Sigurnost u računarskim mrežama

IDS sistemi: drugi deo
(IDS sistemi zasnovani na mašinskom učenju)

Nemanja Maček

Sadržaj

- Veštačka inteligencija i mašinsko učenje
- Induktivno empirijsko učenje funkcionalnih preslikavanja
- Klasifikacija i problem prenaučenosti
- Anomalije
- Neke značajnije metode klasifikacije
- Analiza obučavajućih skupova: KDD Cup’99
- Praktična demonstracija
  - Sinteza obučavajućih skupova
  - Upotreba alata Weka i Matlab Statistical and Machine Learning Toolbox
  - Upotreba Python biblioteka

Šta je veštačka inteligencija?

- VI trenutno obuhvata mnoštvo podoblasti, od vrlo opštih, kao što su učenje i percepcija, do uskih zadataka kao što je:
  - Igranje šaha
  - Dokazivanje matematičkih teorema
  - Medicinska dijagnostika
  - Automatsko prevođenje
  - Prepoznavanje govora
  - Robotika
  - ...

Šta je veštačka inteligencija?

- I pored poluvekovne istorije VI je i dalje oblast koju je teško precizno definisati.
- Primeri nekih definicija dobijenih dihotomijom u odnosu na dimenzije rezonovanja:
  - **Sistemi koji razmišljaju kao ljudi** (ljudsko rezonovanje).
    - „[Automatizacija] aktivnosti koje asociramo sa ljudskim razmišljanjem, aktivnosti kao što su donošenje odluka, rešavanje problema, učenje...“
  - **Sistemi koji razmišljaju racionalno** (racionalno rezonovanje).
    - „Izučavanje računanja koje omogućava opažanje, rasuđivanje i delovanje.“
  - **Sistemi koji se ponašaju kao ljudi** (ljudsko ponašanje).
    - „Izučavanje kako naterati računare da rade stvari u kojima su, trenutno, ljudi bolji.“
  - **Sistemi koji se ponašaju racionalno** (racionalno ponašanje).
    - „Računarska inteligencija je nauka o dizajniranju inteligentnih agenata.“
    - „VI ... se bavi inteligentnim ponašanjem veštackih naprava.“
- Ljudski-centrirani pristupi dominantno moraju biti **empirijska nauka**.
  - Uključuje hipotezu i eksperimentalnu potvrdu o ljudskom ponašanju.
- Racionalistički pristup koristi kombinaciju matematike i inžinjerstva.

Pristup zasnovan na Tjuringovom testu

- Alan Tjuring (1950): „*Computing Machinery and Intelligence*“
  - Umesto pitanja da li mašine mogu da misle, postavlja se pitanje da li mašina može da prodje test inteligentnog ponašanja.
  - Test je nakon toga nazvan Tjuringov test.
- Mašina prolazi Tjuringov test ako osoba ispitivač, posle postavljanja nekoliko pitanja u pisanoj formi, ne može da odredi da li je pisani odgovor dao čovek ili mašina.
- Mašina koja bi eventualno prošla Tjuringov test, morala bi da ima sledece sposobnosti:
  - **Obrada prirodnih jezika** (da bi mogla uspešno da komunicira u prirodnom jeziku)
  - **Reprezentacija znanja** (da bi memorisala ono što zna i prima na osnovu senzora)
  - **Automatsko rezonovanje** (da bi koristila memorisane informacije za odgovaranje na pitanja i za donošenje novih zaključaka)
  - **Mašinsko učenje** (da bi se adaptirala novim okolnostima).

Pristup zasnovan na Tjuringovom testu

- Tjuringov test je namerno izbegavao direktnu fizičku interakciju između ispitivača i računara.
- Razlog: fizička simulacija čoveka nepotrebna za inteligenciju.
- Takozvani potpuni Tjuringov test uključuje video signal.
  - Ispitivač može da testira percepcijske i motorne sposobnosti ispitanika.
- Da bi prošao potpuni Turingov test racunar mora da ovlada:
  - **Računarskom vizijom** (u cilju vizuelene percepcije okoline i objekata u njoj)
  - **Robotikom** (radi kretanja u prostorui manipulacijom objekata u njemu).
- Ovih 6 disciplina cine dominantan deo VI!

Fenomen podataka

- „**U Boga verujemo, svi ostali neka donesu podatke**.“
  - Na Webu se ova izjava pripisuje podjednako i Vilijemu Demingu i Robertu Hajdenu.
  - Profesor Hajden tvrdi da ovo nije njegova izjava.
  - Ironija je u tome da se ne mogu naći „podaci“ koji bi potvrdili da je Deming zaista njen autor. [3]

Mašinsko učenje

- Učenje je proces koji se kod ljudi odvija gotovo neprekidno.
- Postoje različite vrste učenja:
  - Čisto memorisanje podataka
  - Učenje motornih veština
  - Sticanja sposobnosti analitičkog i kreativnog mišljenja
  - ...
- Sposobnost učenja se smatra osnovnom odlikom inteligentnih bića!
  - Nije iznenađujuće što je mašinsko učenje jedna od centralnih oblasti istraživanja u VI.
- **Mašinsko učenje** je oblast koja proučava procese na kojima se zasniva učenje kod ljudi i kod veštačkih sistema.
  - **Veštalki sistemi** koji su sposobni da uče evidentno poboljšavaju svoje performanse.
  - **Biološki sistemi** povećavaju verovatnoću svog opstanka i produžetka vrste.

Mašinsko učenje

- **Formalna definicija mašinskog učenja** (Tom Mičel):
  - Za jedan sistem VI kažemo da uči zadatu klasu zadataka T na osnovu iskustva E i zadate mere performansi P, ako se njegove performanse za rešavanje zadataka iz klase T, poboljšavaju sa iskustvom E.
- Da bi jedan problem učenja bio dobro definisan neophodno je identifikovati tri komponente:
  - **Klasu zadataka** (T)
  - **Meru performanse koju treba poboljšati** (P)
  - **Izvor iskustva** (E).
- Primer: problem učenja prepoznavanja rukom pisanih znakova.
  - Zadatak T: prepoznavanje i klasifikacija rukom pisanih reči u zadatim slikama
  - Mera performansi P: procenat tačno klasifkovanih reči.
  - Izvor iskustva E: baza rukom pisanih reči sa tačnom klasifikacijom (obučavajuci skup).

Mašinsko učenje

- U savremenoj oblasti mašinskog ucenja dominiraju sledeci pristupi:
  - **Induktivno učenje**
  - **Analitičko učenje** (analogija sa logikom)
  - **Učenje na slučajevima** (engl. *case-based learning* – analogija sa ljudskim pamćenjem)
  - **Neuralne mreže** (analogija sa neurobiologijom)
  - **Genetski algoritmi** (analogija sa evolucijom)
  - **Hibridni modeli** (kombinacija više pristupa).
- Od najvećeg značaja za tekuću praksu u domenu računarstva i VI je **induktivno učenje**.
- Suština ovog tipa učenja je **učenje na osnovu raspoloživih primera**.
  - U svakodnevnom jeziku to možemo da nazovemo: **učenje iz iskustva** (drugih).
- S obzirom na objekat učenja najopštije je učenje funkcionalnih, tj. **ulazno-izlaznih preslikavanja**.

Induktivno empirijsko učenje funkcionalnih preslikavanja

- Ključni elementi u induktivnom učenju funkcionalnih preslikavanja su:
  - Nepoznato preslikavanje
  - Obučavajuci skup ulazno – izlaznih parova
  - Skup hipoteza unutar koga biramo finalnu hipotezu putem algoritma obučavanja.
- Ulazi su po pravilu *n* dimenzioni vektori.
  - U literaturi se pominju pod nazivima: **vektori obeležja**, uzorci, primeri i instance.
- Komponente ulaznih vektora se nazivaju **obeležja** ili atributi i mogu biti:
  - **Kontinualni** (beskonačan broj vrednosti)
  - **Diskretni** (konačan broj vrednosti).
- Ove komponente mogu biti tri različite prirode:
  - Realni brojevi, npr. $x_i = 0.34$
  - Diskretni brojevi, npr. $x_i \in \{0, 1, 2, 3\}$
  - Kategorijalne varijable, npr. komponenta x_i označava boju koja može uzeti vrednosti {crven, plav, zelen}.

Induktivno empirijsko učenje funkcionalnih preslikavanja

- Izlazni prostor može biti:
  - Kategorijalan sa K distikntnih vrednosti, kada obučeni sistem obavlja **klasifikaciju**, **prepoznavanje** ili **kategorizaciju**.
    - Izlaz se naziva oznaka, klasa, kategorija ili odluka.
  - Realan, kada obućeni sistem realizuje **regresiju** (funkcionalni estimator).
  - Vektorski, sa komponentama koje mogu biti realne ili kategorijalne varijable.
- Obučavanje se vrši na osnovu **obučavajuceg skupa** (N ulazno-izlaznih parova).
- Obučavanje možemo posmatrati kao pretragu u prostoru hipoteza u cilju **izbora najpogodnije hipoteze** koja je u saglasnosti sa obučavajucim skupom.
- Postoji više načina na koji se obučavajuci skup može koristiti:
  - **Batch metod**: celokupan obučavajući skup se koristi u izračunavanju hipoteza.
  - **Inkrementalni metod**: koristi se sekvencijalno, po jedna instanca obučavajuceg skupa, koja modifikuje tekuću hipotezu.
  - **On-line metod**: korišćenje jedne instance obučavajućeg skupa u trenutku kada ona postane dostupna.

Obučavajući i test skup, generalizacija

- Pitanje **merenja kvaliteta obučenog sistema** je jedno od ključnih pitanja mašinskog učenja.
- Izabrana hipoteza koja je najbolja moguća u odnosu na dati obučavajuci skup:
  - Može se pokazati kao dobra na novim primerima koji nisu učestvovali u obučavanju.
  - Može se međutim pokazati i kao loša.
- Opšteprihvacena mera kvaliteta jednog sistema mašinskog učenja je njegova sposobnost **generalizacije**.
  - Generaliacija: ponašanje sistema na instancama koje nisu vidjene u fazi obučavanja.
- Stoga tipično sintezu prate dva skupa: **obučavajući skup** (trening skup) i **test skup**.
  - Test skup se koristi samo za procenu generalizacionih sposobnosti sintetisanog sistema.
  - Test skup se NE SME koristiti u fazi obučavanja.
- Uobičajene numeričke mere performansi su ili srednjekvadratna greška između izlaza sistema i ciljnih vrednosti ili ukupan broj grešaka.

Vrste obučavanja

- **Obučavanje sa učiteljem**, nadgledano obučavanje (engl. *supervised learning*).
  - Obučavajući skup je oblika $\{x_i, f(x_i)\}$, gde je $f(x_i)$ ciljna vrednost vektora obeležja x_i.
  - Zadatak obučavanja: nalaženje aproksimacije za f.
  - Mera performansi: kvalitet aproksimaciije u tačkama koje ne pripadaju obuč. skupu.
- Obučavanje sa učiteljem:
  - **Transduktivno**. Test primeri bez ciljne klasifikacije se znaju već u trenutku obučavanja.
    - Cilj je da se sistem dobro ponaša samo na tim test primercima.
  - **Induktivno**. Cilj je da se sistem dobro ponaša na svim primerima dobijenim iz iste raspodele kao i obučavajući skup.
- **Obučavanje bez učitelja**, nenadgledano obučavanje (engl. *unsupervised learning*).
  - U obućavajućem skupu su prisutne samo ulazne instance x_i.
  - Tipičan pristup je **klasterovanje**, odnosno grupisanje raspoloživih podataka u manji broj grupa unutar kojih su podaci sličniji u poređenju sa podacima iz ostalih grupa.
  - Imenovanjem klastera se dolazi posrednim putem do označenih uzoraka.

Vrste obučavanja

- **Obučavanje sa podsticanjem** (engl. *reinforcement learning*).
  - Poreklo vodi iz teorije upravljanja u kome je dinamičko okruženje opisano trojkom **(stanje, akcija, nagrada)**.
  - U ovom tipu učenja potrebno je naučiti kako vršiti preslikavanja situacija u akcije a da se pri tome maksimizira nagrada.
  - Za razliku od obučavanja sa učiteljem algoritmu obučavanja nije rečeno koje akcije da preduzima u datoj situaciji.
  - Dobar primer da se shvati ovaj scenario je učenje igranja šaha.
    - Stanja: pozicije na table.
    - Akcije: mogući potezi za datu poziciju.
    - Nagrada za izabrani potez: pobeda.
    - Kazna: gubitak igre.
  - Nagrada i kazna **kasne u odnosu na trenutak izbora akcije**, što je tipično za ovu vrstu obučavanja.

Klasifikacija

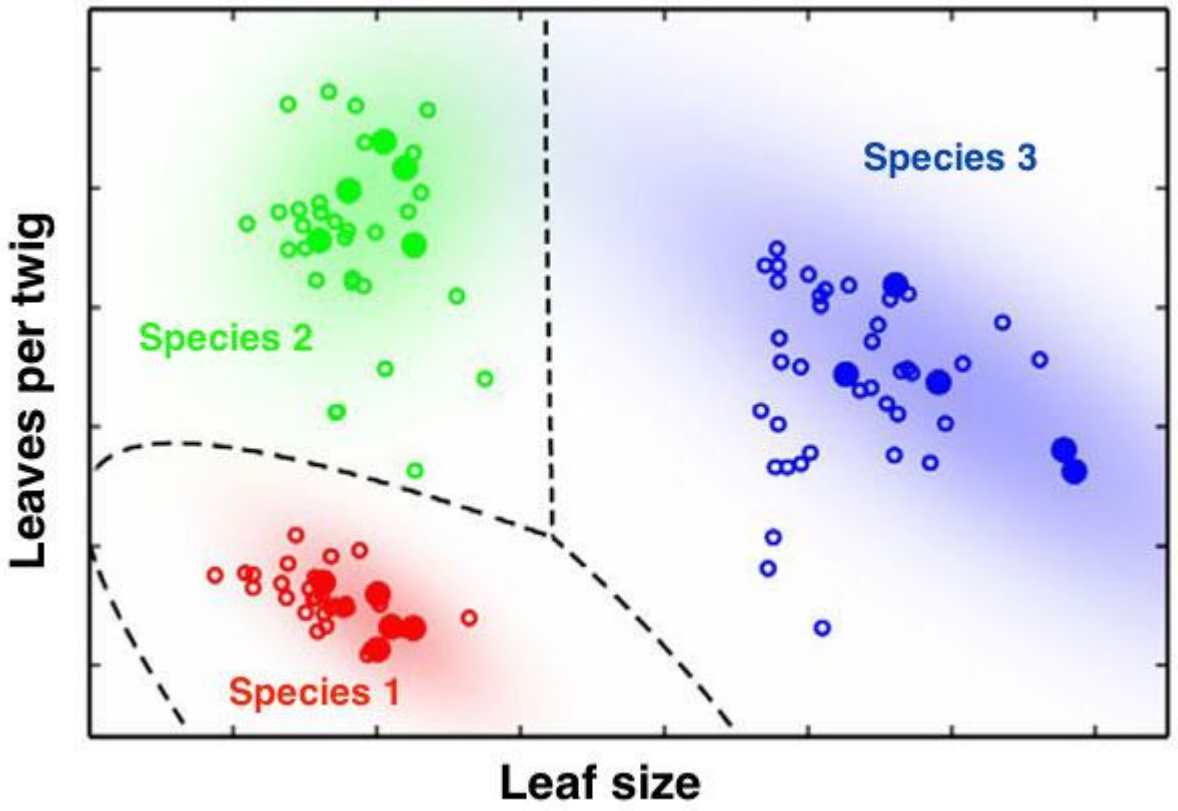
- **Klasifikacija** je razvrstavanje nepoznate instance u jednu od unapred ponuđenih klasa.
  - Svaka instanca može se predstaviti **skupom njenih obeležja**.
  - Takođe, svakoj instanci se može dodati kao obeležje i **oznaka klase** kojoj instanca pripada.

sepallength	sepalwidth	petallength	petalwidth	class
5.3	3.7	1.5	0.2	Iris-setosa
5	3.3	1.4	0.2	Iris-setosa
7	3.2	4.7	1.4	Iris-versicolor
6.2	2.8	4.8	1.8	Iris-virginica

- Klasifikacija je odredivanje vrednosti obeležja klase na osnovu preostalih obeležja instance.
- Postoji veliki broj metoda kojima se ovaj problem rešava: VNM, SVM, stabla odlučivanja, …

Klasifikacija

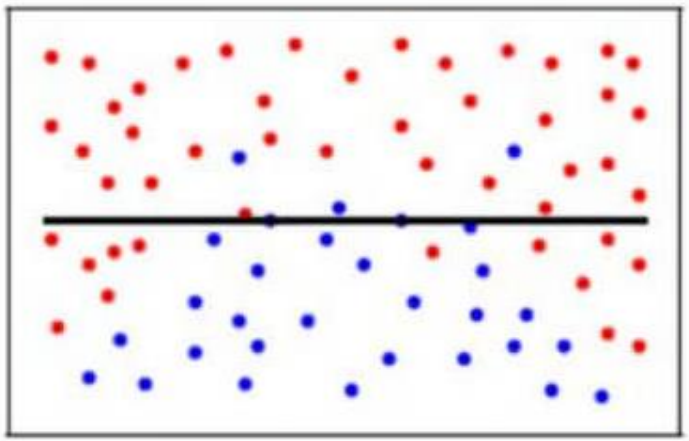
- Primer:

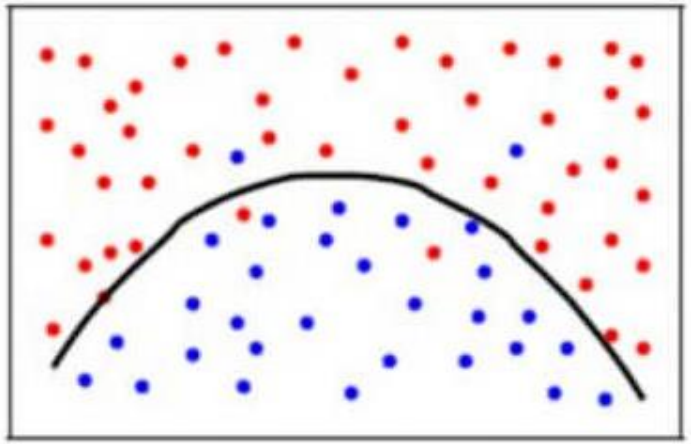


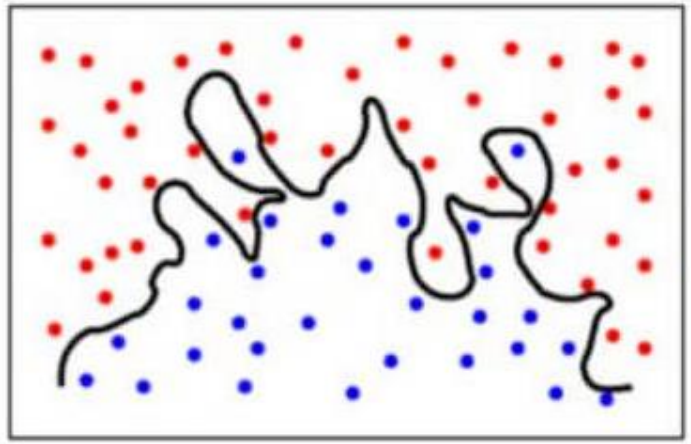
* Slika preuzeta sa Web lokacije: https://astrobites.org/2015/04/15/c-3po-phd-machine-learning-in-astronomy/

Prenaučenost

- Prilikom obuke je moguće da se model **previše prilagodi specifičnostima obučavajućeg skupa**.
  - U tom slučaju model daje loše rezultate kada se primeni na test skupu ili drugim podacima.
  - Iako je potrebno da se prilikom obuke postigne visok nivo tačnosti neophodno je da se spreči **prenaučenost**.
- Zbog čega dolazi do prenaučenosti?
  - Osnovni razlog je bogatstvo prostora hipoteza, odnosno skupa dopustivih modela.
  - U bogatijem skupu je lakše naći model koji dobro odgovara podacima.





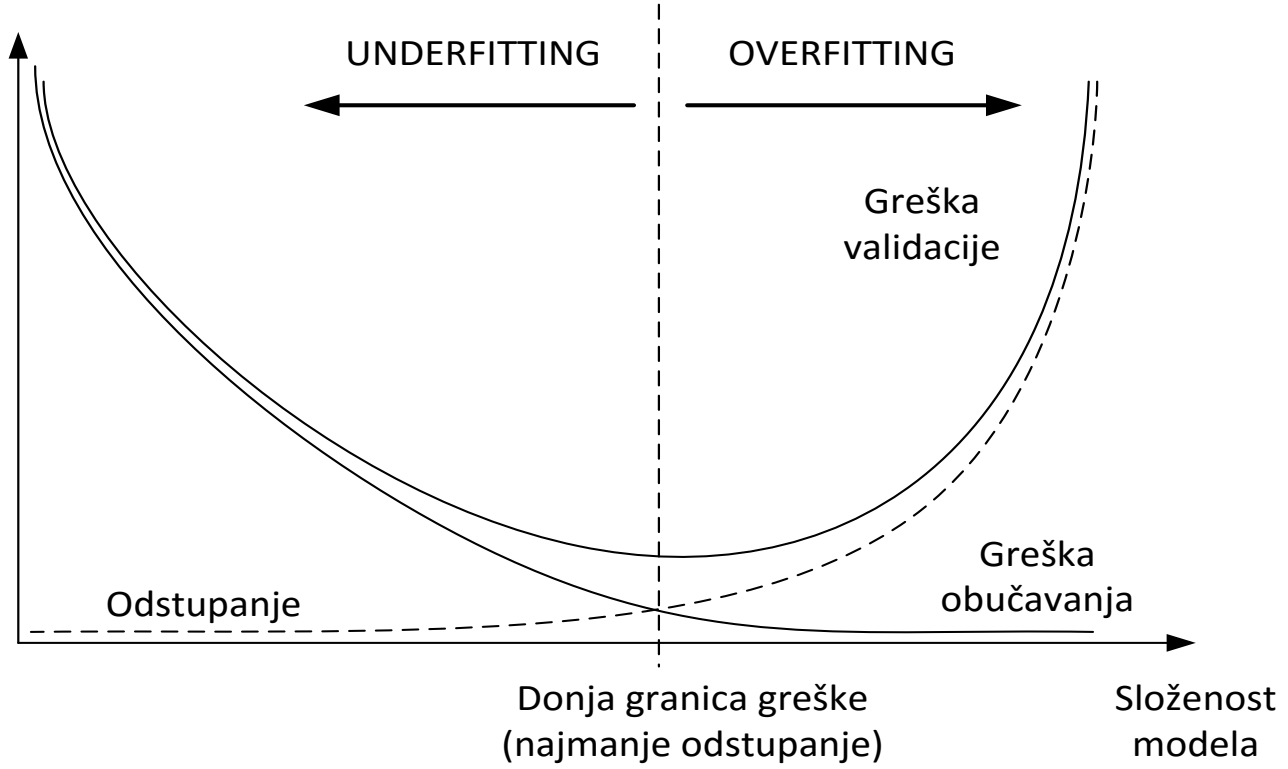


Prenaučenost

- Primer *underfitting*-a.
  - Prilikom učenja dopuštaju se samo **stabla odlučivanja dubine 1** koja testiraju samo jedan atribut svake instance.
    - Takva stabla ne mogu lako postići visoku preciznost klasifikacije.
- Primer *overfitting*-a.
  - Prilikom učenja dozvoljava se upotreba **stabala odlučivanja proizvoljne dubine**.
    - Moguće je naći stablo koje **precizno opisuju svaku (čak i najnebitniju) specifičnost obučavajućih podataka**.
    - Takvo stablo je u stanju da razlikuje bilo koje dve instance u obučavajućem skupu.
    - Takvo stablo postiže savršenu preciznost na obučavajućem skupu.
  - Ovakva stabla se u praksi pokazuju nepouzdanim!
  - Razlog: širi skupovi podataka **ne moraju uvek imati sve specifičnosti obučavajućeg skupa**!

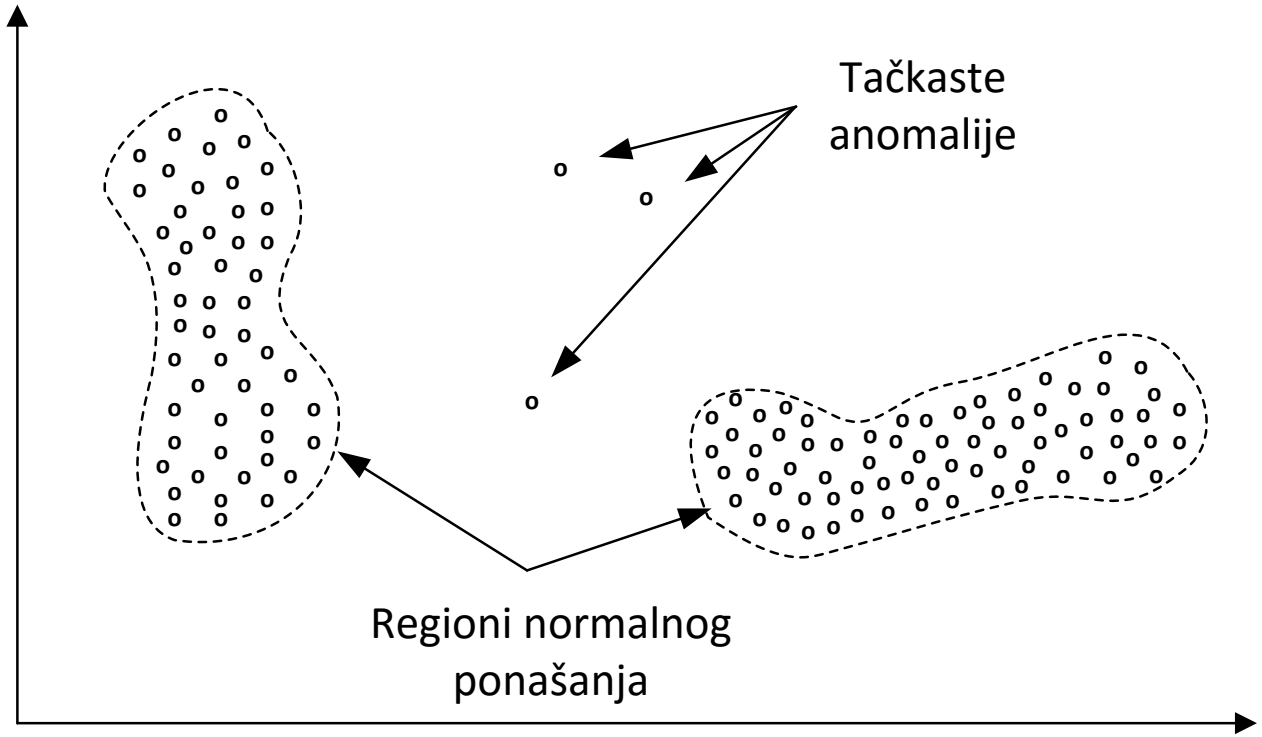
Prenaučenost

- Ponašanje odstupanja greške na širem skupu podataka od greške na obučavajućim podacima (**greška validacije**). Validacija nije isto što i test!



Anomalije

- Anomalije su oblici u podacima koji ne pripadaju očekivanom ponašanju
- Nazivaju se i autlajeri, izuzeci, osobenosti, iznenađenja i sl.



Detekcija anomalije

- Detekcija anomalija obavlja se definisanjem regiona koji predstavlja normalno ponašanje, i pronalaženjem podataka koji ne pripadaju tim regionima.
- Nekoliko faktora čine ovaj, na prvi pogled jednostavan zadatak, znatno složenijim:
  - **Definisanje reprezentativnog normalnog ponašanja**, tj. regiona koji obuhvataju sve moguće varijacije normalnog ponašanja je komplikovan i težak zadatak.
  - **Određivanje granica između normalnog ponašanja i anomalija**.
    - Granice se često se ne mogu precizno odrediti, što dovodi do greške u detekciji.
  - **Evolviranje normalnog ponašanja sa vremenom**.
    - Normalno ponašanje se u realnim sistemima vremenom menja.
    - Skup regiona normalnog ponašanja može postati neadekvatan nakon nekog vremena.
  - **Broj obeleženih zapisa** u obučavajućem skupu i skupu za validaciju algoritama za detekciju može biti nedovoljan, jer su obeleženi podaci za obuku jednostavno nedostupni.

Detekcija anomalije

- **Egzaktna definicija autlajera**. Ne postoji jedinstven način definisanja granice odstupanja od normalnog ponašanja za različite problemske domene.
  - Primer: malo odstupanje telesne temperature u medicine ukazuje na anomaliju, dok u bankarstvu malo odstupanje valute ne mora ukazati na anomaliju.
  - Tehnike definisanja granice, odstupanja i detekcije anomalije razvijene za primene u jednom domenu ne moraju automatski biti primenljive i u drugom.
- **Postojanje šuma**.
  - Podaci koji ukazuju na normalno ponašanje mogu sadržati šum koji je sličan oblicima prisutnim u anomalijama, a koji se ne može se ukloniti.
- Specifičnost problema određuju:
  - **Priroda ulaznih podataka**.
  - **Dostupnost obeleženih podataka** u skupovima za obučavanje i validaciju
  - **Vrsta anomalije** koju treba detektovati.

Priroda ulaznih podataka

- Jedan od najbitnijih aspekata detekcije anomalije je **priroda ulaznih podataka**.
- Svaka instanca podataka opisana je **vektorom obeležja**.
- Obeležja mogu biti:
  - Istog tipa (na primer, sva obeležja su realni brojevi)
  - Različitog tipa (na primer, kombinacija binarnih i kontinualnih obeležja).
- Priroda ulaznih podataka odnosi se na prirodu obeležja kojima su opisane instance.
- Time se određuje koja je tehnika pogodna za detekciju anomalije.
  - Primer: Support Vector Machines klasifikator radi samo sa numeričkim podacima koji su standardizovani.

Tačkaste anomalije

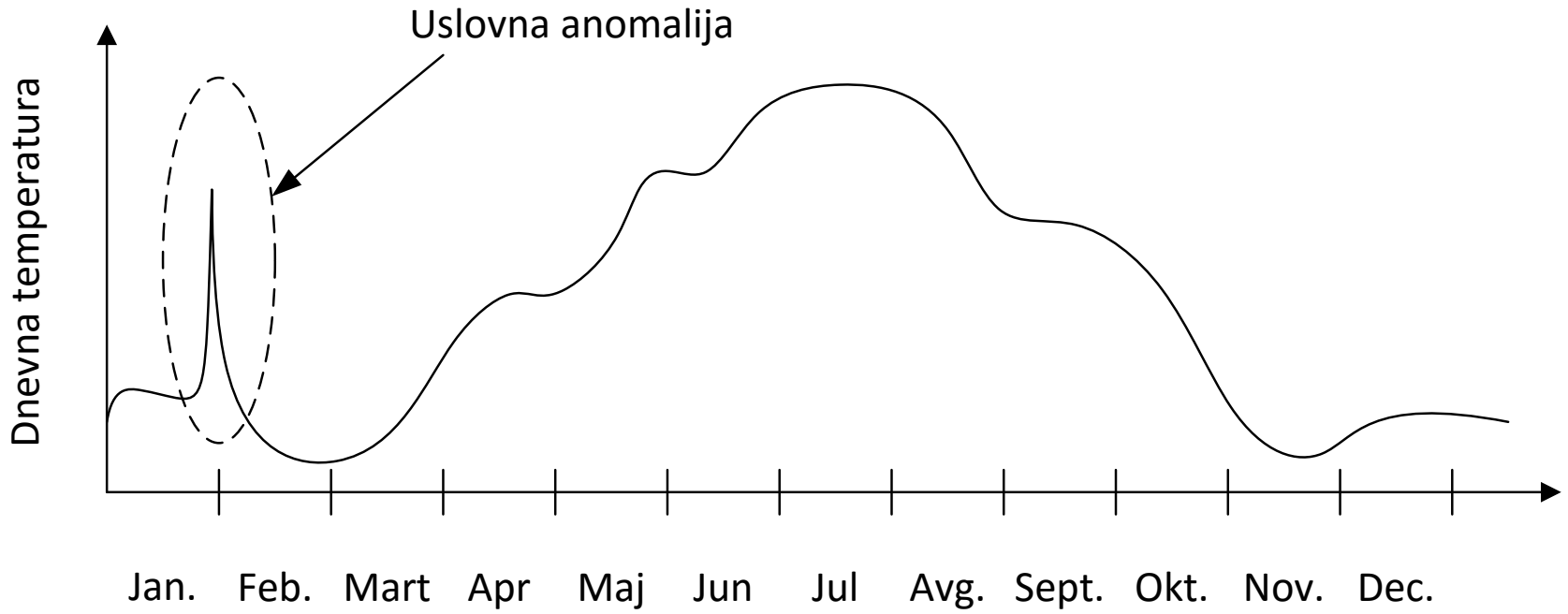
- Ukoliko se jedna instanca podataka može smatrati anomalijom u odnosu na ostatak podataka, onda se takva instanca naziva **tačkasta anomalija**.
- Primer tačkaste anomalije: **detekcija prevare kreditnim karticama**.
  - Neka je skup podataka zadat kartičnim transakcijama određene osobe, i radi jednostavnosti opisan jednim atributom (količina potrošenog novca).
  - Region normalnog ponašanja definisan je donjom i gornjom granicom normalne potrošnje.
  - Ukoliko je nakon obavljene transakcije potrošena suma je znatno veća od opsega definisanog normalnom potrošnjom, transakcija se smatra tačkastom anomalijom.

Uslovne anomalije

- Ukoliko instanca podataka predstavlja anomaliju **u specifičnom kontekstu**, ali ne i na drugi način, onda se ta instanca naziva kontekstualnom ili **uslovnom** anomalijom.
- Primer uslovne anomalije:
  - Učestalost korišćenja sistemskih poziva ili slanja zahteva za otvaranjem TCP veza serveru u određenim vremenskim intervalima.
- Pojam konteksta je uslovljen strukturom skupa podataka i mora biti specificiran kao deo formulacije problema.
- Svaka instanca podataka je definisana pomoću dva skupa atributa:
  - **Kontekstualnim atributima** (određuju kontekst, tj. uslov).
  - **Atributima ponašanja** (definišu karakteristike instance ne uzimajući kontekst u obzir).
- Instanca podataka sa identičnim atributima ponašanja koja predstavlja anomaliju u datom kontekstu može u drugom kontekstu predstavljati normalno ponašanje.
- Istraživanja u oblasti detekcije kontekstualnih anomalija najčešće su vezana za podatke promenljive u vremenu i prostorne podatke.

Uslovne anomalije

- Primer uslovne anomalije nevezan za računarski sistem.



Kolektivne anomalije

- Ukoliko **skup instanci** predstavlja anomaliju u odnosu na celokupan skup podataka, onda se takva anomalija naziva **kolektivna**.
  - Individualne instance u ovom slučaju ne predstavljaju anomaliju
  - Njihovo **zajedničko pojavljivanje** jeste anomalija.
- Primer kolektivne anomalije:
  - Sekvenca akcija koje se dešavaju u računarskom sistemu:
    - http-web, buffer-overflow, http-web, http-web, smtp-mail, ftp, http-web, ssh, smtp-mail, http-web, buffer-overflow, ssh, ftp, http-web, ftp, smtp-mail, http-web . . .
  - Sekvenca (buffer-overflow, ssh, ftp) opisuje napad na udaljeni sistem koji se izvršava:
    - Prepunjenjem bafera
    - Otvaranjem ssh veze
    - Kopiranjem podataka na udaljeni sistem koristeći ftp protokol.
  - Pojedine instance te sekvence ne predstavljaju tačkaste anomalije, niti kolektivne anomalije ukoliko se pojavljuju u drugim sekvencama.

Tehnike detekcije anomalija

- Tehnike **nenadgledane** (engl. *unsupervised*) detekcije anomalija ne zahtevaju obučavajći skup.
  - Mogu se primeniti u najvećem broju domena primene.
  - Pretpostavka: instance normalnog ponašanja mnogo se češće pojavljuju od anomalija.
  - Ukoliko pretpostavka nije tačna, algoritam će prijaviti veliki broj pogrešno detektovanih instanci.
- Tehnike **nadgledane** (engl. *supervised*) detekcije anomalija.
  - Instance obučavajućeg skupa su obeležene i pripadaju oblicima normalnog ili anomalnog ponašanja.
  - Problem se svodi na **binarnu klasifikaciju**.
  - Dva osnovna problema karakteristična za nadgledanu detekciju anomalija su:
    - Broj instanci koje opisuju anomalije je znatno manji od broja instanci koje opisuju normalno ponašanje.
    - Prikupljanje dovoljno tačnih i reprezentativnih instanci koje opisuju anomalije je najčešće veoma težak zadatak.

Tehnike detekcije anomalija

- Postoje dva pristupa **polu-nadgledanoj** (engl. *semi-supervised*) detekciji anomalija.
  - **Obeleženo normalno ponašanje**.
    - Obeležene su samo instance obučavajućeg skupa koje pripadaju oblicima normalnog ponašanja.
    - Pošto instance koje opisuju anomalije nisu obeležene, ove tehnike polu-nadgledane detekcija anomalija su **primenljive u većem broju domena primene** od nadgledanih tehnika.
    - Tipičan pristup u ovom slučaju je formiranje modela koji će instance svrstati u klasu normalnog ponašanja. Model se koristi za identifikaciju anomalija u test skupu.
  - **Obeleženo anomalno ponašanje**.
    - Drugi pristup polu-nadgledanoj detekciji je formiranje modela na osnovu obučavajućeg skupa u kome su obeležene samo instance koje opisuju anomalije.
    - Ovaj pristup se ne koristi često jer je jako teško formirati obučavajući skup u kome su opisane sve anomalije koje se mogu pojaviti u podacima.

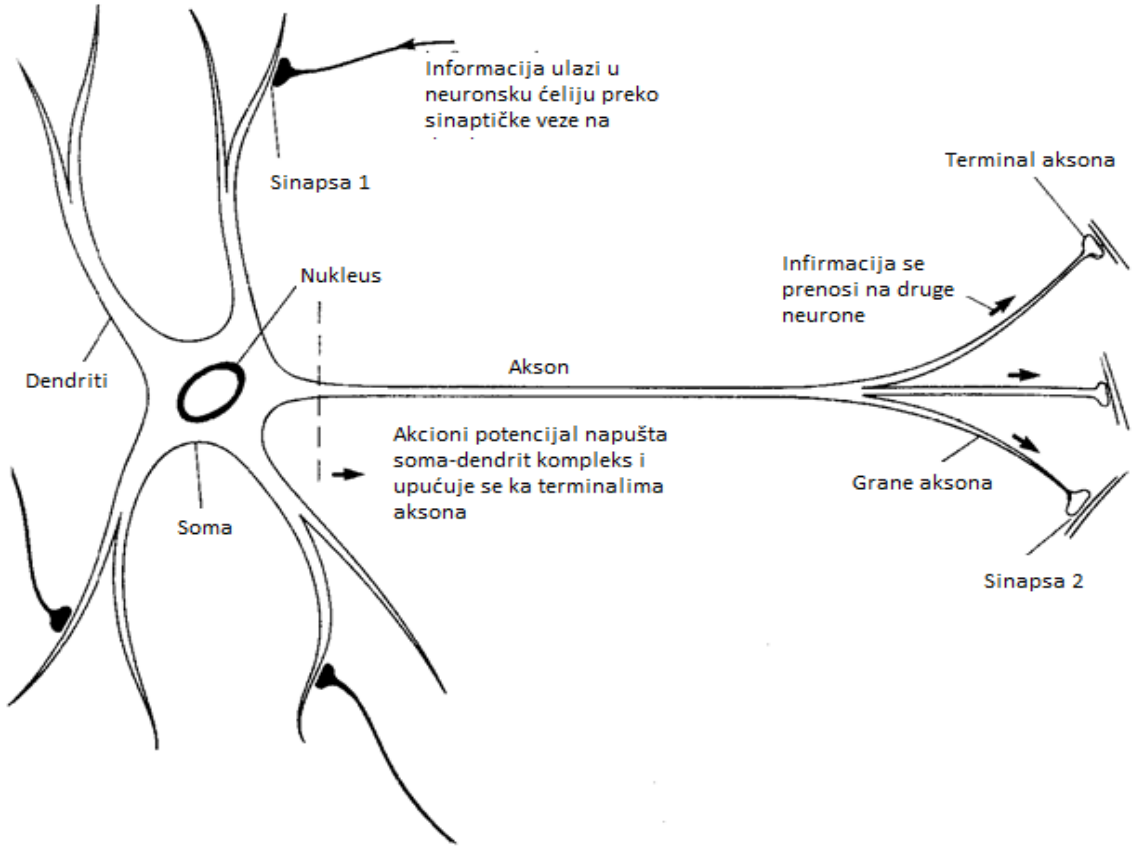
Klasifikacija metoda mašinskog učenja

- Metode mašinskog učenja koje se koriste za klasifikaciju mogu se podeliti na:
  - **Osnovne** (veštačke neuronske mreže, Support Vector Machines, stabla odlučivanja, Naive Bayes)
  - **Hibridne** (na primer, stablo odlučivanja na čijim se listovima nalazi Naive Bayes klasifikator obučen instancama koje pripadaju tom listu)
  - **Inkrementalne** (Naive Bayes Updatable)
  - **Hibridne inkrementalne** (Hoeffding Tree)
  - **Osnovne ensemble** (Random Forest)
  - **Hibridne ensemble** (slaganje)
  - **Hibridne inkrementalne ensemble** (Ada Hoeffding Option Tree).

Veštačke neuronske mreže

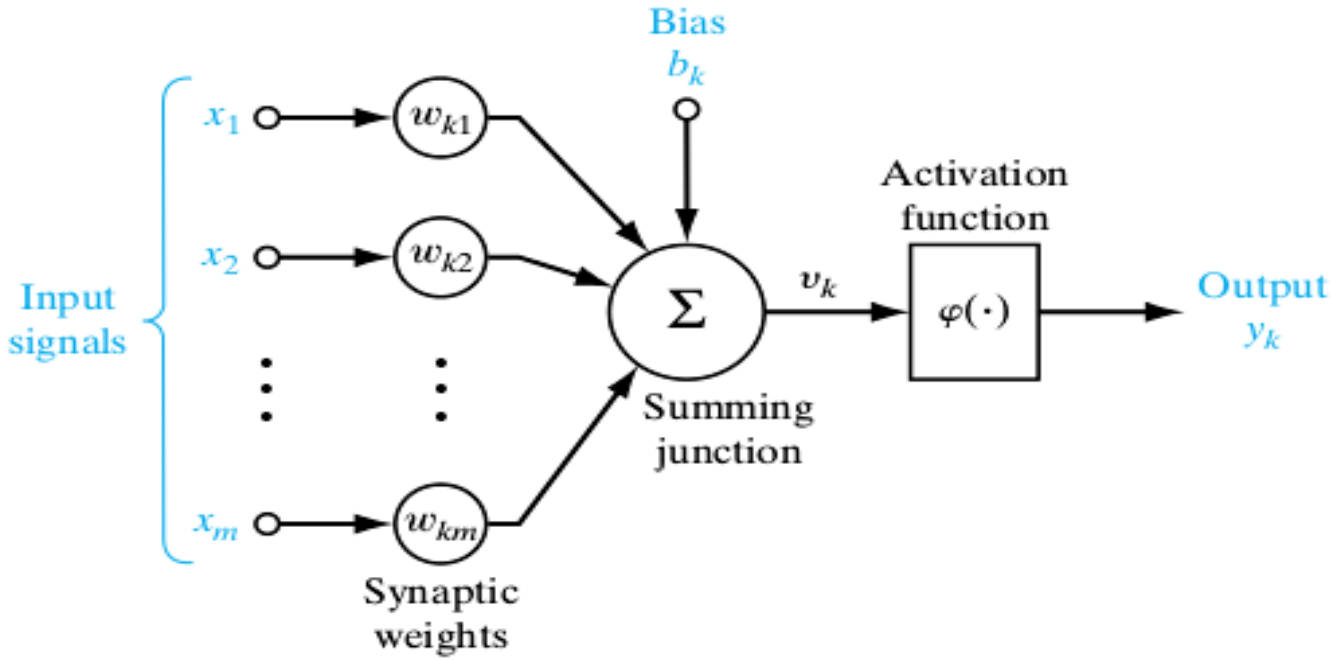
- Neuronska mreža je **masovno paralelizovan distribuirani procesor** sa prirodnom sposobnošću **memorisanja iskustvenog znanja** i obezbedivanja njegovog korišćenja.
- VNM simuliraju način rada ljudskog mozga pri obavljanju datog zadatka ili neke funkcije.
- Veštačke neuronske mreže podsećaju na ljudski mozak u dva pogleda:
  - Neuronska mreža zahvata znanje kroz **proces obučavanja**.
  - Težine medjuneuronskih veza (**jačina sinaptičkih veza**) služe za memorisanje znanja.
- Obučavanjem se na algoritamski (sistematičan) način menjaju sinaptičke težine u cilju dostizanja željenih performansi mreže.
- Osnovnu računarsku snagu neuronskih mreža čini masivni paralelizam, sposobnost obučavanja i generalizacija.
- Neuronske mreže su u okviru mašinskog učenja već stekle status dobro teorijski obradjene oblasti, koja je stavlja u istu ravan sa ostalim top tehnikama, kao što su SVM.

- Funkcionisanje jednog biološkog neurona [3]:

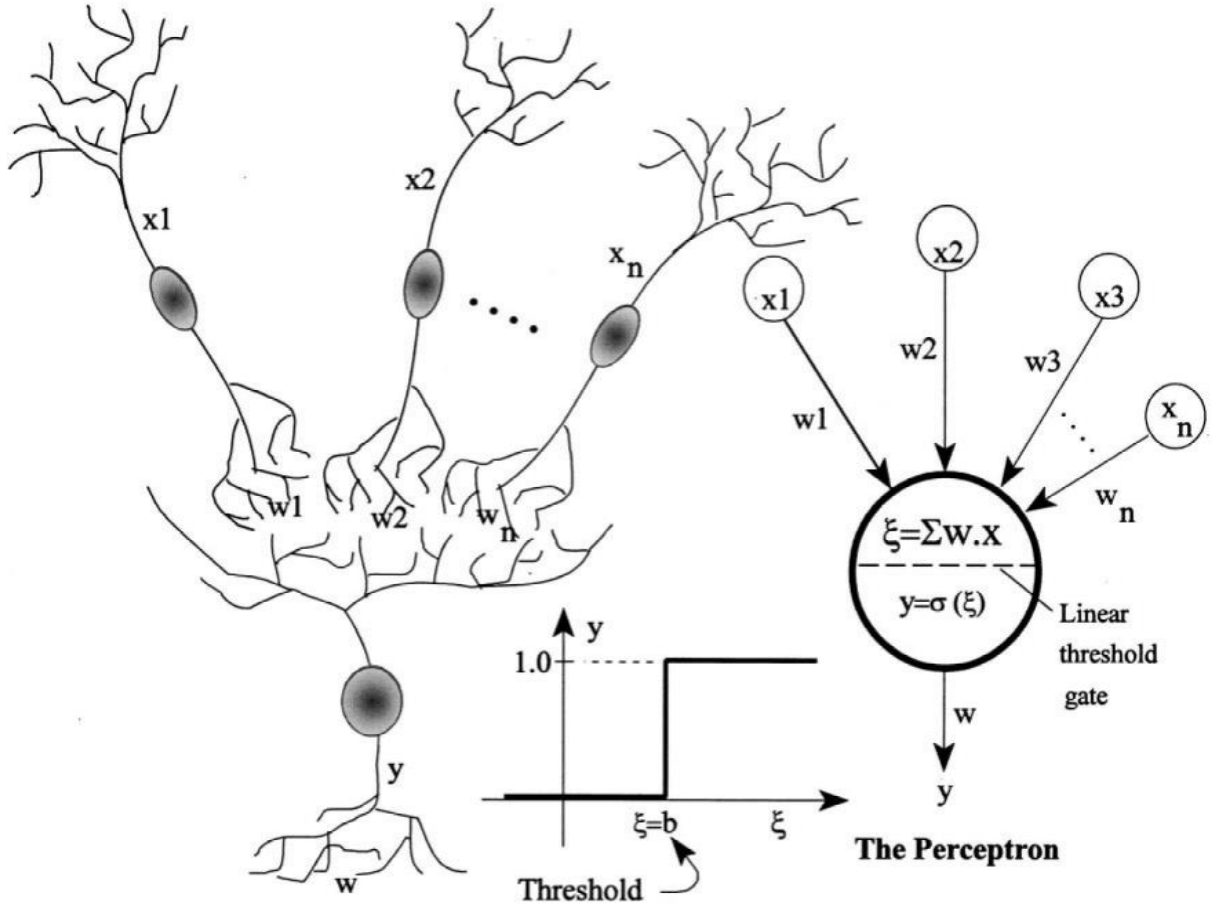


- Model neurona:
  - **Skup sinaptičkih težina**. Pozitivne težine odgovaraju ekscitirajućim sinaptičkim vezama, a negativne inhibitornim.
  - **Sumator** (linearni kombajner). Formira težinsku sumu ulaza.
  - **Aktivaciona funkcija**. Limitira amplitudu izlaznog signala neurona. Tipično se uzima normalizacija izlaza na interval [0,1] ili [-1,1].

- Model neurona.

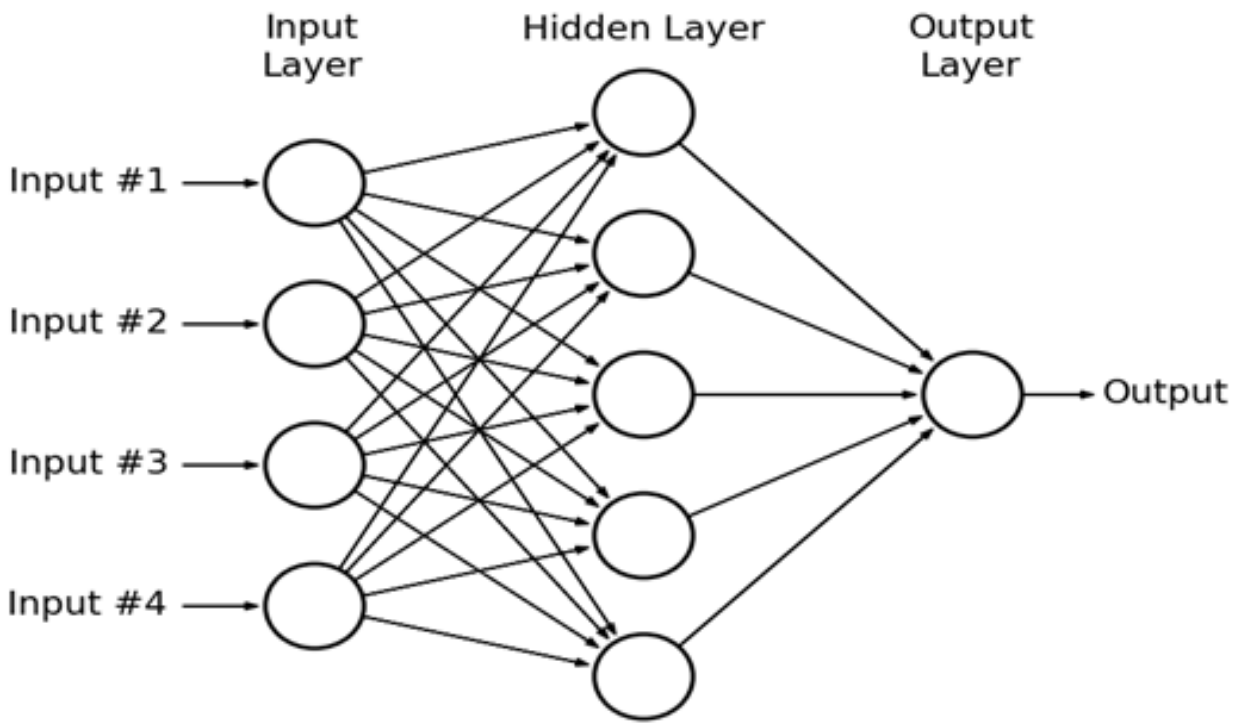


- Analogija.


x1
x2
x n
w1
w2
w n
y
w
x1
x2
x3
x n
w1
w2
w3
w n
ξ=Σw.x
y=σ (ξ)
Linear
threshold
gate
w
y
The Perceptron
y
1.0
ξ=b
ξ
Threshold

Višeslojne neuronske mreže sa prostiranjem signala unapred

- *Feed forward multilayer neural network, FFANN*.



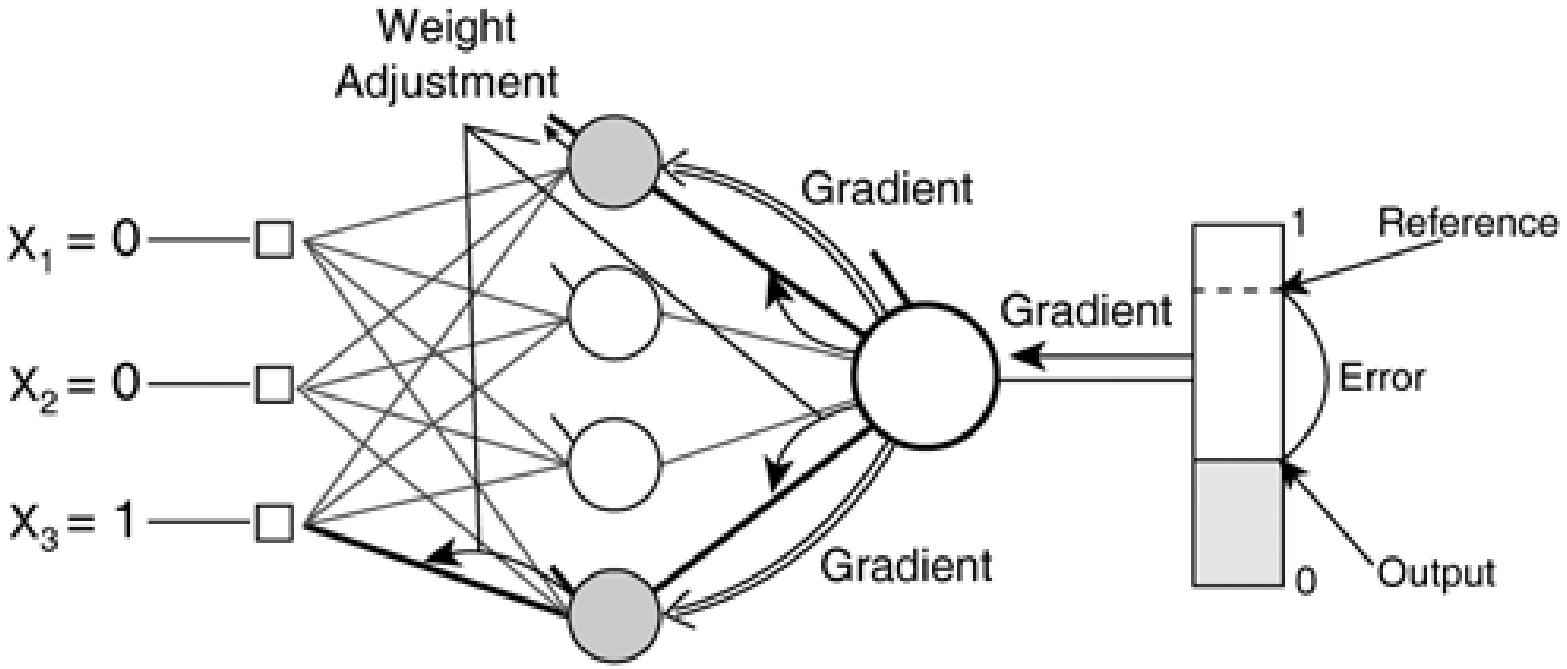
Hornik Stinchombe White-ova teorema

- Definicija Borel merljive funkcije. **Borel merljive funkcije** na kompaktnim skupovima obuhvataju sve neprekidne i u delovima neprekidne funkcije (sa konačno ili prebrojivo mnogo diskontinuiteta na skupovima mere nula).
- **HSW teorema**: višeslojna neuronska mreža sa najmanje jednim skrivenim slojem i aktivacionom funkcijom koja je neopadajuća, a čije vrednosti pripadaju intervalu [0,1] ili [-1,1] aproksimira bilo koju Borel merljivu funkciju na kompaktnim skupovima, sa proizvoljnom tačnošću, pod uslovom da je na raspolaganju dovoljan broj neurona u skrivenom sloju.
  - Odavde sledi da je FFANN **univerzalni aproksimator**.
  - Neuspeh FFANN da u nekom konkretnom slučaju restauriše preslikavanje implicitno zadato obučavajućim skupom ne potiče od osnovnog restauratorskog principa FFANN, već od:
    - Neadekvatnog izbora arhitekture
    - Parametara obučavanja
    - Obučavajućih skupova
    - ...

Obučavanje mreža algoritmom propagacije greške unazad

- **Algoritam propagacije greške unazad** (engl. *backpropagation*) traži minimum funkcije greške u polju težinskih koeficijenata korišćenjem **metode stepenastog opadanja** (engl. *gradient descent*).
  - Kombinacija tešinskih koeficijenata koja minimizuje funkciju greške se smatra rešenjem problema učenja.
- Obzirom da ovaj metod zahteva izračunavanje stepena funkcije greške prilikom svakog iterativnog koraka, mora se garantovati kontinuitet i diferencijabilnost funkcije greške.
- Ovaj algoritam obuhvata dve faze:
  - Ulazni vektor se propagira od ulaznog ka izlaznom sloju, dajući izlaz.
  - Signal greške se od izlaznog ka ulaznom sloju u cilju **korigovanja sinaptičkih težina**.
- Algoritam propagiranja greške unazad ima **dobra svojstva generalizacije** (neuronska mreža dobro generalizuje ukoliko daje dobre interpolacije za nove ulaze, koji nisu bili prisutni u postupku obučavanja).

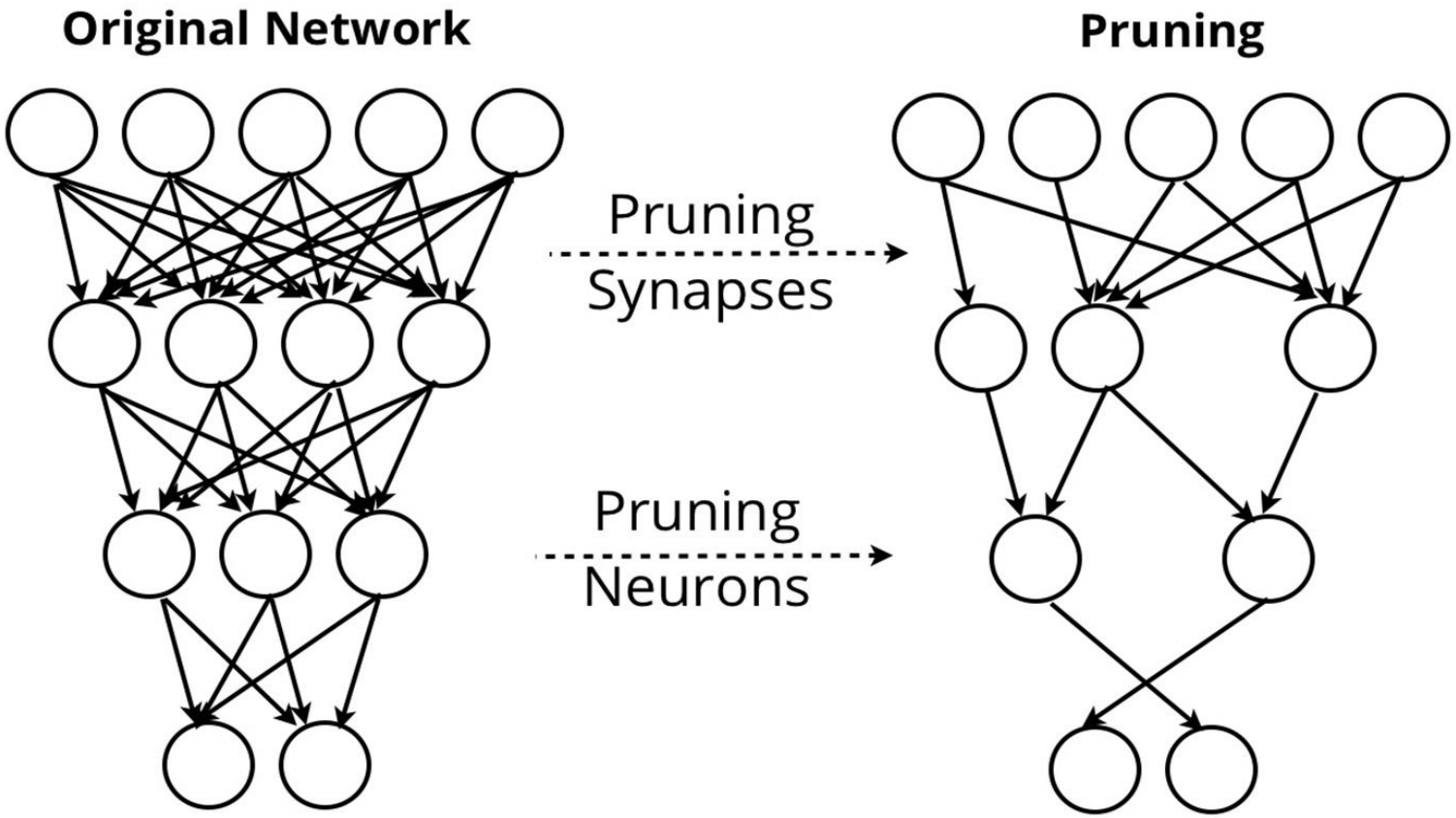
Obučavanje mreža algoritmom propagacije greške unazad



Obučavajući skup i generalizacija

- Neuronska mreža sa **isuviše slobodnih parametara za zadati obučavajući skup** može biti dobro obučena, sa velikom verovatnoćom loše generalizacije.
- Pošto je generalizacija važno svojstvo, razvijeno je više procedura za njeno poboljšanje:
  - **Smanjivanje osetljivosti mreže**.
    - Da bi mreža posedovala dobra svojstva generalizacije, potrebno je da male promene ulazna ne izazivaju velike promene na izlazu mreže.
    - Primena ovog principa je proširivanje obučavajućeg skupa varijacijama ulaznih signala (dodavanje šuma oko svakog elementa obučavajućeg skupa).
  - **Regularizacija**.
    - Proširivanje kriterijumske funkcije tzv. regularizacionim članom.
  - **Rano zaustavljanje** (engl. *early stopping*).
  - **Kresanje** (engl. *prunning*).
    - Ideja kresanja se svodi na ostvarivanje što boljih performansi sa što siromašnijom arhitekturom.

Obučavajući skup i generalizacija



Support Vector Machines

- Metoda vektora oslonca (*Support Vector Machines*, SVM) je linearna metoda učenja koja funkciju odluke traži u skupu funkcija (hipoteza) koje su linearna kombinacija ulaznih vrednosti.
- Podaci koji se u originalnom, ulaznom prostoru ne mogu podeliti linearnim funkcijama, preslikavaju se uz pomoć jezgra u **prostor atributa** (*kernel induced feature space*).
  - Prostor atributa je visokodimenzionalan.
  - U njemu se podaci mogu podeliti upotrebom linearnih hipoteza.
- Preslikavanje povećava ekspresivnost linearnih metoda, ali dovodi i do povećanja rizika od prenaučenosti.
- **Statistička teorija učenja** definiše koje parametre treba kontrolisati kako bi se dostigao odgovarajući stepen generalizacije i smanjio rizik od prenaučenosti.
- **Klasifikator maksimalne margine** ne dopušta pogrešnu klasifikaciju primera za učenje i primenljiv je samo na skupu podataka koji su linearno razdeljivi u prostoru atributa.
- Ovo ograničenje mogućnosti primene motivisalo je razvoj **klasifikatora meke margine**.

Support Vector Machines

- **Formalna definicija**.
- Metoda vektora oslonca se u opštem slučaju definiše kao:

1. Algoritam učenja korišćenjem linearnih metoda
2. u prostoru formiranim funkcijom jezgra,
3. pri čemu se greška generalizacije kontroliše upotrebom statističke teorije učenja i
4. primenjuje teorija optimizacije za rešavanje konveksnog kvadratnog problema na čije se rešavanje svodi učenje metodom SVM.

Binarna Support Vector Machines klasifikacija

- Podaci za problem binarne klasifikacije sastoje se od objekata x_i označenih labelom y_i koja određuje klasu pripadnosti: +1 (pozitivna klasa) ili -1 (negativna klasa).
- Neka je x vektor čije su komponente x_i.
- Linearni klasifikator zasnovan je na **linearnoj funkciji razdvajanja** definisanom skalarnim proizvodom dva vektora:

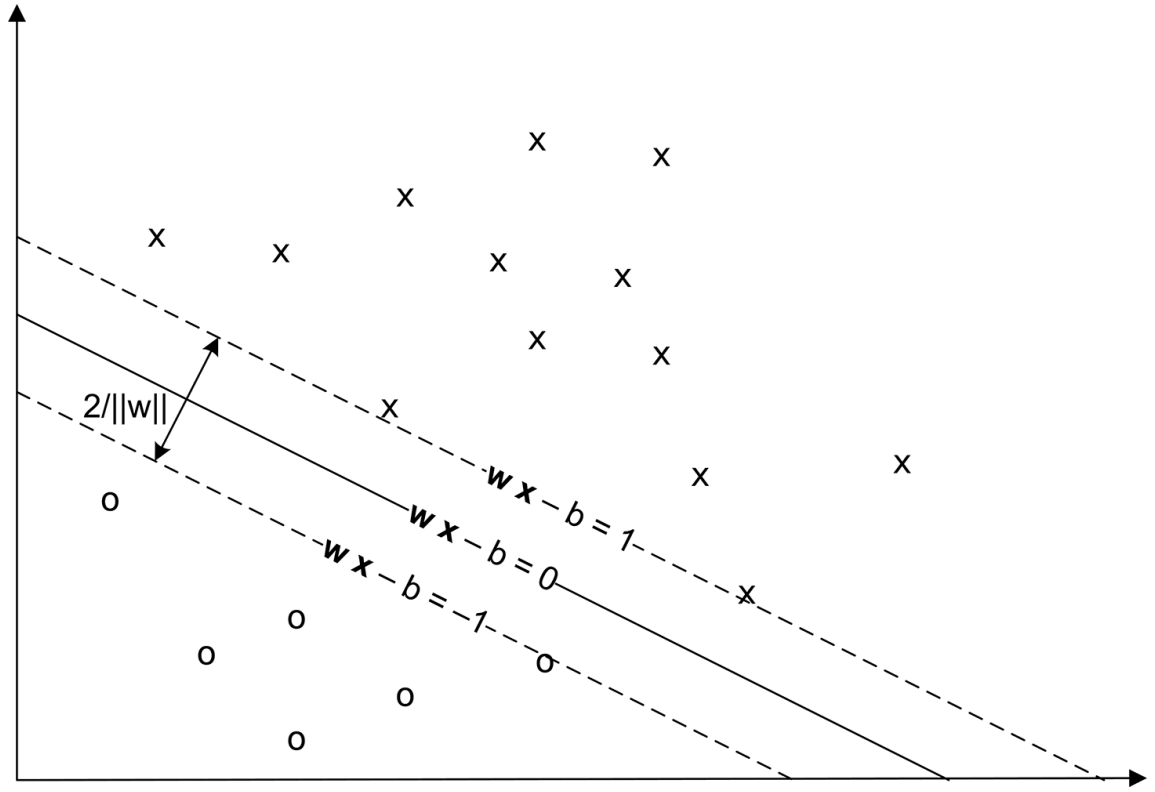
$$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b = \sum w_i x_i + b$$

- Vektor w je težinski vektor, a b pristrasnost kojom je određena udaljenost hiper-ravni u odnosu na koordinatni početak.
- **Hiper-ravan** deli prostor na dva dela, dok znak funkcije $f(x)$ označava stranu hiper-ravni.

$$\left\{\mathbf{x}: f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b = 0\right\}$$

Binarna Support Vector Machines klasifikacija

- Hiper-ravan deli prostor atributa na dva dela u binarnom klasifikatoru maksimalne margine.



Binarna Support Vector Machines klasifikacija

- **Težiski vektor** se može izraziti kao linearna kombinacija obučavajućih primera, što je poznato kao dualno predstavljanje granice odluke.
- Funkcija razdvajanja u tom slučaju ima oblik:

$$f(\mathbf{x}) = \sum \alpha_i x_i{}^{\mathrm{T}} \mathbf{x} + b$$

- Ova jednačina u *feature space* ima oblik:

$$f(\mathbf{x}) = \sum \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b = \sum \alpha_i \Phi(\mathbf{x}_i)^{\mathrm{T}} \Phi(\mathbf{x}) + b$$

- Iako svaka funkcija koja zadovolji uslove date Mercerovom teoremom može biti korišćena kao jezgro, funkcija jezgra $k(\mathbf{x},\mathbf{x}')$ mora biti efikasno izračunljiva pošto prostor atributa može biti visokodimenzionalan.

Klasifikator maksimalne margine

- **Klasifikator maksimalne margine** je funkcija razdvajanja koja maksimizuje geometrijsku marginu $1/\|\mathbf{w}\|$, gde je $\|\mathbf{w}\|$ norma težinskog vektora.
- To je ekvivalentno minimizaciji izraza $\|\mathbf{w}\|^2$, iz čega proizilazi problem uslovne optimizacije: minimizacija izraza $\frac{1}{2}\|\mathbf{w}\|^2$, uz ograničenje:

$$(\forall i = 1,\ldots,n)\, y_i(\mathbf{w}^{\mathrm{T}} x_i + b) \geq 1$$

- Ograničenja u ovoj formulaciji obezđuju da će klasifikator maksimalne margine ispravno klasifikovati svaki primer, što je moguće samo ako su podaci apsolutno linearno razdvojivi.
- U praksi, podaci često nisu linearno razdvojivi, a čak i ako jesu, potrebno je tolerisati neispravnu klasifikaciju određenih tačaka.

Klasifikator meke margine

- Da bi se dozvolila neispravna klasifikacija određenih tačaka, potrebno je prethodnu jednačinu izmeniti uvođenjem promenljive kojom se definiše **tolerancija**:

$$(\forall i = 1,\ldots,n)\, y_i(\mathbf{w}^{\mathrm{T}} x_i + b) \geq 1 - \xi_i$$

- Na taj način se toleriše neispravna klasifikacija primera koji se nalaze u opsegu greške margine ($0 \leq \xi_i \leq 1$)
- Ukoliko nisu u okviru greške margine, primeri su pogrešno klasifikovani ($\xi_i > 1$).
- Granica pogrešno klasifikovanih primera je $\sum \xi_i$.
- **Konstantom meke margine** $C > 0$ određen je odnos između maksimizacije margine i minimizacije stepena tolerancije.
- Klasifikator koji toleriše ispravnu klasifikaciju primera koji se nalaze u opsegu greške naziva se **klasifikator meke margine**.

Binarna Support Vector Machines klasifikacija

- Problem optimizacije klasifikatora meke margine svodi se na minimizaciju izraza::

$$\tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum \xi_i$$

- uz ograničenje: $(\forall i = 1,\ldots,n) y_i(\mathbf{w}^{\mathrm{T}} x_i + b) \geq 1 - \xi_i$
- Primenom Lagranžovih koeficijenata (dualna formulacija), problem optimizacije postaje maksimizovanje izraza:

$$\sum \alpha_i - \tfrac{1}{2}\sum_i \sum_j y_i y_j \alpha_i \alpha_j \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_j$$

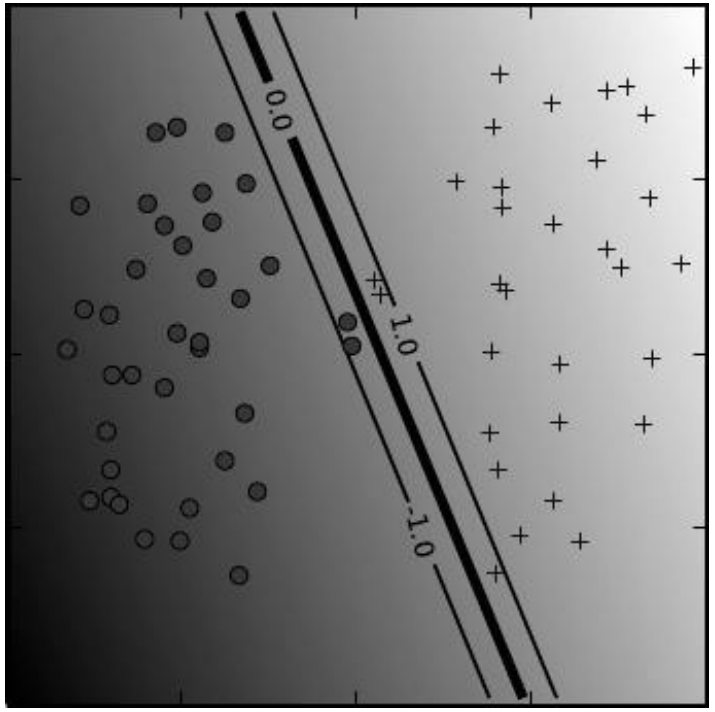
- Na osnovu dualne formulacije težinski vektor može se predstaviti kao: $\mathbf{w} = \sum y_i \alpha_i \mathbf{x}_i = 0$
- Primeri x_i za koje je $\alpha_i > 0$ su tačke koje se nalaze na ili u okviru margine.
- Te tačke nazivaju se **vektori oslonca** (*Support Vectors*).
- Obučavanje klasifikatora svodi se na pronalaženje **optimalne margine hiper-ravni**, tj. na odabir optimalnih parametara α_i i b.

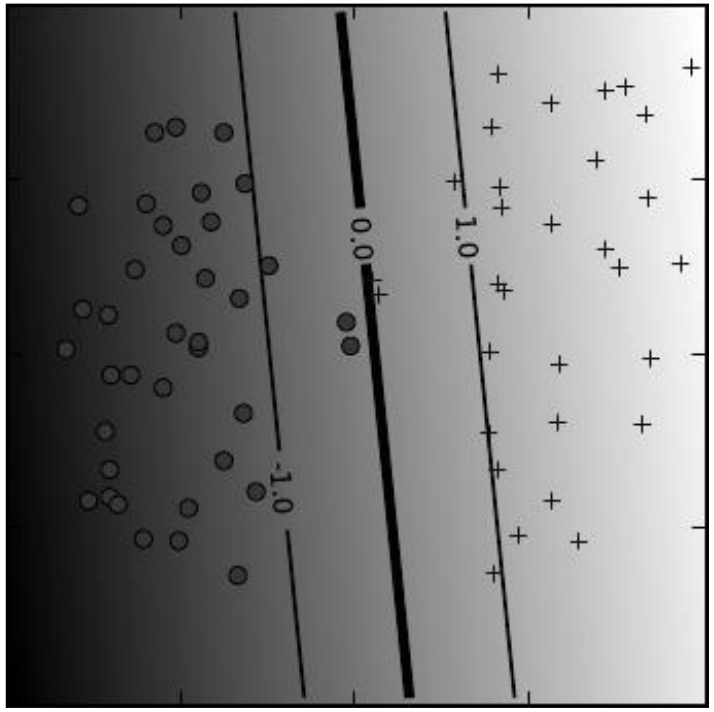
Hiper-parametri klasifikatora

- Klasifikator je takođe definisan takozvanim **hiper-parametrima**:
  - Konstantom meke margine *C*
  - Parametrima konkretnog jezgra.
- Parametri se mogu odabrati empirijski ili upotrebom **unakrsne validacije** (engl. *cross-validation*) i metode ***grid-search*** (pretraga po rešetci).
- Za velike vrednosti parametra *C*, tačke najbliže hiper-ravni utiču na njenu orijentaciju, što za posledicu može imati hiper-ravan koja se nalazi blizu velikog broja drugih tačaka.
- Kada se parametar *C* smanji, te tačke postaju greške margine – orijentacija hiper-ravni se menja, što obezbeđuje dovoljno široku marginu za ostatak podataka.

Uticaj konstante meke margine na klasifikaciju

- Linearno jezgro, C = 100 (levo) i C = 10 (desno).





Odabir kernela

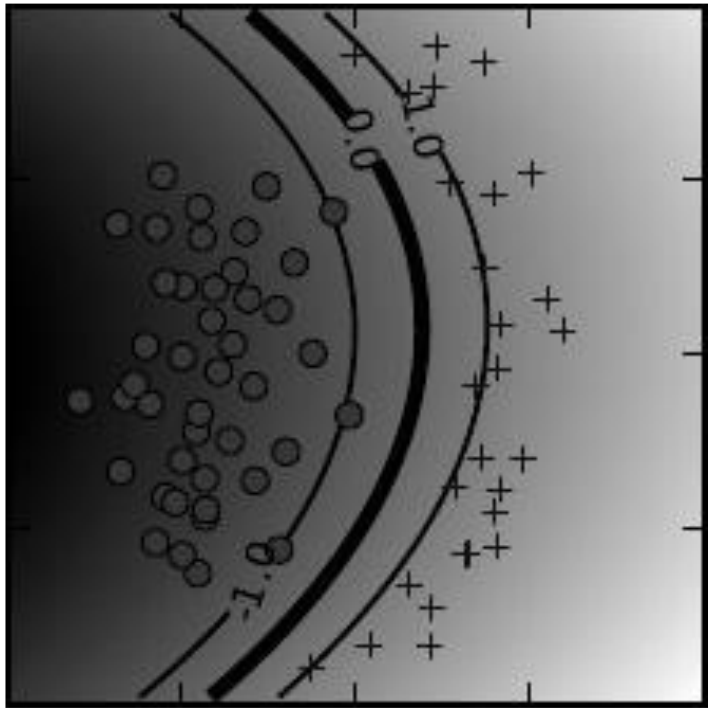
- Odabir jezgra **zavisi od konkretnog problema koji se rešava**.
- Na primer, gausovo tj. RBF (*Radial Basis Function*) jezgro definisano izrazom:

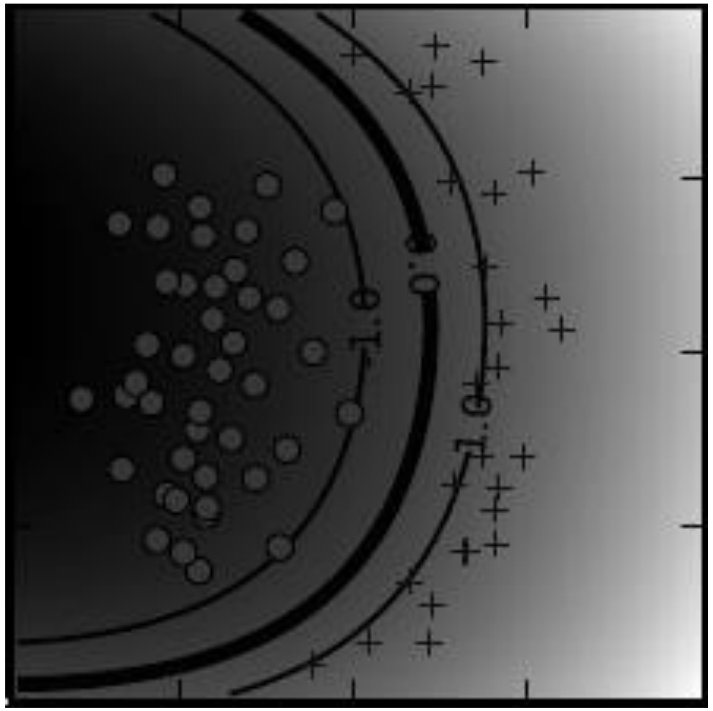
$$k(\mathbf{x},\mathbf{x}') = e^{-\gamma\left\|\mathbf{x}-\mathbf{x}'\right\|^2}$$

  postiže bolje performanse po pitanju tačnosti klasifikacije i vremenu konvergencije u odnosu na polinomijalno.
- Parametar $\gamma = 1/2\sigma^2$ ima sličnu ulogu kao stepen u polinomnom jezgru i služi za kontrolu fleksibilnosti klasifikatora.
- Ukoliko je polinomijalno jezgro stepena dva dovoljno fleksibilno da diskriminiše klase, onda će jezgro stepena pet formirati sličnu granicu odlučivanja, ali sa većom zakrivljenošću.
- Slično, povećanje vrednosti parametra γ dovodi do veće lokalizacije vektora oslonaca i veče zakrivljenosti granice odlučivanja.
- Preterano velika vrednost parametra γ može dovesti klasifikator u stanje prenaučenosti.

Ponašanje polinomijalnog jezgra u odnosu na stepen

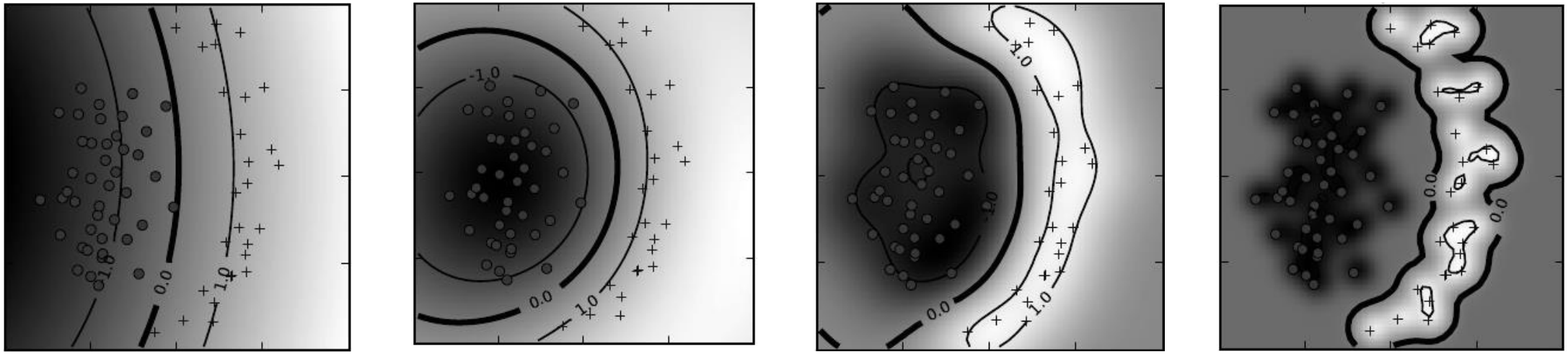
- Polinomijalno jezgro: $d = 2$ (levo) i $d = 5$ (desno).





Ponašanje RBF jezgra u odnosu na parametar γ

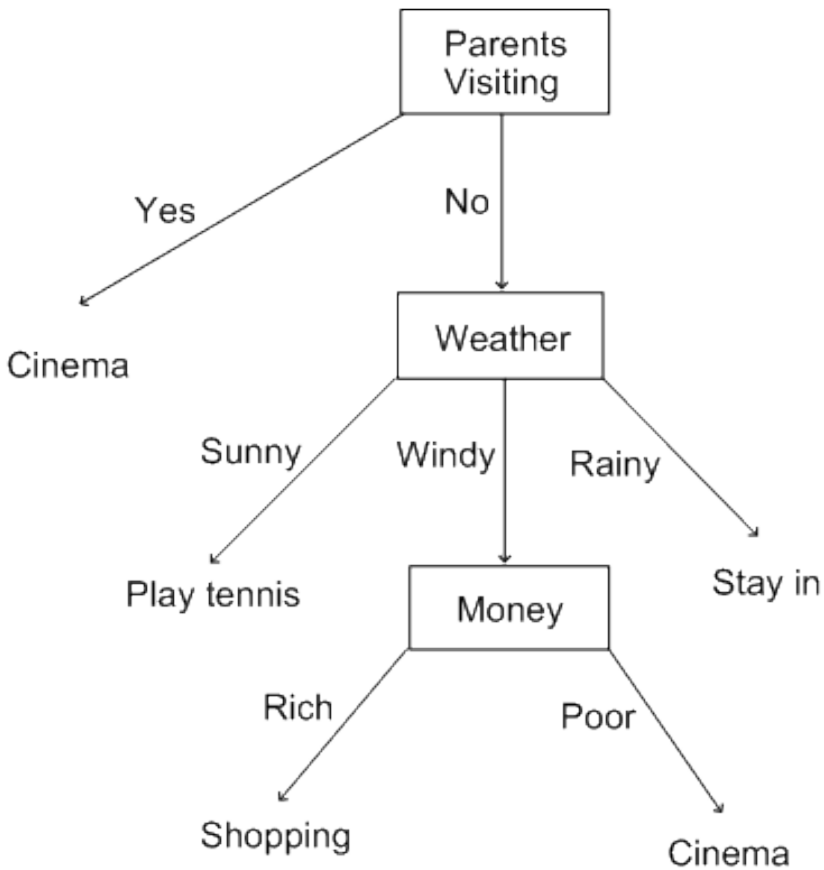
- RBF jezgro: $\gamma = 0{,}1$; $\gamma = 1$; $\gamma = 10$; $\gamma = 100$ (redom, s leve ka desnoj slici).

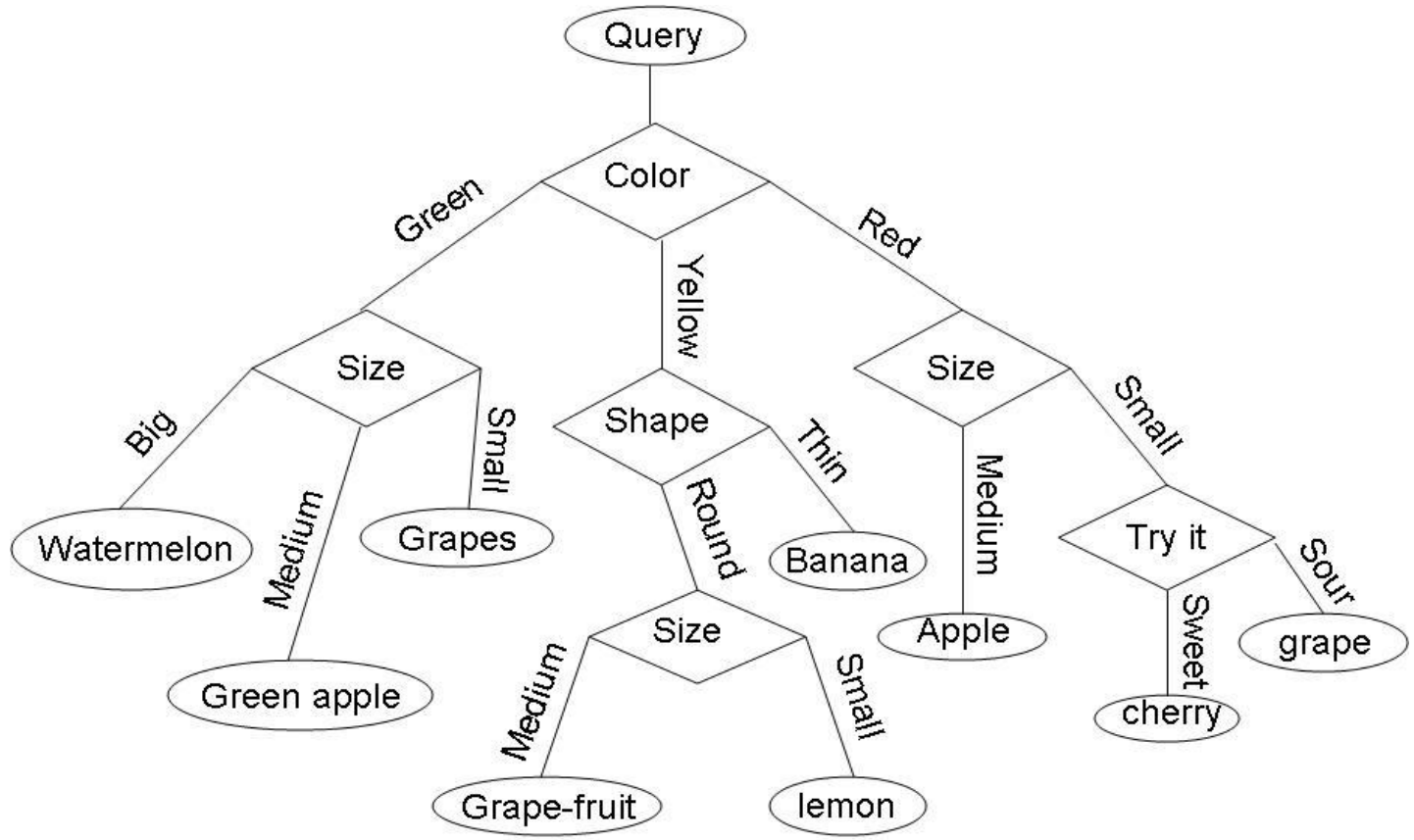


Stabla odlučivanja

- **Stablo odlučivanja** (decision tree) je klasifikacioni algoritam u formi stablaste strukture.
- Stablo sadrži dva tipa čvorova povezanih granama.
  - **Krajnji čvor**, kojim se završava odredjena grana stabla, definiše klasu kojoj pripadaju primeri koji zadovoljavaju uslove na toj grani stabla.
  - **Čvor odluke** definiše uslov u formi vrednosti određenog atributa. Iz čvora odluke izlaze grane koje zadovoljavaju određene vrednosti tog atributa.
- Kretanjem po granama stabla koje instanca vrednostima svojih atributa zadovoljava, od čvora odlučivanja u korenu stabla do krajnjeg čvora koji klasifikuje instancu u jednu od postojećih klasa problema.
- Da bi se tehnika stabla odlučivanja upotrebila za klasifikaciju, instance moraju biti opisane konačnim brojem atributa, prethodno mora biti definisan konačan broj klasa.
- Svaka instanca možea pripadati samo jednoj od postojećih klasa, kojih mora biti znatno manje od broja instanci.

Stabla odlučivanja



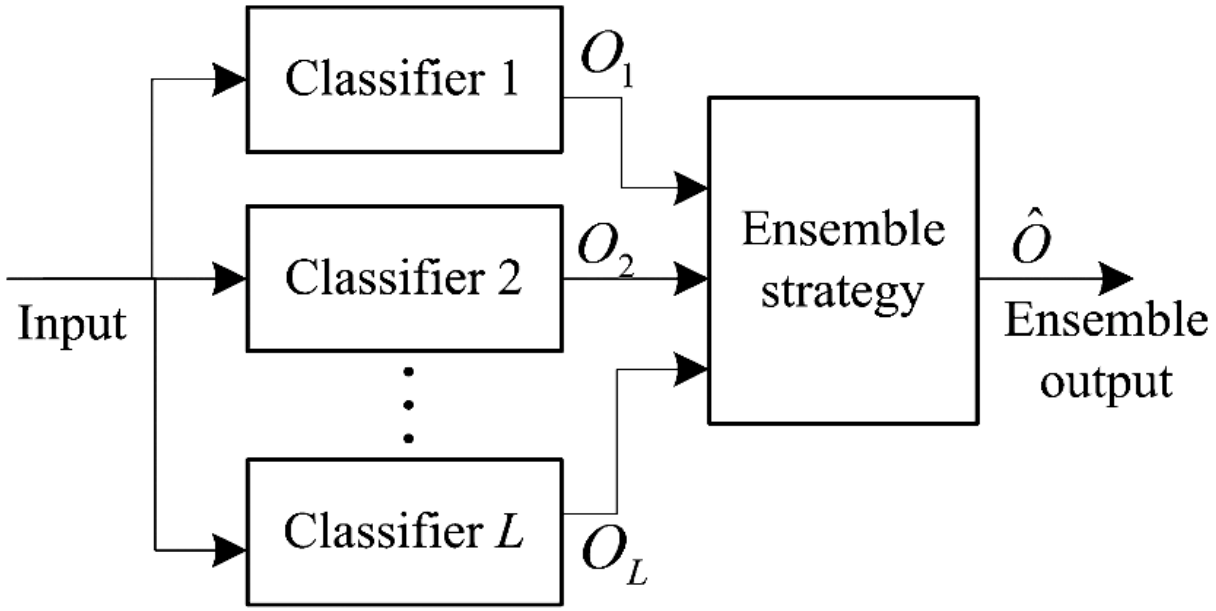


Stabla odlučivanja

- Većina postojećih algoritama za formiranje stabla odlučivanja su varijacije osnovnog algoritma **ID3** (*Iterative Dichotomiser*).
  - ID3 pretražuje atribute svih instanci u skupu podataka.
  - Ukoliko atribut savršeno razdvaja klase, algoritam se zaustavlja.
  - U suprotnom, algoritam se rekurzivno izvršava na *m* podskupova (gde je *m* broj mogućih vrednosti atributa), tražeći najbolje atribute za njihovo razdvajanje.
  - Algoritam traži trenutno najbolji atribut i nikad ne proverava ispravnost prethodnih razdvajanja.
  - Osnovni deo algoritma je selekcija atributa sa najheterogenijom strukturom vrednosti za stvaranje čvora odlučivanja upotrebom koncepta entropije.
- Stabla odlučivanja imaju svoje nedostatke:
  - Mogućnost generalizacije je manja u odnosu na druge metode.
  - Empirijski je utvrđeno da su stabla odlučivanja jako osetljiva na podatke u obučavajućem skupu i da ne mogu da se obučavaju na neuravnoteženim skupovima.

Ensembli

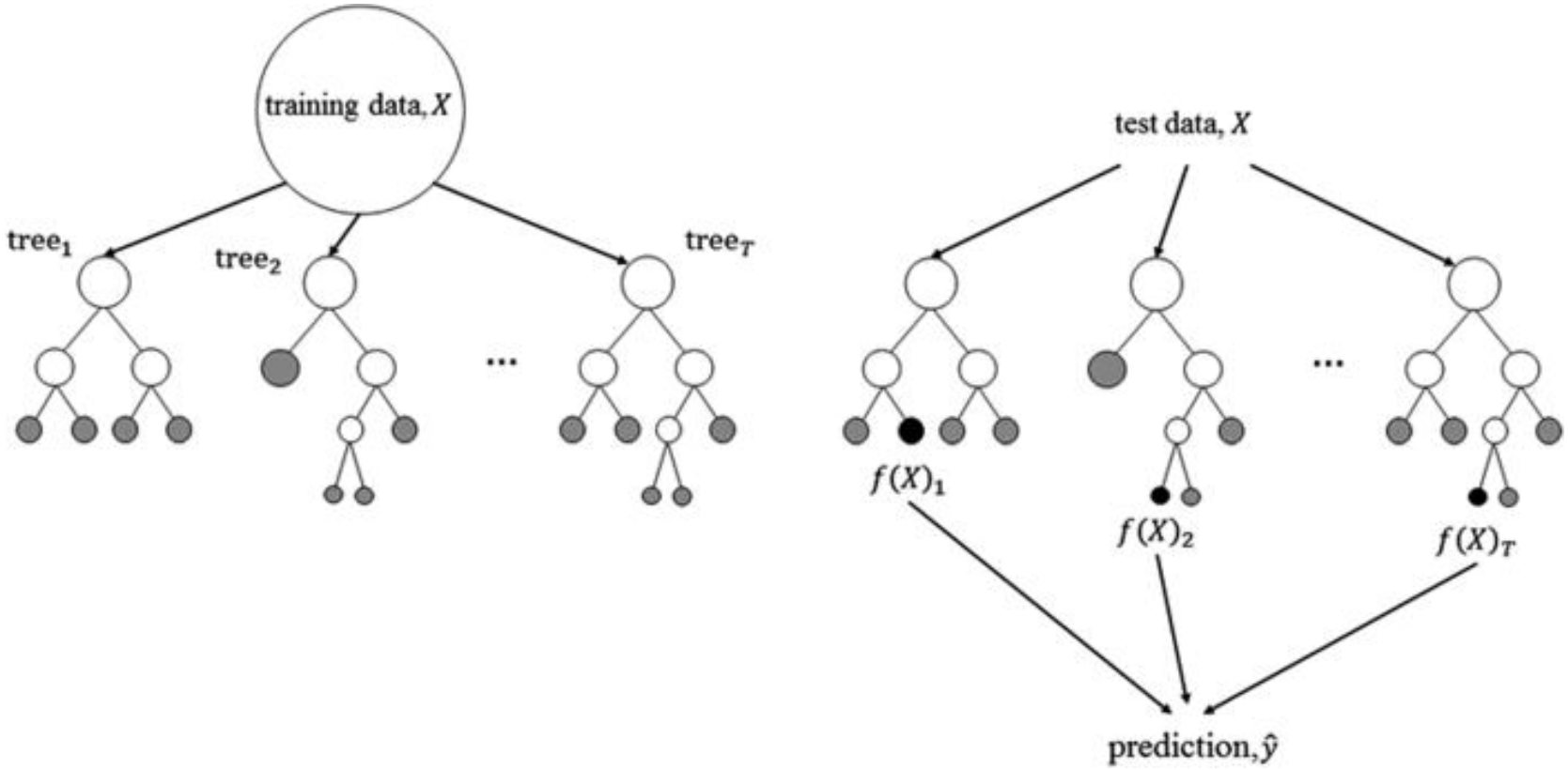
- **Ensembli** koriste više algoritama učenja kako bi ostvarili bolje prediktivne performance od pojedinačnih konstitutivnih algoritama.
- Ensembli se najčešće formiraju upotrebom *boosting*, *bagging* i *stacking* tehnika.



Ensembli: boosting, bagging, stacking

- **Boosting** inkrementalno gradi ensembl.
  - Svaka nova instanca modela se obučava sa naglasom greške u prethodnoj instanci.
  - Naglašavaju se instance obučavajućeg skupa koje su prethodne instance modela pogrešno klasifikovale.
- **Bagging ensemblu** zasnovan je na tehnici glasanja (engl. voting).
  - Svaki model, obučen nasumično odabranim podskupom obučavajućeg skupa, ima jednako parvo glasa.
  - Primer: **random forest** je bagging ensembl koji kao modele koristi slučajno formirana stabla odlučivanja i veoma visoku tačnost klasifikacije.
- **Stacking** gradi hibridni ensembl obučavanjem kombajnera izlazima nekoliko drugih algoritama obučenih raspoloživim podacima.
  - Kombajner određuje konačnu predikciju koristeći izlaze ostalih algoritama kao dodatne ulaze.
  - U praksi se kao kombajner često koristi *single-layer logistic regression* model.

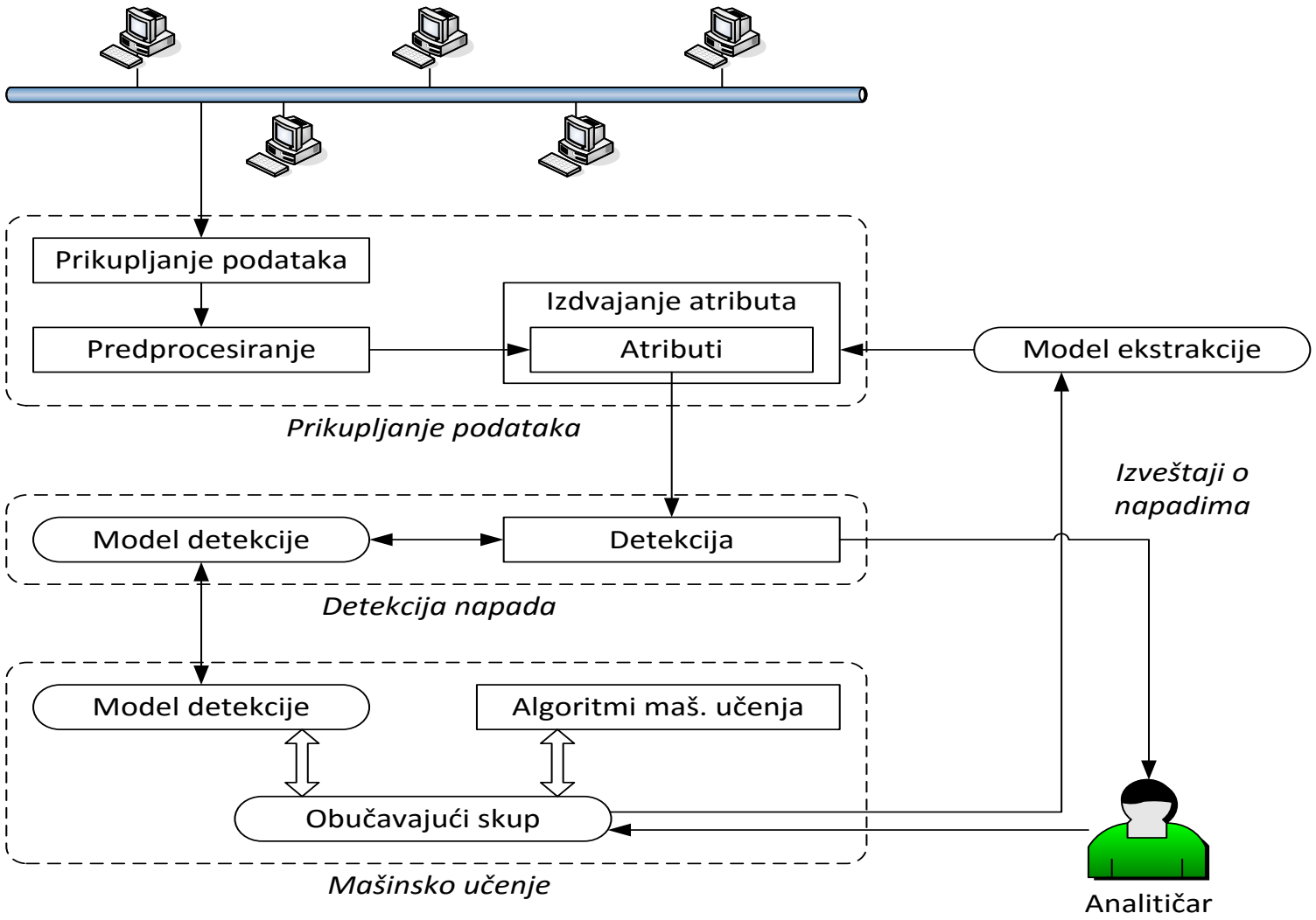
Random Forest



Mičelova definicija mašinskog učenja u kontekstu IDS sistema

- Formalna definiciju mašinskog učenja po Tom Mičel-u: „program uči iz **iskustva E** u odnosu na **klasu zadataka T** i **meru performansi P**, ako se mera performansi P poboljšava u vezi zadatka T nakon iskustva E“.
- Shodno ovoj formalnoj definiciji:
  - IDS uči da klasifikuje događaje (zadatak T)
  - Mera performansi P ovog zadatka je tačnost klasifikacije
  - Iskustvo E je zadati obučavajući skup pravila.

Radni okvir IDS sistema zasnovanih na mašinskom učenju



Mašinsko učenje i detekcija potpisa

- IDS **detektuje potpise** tako što:
  - prikuplja podatke sa senzora,
  - predprocesira podatke (uklanja šum, izdvaja atribute i normalizuje njihove vrednosti),
  - identifikuje pretnju na osnovu potpisa (poredi vrednosti atributa sa svim pravilima u bazi),
  - ažurira pravila,
  - oglašava se alarmom ili reaguje na napad.
- Tačnost sistema za detekciju potpisa zavisi od znanja koje je smešteno je u bazi pravila kojima su opisani napadi.
- Sistem zasnovan na jednostavnoj pretrazi baze potpisa ne može da otkrije napade koji nisu opisani ni jednim pravilom.

Mašinsko učenje i detekcija potpisa

- Otkrivanje varijacije poznatih napada je moguće ukoliko se u sistem ugradi pogodna metoda mašinskog učenja u fazi predprocesiranja ili klasifikacije.
- Pošto su podaci u bazi pravila označeni, najčešće se koriste metode **nadgledanog učenja** (veštačke neuronske mreže, metode vektora oslonca, stabla odlučivanja).
- Tehnike detekcije u tom slučaju zasnovane su na **merama sličnosti** tekuće aktivnosti i potpisa poznatih napada.
- Aktivnost dovoljno slična nekom potpisu klasifikuje se kao napad.
- Nedostatak ovakvih sistema je nemogućnost otkrivanja napada koji nisu slični ni jednom potpisu.

Mašinsko učenje i detekcija anomalija

- IDS **detektuje anomalije** tako što:
  - prikuplja podatke sa senzora,
  - predprocesira podatke,
  - gradi profil normalnog ponašanja mašinskim učenjem (najčešće su označeni podaci koji odgovaraju normalnom ponašanju, dok anomalije nisu),
  - identifikuje napad na osnovu odstupanja od profila normalnog ponašanja,
  - oglašava se alarmom ili reaguje na napad.
- Identifikacija napada najčešće se obavlja metodama mašinskog učenja koje klasifikuju instance pomoću mera sličnosti između ulaza i profila normalnog ponašanja.
  - Primer: metoda najbližih suseda.
- Osnovni nedostatak tehnika za detekciju anomalija je veliki broj lažnih alarma nastalih usled evolviranja normalnog ponašanja.

Analiza obučavajućih skupova: KDD Cup’99

- Skup podataka KDD Cup ’99 sastoji se od:
  - potpunog obučavajućeg skupa (oko 4.900.000 instanci),
  - 10% obučavajućeg skupa (sadrži približno 10% instanci iz potpunog obučavajućeg skupa),
  - skupa za testiranje, koji sadrži nove tipove napada.
- Prema Kendalu, napadi iz skupa KDD Cup ’99 mogu se klasifikovati u četiri kategorije:
  - ispitivački napadi (**probing**),
  - napadi odbijanja usluga (Denial of Service, **DoS**),
  - neovlašćeno povećanje privilegija (User to Root, **U2R**),
  - neovlašćenog sticanje pristupa udaljenom računaru (Remote to Local, **R2L**).

Analiza obučavajućih skupova: KDD Cup’99

- Obeležja instanci skupa KDD Cup’99
  - 32 numerička obeležja, na primer:
    - trajanje konekcije,
    - broj bajtova poslatih sa izvorišta ka odredištu,
    - broj neispravnih fragmenata,
    - broj operacija koje kreiraju nove datoteke,
    - broj operacija koje menjaju prava pristupa,
    - broj aktivnih komandnih interpretera.
  - 6 binarnih obeležja, na primer:
    - obeležje koji ukazuje na napad tipa „land“, tj proverava da li su odredišna i izvorišna adresa identične,
    - obeležje koje ukazuje na pokušaj pokretanja komande kojom se stiču administrativne privilegije na operativnom sistemu.
  - 3 kategorička obeležja (protokol, odredišni servis i fleg).

Analiza obučavajućih skupova: KDD Cup'99

- Kategorije obeležja kojima su opisane instance skupa KDD Cup'99:
  - **Osnovna obeležja** određuju se vrednostima iz zaglavlja paketa.
    - Primer: trajanje, protokol i servis.
  - **Obeležja zasnovana na sadržaju** određuju se analizom sadržaja TCP paketa.
    - Primeri:
      - „failed logins", tj. broj neuspešnih pokušaja prijavljivanja na sistem.
      - „root shell" koji određuje da li je otvoren komandni interpreter sa root privilegijama.
  - **Vremenska obeležja** saobraćaja opisuju osobine koje zastarevaju nakon isteka definisanog vremenskog intervala.
    - Primer: broj konekcija ka istom sistemu u zadatom vremenskom intervalu.
  - **Sistemska obeležja saobraćaja**, za razliku od vremenskih, zasnovani su na prozoru čiji je interval zadat brojem konekcija.
    - Sistemska obeležja su pogodna za opisivanje napada čije je trajanje duže od intervala propisanog vremenskim obeležjima.

Analiza obučavajućih skupova: KDD Cup’99

- **Kritička analiza**.
- Osnovni nedostatak: skup NIJE verodostojna simulacija realnog mrežnog saobraćaja.
  - Broj instanci napada je nerealistično veliki u odnosu na broj instanci normalnog saobraćaja.
  - Odnos instanci različitih kategorija napada u skupu je nerealan.
- Precizna definicija napada ne postoji.
  - Primer: ispitivanje mreže se ne smatra napadom ukoliko administrator proverava da li su zaštitini mehanizmi ispravno konfigurisani.
- Skupovi je isuviše složeni za formiranje modela zasnovanog na mašinskom učenju koji je sposoban da dovoljno tačno detektuje R2L i U2R napade.
  - Razlozi:
    - Jako mali broj instanci U2R napada.
    - Postojanje instanci R2L koje su identične ili veoma slične normalnom saobraćaju
    - Nepostojanje nekih R2L instanci u test skupu
- Postojanje duplikata.

Diskusija i praktična demonstracija

- Sinteza obučavajućih skupova.
- Praktična demonstracija.

Literatura

1. D. Pleskonjić, N. Maček, B. Đorđević, M. Carić (2007): Sigurnost računarskih sistema i mreža. Mikro knjiga, Beograd.
2. M. Milosavljević (2015): Veštačka inteligencija. Univerzitet Singidunum.
3. M. Milosavljević: predavanja iz predmeta Veštačka inteligencija i Mašinsko učenje, Univerzitet Singidunum.
4. N. Maček, M. Milosavljević (2014): Reducing U2R and R2L False Positive Rates with Support Vector Machines. SJEE Vol. 11, Issue 1, pp 175-188.
5. N. Maček (2015): Detekcija upada mašinskim učenjem / Machine Learning in Intrusion Detection. Zadužbina Andrejević.

Hvala na pažnji

Pitanja su dobrodošla.