Ansambli modelov
Primer: Housing

Jure Žabkar
jure.zabkar@fri.uni-lj.si

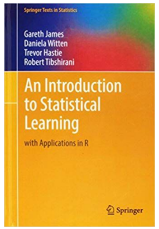
11. 5. 2021

A.I.LAB
Ljubljana

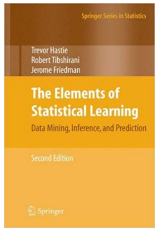
Vsebina

- Bagging
- Naključni gozdovi
- Boosting
- Primer modeliranja: podatki Housing

Literatura



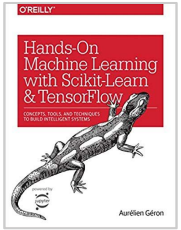
str. 316



str. 282 (Bagging)

str. 587 (naključni gozdovi)

str. 605 (Boosting)

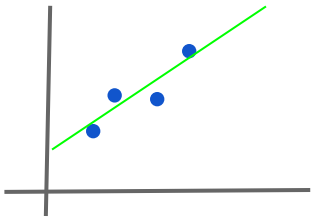


str. 175

Strojno učenje

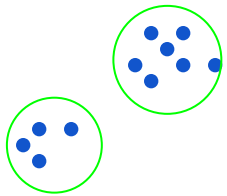
Nadzorovano

Regresija, Klasifikacija

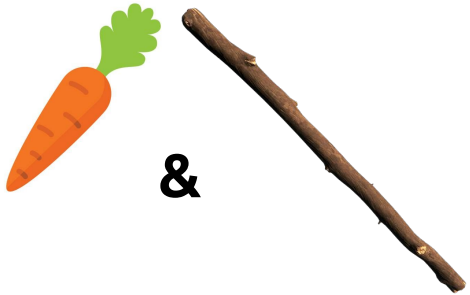


Nenadzorovano

Gručenje, povezovalna pravila



Spodbujevano učenje



Drevesa

- Enostavna, razložljiva
- Klasifikacijska in regresijska
- Obravnavajo tako diskretne kot zvezne atribute

- **Nizka** napovedna točnost v primerjavi z drugimi algoritmi
- **Nestabilna**: majhne spremembe v podatkih zelo vplivajo na naučeni model

Pristranskost in varianca

Napaka modela je sestavljena iz treh vrst napak:

Pristranskost (bias): napaka zaradi napačnih predpostavk (predpostavimo, da je odvisnost v podatkih linearna, a je v resnici kvadratna). Tak model se **premalo** prilega podatkom.

Varianca (variance): napaka zaradi občutljivosti modela na majhne spremembe v podatkih. Kompleksnejši modeli imajo večjo varianco in se lahko **pretirano** prilagodijo podatkom.

Neodpravljiva napaka (irreducible error), npr. šum v podatkih

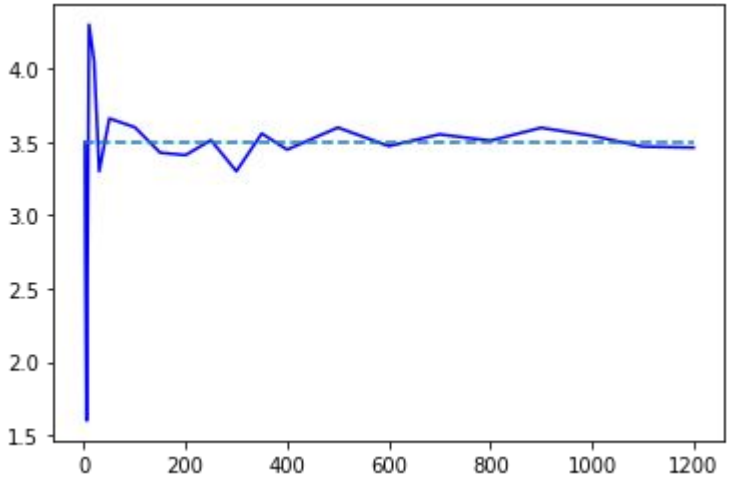

A
B
C
D
50:50

modrost množice ?

Več glav več ve!

Zakon velikih števil

Povprečen rezultat, pridobljen po velikem številu poskusov bo blizu pričakovani vrednosti.



Bagging

- Zmanjša varianco, poveča točnost
- Učenje: bootstrap na učni množici
- Povprečimo napoved velikega števila dreves;
  Drevesa so lahko globoka, neporezana
- Izgubimo razložljivost; ni pretiranega prilagajanja

Bagging

n … število primerov v učni množici L

V vsaki iteraciji:

- Vzorči n primerov s ponavljanjem iz L
- Na vzorcu poženi učni algoritem (npr. učenje drevesa)
- Shrani model

Klasifikacija

- Z vsakim modelom napovej razred za testni primer
- Kot napoved ansambla vrni najbolj pogosto napoved.

Naključni gozd (random forest)

- Bagging dela korelirana drevesa
- Naključni gozd pri gradnji dovoljuje le majhno podmnožico atributov, s čimer preprečuje, da bi pomembni atributi dominirali v vseh drevesih in s tem naredili drevesa korelirana
- Izgubimo razložljivost; ni pretiranega prilagajanja

Naključni gozd (random forest)

Učna množica L, ki vsebuje n primerov in p atributov

Izberemo $m<<p$ (npr. $m=p^{1/2}$)

Algoritem je tak kot za Bagging, le da v vsakem vozlišču naključno izbere m atributov, med katerimi se nato odloča za najbolj informativnega v tem vozlišču.

Boosting

- Uporabi celotno učno množico, brez bootstrap-a
- Uči se na ostankih, namesto na originalnem razredu
- Izgubimo razložljivost, lahko se pretirano prilagodi podatkom (če veliko modelov)

1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all i in the training set.
2. For $b = 1, 2, \ldots, B$, repeat:
   (a) Fit a tree $\hat{f}^b$ with d splits ($d+1$ terminal nodes) to the training data (X, r).
   (b) Update $\hat{f}$ by adding in a shrunken version of the new tree:
   $$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x).$$
   (c) Update the residuals,
   $$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i).$$
3. Output the boosted model,
   $$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x).$$