Modeli zasnovani na instancama

Mašinsko učenje 2020/21.

Matematički fakultet
Univerzitet u Beogradu

Modeli zasnovani na instancama

- Predstavljaju *neparametarski* pristup mašinskom učenju

Modeli zasnovani na instancama

- Predstavljaju *neparametarski* pristup mašinskom učenju
- Parametarski modeli pretpostavljaju postojanje konačnog skupa parametara čije vrednosti definišu model.

Modeli zasnovani na instancama

- Predstavljaju *neparametarski* pristup mašinskom učenju
- Parametarski modeli pretpostavljaju postojanje konačnog skupa parametara čije vrednosti definišu model.
- *Neparametarski modeli* su modeli koji se ne mogu opisati konačnim skupom parametara i čiji broj parametara može zavisiti od veličine skupa za obučavanje i stoga je neograničen

Modeli zasnovani na instancama

- Predstavljaju *neparametarski* pristup mašinskom učenju
- Parametarski modeli pretpostavljaju postojanje konačnog skupa parametara čije vrednosti definišu model.
- *Neparametarski modeli* su modeli koji se ne mogu opisati konačnim skupom parametara i čiji broj parametara može zavisiti od veličine skupa za obučavanje i stoga je neograničen
- Ovakvi metodi često moraju da čuvaju skup podataka za obučavanje kako bi davali predviđanja na novim instancama jer su modeli često i izraženi u terminima tih podataka, zbog čega i predviđanje na osnovu ovakvih metoda može biti računski zahtevno

Modeli zasnovani na instancama

- Predstavljaju *neparametarski* pristup mašinskom učenju
- Parametarski modeli pretpostavljaju postojanje konačnog skupa parametara čije vrednosti definišu model.
- *Neparametarski modeli* su modeli koji se ne mogu opisati konačnim skupom parametara i čiji broj parametara može zavisiti od veličine skupa za obučavanje i stoga je neograničen
- Ovakvi metodi često moraju da čuvaju skup podataka za obučavanje kako bi davali predviđanja na novim instancama jer su modeli često i izraženi u terminima tih podataka, zbog čega i predviđanje na osnovu ovakvih metoda može biti računski zahtevno
- Prednost je u tome što ne pretpostavljaju formu modela tako striktno kao parametarski modeli (npr. pretpostavka normalne raspodele), već ta forma može slobodnije da zavisi od podataka

Pregled

Osnove neparametarske gustine raspodele

- Ocena gustine raspodele predstavlja najteži problem mašinskog učenja.

Osnove neparametarske gustine raspodele

- Ocena gustine raspodele predstavlja najteži problem mašinskog učenja.
- Osnovni princip iza modelovanja gustine raspodele je da svaka od tačaka koje imamo na raspolaganju svedoči o tome da gustina raspodele u njenoj okolini treba da bude nezanemarljiva (jer se *desila*, a neverovatne stvari se *ne dešavaju*)

Osnove neparametarske gustine raspodele

- Ocena gustine raspodele predstavlja najteži problem mašinskog učenja.
- Osnovni princip iza modelovanja gustine raspodele je da svaka od tačaka koje imamo na raspolaganju svedoči o tome da gustina raspodele u njenoj okolini treba da bude nezanemarljiva (jer se *desila*, a neverovatne stvari se *ne dešavaju*)
- Svaka od tih tačaka, posebno kada se jave u velikom broju, daje nekakve doprinose toj oceni gustine i tu gde ih ima više gustina treba da bude veća a gde ih ima manje, gustina treba da bude manja

Osnove neparametarske gustine raspodele - ekstreman model

- Svaka tačka potpuno pouzdano svedoči o nenula verovatnoći svog dešavanja

Osnove neparametarske gustine raspodele - ekstreman model

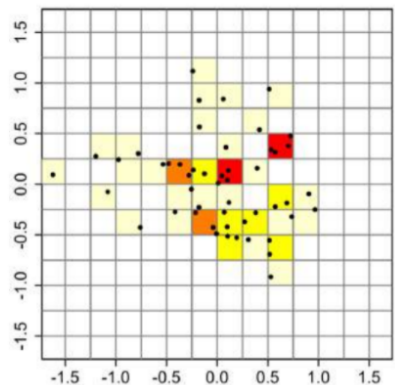
- Svaka tačka potpuno pouzdano svedoči o nenula verovatnoći svog dešavanja
- To znači da možemo da kreiramo model gustine raspodele koji raspodeljuje tu masu verovatnoće (koja se sumira na 1) *podjednako* na lokacijama gde su se tačke zaista našle i nula na svim ostalim

Osnove neparametarske gustine raspodele - ekstreman model

- Svaka tačka potpuno pouzdano svedoči o nenula verovatnoći svog dešavanja
- To znači da možemo da kreiramo model gustine raspodele koji raspodeljuje tu masu verovatnoće (koja se sumira na 1) *podjednako* na lokacijama gde su se tačke zaista našle i nula na svim ostalim
- preprilagođavanje

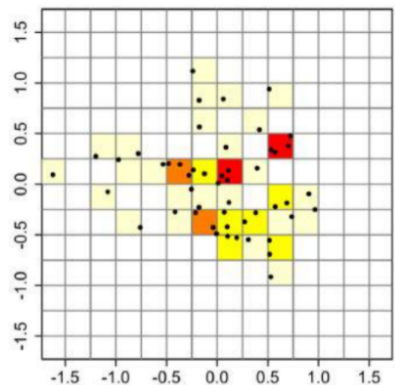
Osnove neparametarske gustine raspodele - realniji model

- U ekstremnom modelu smo pretpostavljali da pojavljivanje jedne tačke svedoči samo o pojavljivanju baš te tačke



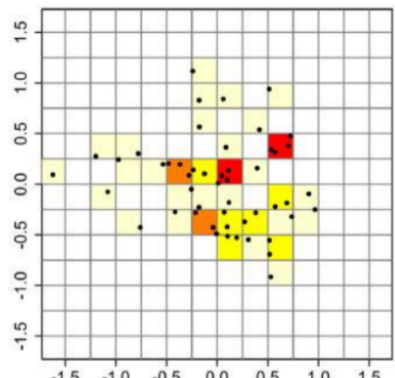
Osnove neparametarske gustine raspodele - realniji model

- U ekstremnom modelu smo pretpostavljali da pojavljivanje jedne tačke svedoči samo o pojavljivanju baš te tačke
- U realnijem modelu, pretpostavimo da pojavljivanje neke tačke svedoči o pojavljivanju i baš te tačke i još nekih tačaka u nekoj njenoj okolini

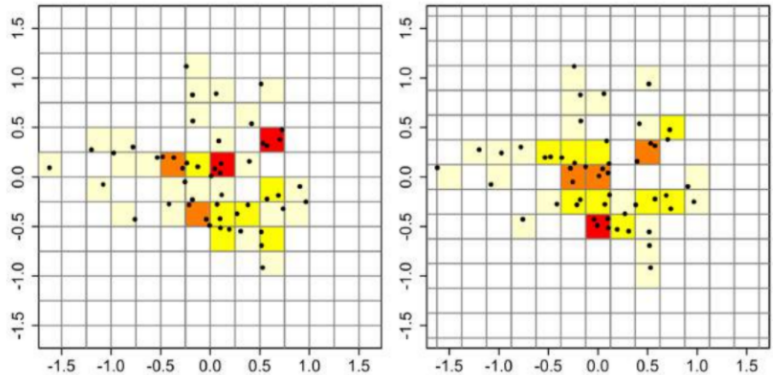


Osnove neparametarske gustine raspodele - realniji model

- U ekstremnom modelu smo pretpostavljali da pojavljivanje jedne tačke svedoči samo o pojavljivanju baš te tačke
- U realnijem modelu, pretpostavimo da pojavljivanje neke tačke svedoči o pojavljivanju i baš te tačke i još nekih tačaka u nekoj njenoj okolini
- Jedan model koji je konstruisan na ovim osnovama je *histogram*

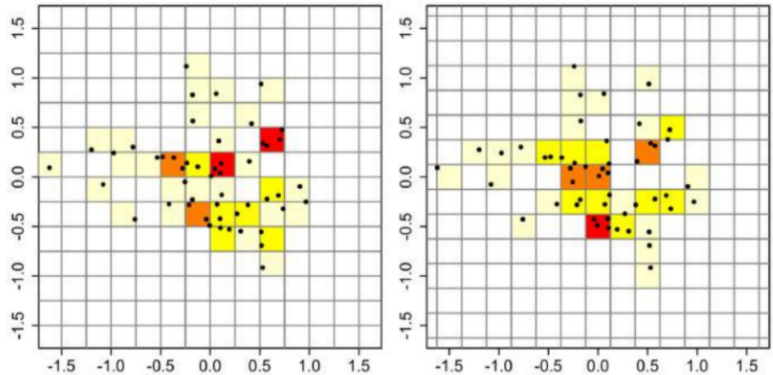


Histogram - mane



- tako ocenjena gustina raspodele zavisi od pozicija korpica, koje bi se pri konstrukciji histograma, mogle proizvoljno translirati

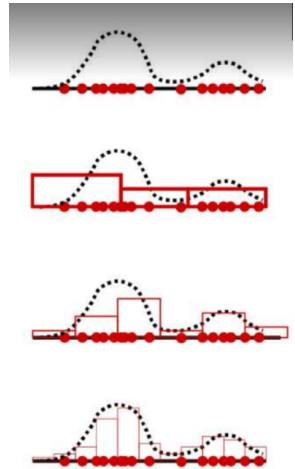
Histogram - mane



- tačke prekida histograma nisu posledica gustine podataka, već izbora lokacija korpica

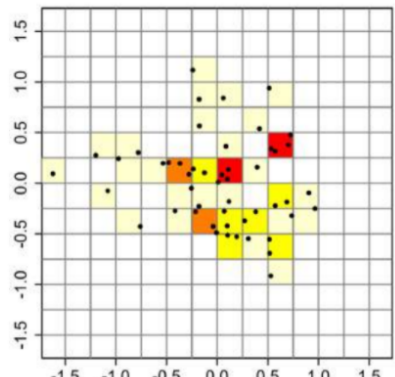
Histogram - mane

- oblik histograma drastično zavisi od širine, odnosno broja korpica



Histogram - mane

- zbog *prokletstva dimenzionalnosti*, većina korpica će biti prazna u slučaju prostora veće dimenzionalnosti



Alternativa

- Zbog navedenih mana, histogram se ne koristi za ocenjivanje gustine raspodele

Alternativa

- Zbog navedenih mana, histogram se ne koristi za ocenjivanje gustine raspodele
- Glavna mana je predefinisana geometrija korpica koja ne zavisi od podataka koje imamo na raspolaganju

Alternativa

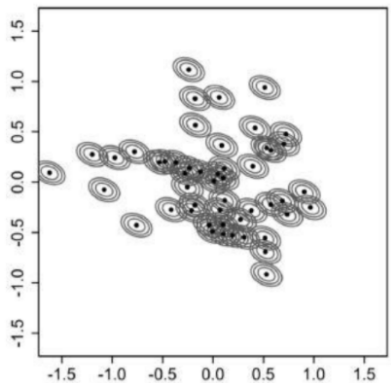
- Zbog navedenih mana, histogram se ne koristi za ocenjivanje gustine raspodele
- Glavna mana je predefinisana geometrija korpica koja ne zavisi od podataka koje imamo na raspolaganju
- Alternativno, krenimo od tačaka za koje određujemo gustinu raspodele

Alternativa

- Zbog navedenih mana, histogram se ne koristi za ocenjivanje gustine raspodele
- Glavna mana je predefinisana geometrija korpica koja ne zavisi od podataka koje imamo na raspolaganju
- Alternativno, krenimo od tačaka za koje određujemo gustinu raspodele
- Pokazaćemo kako se matematički izvode modeli koji na ovaj način određuju gustinu raspodele

Alternativa

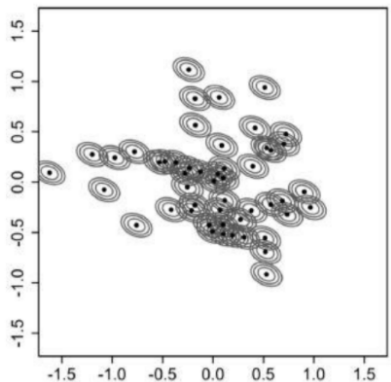
- Razmotrimo oblast $\mathcal{R}$ koja sadrži tačku x u čijoj okolini treba oceniti gustinu raspodele $p(x)$



Alternativa

- Razmotrimo oblast $\mathcal{R}$ koja sadrži tačku x u čijoj okolini treba oceniti gustinu raspodele $p(x)$
- Verovatnoća pridružena ovoj oblasti je

$$P = \int_{\mathcal{R}} p(x)dx$$

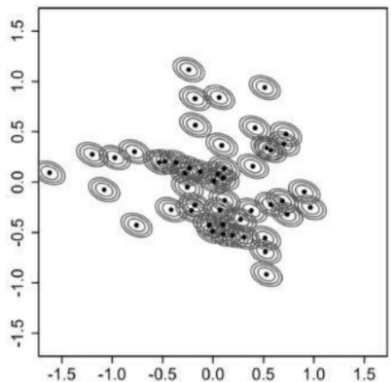


Alternativa

- Razmotrimo oblast $\mathcal{R}$ koja sadrži tačku x u čijoj okolini treba oceniti gustinu raspodele $p(x)$
- Verovatnoća pridružena ovoj oblasti je

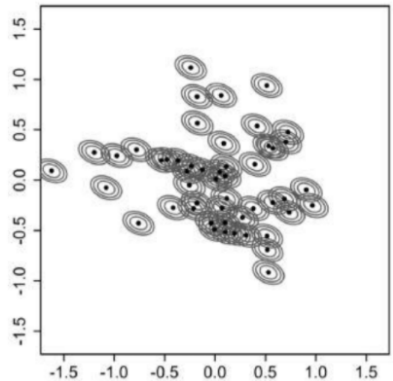
$$P = \int_{\mathcal{R}} p(x)dx$$

- Za potrebe ovog teorijskog razmatranja, pretpostavimo da nam je gustina raspodele $p(x)$ poznata



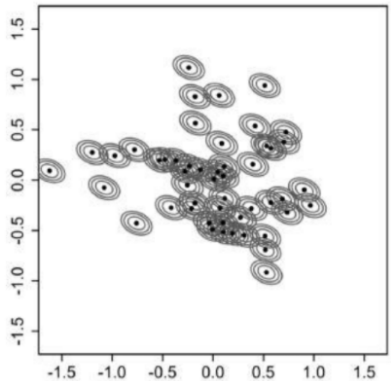
Alternativa

- Ako N opažanja (međusobno nezavisnih) dolazi iz raspodele $p(x)$, svako ima tu istu verovatnoću P pripadanja oblasti $\mathcal{R}$.



Alternativa

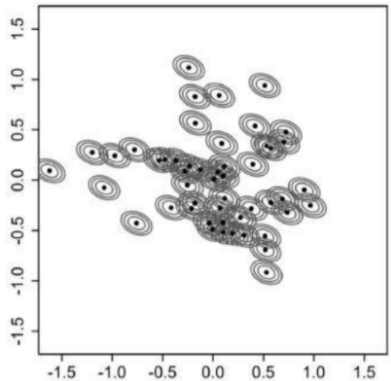
- Ako N opažanja (međusobno nezavisnih) dolazi iz raspodele $p(x)$, svako ima tu istu verovatnoću P pripadanja oblasti $\mathcal{R}$.
- Kako je raspodeljen broj tačaka k koje pripadaju ovoj oblasti od N koje su slučajno odabrane? Od N pokušaja imamo k uspeha sa verovatnoćom P?



Alternativa

- Ako N opažanja (međusobno nezavisnih) dolazi iz raspodele $p(x)$, svako ima tu istu verovatnoću P pripadanja oblasti $\mathcal{R}$.
- Kako je raspodeljen broj tačaka k koje pripadaju ovoj oblasti od N koje su slučajno odabrane? Od N pokušaja imamo k uspeha sa verovatnoćom P?
- Broj tačaka k je raspodeljen u skladu sa binomnom raspodelom:

$$p(k) = \binom{N}{k} P^k (1-P)^{N-k}$$

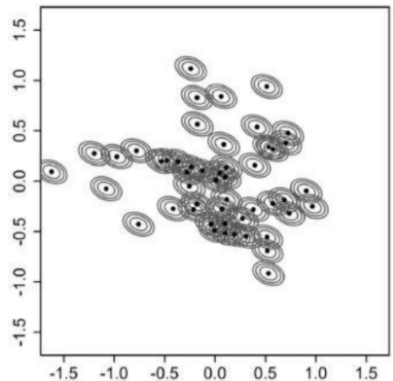


Alternativa

- Ako N opažanja (međusobno nezavisnih) dolazi iz raspodele $p(x)$, svako ima tu istu verovatnoću P pripadanja oblasti $\mathcal{R}$.
- Kako je raspodeljen broj tačaka k koje pripadaju ovoj oblasti od N koje su slučajno odabrane? Od N pokušaja imamo k uspeha sa verovatnoćom P?
- Broj tačaka k je raspodeljen u skladu sa binomnom raspodelom:

$$p(k) = \binom{N}{k} P^k (1-P)^{N-k}$$

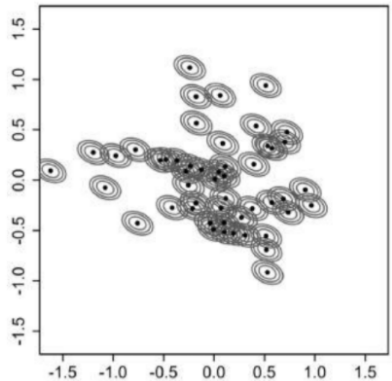
- Verovatnoća da je od N tačaka njih k u oblasti $\mathcal{R}$



Alternativa

- Broj tačaka k koje pripadaju ovoj oblasti je raspodeljen u skladu sa binomnom raspodelom:

$$p(k) = \binom{N}{k} P^k (1-P)^{N-k}$$



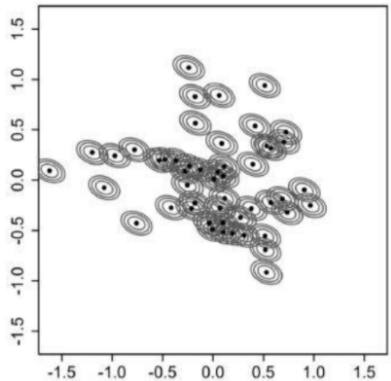
Alternativa

- Broj tačaka k koje pripadaju ovoj oblasti je raspodeljen u skladu sa binomnom raspodelom:

$$p(k) = \binom{N}{k} P^k (1-P)^{N-k}$$

- Očekivanje i varijansa za binomnu raspodelu:

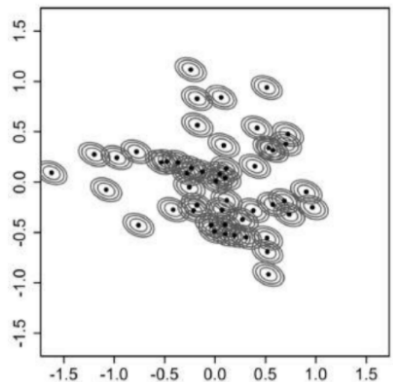
$$\mathbb{E}(k) = NP, \mathrm{var}[k] = NP(1-P)$$



Alternativa

- Očekivanje i varijansa za binomnu raspodelu za veličinu k:

$$\mathbb{E}(k) = NP, \text{var}[k] = NP(1-P)$$



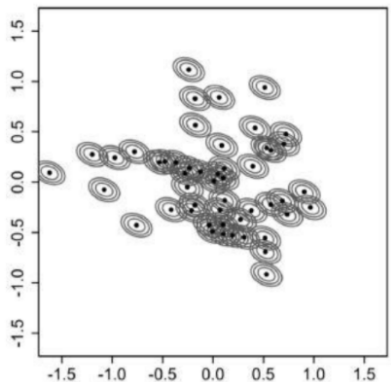
Alternativa

- Očekivanje i varijansa za binomnu raspodelu za veličinu k:

$$\mathbb{E}(k) = NP, \text{var}[k] = NP(1-P)$$

- Razmotrimo očekivanje i varijansu za udeo k u N, odnosno k/N:

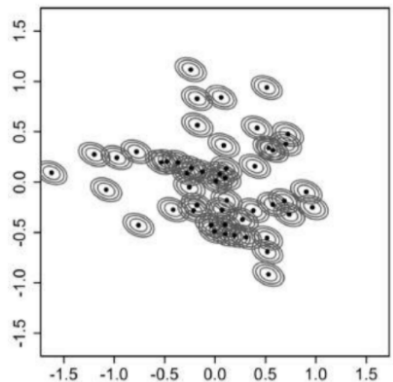
$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$



Alternativa

- Očekivanje i varijansa za udeo k/N:

$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

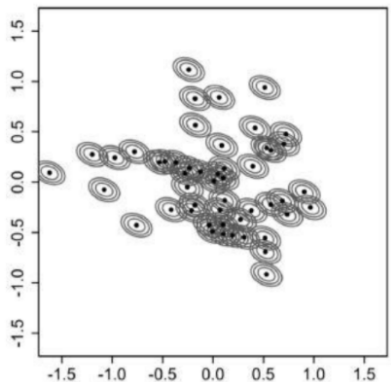


Alternativa

- Očekivanje i varijansa za udeo k/N:

  $$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

- Sa povećanjem broja N, varijansa veličine k/N se linearno smanjuje



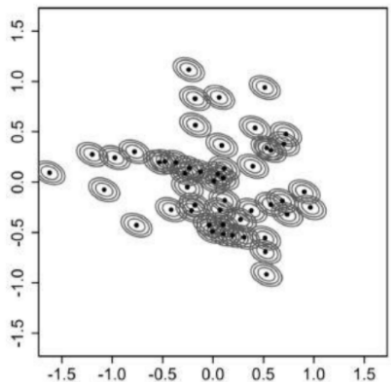
Alternativa

- Očekivanje i varijansa za udeo k/N:

$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

- Sa povećanjem broja N, varijansa veličine k/N se linearno smanjuje
- Možemo zaključiti da je k/N približno jednako P jer je varijansa za dovoljno veliko N vrlo mala

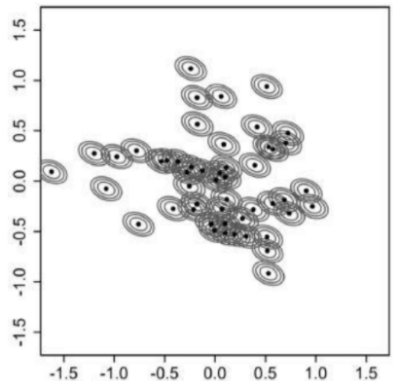
$$\frac{k}{N} \approx P$$



Alternativa

- Očekivanje i varijansa za udeo k/N:

$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

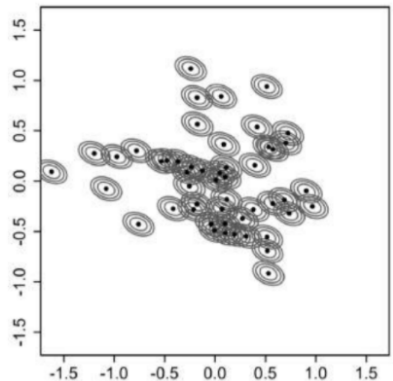


Alternativa

- Očekivanje i varijansa za udeo k/N:

$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

- Pretpostavimo da je $\mathcal{R}$ segment na realnoj pravoj

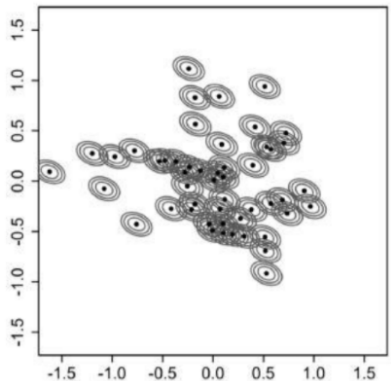


Alternativa

- Očekivanje i varijansa za udeo k/N:

$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

- Pretpostavimo da je $\mathcal{R}$ segment na realnoj pravoj
- Ako je zapremina oblasti $\mathcal{R}$ (dužina segmenta) dovoljno mala, onda se funkcija $p(x)$ ne menja mnogo u nekoj dovoljno maloj okolini tačke x i mozemo pretpostaviti da je konstanta

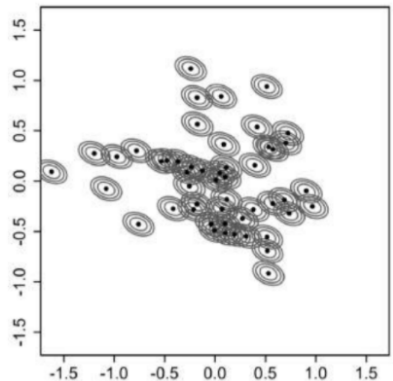


Alternativa

- Očekivanje i varijansa za udeo k/N:

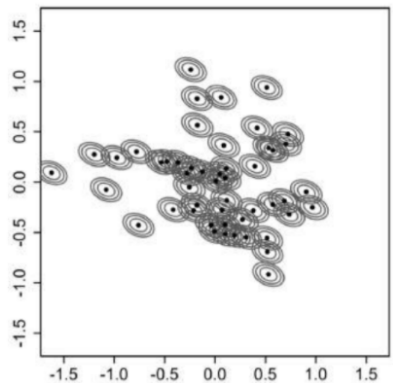
$$\mathbb{E}[k/N] = P, \text{var}[k/N] = P(1-P)/N$$

- Pretpostavimo da je $\mathcal{R}$ segment na realnoj pravoj
- Ako je zapremina oblasti $\mathcal{R}$ (dužina segmenta) dovoljno mala, onda se funkcija $p(x)$ ne menja mnogo u nekoj dovoljno maloj okolini tačke x i mozemo pretpostaviti da je konstanta
- u tom slucaju, verovatnoća se moze aproksimirati prostim pravougaonikom - zapremina regiona (duzina intervala) puta p(x): $P \approx p(x)V$



Alternativa

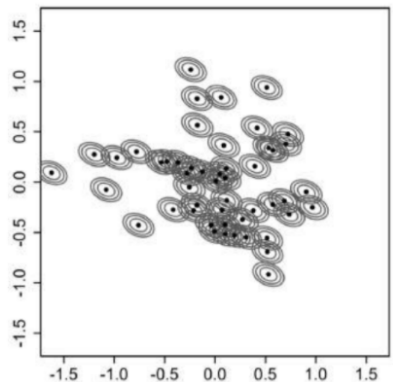
- Pokazali smo da važi $\frac{k}{N} \approx P$ i $P \approx p(x)V$



Alternativa

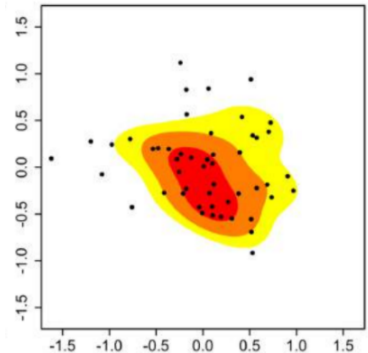
- Pokazali smo da važi $\frac{k}{N} \approx P$ i $P \approx p(x)V$
- Odatle, važi

$$p(x) \approx \frac{k}{NV}$$



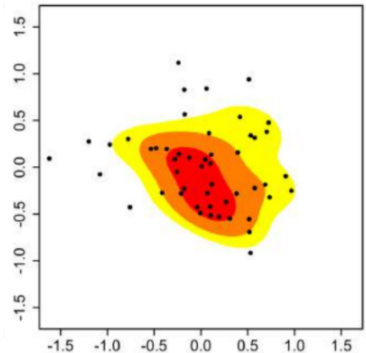
Alternativa

- Analizirajmo formulu $p(x) \approx \frac{k}{NV}$



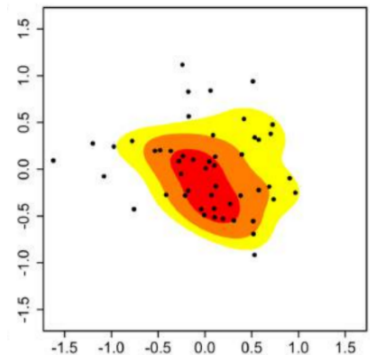
Alternativa

- Analizirajmo formulu $p(x) \approx \frac{k}{NV}$
- Između veličina k i V postoji očigledna zavisnost



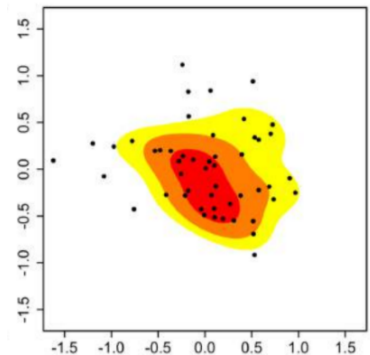
Alternativa

- Analizirajmo formulu $p(x) \approx \frac{k}{NV}$
- Između veličina k i V postoji očigledna zavisnost
- U većoj zapremini V, očekuje se više tačaka k



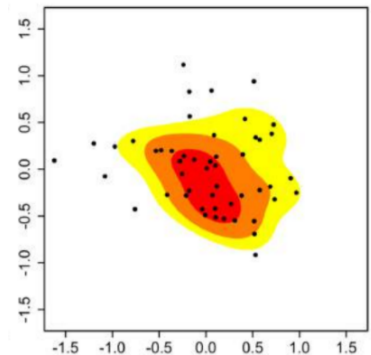
Alternativa

- Analizirajmo formulu $p(x) \approx \frac{k}{NV}$
- Između veličina k i V postoji očigledna zavisnost
- U većoj zapremini V, očekuje se više tačaka k
- Slično, kako bi bio dosegnut veliki broj tačaka k, potrebna je velika zapremina V



Alternativa

- Analizirajmo formulu $p(x) \approx \frac{k}{NV}$
- Između veličina k i V postoji očigledna zavisnost
- U većoj zapremini V, očekuje se više tačaka k
- Slično, kako bi bio dosegnut veliki broj tačaka k, potrebna je velika zapremina V
- Otud, pristupi oceni gustine raspodele mogu biti formulisani oslanjajući se na neku od ove dve veličine, prema čemu razlikujemo *pristupe zasnovane na kernelima* (fiksiramo V i brojimo koliko tačaka upada u nekakav oblik te zapremine) i *pristupe zasnovane na najbližim susedima* (fiksiramo k i posmatramo dobijenu zapreminu)



Pregled

Metodi zasnovani na kernelima

- Neformalno, *kernel* predstavlja funkciju sličnosti

Metodi zasnovani na kernelima

- Neformalno, *kernel* predstavlja funkciju sličnosti
- Što je vrednost kernela za neke dve instance veća, to se one mogu smatrati sličnijim

Metodi zasnovani na kernelima

- Neformalno, *kernel* predstavlja funkciju sličnosti
- Što je vrednost kernela za neke dve instance veća, to se one mogu smatrati sličnijim
- Što je manja, to se te dve instance mogu smatrati različitijim

Metodi zasnovani na kernelima

- Neformalno, *kernel* predstavlja funkciju sličnosti
- Što je vrednost kernela za neke dve instance veća, to se one mogu smatrati sličnijim
- Što je manja, to se te dve instance mogu smatrati različitijim
- Kerneli najčešće imaju određene parametre, kojima se finije podešava njihovo ponašanje

Metodi zasnovani na kernelima

- Neformalno, *kernel* predstavlja funkciju sličnosti
- Što je vrednost kernela za neke dve instance veća, to se one mogu smatrati sličnijim
- Što je manja, to se te dve instance mogu smatrati različitijim
- Kerneli najčešće imaju određene parametre, kojima se finije podešava njihovo ponašanje
- Ovi parametri su u vezi sa *zapreminom* okoline neke tačke u čijoj se okolini ocenjuje gustina raspodele

Lokalnost metoda zasnovanih na kernelima

- Ranije izlagani metodi izražavaju zavisnost ciljne promenljive od atributa pomoću koeficijenata koji kontrolišu dejstvo promene tog atributa na promenu ciljne promenljive

Lokalnost metoda zasnovanih na kernelima

- Ranije izlagani metodi izražavaju zavisnost ciljne promenljive od atributa pomoću koeficijenata koji kontrolišu dejstvo promene tog atributa na promenu ciljne promenljive
- Tipično, veće vrednosti atributa vode većim vrednostima ciljne promenljive, a male manjim, ili suprotno

Lokalnost metoda zasnovanih na kernelima

- Ranije izlagani metodi izražavaju zavisnost ciljne promenljive od atributa pomoću koeficijenata koji kontrolišu dejstvo promene tog atributa na promenu ciljne promenljive
- Tipično, veće vrednosti atributa vode većim vrednostima ciljne promenljive, a male manjim, ili suprotno
- Nasuprot tome, pristup zasnovan na kernelima, kao merama sličnosti, se zasniva na *bliskosti vrednosti atributa*

Lokalnost metoda zasnovanih na kernelima

- Ranije izlagani metodi izražavaju zavisnost ciljne promenljive od atributa pomoću koeficijenata koji kontrolišu dejstvo promene tog atributa na promenu ciljne promenljive
- Tipično, veće vrednosti atributa vode većim vrednostima ciljne promenljive, a male manjim, ili suprotno
- Nasuprot tome, pristup zasnovan na kernelima, kao merama sličnosti, se zasniva na *bliskosti vrednosti atributa*
- Ukoliko su vrednosti atributa nove instance bliske vrednostima atributa neke instance iz skupa za obučavanje, i vrednost ciljne promenljive nove instance treba da bude bliska ciljnoj vrednosti te instance, bez obzira da li su vrednosti atributa nove instance same za sebe visoke ili niske

Pregled

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$
  - $\int K(x)dx = 1$

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$
  - $\int K(x)dx = 1$
  - $K(-x) = K(x)$

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$
  - $\int K(x)dx = 1$
  - $K(-x) = K(x)$
  - K ne raste sa udaljavanjem od nule

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$
  - $\int K(x)dx = 1$
  - $K(-x) = K(x)$
  - K ne raste sa udaljavanjem od nule
- *Skalirani kernel* se definiše kao $K_\sigma = \frac{1}{\sigma^n} K\left(\frac{x}{\sigma}\right)$ za $\sigma > 0$, gde je n dimenzija vektora x.

Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$
  - $\int K(x)dx = 1$
  - $K(-x) = K(x)$
  - K ne raste sa udaljavanjem od nule
- *Skalirani kernel* se definiše kao $K_\sigma = \frac{1}{\sigma^n} K\left(\frac{x}{\sigma}\right)$ za $\sigma > 0$, gde je n dimenzija vektora x.
- Metaparametar σ se naziva *širinom* (eng. bandwidth) kernela

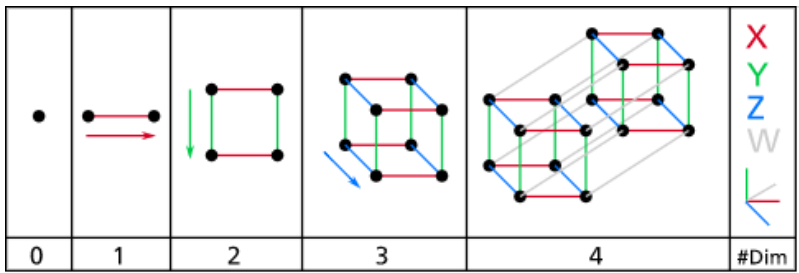
Ocena gustine raspodele zasnovane na kernelima

- *Glačajućim kernelom* (eng. *smoothing kernel*) nazivamo funkciju za koju važi:
  - $K(x) \geq 0$
  - $\int K(x)dx = 1$
  - $K(-x) = K(x)$
  - K ne raste sa udaljavanjem od nule
- *Skalirani kernel* se definiše kao $K_\sigma = \frac{1}{\sigma^n} K\left(\frac{x}{\sigma}\right)$ za $\sigma > 0$, gde je n dimenzija vektora x.
- Metaparametar σ se naziva *širinom* (eng. bandwidth) kernela
- Skalirani kernel zadovoljava sve uslove koje zadovoljava i polazni kernel.

Ocena gustine raspodele zasnovane na kernelima

- Definišimo karakterističnu funkciju jedinične hiperkocke sa centrom u koordinatnom početku

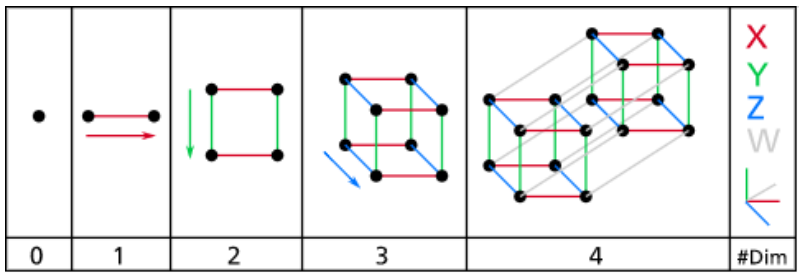
$$K(x) = \begin{cases} 1, & |x_i| \leq 1/2, \\ 0, & \text{inače} \end{cases} \qquad i = 1, \ldots, n$$



Ocena gustine raspodele zasnovane na kernelima

- Definišimo karakterističnu funkciju jedinične hiperkocke sa centrom u koordinatnom početku

$$K(x) = \begin{cases} 1, & |x_i| \leq 1/2, \\ 0, & \text{inače} \end{cases} \qquad i = 1, \ldots, n$$

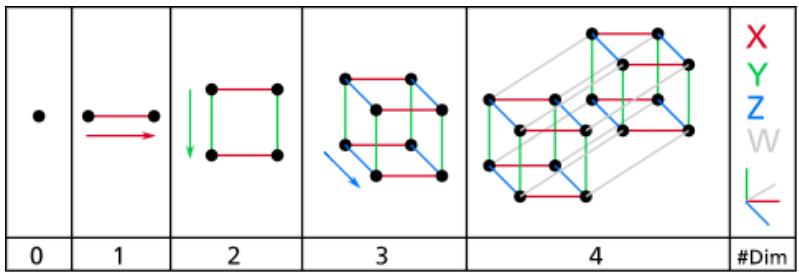


- Promenljiva x predstavlja n-dimenzioni vektor sa koordinatama $(x_1, \ldots, x_n)$

Ocena gustine raspodele zasnovane na kernelima

- Definišimo karakterističnu funkciju jedinične hiperkocke sa centrom u koordinatnom početku

$$K(x) = \begin{cases} 1, & |x_i| \leq 1/2, \\ 0, & \text{inače} \end{cases} \qquad i = 1, \ldots, n$$



- Promenljiva x predstavlja n-dimenzioni vektor sa koordinatama $(x_1, \ldots, x_n)$
- Funkcija K je primer kernela i naziva se *Parzenovim prozorom*.

Ocena gustine raspodele zasnovane na kernelima

- Karakterističnu funkciju jedinične hiperkocke sa centrom u koordinatnom početku:

$$K(x) = \begin{cases} 1, & |x_i| \leq 1/2, \\ 0, & \text{inače} \end{cases} \qquad i = 1, \ldots, n$$

Ocena gustine raspodele zasnovane na kernelima

- Karakterističnu funkciju jedinične hiperkocke sa centrom u koordinatnom početku:

$$K(x) = \begin{cases} 1, & |x_i| \leq 1/2, \qquad i = 1, \ldots, n \\ 0, & \text{inače} \end{cases}$$

- Za hiperkocku stranice σ sa centrom u tački x, karakteristična funkcija je $K\left(\frac{x - x_i}{\sigma}\right)$

Ocena gustine raspodele zasnovane na kernelima

- Karakterističnu funkciju jedinične hiperkocke sa centrom u koordinatnom početku:

$$K(x) = \begin{cases} 1, & |x_i| \leq 1/2, \\ 0, & \text{inače} \end{cases} \qquad i = 1, \ldots, n$$

- Za hiperkocku stranice σ sa centrom u tački x, karakteristična funkcija je $K\left(\frac{x-x_i}{\sigma}\right)$
- Izrazimo broj tačaka koje se nalaze u pomenutoj hiperkocki od ukupno N tačaka iz skupa za obučavanje

$$k = \sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right)$$

Ocena gustine raspodele zasnovane na kernelima

- Broj tačaka koje se nalaze u pomenutoj hiperkocki od ukupno N tačaka iz skupa za obučavanje

$$k = \sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right)$$

Ocena gustine raspodele zasnovane na kernelima

- Broj tačaka koje se nalaze u pomenutoj hiperkocki od ukupno N tačaka iz skupa za obučavanje

$$k = \sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right)$$

- Prethodno smo pokazali aproksimaciju $p(x) \approx \frac{k}{NV}$

Ocena gustine raspodele zasnovane na kernelima

- Broj tačaka koje se nalaze u pomenutoj hiperkocki od ukupno N tačaka iz skupa za obučavanje

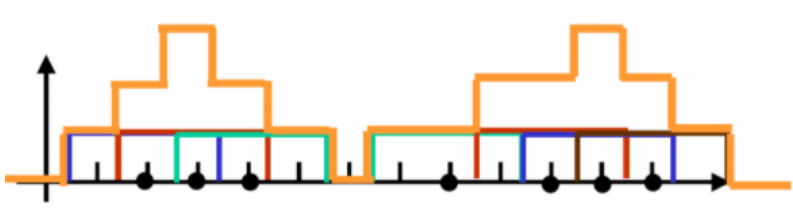
$$k = \sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right)$$

- Prethodno smo pokazali aproksimaciju $p(x) \approx \frac{k}{NV}$
- Na osnovu toga, važi

$$p(x) = \frac{1}{N}\frac{1}{\sigma^n}\sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\sigma^n}K\left(\frac{x - x_i}{\sigma}\right) = \frac{1}{N}\sum_{i=1}^{N} K_\sigma(x - x_i)$$

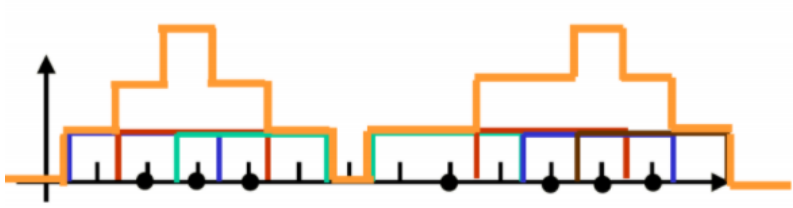
Ocena jednodimenzionalne gustine raspodele pomoću Parzenovih prozora širine 3

- Svaki prozor je određen jednom tačkom iz skupa za obučavanje i obuhvata prostor 1.5 levo i desno od te tačke (data je širina 3)



Ocena jednodimenzionalne gustine raspodele pomoću Parzenovih prozora širine 3

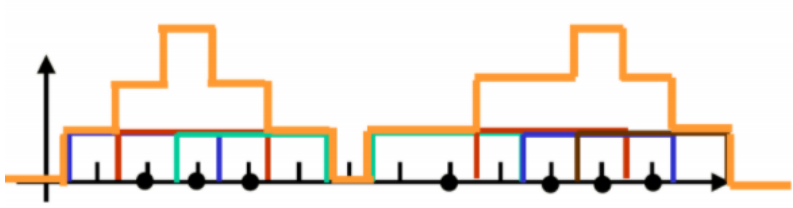
- Svaki prozor je određen jednom tačkom iz skupa za obučavanje i obuhvata prostor 1.5 levo i desno od te tačke (data je širina 3)



- Svaka tačka daje onoliki doprinos koliko iznosi broj tačaka koje se nalaze u Parzenovom prozoru koji ona definiše

Ocena jednodimenzionalne gustine raspodele pomoću Parzenovih prozora širine 3

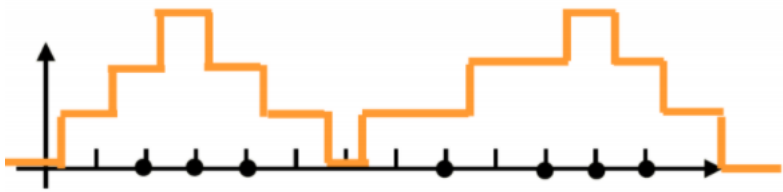
- Svaki prozor je određen jednom tačkom iz skupa za obučavanje i obuhvata prostor 1.5 levo i desno od te tačke (data je širina 3)



- Svaka tačka daje onoliki doprinos koliko iznosi broj tačaka koje se nalaze u Parzenovom prozoru koji ona definiše
- Ukupan broj tačaka je zbir svih vrednosti Parzenovih prozora koji se preklapaju

Ocena gustine raspodele zasnovane na kernelima

- Dobijena je prekidna ocena gustine koja je vrlo slična histogramu



Prednosti Parzenovih prozora u odnosu na histogram

- Kod histograma su korpice (podeoci) unapred fiksirani dok kod Parzenovih prozora zavise od podataka

Prednosti Parzenovih prozora u odnosu na histogram

- Kod histograma su korpice (podeoci) unapred fiksirani dok kod Parzenovih prozora zavise od podataka
- Zbog toga se kod histograma može dogoditi da tačka x pripada jednoj korpici i da je bliža većini tačaka iz susedne korpice nego većini tačaka iz svoje

Prednosti Parzenovih prozora u odnosu na histogram

- Kod histograma su korpice (podeoci) unapred fiksirani dok kod Parzenovih prozora zavise od podataka
- Zbog toga se kod histograma može dogoditi da tačka x pripada jednoj korpici i da je bliža većini tačaka iz susedne korpice nego većini tačaka iz svoje
- Time, kod histograma, tačke iz njene korpice, koje su joj daleko, određuju vrednost gustine raspodele u tački x, a tačke iz druge korpice kojima je bliža nemaju nikakav uticaj

Prednosti Parzenovih prozora u odnosu na histogram

- Kod histograma su korpice (podeoci) unapred fiksirani dok kod Parzenovih prozora zavise od podataka
- Zbog toga se kod histograma može dogoditi da tačka x pripada jednoj korpici i da je bliža većini tačaka iz susedne korpice nego većini tačaka iz svoje
- Time, kod histograma, tačke iz njene korpice, koje su joj daleko, određuju vrednost gustine raspodele u tački x, a tačke iz druge korpice kojima je bliža nemaju nikakav uticaj
- Kod Parzenovih prozora ovaj problem ne postoji

Kako se prevazilazi prekidnost kod Parzenovih prozora

- Može se koristiti neki neprekidni kernel, npr. Gausov

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

Kako se prevazilazi prekidnost kod Parzenovih prozora

- Može se koristiti neki neprekidni kernel, npr. Gausov

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

- za svaku tačku iz skupa za obučavanje, postavljamo centrirano oko nje po jedno Gausovo zvono širine σ

Kako se prevazilazi prekidnost kod Parzenovih prozora

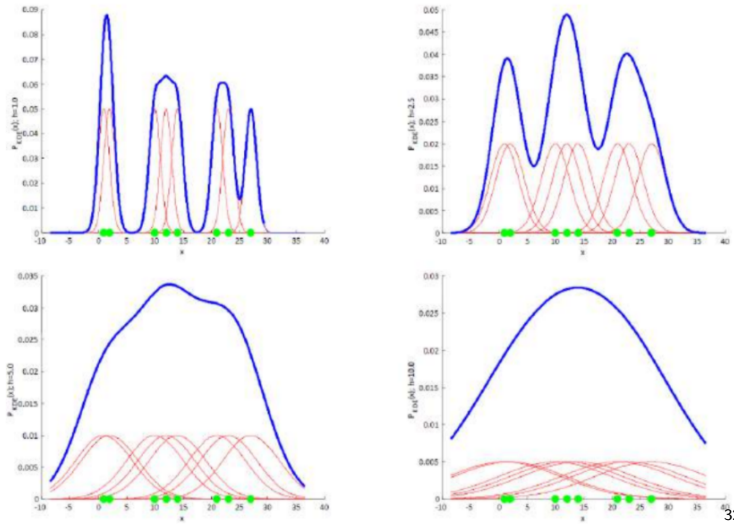
- Može se koristiti neki neprekidni kernel, npr. Gausov

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

- za svaku tačku iz skupa za obučavanje, postavljamo centrirano oko nje po jedno Gausovo zvono širine σ
- broj tačaka dobijamo na isti način - sabiranjem vrednosti kernela

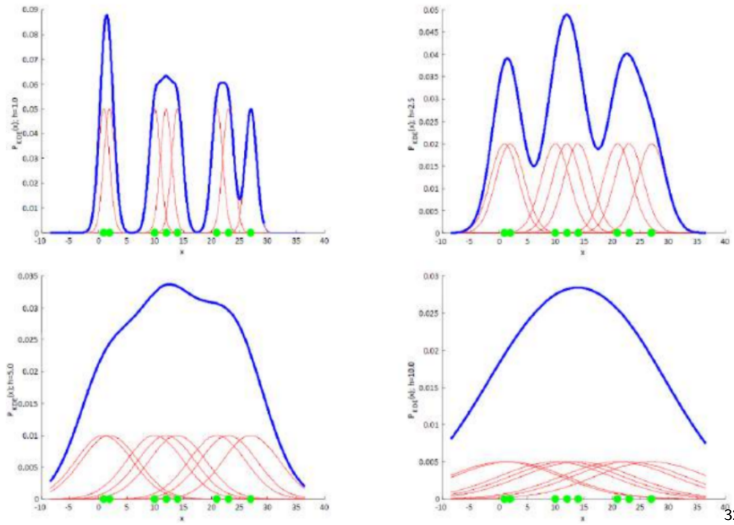
Ocena gustine jednodimenzionalne raspodele pomoću Gausovih kernela različitih širina

- Različite širine zvona daju drastično različite ocene gustine raspodele



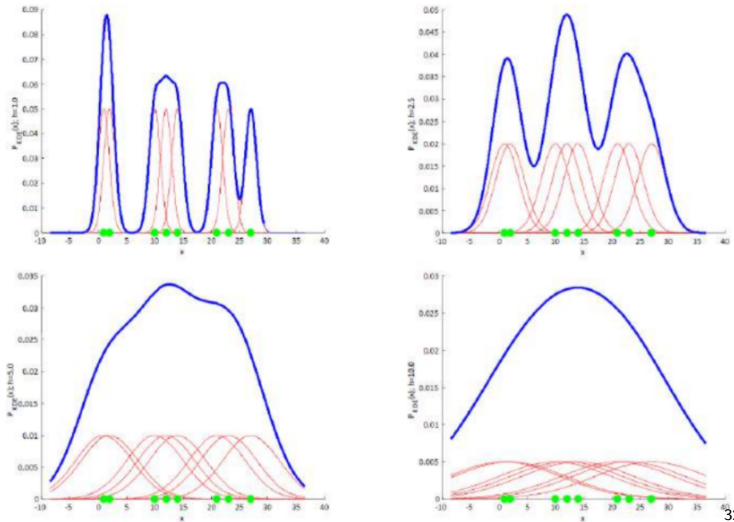
Ocena gustine jednodimenzionalne raspodele pomoću Gausovih kernela različitih širina

- Različite širine zvona daju drastično različite ocene gustine raspodele
- Uska zvona vode preprilagođavanju a široka potprilagođavanju



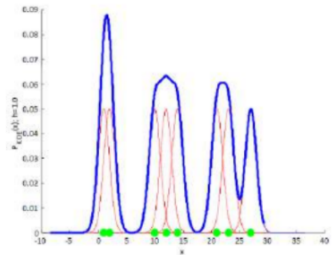
Ocena gustine jednodimenzionalne raspodele pomoću Gausovih kernela različitih širina

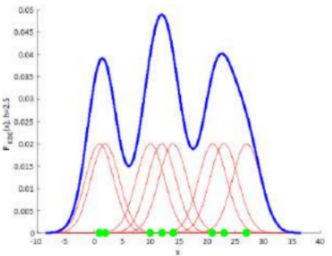
- Različite širine zvona daju drastično različite ocene gustine raspodele
- Uska zvona vode preprilagođavanju a široka potprilagođavanju
- Koje σ je najbolje?

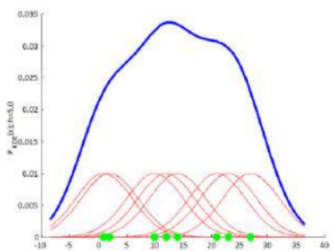


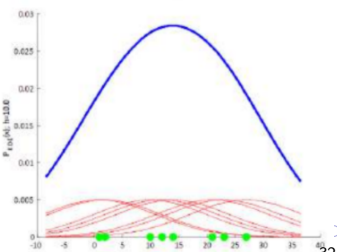
Ocena gustine jednodimenzionalne raspodele pomoću Gausovih kernela različitih širina

- Različite širine zvona daju drastično različite ocene gustine raspodele
- Uska zvona vode preprilagođavanju a široka potprilagođavanju
- Koje σ je najbolje?
- σ je metaparametar, o određivanju vrednosti metaparametara kasnije



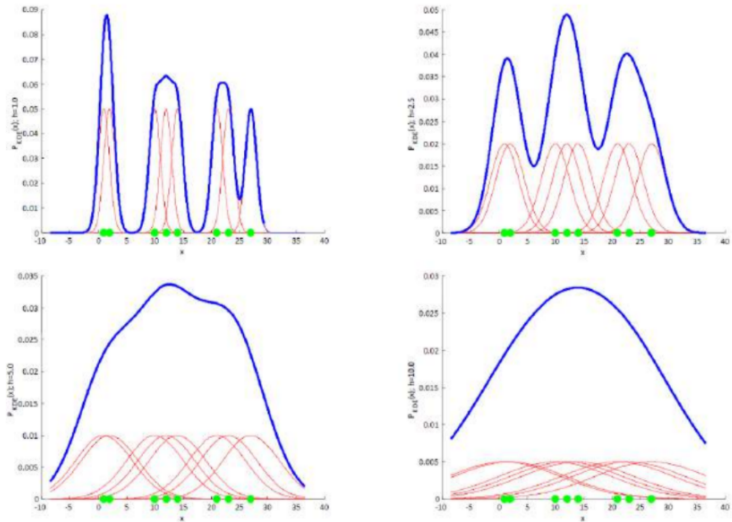






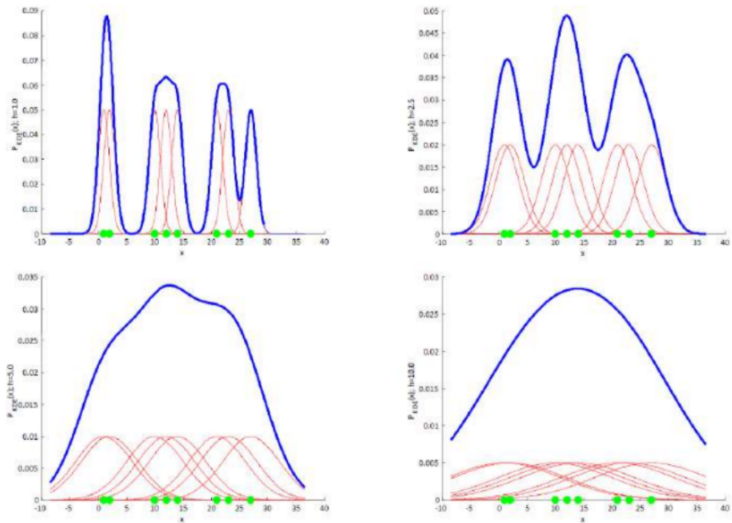
Mana ocene gustine raspodele zasnovane na kernelima

- Širina kernela σ je uvek ista i nezavisna od tačke x



Mana ocene gustine raspodele zasnovane na kernelima

- Širina kernela σ je uvek ista i nezavisna od tačke x
- Tako, jedna vrednost σ u oblastima visoke gustine može rezultovati preširokim kernelom, dok bi u oblastima u kojima je gustina značajno niža taj isti krenel mogao biti premale širine i voditi preprilagođavanju



Pregled

Nadaraja-Votson metod zasnovan na kernelima

- Videli smo kako možemo pomoću kernela da ocenimo gustinu raspodele

Nadaraja-Votson metod zasnovan na kernelima

- Videli smo kako možemo pomoću kernela da ocenimo gustinu raspodele
- Hajde da vidimo kako to možemo da iskoristimo u rešavanju konkretnih problema, npr. regresije

Nadaraja-Votson metod zasnovan na kernelima

- Videli smo kako možemo pomoću kernela da ocenimo gustinu raspodele
- Hajde da vidimo kako to možemo da iskoristimo u rešavanju konkretnih problema, npr. regresije
- Setimo se da je regresiona funkcija, čija aproksimacija se traži u problemu regresije, definisana kao uslovno očekivanje ciljne promenljive pri datim vrednostima atributa:

$$r(x) = \mathbb{E}[y|x] = \int yp(y|x)dy = \int y\frac{p(x,y)}{p(x)}dy$$

Nadaraja-Votson metod zasnovan na kernelima

- Regresiona funkcija

$$r(x) = \mathbb{E}[y|x] = \int yp(y|x)dy = \int y\frac{p(x,y)}{p(x)}dy$$

Nadaraja-Votson metod zasnovan na kernelima

- Regresiona funkcija

$$r(x) = \mathbb{E}[y|x] = \int yp(y|x)dy = \int y\frac{p(x,y)}{p(x)}dy$$

- Jedan način da se izvede ocena ove funkcije je da se u datom integralu upotrebe ocene gustina raspodela koje figurišu u njemu

Nadaraja-Votson metod zasnovan na kernelima

- Regresiona funkcija

$$r(x) = \mathbb{E}[y|x] = \int yp(y|x)dy = \int y\frac{p(x,y)}{p(x)}dy$$

- Jedan način da se izvede ocena ove funkcije je da se u datom integralu upotrebe ocene gustina raspodela koje figurišu u njemu
- Ukoliko su definisani neki glačajući kerneli na prostoru atributa $\mathcal{X}$ i na skupu vrednosti ciljne promenljive $\mathcal{Y}$, njihov proizvod predstavlja kernel na prostoru $\mathcal{X} \times \mathcal{Y}$

Nadaraja-Votson metod zasnovan na kernelima

- Regresiona funkcija

$$r(x) = \mathbb{E}[y|x] = \int yp(y|x)dy = \int y\frac{p(x,y)}{p(x)}dy$$

Nadaraja-Votson metod zasnovan na kernelima

- Regresiona funkcija

$$r(x) = \mathbb{E}[y|x] = \int yp(y|x)dy = \int y\frac{p(x,y)}{p(x)}dy$$

- Regresiona funkcija se stoga može aproksimirati sledećim modelom:

$$f_\sigma(x) = \int y\frac{\sum_{i=1}^{N} K_\sigma(x-x_i)K_\sigma(y-y_i)}{\sum_{j=1}^{N} K_\sigma(x-x_j)}dy =$$

$$\frac{\sum_{i=1}^{N} K_\sigma(x-x_i)\int yK_\sigma(y-y_i)dy}{\sum_{j=1}^{N} K_\sigma(x-x_j)} =$$

$$\frac{\sum_{i=1}^{N} K_\sigma(x-x_i)y_i}{\sum_{j=1}^{N} K_\sigma(x-x_j)}$$

Nadaraja-Votson metod zasnovan na kernelima

- Ovaj model se naziva model regresije Nadaraja-Votson

$$f_\sigma(x) = \frac{\sum_{i=1}^{N} K_\sigma(x - x_i) y_i}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

Nadaraja-Votson metod zasnovan na kernelima

- Ovaj model se naziva model regresije Nadaraja-Votson

$$f_\sigma(x) = \frac{\sum_{i=1}^{N} K_\sigma(x - x_i) y_i}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

- Razmotrimo njegov smisao

Nadaraja-Votson metod zasnovan na kernelima

- Ovaj model se naziva model regresije Nadaraja-Votson

$$f_\sigma(x) = \frac{\sum_{i=1}^{N} K_\sigma(x - x_i) y_i}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

- Razmotrimo njegov smisao
- Svakoj vrednosti ciljne promeljive y_i pridružena je težina

$$\frac{K_\sigma(x - x_i)}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

Nadaraja-Votson metod zasnovan na kernelima

- Ovaj model se naziva model regresije Nadaraja-Votson

$$f_\sigma(x) = \frac{\sum_{i=1}^{N} K_\sigma(x - x_i) y_i}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

- Razmotrimo njegov smisao
- Svakoj vrednosti ciljne promeljive y_i pridružena je težina

$$\frac{K_\sigma(x - x_i)}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

- Primetimo da je ova težina proporcionalna sličnosti tačke x sa tačkom x_i

Nadaraja-Votson metod zasnovan na kernelima

- Ovaj model se naziva model regresije Nadaraja-Votson

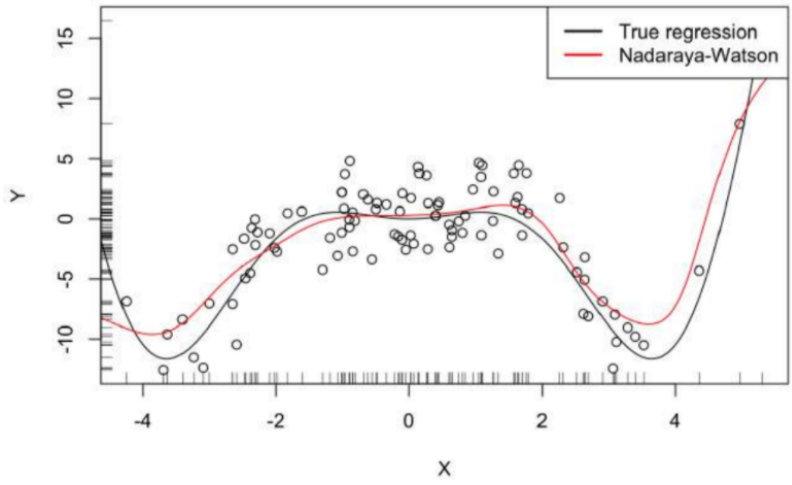
$$f_\sigma(x) = \frac{\sum_{i=1}^{N} K_\sigma(x - x_i) y_i}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

- Razmotrimo njegov smisao
- Svakoj vrednosti ciljne promeljive y_i pridružena je težina

$$\frac{K_\sigma(x - x_i)}{\sum_{j=1}^{N} K_\sigma(x - x_j)}$$

- Primetimo da je ova težina proporcionalna sličnosti tačke x sa tačkom x_i
- Kako se ove težine sabiraju na 1, zaključujemo da je dobijeno rešenje težinski prosek vrednosti y_i, pri čemu se pridaje veća težina sličnijim instancama

Primer Nadaraja-Votson modela sa Gausovim kernelom



Mana Nadaraja-Votson metoda

- Koja je minimalna a koja maksimalna vrednost ocene koja se dobija ovim modelom?

Mana Nadaraja-Votson metoda

- Koja je minimalna a koja maksimalna vrednost ocene koja se dobija ovim modelom?
- Najmanje i najveće viđeno y_i

Mana Nadaraja-Votson metoda

- Koja je minimalna a koja maksimalna vrednost ocene koja se dobija ovim modelom?
- Najmanje i najveće viđeno y_i
- Nadaraja-Votson model ne može da ekstrapolira

Nadaraja-Votson metod zasnovan na kernelima

- Uporedimo model Nadaraja-Votson sa polinomijalnim modelom linearne regresije

Nadaraja-Votson metod zasnovan na kernelima

- Uporedimo model Nadaraja-Votson sa polinomijalnim modelom linearne regresije
- Kod polinomijalnog modela linearne regresije (parametarski model), morali smo da pretpostavimo formu modela (linearan u odnosu na koeficijente w_i) i formu baznih funkcija (polinomi)

Nadaraja-Votson metod zasnovan na kernelima

- Uporedimo model Nadaraja-Votson sa polinomijalnim modelom linearne regresije
- Kod polinomijalnog modela linearne regresije (parametarski model), morali smo da pretpostavimo formu modela (linearan u odnosu na koeficijente w_i) i formu baznih funkcija (polinomi)
- Kod Nadaraja-Votson modela (neparametarski model), takva pretpostavka se ne traži

Nadaraja-Votson metod zasnovan na kernelima

- Uporedimo model Nadaraja-Votson sa polinomijalnim modelom linearne regresije
- Kod polinomijalnog modela linearne regresije (parametarski model), morali smo da pretpostavimo formu modela (linearan u odnosu na koeficijente w_i) i formu baznih funkcija (polinomi)
- Kod Nadaraja-Votson modela (neparametarski model), takva pretpostavka se ne traži
- Ipak, s druge strane, traži izbor kernela i izražava rešenje u terminima celog skupa za obučavanje, koji je stoga potrebno čuvati i koji se ceo koristi prilikom predviđanja

Pregled

Mercerov kernel

- Neka je $\mathcal{X}$ neprazan skup i neka je data simetrična funkcija $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$

Mercerov kernel

- Neka je $\mathcal{X}$ neprazan skup i neka je data simetrična funkcija $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
- Ako je za svako $n \in \mathbb{N}$ i svako $x_1, \ldots, x_n \in \mathcal{X}$ matrica dimenzija $n \times n$ sa elementima $k(x_i, x_j)$ pozitivno semidefinitna, funkcija k je *pozitivno semidefinitan kernel* ili *Mercerov kernel*

Mercerov kernel

- Neka je $\mathcal{X}$ neprazan skup i neka je data simetrična funkcija $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
- Ako je za svako $n \in \mathbb{N}$ i svako $x_1, \ldots, x_n \in \mathcal{X}$ matrica dimenzija $n \times n$ sa elementima $k(x_i, x_j)$ pozitivno semidefinitna, funkcija k je *pozitivno semidefinitan kernel* ili *Mercerov kernel*
- U nastavku se pod izrazom kernel podrazumeva Mercerov kernel.

Mercerov kernel

- Za svaki kernel k postoji preslikavanje Φ_k iz $\mathcal{X}$ u neki vektorski prostor $\mathcal{H}_k$ sa skalarnim proizvodom, tako da važi

$$k(x, y) = \Phi_k(x) \cdot \Phi_k(y)$$

Mercerov kernel

- Za svaki kernel k postoji preslikavanje Φ_k iz $\mathcal{X}$ u neki vektorski prostor $\mathcal{H}_k$ sa skalarnim proizvodom, tako da važi

$$k(x, y) = \Phi_k(x) \cdot \Phi_k(y)$$

- Svaki kernel se može posmatrati kao skalarni proizvod u nekom vektorskom prostoru

Mercerov kernel

- Za svaki kernel k postoji preslikavanje Φ_k iz $\mathcal{X}$ u neki vektorski prostor $\mathcal{H}_k$ sa skalarnim proizvodom, tako da važi

$$k(x, y) = \Phi_k(x) \cdot \Phi_k(y)$$

- Svaki kernel se može posmatrati kao skalarni proizvod u nekom vektorskom prostoru
- Kernel se može smatrati merom sličnosti nad elementima skupa $\mathcal{X}$

Mercenov kernel

- Za svaki kernel k postoji preslikavanje Φ_k iz $\mathcal{X}$ u neki vektorski prostor $\mathcal{H}_k$ sa skalarnim proizvodom, tako da važi

$$k(x, y) = \Phi_k(x) \cdot \Phi_k(y)$$

Mercenov kernel

- Za svaki kernel k postoji preslikavanje Φ_k iz $\mathcal{X}$ u neki vektorski prostor $\mathcal{H}_k$ sa skalarnim proizvodom, tako da važi

$$k(x, y) = \Phi_k(x) \cdot \Phi_k(y)$$

- Za skup $\mathcal{X}$ nije pretpostavljena nikakva struktura, ne mora biti čak ni vektorski prostor

Mercenov kernel

- Za svaki kernel k postoji preslikavanje Φ_k iz $\mathcal{X}$ u neki vektorski prostor $\mathcal{H}_k$ sa skalarnim proizvodom, tako da važi

$$k(x, y) = \Phi_k(x) \cdot \Phi_k(y)$$

- Za skup $\mathcal{X}$ nije pretpostavljena nikakva struktura, ne mora biti čak ni vektorski prostor
- To znači za proizvoljne objekte nad kojima možemo definisati kernele, možemo imati značajan deo tehničkih pogodnosti koje pruža skalarni proizvod

Mercerov kernel

- Kerneli imaju određena poželjna svojstva, kao da je linearna kombinacija kernela također kernel, da je proizvod kernela također kernel i slično

Mercerov kernel

- Kerneli imaju određena poželjna svojstva, kao da je linearna kombinacija kernela takođe kernel, da je proizvod kernela takođe kernel i slično
- Ova svojstva se mogu upotrebiti za konstrukciju novih kernela od već poznatih.

Kernelizovani metod potpornih vektora za klasifikaciju

- Problem: hiperravan ne mora predstavljati adekvatnu granicu među klasama.

Kernelizovani metod potpornih vektora za klasifikaciju

- Problem: hiperravan ne mora predstavljati adekvatnu granicu među klasama.
- Oblici granica u praksi mogu biti proizvoljni (na primer, moglo bi biti potrebno da granica bude kružna)

Kernelizovani metod potpornih vektora za klasifikaciju

- Problem: hiperravan ne mora predstavljati adekvatnu granicu među klasama.
- Oblici granica u praksi mogu biti proizvoljni (na primer, moglo bi biti potrebno da granica bude kružna)
- Formulacija sa mekim pojasom ne rešava ovaj problem, pošto meki pojas pretpostavlja da je hiperravan ugrubo dobar oblik granice, samo što podaci nekada završe sa pogrešne strane

Kernelizovani metod potpornih vektora za klasifikaciju

- Metod potpornih vektora ima sledeću formu modela:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i (x_i \cdot x) + w_0$$

pri čemu važi

$$w_0 = -\frac{\min_{(x,1)\in\mathcal{D}} w \cdot x + \max_{(x,-1)\in\mathcal{D}} w \cdot x}{2}$$

Kernelizovani metod potpornih vektora za klasifikaciju

- Metod potpornih vektora ima sledeću formu modela:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i (x_i \cdot x) + w_0$$

pri čemu važi

$$w_0 = -\frac{\min_{(x,1)\in\mathcal{D}} w \cdot x + \max_{(x,-1)\in\mathcal{D}} w \cdot x}{2}$$

- Rešenje se oslanja na određene instance iz skupa za obučavanje (određene x_i) tako što računa skalarni proizvod instance koju treba klasifikovati sa njima ($x_i \cdot x$)

Kernelizovani metod potpornih vektora za klasifikaciju

- Metod potpornih vektora ima sledeću formu modela:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i (x_i \cdot x) + w_0$$

pri čemu važi

$$w_0 = -\frac{\min_{(x,1)\in\mathcal{D}} w \cdot x + \max_{(x,-1)\in\mathcal{D}} w \cdot x}{2}$$

- Rešenje se oslanja na određene instance iz skupa za obučavanje (određene x_i) tako što računa skalarni proizvod instance koju treba klasifikovati sa njima ($x_i \cdot x$)
- Skalarni proizvod je jedna vrsta kernela (važi $k(x, x') = \Phi(x) \cdot \Phi(x')$) i kerneli se mogu posmatrati kao njegova uopštenja

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

- Šta ako bi u tom prostoru podaci za obučavanje mogli biti razdvojeni pomoću hiperravni, iako u polaznom prostoru taj oblik nije odgovarajući?

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

- Šta ako bi u tom prostoru podaci za obučavanje mogli biti razdvojeni pomoću hiperravni, iako u polaznom prostoru taj oblik nije odgovarajući?
- Kako je na podacima moguće vršiti različite vrste pretprocesiranja, ukoliko bismo znali preslikavanje Φ mogli bismo ga primeniti na date podatke i dobiti novi skup podataka nad kojim bismo mogli primeniti standardni metod potpornih vektora.

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

- Preslikavanje Φ nije poznato

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

- Preslikavanje Φ nije poznato
- Čak i da znamo Φ, moglo bi da se desi da ovo preslikavanje slika podatke u neki beskonačno dimenzionalni prostor

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

- Preslikavanje Φ nije poznato
- Čak i da znamo Φ, moglo bi da se desi da ovo preslikavanje slika podatke u neki beskonačno dimenzionalni prostor
- Bolja strategija: zamena skalarnog proizvoda kernelom u svim formulama, što je sa matematičke tačke gledišta isto

Kernelizovani metod potpornih vektora za klasifikaciju

- Za svaki kernel postoji preslikavanje Φ u neki vektorski prostor u kojem važi

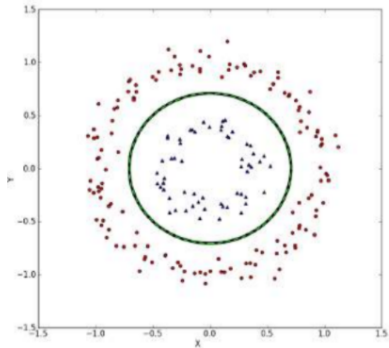
$$k(x, x') = \Phi(x) \cdot \Phi(x')$$

- Preslikavanje Φ nije poznato
- Čak i da znamo Φ, moglo bi da se desi da ovo preslikavanje slika podatke u neki beskonačno dimenzionalni prostor
- Bolja strategija: zamena skalarnog proizvoda kernelom u svim formulama, što je sa matematičke tačke gledišta isto
- U tom slučaju model postaje:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + w_0$$

Kernelizovani metod potpornih vektora za klasifikaciju - primer

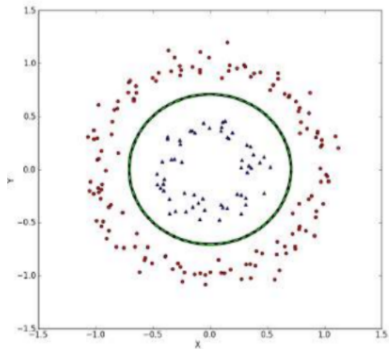
- Neka su podaci raspoređeni tako da elementi jedne klase unutar lopte određenog poluprečnika u prostoru atributa, dok su elementi druge klase van te lopte



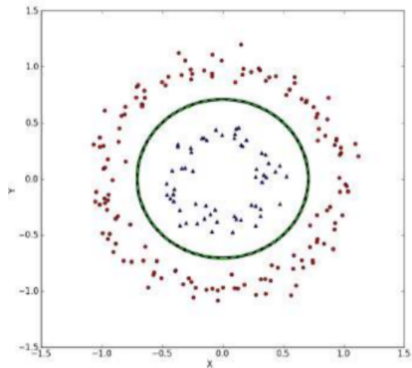
Kernelizovani metod potpornih vektora za klasifikaciju - primer

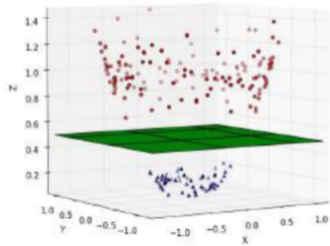
- Neka su podaci raspoređeni tako da elementi jedne klase unutar lopte određenog poluprečnika u prostoru atributa, dok su elementi druge klase van te lopte
- Definišimo kernel zasnovan na preslikavanju $\Phi(x) = (x_1, x_2, x_1^2 + x_2^2)$:

$$k(x, x') = (x_1, x_2, x_1^2 + x_2^2) \cdot (x_1', x_2', x_1'^2 + x_2'^2)$$



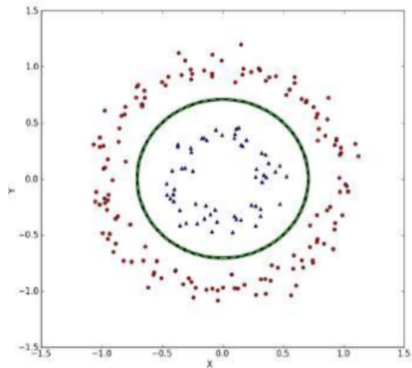
Kernelizovani metod potpornih vektora za klasifikaciju - primer

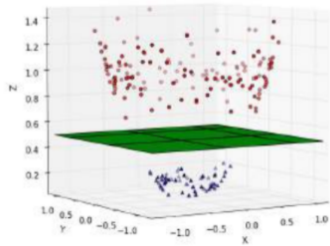




- Ovo preslikavanje očigledno slika podatke u prostor veće dimenzije u kojem se mogu razdvojiti pomoću hiperravni

Kernelizovani metod potpornih vektora za klasifikaciju - primer





- Ovo preslikavanje očigledno slika podatke u prostor veće dimenzije u kojem se mogu razdvojiti pomoću hiperravni
- U polaznom prostoru, inverzna slika razdvajajuće hiperravni izgleda kao kružnica.

Kernelizovani metod potpornih vektora za klasifikaciju

- Šta ako nam raspored instanci u prostoru nije poznat?

Kernelizovani metod potpornih vektora za klasifikaciju

- Šta ako nam raspored instanci u prostoru nije poznat?
- Ispostavlja se da je jedna vrsta kernela dovoljna (iako ne nužno uvek najpogodnija)

Kernelizovani metod potpornih vektora za klasifikaciju

- Šta ako nam raspored instanci u prostoru nije poznat?
- Ispostavlja se da je jedna vrsta kernela dovoljna (iako ne nužno uvek najpogodnija)
- To je takozvani Gausov kernel:

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

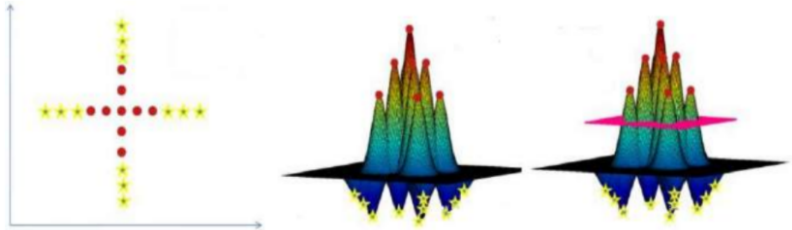
Kernelizovani metod potpornih vektora za klasifikaciju

- Šta ako nam raspored instanci u prostoru nije poznat?
- Ispostavlja se da je jedna vrsta kernela dovoljna (iako ne nužno uvek najpogodnija)
- To je takozvani Gausov kernel:

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

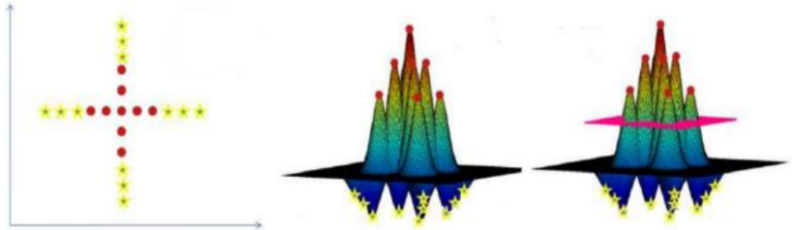
- Veće vrednosti parametra γ vode širim Gausovim zvonima, a manje užim

Kernelizovani metod potpornih vektora za klasifikaciju



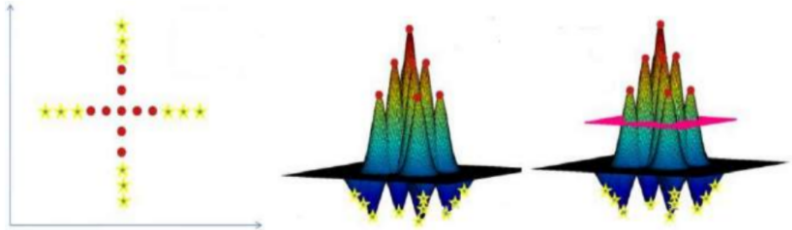
- Za dovoljno malo γ svaka tačka može biti potporni vektor sa pridruženom okolinom u kojoj nema drugih tačaka

Kernelizovani metod potpornih vektora za klasifikaciju



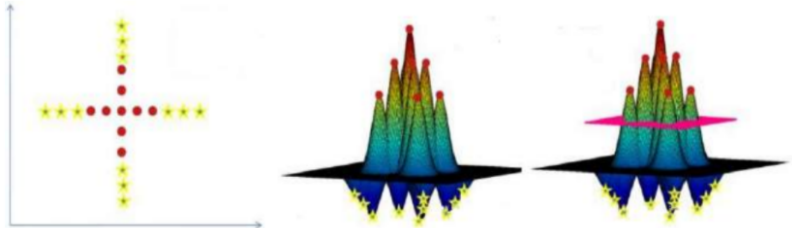
- Za dovoljno malo γ svaka tačka može biti potporni vektor sa pridruženom okolinom u kojoj nema drugih tačaka
- Množenjem Gausovog zvona znakom klase može se postići njena tačna klasifikacija

Kernelizovani metod potpornih vektora za klasifikaciju



- Za dovoljno malo γ svaka tačka može biti potporni vektor sa pridruženom okolinom u kojoj nema drugih tačaka
- Množenjem Gausovog zvona znakom klase može se postići njena tačna klasifikacija
- Ukoliko se dozvole kerneli sa dovoljno malom širinom γ, ovo je moguće postići na svakom neprotivrečnom skupu podataka

Kernelizovani metod potpornih vektora za klasifikaciju



- Za dovoljno malo γ svaka tačka može biti potporni vektor sa pridruženom okolinom u kojoj nema drugih tačaka
- Množenjem Gausovog zvona znakom klase može se postići njena tačna klasifikacija
- Ukoliko se dozvole kerneli sa dovoljno malom širinom γ, ovo je moguće postići na svakom neprotivrečnom skupu podataka
- Ipak, u praksi se nikad ne omogućava korišćenje proizvoljnih vrednosti parametra γ, već se γ koristi kao metaparametar, nalik regularizacionom metaparametru.

Kernelizovani metod potpornih vektora za klasifikaciju

- Metod potpornih vektora ima sledeću formu modela:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + w_0$$

Kernelizovani metod potpornih vektora za klasifikaciju

- Metod potpornih vektora ima sledeću formu modela:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + w_0$$

- U slučaju linearnog kernela, odnosno običnog skalarnog proizvoda, koeficijenti linearnog modela su se mogli eksplicitno izračunati zahvaljujući distributivnosti skalarnog proizvoda u odnosu na sabiranje:

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i$$

Kernelizovani metod potpornih vektora za klasifikaciju

- Metod potpornih vektora ima sledeću formu modela:

$$f_{w,w_0}(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + w_0$$

- U slučaju linearnog kernela, odnosno običnog skalarnog proizvoda, koeficijenti linearnog modela su se mogli eksplicitno izračunati zahvaljujući distributivnosti skalarnog proizvoda u odnosu na sabiranje:

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i$$

- U ovom, opštijem slučaju to nije moguće, što znači da je čuvanje instanci neophodno, kako bi se izračunale vrednosti kernela

Kernelizovani metod potpornih vektora za klasifikaciju

- Čuvanje instanci i njihovo korišćenje tokom predviđanja je uobičajeno za metode zasnovane na kernelima

Kernelizovani metod potpornih vektora za klasifikaciju

- Čuvanje instanci i njihovo korišćenje tokom predviđanja je uobičajeno za metode zasnovane na kernelima
- Međutim, u ovom kontekstu već pomenuto svojstvo metoda potpornih vektora da model zavisi samo od nekih instanci, dobija na važnosti

Kernelizovani metod potpornih vektora za klasifikaciju

- Čuvanje instanci i njihovo korišćenje tokom predviđanja je uobičajeno za metode zasnovane na kernelima
- Međutim, u ovom kontekstu već pomenuto svojstvo metoda potpornih vektora da model zavisi samo od nekih instanci, dobija na važnosti
- Za razliku od drugih metoda zasnovanih na kernelima, u slučaju ovog metoda dovoljno je čuvati samo neke instance, a prilikom predviđanja, zahvaljujući manjem broju instanci za koje se računaju vrednosti kernela, izračunavanje je brže

Kernelizovanje drugih metoda

- Zamenom skalarnog proizvoda kernelom, može se kernelizovati i metod potpornih vektora za regresiju

Kernelizovanje drugih metoda

- Zamenom skalarnog proizvoda kernelom, može se kernelizovati i metod potpornih vektora za regresiju
- I šire, mnogi metodi mašinskog učenja kod kojih se model može izraziti u terminima skalarnih proizvoda sa instancama iz skupa za obučavanje, mogu se kernelizovati, što često vodi boljim prediktivnim performansama.

Pregled

Metodi zasnovani na najbližim susedima

- Kao što metodi zasnovani na kernelima počivaju na upotrebi funkcija *sličnosti*, tako metodi zasnovani na najbližim susedima počivaju na upotrebi funkcija *rastojanja*

Metodi zasnovani na najbližim susedima

- Kao što metodi zasnovani na kernelima počivaju na upotrebi funkcija *sličnosti*, tako metodi zasnovani na najbližim susedima počivaju na upotrebi funkcija *rastojanja*
- Tipičan metaparametar ovakvih metoda je *broj suseda* koji se razmatra

Pregled

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V
- Veličine k i V su međusobno zavisne

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V
- Veličine k i V su međusobno zavisne
  - fiksiramo V - metodi zasnovani na kernelima

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V
- Veličine k i V su međusobno zavisne
  - fiksiramo V - metodi zasnovani na kernelima
  - fiksiramo k - metodi zasnovani na najbližim susedima

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V
- Veličine k i V su međusobno zavisne
  - fiksiramo V - metodi zasnovani na kernelima
  - fiksiramo k - metodi zasnovani na najbližim susedima
- Do metoda k najbližih suseda za ocenu gustine raspodele dolazi se kada se umesto fiksiranja zapremine, kao u slučaju Parzenovog prozora, kao princip konstruisanja ocene gustine raspodele uzme fiksiranje broja tačaka k koji treba da budu obuhvaćeni okolinom tačke u kojoj se ocenjuje gustina raspodele

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V
- Veličine k i V su međusobno zavisne
  - fiksiramo V - metodi zasnovani na kernelima
  - fiksiramo k - metodi zasnovani na najbližim susedima
- Do metoda k najbližih suseda za ocenu gustine raspodele dolazi se kada se umesto fiksiranja zapremine, kao u slučaju Parzenovog prozora, kao princip konstruisanja ocene gustine raspodele uzme fiksiranje broja tačaka k koji treba da budu obuhvaćeni okolinom tačke u kojoj se ocenjuje gustina raspodele
- Tada se računa sfera najmanje zapremine V_k koja sadrži k tačaka i dolazi se do ocene

$$p(x) = \frac{k}{NV_k}$$

Ocena gustine raspodele zasnovana na najbližim susedima

- Pokazali smo da važi $p(x) \approx \frac{k}{NV}$, gde je N broj opažanja koje dolaze iz raspodele $p(x)$, V zapremina prostora koji posmatramo, k broj instanci od N koji se nalaze u prostoru zapremine V
- Veličine k i V su međusobno zavisne
  - fiksiramo V - metodi zasnovani na kernelima
  - fiksiramo k - metodi zasnovani na najbližim susedima
- Do metoda k najbližih suseda za ocenu gustine raspodele dolazi se kada se umesto fiksiranja zapremine, kao u slučaju Parzenovog prozora, kao princip konstruisanja ocene gustine raspodele uzme fiksiranje broja tačaka k koji treba da budu obuhvaćeni okolinom tačke u kojoj se ocenjuje gustina raspodele
- Tada se računa sfera najmanje zapremine V_k koja sadrži k tačaka i dolazi se do ocene

$$p(x) = \frac{k}{NV_k}$$

- Primetimo da oblik sfere zavisi od izabrane metrike.

Ocena gustine raspodele zasnovana na najbližim susedima

- Ocena gustine raspodele zasnovana na najbližim susedima:

$$p(x) = \frac{k}{NV_k}$$

Ocena gustine raspodele zasnovana na najbližim susedima

- Ocena gustine raspodele zasnovana na najbližim susedima:

$$p(x) = \frac{k}{NV_k}$$

- Problem sa ovakvom ocenom gustine je što za konačne uzorke zapravo ne predstavlja validnu gustinu raspodele, zato što integral po x divergira

Ocena gustine raspodele zasnovana na najbližim susedima

- Ocena gustine raspodele zasnovana na najbližim susedima:

$$p(x) = \frac{k}{NV_k}$$

- Problem sa ovakvom ocenom gustine je što za konačne uzorke zapravo ne predstavlja validnu gustinu raspodele, zato što integral po x divergira
- Recimo, u slučaju $k = 1$ će u tačkama uzorka zapremina najmanje sfere biti nula, a ocena gustine beskonačna

Ocena gustine raspodele zasnovana na najbližim susedima

- Ocena gustine raspodele zasnovana na najbližim susedima:

$$p(x) = \frac{k}{NV_k}$$

- Problem sa ovakvom ocenom gustine je što za konačne uzorke zapravo ne predstavlja validnu gustinu raspodele, zato što integral po x divergira
- Recimo, u slučaju $k = 1$ će u tačkama uzorka zapremina najmanje sfere biti nula, a ocena gustine beskonačna
- Ipak, ova ocena može biti korisna, makar za izvođenje drugih metoda

Ocena gustine raspodele zasnovana na najbližim susedima

- Male vrednosti broja k vode preprilagođavanju tako što se masa verovatnoće raspoređuje sve više na tačke iz skupa za obučavanje, a sve manje na druge tačke

Ocena gustine raspodele zasnovana na najbližim susedima

- Male vrednosti broja k vode preprilagođavanju tako što se masa verovatnoće raspoređuje sve više na tačke iz skupa za obučavanje, a sve manje na druge tačke
- Velike vrednosti broja k vode usrednjavanju i gubljenu informacije, tako što se masa verovatnoće raspoređuje ravnomernije nego što bi trebalo, čime se gubi struktura raspodele podataka koja u konkretnom slučaju možda i nije ravnomerna

Pregled

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja
- Može služiti za klasifikaciju sa proizvoljnim brojem klasa, kao i za regresiju

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja
- Može služiti za klasifikaciju sa proizvoljnim brojem klasa, kao i za regresiju
- Osnovna pretpostavka ovog algoritma je postojanje rastojanja nad prostorom atributa

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja
- Može služiti za klasifikaciju sa proizvoljnim brojem klasa, kao i za regresiju
- Osnovna pretpostavka ovog algoritma je postojanje rastojanja nad prostorom atributa
- Najčešće se pretpostavlja vektorska reprezentacija instanci i euklidsko rastojanje, ali takve pretpostavke nisu neophodne

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja
- Može služiti za klasifikaciju sa proizvoljnim brojem klasa, kao i za regresiju

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja
- Može služiti za klasifikaciju sa proizvoljnim brojem klasa, kao i za regresiju
- Osnovna pretpostavka ovog algoritma je postojanje rastojanja nad prostorom atributa

Algoritam k najbližih suseda

- Algoritam k najbližih suseda verovatno je najjednostavniji algoritam mašinskog učenja
- Može služiti za klasifikaciju sa proizvoljnim brojem klasa, kao i za regresiju
- Osnovna pretpostavka ovog algoritma je postojanje rastojanja nad prostorom atributa
- Najčešće se pretpostavlja vektorska reprezentacija instanci i euklidsko rastojanje, ali takve pretpostavke nisu neophodne

Algoritam k najbližih suseda

- Procena verovatnoće klase na osnovu vrednosti atributa može se uraditi korišćenjem Bajesove formule i ocene gustine raspodele pomoću k najbližih suseda

Algoritam k najbližih suseda

- Procena verovatnoće klase na osnovu vrednosti atributa može se uraditi korišćenjem Bajesove formule i ocene gustine raspodele pomoću k najbližih suseda
- N_y - broj podataka koji pripada klasi y

Algoritam k najbližih suseda

- Procena verovatnoće klase na osnovu vrednosti atributa može se uraditi korišćenjem Bajesove formule i ocene gustine raspodele pomoću k najbližih suseda
- N_y - broj podataka koji pripada klasi y
- k_y broj tačaka iz k najbližih suseda koji pripadaju klasi y

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} = \frac{\frac{k_y}{N_y V_k}\frac{N_y}{N}}{\frac{k}{NV_k}} = \frac{k_y}{k}$$

Algoritam k najbližih suseda

- Na osnovu izvedene uslovne raspodele $p(y|x) = \frac{k_y}{k}$, predviđanje se vrši na uobičajen način:

$$f(x) = \operatorname*{argmax}_y p(y|x)$$

Algoritam k najbližih suseda

- Na osnovu izvedene uslovne raspodele $p(y|x) = \frac{k_y}{k}$, predviđanje se vrši na uobičajen način:
$$f(x) = \underset{y}{\operatorname{argmax}}\, p(y|x)$$
- Algoritam k najbližih suseda klasifikuje nepoznatu instancu tako što pronalazi k instanci iz skupa za obučavanje koje su joj najbliže u smislu neke izabrane metrike i pridružuje joj klasu koja se najčešće javlja među tih k instanci

Algoritam k najbližih suseda

- Na osnovu izvedene uslovne raspodele $p(y|x) = \frac{k_y}{k}$, predviđanje se vrši na uobičajen način:
$$f(x) = \operatorname*{argmax}_y p(y|x)$$
- Algoritam k najbližih suseda klasifikuje nepoznatu instancu tako što pronalazi k instanci iz skupa za obučavanje koje su joj najbliže u smislu neke izabrane metrike i pridružuje joj klasu koja se najčešće javlja među tih k instanci
- Diskriminativni probabilistički algoritam

Algoritam k najbližih suseda

- Na osnovu izvedene uslovne raspodele $p(y|x) = \frac{k_y}{k}$, predviđanje se vrši na uobičajen način:
$$f(x) = \underset{y}{\operatorname{argmax}}\, p(y|x)$$
- Algoritam k najbližih suseda klasifikuje nepoznatu instancu tako što pronalazi k instanci iz skupa za obučavanje koje su joj najbliže u smislu neke izabrane metrike i pridružuje joj klasu koja se najčešće javlja među tih k instanci
- Diskriminativni probabilistički algoritam
- Ne predstavlja najbolji izbor za rešavanje nekog problema, ali neretko daje relativno dobre rezultate, a izuzetno lako se implementira i primenjuje

Algoritam k najbližih suseda - primer

Hladnoća	Curenje iz nosa	Glavobolja	Groznica	Grip
Da	Ne	Blaga	Da	Ne
Da	Da	Ne	Ne	Da
Da	Ne	Jaka	Da	Da
Ne	Da	Blaga	Da	Da
Ne	Ne	Ne	Ne	Ne
Ne	Da	Jaka	Da	Da
Ne	Da	Jaka	Ne	Ne
Da	Da	Blaga	Da	Da

- U odnosu na date podatke, potrebno je klasifikovati instancu (*Da*, *Ne*, *Blaga*, *Ne*)

Algoritam k najbližih suseda - primer

Hladnoća	Curenje iz nosa	Glavobolja	Groznica	Grip
Da	Ne	Blaga	Da	Ne
Da	Da	Ne	Ne	Da
Da	Ne	Jaka	Da	Da
Ne	Da	Blaga	Da	Da
Ne	Ne	Ne	Ne	Ne
Ne	Da	Jaka	Da	Da
Ne	Da	Jaka	Ne	Ne
Da	Da	Blaga	Da	Da

- U odnosu na date podatke, potrebno je klasifikovati instancu $(Da, Ne, Blaga, Ne)$
- Kako su vrednosti atributa kategoričke, potrebno je definisati specifičnu funkciju rastojanja, na primer $d(x, x') = \sum_{i=1}^{n} I(x_i \neq x_i')$

Algoritam k najbližih suseda - primer

Hladnoća	Curenje iz nosa	Glavobolja	Groznica	Grip
Da	Ne	Blaga	Da	Ne
Da	Da	Ne	Ne	Da
Da	Ne	Jaka	Da	Da
Ne	Da	Blaga	Da	Da
Ne	Ne	Ne	Ne	Ne
Ne	Da	Jaka	Da	Da
Ne	Da	Jaka	Ne	Ne
Da	Da	Blaga	Da	Da

- U odnosu na date podatke, potrebno je klasifikovati instancu (Da, Ne, $Blaga$, Ne)
- Kako su vrednosti atributa kategoričke, potrebno je definisati specifičnu funkciju rastojanja, na primer $d(x, x') = \sum_{i=1}^{n} I(x_i \neq x_i')$
- Ako se u obzir uzima 1 najbliži sused, najbliži je prvi primer iz tabele, pa je predviđena klasa Ne.

Algoritam k najbližih suseda

- Primetimo da ovaj algoritam nema eksplicitnu formu modela, funkciju greške u odnosu na koju bi se model obučavao, pa ni fazu obučavanja uopšte, osim izbora vrednosti metaparametra k

Algoritam k najbližih suseda

- Primetimo da ovaj algoritam nema eksplicitnu formu modela, funkciju greške u odnosu na koju bi se model obučavao, pa ni fazu obučavanja uopšte, osim izbora vrednosti metaparametra k
- Slično važi i za metod Nadaraja-Votson, pa i za mnoge druge (ali ne sve) metode zasnovane na instancama

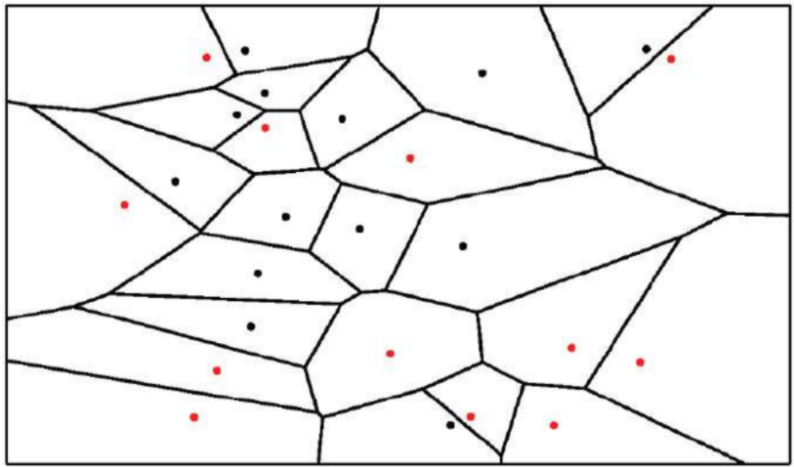
Algoritam k najbližih suseda

- Primetimo da ovaj algoritam nema eksplicitnu formu modela, funkciju greške u odnosu na koju bi se model obučavao, pa ni fazu obučavanja uopšte, osim izbora vrednosti metaparametra k
- Slično važi i za metod Nadaraja-Votson, pa i za mnoge druge (ali ne sve) metode zasnovane na instancama
- U slučaju metoda k najbližih suseda, sav rad se obavlja u fazi predviđanja

Algoritam k najbližih suseda

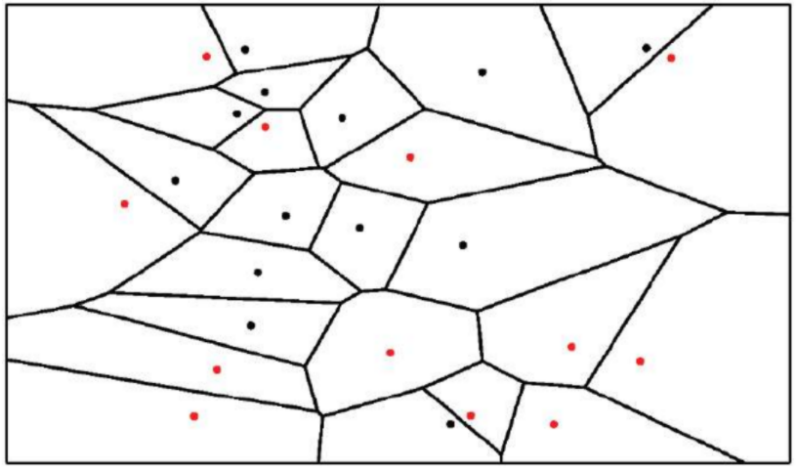
- Primetimo da ovaj algoritam nema eksplicitnu formu modela, funkciju greške u odnosu na koju bi se model obučavao, pa ni fazu obučavanja uopšte, osim izbora vrednosti metaparametra k
- Slično važi i za metod Nadaraja-Votson, pa i za mnoge druge (ali ne sve) metode zasnovane na instancama
- U slučaju metoda k najbližih suseda, sav rad se obavlja u fazi predviđanja
- Potrebno je čuvati sve instance ili, eventualno, neki podskup predstavnika ukoliko je broj instanci preveliki.

Algoritam k najbližih suseda - vizualizacija



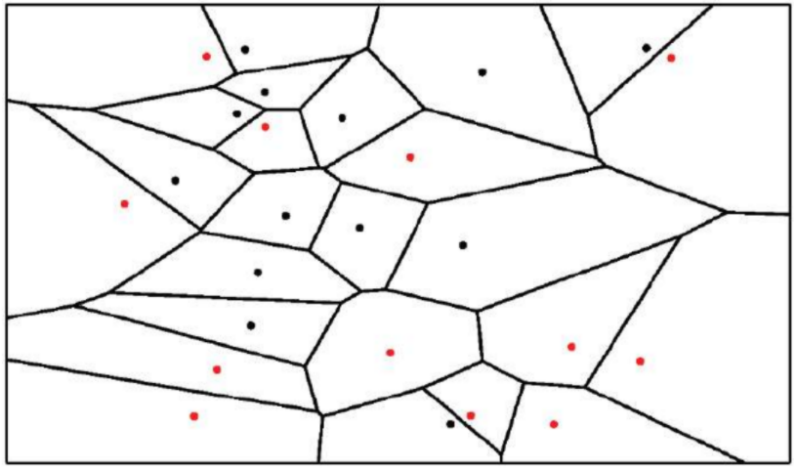
- Podela prostora atributa pomoću algoritma k najbližih suseda za $k = 1$

Algoritam k najbližih suseda - vizualizacija



- Podela prostora atributa pomoću algoritma k najbližih suseda za $k = 1$
- Svakoj instanci je pridružen skup tačaka koje su bliže toj instanci nego drugim instancama iz skupa za obučavanje

Algoritam k najbližih suseda - vizualizacija



- Za razliku od prethodnih algoritama klasifikacije koji su pretpostavljali da je razdvajajuća granica među klasama hiperravan, ovaj algoritam dozvoljava značajno komplikovaniju razdvajajuću granicu među klasama

Algoritam k najbližih suseda - vizualizacija za $k = 1$ i $k = 10$

- U ekstremnom slučaju, kada je k jednako veličini skupa za obučavanje, ceo prostor se klasifikuje u istu – većinsku klasu

Algoritam k najbližih suseda - vizualizacija za $k = 1$ i $k = 10$

- U ekstremnom slučaju, kada je k jednako veličini skupa za obučavanje, ceo prostor se klasifikuje u istu – većinsku klasu
- Ovo je slično kako ponašanju metoda zasnovanih na kernelima u odnosu na izbor metaparametra σ, tako i ponašanju regularizacije kod parametarskih metoda

Algoritam k najbližih suseda - vizualizacija za $k = 1$ i $k = 10$

- Što je vrednost k manja, ovaj algoritam je u većoj opasnosti od preprilagođavanja podacima za obučavanje

Algoritam k najbližih suseda - vizualizacija za $k = 1$ i $k = 10$

- Što je vrednost k manja, ovaj algoritam je u većoj opasnosti od preprilagođavanja podacima za obučavanje
- Što je vrednost k veća, ta opasnost je manja, ali lakše dolazi do potprilagođavanja

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- U visokodimenzionalnim prostorima, rastojanja se ponašaju drugačije nego što po intuiciji trodimenzionalnog prostora očekujemo

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- U visokodimenzionalnim prostorima, rastojanja se ponašaju drugačije nego što po intuiciji trodimenzionalnog prostora očekujemo
- Razmotrimo primer dva niza zapremina n dimenzionalnih kocki – sa stranicom dužine 1 i stranicom dužine 0.99. Važi

$$\lim_{n\to\infty} \frac{V_{0.99}^n}{V_1^n} = \lim_{n\to\infty} \frac{0.99^n}{1^n} = 0$$

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- U visokodimenzionalnim prostorima, rastojanja se ponašaju drugačije nego što po intuiciji trodimenzionalnog prostora očekujemo
- Razmotrimo primer dva niza zapremina n dimenzionalnih kocki – sa stranicom dužine 1 i stranicom dužine 0.99. Važi

$$\lim_{n\to\infty} \frac{V_{0.99}^n}{V_1^n} = \lim_{n\to\infty} \frac{0.99^n}{1^n} = 0$$

- Slično se može pokazati i za lopte vrlo bliskih poluprečnika, sa centrom u istoj tački

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- U visokodimenzionalnim prostorima, rastojanja se ponašaju drugačije nego što po intuiciji trodimenzionalnog prostora očekujemo
- Razmotrimo primer dva niza zapremina n dimenzionalnih kocki – sa stranicom dužine 1 i stranicom dužine 0.99. Važi

$$\lim_{n\to\infty} \frac{V_{0.99}^n}{V_1^n} = \lim_{n\to\infty} \frac{0.99^n}{1^n} = 0$$

- Slično se može pokazati i za lopte vrlo bliskih poluprečnika, sa centrom u istoj tački
- Zaključujemo da su u visokodimenzionalnim prostorima praktično sve tačke lopte vrlo daleko od njenog centra.

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Zaključujemo da su u visokodimenzionalnim prostorima praktično sve tačke lopte vrlo daleko od njenog centra.

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Zaključujemo da su u visokodimenzionalnim prostorima praktično sve tačke lopte vrlo daleko od njenog centra.
- Štaviše da su uglavnom na istom rastojanju od centra (npr. na rastojanju većem od $0.99r$, a manjem od r).

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Zaključujemo da su u visokodimenzionalnim prostorima praktično sve tačke lopte vrlo daleko od njenog centra.
- Štaviše da su uglavnom na istom rastojanju od centra (npr. na rastojanju većem od $0.99r$, a manjem od r).
- Nekoliko teorijskih rezultata pokazuje da je u visokodimenzionalnim prostorima varijacija distanci između nasumice generisanih tačaka mala

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Zaključujemo da su u visokodimenzionalnim prostorima praktično sve tačke lopte vrlo daleko od njenog centra.
- Štaviše da su uglavnom na istom rastojanju od centra (npr. na rastojanju većem od $0.99r$, a manjem od r).
- Nekoliko teorijskih rezultata pokazuje da je u visokodimenzionalnim prostorima varijacija distanci između nasumice generisanih tačaka mala
- Ovi uvidi su obeshrabrujući za primenu algoritma k najbližih suseda, pošto se on zasniva na razlikovanju bliskih i dalekih suseda, a ako su te razlike male, njegova predviđanja nisu pouzdana

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Prokletstvo dimenzionalnosti

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Prokletstvo dimenzionalnosti
- Ukoliko se tačka klasifikuje na osnovu bliskih suseda, očekuje se da postoje bliski susedi

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Prokletstvo dimenzionalnosti
- Ukoliko se tačka klasifikuje na osnovu bliskih suseda, očekuje se da postoje bliski susedi
- To je moguće ako je prostor gusto popunjen podacima za obučavanje

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Prokletstvo dimenzionalnosti
- Ukoliko se tačka klasifikuje na osnovu bliskih suseda, očekuje se da postoje bliski susedi
- To je moguće ako je prostor gusto popunjen podacima za obučavanje
- Kako dimenzionalnost prostora raste, broj tačaka potreban za održavanje nekog konstantnog stepena gustine raste eksponencijalno u odnosu na dimenziju i nije za očekivati da će dovoljna količina podataka biti na raspolaganju

Algoritam k najbližih suseda - mane

- Loše se ponaša sa višedimenzionim podacima
- Prokletstvo dimenzionalnosti
- Ukoliko se tačka klasifikuje na osnovu bliskih suseda, očekuje se da postoje bliski susedi
- To je moguće ako je prostor gusto popunjen podacima za obučavanje
- Kako dimenzionalnost prostora raste, broj tačaka potreban za održavanje nekog konstantnog stepena gustine raste eksponencijalno u odnosu na dimenziju i nije za očekivati da će dovoljna količina podataka biti na raspolaganju
- Čak i ako jeste, može se ispostaviti da predviđanja budu računski skupa

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Atributi koji se mere na različitim skalama

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Atributi koji se mere na različitim skalama
- Razmotrimo slučaj euklidske metrike

$$d(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x'_i)^2}$$

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Atributi koji se mere na različitim skalama
- Razmotrimo slučaj euklidske metrike

$$d(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x_i')^2}$$

- Ukoliko atributi x_1 i x_2 predstavljaju dužine, ali je prvi meren u milimetrima, a drugi u metrima, razlika $(x_1 - x_1')^2$ će tipično biti mnogo veća od razlike $(x_2 - x_2')^2$, prosto zbog različite skale na kojoj se vrednosti izražavaju, a ne zbog razlike u stvarnim dužinama

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Atributi koji se mere na različitim skalama
- Razmotrimo slučaj euklidske metrike

$$d(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x_i')^2}$$

- Ukoliko atributi x_1 i x_2 predstavljaju dužine, ali je prvi meren u milimetrima, a drugi u metrima, razlika $(x_1 - x_1')^2$ će tipično biti mnogo veća od razlike $(x_2 - x_2')^2$, prosto zbog različite skale na kojoj se vrednosti izražavaju, a ne zbog razlike u stvarnim dužinama
- Atribut x_1 će više uticati na ishod klasifikacije, nego atributa x_2, što ne mora biti opravdano

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Atributi koji se mere na različitim skalama
- Razmotrimo slučaj euklidske metrike

$$d(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x_i')^2}$$

- Ukoliko atributi x_1 i x_2 predstavljaju dužine, ali je prvi meren u milimetrima, a drugi u metrima, razlika $(x_1 - x_1')^2$ će tipično biti mnogo veća od razlike $(x_2 - x_2')^2$, prosto zbog različite skale na kojoj se vrednosti izražavaju, a ne zbog razlike u stvarnim dužinama
- Atribut x_1 će više uticati na ishod klasifikacije, nego atributa x_2, što ne mora biti opravdano
- Otud se pre primene algoritma k najbližih suseda obavezno primenjuje neki vid pretprocesiranja koji svodi različite atribute na istu skalu (npr. standardizacija)

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Ponovljeni atributi ili visoka koreliranost atributa

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Ponovljeni atributi ili visoka koreliranost atributa
- Neka su podaci opisani pomoću dva visoko korelirana atributa i neka je potom jedan umnožen 100 puta

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Ponovljeni atributi ili visoka koreliranost atributa
- Neka su podaci opisani pomoću dva visoko korelirana atributa i neka je potom jedan umnožen 100 puta
- U euklidskom rastojanju će razlika po prvom atributu figurisati jednom, a po drugom 100 puta

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Ponovljeni atributi ili visoka koreliranost atributa
- Neka su podaci opisani pomoću dva visoko korelirana atributa i neka je potom jedan umnožen 100 puta
- U euklidskom rastojanju će razlika po prvom atributu figurisati jednom, a po drugom 100 puta
- Otud će uticaj prvog atributa na predviđanje biti zanemarljiv u odnosu na uticaj drugog, a oba mogu biti podjednako informativna, ili čak prvi može biti informativniji

Algoritam k najbližih suseda - problemi primene

- Postoje još neki problemi primene ovog algoritma koji se mogu otkloniti određenim pretprocesiranjima
- Ponovljeni atributi ili visoka koreliranost atributa
- Neka su podaci opisani pomoću dva visoko korelirana atributa i neka je potom jedan umnožen 100 puta
- U euklidskom rastojanju će razlika po prvom atributu figurisati jednom, a po drugom 100 puta
- Otud će uticaj prvog atributa na predviđanje biti zanemarljiv u odnosu na uticaj drugog, a oba mogu biti podjednako informativna, ili čak prvi može biti informativniji
- Ovakvi problemi se rešavaju ili ublažavaju tehnikama smanjenja dimenzionalnosti podataka poput *analize glavnih pravaca* (eng. principal component analysis).

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena
  - proizvoljni oblici granica između klasa

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena
  - proizvoljni oblici granica između klasa
- Mane

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena
  - proizvoljni oblici granica između klasa
- Mane
  - loše ponašanje predviđanja u visokodimenzionalnim prostorima

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena
  - proizvoljni oblici granica između klasa
- Mane
  - loše ponašanje predviđanja u visokodimenzionalnim prostorima
  - neotpornost na ponovljene i visoko korelirane atribute

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena
  - proizvoljni oblici granica između klasa
- Mane
  - loše ponašanje predviđanja u visokodimenzionalnim prostorima
  - neotpornost na ponovljene i visoko korelirane atribute
  - nedostatak interpretabilnosti

Algoritam k najbližih suseda - prednosti i mane sumirano

- Prednosti
  - jednostavnost
  - laka implementacija i primena
  - proizvoljni oblici granica između klasa
- Mane
  - loše ponašanje predviđanja u visokodimenzionalnim prostorima
  - neotpornost na ponovljene i visoko korelirane atribute
  - nedostatak interpretabilnosti
  - standardne mane algoritama zasnovanih na instancama: potreba za čuvanjem instanci iz skupa za obučavanje i duže vreme primene modela

Algoritam k najbližih suseda - regresija

- Za nepoznatu instancu nalazimo k najbližih suseda i uprosečavamo vrednosti ciljne promenljive tih instanci

Algoritam k najbližih suseda - regresija

- Za nepoznatu instancu nalazimo k najbližih suseda i uprosečavamo vrednosti ciljne promenljive tih instanci
- I u slučaju regresije važe prethodne primedbe vezane za prednosti i mane algoritma.