Strojno učenje

Tomislav Šmuc

Pravila igre .....

Strojno učenje (61529)

T. Šmuc, M. Bošnjak

Elementi ocjenjivanja

1. sudjelovanje u nastavi: 10%,
2. kolokvij: 30%,
3. projektni zadatak: 30%,
4. završni ispit: 30%.

Raspodjela bodova
x% = max bodova za određenu aktivnost

Tablica ocjenjivanja

Min =(50% od ukupnog broja sati predavanja i vježbi,
50% od ukupnog mogućeg broja bodova)

Prolazne ocjene:

50% - 60%	dovoljan (2),
61% - 73%	dobar (3),
74% - 86%	vrlo dobar (4),
87% - 100%	izvrstan (5)

Pravila igre .....

Strojno učenje (61529)

T. Šmuc, M. Bošnjak

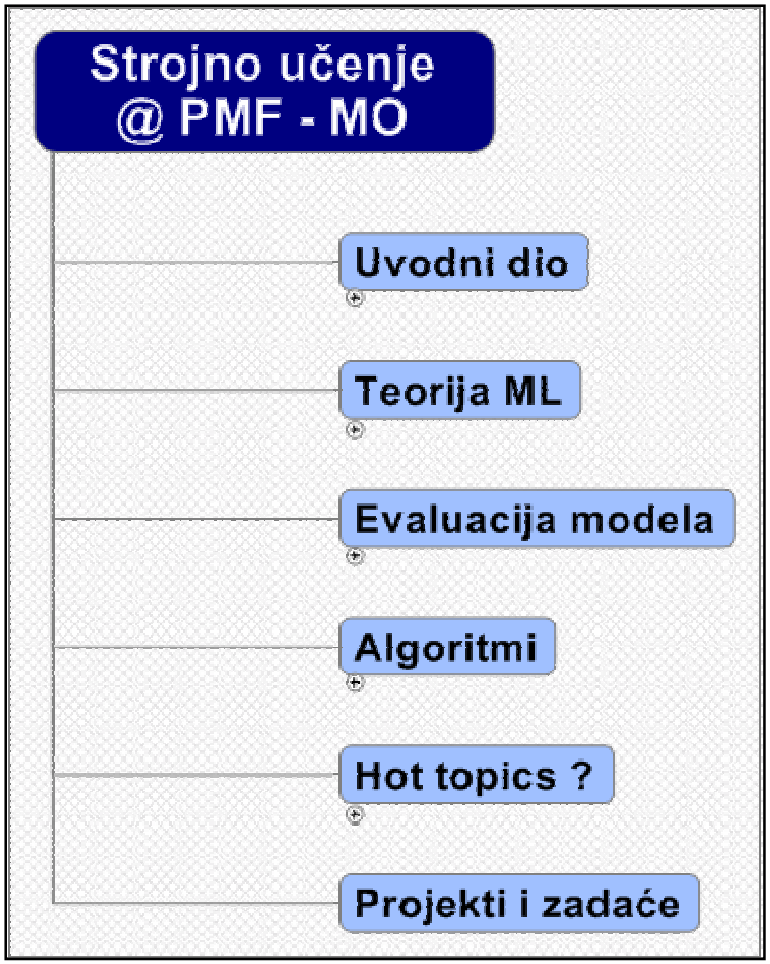
Nedoumice:

(ovisno o broju studenata)

- kolokvij vs. Zadaće
- projektni zadatak (jedan ili više studenata)
- vježbe – WEKA (+R?)

Sve informacije o predmetu u toku semestra mogu se naći na:
http://web.math.hr/nastava/su/

Pravila | Sadržaj



Uvodni dio

Opservacije o SU
Oblici SU:Tipologija i podjele
Osnovni pojmovi
Struktura problema strojnog učenja

Teorija strojnog učenja

- PAC learning
- Structural risk minimization (VC dimenzija)
- Bayes, bias & variance
- MDL
- Evaluacija modela
  - Nadzirano učenje: T&T, Xv, OOB
  - Ne-nadzirano učenje (CVI) – možda ?

Algoritmi

Nadzirano učenje

Većinu od:

- FIND-S; Candidate elimination
- Decision trees
- Naive Bayes
- k-nn
- Perceptron/Winnow/Neuronske mreže
- Rule learning
- Kernel machines; SVM
- Ensembles: RF (PARF)
  - Resampling based methods: boosting, bagging, stacking

Nenadzirano učenje

- “Clustering primjera”:
  - k-means
  - Hijerarhijske metode
  - (SOM)
- “Redukcija dimenzija ~ Clustering atributa”:
  - PCA
  - (zadaće? - ICA, “manifold learning”)
- Asocijativna pravila
  - APRIORI algoritam

Projekti / zadaće

Zasad još samo u obliku ideja:

- **Po principu: Čitaj/programiraj/testiraj/piši**
- **Hot topics:**
  - "Novi" algoritmi – realizacija
  - Specijalni problemi, testovi i usporedbe
  - "DM/ML/KD Challenges" (DMC 2011, KDDCup 2011, Kaggle, TunedIT !)
  - Prema posebnim dogovorima.....

Osnovna

- Machine Learning, Mitchell
- Elements of Statistical Learning, Hastie, Tibshirani, Friedman (downloadable!)

Preporučljivo

- Pattern Recognition, (2nd ed) Duda,Hart & Stork
- Neural Networks for Pattern Recognition, Bishop

+ VideoLectures.Net (MIT, CMU, Stanford...)

Članci – uz pojedina predavanja

Opservacije

- **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 informacija predstavljaju:**
  - **za biološke, kognitivne i socijalne znanosti ono što npr. algebra predstavlja za klasičnu fiziku**.

Konceptualne razlike u pogledu na svijet:

"Doba fizike" (1900-1950)

- Fokus na fizikalnu bazu svemira
- Naglasak na objašnjavanju svijeta (fenomena) fizikalnim procesima

"Doba računarstva" (1950- ....)

- Fokus na informacijskom (algoritamskom) objašnjavanju prirodnih fenomena
- Procesima koje podržavaju (inteligentna) bića: skupljanje, spremanje, procesiranje i korištenje podataka/informacija/znanja

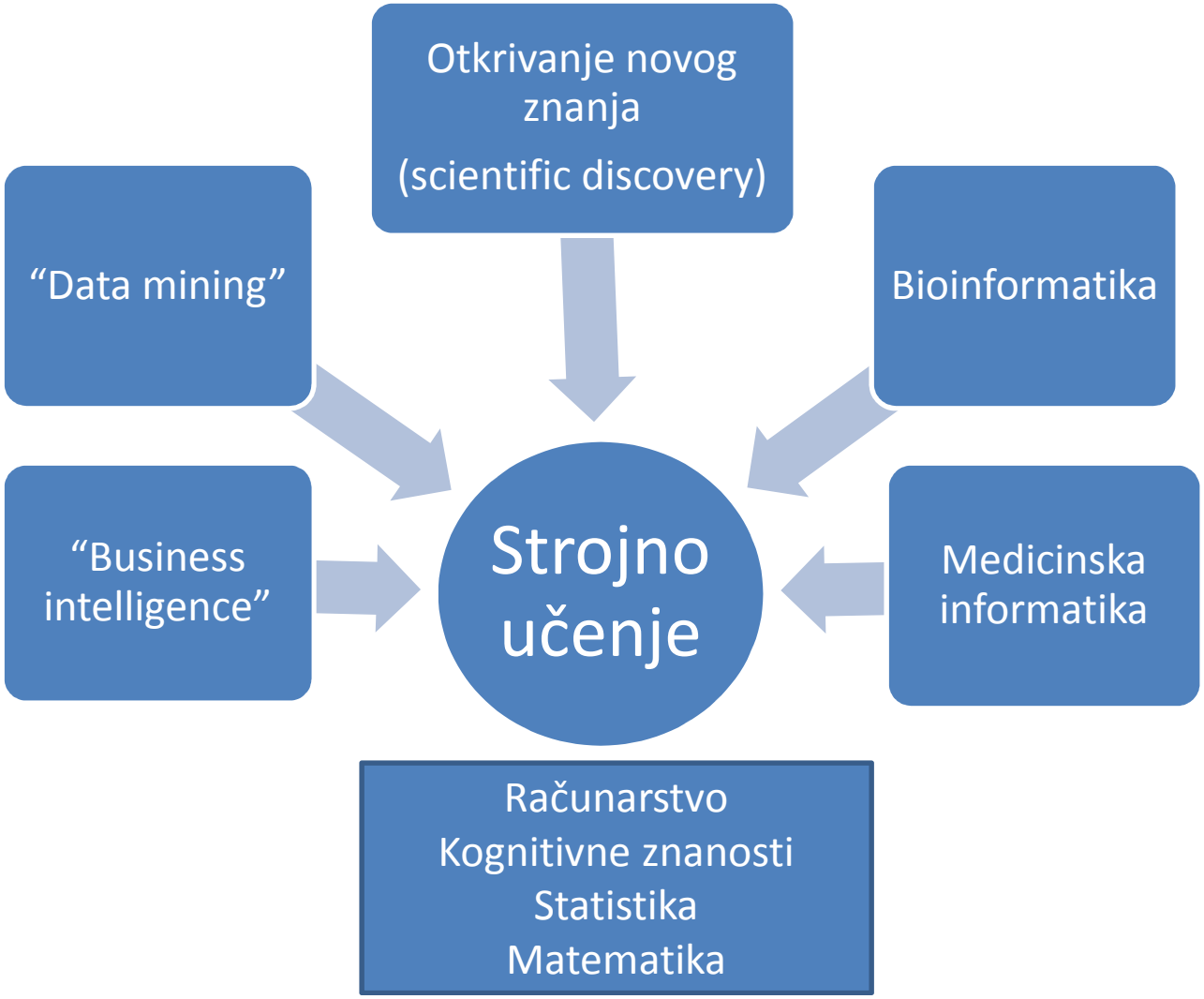
Zašto (proučavati) strojno učenje?

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 (npr. mamografija)
- Otkrivanje novih zakonitosti u znanosti (Scientific discovery)
- "Spam Filtering", prijevare (npr. kreditne kartice), upadi u rač.sustave (npr. DOS attacks)
- pametno pretraživanje – sustavi za preporučivanje (google; amazon)
  - (searchpoint.si;)
- automatsko prevođenje govora & prepoznavanje govornika & razdvajanje više istovremenih govornika
- lociranje/identificiranje/praćenje objekata u slikama & na filmu & online (vidi DARPA challenge)


Otkrivanje novog znanja
(scientific discovery)
"Data mining"
Bioinformatika
"Business intelligence"
Strojno učenje
Medicinska informatika
Računarstvo
Kognitivne znanosti
Statistika
Matematika

Strojno učenje: usko vezane discipline

- Data mining – rudarenje podataka;
- 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;
  - karakteriziranju što je naučljivo i pod kojim uvjetima;
  - metodama koje garantiraju kvalitetu naučenih modela

Što je učenje? = Memoriranje + zaključivanje

Zaključivanje $\forall x$ *Položio (x,Ispit)* $\Rightarrow$ *Prijavio(x, Ispit)*

Dedukcija ⇒

Položio(Pero, Ispit)

Prijavio (Pero, Ispit)

Indukcija ⇒

Položio(Pero, Ispit) $\wedge$ *Prijavio (Pero, Ispit)*

Položio(Ivica, Ispit) $\wedge$ *Prijavio (Ivica, Ispit)*

$\neg$ *Položio(Kreso, Ispit)* $\wedge$ *Prijavio (Kreso, Ispit)*

$\neg$ *Položio(Ante, Ispit)* $\wedge \neg$ *Prijavio (Ante, Ispit)*

.....

$\forall$ *x Položio (x,Ispit)* $\Rightarrow$ *Prijavio(x, Ispit) ?*

Abdukcija ⇒

$\forall$ *x Položio (x,Ispit)* $\Rightarrow$ *Prijavio(x, Ispit)*

Prijavio(Jožа, Ispit)

Položio (Jožа,Ispit)

Mogu li strojevi učiti bolje:

•Neki zadaci su najbolje definirani primjerima (npr. dijagnoza)

•Ako imamo velike količine podataka – one mogu kriti 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 ....

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 – 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 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)

Učenje skupa pravila

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

II podjela (Problemi i zadaci)

Problemi

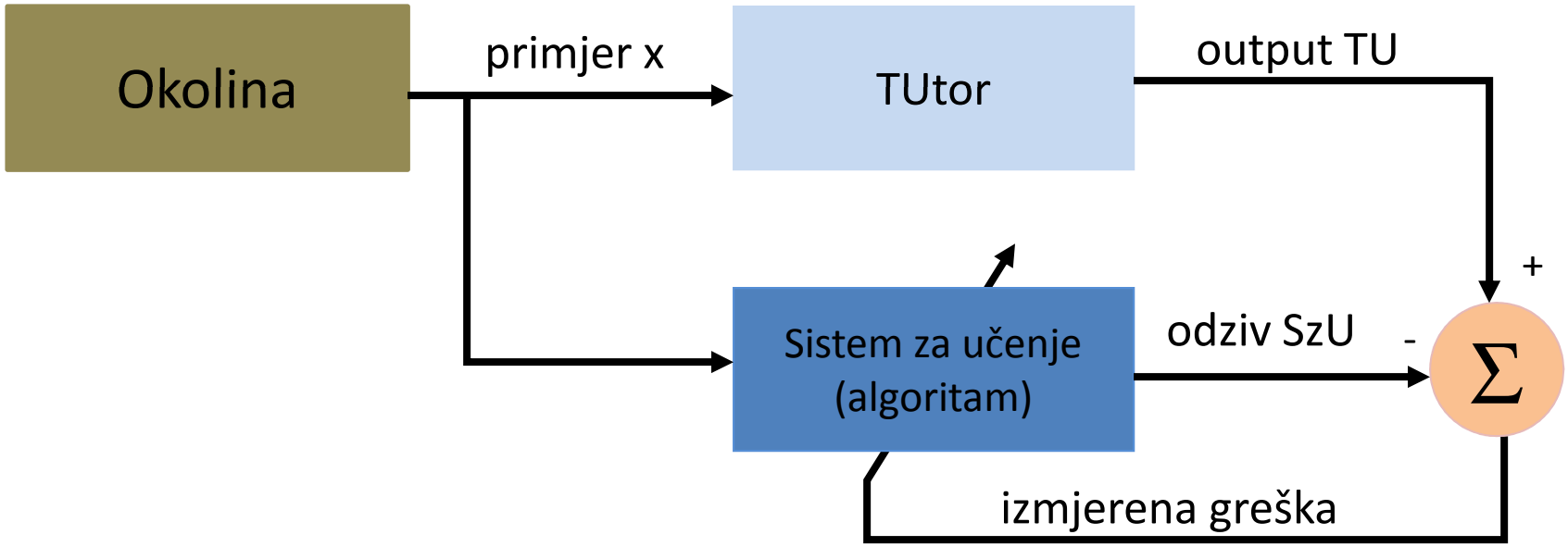
1. Učenje uz “učitelja” (nadzirano učenje)
2. Učenje uz “kritičara” (reinforcement learning)
3. Učenje bez nadzora (nenadzirano učenje)

Zadaci

Prepoznavanje uzorka (klasifikacija)
Povezivanje uzoraka (učenje pravila)
Aproksimacija funkcija (regresija)
Kontrola ili upravljanje procesima
Filtriranje (redukcija varijabli/atributa/dimenzija)

Nadzirano učenje (učenje uz učitelja/tutora)

- Znanje – svodi se na primjere ($input_i$,$output_i$) = (x_i,y_i)
- Cilj: minimizirati grešku 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 $f_n(\mathbf{x})$:

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

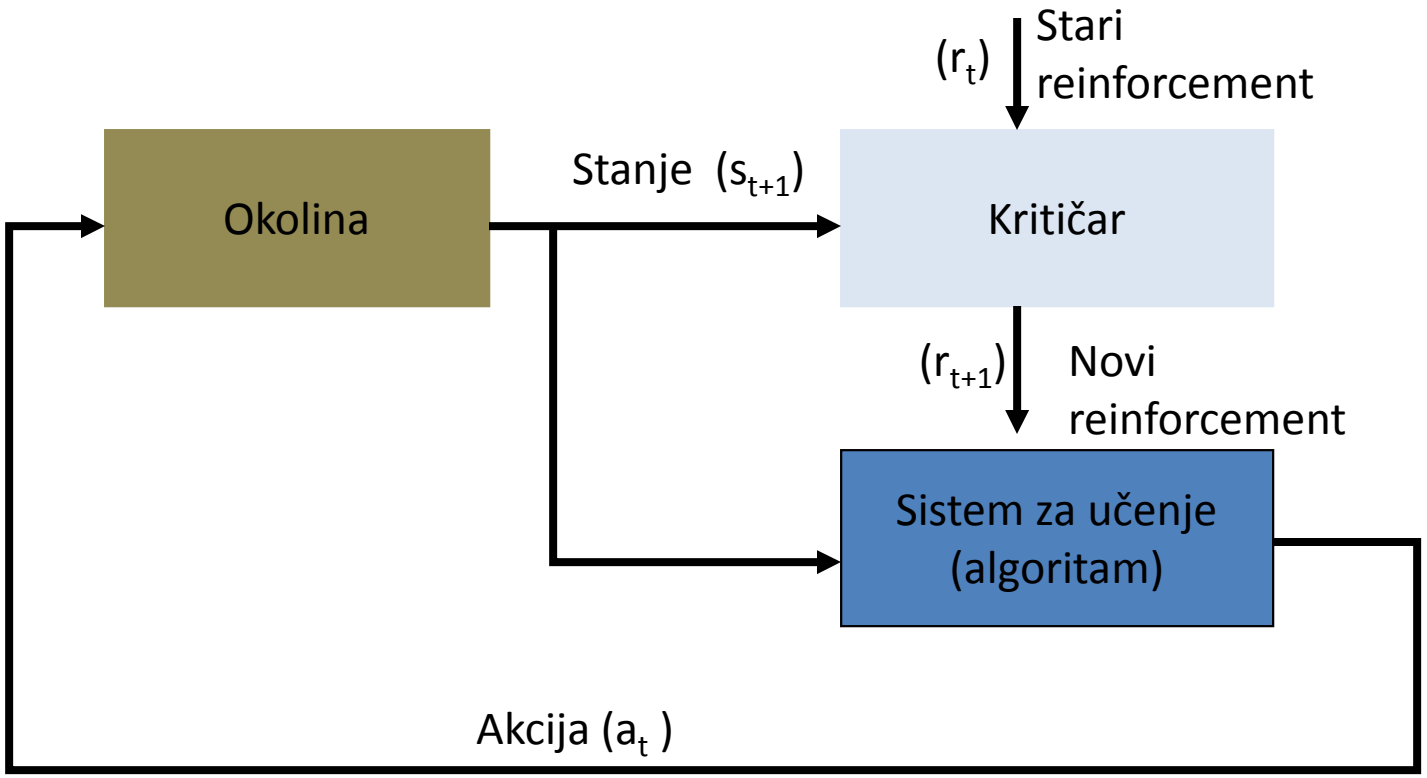
Primjer:

Opis input-output odnosa nekog nekog složenog sustava

- reakcija srca (ritam) – na različite podražaje
- raspodjela snage u jezgri reaktora – nakon spuštanja kontrolnih šipki

Reinforcement learning – učenje niza akcija

- učenje – kroz interakciju s okolinom (ili barem simulacija interakcije)
- istraživanje odnosa stanja i akcija
- povratna veza (feed-back) kroz odgođeni primarni reinforcement
- Cilj: maksimiziranje akumuliranih budućih “reinforcements”



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)
- Učenje poteza u igri (šah, backgammon)

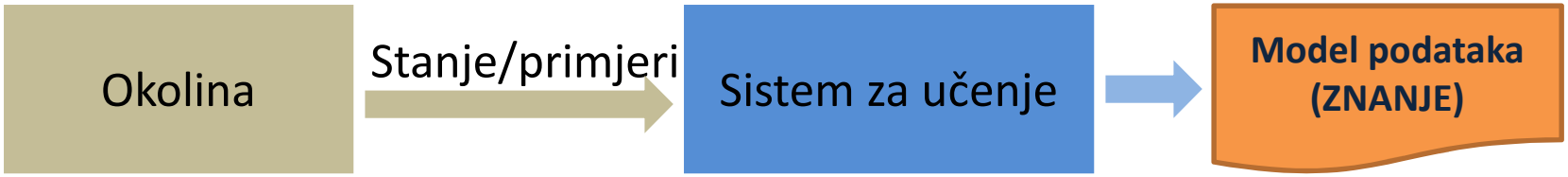
Karakteristike ovih problema:

- Odgođeno nagrađivanje umjesto instantnog za dobre akcije (tzv. temporal credit assignment problem)
- Nema pravog nadzora – supervizije procesa učenja (primjera 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!



Outlier detection: inputs are anything.

Goal is to select **highly unusual cases** from new and given data.

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: igranje šaha (povlačenje pravih poteza)
q: omjer dobivenih i izgubljenih igara
e: igranje igara protiv samog sebe

t: prepoznavanje slova u tekstu
q: postotak točno prepoznatih slova
e: baza slika sa ručno napisanim tekstovima (pixel po pixel)

t: određivanje funkcije gena
q: postotak točno "anotiranih" funkcija za gene s poznatom funkcijom
e: dostupne baze ekspertno/eksperimentalno anotiranih gena (bioinformatika)

t: kome ponuditi novu policu osiguranja
q: postotak točno određenih kandidata iz baze poznatih primjera
e: baza podataka o klijentima osiguravajućeg društva

Uobičajeni postupak – stvaranje modela iz skupa primjera

1. Pretpostavka: postoji distribucija vjerojatnosti p(**x**, y) čiji pravi oblik neznamo

   Obično na raspolaganju imamo tek (konačan) skup primjera (parova **x** i y) (obično uz određeni "šum" ili greške):

   $\{\boldsymbol{x}_1, y_1; \ldots; \boldsymbol{x}_n, y_n; \ldots; \boldsymbol{x}_N, y_N\}$

   Primjeri iz skupa su iid (independently and identically distributed)

2. Koristeći samo $\{\boldsymbol{x}_1, y_1; \ldots; \boldsymbol{x}_n, y_n; \ldots; \boldsymbol{x}_N, y_N\}$ 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(**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 sampliranja i učenja:

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

= overfitting (ili suprotno od generalizacije)

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

Generalizacija ~ sposobnost dobre predikcije na novim primjerima !

Kapacitet = kompleksnost (prostora) hipoteza

Osnova:
Hipoteza 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
> (da - 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 odnosno ako je kompleksnost mogućih hipoteza ograničena !

algoritam/učenik koji nema ugrađen induktivni bias ne može generalizirati !

Što nam treba ?

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

---

Cilj: podesiti vrijednosti parametara ciljne funkcije modela tako minimiziramo vrijednost funkcije cilja

Generalni oblik funkcije cilja

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

Θ – oznaka modela
L (**X**,Y|Θ) – mjera greške modela (loss function)
P (Θ) – funkcija kojom se penalizira kompleksnost modela

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