

Strojno učenje

Uvod

Tomislav Šmuc

- Opservacije o Strojnom Učenju
- Oblici Strojnog Učenja: Tipologija i podjele
- Osnovni pojmovi

Uloga strojnog učenja

- **Jezik računarstva (računalni jezici, algoritmi, baze podataka)**
 - **najbolje je što imamo za opisivanje procesa (prirodnih ali i umjetnih sistema) spremanja, manipuliranja i korištenja podataka, informacija i znanja.**
- **Algoritmi za procesiranje podataka/informacija predstavljaju:**
 - **za biološke, kognitivne i socijalne znanosti ono što npr. algebra predstavlja za klasičnu fiziku**

Zašto (proučavati) strojno učenje?

Znanstveni izazovi:

- Kako uče životinje ili ljudi
- Zahtjevi da bi se nešto moglo naučiti: Precizni uvjeti pod kojima su neki ciljevi učenja ostvarivi
- Kako poboljšati učenje – aktivno i pasivno učenje
- Računalne arhitekture strojnog učenja

Korisne primjene:

- Medicinska dijagnostika (EKG, EEG, mamografija)
- Otkrivanje novih zakonitosti u znanosti
- “Spam Filtering”, prijevare (npr. kreditne kartice), upadi u rač.sustave
- pametno pretraživanje – sustavi za preporučivanje (google; amazon)
- automatsko prevođenje
- praćenje objekata u slikama & videu



Strojno učenje: usko vezane discipline


- Data mining – rudarenje podataka;
- Data science
- Adaptivno procesiranje signala
- Probabilističko zaključivanje (Bayesove mreže)

Razlika između statistike i strojnog učenja

- **Primijenjena statistika:**
 - obično primjenjujemo na "malim" skupovima podataka
 - uloga statističara je velika – računalo je pomoćni alat
- **Strojno učenje - naglasak je:**
 - na automatiziranju otkrivanja i korištenja pravilnosti u podacima;
 - metodama koje garantiraju kvalitetu naučenih model;
 - karakteriziranju što je „naučljivo” i pod kojim uvjetima;


Učenje = Memoriranje + zaključivanje

Zaključivanje $\forall x \text{ Položio}(x, \text{Ispit}) \Rightarrow \text{Prijavio}(x, \text{Ispit})$

Dedukcija  $\text{Položio}(\text{Pero}, \text{Ispit})$


 $\text{Prijavio}(\text{Pero}, \text{Ispit})$

...koristi strukturu kojom se generiraju istiniti zaključci /zaključak slijedi iz premisa/

Indukcija  $\text{Položio}(\text{Pero}, \text{Ispit}) \wedge \text{Prijavio}(\text{Pero}, \text{Ispit})$
 $\text{Položio}(\text{Ivica}, \text{Ispit}) \wedge \text{Prijavio}(\text{Ivica}, \text{Ispit})$
 $\neg \text{Položio}(\text{Kreso}, \text{Ispit}) \wedge \text{Prijavio}(\text{Kreso}, \text{Ispit})$
 $\neg \text{Položio}(\text{Ante}, \text{Ispit}) \wedge \neg \text{Prijavio}(\text{Ante}, \text{Ispit})$

 $\forall x \text{ Položio}(x, \text{Ispit}) \Rightarrow \text{Prijavio}(x, \text{Ispit}) ?$

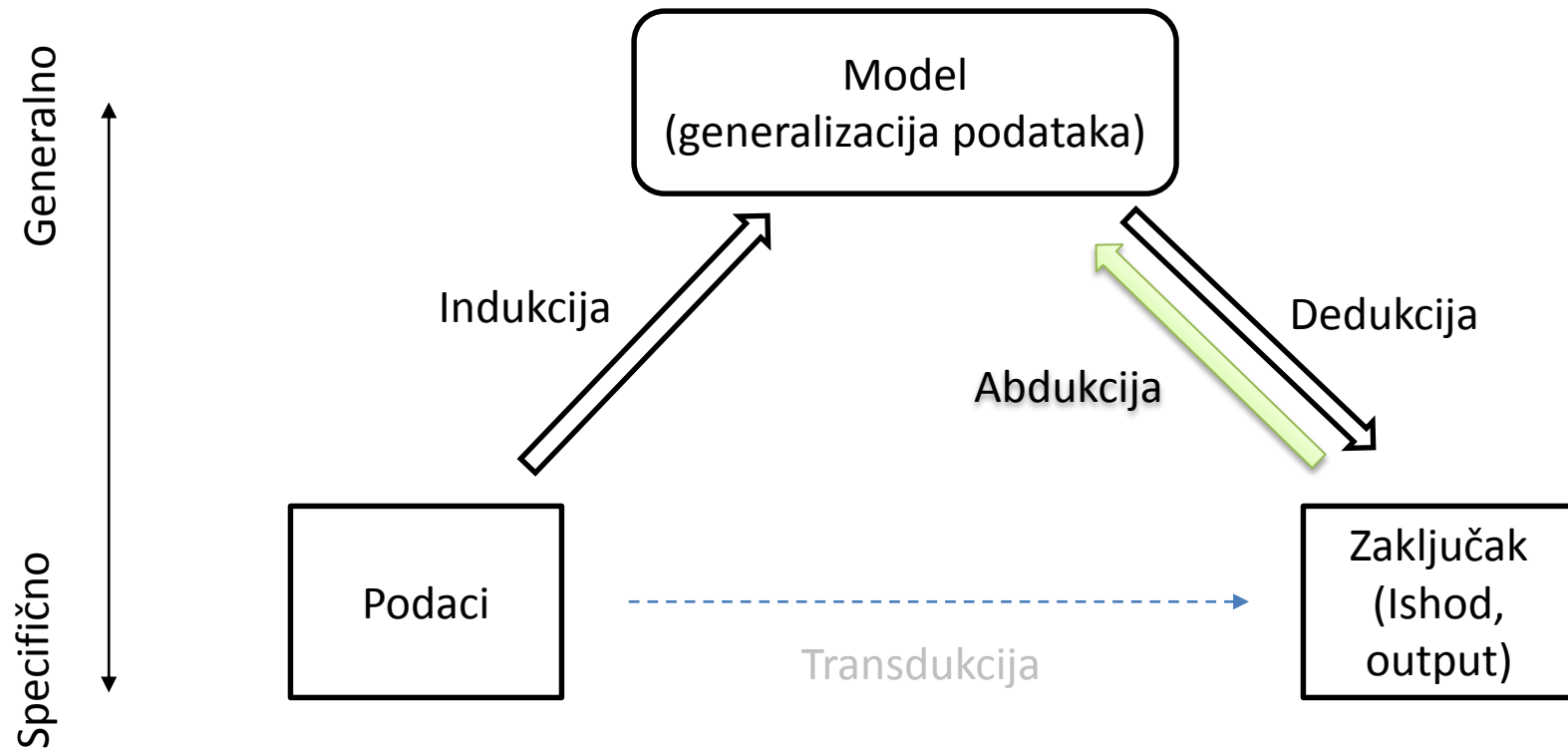
.... zaključivanje na osnovu opservacija/primjera
 ind. zaključivanje => nema garancije da su 100% točne (određena vjerojatnost da su točne)

Abdukcija  $\forall x \text{ Položio}(x, \text{Ispit}) \Rightarrow \text{Prijavio}(x, \text{Ispit})$
 $\text{Prijavio}(\text{Joža}, \text{Ispit})$

 $\text{Položio}(\text{Joža}, \text{Ispit})$

Zaključivanje vodi ka najvjerojatnijem mogućem objašnjenju

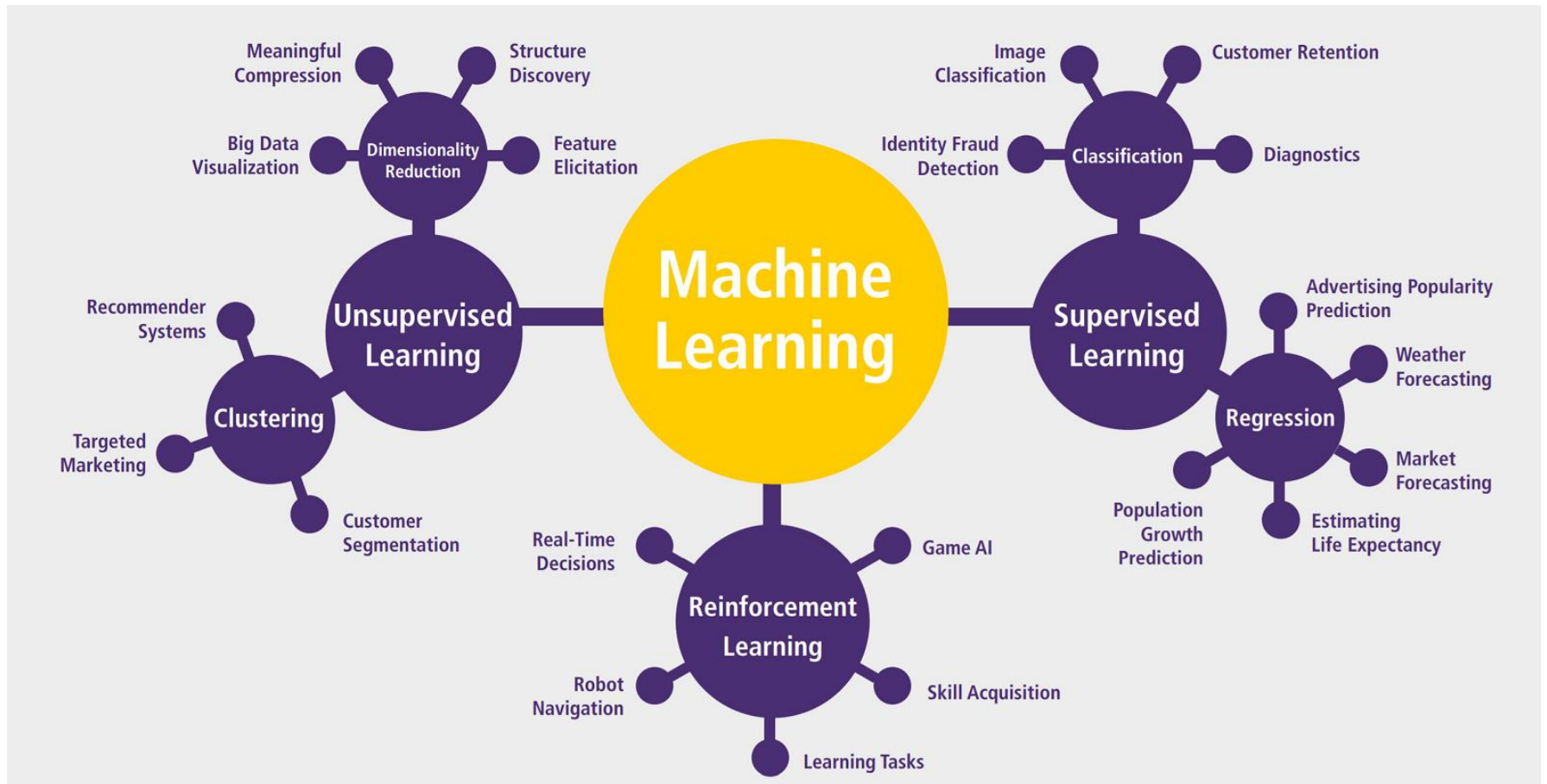
Strojno učenje, indukcija i dedukcija



Abdukcija – rijetko korištena u strojnom učenju
(...ali često u znanosti!)

Mogu li strojevi učiti bolje:

- Neki zadaci su najbolje definirani primjerima (npr. dijagnoza)
- Ako imamo velike količine podataka – one mogu biti vrlo korisne pravilnosti ili prediktivne relacije (data mining)
- U realnim okruženjima karakteristike kompleksnih sustava (npr. promet u gradu) se brzo mijenjaju – software koji ima mogućnost brzog adaptiranja može povećati korisnost/sigurnost



Jha, V. An overview of machine learning techniques

Oblici strojnog učenja – I podjela

Nadzirano učenje (en. Supervised Learning)

- eksplicitna informacija o primjerima i vrijednosti njihove ciljne varijable (en. label)
- cilj: napraviti model koji će raditi predikcije na još neviđenim (novim) primjerima (klasificiranje objekata, predikcija prodaje, cijene ...)
 - Klasifikacija
 - Regresija
 - Predikcija – predviđanje (en. Forecasting) – vremenske serije

Nenadzirano učenje - Unsupervised Learning:

- imamo samo primjere, bez ikakve anotacije ili povratne informacije o njihovoj kategorizaciji
- cilj : grupirati primjere, otkriti neku strukturnu pravilnost u podacima, projekcija podataka u niže-dimenzionalne prostore
 - Grupiranje - Clustering
 - Otkrivanje-detekcija iznimaka (Outlier detection)
 - Kompresija podataka

Oblici strojnog učenja (nastavak)

Učenje s podrškom (povratnom vezom - en. reinforcement learning):

- dobivamo “nagradu” s odgodom ukoliko akcije predviđene našim modelom daju uspjeha
- nema eksplicitnog nadzora
- obično vezano uz učenje sekvenci akcija (igre, roboti!)
(AlphaGO!)

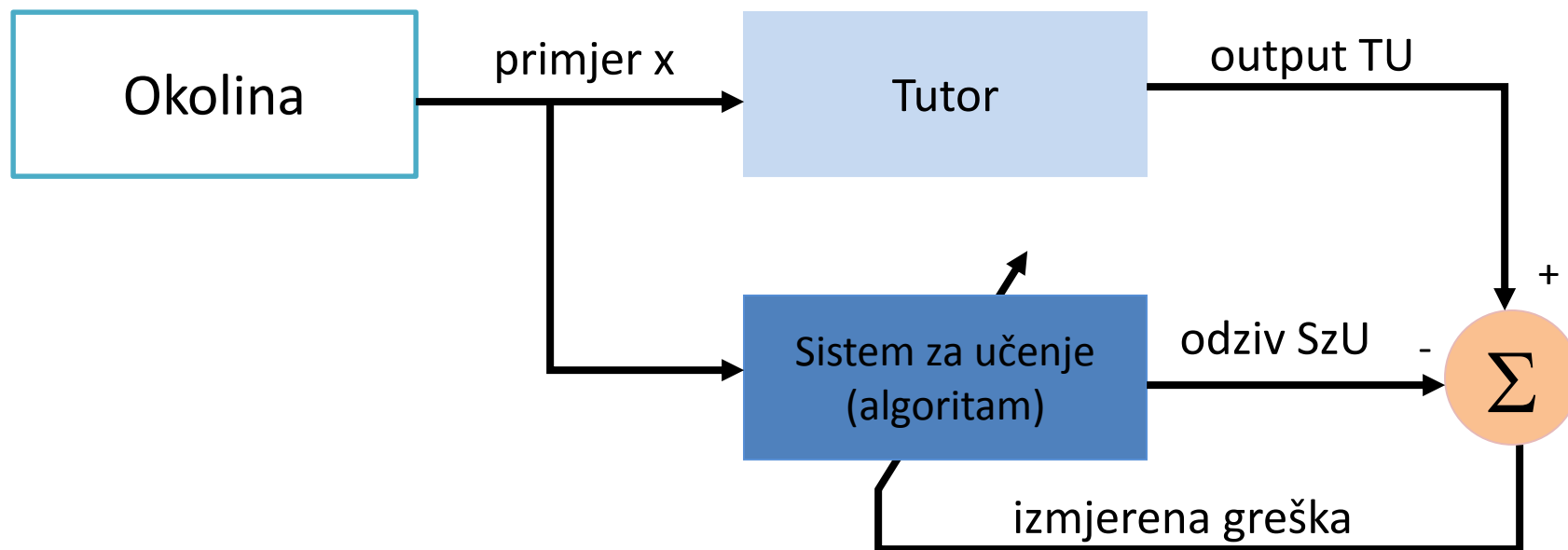
Učenje skupa pravila

- iz danog većeg skupa mjerenja ili transakcija, otkriti učestalo ponavljane uzorke mjerenja (npr. asocijativna pravila)

{Mlijeko, Jogurt} → Kruh [support=5%, confidence=80%]

Nadzirano učenje (učenje uz učitelja/tutora)

- Znanje – svodi se na primjere $(input_i, output_i) = (x_i, y_i)$
- Cilj: minimizirati grešku => razliku između stvarnog outputa (učenik) i željenog outputa (učitelj)



Nadzirano učenje (učenje uz učitelja/tutora)

Klasifikacija:

- Output je kategorički
- Input može biti bilo što
- Cilj - da naučeni model odabire korektnu klasu

Predikcija:

- klasifikacija/regresija
 - npr. vezana uz vremenski ovisne događaje
 - odrediti klasu/output koristeći nove ulazne sekvence (podatke) kao i neke prethodne sekvence/podatke i njihove klase/output na nekim prethodnim sekvencama

Regresija - aproksimacija funkcija

Cilj – aproksimirati nepoznatu funkciju $f_n(\mathbf{x})$

- tako da je preslikavanje $F_a(\mathbf{x})$ realizirano sistemom za učenje približno isto kao i ono nepoznate funkcije $f_n(\mathbf{x})$:

$$|F_a(\mathbf{x}) - f_n(\mathbf{x})| < \varepsilon \quad \forall \mathbf{x}$$

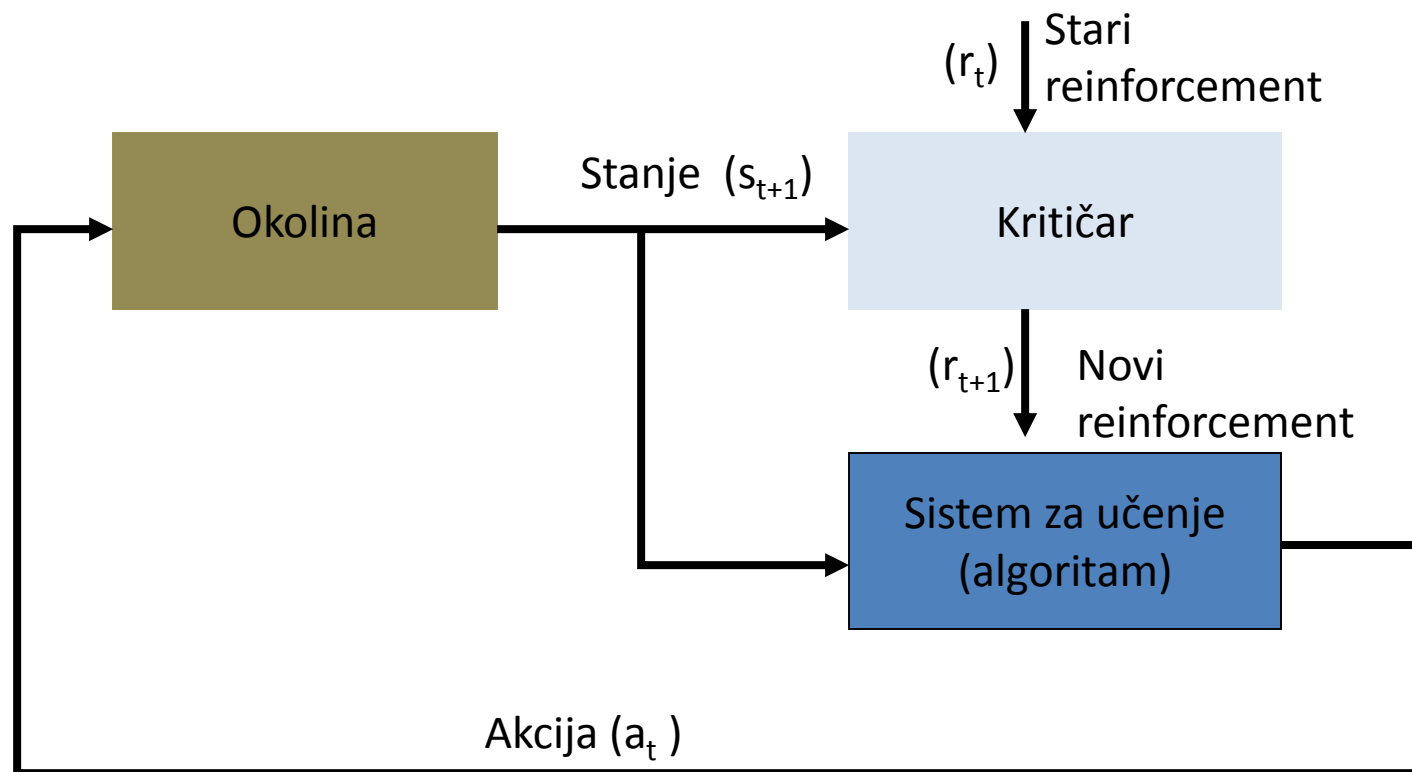
Primjer:

Opis input-output odnosa nekog nekog složenog sustava kod kojeg je output kontinuirana veličina

- ritam srca u ovisnosti o različitim karakteristikama organizma, uvjetima i podražajima
- energija oslobođena u ložištu peći

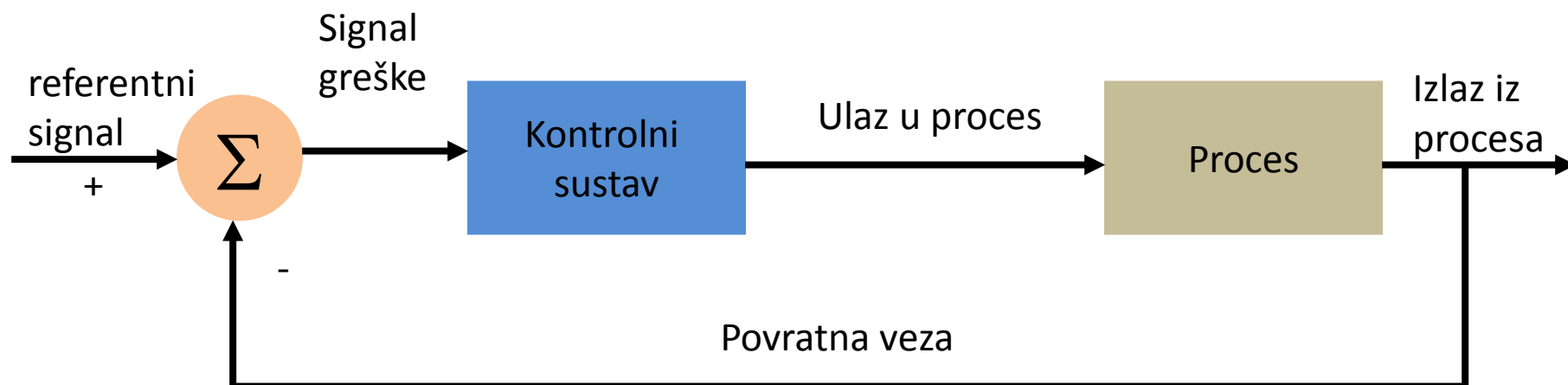
Učenje podrškom - reinforcement learning – učenje niza akcija

- učenje – kroz interakciju s okolinom (ili preko simulacije interakcije)
- istraživanje odnosa stanja i akcija
- Cilj: maksimiziranje akumuliranih budućih “reinforcements”



Primjer: kontrola procesa – učenje funkcije kontrole

- Podesiti parametre nekog kontrolnog sustava tako da na kraju vodi proces na optimalan način



Primjeri učenja kontrole (Reinforcement Learning)

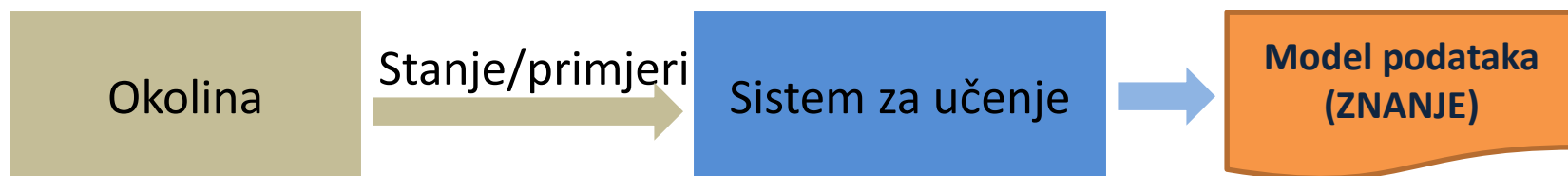
- Navigacija robota
- Učenje akcija koje maksimiziraju/optimiraju output nekog pogona (tvornice), procesa trgovanja dionicama na burzi....
- Učenje poteza u igri (šah, backgammon)

Karakteristike ovih problema:

- Odgođeno nagrađivanje umjesto instantnog - za dobre akcije (tzv. temporal credit assignment problem)
- Nema pravog nadzora – supervizije procesa učenja (primjeri su u obliku stanje, akcija)
- Postoji potreba za aktivnim istraživanjem prostora stanja i akcija

Nenadzirano učenje

- nema učitelja niti kritičara
- samo-organizirajuća svojstva
- mjera dobrote/kvalitete – neovisna o zadatku
- potrebno je naći pravilnosti u podacima – otkriti klase automatski – te otkriti što one znače ili predstavljaju!



Primjeri:

- segmentacija/grupiranje primjera
- otkrivanje izuzetaka (outlier detection) u podacima
- preporučivanje proizvoda (*recommender sys*)

Vrlo poopćena definicija strojnog učenja

Poboljšati kvalitetu izvršavanja zadatka t ,

s obzirom na definiranu mjeru uspješnosti q ,

na osnovu dostupnog iskustva (znanja i podataka) e

Primjeri:

t : prepoznavanje slova u tekstu

q : postotak točno prepoznatih slova

e : baza slika sa ručno napisanim tekstovima (pixel po pixel)

t : kome ponuditi novu policu osiguranja

q : postotak točno određenih kandidata iz baze poznatih primjera

e : baza podataka o klijentima osiguravajućeg društva

•!

t : igranje šaha (povlačenje pravih poteza)

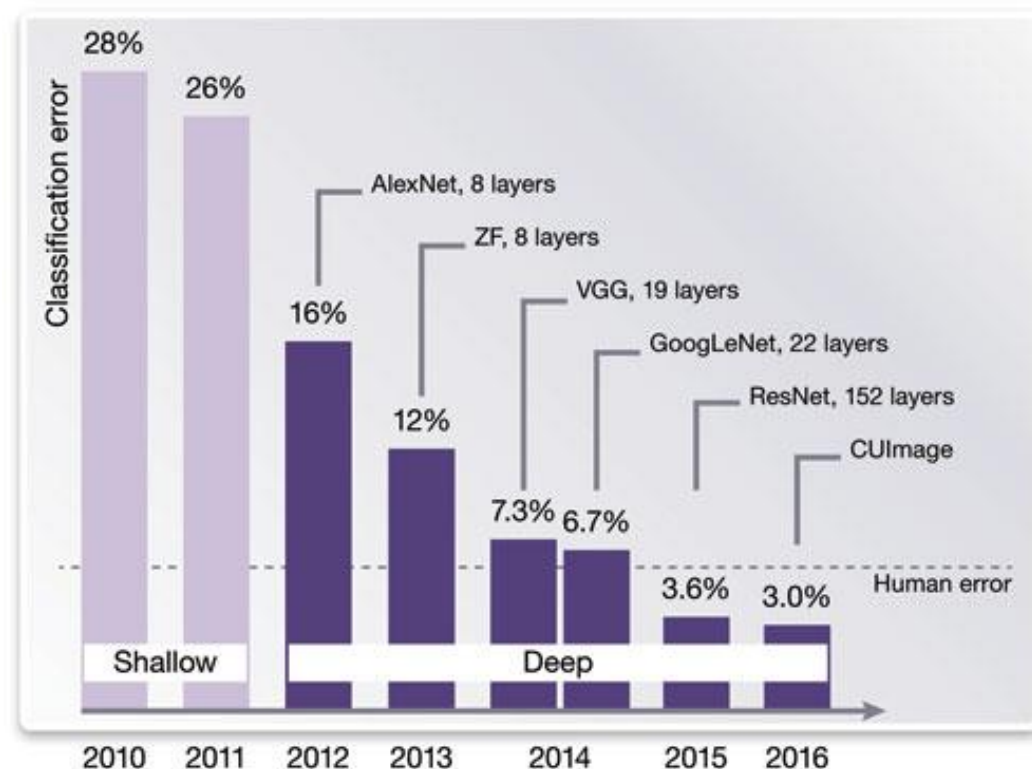
q : omjer dobivenih i izgubljenih igara

e : igranje igara protiv samog sebe

Novije primjene strojnog učenja (Duboke rekurentne mreže)

- Text-to-speech synthesis
- Language translation
- Image caption generation
- Learning to program...

ILSVRC – ImageNet Large Scale
Visual Recognition Challenge



Stvaranje modela iz skupa primjera

1. Pretpostavka: postoji distribucija vjerojatnosti $p(\mathbf{x}, y)$ čiji pravi oblik neznamo

Obično na raspolaganju imamo tek (konačan) skup primjera (parova \mathbf{x} i y) (obično uz određeni “šum” ili greške):

$$\{\mathbf{x}_1, y_1; \dots; \mathbf{x}_n, y_n; \dots; \mathbf{x}_N, y_N\}$$

Primjeri iz skupa su iid (independently and identically distributed)

2. Koristeći samo $\{\mathbf{x}_1, y_1; \dots; \mathbf{x}_n, y_n; \dots; \mathbf{x}_N, y_N\}$ generiramo model M koji za neki $\{\mathbf{x}_n\}$ daje $\{z_n\}$ (po mogućnosti tako da je što češće $z_n = y_n$, t.j. da model što manje griješi)
3. Zamislimo situaciju da je neki novi skup primjera $\{\mathbf{x}_k, y_k\}$ “izvučen” iz distribucije $p(\mathbf{x}, y)$. Naš model koristimo za određivanje $\{z_k\}$. (ukoliko znamo stvarni y_k možemo odrediti i grešku na novom, tzv. testnom skupu)

Osnovni problemi u ovom postupku:

- Koji algoritmi i u kojim uvjetima dobro aproksimiraju funkcije i pod kojim uvjetima?
- Kako broj primjera utječe na točnost modela?
- Kako kompleksnost reprezentacije mogućih modela/hipoteza utječe na točnost?
- Kako šum ili greške utječu na točnost?
- Koji su teoretski limiti “naučljivosti”?

Generalizacija

Osnovni problem train/test uzorkovanja i učenja:

- opasnost da ćemo dobiti model koji odlično “radi” samo na podacima na kojima je *istreniran*

=> **overfitting** (ili suprotno od generalizacije)

- učenje primjera napamet = savršen rezultat na skupu za učenje
= slučajno pogađanje na novim primjerima

Generalizacija ~ sposobnost dobre predikcije na novim primjerima !

Kapacitet = kompleksnost (prostora) hipoteza

Osnovna pretpostavka induktivnog učenja

=> generalizacija je moguća

Ako naš model dobro radi na većini podataka na kojima je treniran, te ako nije prekompleksan – vjerojatno je da će dobro raditi i na novim podacima (napomena: ako su generirani iz iste distribucije kao i podaci za treniranje...)

Ova empirijska izjava je inače prilično dobro formalizirana u više desetaka godina istraživanja u području teorije strojnog učenja...
(SRM&VC-dimension, PAC learning, Occam's razor, MDL....)

Induktivna pristranost (inductive bias)

Posljedica osnovne hipoteze induktivnog učenja:

- generalizacija je moguća samo ako unaprijed radimo neke pretpostavke o konačnom izgledu hipoteze/modela odnosno ako je kompleksnost mogućih hipoteza/modela ograničena !
- **algoritam/učenik koji nema ugrađenu induktivnu pristranost ne može generalizirati !**

Induktivna pristranost – kako se postiže?

- Učenje napamet – nema induktivne pristranosti!
- Linearna regresija – odnos između ulaznih varijabli i ciljne varijable je linearan
- Stabla odlučivanja – preferiranje plitkih stabala
- Naivni Bayes – ulazne varijable su međusobno nezavisne
- SVM – maksim. margine između različitih klasa primjera
- Regularizacija – penaliziranje složenosti modela (/hipoteza) u funkciji cilja (strojno učenje ~optimizacijski problem)

Što čini metodu/algoritam strojnog učenja ?

- **reprezentacija modela/ciljne funkcije**
(linearna funkcija, polinom, neuralna mreža....)
(sa nepoznatim vrijednostima slobodnih parametara)
- **algoritam pretraživanja/optimizacije**
(strojno učenje = pretraživanje/optimizacijski problem)
- **metoda procjene greške** na “neviđenim” primjerima –
skalarna **funkcija cilja** kojom ćemo kvantificirati kako
dobro radi/generalizira naš naučeni model

Cilj: podesiti vrijednosti parametara modela tako
minimiziramo vrijednost funkcije cilja

Generalni oblik funkcije cilja

$$\phi(\mathbf{X}, \mathbf{Y}; \mathbf{w}) = L(\mathbf{X}, \mathbf{Y} | \mathbf{w}) + R(\mathbf{w})$$

\mathbf{w} – parametri modela

$L(\mathbf{X}, \mathbf{Y} | \mathbf{w})$ – mjera greške modela (loss function, npr RMSE)

$R(\mathbf{w})$ – funkcija kojom se penalizira kompleksnost modela

- oblik $\phi(\mathbf{X}, \mathbf{Y}; \mathbf{w})$ opet govori o tome da želimo postići što manju grešku na skupu za učenje (L) ali isto tako da to želimo postići sa što jednostavnijim modelom \mathbf{w} !