

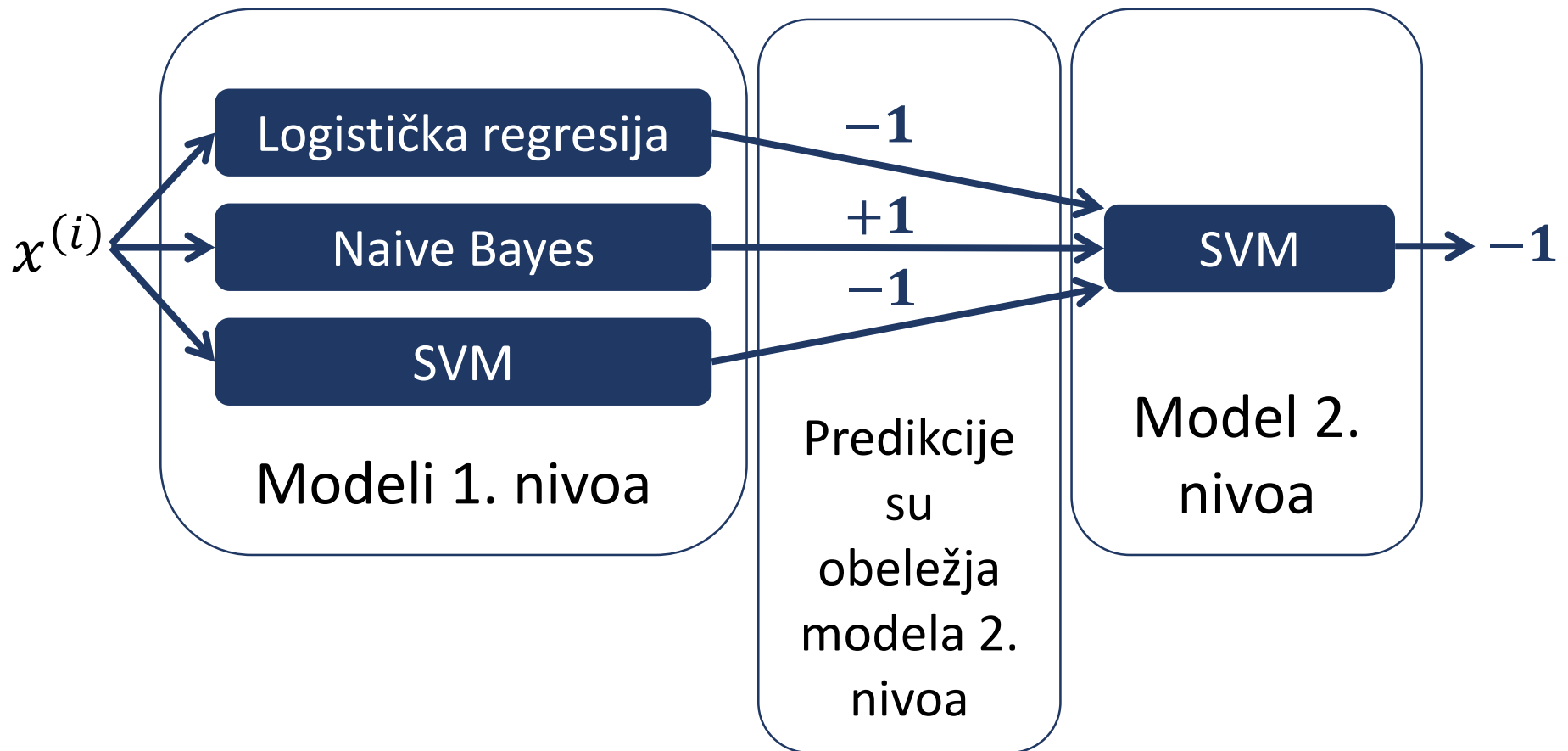


Još neke ideje za kombinovanje klasifikatora

---

# Stacking

- Ideja: naučiti da li su modeli dobro naučeni



# Stacking

Ulaz	<ul style="list-style-type: none"><li>• <math>T = \{(x^{(i)}, y^{(i)}), i = 1, \dots, N\}</math></li><li>• Modeli 1. nivoa <math>L_1, L_2, \dots, L_M</math></li><li>• Model 2. nivoa <math>L</math></li></ul>
Postupak	<ol style="list-style-type: none"><li>1. Trenirati modele 1. nivoa:<ul style="list-style-type: none"><li>• for <math>m = 1 \dots M</math>: <math>h_m = L_m(T)</math></li></ul></li><li>2. Trenirati meta-model <math>L</math>:<ul style="list-style-type: none"><li>• <math>T' = \{\}</math> (generišemo novi trening skup za <math>L</math>)</li><li>• for <math>i = 1 \dots N</math>: for <math>m = 1 \dots M</math>: <math>\hat{y}_m^{(i)} = h_m(x^{(i)})</math> <math>T' = T' \cup \{(\hat{y}_1^{(i)}, \hat{y}_2^{(i)}, \dots, \hat{y}_M^{(i)}), y^{(i)}\}</math></li><li>• <math>h' = L(T')</math></li></ul></li></ol>
Izlaz	$g(x) = h'(h_1(x), h_2(x), \dots, h_M(x))$

# *Before the fact / After the fact*

- *Before the fact*: rešenja se razvijaju sa ciljem da budu kombinovana
  - Ne maksimizujemo performanse individualnih klasifikatora – svaki klasifikator se kreira tako ta bude dobar deo mešavine
  - Bagging, boosting
- *After the fact*: kombinuje već postojeća rešenja
  - Rešenja su razvijena nezavisno, ne razmišljajući o tome da ćemo ih kombinovati
  - Maksimizovane su performanse svakog klasifikatora posebno

# Before the fact – primer



Eyes detector

Mouth detector

Ears detector

Skin detector

Meta Classifier

High accuracy!

Low individual accuracy  
Computationally efficient

# After the fact – primer

- Netflix takmičenje
- Svaki tim je razvio rešenje i maksimizovao performanse *individualno* (bez da ima u vidu mogućnost kombinovanja svog rešenja sa rešenjima drugih timova)
- Kakve težine da dodelimo različitim hipotezama?
  - Ako je dat skup hipoteza  $h_1, \dots, h_M$ :

$$g(x) = \sum_{m=1}^M \alpha_m h_m(x)$$

- Principijalni izbor  $\alpha_m$ : minimizovati grešku na „agregacionom skupu“ (različitom od trening, validacionog i test skupa)
- Može se uraditi metodom najmanjih kvadrata

# After the fact – primer

- Drugo mesto na Netflix takmičenju:
  - Bilo ko može da se pridruži! Pošaljite nam model i podelićemo nagradu prema vašem doprinosu
- Kako odrediti doprinos modela u ansamblu?
  - Da li  $\alpha_m$  može da se koristi za određivanje doprinosa? Šta ako je  $\alpha_m$  negativno?
  - Da bi se odredio doprinos svakog novog modela, tim je na agregacionom skupu evaluirao ansambl konstrisan sa i ansambl konstruisan bez tog modela
  - Ako je novi model „radio nešto neobično“ (različito od ostalih modela) – ovo je rezultovalo većim doprinosima u smislu performansi
  - Modeli sa manjim performansama su često doprinosili više od svojih konkurenata znatno većih performansi

# Zašto metodi ansambla rade dobro?

- Statistički razlozi
- Skup klasifikatora sa sličnim performansama na trening skupu može imati različite generalizacione performanse
- Kombinovanje izlaza više ovakvih klasifikatora smanjuje rizik odabira klasifikatora sa lošim performansama



# Zašto metodi ansambla rade dobro?

- Recimo da imamo ansambl od 25 binarnih klasifikatora, gde svaki klasifikator ima grešku od  $\epsilon = 0.35$
- Ako bi svi klasifikatori bili identični, svaki će pogrešno klasifikovati iste primere – greška ansambla će ostati na 0.35
- Ako su klasifikatori nezavisni (greške koje klasifikatori prave nisu u korelaciji), greška ansambla će biti značajno manja:

$$\epsilon_{ensemble} = \sum_{i=13}^{25} \binom{25}{i} \epsilon^i (1 - \epsilon)^{25-i} = 0.06$$

(ansambl pravi grešku samo ukoliko više od polovine (više od 13) klasifikatora da pogrešnu predikciju)

# Uslovi za uspešan ansambl

- Dva neophodna i dovoljna uslova da bi ansambl klasifikatora imao veće performanse od svih svojih individualnih članova su:
  1. Klasifikatori moraju biti tačni
  2. Klasifikatori moraju biti međusobno različiti (*diverse*)

# Još razloga za uvođenje metoda ansambla

- Previše podataka
  - Ako je količina podataka prevelika, treniranje jednog klasifikatora može da traje predugo ili da uopšte ne bude izvodljivo
  - Rešenje je da se trenira više različitih klasifikatora na različitim delovima podataka
- Premalo podataka
  - Smanjuju varijansu, tj. mogućnost overfitting-a

# Zaključak

- Kompleksni obučavajući problemi se često mogu efektivno rešiti kombinovanjem više slabih prediktora korišćenjem modela ansambla
- Videli smo nekoliko reprezentativnih modela ansambla:
  - glasanje/uprosečavanje ili *stacking*
  - Bagging: Random Forests
  - Boosting: AdaBoost/Gradient boosting
- Prednosti modela ansambla:
  - Redukuju varijansu
  - Redukuju sistematsko odstupanje
  - Redukuju i varijansu i sistematsko odstupanje
- Takođe, mogu da budu značajno brži od kompleksnih modela