRS", *Philosophical transactions of the Royal Society of London* 53, s. 370-418.
- Bellman, Richard, 1957. "A markovian decision process", *Journal of mathematics and mechanics*, s. 679-684.
- Feldbaum, Alexander A., 1960. "Теория Дуального управления", *Автоматика и Телемеханика* 21(9), s. 1240-1249.
- Fisher, Ronald A, 1936. "The use of multiple measurements in taxonomic problems", *Annals of eugenics* 7(2), s. 179-188.
- Gummeson, Anna, Arvidsson, Ida, Ohlsson, Mattias, Overgaard, Niels Christian, Krzyzanowska, Agnieszka., Heyden, Anders, Bjartell, Anders & Åström, Kalle, 2017. "Automatic Gleason grading of H and E stained microscopic prostate images using deep convolutional neural networks", *Medical Imaging 2017: Digital Pathology* 101400S.
- Ha, David & Schmidhuber, Jürgen, 2018. "World models", *ArXiv preprint arXiv:1803.10122*.
- Howard, Ronald A., 1960. *Dynamic programming and Markov processes*. John Wiley.
- Hilbert, David, 1906. "Grundzüge einer allgemeinen Theorie der linearen Integralgleichungen. Fünfte Mitteilung", *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, s. 439 - 480.
- Kaelbling, Leslie P. & Littman, Michael L. & Moore, Andrew W., 1996. "Reinforcement learning: A survey", *Journal of artificial intelligence research* 4, s. 237-285.
- Karl, Maximilian & Soelch, Maximilian & Becker-Ehmck, Philip & Benbouzid, Djalel & van der Smagt, Patrick & Bayer, Justin, 2017. "Unsupervised real-time control through variational empowerment", *ArXiv preprint arXiv:1710.05101*.
- Karras, Tero & Aila, Timo & Laine, Samuli & Lehtinen, Jaakko, 2017. "Progressive growing of GANs for improved quality, stability, and variation", *ArXiv preprint arXiv:1710.10196*.
- Klyubin, Alexander S., Polani, Daniel & Nehaniv, Chrystopher L., 2005. "Empowerment: A universal agent-centric measure of control", *2005 IEEE Congress on Evolutionary Computation* 1, s. 128-135.
- LeCun, Yann, Bengio, Yoshua & Hinton, Geoffrey, 2015. "Deep learning", *Nature* 521(7553), s. 436-444.
- Parzen, Emanuel, 1962. "On estimation of a probability density function and mode", *The annals of mathematical statistics* 33(3), s. 1065-1076.
- Rosenblatt, Frank, 1957. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory.

1 För en diskussion om de etiska implikationerna av detta, se Maria Hedlunds bidrag till detta nummer.

- Rosenblatt, Murray, 1956. "Remarks on some nonparametric estimates of a density function", *Annals of Mathematical Statistics* 27(3), s. 832–837.
- Russel, Stuart, 2019. "Human compatible: Artificial intelligence and the problem of control", *Viking, Penguin Random House*.
- Silver, David, Huang, Aja, Maddison, Chris J., Guez, Arthur, Sifre, Laurent, Van Den Driessche, George, Schrittwieser, Julian, Antonoglou, Ioannis, Panneershelvam, Veda & Lanctot, Marc et al., 2016. "Mastering the game of go with deep neural networks and tree search", *Nature* 529 (7587), s. 484.
- Silver, David, Hubert, Thomas, Schrittwieser, Julian, Antonoglou, Ioannis, Lai, Matthew, Guez, Arthur, Lanctot, Marc, Sifre, Laurent, Kumaran, Dharshan & Graepel, Thore, et al., 2018. "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play", *Science* 362(6419), s. 1140–1144.
- Simonsson, Per, 2018. *CellaVision Story: Innovation, människor & miljö*. Lund: Historiska Media.
- Sterling, Peter & Laughlin, Simon, 2015. *Principles of neural design*. Cambridge: MIT Press.
- Ström, Jakob, Åström, Kalle & Akenine-Möller, Tomas, 2017. *Immersive linear algebra*, <http://immersivemath.com/ila>.
- Sutton, Richard S. & Barto, Andrew G., 2011. *Reinforcement learning: An introduction*. Cambridge: The MIT Press.
- Vapnik, Vladimir, 1963. "Pattern recognition using generalized portrait method", *Automation and remote control* 24, s. 774–780.
- Åström, Karl Johan, 1965. "Optimal control of markov processes with incomplete state information", *Journal of mathematical analysis and applications* 10(1), s. 174–205.
- Åström, Karl Johan & Helmersson, Anders, 1986. "Dual control of an integrator with unknown gain", *Computers & Mathematics with Applications* 12(6), s. 653–662.

Figurer

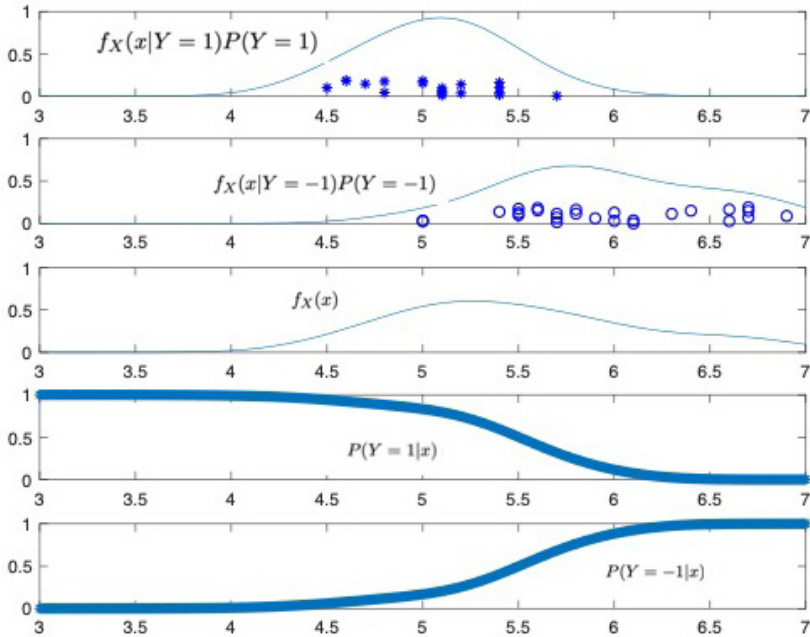
Figur 1. I den här texten presenterar vi några olika modeller för hur agenter fattar beslut i en värld baserat på sinnesintryck.



Figur 2. Inom övervakat lärande tränar man upp hur beslutsprocessen för en agent ska vara baserat på ett stort antal exempel på korrekta beslut för olika sinnesintryck. Man har då oftast inte en explicit modell hur världen fungerar och är inte alltid intresserad av återkoppling eller tidsaspekt.



Figur 3. Figuren illustrerar några av stegen i hur man tränar en klassificerare. De två översta graferna visar exempeldata och sannolikhetsfördelningar för två klasser. Mittenraden visar den totala sannolikhetsfördelningen. De två understa graferna visar a posteriorifördelningarna för de två klasserna.



Figur 4. En självkodare innehåller både en analys/kodningsdel och en syntes/avkodningsdel. Analysdelen tar sinnesintryck och returnerar några få tal. Syntesdelen tar den komprimerade representationen och återskapar sinnesintrycken. Exemplet är hämtat från Ström et al (2015).

