

Istraživanje podataka 2

2025/2026. šk. godina

19. Januar 2026

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

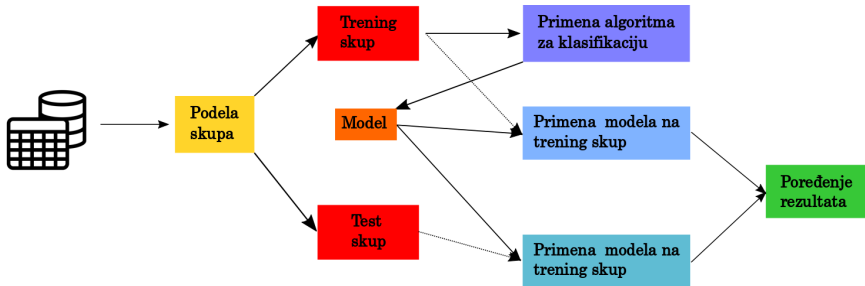
Klasifikacija

- Ulazni podaci: svaki slog (instanca) je oblika (x, y) gde je x skup (ulaznih) atributa, a y je ciljni atribut (klasa).
- Cilj klasifikacije: pronaći funkciju f (model klasifikacije) koja preslikava skup atributa x u jednu od predefinisanih oznaka klase y .
- Podela skupa na trening i test skup.

Klasifikacija

Drveća odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambel metode

Klasifikacija



Klasifikacija - mere za ocenu modela

- preciznost (eng. accuracy)

$$\text{preciznost} = \frac{\text{Broj slogova čija klasa je dobro predviđena modelom}}{\text{Ukupan broj slogova}}$$

- stopa greške, (eng. error rate)

$$\text{stopa greške} = \frac{\text{Broj slogova čija klasa nije dobro predviđena modelom}}{\text{Ukupan broj slogova}}$$

Klasifikacija

- Problem preprilagođavanja (eng. overfitting) - model se previše prilagodi specifičnostima trening podataka, zbog čega postiže visoku tačnost na trening skupu, ali daje loše rezultate na novim, neviđenim podacima.

Klasifikacija
Drveta odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

Drveta odlučivanja

- Model klasifikacije se predstavlja kao drvo odlučivanja koje ima
 - unutrašnje čvorove. Svaki unutrašnji čvor sadrži uslov nad test atributom koji služi za podelu slogova koji imaju različite karakteristike tako da se dobiju *čistije* grupe slogova. Grane koje izlaze iz unutrašnjeg čvora odgovaraju mogućim vrednostima test atributa.
 - listove. Svakom listu je dodeljena jedna klasa.

Klasifikacija

Drveta odlučivanja

Naivni Bajesovski klasifikatori

K najbližih suseda

Veštačke neuronske mreže

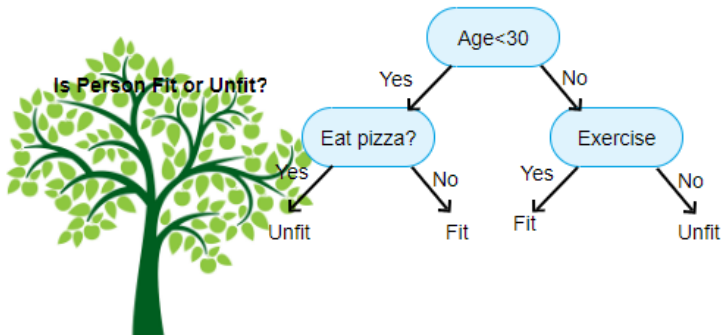
SVM - Metod podržavajućih (potpornih) vektora

Klasifikacija - mere za ocenu modela

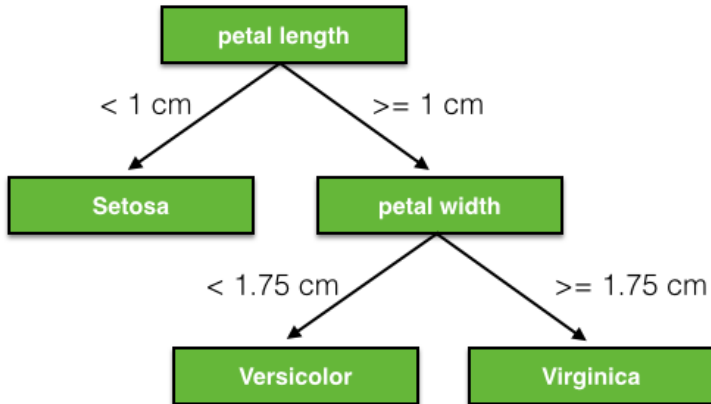
Unakrsna validacija

Ansambl metode

Primer drveta odlučivanja



Primer drveta odlučivanja



Drveta odlučivanja - klasifikacija sloga

Klasifikacija sloga: počevši od korena drveta odlučivanja, primenjuje se test uslov nad slogom i prati se grana koja odgovara dobijenom rezultatu. Ukoliko se pri spuštanju niz drvo odlučivanja naiđe na unutrašnji čvor, postupak se ponavlja (test uslov se primenjuje na slog i prati se grana koja odgovara rezultatu testa). Ako se naiđe na list, slogu se dodeljuje klasa koja je pridružena tom listu.

Drveta odlučivanja - pravljenje drveta odlučivanja

Opšti algoritam

- 1 Neka je D_t skup slogova za trening koji se nalaze u čvoru t , a $y = y_1, \dots, y_c$ su oznake klase
- 2 Ako D_t sadrži samo slogove koji pripadaju jednoj klasi y_t , tada je t list označen sa y_t
- 3 Ako D_t sadrži slogove koji se nalaze u više od jedne klase, tada se koristi test atribut radi podele podataka u manje podskupove. Na dobijene podskupove se zatim rekurzivno primenjuje kompletna procedura.

Mere nečistoće

$p(j|t)$ je relativna frekvencija klase j u čvoru t

Ginijev indeks

$$Gini(t) = 1 - \sum_j [p(j|t)]^2$$

Entropija

$$Entropy(t) = - \sum_j p(j|t) * \log_2 p(j|t)$$

Greška klasifikacije

$$Error(t) = 1 - \max_j p(j|t)$$

Klasifikacija

Drvena odlučivanja

Naivni Bajesovski klasifikatori

K najbližih suseda

Veštačke neuronske mreže

SVM - Metod podržavajućih (potpornih) vektora

Klasifikacija - mere za ocenu modela

Unakrsna validacija

Ansambel metode

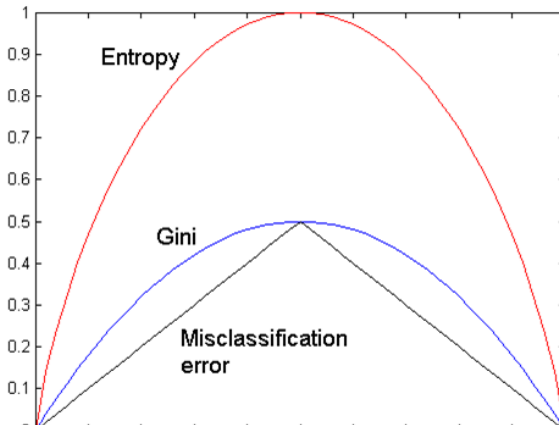
Mere nečistoće

Dobit

$$\Delta = I(\text{parent}) - \sum_{j=1}^k \frac{N(v_j)}{N} * I(v_j)$$

Klasifikacija
Drvea odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Mere nečistoće



Drveta odlučivanja - podela prema tipu atributa

- Imenski atributi: binarna ili višestruka podela
- Redni atributi: binarna ili višestruka podela vodeći računa o uređenju
- Neprekidni atributi: potrebno je pronaći najbolju tačku/tačke prekida za binarna ili višestruku podelu

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori**
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

Naivni Bajesovski klasifikatori

- Uslovna verovatnoća

$$P(C|A) = \frac{P(A, C)}{P(A)}$$

- Verovatnoću da se zajedno dese događaj A i događaj C možemo računati sa

$$P(A, C) = P(C|A) * P(A)$$

kao i sa

$$P(A, C) = P(A|C) * P(C)$$

Naivni Bajesovski klasifikatori

- Bajesova teorema

$$P(C|A) = \frac{P(A|C) * P(C)}{P(A)}$$

- Pretpostavka o nezavisnosti između atributa

$$P(C|A_1, A_2, \dots, A_n) = \frac{\prod_{i=1}^n P(A_i|C) * P(C)}{P(A)}$$

- Određivanje klase

$$\hat{C} = \arg \max_C \prod_{i=1}^n P(A_i|C) * P(C)$$

Zadatak 1

Dat je skup podataka:

Record	A	B	C	Class
1	0	0	0	+
2	0	0	1	-
3	0	1	1	-
4	0	1	1	-
5	0	0	1	+
6	1	0	1	+
7	1	0	1	-
8	1	0	1	-
9	1	1	1	+
10	1	0	1	+

Zadatak 1

Da bismo odredili klasu test instance X , računamo uslovne verovatnoće $P(+|X)$ i $P(-|X)$.

$$P(+|X) = \frac{P(A = 0|+) * P(B = 1|+) * P(C = 0|+) * P(+)}{P(X)}$$

$$P(A = 0|+) = \frac{2}{5}$$

$$P(B = 1|+) = \frac{1}{5}$$

$$P(C = 0|+) = \frac{1}{5}$$

$$P(+) = \frac{1}{5}$$

Zadatak 1

$$P(+|X) = \frac{P(A = 0|+) * P(B = 1|+) * P(C = 0|+) * P(+)}{P(X)} =$$
$$\frac{\frac{2}{5} * \frac{1}{5} * \frac{1}{5} * \frac{1}{2}}{P(X)} =$$
$$\frac{\frac{1}{5^3}}{P(X)}$$

Zadatak 1

$$P(-|X) = \frac{P(A = 0|-) * P(B = 1|-) * P(C = 0|-) * P(-)}{P(X)}$$

$$P(A = 0|-) = \frac{3}{5}$$

$$P(B = 1|-) = \frac{2}{5}$$

$$P(C = 0|-) = 0$$

$$P(-) = \frac{1}{2}$$

Zadatak 1

$$P(-|X) = \frac{P(A = 0|-) * P(B = 1|-) * P(C = 0|-) * P(-)}{P(X)} =$$
$$\frac{\frac{3}{5} * \frac{2}{5} * 0 * \frac{1}{2}}{P(X)} = 0$$

Kako je $P(+|X) > P(-|X)$ test instancu X klasifikujemo klasom $+$.

Klasifikacija
Drveta odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda**
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

K najbližih suseda

- K najbližih suseda - KNN
- Klasifikacija instance je zasnovana na sličnosti sa drugim instancama
- Test instanca se klasifikuje tako što se u trening skupu pronalazi k najbližih instanci (suseda) i test instanci se dodeljuje klasa koja je najzastupljenija među k najbližih suseda.

K najbližih suseda

Osnovni parametri algoritma KNN su:

- k - broj suseda
- *dist* - funkcija za računanje rastojanja između instanci

K najbližih suseda

Neka je:

- k zadati broj suseda
- D skup instanci za treniranje; svaka instanca je predstavljena sa (x, y) gde su x vrednosti atributa za predviđanje, a y klasa kojoj instanca pripada
- svaka test instanca z je predstavljena sa (x', y') gde su x' vrednosti atributa za predviđanje, a y' dodeljena klasa

K najbližih suseda

Algoritam

Za svaku test instancu $z = (x', y')$ uraditi

- izračunatu $\text{dist}(x', x)$, rastojanje između instance z i svake instance $(x, y) \in D$
- izdvojiti $D_z \subseteq D$, skup k najbližih trening instanci test instanci z
- $y' = \arg \max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$

K najbližih suseda

- U najjednostavnijem obliku algoritma KNN, svaki sused ima istu težinu, te klasa koja je najzastupljenija među susedima test instance se dodeljuje test instanci.
- Nekada je bolje susedima dodeliti težine tako da bliži susedi imaju veći uticaj pri određivanju klase test instance. Tada su težine suseda proporcionalne vrednosti $\frac{1}{d}$, gde je d rastojanje suseda od test instance.

K najbližih suseda

Za računanje rastojanja između instanci najčešće se koriste Euklidsko i Menhetn rastojanje, zbog čega je u okviru preprocesiranja podataka potrebno izvršiti:

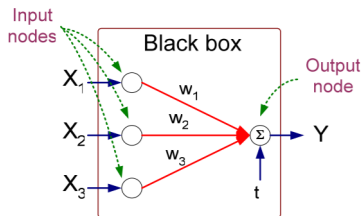
- pretvaranje kategoričkih atributa u numeričke
- transformisati numeričke attribute tako da imaju isti ili sličan opseg vrednosti

Klasifikacija
Drveta odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 **Veštačke neuronske mreže**
- 6 SVM - Metod podržavajućih (potpornih) vektora

Perceptron - neuronska mreža bez skrivenih slojeva



Model Perceptrona

$$Y = I(\sum_i w_i X_i - t) \quad \text{or}$$

$$Y = \text{sign}(\sum_i w_i X_i - t)$$

Perceptron

Učenje modela

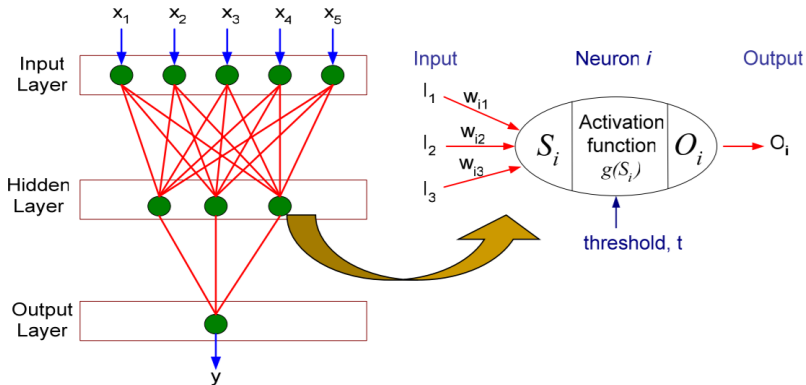
$$w_j^{(k+1)} = w_j^{(k)} + \eta(y_i - \hat{y}_i^{(k)})x_{ij}$$

Za

- $y = +1$ i $\hat{y} = -1$ važi $(y - \hat{y}) = 2$
- $y = -1$ i $\hat{y} = +1$ važi $(y - \hat{y}) = -2$
- za linearno razdvojive probleme klasifikacije

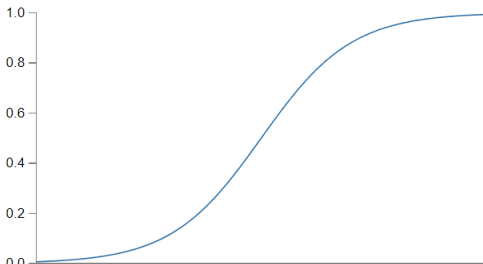
Klasifikacija
Drveća odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Neuronska mreža sa skrivenim slojevima

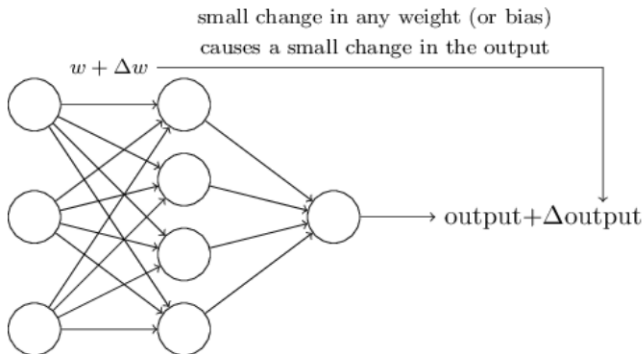


Neuronska mreža sa skrivenim slojevima

Koriste se i druge aktivacione funkcije (osim *sign*). Npr. pozadinska (eng. logistic/sigmoid) funkcija $\frac{1}{1+e^{-z}}$ kod koje mala promena u težinama dovodi do male promene u izlazu (rezultatu aktivacione funkcije).



Neuronska mreža sa skrivenim slojevima



Pristup gradijentni spust

Gradijentni spust je algoritam za nalaženje lokalnog minimuma funkcije.

Cilj - minimizacija zbira kvadrata greške

$$E(w) = \frac{1}{2} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

Učenje modela

$$w_j = w_j - \eta * \frac{\partial E(w)}{\partial w_j}$$

Pristup gradijentni spust

- Kod višeslojnih mreža, problem je što stvarni izlazi skrivenih čvorova nisu poznati, jer ne postoje trening oznake za njihove izlaze.
- Potrebna neka vrsta „povratne informacije“ iz kasnijih slojeva ka ranijim slojevima o očekivanim izlazima i greškama radi ažuriranja težina tih čvorova.

Pristup gradijentni spust

Algoritam propagacije unatrag (eng. backpropagation).

Faze:

- *unapred* - težine iz prethodne iteracije se koriste za računanje izlazne vrednosti svakog neurona. Konačni izlaz se poredi sa klasnom oznakom kako bi se utvrdilo da li je došlo do greške.
- *unazad* - formula za ažuriranje težina se primenjuje u obrnutom smeru

Ciljni atribut sa više klasa

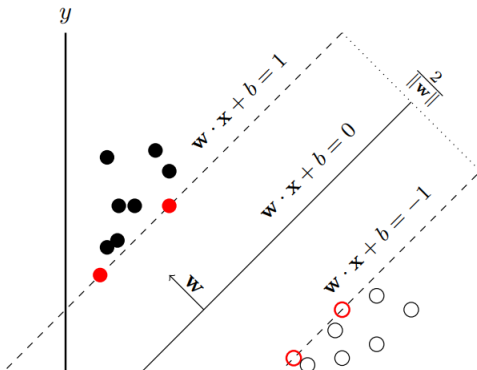
- jedan izlazni neuron za svaku klasu, bira se klasa čiji neuron da najveći izlaz
- funkcija aktivacije izlaznih neurona - funkcija mekog maksimuma (eng. softmax)
$$\text{softmax}(x) = \left(\frac{e^{x_1}}{\sum_{i=1}^C e^{x_i}}, \dots, \frac{e^{x_C}}{\sum_{i=1}^C e^{x_i}} \right)$$

gde je x_i izlaz neurona vezanog za klasu i .
- funkcija gubitka - unakrsna entropija

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora**

Metod podržavajućih (potpornih) vektora - linearno separabilan skup



Metod podržavajućih (potpornih) vektora - linearno separabilan skup

- Pretpostavka: podaci su linearno razdvojitivi
- Klase: 1 i -1
- Cilj: naći optimalnu hiper-ravan, tj. hiper-ravan sa maksimalnom marginom koja razdvaja instance klase 1 i instance klase -1

$$w * x + b = 0$$

- w je normalni pravac hiper-ravni, a b je skalar, tj. pomak

Metod podržavajućih (potpornih) vektora - linearno separabilan skup

- Najbliže instance trening skupa iz različitih klasa nazivaju se podržavajući (potporni) vektori
- Maksimalna margina se određuje rešavanjem optimizacionog problema nelinearnog programiranja koji maksimizuje marginu tako što se ona izražava kao funkcija koeficijenata razdvajajuće hiper-ravni

Metod podržavajućih (potpornih) vektora - linearno separabilan skup

- Ograničenja margina

$$w * x + b = 1$$

$$w * x + b = -1$$

- Ograničenja za instance

- za $y_i = 1$ $w * x_i + b \geq 1$
- za $y_i = -1$ $w * x_i + b \leq -1$
- ili $\forall i$ iz skupa $y_i * (w * x_i + b) \geq 1$

Metod podržavajućih (potpornih) vektora - linearno separabilan skup

- Rastojanje od hiper-ravni: $\frac{|w \cdot x + b|}{||w||}$
- Rastojanje od hiper-ravni do podržavajućih vektora: $\frac{1}{||w||}$
- Margina: $\frac{2}{||w||}$
- Zahtev

$$\max_w \frac{2}{||w||}$$

uz ograničenja $y_i * (w * x_i + b) \geq 1, i = 1, 2, \dots, N$

- Ili

$$\min_w \frac{||w||^2}{2}$$

uz ograničenja $y_i * (w * x_i + b) \geq 1, i = 1, 2, \dots, N$

Metod podržavajućih (potpornih) vektora - linearno separabilan skup

- Optimizacioni problem može da se reši korišćenjem metoda Lagranžovi množioci
- $$L_p = \frac{\|w\|^2}{2} - \sum_{i=1}^N \lambda_i * [y_i * (w * x_i) + b - 1]$$
- nenegativni Lagranžovi množioci λ_i se povezuju sa ograničenjima

Metod podržavajućih (potpornih) vektora - linearno separabilan skup

- Važi

$$w = \sum_{i=1}^l y_i \alpha_i * (x_i)$$

$$\sum_{i=1}^l y_i \alpha_i = 0$$

- Važi $\lambda_i = 0$ osim $y_i * (w * x_i) + b = 1$
- Ako je $\lambda_i > 0$, x_i je potporni vektor

Metod podržavajućih (potpornih) vektora - linearno separabilan skup

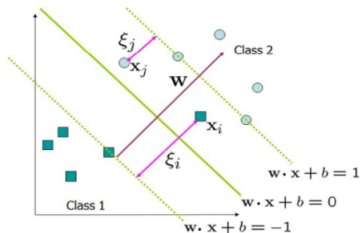
- dualni problem (transformacija Lagranžijana u funkciju od Lagranžijanovih množioca)

$$\max_{\lambda_i} L_D = \sum_{i=1}^N \lambda_i - \frac{1}{2} * \sum_{i=1}^N \sum_{j=1}^N \lambda_i * \lambda_j * y_i * y_j * x_i * x_j$$

- Klasifikacija instance z : $\text{sign}(\sum_{i=1}^N \lambda_i * y_i * x_i * z + b)$

Klasifikacija
Drvea odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Metod podržavajućih (potpornih) vektora - meka margina



Metod podržavajućih (potpornih) vektora - meka margina

- Ako instance skupa nisu linearno razdvojive, uvode se promenljive u ograničenja za svaku instancu kako bi se olabavila
- Ograničenja za instance
 - za $y_i = 1$ $w * x_i + b \geq 1 - \xi_i$
 - za $y_i = -1$ $w * x_i + b \leq -1 + \xi_i$

Metod podržavajućih (potpornih) vektora - meka margina

- Ciljna funkcija

$$\min \frac{||w||^2}{2} + C * \sum_{i=1}^N \xi_i^k$$

- C i k (1 ili 2) su konstante
- $\sum_{i=1}^N \xi_i^k$ - gubitak
- Male vrednosti C daju relaksirane margine, dok velike vrednosti C minimizuju greške na trening podacima i rezultuju uskim marginama
- dualni problem $L_D = \sum_{i=1}^N \lambda_i - \frac{1}{2} * \sum_{i,j} \lambda_i * \lambda_j * y_i * y_j * x_i * x_j$

Nelinearni metod podržavajućih (potpornih) vektora

- Za skupove koji nisu linearno razdvojivi, potrebno je odrediti funkciju Φ koja će skup transformisati u prostor sa linearno razdvojivim instancama
- Traži se $w * \Phi(x) + b = 0$
- Klasifikacija instance z : $sign(\sum_{i=1}^N \lambda_i * y_i * \Phi(x_i) * \Phi(z) + b)$
- Kernel trik: računanje $\Phi(x_i) * \Phi(z)$ u transformisanom prostoru koristeći originalne atribute

Nelinearni metod podržavajućih (potpornih) vektora

- Zahtev

$$\min \frac{\|w\|^2}{2} + C * \sum_{i=1}^N \xi_i^k$$

- Ograničenja za instance

$$y_i * (w * \Phi(x_i) + b) \geq 1 - \xi_i, i = 1, 2, \dots, N$$

$$\xi_i \geq 0$$

Nelinearni metod podržavajućih (potpornih) vektora

Neke kernel funkcije:

- Gausov radijalni bazni kernel (RBF):

$$K(X_i, X_j) = e^{-\frac{\|X_i - X_j\|^2}{2\sigma^2}}$$

- Polinomijalni kernel:

$$K(X_i, X_j) = (X_i \cdot X_j + c)^h$$

- Sigmoidni kernel:

$$K(X_i, X_j) = \tanh(\kappa X_i \cdot X_j - \delta)$$

Klasifikacija
Drveta odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Mere za ocenu modela

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

Klasifikacija - mere za ocenu modela

Najčešće korišćene mere za ocenu modela:

- *preciznost* = $\frac{\text{Broj slogova čija klasa je dobro predviđena modelom}}{\text{Ukupan broj slogova}}$ (eng. accuracy)
- stopa greške = $\frac{\text{Broj slogova čija klasa nije dobro predviđena modelom}}{\text{Ukupan broj slogova}}$ (eng. error rate)

Nisu dovoljne za skupove podataka sa neuravnoteženim klasama.

Klasifikacija - mere za ocenu modela

Za bolji uvid kako se model ponaša za svaku klasu koristi se matrica konfuzije.

Tabelle: Matrica konfuzije za 4 klase

		Dodeljena klasa			
		C_1	C_2	C_3	C_4
Stvarna klasa	C_1				
	C_2		x		
	C_3				
	C_4	y			

Klasifikacija - mere za ocenu modela

- Postoje mere koje daju bolji uvid kako se model ponaša za svaku klasu.
- Pri binarnoj klasifikaciji, retka klasa se označava kao pozitivna a većinska klasa kao negativna

Klasifikacija - mere za ocenu modela

- stopa stvarno pozitivnih ili osetljivost (eng. true positive rate, sensitivity) $TPR = \frac{TP}{TP+FN}$
- stopa stvarno negativnih ili specifičnost (eng. true negative rate, specificity) $TNR = \frac{TN}{FP+TN}$
- stopa lažno pozitivnih (eng. false positive rate) $FPR = \frac{FP}{FP+TN}$
- stopa lažno negativnih (eng. false negative rate)
 $FNR = \frac{FN}{TP+FN}$

Klasifikacija - mere za ocenu modela

- preciznost (eng. precision) $p = \frac{TP}{TP+FP}$

Za skup sa više klasa preciznost se računa za svaku klasu C_i sa

$$p = \frac{\text{broj instanci klase } C_i \text{ kojima model dodeljuje klasu } C_i}{\text{broj instanci kojima model dodeljuje klasu } C_i}$$

- odziv (eng. recall) $r = \frac{TP}{TP+FN}$

Za skup sa više klasa odziv se računa za svaku klasu C_i sa

$$r = \frac{\text{broj instanci klase } C_i \text{ kojima model dodeljuje klasu } C_i}{\text{broj instanci klase } C_i}$$

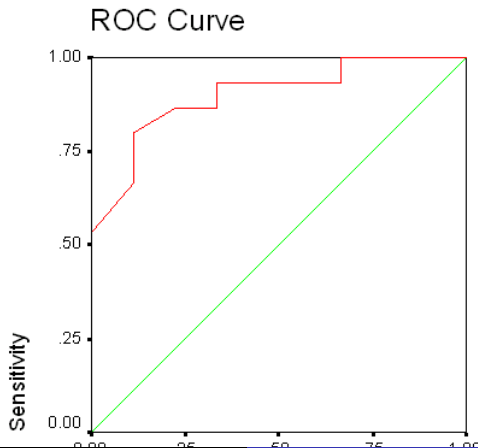
Klasifikacija - mere za ocenu modela

- F_1 uzima u obzir preciznost i odziv $F_1 = \frac{2rp}{r+p} = \frac{2}{\frac{1}{r} + \frac{1}{p}}$
 - harmonijska sredina preciznosti i odziva
 - bliža je manjoj vrednosti

Klasifikacija - mere za ocenu modela

- ROC kriva (receiver operating characteristic curve)
 - grafički prikaz kompromisa između *TPR* i *FPR*
 - x osa - *FPR* ili 1-specifičnost
 - y osa - *TPR* ili osetljivost

Klasifikacija - mere za ocenu modela



Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0)

Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0) - model svakoj instanci dodeljuje negativnu klasu

Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0) - model svakoj instanci dodeljuje negativnu klasu
 - (TPR=1 i FPR=1)

Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0) - model svakoj instanci dodeljuje negativnu klasu
 - (TPR=1 i FPR=1) - model svakoj instanci dodeljuje pozitivnu klasu

Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0) - model svakoj instanci dodeljuje negativnu klasu
 - (TPR=1 i FPR=1) - model svakoj instanci dodeljuje pozitivnu klasu
 - (TPR=1 i FPR=0)

Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0) - model svakoj instanci dodeljuje negativnu klasu
 - (TPR=1 i FPR=1) - model svakoj instanci dodeljuje pozitivnu klasu
 - (TPR=1 i FPR=0) - idealan model

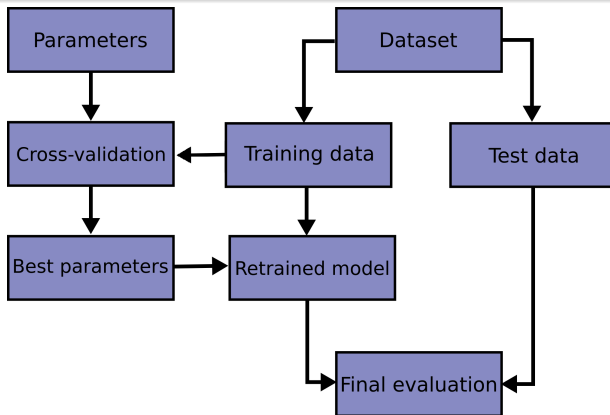
Klasifikacija - ROC kriva

- Interpretacija određenih tačaka
 - (TPR=0 i FPR=0) - model svakoj instanci dodeljuje negativnu klasu
 - (TPR=1 i FPR=1) - model svakoj instanci dodeljuje pozitivnu klasu
 - (TPR=1 i FPR=0) - idealan model
- AUC (eng. area under the ROC curve) - površina ispod ROC krive

Outline

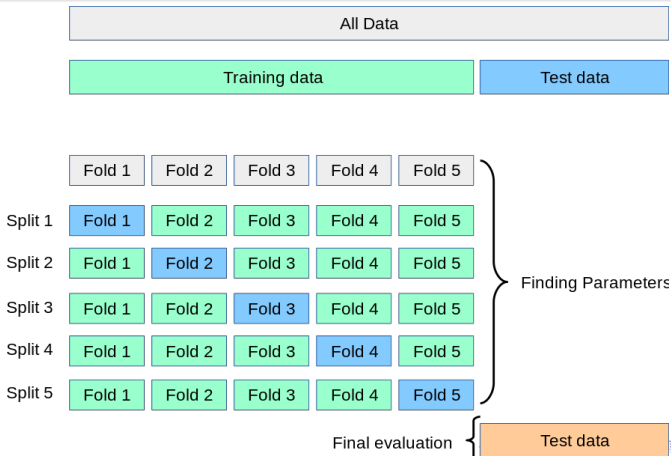
- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

Unakrsna validacija



Klasifikacija
Drвета odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansaml metode

Unakrsna validacija



Klasifikacija
Drveta odlučivanja
Naivni Bajesovski klasifikatori
K najbližih suseda
Veštačke neuronske mreže
SVM - Metod podržavajućih (potpornih) vektora
Klasifikacija - mere za ocenu modela
Unakrsna validacija
Ansambl metode

Outline

- 1 Klasifikacija
- 2 Drveta odlučivanja
- 3 Naivni Bajesovski klasifikatori
- 4 K najbližih suseda
- 5 Veštačke neuronske mreže
- 6 SVM - Metod podržavajućih (potpornih) vektora

Ansambl metode

- Različiti klasifikatori mogu davati različite predikcije za iste test instance.
- Ansambl metod je pristup kojim se povećava tačnost predikcije kombinovanjem rezultata više klasifikatora.
- Osnovna ideja je da se rezultati različitih klasifikatora se kombinuju u jednu predikciju.
- Klasifikatori mogu biti razvijeni korišćenjem različitih algoritama ili istog algoritma na različitim trening skupovima

Ansambl metode

Ulaz: Trening skup D , bazni algoritmi $A_1, \dots, A_r$, test instance $\mathcal{T}$

- 1 $j \leftarrow 1$
- 2 **ponavlja**
 - izaberi algoritam $Q_j \in \{A_1, \dots, A_r\}$
 - konstruši skup $f_j(D)$
 - nauči model M_j primenom Q_j nad $f_j(D)$
 - $j \leftarrow j + 1$
- 3 **do** ispunjenja uslova zaustavljanja
- 4 za svaku $T \in \mathcal{T}$ vrati klasu kombinovanjem predikcija svih modela M_j

Ansambl metode

- Različiti klasifikatori mogu davati različite predikcije za iste test instance.
- Ansambl metod je pristup kojim se povećava tačnost predikcije kombinovanjem rezultata više klasifikatora.
- Osnovna ideja je da se rezultati različitih klasifikatora se kombinuju u jednu predikciju.
- Klasifikatori mogu biti razvijeni korišćenjem različitih algoritama ili istog algoritma na različitim trening skupovima

Vrste ansambl metoda

- Ansambl metode usmerene na podatke - koristi se jedan algoritam učenja (npr. SVM ili stablo odlučivanja), a glavna varijacija se odnosi na način na koji se konstruiše izvedeni skup podataka, može se dobiti uzorkovanjem podataka, fokusiranjem na pogrešno klasifikovane delove trening skupa u prethodnim komponentama, manipulacijom atributima ili manipulacijom klasnim oznakama.
- Ansambl metode usmerene na modele) - koriste se različiti algoritmi, a skup za svaku komponentu ansambla je originalni skup.

Uzroci grešaka klasifikatora

- Pristrasnost (eng. bias) - svaki klasifikator pravi određene pretpostavke o prirodi granice odlučivanja između klasa.
- Varijansa - slučajne varijacije u izboru trening podataka dovode do različitih modela.

Različiti pristupi ansambla povećavaju tačnost smanjenjem uticaja pristrasnosti, varijanse ili kombinacije oba.

Bagging

- Pakovanje (Bagging skraćeno od bootstrapped aggregating) je pristup koji pokušava da smanji varijansu predikcije.
- Zasniva se na ideji da k nezavisnih i identično raspodeljenih predikcija smanjuje varijansu k puta.
- Instance se uzorkuju uniformno iz originalnog skupa sa vraćanjem (eng. bootstrap).
- Izvlači se uzorak iste veličine kao originalni skup.
- Taj uzorak može sadržati duplikate i tipično sadrži približno 63% početnog skupa jer se svaki uzorak bira sa verovatnoćom $1 - (1 - 1/N)^N$
- Ako je N dovoljno veliko verovatnoća konvergira ka $1 - 1/e \approx 0,632$

Bagging

- Nezavisno se izvlači ukupno k različitih bootstrap uzoraka, svaki veličine n , i klasifikator se trenira na svakom od njih.
- Za datu test instancu, ansambl vraća klasu za koju je glasala većina različitih klasifikatora.

Boosting

- Pojačavanje (eng. boosting)
- Svakom trening primeru dodeljuje se težina, i klasifikatori se treniraju uz korišćenje tih težina.
- Težine se iterativno menjaju na osnovu uspeha klasifikatora čime budući modeli zavise od prethodnih.
- Ideja je da se u narednim iteracijama fokus prebaci na pogrešno klasifikovane instance povećanjem njihove težine.
- Pretpostavka je da su greške posledica pristrasnosti.

Kofa modela

- Kofa modela (eng. Bucket of Models)
- Zasniva se na ideji da je teško unapred znati koji klasifikator najbolje radi na datom skupu podataka.
- Radi se selekcija modela: skup se deli na dva dela, A i B. Svaki algoritam se trenira na A, a zatim evaluira na B. Pobednik u tom „takmičenju“ se izabere i zatim ponovo trenira na celom skupu D.
- Može smanjiti i bias i varijansu

Slaganje modela

- Slaganje modela (eng. stacking)
- Opšti pristup sa dva nivoa klasifikacije.
- Trening podaci se dele na A i B.
 - 1 Na skupu A se trenira k klasifikatora.
 - 2 Na skupu B se izračunaju izlazi svih k klasifikatora. Formira se novi skup od k atributa gde je svaki atribut izlaz jednog klasifikatora, sa klasom. Zatim se trenira klasifikator nad tom novom reprezentacijom.
- Za test instancu, modeli prvog nivoa generišu k izlaza, a klasifikator drugog nivoa na osnovu njih daje finalnu predikciju.
- Može smanjiti i bias i varijansu

Obrada teksta na engleskom jeziku

- 1 eliminacija stop reči
- 2 svođenje reči na koren pomoću Porterovog stemera
<https://tartarus.org/martin/PorterStemmer/>
- 3 pravljenje term matrice