

12. Ansambli

Strojno učenje 1, UNIZG FER, ak. god. 2021./2022.

Jan Šnajder, natuknice s predavanja, v1.3

1 Kombiniranje klasifikatora

- “**No free lunch theorem**” – niti jedan algoritam ne radi najbolje na svim problemima
- Različiti algoritmi ne griješe na isti način, pa ih ima smisla **kombinirati**
- Kombinacija **osnovnih klasifikatora (base learners)** u **meta-klasifikator**
- Kod nekih metoda, osnovni klasifikatori su **slabi klasifikatori** (*weak learners*)
- Osnovni klasifikatori trebaju biti **različiti i komplementarni**
- Komplementarnost \Rightarrow različiti skupovi za učenje, značajke, algoritmi/hiperparametri

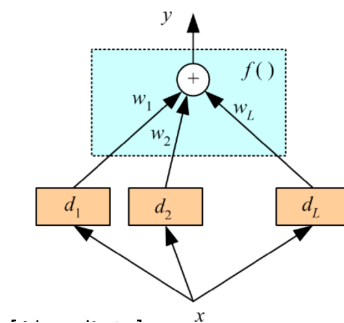
2 Glasanje i stacking

- **Glasanje** – težinska predikcija L osnovnih klasifikatora:

$$h(\mathbf{x}; \mathbf{w}) = \operatorname{argmax}_y \sum_{j=1}^L w_j \mathbf{1}\{h_j(\mathbf{x}) = y\}$$

gdje su w_j težine osnovnih klasifikatora, $w_j \geq 0$, $\sum w_j = 1$

- $w_j = 1/L \Rightarrow$ većinsko glasanje



- Glasanje kod regresije:

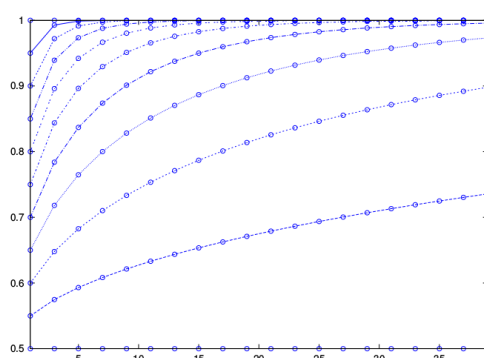
$$h(\mathbf{x}; \mathbf{w}) = \frac{1}{L} \sum_{j=1}^L w_j h_j(\mathbf{x})$$

- Vjerojatnost da većina od L klasifikatora pogriješi:

$$\sum_{i=[L/2]}^L \binom{L}{i} E^i (1-E)^{L-i}$$

⇒ pogreška ansambla ovisi o L i pogrešci E osnovnih klasifikatora

- $E < 0.5 \Rightarrow$ točnost ansambla monotono raste sa L :

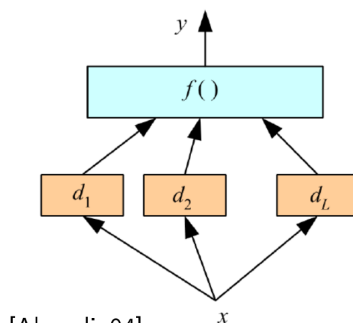


- **Varijanca** ansambla (uz pretpostavku da su h_j nezavisni i imaju jednaku varijancu):

$$\text{Var}(h(\mathbf{x})) = \text{Var}\left(\frac{1}{L} \sum_j h_j(\mathbf{x})\right) = \frac{1}{L^2} \text{Var}\left(\sum_j h_j(\mathbf{x})\right) = \frac{1}{L^2} L \text{Var}(h_j(\mathbf{x})) = \frac{1}{L} \text{Var}(h_j(\mathbf{x}))$$

⇒ varijanca se smanjuje proporcionalno sa L

- **Stacking** – meta-klasifikator treniran na predikcijama osnovnih klasifikatora



- Meta-klasifikator se uči na izdvojenom skupu (**unakrsna provjera**)

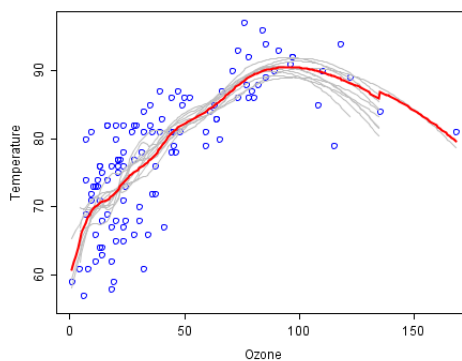
3 Bagging

- **Bagging (bootstrap aggregating)** – treniranje na L poduzoraka skupa za učenje
- Poduzorci dobiveni **uzorkovanjem s ponavljanjem** (*sampling with replacement*)
- Vjerojatnost neuključivanja primjera u uzorak veličine N :

$$\lim_{N \rightarrow \infty} \left(1 - \frac{1}{N}\right)^N = \frac{1}{e} \approx 0.368$$

⇒ vjerojatnost uključivanja primjera je $1 - 0.368 = 0.632 \Rightarrow$ **“0.632 bootstrap”**

- Primjer: bagging za regresiju:



- Problem: osnovni klasifikatori su visoko korelirani
- **Slučajna šuma** – bagging sa **stablina odluke** kao osnovnim klasifikatorima
- Dodatna diverzifikacija slučajnim **odabirom podskupom značajki**
- Konačna predikcija \Rightarrow većinsko glasanje

Slučajna šuma

- 1: $forest \leftarrow \emptyset$
- 2: Za $j = 1 \dots L$
- 3: $\mathcal{D}_j \leftarrow$ bootstrap uzorak
- 4: $\mathcal{F}_j \leftarrow$ odabir n' značajki
- 5: $h_j \leftarrow$ treniraj stablo odluke na \mathcal{D}_j sa značajkama \mathcal{F}_j
- 6: $forest \rightarrow forest \cup \{h_j\}$

4 Boosting

- **Boosting** – slijedno učenje algoritama na pogreškama prethodnih algoritama

- Krenuvši od **slabih klasifikatora**, dobivamo jak klasifikator
- **AdaBoost** – iterativno mijenjanje distribucije uzorkovanja u ovisnosti o pogrešci
- Početna vjerojatnost uzorkovanja $1/N$, raste za pogrešno klasificirane primjere

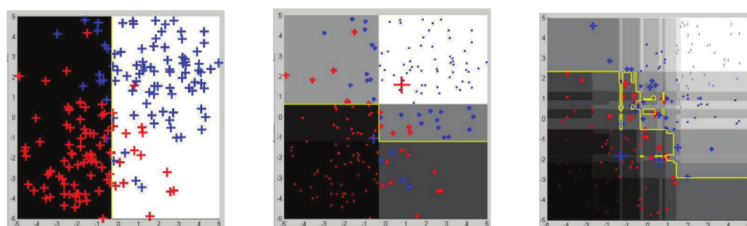
AdaBoost

- 1: inicijaliziraj težine primjera na $w_j^i = 1/N$
- 2: Za $j = 1 \dots L$
- 3: $\mathcal{D}_j \leftarrow$ bootstrap uzorak s težinama \mathbf{w}_j
- 4: $h_j \leftarrow$ treniraj klasifikator na \mathcal{D}_j
- 5: izračunaj pogrešku učenja E_j
- 6: pouzdanost $\alpha_j \leftarrow \ln \frac{1-E_j}{E_j}$
- 7: ažuriraj težine primjera: $w_{j+1}^i \leftarrow w_j^i \exp(\alpha_j \mathbf{1}\{h_j(\mathbf{x}^{(i)}) \neq y^{(i)}\})$
- 8: normaliziraj vektor: $\mathbf{w}_{j+1} \leftarrow \frac{\mathbf{w}_{j+1}}{\|\mathbf{w}_{j+1}\|}$

- Konačna predikcija \Rightarrow težinsko glasanje:

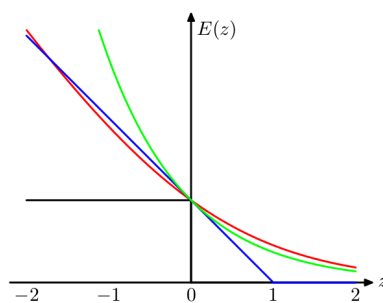
$$h(\mathbf{x}; \boldsymbol{\alpha}) = \operatorname{argmax}_y \sum_{j=1}^L \alpha_j \mathbf{1}\{h_j(\mathbf{x}) = y\}$$

- Osnovni klasifikator: često **panj odluke (decision stump)** – stablo odluke dubine 1
- Primjer (iz MLPP):



- Osnovni modeli zapravo minimiziraju **eksponencijalni gubitak** (zelena krivulja):

$$L(y, h(\mathbf{x})) = \exp(-yh(\mathbf{x}))$$



⇒ strogo kažnjavanje pogrešno klasificiranih primjera

- Alternativni algoritmi boostinga: **logit boost**, **gradient boosting**

SAŽETAK