

2. Osnovni koncepti

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

Jan Šnajder, natuknice s predavanja, v1.6

1 Primjena algoritma strojnog učenja

1. Priprema i analiza podataka
2. Opcionalno: Označavanje podataka za učenje i ispitivanje
3. Ekstrakcija značajki
4. Opcionalno: Redukcija dimenzionalnosti
5. **Odabir modela**
6. **Učenje modela**
7. **Vrednovanje modela**
8. Dijagnostika i ispravljanje
9. Instalacija

2 Primjeri, hipoteza, model

- Primjer je **vektor značajki**: $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$
- \mathcal{X} je **ulazni prostor (prostor primjera)**; \mathcal{Y} je skup oznaka
- Skup označenih primjera: $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^N \subseteq \mathcal{X} \times \mathcal{Y}$

	x_1	x_2	\dots	x_n	y
$\mathbf{x}^{(1)} =$	$x_1^{(1)}$	$x_2^{(1)}$	\dots	$x_n^{(1)}$	$y^{(1)}$
$\mathbf{x}^{(2)} =$	$x_1^{(2)}$	$x_2^{(2)}$	\dots	$x_n^{(2)}$	$y^{(2)}$
\vdots					
$\mathbf{x}^{(N)} =$	$x_1^{(N)}$	$x_2^{(N)}$	\dots	$x_n^{(N)}$	$y^{(N)}$

- **Hipoteza** – funkcija koja primjerima dodijeljuje oznake: $h : \mathcal{X} \rightarrow \mathcal{Y}$
- **Binarna klasifikacija**: $h : \mathcal{X} \rightarrow \{0, 1\}$

- Hipoteza je definirana do na parametre θ : pišemo $h(\mathbf{x}; \theta)$
 - Regresija u $\mathcal{X} = \mathbb{R}$: $h(x; \theta_0, \theta_1) = \theta_1 x + \theta_0$
 - Klasifikacija pravcem u $\mathcal{X} = \mathbb{R}^2$: $h(x_1, x_2; \theta_0, \theta_1, \theta_2) = \mathbf{1}\{\theta_1 x_1 + \theta_2 x_2 + \theta_0 \geq 0\}$
gdje $\mathbf{1}\{P\} = \begin{cases} 1 & \text{ako } P \equiv \top \\ 0 & \text{inače} \end{cases}$
- **Model** – skup hipoteza parametriziranih s θ : $\mathcal{H} = \{h(\mathbf{x}; \theta)\}_{\theta}$
- **Učenje (treniranje) modela** – pretraživanje skupa \mathcal{H} za najboljom hipotezom

3 Empirijska pogreška i funkcija gubitka

- **Empirijska pogreška** $E(h|\mathcal{D})$ – iskazuje netočnost hipoteze h na skupu podataka \mathcal{D}
 - Pogreška klasifikacije: $E(h|\mathcal{D}) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{h(\mathbf{x})^{(i)} \neq y^{(i)}\}$
- **Funkcija gubitka** (*loss function*) $L(y, h(\mathbf{x}))$ – mjeri pogrešku na jednom primjeru
 - **Gubitak nula-jedan** (*zero-one loss*): $L(y, h(\mathbf{x})) = \mathbf{1}\{h(\mathbf{x})^{(i)} \neq y^{(i)}\}$
- Empirijska pogreška je **očekivana vrijednost** funkcije gubitka na skupu \mathcal{D}

4 Tri komponente algoritma strojnog učenja

1. **Model**: $\mathcal{H} = \{h(\mathbf{x}; \theta)\}_{\theta}$
2. **Funkcija pogreške**: $E(h|\mathcal{D})$ odnosno $E(\theta|\mathcal{D})$
3. **Optimizacijski postupak** koji minimizira empirijsku pogrešku:

$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} E(h|\mathcal{D})$$

odnosno:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} E(\theta|\mathcal{D})$$

5 Složenost modela

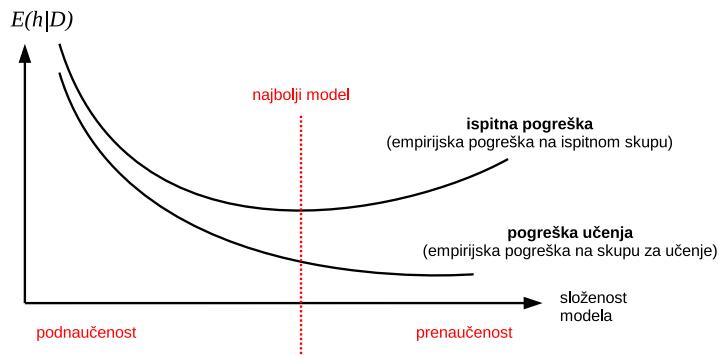
- U idealnom slučaju, $E(h|\mathcal{D}) = 0$
- Ako $\forall h \in \mathcal{H}. E(h|\mathcal{D}) > 0$, onda model nije dovoljne **složenosti (kapaciteta)**
- **Šum** – neželjena anomalija u podatcima
- Uzroci: nepreciznost, pogreške u označavanju, nedostajuće značajke, subjektivnost
- Posljedica šuma: granica između klasa je nepotrebno složena
- Presložen model previše se prilagođava šumu (uči šum)

6 Odabir modela

- Odabir modela iz **familije modela** $\{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_k\}$
- Složenost modela određena je **hiperparametrima** (npr. stupanj nelinearnosti)
- **Odabir modela = optimizacija hiperparametara**
- Preferiramo jednostavnije modele jer bolje **generaliziraju**, lakše se uče i tumače
- **Podnaučenost** – \mathcal{H} je prejednostavan u odnosu na stvarnu funkciju
- **Prenaučenost** – \mathcal{H} je presložen u odnosu na stvarnu funkciju
- Prenaučena hipoteza nije točna na neviđenim primjerima \Rightarrow loša generalizacija

7 Unakrsna provjera

- Ideja: dio primjera iz označenog skupa koristiti kao “neviđene” primjere
- Disjunktna podjela skupa na **skup za učenje** i **skup za ispitivanje**: $\mathcal{D} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{test}}$
- **Pogreška učenja** (*train error*): $E(h|\mathcal{D}_{\text{train}})$
- **Ispitna pogreška** (*test error*): $E(h|\mathcal{D}_{\text{test}})$
- $E(h|\mathcal{D}_{\text{train}})$ pada sa složenošću modela, $E(h|\mathcal{D}_{\text{test}})$ tipično prvo opada a zatim raste
- Skica: pogreška učenja i ispitna pogreška kao funkcije složenosti modela



- Optimalan model je onaj koji minimizira $E(h|\mathcal{D}_{\text{test}})$

3. Regresija

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

Jan Šnajder, natuknice s predavanja, v1.5

1 Jednostavna regresija

- Označen skup podataka: $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}, \mathbf{x} \in \mathbb{R}^n, y \in \mathbb{R}$
- Hipoteza $h : \mathbb{R}^n \rightarrow \mathbb{R}$
- \mathbf{x} – ulazne/nezavisne/prediktorske varijable; y – izlazna/zavisna/kriterijska varijabla
- **Linearna regresija:**

$$h(\mathbf{x}; \mathbf{w}) = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$$

- **Jednostavna regresija** ($n = 1$):

$$h(x; w_0, w_1) = w_0 + w_1 x$$

- Funkcija gubitka je **kvadratni gubitak**: $L(y, h(x)) = (y - h(x))^2$
- Funkcija pogreške je zbroj kvadratnih gubitaka (**reziduala**):

$$E(h|\mathcal{D}) = \frac{1}{2} \sum_{i=1}^N (y^{(i)} - h(x^{(i)}))^2$$

- Optimizacijski postupak:

$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} E(h|\mathcal{D}) = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^N (y^{(i)} - h(x^{(i)}))^2$$

- Za jednostavnu regresiju:

$$\nabla_{w_0, w_1} E(h|\mathcal{D}) = 0$$

$$\frac{\partial}{\partial w_0} \left[\frac{1}{2} \sum_i^N (y^{(i)} - (w_1 x^{(i)} + w_0))^2 \right] = 0$$

$$\frac{\partial}{\partial w_1} \left[\frac{1}{2} \sum_i^N (y^{(i)} - (w_1 x^{(i)} + w_0))^2 \right] = 0$$

⋮

$$w_0 = \bar{y} - w_1 \bar{x}$$

$$w_1 = \frac{\sum_i^N x^{(i)} y^{(i)} - N \bar{x} \bar{y}}{\sum_i^N (x^{(i)})^2 - N \bar{x}^2}$$

2 Vrste regresije

- Ulagne varijable: **jednostavna** ($n = 1$) ili **višestruka** ($n > 1$)
- Izlagne varijable: **univarijatna** ($f(\mathbf{x}) = y$) ili **multivarijatna** ($f(\mathbf{x}) = \mathbf{y}$)

	Jedan izlaz	Više izlaza
Jedan ulaz	(Univarijatna) jednostavna	Multivarijatna jednostavna
Više ulaza	(Univarijatna) višestruka	Multivarijatna višestruka

- Mi radimo samo univarijatnu regresiju

3 Tri komponente linearne regresije

- (1) Model:

$$h(\mathbf{x}; \mathbf{w}) = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n = \sum_{i=1}^n w_i x_i + w_0 = h(\mathbf{x}; \mathbf{w}) = \mathbf{w}^T \mathbf{x}$$

- (2) Funkcija gubitka i funkcija pogreške:

$$\begin{aligned} L(y^{(i)}, h(\mathbf{x}^{(i)})) &= (y^{(i)} - h(\mathbf{x}^{(i)}))^2 \\ E(h|\mathcal{D}) &= \frac{1}{2} \sum_{i=1}^N (y^{(i)} - h(\mathbf{x}^{(i)}))^2 \end{aligned}$$

- (3) Optimizacijski postupak:

$$\underset{\mathbf{w}}{\operatorname{argmin}} E(\mathbf{w}|\mathcal{D})$$

⇒ **metoda najmanjih kvadrata** (*ordinary least squares, OLS*)

- Postoji rješenje u **zatvorenoj formi**

4 Postupak najmanjih kvadrata

- Označeni primjeri daju N jednadžbi s $(n+1)$ nepoznanica:

$$\forall (\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D}. \mathbf{w}^T \mathbf{x} = y^{(i)}$$

- Matrično:

$$\underbrace{\begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \dots & x_n^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \dots & x_n^{(2)} \\ \vdots & & & & \\ 1 & x_1^{(N)} & x_2^{(N)} & \dots & x_n^{(N)} \end{pmatrix}}_{\mathbf{X}} \cdot \underbrace{\begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_n \end{pmatrix}}_{\mathbf{w}} = \underbrace{\begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(N)} \end{pmatrix}}_{\mathbf{y}}$$

- Matrica \mathbf{X} je **matrica dizajna**
- Egzaktno rješenje je $\mathbf{w} = \mathbf{X}^{-1}\mathbf{y}$, ali ono ne postoji ako:
 - \mathbf{X} nije kvadratna \Rightarrow pre/pododređenost sustava
 - \mathbf{X} je kvadratna, ali je sustav nekonzistentan
- Umjesto egzaktnog, tražimo približno rješenje (najmanja kvadratna odstupanja)
- Funkcija pogreške u matričnom obliku:

$$\begin{aligned} E(\mathbf{w}|\mathcal{D}) &= \frac{1}{2}(\mathbf{X}\mathbf{w} - \mathbf{y})^T(\mathbf{X}\mathbf{w} - \mathbf{y}) \\ &= \frac{1}{2}(\mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w} - \mathbf{w}^T\mathbf{X}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\mathbf{w} + \mathbf{y}^T\mathbf{y}) \\ &= \frac{1}{2}(\mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w} - 2\mathbf{y}^T\mathbf{X}\mathbf{w} + \mathbf{y}^T\mathbf{y}) \end{aligned}$$

uz $(A^T)^T = A$ i $(AB)^T = B^T A^T$

- Minimizacija:

$$\begin{aligned} \nabla_{\mathbf{w}} E &= \frac{1}{2} \left(\mathbf{w}^T (\mathbf{X}^T \mathbf{X} + (\mathbf{X}^T \mathbf{X})^T) - 2\mathbf{y}^T \mathbf{X} \right) = \mathbf{w}^T \mathbf{X}^T \mathbf{X} - \mathbf{y}^T \mathbf{X} = \mathbf{0} \\ \mathbf{w}^T &= \mathbf{y}^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} \\ \mathbf{w} &= (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = \mathbf{X}^+ \mathbf{y} \end{aligned}$$

uz $\frac{d}{dx} Ax = A$ i $\frac{d}{dx} x^T Ax = x^T(A + A^T)$

- Matrica $\mathbf{X}^+ = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$ je Moore-Penroseov **pseudoinverz** matrice dizajna \mathbf{X}
- Pseudoinverz minimizira normu $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2$
- Ako je \mathbf{X} kvadratna i punog ranga, onda $\mathbf{X}^+ = \mathbf{X}^{-1}$
- $\mathbf{X}^T \mathbf{X}$ je **Gramova matrica**; $\text{rang}(\mathbf{X}^T \mathbf{X}) = \text{rang}(\mathbf{X})$
- Ako je $\text{rang}(\mathbf{X}) = n + 1$, onda $\mathbf{X}^+ = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$
- Dimenzija Gramove matrice je $(n + 1) \times (n + 1) \Rightarrow$ izračun inverza je moguće skup
- Ako $\text{rang}(\mathbf{X}) < n + 1$ (plitka matrica), onda \mathbf{X}^+ računamo pomoću SVD-a

5 Probabilistička interpretacija regresije

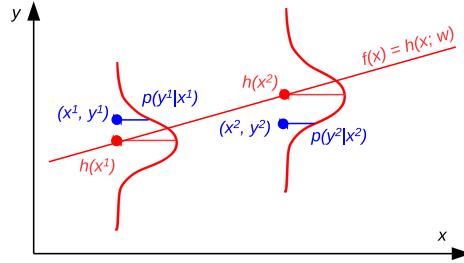
- Opažena oznaka je zbroj vrijednosti funkcije i šuma: $y^{(i)} = f(\mathbf{x}^{(i)}) + \varepsilon_i$
- Šum modeliramo kao normalno distribuiranu **slučajnu varijablu**: $\varepsilon \sim \mathcal{N}(0, \sigma^2)$
- **Normalna razdioba**:

$$p(Y = y | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y - \mu)^2}{2\sigma^2}\right)$$

- Vjerojatnost oznake za zadani primjer: $p(y|\mathbf{x}) = \mathcal{N}(f(\mathbf{x}), \sigma^2)$
- Vjerojatnost da je cijeli skup primjera \mathbf{X} označen oznakama \mathbf{y} (uz pretpostavku **iid**):

$$p(\mathbf{y}|\mathbf{X}) = \prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)})$$

\Rightarrow **izglednost** (*likelihood*) (vjerojatnost oznaka pod modelom)



- Radi matematičke jednostavnosti, radimo s logaritmom izglednosti \Rightarrow **log-izglednost**
- Tražimo \mathbf{w} koji označe čini najvjerojatnijim \Leftrightarrow maksimizacija log-izglednosti
- Vrijedi $h(\mathbf{x}; \mathbf{w}) = f(\mathbf{x})$ (hipoteza treba aproksimirati funkciju $f(\mathbf{x})$)
- Log-izglednost težina \mathbf{w} :

$$\begin{aligned} \ln p(\mathbf{y}|\mathbf{X}, \mathbf{w}) &= \ln \prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}) = \ln \prod_{i=1}^N \mathcal{N}(f(\mathbf{x}^{(i)}), \sigma^2) \\ &= \ln \prod_{i=1}^N \mathcal{N}(h(\mathbf{x}^{(i)}; \mathbf{w}), \sigma^2) \\ &= \ln \prod_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - h(\mathbf{x}^{(i)}; \mathbf{w}))^2}{2\sigma^2}\right) \\ &= \underbrace{-N \ln(\sqrt{2\pi}\sigma)}_{=\text{konst.}} - \frac{1}{2\sigma^2} \sum_{i=1}^N (y^{(i)} - h(\mathbf{x}^{(i)}; \mathbf{w}))^2 \\ &\propto -\frac{1}{2} \sum_{i=1}^N (y^{(i)} - h(\mathbf{x}^{(i)}; \mathbf{w}))^2 \end{aligned}$$

\Rightarrow maksimizacija izglednosti \Leftrightarrow minimizacija pogreške kvadratnog gubitka

4. Regresija II

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

Jan Šnajder, natuknice s predavanja, v1.5

1 Nelinearna regresija

- Veza između nezavisnih varijabli i zavisne varijable često je **nelinearna**
- Neki nelinearni regresijski modeli:
 - Linearna višestruka regresija:

$$h(\mathbf{x}; \mathbf{w}) = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n$$

- Jednostruka polinomijalna regresija ($n = 1$):

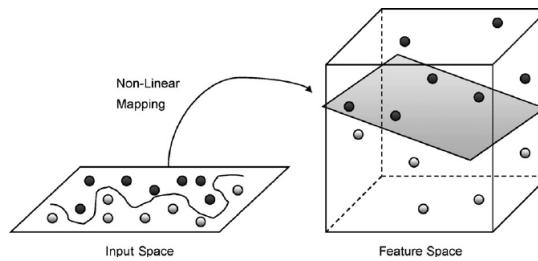
$$h(x; \mathbf{w}) = w_0 + w_1x + w_2x^2 + \cdots + w_dx^d$$

- Višestruka polinomijalna regresija ($n = 2, d = 2$):

$$h(\mathbf{x}; \mathbf{w}) = w_0 + w_1x_1 + w_2x_2 + w_3x_1x_2 + w_4x_1^2 + w_5x_2^2$$

gdje je x_1x_2 **interakcijska značajka** (*cross-term*)

- Umjesto da mijenjamo model, mijenjamo podatke \Rightarrow **preslikavanje u prostor značajki**



- **Bazne funkcije** (nelinearne funkcije ulaznih varijabli):

$$\{\phi_0, \phi_1, \phi_2, \dots, \phi_m\}, \quad \phi_j : \mathbb{R}^n \rightarrow \mathbb{R}, \quad \phi_0(\mathbf{x}) = 1$$

- **Funkcija preslikavanja** u prostor značajki:

$$\begin{aligned}\phi : \mathbb{R}^n &\rightarrow \mathbb{R}^{m+1} : \\ \phi(\mathbf{x}) &= (\phi_0(\mathbf{x}), \dots, \phi_m(\mathbf{x}))\end{aligned}$$

- Model s ugrađenom funkcijom preslikavanja:

$$h(\mathbf{x}; \mathbf{w}) = \sum_{j=0}^m w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$$

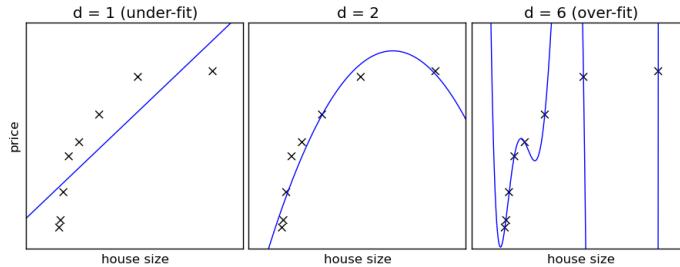
- Ova je **linearan model regresije** (linearan u parametrima) \neq linearna regresija
- Uobičajene funkcije preslikavanja:
 - Linearna višestruka regresija: $\boldsymbol{\phi}(\mathbf{x}) = (1, x_1, x_2, \dots, x_n)$
 - Jednostruka polinomijalna regresija: $\boldsymbol{\phi}(x) = (1, x, x^2, \dots, x^m)$
 - Višestruka polinomijalna regresija drugog stupnja: $\boldsymbol{\phi}(\mathbf{x}) = (1, x_1, x_2, x_1 x_2, x_1^2, x_2^2)$
- Matrica dizajna s preslikavanjem:

$$\boldsymbol{\Phi} = \begin{pmatrix} 1 & \phi_1(\mathbf{x}^{(1)}) & \dots & \phi_m(\mathbf{x}^{(1)}) \\ 1 & \phi_1(\mathbf{x}^{(2)}) & \dots & \phi_m(\mathbf{x}^{(2)}) \\ \vdots & & & \\ 1 & \phi_1(\mathbf{x}^{(N)}) & \dots & \phi_m(\mathbf{x}^{(N)}) \end{pmatrix}_{N \times (m+1)} = \begin{pmatrix} \boldsymbol{\phi}(\mathbf{x}^{(1)})^T \\ \boldsymbol{\phi}(\mathbf{x}^{(2)})^T \\ \vdots \\ \boldsymbol{\phi}(\mathbf{x}^{(N)})^T \end{pmatrix}_{N \times (m+1)}$$

- Rješenje najmanjih kvadrata: $\mathbf{w} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T \mathbf{y} = \boldsymbol{\Phi}^+ \mathbf{y}$

2 Prenaučenost

- Odabir preslikavanja $\boldsymbol{\phi}$ je **hiperparametar** modela
- Nelinearan model je složeniji od linearног \Rightarrow sklonost **prenaučenosti**



- Rješenje: učiti na više primjera, odabir modela, regularizacija, bayesovska regresija

3 Regularizacija

- Složeniji model \Leftrightarrow veće magnitude parametara (težina) \mathbf{w}
- Ograničavanje rasta parametara pri učenju \Rightarrow **regularizacija**
- **Rijetki modeli** (*sparse models*) – modeli s težinama pritegnutima na nulu

- Regularizirana funkcija pogreške:

$$E_R(\mathbf{w}|\mathcal{D}) = E(\mathbf{w}|\mathcal{D}) + \underbrace{\lambda \Omega(\mathbf{w})}_{\text{reg. izraz}}$$

gdje je λ **regularizacijski faktor** \Rightarrow kompromis između jednostavnosti i složenosti

- Regularizacijski izraz je p -norma vektora težina:

$$\Omega(\mathbf{w}) = \|\mathbf{w}\|_p = \left(\sum_{j=1}^m |w_j|^p \right)^{\frac{1}{p}}$$

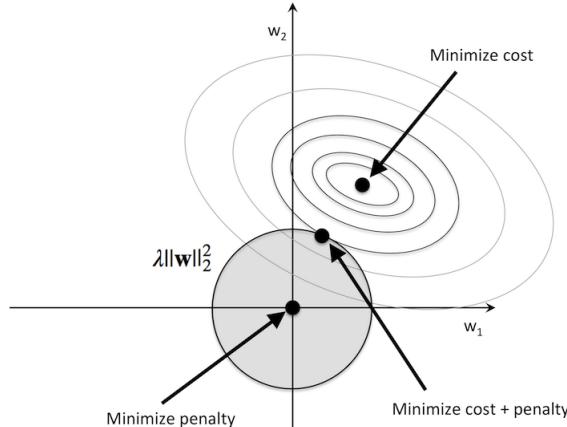
- L2-norma ($p = 2$): $\|\mathbf{w}\|_2 = \sqrt{\sum_{j=1}^m w_j^2} = \sqrt{\mathbf{w}^T \mathbf{w}}$
- L1-norma ($p = 1$): $\|\mathbf{w}\|_1 = \sum_{j=1}^m |w_j|$
- L0-norma ($p = 0$): $\|\mathbf{w}\|_0 = \sum_{j=1}^m \mathbf{1}\{w_j \neq 0\}$

- **L2-regularizacija** (Tikhonovljeva regularizacija):

$$E_R(\mathbf{w}|\mathcal{D}) = \frac{1}{2} \sum_{i=1}^N (\mathbf{w}^T \phi(\mathbf{x}^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2} \|\mathbf{w}\|_2^2$$

\Rightarrow **hrbatna regresija** (*ridge regression*)

- Skica: izokonture L2-regularizirane funkcije pogreške

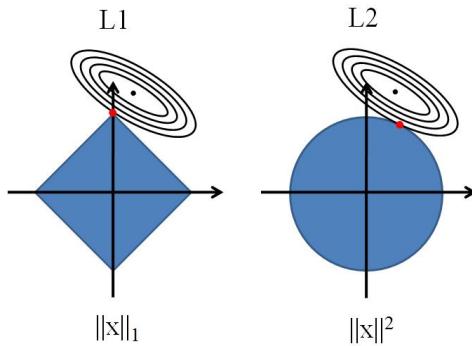


- **L1-regularizacija (LASSO):**

$$E_R(\mathbf{w}|\mathcal{D}) = \frac{1}{2} \sum_{i=1}^N (\mathbf{w}^T \phi(\mathbf{x}^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2} \|\mathbf{w}\|_1$$

\Rightarrow daje rijetke modele

- Skica: usporedba izokontura L1- i L2-regulariziranih funkcija pogreške



- **L0-regularizacija:**

$$E_R(\mathbf{w}|\mathcal{D}) = \frac{1}{2} \sum_{i=1}^N (\mathbf{w}^T \phi(\mathbf{x}^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2} \sum_{j=1}^m \mathbf{1}\{w_j \neq 0\}$$

⇒ efektivno provodi **odabir značajki**

- L0-regularizacija je NP-potpuna, L1-regularizacija nema rješenje u zatvorenoj formi
- Rješenje najmanjih kvadrata s L2-regularizacijom:

$$\begin{aligned} E_R(\mathbf{w}|\mathcal{D}) &= \frac{1}{2} (\Phi \mathbf{w} - \mathbf{y})^T (\Phi \mathbf{w} - \mathbf{y}) + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w} \\ &= \frac{1}{2} (\mathbf{w}^T \Phi^T \Phi \mathbf{w} - 2\mathbf{y}^T \Phi \mathbf{w} + \mathbf{y}^T \mathbf{y} + \lambda \mathbf{w}^T \mathbf{w}) \\ \nabla_{\mathbf{w}} E_R &= \Phi^T \Phi \mathbf{w} - \Phi^T \mathbf{y} + \lambda \mathbf{w} \\ &= (\Phi^T \Phi + \lambda \mathbf{I}) \mathbf{w} - \Phi^T \mathbf{y} = 0 \\ \mathbf{w} &= (\Phi^T \Phi + \lambda \mathbf{I})^{-1} \Phi^T \mathbf{y} \end{aligned}$$

gdje $\lambda \mathbf{I} = \text{diag}(0, \lambda, \dots, \lambda)$ (težinu w_0 ne regulariziramo)

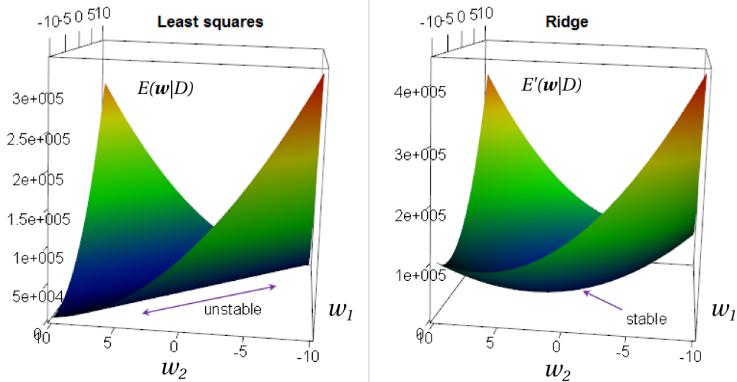
4 Regularizacija i kondicija matrice

- Rješenje najmanjih kvadrata: $\mathbf{w} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$
- $(\Phi^T \Phi)^{-1}$ definiran $\Leftrightarrow \text{rang}(\Phi^T \Phi) = \text{rang}(\Phi) = m + 1 \Leftrightarrow$ linearno nezavisni stupci
- Linearno zavisni stupci \Leftrightarrow redundantne značajke \Leftrightarrow **savršena multikolinearnost**
- **Multikolinearnost** – dvije varijable ili više njih su visoko korelirane
- Multikolinearnost daje **numerički nestabilno rješenje** \Rightarrow **prenaučenost**
- Nestabilnost rješenja iskazuje se **kondicijskim brojem** matrice
- $m \gg N \Leftrightarrow$ “široka i plitka” matrica dizajna $\Rightarrow \text{rang}(\Phi) < m + 1 \Rightarrow$ multikolinearnost
- Regularizacija smanjuje multikolinearnost:

$$\mathbf{w} = (\Phi^T \Phi + \lambda \mathbf{I})^{-1} \Phi^T \mathbf{y}$$

⇒ dodavanje dijagonale smanjuje linearnu zavisnost \Rightarrow **rekondicioniranje matrice**

- Regularizacijom funkcija pogreške postaje konveksnija (nestaje hrbat)



5 Napomene

- Magnituda parametra w_i odgovara **važnosti značajke**, osim ako je model prenaučen
- Regularizacija **sprječava prenaučenost** prigušujući vrijednosti značajki
- Ako je model nelinearan, regularizacijom smanjujemo nelinearnost
- Težinu w_0 treba izuzeti iz regularizacijskog izraza ili treba centrirati podatke
- Odabir hiperparametra λ najčešće se provodi **unakrsnom provjerom**

5. Linearni diskriminativni modeli

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

Jan Šnajder, natuknice s predavanja, v1.9

1 Linearni diskriminativni modeli

- Linearni diskriminativni modeli – granica je linearna \Rightarrow **hiperravnina**:
$$h(\mathbf{x}; \mathbf{w}) = \mathbf{w}^T \mathbf{x}$$
- Granica između klasa je na $h(\mathbf{x}) = 0$ (ponekad: $h(\mathbf{x}) = 0.5$)
- **Diskriminativan model** – izravno modelira granicu između klasa
- Generativni vs. diskriminativni modeli

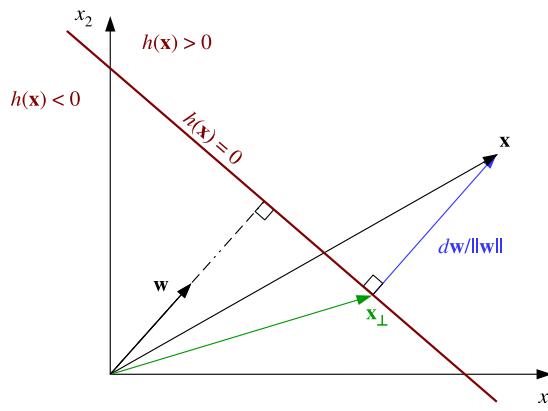
2 Geometrija linearog modela

- BSO, razmatramo sljedeći model:

$$h(\mathbf{x}; \mathbf{w}) = w_1 x_1 + w_2 x_2 + w_0$$

- Granica je pravac:

$$w_1 x_1 + w_2 x_2 + w_0 = 0$$



- \mathbf{w} je **normala** hiperravnine:

$$\begin{aligned} h(\mathbf{x}_1) &= h(\mathbf{x}_2) \\ \mathbf{w}^T \mathbf{x}_1 + w_0 &= \mathbf{w}^T \mathbf{x}_2 + w_0 \\ \mathbf{w}^T (\mathbf{x}_1 - \mathbf{x}_2) + w_0 - w_0 &= 0 \\ \mathbf{w}^T (\mathbf{x}_1 - \mathbf{x}_2) &= 0 \end{aligned}$$

- Udaljenost primjera \mathbf{x} od hiperravnine:

$$\begin{aligned} \mathbf{x} &= \mathbf{x}_{\perp} + d \frac{\mathbf{w}}{\|\mathbf{w}\|} \\ &= \underbrace{\mathbf{w}^T \mathbf{x} + w_0}_{h(\mathbf{x})} = \underbrace{\mathbf{w}^T \mathbf{x}_{\perp} + w_0}_{=h(\mathbf{x}_{\perp})=0} + d \frac{\mathbf{w}^T \mathbf{w}}{\|\mathbf{w}\|} \\ h(\mathbf{x}) &= d \|\mathbf{w}\| \quad \Rightarrow d = \frac{h(\mathbf{x})}{\|\mathbf{w}\|} \end{aligned}$$

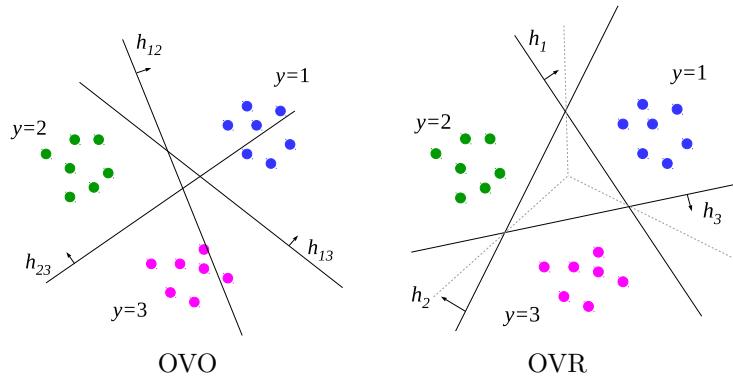
3 Višeklasna klasifikacija

- Shema **jedan-naspram-jedan** (**one-vs-one**, **OVO**) – $\binom{K}{2}$ binarnih modela:

$$h(\mathbf{x}) = \operatorname{argmax}_i \sum_{i \neq j} \operatorname{sgn}(h_{ij}(\mathbf{x})), \quad h_{ji}(\mathbf{x}) = -h_{ij}(\mathbf{x})$$

- Shema **jedan-naspram-ostali** (**one-vs-rest**, **OVR**) – K binarnih modela:

$$h(\mathbf{x}) = \operatorname{argmax}_j h_j(\mathbf{x})$$



- OVR ima manje modela od OVO, ali potencira neuravnotežnost klase

4 Klasifikacija regresijom

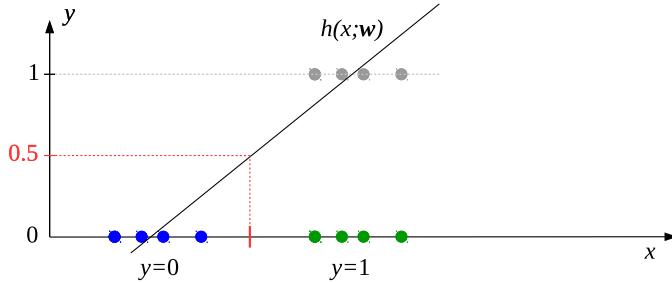
- Funkcija pogreške:

$$E(\mathbf{w}|\mathcal{D}) = \frac{1}{2} \sum_{i=1}^N (\mathbf{w}^T \phi(\mathbf{x}^{(i)}) - y^{(i)})^2 = \frac{1}{2} (\Phi \mathbf{w} - \mathbf{y})^T (\Phi \mathbf{w} - \mathbf{y})$$

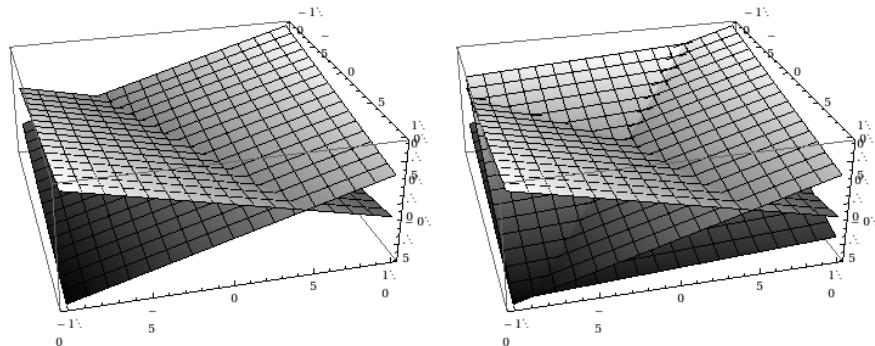
- Minimizator:

$$\mathbf{w}^* = (\Phi^\top \Phi)^{-1} \Phi^\top \mathbf{y} = \Phi^+ \mathbf{y}$$

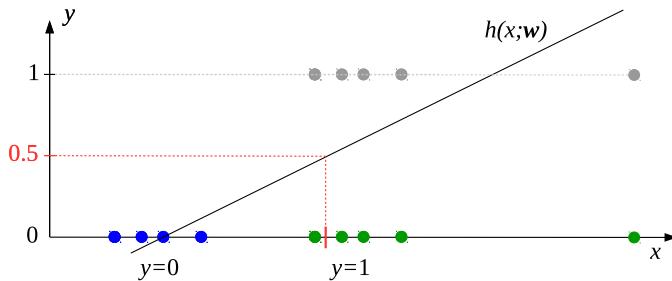
- Ideja: hipoteza koja predviđa $y = 1$ i $y = 0$ za primjere prve odnosno druge klase
- Model: $h(\mathbf{x}; \mathbf{w}) = \mathbf{1}\{\mathbf{w}^\top \phi(\mathbf{x}) \geq 0.5\}$
- Skica za $n = 1$:



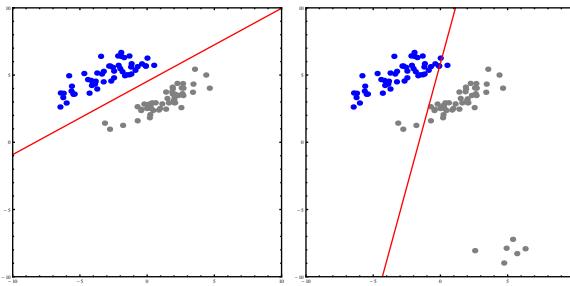
- Proširivo na $K > 2$ klase shemom OVR ili OVO



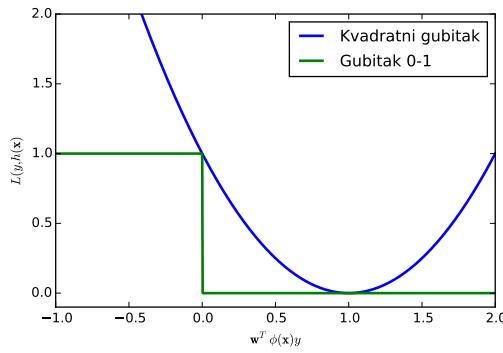
- Nedostatci: izlazi nisu vjerojatnosti, nerobusnost na vrijednosti koje odskaču
- Skica: nerobusnost na vrijednosti koje odskaču ($n = 1$):



- Nerobusnost na vrijednosti koje odskaču ($n = 2$):



- Uzrok: funkcija gubitka L kažnjava i dobro klasificirane primjere
- Za $y \in \{-1, +1\}$: ispravna klasifikacija $\Leftrightarrow \mathbf{w}^T \phi(\mathbf{x})y > 0$
- Skica: L kao funkcija od $\mathbf{w}^T \phi(\mathbf{x})y$



- Idealan gubitak je gubitak 0-1, ali nije konveksan, pa nije pogodan za optimizaciju

5 Perceptron

- Model:
- $$h(\mathbf{x}) = f(\mathbf{w}^T \phi(\mathbf{x}))$$

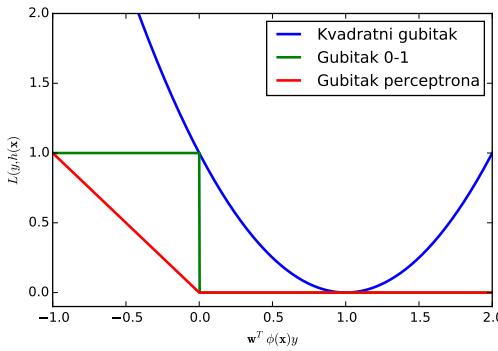
- **Funkcija praga** kao aktivacijska funkcija:

$$f(\alpha) = \begin{cases} +1 & \text{ako } \alpha \geq 0 \\ -1 & \text{inače} \end{cases}$$

- Perceptron – umjetni neuron (McCulloch & Pitts, 1943.)

- Funkcija gubitka:

$$L(y, h(\mathbf{x})) = \max(0, -\mathbf{w}^T \phi(\mathbf{x})y)$$

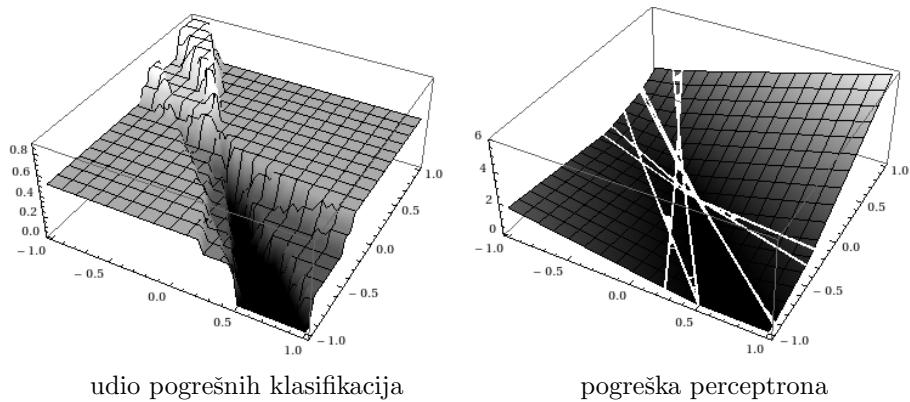


- Funkcija pogreške:

$$E(\mathbf{w}|\mathcal{D}) = - \sum_{i : f(\mathbf{w}^T \phi(\mathbf{x}^{(i)})) \neq y^{(i)}} \mathbf{w}^T \phi(\mathbf{x}^{(i)}) y^{(i)} = \sum_{i=1}^N \max(0, -\mathbf{w}^T \phi(\mathbf{x}^{(i)}) y^{(i)})$$

⇒ kažnjava samo netočno klasificirane primjere (za razliku od regresije)

- Površina pogreške u prostoru parametara:



- Ne postoji minimizator u zatvorenoj formi ⇒ primjenjujemo **gradijentni spust**
- Gradijentni spust: težine ažuriramo u smjeru suprotnome od gradijenta
- Gradijent funkcije gubitka za netočno klasificirane primjere:

$$\nabla_{\mathbf{w}} L = \nabla_{\mathbf{w}} (-\mathbf{w}^T \phi(\mathbf{x}) y) = -\phi(\mathbf{x}) y$$

- Ažuriranja težina – **Widrow-Hoffovo pravilo**:

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla L(\mathbf{w}|\mathcal{D})$$

tj.

$$\mathbf{w} \leftarrow \mathbf{w} + \eta \phi(\mathbf{x}) y$$

gdje je η stopa učenja

Algoritam perceptron-a

```
1: inicijaliziraj  $\mathbf{w} \leftarrow (0, \dots, 0)$ 
2: ponavljam do konvergencije
3:   za  $i = 1, \dots, N$ 
4:     ako  $f(\mathbf{w}^T \phi(\mathbf{x}^{(i)})) \neq y^{(i)}$  onda  $\mathbf{w} \leftarrow \mathbf{w} + \eta \phi(\mathbf{x}^{(i)}) y^{(i)}$ 
```

- Nedostatci:
 - Izlazi modela nisu vjerojatnosti
 - Konvergira samo ako su primjeri linearno odvojivi (Rosenblatt, 1962)
 - Rezultat ovisi o početnim težinama

6. Logistička regresija

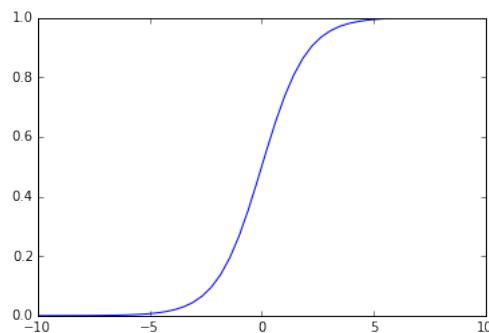
Strojno učenje 1, UNIZG FER, ak. god. 2021./2022.

Jan Šnajder, natuknice s predavanja, v1.5

1 Model logističke regresije

- **Logistička (sigmoidalna) funkcija:**

$$\sigma(\alpha) = \frac{1}{1 + \exp(-\alpha)}$$



- Funkcija je derivabilna:

$$\frac{\partial \sigma(\alpha)}{\partial \alpha} = \frac{\partial}{\partial \alpha} (1 + \exp(-\alpha))^{-1} = \sigma(\alpha)(1 - \sigma(\alpha))$$

- Model logističke regresije:

$$h(\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T \phi(\mathbf{x})) = \frac{1}{1 + \exp(-\mathbf{w}^T \phi(\mathbf{x}))} = P(y = 1 | \mathbf{x})$$

⇒ izlaz modela možemo tumačiti kao vjerojatnost da primjer pripada klasi $y = 1$

- Ovo je primjer **poopćenog linearog modela** (*generalized linear model, GLM*)
- GLM – linearni modeli s (nelinarnom) **aktivacijskom funkcijom** f :

$$h(\mathbf{x}; \mathbf{w}) = f(\mathbf{w}^T \phi(\mathbf{x}))$$

gdje je $f : \mathbb{R} \rightarrow [0, 1]$ ili $f : \mathbb{R} \rightarrow (0, 1)$ ili ($f : \mathbb{R} \rightarrow [-1, +1]$ ili $f : \mathbb{R} \rightarrow (-1, +1)$)

2 Pogreška unakrsne entropije

- Izlaz modela je **Bernoullijeva varijabla**:

$$P(y|\mu) = \begin{cases} \mu & \text{ako } y = 1 \\ 1 - \mu & \text{inače} \end{cases} = \mu^y(1 - \mu)^{1-y}$$

- U našem slučaju, y je oznaka primjera, a μ je izlaz modela, tj. $\mu = h(\mathbf{x}; \mathbf{w})$, pa:

$$P(y^{(i)}|\mathbf{x}^{(i)}) = h(\mathbf{x}; \mathbf{w})^y(1 - h(\mathbf{x}; \mathbf{w}))^{1-y}$$

- Log-izglednost oznaka iz skupa označenih primjera:

$$\begin{aligned} \ln P(\mathbf{y}|\mathbf{X}, \mathbf{w}) &= \ln \prod_{i=1}^N p(y^{(i)}|\mathbf{x}^{(i)}) = \\ &= \sum_{i=1}^N \left(y^{(i)} \ln h(\mathbf{x}^{(i)}; \mathbf{w}) + (1 - y^{(i)}) \ln (1 - h(\mathbf{x}^{(i)}; \mathbf{w})) \right) \end{aligned}$$

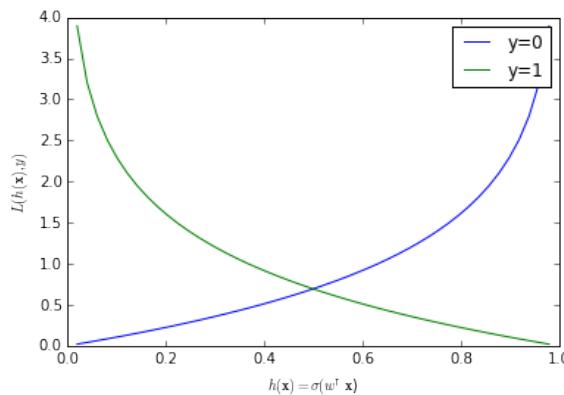
- Empirijska pogreška je negativna log-izglednost:

$$E(\mathbf{w}|\mathcal{D}) = \frac{1}{N} \sum_{i=1}^N \left(-y^{(i)} \ln h(\mathbf{x}^{(i)}; \mathbf{w}) - (1 - y^{(i)}) \ln (1 - h(\mathbf{x}^{(i)}; \mathbf{w})) \right)$$

⇒ **pogreška unakrsne entropije (cross-entropy error)**

- Gubitak unakrsne entropije (cross-entropy loss):**

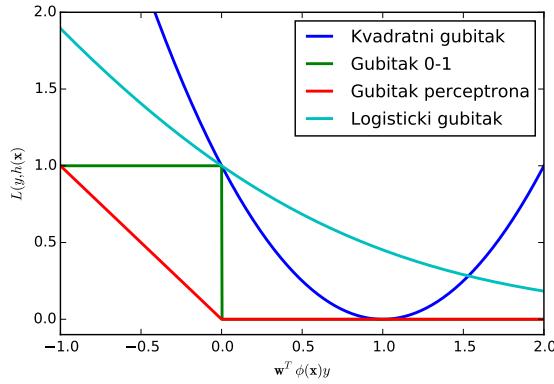
$$L(y, h(\mathbf{x})) = -y \ln h(\mathbf{x}) - (1 - y) \ln (1 - h(\mathbf{x}))$$



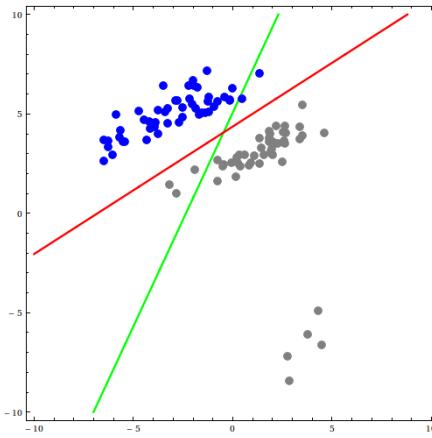
- Reformulacija $y \in \{0, 1\} \rightarrow y \in \{-1, +1\}$ i skaliranje sa $1/\ln 2$:

$$L(y, h(\mathbf{x})) = \frac{1}{\ln 2} \ln (1 + \exp(-y \mathbf{w}^T \phi(\mathbf{x})))$$

- Usporedba funkcija gubitaka:



- Logistička regresija robusnija je od modela linearne regresije:



- Minimizacija u zatvorenoj formi nije moguća \Rightarrow iterativna optimizacija

3 Gradijentni spust

- **Gradijentni spust** – minimum nalazimo krećući se u smjeru suprotnom od gradijenta:

$$\mathbf{x} \leftarrow \mathbf{x} - \eta \nabla f(\mathbf{x})$$

- η je **stopa učenja**: prevelika $\eta \Rightarrow$ divergencija; premalena $\eta \Rightarrow$ spora konvergencija
- Želimo **globalnu konvergenciju** (konvergencija uvijek i svugdje)
- Ostvarivo **linijskim pretraživanjem** – η koji minimizira $f(\mathbf{x})$ u smjeru spusta $\Delta\mathbf{x}$:

$$g(\eta) = f(\mathbf{x} + \eta \Delta\mathbf{x})$$

- Pronađeni optimum bit će globalni optimum ako je $f(\mathbf{x})$ **konveksna**
- Funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ je **konveksna** akko

(1) Njezina domena $\text{dom}(f)$ je **konveksni skup**:

Za svaki $\mathbf{x}_1, \dots, \mathbf{x}_n \in \text{dom}(f)$ i za svaki $\alpha_1, \dots, \alpha_n$ takav da $\sum_i \alpha_i = 1$ vrijedi:

$$\sum_{i=1}^n \alpha_i \mathbf{x}_i \in \text{dom}(f)$$

(2) Za svaki $\mathbf{x}_1, \mathbf{x}_2 \in \text{dom}(f)$ i svaki $\alpha \in [0, 1]$ vrijedi:

$$f(\mathbf{x}) = f(\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2) \leq \alpha f(\mathbf{x}_1) + (1 - \alpha) f(\mathbf{x}_2)$$

- Empirijska pogreška je konveksna \Leftrightarrow funkcija gubitka L je konveksna
- Dvije varijante gradijentnog spusta:
 - **Batch** (grupni): $\mathbf{w} \leftarrow \mathbf{w} - \eta \sum_{i=1}^N \nabla L(y^{(i)}, h(\mathbf{x}^{(i)}; \mathbf{w}))$
 - **Stohastički (SGD)**: $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla L(y^{(i)}, h(\mathbf{x}^{(i)}; \mathbf{w}))$
- SGD je pogodan za on-line učenje (big data, data streams)

4 Gradijentni spust za logističku regresiju

- Gradijent funkcije gubitka i funkcije pogreške:

$$\begin{aligned} E(\mathbf{w}|\mathcal{D}) &= \frac{1}{N} \sum_{i=1}^N \left(-y^{(i)} \ln h(\mathbf{x}^{(i)}; \mathbf{w}) - (1 - y^{(i)}) \ln (1 - h(\mathbf{x}^{(i)}; \mathbf{w})) \right) \\ \nabla_{\mathbf{w}} E(\mathbf{w}|\mathcal{D}) &= \frac{1}{N} \sum_{i=1}^N \nabla L(y^{(i)}, h(\mathbf{x}^{(i)}; \mathbf{w})) \\ \nabla L(y, h(\mathbf{x})) &= \left(-\frac{y}{h(\mathbf{x})} + \frac{1-y}{1-h(\mathbf{x})} \right) h(\mathbf{x})(1-h(\mathbf{x})) \phi(\mathbf{x}) = (h(\mathbf{x}) - y) \phi(\mathbf{x}) \\ \nabla E(\mathbf{w}|\mathcal{D}) &= \sum_{i=1}^N (h(\mathbf{x}^{(i)}) - y^{(i)}) \phi(\mathbf{x}^{(i)}) \end{aligned}$$

(faktor $1/N$ može se apsorbirati u stopu učenja η)

Logistička regresija (grupni gradijentni spust)

- ```

1: $\mathbf{w} \leftarrow (0, 0, \dots, 0)$
2: ponavljam do konvergencije
3: $\Delta \mathbf{w} \leftarrow (0, 0, \dots, 0)$
4: za $i = 1, \dots, N$
5: $h \leftarrow \sigma(\mathbf{w}^T \phi(\mathbf{x}^{(i)}))$
6: $\Delta \mathbf{w} \leftarrow \Delta \mathbf{w} - (h - y^{(i)}) \phi(\mathbf{x}^{(i)})$
7: $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru spusta $\Delta \mathbf{w}$
8: $\mathbf{w} \leftarrow \mathbf{w} + \eta \Delta \mathbf{w}$

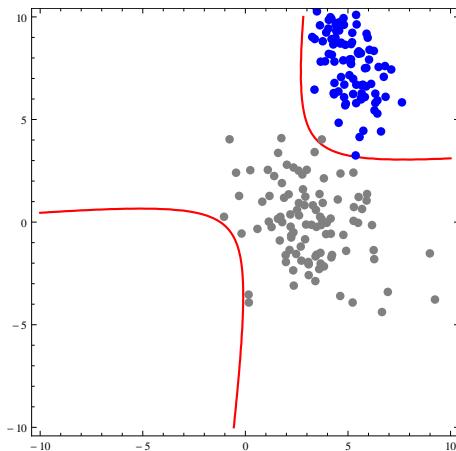
```

### Logistička regresija (stohastički gradijentni spust)

- 1:  $\mathbf{w} \leftarrow (0, 0, \dots, 0)$
- 2: **ponavlja** do konvergencije
- 3: slučajno permutiraj primjere u  $\mathcal{D}$
- 4: **za**  $i = 1, \dots, N$
- 5:  $h \leftarrow \sigma(\mathbf{w}^T \phi(\mathbf{x}^{(i)}))$
- 6:  $\Delta \mathbf{w} \leftarrow -(h - y^{(i)}) \phi(\mathbf{x}^{(i)})$
- 7:  $\eta \leftarrow$  optimum linijskim pretraživanjem u smjeru spusta  $\Delta \mathbf{w}$
- 8:  $\mathbf{w} \leftarrow \mathbf{w} + \eta \Delta \mathbf{w}$

## 5 Regularizirana regresija

- Prednosti regularizacije:
  - Sprječavanje pretjerane nelinearnosti
  - Suzbijanje nepotrebnih značajki
  - Sprječavanje otvrdnjavanja sigmoide kod linearne odvojivih problema
- Primjer prenaučenosti ( $n = 2$ ,  $\phi(\mathbf{x}) = (1, x_1, x_2, x_1x_2, x_1^2, x_2^2)$ ):



- L2-regularizirana pogreška:

$$E_R(\mathbf{w} | \mathcal{D}) = \sum_{i=1}^N \left( -y^{(i)} \ln h(\mathbf{x}^{(i)}) - (1 - y^{(i)}) \ln (1 - h(\mathbf{x}^{(i)})) \right) + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}$$

- Ažuriranje težina:

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \left( \sum_{i=1}^N (h(\mathbf{x}^{(i)}) - y^{(i)}) \phi(\mathbf{x}^{(i)}) + \lambda \mathbf{w} \right)$$

ekvivalentno:

$$\mathbf{w} \leftarrow \mathbf{w} (1 - \eta \lambda) - \eta \sum_{i=1}^N (h(\mathbf{x}^{(i)}) - y^{(i)}) \phi(\mathbf{x}^{(i)})$$

- Napomena: Težina  $w_0$  se ne regularizira

### L2-regularizirana logistička regresija (grupni gradijentni spust)

```

1: $\tilde{\mathbf{w}} \leftarrow (0, 0, \dots, 0)$ // $\tilde{\mathbf{w}}$ je prošireni vektor (w_0, \mathbf{w})
2: ponavljam do konvergencije
3: $\Delta w_0 \leftarrow 0$
4: $\Delta \mathbf{w} \leftarrow (0, 0, \dots, 0)$
5: za $i = 1, \dots, N$
6: $h \leftarrow \sigma(\tilde{\mathbf{w}}^T \phi(\tilde{\mathbf{x}}^{(i)}))$
7: $\Delta w_0 \leftarrow \Delta w_0 - (h - y^{(i)})$
8: $\Delta \mathbf{w} \leftarrow \Delta \mathbf{w} - (h - y^{(i)}) \phi(\mathbf{x}^{(i)})$
9: $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru spusta $\Delta \tilde{\mathbf{w}}$
10: $w_0 \leftarrow w_0 + \eta \Delta w_0$
11: $\mathbf{w} \leftarrow \mathbf{w}(1 - \eta \lambda) + \eta \Delta \mathbf{w}$

```

### L2-regularizirana logistička regresija (stohastički gradijentni spust)

```

1: $\tilde{\mathbf{w}} \leftarrow (0, 0, \dots, 0)$ // $\tilde{\mathbf{w}}$ je prošireni vektor (w_0, \mathbf{w})
2: ponavljam do konvergencije:
3: slučajno permutiraj primjere u \mathcal{D}
4: za $i = 1, \dots, N$
5: $h \leftarrow \sigma(\tilde{\mathbf{w}}^T \phi(\tilde{\mathbf{x}}^{(i)}))$
6: $\Delta w_0 \leftarrow -(h - y^{(i)})$
7: $\Delta \mathbf{w} \leftarrow -(h - y^{(i)}) \phi(\mathbf{x}^{(i)})$
8: $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru spusta $\Delta \tilde{\mathbf{w}}$
9: $w_0 \leftarrow w_0 + \eta \Delta w_0$
10: $\mathbf{w} \leftarrow \mathbf{w}(1 - \eta \lambda) + \eta \Delta \mathbf{w}$

```

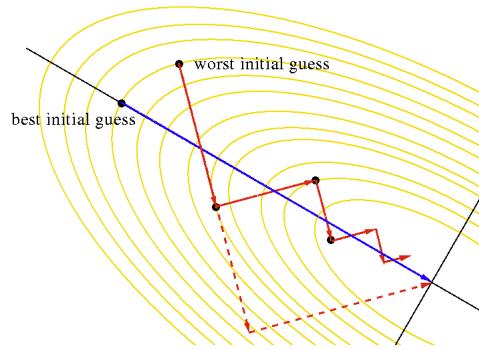
## 7. Logistička regresija II

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

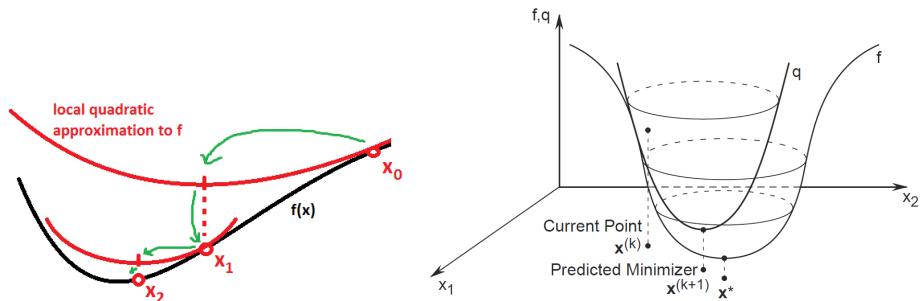
Jan Šnajder, natuknice s predavanja, v1.5

### 1 Alternative gradijentnom spustu

- Gradijentni spust s linijskim pretraživanjem ima cik-cak trajektoriju  $\Rightarrow$  sporo



- Alternativa: **optimizacija drugog reda**, npr. **Newtonov postupak**
- Ideja: skok iz trenutačnog minimuma do minimuma kvadratne aproks. funkcije



- Kvadratna aproksimacija  $f(\mathbf{x})$  u točki  $\mathbf{x}_t$  razvojem u **Taylorov red** drugog reda:

$$f(\mathbf{x}) \approx f_{\text{quad}}(\mathbf{x}) = f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^T (\mathbf{x} - \mathbf{x}_t) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_t)^T \mathbf{H}_t (\mathbf{x} - \mathbf{x}_t)$$

gdje je  $\mathbf{H}_t$  **Hesseova matrica** funkcije  $f(\mathbf{x})$  u točki  $\mathbf{x}_t$

$$\mathbf{H} = \nabla \nabla f(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

- $f(\mathbf{x})$  je konveksna  $\Leftrightarrow \mathbf{H}$  je pozitivno semi-definitna (ali ne nužno pozitivno definitna!)

- Ažuriranje parametara:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \mathbf{H}_t^{-1} \nabla f(\mathbf{x}_t)$$

- Ne radi ako  $\mathbf{H}$  nije invertibilna  $\Rightarrow$  multikolinearnost  $\Rightarrow$  treba regularizirati
- Specifično, za logističku regresiju:

$$\mathbf{H} = \Phi^T \mathbf{S} \Phi$$

gdje  $\mathbf{S} = \text{diag}(h(\mathbf{x}^{(i)})(1 - h(\mathbf{x}^{(i)})))$

- Pravilo ažuriranja:

$$\mathbf{w} \leftarrow \mathbf{w} - \mathbf{H}^{-1} \nabla E(\mathbf{w} | \mathcal{D}) \quad (\eta = 1)$$

$\Rightarrow$  algoritam **iteratively reweighted least squares (IRLS)**

- Izračun  $\mathbf{H}_t$  u svakom koraku je potencijalno skup
- Alternativa: **kvazi-Newtonovi postupci** (BFSG, L-BFSG) – aproksimiraju  $\mathbf{H}_t$
- Uključivanje L2-regularizacije je jednostavno:

$$\nabla E_R(\mathbf{w} | \mathcal{D}) = \nabla E(\mathbf{w} | \mathcal{D}) + \lambda \mathbf{w}$$

$$\mathbf{H}_R = \mathbf{H} + \lambda I$$

- L1-regularizacija: **podgradijentne metode** (koordinatni spust, proksimalne metode)

## 2 Višeklasna logistička regresija

- OVO/OVR ne daje vjerojatnosnu distribuciju po klasama
- **Funkcija softmax:**  $\text{softmax} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ , gdje za komponentu  $k$  vrijedi:

$$\text{softmax}_k(x_1, \dots, x_n) = \frac{\exp(x_k)}{\sum_j \exp(x_j)}$$

$\Rightarrow$  normalizira tako da  $\sum x_k = 1$  te smanjuje male i povećava velike vrijednosti

- **Multinomijalna logistička regresija (MNR, maximum entropy classifier):**

$$h_k(\mathbf{x}; \mathbf{W}) = \frac{\exp(\mathbf{w}_k^T \phi(\mathbf{x}))}{\sum_j \exp(\mathbf{w}_j^T \phi(\mathbf{x}))} = P(y = k | \mathbf{x}, \mathbf{W})$$

gdje  $\mathbf{W} = (\mathbf{w}_1, \dots, \mathbf{w}_K)$

- Izlaz je **multinulijeva** (kategorička) varijabla  $\mathbf{y} = (y_1, y_2, \dots, y_K)^T$ , s distribucijom:

$$P(\mathbf{y}|\boldsymbol{\mu}) = \prod_{k=1}^K \mu_k^{y_k}$$

- Log-izglednost označenih primjera:

$$\begin{aligned} \ln P(\mathbf{y}|\mathbf{X}, \mathbf{W}) &= \ln \prod_{i=1}^N P(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}) = \ln \prod_{i=1}^N \prod_{k=1}^K \mu_k^{y_k^{(i)}} = \ln \prod_{i=1}^N \prod_{k=1}^K h_k(\mathbf{x}^{(i)}; \mathbf{W})^{y_k^{(i)}} \\ &= \sum_{i=1}^N \sum_{k=1}^K y_k^{(i)} \ln h_k(\mathbf{x}^{(i)}; \mathbf{W}) \end{aligned}$$

$\Rightarrow$  poopćena **pogreška unakrsne entropije**:

$$E(\mathbf{W}|\mathcal{D}) = - \sum_{i=1}^N \sum_{k=1}^K y_k^{(i)} \ln h_k(\mathbf{x}^{(i)}; \mathbf{W})$$

- Funkcija gubitka:

$$L(\mathbf{y}, h_k(\mathbf{x})) = - \sum_{k=1}^K y_k \ln h_k(\mathbf{x}; \mathbf{W})$$

- Gradijent za klasu  $k$ :

$$\nabla_{\mathbf{w}_k} E(\mathbf{W}|\mathcal{D}) = \sum_{i=1}^N (h_k(\mathbf{x}^{(i)}; \mathbf{W}) - y_k^{(i)}) \phi(\mathbf{x}^{(i)})$$

$\Rightarrow$  gradijent je isti kao i za binarnu logističku funkciju

- On-line ažuriranje:

$$\mathbf{w}_k \leftarrow \mathbf{w} - \eta (h(\mathbf{x}^{(i)}; \mathbf{w}) - y^{(i)}) \phi(\mathbf{x}^{(i)})$$

$\Rightarrow$  algoritam **least-mean-squares (LMS)** ili **Widrow-Hoffovo pravilo**

- Isto dobivamo za on-line optimizaciju linearne regresije

### 3 Poopćeni linearni modeli i eksponencijalna familija

- Unificirani pogled na tri poopćena linearna modela koja smo razmatrali
- Linearna regresija:

$$h(\mathbf{x}; \mathbf{w}) = \mathbf{w}^T \phi(\mathbf{x})$$

$$P(y|\mathbf{x}, \mathbf{w}) = \mathcal{N}(\mu, \sigma^2) = \mathcal{N}(h(\mathbf{x}), \sigma^2)$$

$$L(y, h(\mathbf{x})) = (h(\mathbf{x}) - y)^2$$

$$\nabla_{\mathbf{w}} L(y, h(\mathbf{x})) = (h(\mathbf{x}) - y) \phi(\mathbf{x})$$

- Logistička regresija:

$$h(\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T \phi(\mathbf{x})) = \frac{1}{1 + \exp(-\mathbf{w}^T \phi(\mathbf{x}))} = P(y = 1 | \mathbf{x}, \mathbf{w})$$

$$P(y|\mathbf{x}, \mathbf{w}) = \mu^y (1 - \mu)^{(1-y)} = h(\mathbf{x})^y (1 - h(\mathbf{x}))^{(1-y)}$$

$$L(y, h(\mathbf{x})) = -y \ln h(\mathbf{x}) - (1 - y) \ln (1 - h(\mathbf{x}))$$

$$\nabla_{\mathbf{w}} L(y, h(\mathbf{x})) = (h(\mathbf{x}) - y) \phi(\mathbf{x})$$

- Multinomijalna logistička regresija:

$$h_k(\mathbf{x}; \mathbf{W}) = \text{softmax}(\mathbf{w}^T \phi(\mathbf{x})) = \frac{\exp(\mathbf{w}_k^T \phi(\mathbf{x}))}{\sum_j \exp(\mathbf{w}_j^T \phi(\mathbf{x}))} = P(y = k | \mathbf{x}, \mathbf{w})$$

$$P(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \prod_{k=1}^K \mu_k^{y_k} = \prod_{k=1}^K h_k(\mathbf{x})^{y_k}$$

$$L(\mathbf{y}, h_k(\mathbf{x})) = - \sum_{k=1}^K y_k \ln h_k(\mathbf{x}; \mathbf{W})$$

$$\nabla_{\mathbf{w}_k} L(y_k, h_k(\mathbf{x})) = (h_k(\mathbf{x}) - y_k) \phi(\mathbf{x})$$

- Sve tri korištene distribucije pripadaju **eksponencijalnoj familiji distribucija**:

$$p(\mathbf{x}|\boldsymbol{\theta}) = h(\mathbf{x}) \exp(\boldsymbol{\theta}^T \phi(\mathbf{x}) - A(\boldsymbol{\theta}))$$

- Ključno za poopćene linearne modele – distribucija određuje aktivacijsku funkciju:
  - Gauss  $\leftrightarrow$  funkcija identiteta, Bernoulli  $\leftrightarrow$  logistička, Multinoulli  $\leftrightarrow$  softmax

## 4 Adaptivne bazne funkcije

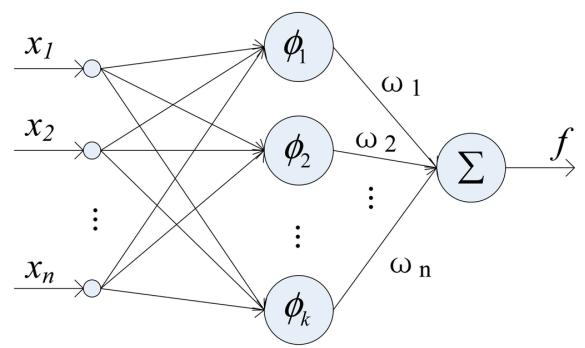
- Model s baznim funkcijama:

$$h(\mathbf{x}; \mathbf{w}) = f(\mathbf{w}^T \phi(\mathbf{x})) = f\left(\sum_{j=0}^m w_j \phi_j(\mathbf{x})\right)$$

- Fiksne (u obliku i broju) adaptivne funkcije mogu biti ograničavajuće
- **Parametrizirane bazne funkcije** – svaka bazna funkcija je poopćeni linearan model:

$$h(\mathbf{x}; \mathbf{w}) = f\left(\sum_{j=0}^m w_j^{(2)} \underbrace{f\left(\sum_{i=0}^n w_{ji}^{(1)} x_i\right)}_{=\phi_j(\mathbf{x})}\right) = f(\mathbf{w}^{(2)\top} f(\mathbf{W}^{(1)} \mathbf{x}))$$

- Dobili smo dvoslojnju **neuronsku mrežu**



- Složeniji model, ali ga je lakše pretrenirati te optimizacija nije konveksna

## 8. Stroj potpornih vektora

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

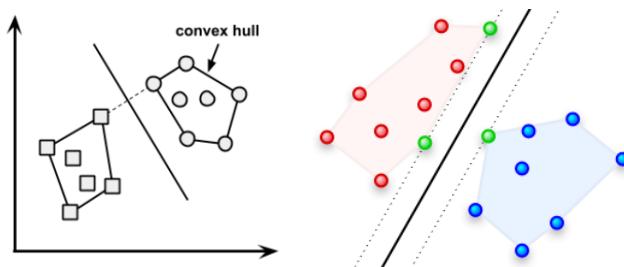
Jan Šnajder, natuknice s predavanja, v2.2

### 1 Problem maksimalne margine

- SVM je **linearan model**:

$$h(\mathbf{x}; \mathbf{w}, w_0) = \mathbf{w}^T \mathbf{x} + w_0$$

- Za nelinearnost možemo upotrijebiti preslikavanje  $\phi$
- **Margina** – udaljenost između hiperravnine i najblžeg primjera
- SVM nalazi **maksimalnu marginu**  $\Rightarrow$  dobra **generalizacija**
- Geometrijski: hiperravnina je simetrala spojice **konveksnih ljesaka** dviju klasa



- Uz pretpostavku **linearne odvojivosti** i uz  $y \in \{-1, +1\}$ , vrijedi:

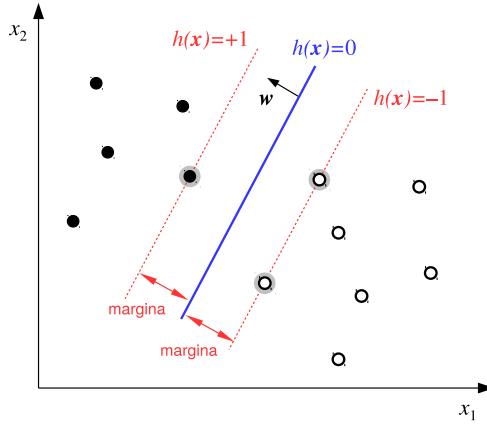
$$\forall (\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D}. y^{(i)} h(\mathbf{x}^{(i)}) \geq 0$$

- Udaljenost primjera  $\mathbf{x}^{(i)}$  od hiperravnine je  $\frac{1}{\|\mathbf{w}\|} |y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0)|$
- Tražimo hiperravninu maksimalne margine:

$$\underset{\mathbf{w}, w_0}{\operatorname{argmax}} \left\{ \frac{1}{\|\mathbf{w}\|} \min_i \{y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0)\} \right\}$$

- Vektor  $(\mathbf{w}, w_0)$  možemo skalirati tako da za primjere najbliže margini vrijedi:

$$y^{(i)}(\mathbf{w}^T \mathbf{x} + w_0) = 1$$



- Onda za sve primjere vrijedi:

$$y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq 1, \quad i = 1, \dots, N$$

- Optimizacijski problem svodi se na:

$$\underset{\mathbf{w}, w_0}{\operatorname{argmax}} \frac{1}{\|\mathbf{w}\|}$$

uz ograničenja:

$$y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq 1, \quad i = 1, \dots, N$$

- Ekvivalentno:

$$\underset{\mathbf{w}, w_0}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{w}\|^2$$

uz ograničenja:

$$y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq 1, \quad i = 1, \dots, N$$

⇒ konveksna optimizacija uz ograničenje, preciznije **kvadratno programiranje**

## 2 Kvadratno programiranje

- Standardni oblik optimizacijskog problema uz ograničenja:

$$\begin{aligned} & \text{minimizirati} && f(\mathbf{x}) \\ & \text{uz ograničenja} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & && h_i(\mathbf{x}) = 0, \quad i = 1, \dots, p \end{aligned}$$

- $f : \mathbb{R}^n \rightarrow \mathbb{R}$  – ciljna funkcija
- $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$  – ograničenja jednakosti
- $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$  – ograničenja nejednakosti

- Rješivo raznim metodama; mi radimo **Lagrangeovu dualnost + algoritam SMO**
- Omogućava optimizaciju u **dualnoj formi** ⇒ SMO, potporni vektori, jezgreni trik

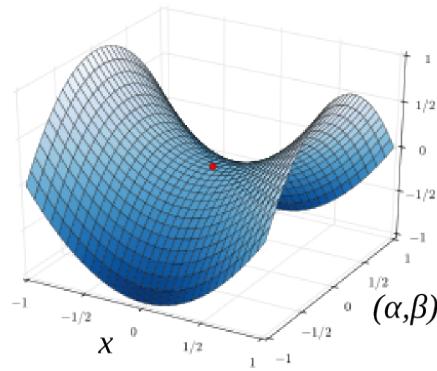
### 3 Lagrangeova dualnost

- Ograničenja kodiramo u **Lagrangeovu funkciju**:

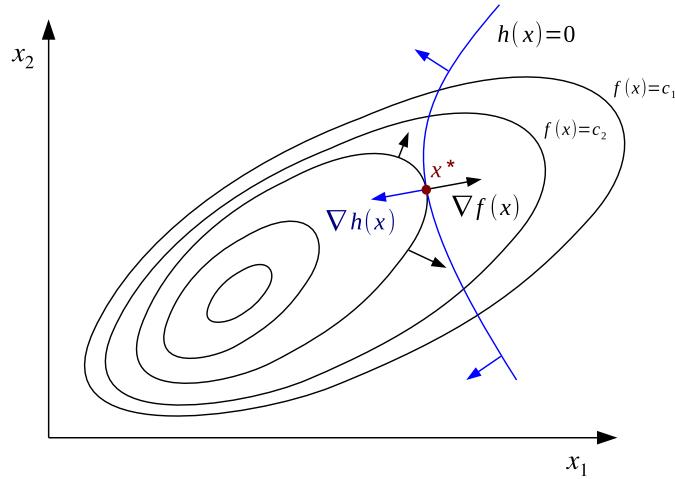
$$L(\mathbf{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\mathbf{x}) + \sum_{i=1}^m \alpha_i g_i(\mathbf{x}) + \sum_{i=1}^p \beta_i h_i(\mathbf{x})$$

gdje  $\alpha_i \geq 0$

- Rješenje originalnog problema je stacionarna točka u kojoj  $\nabla L = 0$
- $\nabla L = 0$  je u **točci sedla** funkcije  $L \Rightarrow$  minimum po  $\mathbf{x}$  a maksimum po  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$



- Objašnjenje Lagrangeove funkcije za **ograničenje jednakosti**:



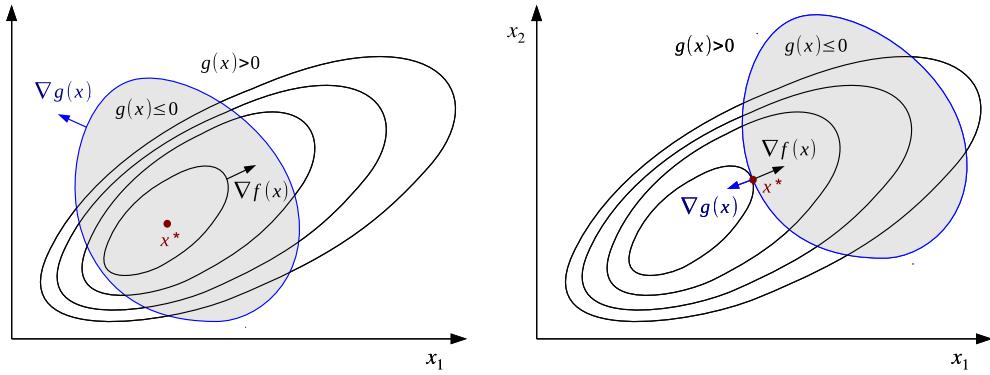
$\Rightarrow$  u stacionarnoj točci, vektori su kolinearni  $\Rightarrow$  postoji  $\beta$  za koju vrijedi:

$$\nabla f(\mathbf{x}) + \beta \nabla h(\mathbf{x}) = 0$$

što odgovara stacionarnoj točki (Lagrangeove) funkcije:

$$L(\mathbf{x}, \beta) = f(\mathbf{x}) + \beta h(\mathbf{x})$$

- Objašnjenje Lagrangeove funkcije za **ograničenje nejednakosti**:



- Moguća su dva slučaja:
  - minimum je unutar ostvarivog područja  $\Rightarrow$  ograničenje nije aktivno ( $\alpha = 0$ )
  - minimum je izvan ostvarivog područja  $\Rightarrow$  za neki  $\alpha > 0$  vrijedi:

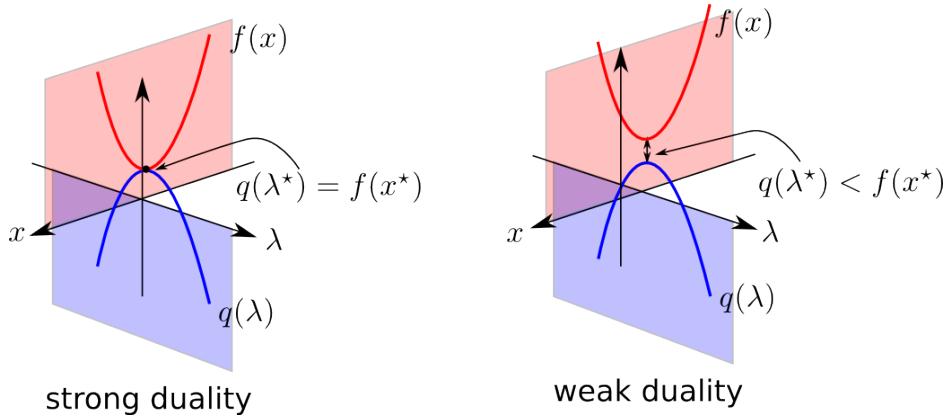
$$\nabla f(\mathbf{x}) = -\alpha \nabla g(\mathbf{x})$$

što odgovara stacionarnoj točki (Lagrangeove) funkcije:

$$L(\mathbf{x}, \alpha) = f(\mathbf{x}) + \alpha g(\mathbf{x})$$

$\Rightarrow$  u svakom slučaju, za točku rješenja vrijedi  $\alpha g(\mathbf{x}) = 0$

- Izvorna ograničenja i dva uvjeta za  $\alpha$  čine **Karush-Kuhn-Tuckerove (KKT) uvjete**
- Načelo dualnosti:** dualni problem je **donja ograda** primarnog problema



- Kod **jake dualnosti** ( $f(\mathbf{x})$  konveksna), primarno i dualno rješenje se poklapaju
- Dualna Lagrangeova funkcija:**

$$\tilde{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\alpha})$$

- Stacionarnu točku od  $L$  nalazimo **maksimizacijom** funkcije  $\tilde{L}$ , tj. dualni problem je:

$$\begin{aligned} &\text{maksimizirati} && \tilde{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}) \\ &\text{uz ograničenja} && \alpha_i \geq 0, \quad i = 1, \dots, m \end{aligned}$$

## 4 Optimizacija maksimalne margine

- Lagrangeova funkcija za problem maksimalne margine:

$$L(\mathbf{w}, w_0, \boldsymbol{\alpha}) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^N \alpha_i \left\{ y^{(i)} (\mathbf{w}^\top \mathbf{x}^{(i)} + w_0) - 1 \right\}$$

gdje  $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_N)$ ,  $\alpha_i \geq 0$ .

- Minimizacija funkcije  $L$  po primarnim varijablama:

$$\begin{aligned} \frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\alpha})}{\partial \mathbf{w}} &= 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^N \alpha_i y^{(i)} \mathbf{x}^{(i)} \\ \frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\alpha})}{\partial w_0} &= 0 \quad \Rightarrow \quad \sum_{i=1}^N \alpha_i y^{(i)} = 0 \end{aligned}$$

- Uvrštavanjem u  $L$  dobivamo dualnu Lagrangeovu funkciju:

$$\begin{aligned} \tilde{L}(\boldsymbol{\alpha}) &= \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^N \alpha_i \left\{ y^{(i)} (\mathbf{w}^\top \mathbf{x}^{(i)} + w_0) - 1 \right\} \\ &= \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y^{(i)} y^{(j)} (\mathbf{x}^{(i)})^\top \mathbf{x}^{(j)} \end{aligned}$$

- Dualni optimizacijski problem SVM-a jest maksimizirati:

$$\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y^{(i)} y^{(j)} (\mathbf{x}^{(i)})^\top \mathbf{x}^{(j)}$$

tako da:

$$\alpha_i \geq 0, \quad i = 1, \dots, N$$

$$\sum_{i=1}^N \alpha_i y^{(i)} = 0, \quad i = 1, \dots, N$$

$\Rightarrow$  kvadratni program rješiv algoritmom **SMO** (*sequential minimal optimization*)

- U točci rješenja vrijede uvjeti KKT:

$$\begin{aligned} y^{(i)} (\mathbf{w}^\top \mathbf{x}^{(i)} + w_0) &\geq 1, \quad i = 1, \dots, N \\ \alpha_i &\geq 0, \quad i = 1, \dots, N \\ \alpha_i (y^{(i)} h(\mathbf{x}^{(i)}) - 1) &= 0, \quad i = 1, \dots, N \end{aligned}$$

- Od  $n + 1$  primarne varijable došli smo na  $N$  dualnih varijabli  $\Rightarrow$  nekad isplativo

## 5 Dualni model SVM-a

- Na temelju jednakosti:

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y^{(i)} \mathbf{x}^{(i)}$$

izvodimo **dualnu formulaciju** modela:

$$h(\mathbf{x}) = \underbrace{\mathbf{w}^T \mathbf{x} + w_0}_{\text{Primarno}} = \underbrace{\sum_{i=1}^N \alpha_i y^{(i)} \mathbf{x}^T \mathbf{x}^{(i)} + w_0}_{\text{Dualno}}$$

⇒ umjesto težina hiperravnine  $\mathbf{w}$ , imamo dualne parametre  $\boldsymbol{\alpha}$

- Predikcija za  $\mathbf{x}^{(i)}$  temelji se na skalarnom umnošku  $\mathbf{x}^T \mathbf{x}^{(i)}$  ⇒ **sličnost vektora**
- Samo vektori za koje  $\alpha_i > 0$  utječu na predikciju ⇒ **potporni vektori**
- Težine hiperravnine (primarno) su linearna kombinacija potpornih vektora (dualno):

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y^{(i)} \mathbf{x}^{(i)}$$

- I težina  $w_0$  se može izraziti pomoću potpornih vektora (v. jednadžbu 7.8 u skripti)

## 9. Stroj potpornih vektora II

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

Jan Šnajder, natuknice s predavanja, v2.2

### 1 Podsjetnik

- Problem maksimalne margine
- Kvadratno programiranje – primarna formulacija
- Lagrangeova dualnost – prijelaz u dualni problem
- Maksimalna margina – dualna formulacija:

$$\underset{\alpha}{\operatorname{argmax}} \left( \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y^{(i)} y^{(j)} (\mathbf{x}^{(i)})^\top \mathbf{x}^{(j)} \right)$$

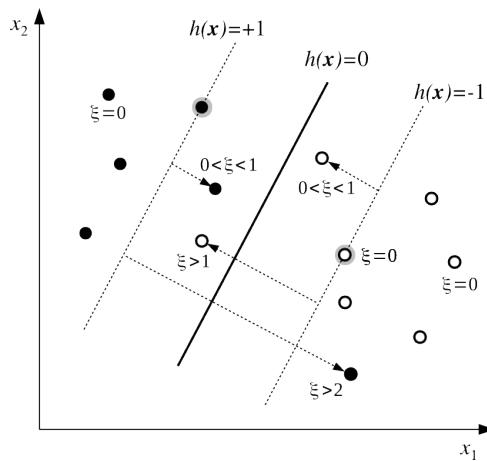
uz ograničenja:

$$\alpha_i \geq 0, \quad i = 1, \dots, N$$

$$\sum_{i=1}^N \alpha_i y^{(i)} = 0$$

### 2 Meka margina

- Gornja formulacija inzistira na **linearnoj odvojivosti**  $\Rightarrow$  uzrokuje **prenaučenost**
- Rješenje: dopustiti ulaske u marginu i pogrešne klasifikacije  $\Rightarrow$  **meka margina**



- Reformulacija ograničenja:

$$y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq 1 - \xi_i, \quad i = 1, \dots, N$$

gdje  $\xi_i \geq 0$  govori koliko je primjer  $\mathbf{x}^{(i)}$  ušao u marginu,  $\xi_i = |y^{(i)} - h(\mathbf{x}^{(i)})|$

- $\xi_i = 0 \Rightarrow$  ispravno klasificiran i izvan margine
- $0 < \xi_i \leq 1 \Rightarrow$  ispravno klasificiran, ali unutar margine
- $\xi_i > 1 \Rightarrow$  pogrešno klasificiran

- Ciljnu funkciju proširujemo **kaznom** za primjere za koje  $\xi_i > 0$ :

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

- $C \uparrow \Rightarrow$  tvrda margina, složen model;  $C \downarrow \Rightarrow$  meka margina, jednostavan model
- Optimizacijski problem meke margine:

$$\underset{\mathbf{w}, w_0, \boldsymbol{\xi}}{\operatorname{argmin}} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \right\}$$

uz ograničenja:

$$\begin{aligned} y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) &\geq 1 - \xi_i, \quad i = 1, \dots, N \\ \xi_i &\geq 0, \quad i = 1, \dots, N \end{aligned}$$

- Pripadna **Lagrangeova funkcija**:

$$L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) - 1 + \xi_i) - \sum_{i=1}^N \beta_i \xi_i$$

- Rješenje zadovoljava **uvjete KKT**:

$$\begin{aligned} y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) &\geq 1 - \xi_i & i = 1, \dots, N \\ \xi_i &\geq 0 & i = 1, \dots, N \\ \alpha_i &\geq 0 & i = 1, \dots, N \\ \beta_i &\geq 0 & i = 1, \dots, N \\ \alpha_i (y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) - 1 + \xi_i) &= 0 & i = 1, \dots, N \\ \beta_i \xi_i &= 0 & i = 1, \dots, N \end{aligned}$$

- Minimum po primarnim parametrima:

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{w}} &= 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^N \alpha_i y^{(i)} \mathbf{x}^{(i)} \\ \frac{\partial L}{\partial w_0} &= 0 \quad \Rightarrow \quad \sum_{i=1}^N \alpha_i y^{(i)} = 0 \\ \frac{\partial L}{\partial \xi_i} &= 0 \quad \Rightarrow \quad \alpha_i = C - \beta_i \end{aligned}$$

- Uvrštavanjem u  $L$  dobivamo **dualnu Lagrangeovu funkciju**:

$$\tilde{L}(\boldsymbol{\alpha}) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y^{(i)} y^{(j)} (\mathbf{x}^{(i)})^\top \mathbf{x}^{(j)}$$

- Pripadni dualni optimizacijski problem:

$$\operatorname{argmax}_{\boldsymbol{\alpha}} \left( \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y^{(i)} y^{(j)} (\mathbf{x}^{(i)})^\top \mathbf{x}^{(j)} \right)$$

uz ograničenja:

$$0 \leq \alpha_i \leq C, \quad i = 1, \dots, N$$

$$\sum_{i=1}^N \alpha_i y^{(i)} = 0, \quad i = 1, \dots, N$$

- Kao i za tvrdnu marginu, uz dodatno ograničenje  $\alpha_i \leq C$
- Vektori za koje  $0 < \alpha_i \leq C$  su **potporni vektori** (oni s  $\alpha_i = C$  su unutar margine)

### 3 Gubitak zglobnice

- Alternativna formulacija SVM-a: funkcija gubitka i minimizacija pogreške
- Vrijedi

$$\xi_i = |y^{(i)} - h(\mathbf{x}^{(i)})| = 1 - y^{(i)} h(\mathbf{x}^{(i)})$$

pa kaznu po primjeru  $\xi_i$  možemo napisati kao funkciju gubitka:

$$L(y, h(\mathbf{x})) = \max(0, 1 - y h(\mathbf{x})).$$

⇒ **gubitak zglobnice (hinge loss)**

- Uvrštavanjem u ciljnu funkciju **primarnog** optimizacijskog problema:

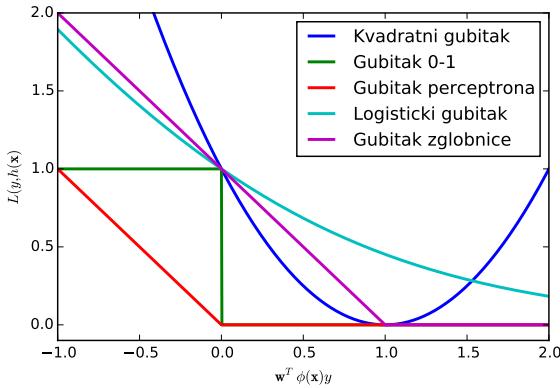
$$\operatorname{argmin}_{\mathbf{w}, w_0, \boldsymbol{\xi}} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \right\}$$

dobivamo:

$$E(\mathbf{w} | \mathcal{D}) = \sum_{i=1}^N \max(0, 1 - y^{(i)} h(\mathbf{x}^{(i)})) + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

gdje  $\lambda = 1/C$

- Može se minimizirati npr. stohastičkim (pod)gradijentnim spustom
- Usporedba funkcija gubitaka:



## 4 Alternativni SVM algoritmi

- Primarna formulacija  $\Rightarrow$ 
  - Gubitak zglobnice  $\Rightarrow$  SGD, SGP, koordinatni spust, ...
  - Kvadratno programiranje (QP)  $\Rightarrow$ 
    - \* Lagrangeova dualnost  $\Rightarrow$  Dualno QP  $\Rightarrow$  SMO
    - \* Koordinatni spust, metode kazne, metode unutarnje točke, ...
- SVM rješavači: SMO, LibSVM, LibLinear, SVM<sup>light</sup>, Pegasos, ...

## 5 Napomene

- SVM regresija (SVR)
- Hiperparametar  $C$  – određuje složenost, odabrati unakrsnom provjerom
- Skaliranje – skalirati značajke, da ne dominiraju one s većim rasponom
- Višeklasna klasifikacija – preporuča se OVO zbog manje neuravnoveženosti klasa
- Probabilistički izlaz – može se aproksimirati Plattovom metodom (izlazna sigmoida)

$$P(y = 1|\mathbf{x}) = \sigma(ah(\mathbf{x}) + b)$$

- Nelinearnost – preslikavanjem  $\phi$  ili **jezgrenim trikom**  $\Rightarrow$  iduće predavanje

# 10. Jezgrene metode

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

Jan Šnajder, natuknice s predavanja, v1.3

## 1 Jezgrene funkcije

- Umjesto težina uz vektor značajki  $\mathbf{x}$ , izračunavamo **sličnost** dvaju primjera
- Naročito prikladno kada se primjeri teško vektoriziraju (npr. jer imaju strukturu)
- **Jezgrena funkcija** (*kernel function*):  $\kappa : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$
- Jezgrena funkcija je **mjera sličnosti** ako zadovoljava:
  - $\kappa(\mathbf{x}, \mathbf{x}) = 1$
  - $0 \leq \kappa(\mathbf{x}, \mathbf{x}') \leq 1$
  - $\kappa(\mathbf{x}, \mathbf{x}') = \kappa(\mathbf{x}', \mathbf{x})$
- Jezgre za strukturirane podatke: **string kernels**, **tree kernels**, **graph kernels**
- Tipične jezgrene funkcije za primjere u vektorskem prostoru:
  - **Linearna jezgra**:  $\kappa(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$
  - **Radijalna bazna funkcija (RBF)**: općenito jezgra tipa  $\kappa(\mathbf{x}, \mathbf{x}') = f(\|\mathbf{x} - \mathbf{x}'\|)$
  - **Gaussova RBF-jezgra**:

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

gdje je  $\sigma^2$  je širina pojasa (*bandwidth*),  $\gamma = 1/2\sigma^2$  je preciznost  
(manja  $\sigma^2 \Leftrightarrow$  veća  $\gamma \Leftrightarrow$  primjeri su međusobno sve različitiji)

- **Ekponencijalna jezgra**:  $\kappa(\mathbf{x}, \mathbf{x}') = \exp(-\gamma\|\mathbf{x} - \mathbf{x}'\|)$
- **Inverzna kvadratna jezgra**:  $\kappa(\mathbf{x}, \mathbf{x}') = \frac{1}{1 + \|\mathbf{x} - \mathbf{x}'\|^2}$

- Umjesto euklidske, može se koristiti **Mahalanobisova udaljenost**:

$$\kappa(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Sigma^{-1} (\mathbf{x} - \mathbf{x}')\right)$$

gdje je  $\Sigma$  kovarijacijska matrica značajki

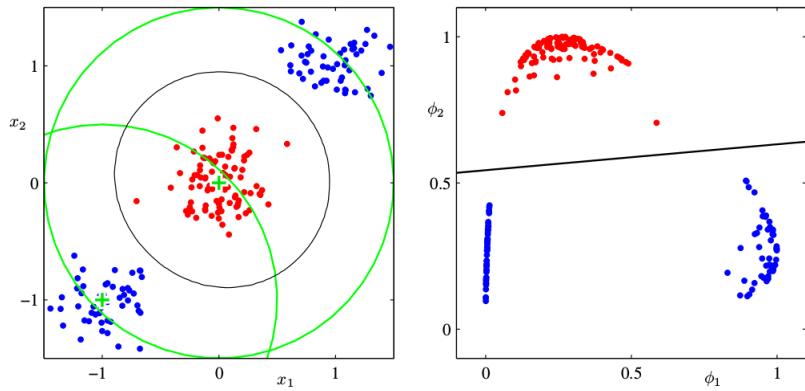
## 2 Jezgreni strojevi

- Preslikavanje  $\phi$  koje za bazne funkcije  $\phi_j$  koristi jezgrene funkcije:

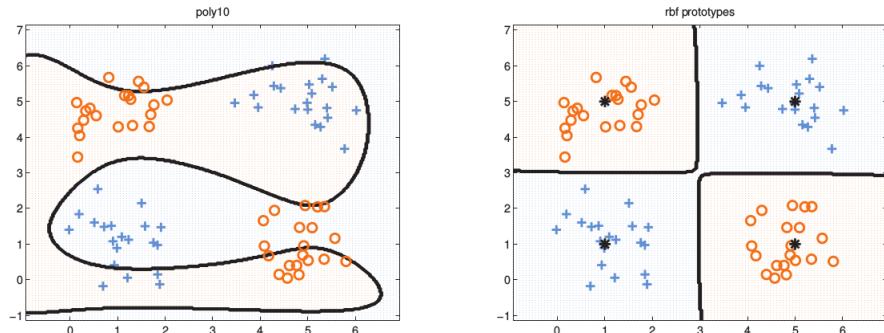
$$\phi(\mathbf{x}) = (1, \kappa(\mathbf{x}, \boldsymbol{\mu}_1), \kappa(\mathbf{x}, \boldsymbol{\mu}_2), \dots, \kappa(\mathbf{x}, \boldsymbol{\mu}_m))$$

gdje su  $\boldsymbol{\mu}_j \in \mathcal{X}$  odabrane točke u prostoru primjera

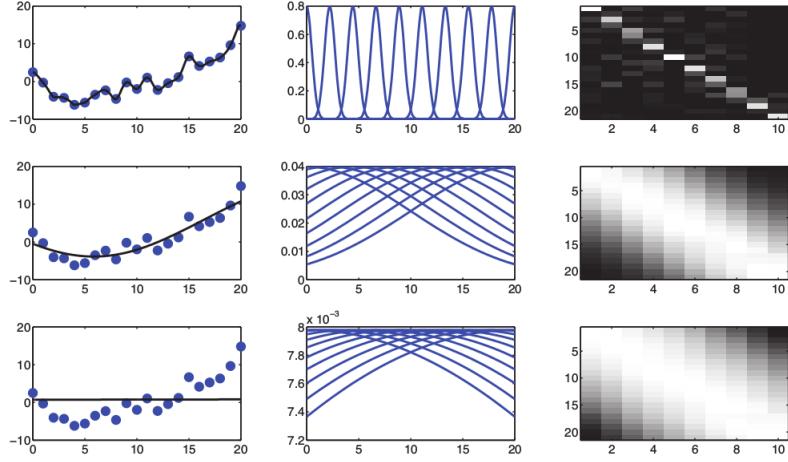
- **Jezgreni stroj** (*kernel machine*) – poopćeni linearni model s takvim preslikavanjem
- Primjer (iz MLPP) – klasifikacija:



- Primjer (iz MLPP) – klasifikacija:



- Primjer (iz MLPP) – regresija:



- Uniforman raspored  $\mu_j \Rightarrow$  neprilagođen podatcima, problem visokih dimenzija
- Alternativa:  $\mu_j$  su primjeri iz skupa za učenje:

$$\phi(\mathbf{x}) = (1, \kappa(\mathbf{x}, \mathbf{x}_1), \kappa(\mathbf{x}, \mathbf{x}_2), \dots, \kappa(\mathbf{x}, \mathbf{x}_N))$$

- Problem: primjera može biti puno; rješenje: **L1-regularizacija**
- **Rijetki jezgreni strojevi:** L1VM, SVM

### 3 Jezgreni trik

- SVM je rijedak jezgreni stroj, no umjesto preslikavanja koristi jezgreni trik
- **Jezgreni trik** – skalarni produkt vektora zamjenjuje se jezgrenom funkcijom:

$$\kappa(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}')$$

- Model SVM:

$$h(\mathbf{x}) = \sum_{i=1}^N \alpha_i y^{(i)} \phi(\mathbf{x})^\top \phi(\mathbf{x}^{(i)}) + w_0 = \sum_{i=1}^N \alpha_i y^{(i)} \kappa(\mathbf{x}, \mathbf{x}^{(i)}) + w_0$$

- Ciljna funkcija (kvadratno programiranje):

$$\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y^{(i)} y^{(j)} \kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$$

- **Inverzno oblikovanje** – odabiremo jezgru i time implicitno definiramo  $\phi$
- Prednosti:
  - Manja računalna složenost (izračun  $\kappa$  je često jeftiniji od izračuna  $\phi$ )
  - Nekad je lakše definirati  $\kappa$  nego  $\phi$  (strukturirani podatci: nizovi, stabla, grafovi)

- Prostor koji inducira  $\kappa$  može biti visoko (potencijalno beskonačno) dimenzijski
- Uvjet:  $\kappa$  odgovara skalarnom produktu u nekom vekt. prostoru  $\Rightarrow$  **Mercerova jezgra**
- Jezgrena matrica (*kernel matrix*):

$$\mathbf{K} = \begin{pmatrix} \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) & \dots & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(N)}) \\ \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(2)}) & \dots & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(N)}) \\ \vdots & \vdots & \ddots & \vdots \\ \kappa(\mathbf{x}^{(N)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(N)}, \mathbf{x}^{(2)}) & \dots & \kappa(\mathbf{x}^{(N)}, \mathbf{x}^{(N)}) \end{pmatrix}$$

- $\mathbf{K} = \Phi\Phi^T \Leftrightarrow \mathbf{K}$  je **Gramova matrica** (matrica skalarnih produkata)
- Gramova matrica je uvijek pozitivno semidefinitna ( $\forall \mathbf{x}, \mathbf{x}^T \mathbf{K} \mathbf{x} \geq 0$ )
- **Mercerov teorem:**  $\mathbf{K}$  je pozitivno semidefinitna  $\Leftrightarrow \kappa$  je Mercerova jezgra
- Inducirani prostor: skalarni produkt + proizvoljna dimenzija  $\Rightarrow$  **Hilbertov prostor**
- Mercerove jezgre: linearna, polinomijalna, RBF-jezgra, string kernels, ...
- Preslikavanje polinomijalne jezgre
  - Općenito:  $\kappa(\mathbf{x}, \mathbf{x}') = (\gamma \mathbf{x}^T \mathbf{x}' + c)^d$
  - Primjer za  $n = 2, d = 2, c = 0, \gamma = 1$ :

$$\begin{aligned} \kappa(\mathbf{x}, \mathbf{z}) &= (\mathbf{x}^T \mathbf{z})^2 = ((x_1, x_2)^T (z_1, z_2))^2 = (x_1 z_1 + x_2 z_2)^2 \\ &= (x_1 z_1)^2 + 2(x_1 z_1)(x_2 z_2) + (x_2 z_2)^2 = x_1^2 z_1^2 + \sqrt{2} x_1 x_2 \sqrt{2} z_1 z_2 + x_2^2 z_2^2 \\ &= (x_1^2, \sqrt{2} x_1 x_2, x_2^2)^T (z_1^2, \sqrt{2} z_1 z_2, z_2^2) = \phi(\mathbf{x})^T \phi(\mathbf{z}) \\ \Rightarrow \phi(\mathbf{x}) &= (x_1^2, \sqrt{2} x_1 x_2, x_2^2) \end{aligned}$$

- Preslikavanje RBF-jezgre:
  - $\mathbf{K}$  je punog ranga  $\Leftrightarrow \phi(\mathbf{x})$  su lin. nezavisni  $\Rightarrow$  **beskonačnodimenzijski prostor**
  - $\gamma \rightarrow \infty \Leftrightarrow \kappa(\mathbf{x}, \mathbf{x}') \rightarrow 0 \Leftrightarrow \phi(\mathbf{x})^T \phi(\mathbf{x}') = 0 \Leftrightarrow$  primjeri su ortonormirani  
 $\Leftrightarrow$  primjeri su vrhovi višedimenzijskog simpleksa  $\Leftrightarrow$  linearno su odvojivi
- Složenije Mercerove jezgre gradimo operacijama koje zadržavaju to svojstvo:

$$\begin{aligned} \kappa(\mathbf{x}, \mathbf{x}') &= \alpha \kappa_1(\mathbf{x}, \mathbf{x}') & \alpha > 0 \\ \kappa(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x}) \kappa_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}') & f - \text{bilo koja funkcija} \\ \kappa(\mathbf{x}, \mathbf{x}') &= q(\kappa_1(\mathbf{x}, \mathbf{x}')) & q - \text{polinom s poz. koef.} \\ \kappa(\mathbf{x}, \mathbf{x}') &= \exp(\kappa_1(\mathbf{x}, \mathbf{x}')) \\ \kappa(\mathbf{x}, \mathbf{x}') &= \kappa_1(\phi(\mathbf{x}), \phi(\mathbf{x}')) & \phi : \mathbb{R}^n \rightarrow \mathbb{R}^{m+1} \\ \kappa(\mathbf{x}, \mathbf{x}') &= \kappa_1(\mathbf{x}, \mathbf{x}') + \kappa_2(\mathbf{x}, \mathbf{x}') \\ \kappa(\mathbf{x}, \mathbf{x}') &= \kappa_1(\mathbf{x}, \mathbf{x}') \kappa_2(\mathbf{x}, \mathbf{x}') \end{aligned}$$

$\Rightarrow$  **multiple kernel learning (MKL)**

## 4 Napomene

- Odabir modela kod SVM-a
  - RBF-jezgra: hiperparametri  $C$  i  $\gamma$  su međuovisni ( $\gamma \uparrow \Leftrightarrow C \downarrow$ )
    - odabir modela najčešće se radi **pretraživanjem po rešetci** (*grid search*)
- Linearna jezgra – ne daje nelinearnost, ali daje rijetka rješenja (potporni vektori)
- Jezgeni trik primjenjiv je na druge algoritme (npr. kernelizirana linearna regresija)
- **Aproksimacija kernela** (kada je  $N$  velik) – aproksimacija preslikavanja  $\phi$  + SGD

# 11. Neparametarske metode

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

Jan Šnajder, natuknice s predavanja, v1.3

## 1 Parametarske vs. neparametarske metode

- **Parametarske metode** – hipoteza je definirana do na parametre  $\theta$ 
  - broj parametara modela  $n$  (složenost modela) ne ovisi o broju primjera  $N$
  - pretpostavljaju da se podatci ravnaju po nekom modelu (distribuciji)
  - primjeri imaju **globalan** utjecaj na izgled hipoteze
- **Neparametarske metode** – hipoteza nije eksplizitno definirana
  - broj parametara ovisi o broju primjera
  - ne pretpostavljaju model (distribuciju) podataka
  - **lokalna** aproksimacija hipoteze u okolini pohranjenih primjera
- NB: Neparametarski modeli imaju parametre (ali nemaju parametre distribucije)!
- Predikcija se ne radi unaprijed nego na zahtjev  $\Rightarrow$  **lijene metode** (*lazy methods*)
- **Induktivna pristranost** neparametarskih metoda: slični primjeri imaju slične oznake
- Preporuke:
  - malo podataka i/ili poznat model/distribucija  $\Rightarrow$  parametarski postupci
  - mnogo podataka i nepoznat model/distribucija  $\Rightarrow$  neparametarski postupci

## 2 SVM

- SVM model:

$$h(\mathbf{x}) = \underbrace{\mathbf{w}^\top \mathbf{x} + w_0}_{\text{Primarno}} = \underbrace{\sum_{i=1}^N \alpha_i y^{(i)} \mathbf{x}^\top \mathbf{x}^{(i)}}_{\text{Dualno}} + w_0$$

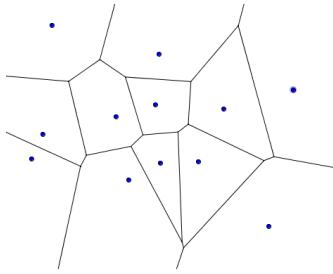
- Primarna formulacija  $\Rightarrow$  parametarski; dualna formulacija  $\Rightarrow$  neparametarski
- Broj parametara proporcionalan broju potpornih vektora, koji ovisi o  $N$
- Prikladno kada  $N \ll n$  (algoritam SMO ima složenost  $\mathcal{O}(N^2)$ )

### 3 Algoritam k-NN

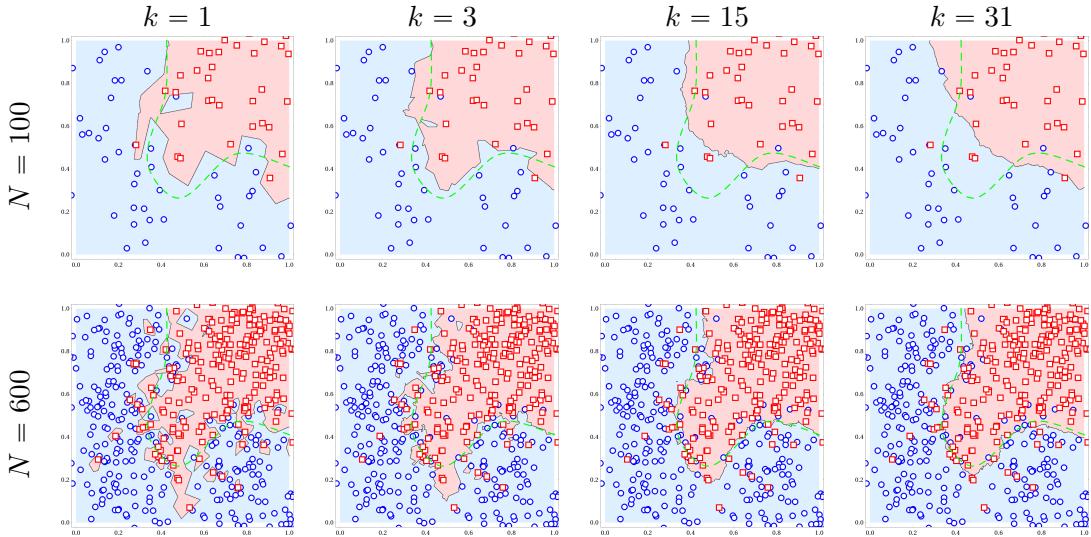
- Neparametarski klasifikacijski algoritam
- Predikcija na temelju većinske oznake  $k$  **najbližih susjeda** (*nearest neighbors*):

$$h(\mathbf{x}) = \operatorname{argmax}_{j \in \{0, \dots, K-1\}} \sum_{(\mathbf{x}^{(i)}, y^{(i)}) \in \text{NN}_k(\mathbf{x})} \mathbf{1}\{y^{(i)} = j\}$$

- $k$  je **hiperparametar** algoritma  $\Rightarrow$  manji  $k$  daje složeniji model
- $k = 1 \Rightarrow$  ulazni prostor particioniran u **Voronoijev dijagram**:



- Primjer: binarna klasifikacija u  $n = 2$  u ovisnosti o  $k$  za  $N = 100$  i  $N = 600$ :

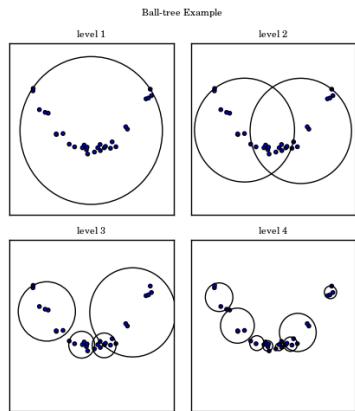


- **Težinski k-NN** – utjecaj primjera ovisi o udaljenosti/sličnosti  $\Rightarrow$  **kernel**:

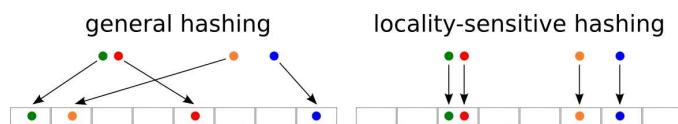
$$h(\mathbf{x}) = \operatorname{argmax}_{j \in \{0, \dots, K-1\}} \sum_{i=1}^N \kappa(\mathbf{x}^{(i)}, \mathbf{x}) \mathbf{1}\{y^{(i)} = j\}$$

- Mjera udaljenosti ne mora biti euklidska (npr. Mahalonbisova udaljenost)
- Računalni problem: **nalaženje nabližeg susjeda** (*nearest neighbor search*)
- Alternative iscrpnom pretraživanju (bitno za velike skupove podataka):

- egzaktne metode: indeksiranje prostora primjera (npr. **ball tree**)



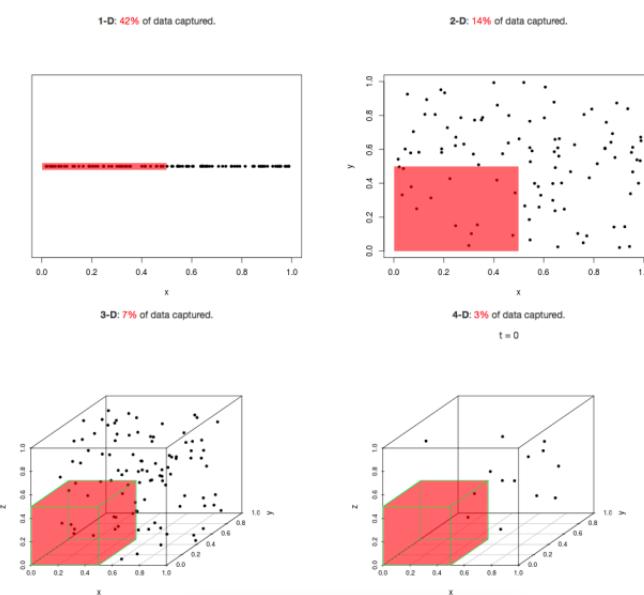
- aproksimativne metode: **locally sensitive hashing (LSH)**



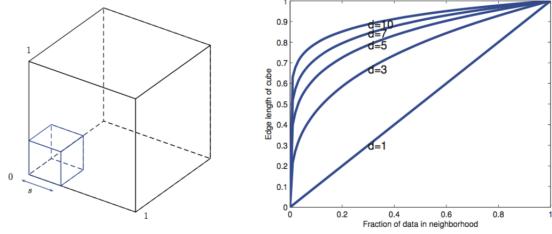
- **Prokletstvo dimenzionalnosti (curse of dimensionality):**

- s porastom dimenzije  $n$  sve točke postaju međusobno vrlo udaljene
- udaljenosti postaju nediskriminative
- općenit problem svih algoritama u visokodimenzijskim prostorima

- Primjer: s porastom broja dimenzija udio podataka u jediničnoj hiperkocki opada:



- Primjer: s porastom broja dimenzija, udaljenost između susjeda raste:



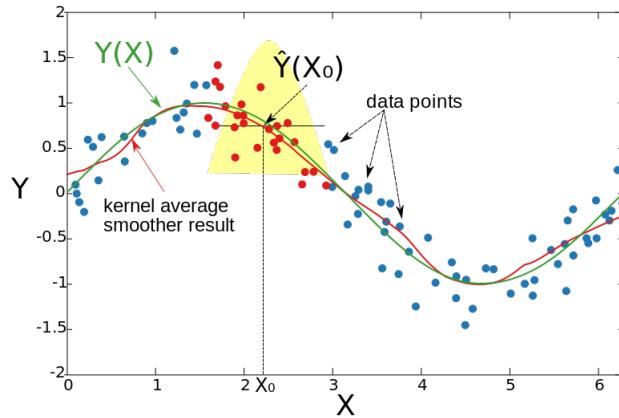
## 4 Neparametarska regresija

- Neparametarska regresija = **modeli zaglađivanja** (*smoothing models*)
- **$k$ -nn smoother** - prosjek vrijednosti  $k$  najbližih susjeda:

$$h(\mathbf{x}) = \frac{1}{k} \sum_{(\mathbf{x}^{(i)}, y^{(i)}) \in \text{NN}_k(\mathbf{x})} y^{(i)}$$

- **Jezgreno zaglađivanje** (*kernel smoothing*):

$$h(\mathbf{x}) = \frac{\sum_{i=1}^N \kappa(\mathbf{x}^{(i)}, \mathbf{x}) y^{(i)}}{\sum_{i=1}^N \kappa(\mathbf{x}^{(i)}, \mathbf{x})}$$



## 5 Stabla odluke

- Neparametarski model jer broj parametara ( $\propto$  broj razina) raste s brojem primjera
- Ulazni prostor rekurzivno dijeli na lokalna područja (dva potprostora)

# 13. Procjena parametara

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

Jan Šnajder, natuknice s predavanja, v1.3

## 1 Motivacija

- **Probabilistički modeli** modeliraju vjerojatnosnu razdiobu primjera  $\mathbf{x}$  i/ili oznaka  $y$
- **Prednosti:** (1) temeljeni na teoriji vjerojatnosti, (2) vjerojatnosna predikcija, (3) ugradnja apriornog znanja, (4) prikladni za male skupove podataka
- Npr. **Bayesov klasifikator** –  $P(y|\mathbf{x}) \propto P(\mathbf{x}|y)P(y)$ 
  - odabrati prikladne razdiobe za  $P(\mathbf{x}|y)$  i  $P(y)$
  - **procijeniti parametre** razdioba na temelju podataka  $\Leftrightarrow$  učenje modela

## 2 Slučajne varijable

- $X$  – slučajna varijabla (s.v.) sa skupom mogućih vrijednosti  $\{x_i\}$
- **Diskretna s.v.:**
  - $P(X = x)$ , kraće  $P(x)$  – **vjerojatnost** da diskretna s.v. poprimi vrijednost  $x$
  - $P(x_i) \geq 0, \sum_i P(x_i) = 1 \Rightarrow$  **diskretna razdioba (distribucija) vjerojatnosti**
- **Kontinuirana s.v.:**
  - $p(x)$  – **funkcija gustoće vjerojatnosti (PDF)**
  - $p(x) \geq 0, \int_{-\infty}^{\infty} p(x) dx = 1 \Rightarrow$  **kontinuirana razdioba (distribucija) vjerojatnosti**
- **Očekivanje** – prosječna vrijednost:  $\mathbb{E}[X] = \sum_x xP(x)$  odnosno  $\mathbb{E}[X] = \int_{-\infty}^{\infty} x p(x) dx$
- **Varijanca** – očekivano odstupanje od očekivanja:  $\text{Var}(X) = \sigma_X^2 = \mathbb{E}[(X - \mathbb{E}[X])^2]$
- **Kovarijanca** – zajednička varijabilnost dviju varijabli:

$$\text{Cov}(X, Y) = \sigma_{X,Y} = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$$

$$\Rightarrow \text{Cov}(X, Y) = \text{Cov}(Y, X), \text{Cov}(X, X) = \text{Var}(X)$$

- **Pearsonov koeficijent korelacijske**:  $\rho_{X,Y} = \frac{\text{Cov}(X,Y)}{\sqrt{\text{Var}(X)\text{Var}(Y)}} = \frac{\sigma_{X,Y}}{\sigma_X \sigma_Y} \in [-1, +1]$

- $\rho_{X,Y} = +1 \Leftrightarrow$  pozitivna **linearna zavisnost**
- $\rho_{X,Y} = 0 \Leftrightarrow$  **linearna nezavisnost**
- $\rho_{X,Y} = -1 \Leftrightarrow$  negativna **linearna zavisnost**

$\Rightarrow$  ne mjeri **nelinearnu zavisnost!**

- **Matrica kovarijacije** – kovarijacija svih parova varijabli **slučajnog vektora**  $(X_1, \dots, X_n)$ :

$$\Sigma = \begin{pmatrix} \text{Var}(X_1) & \text{Cov}(X_1, X_2) & \dots & \text{Cov}(X_1, X_n) \\ \text{Cov}(X_2, X_1) & \text{Var}(X_2) & \dots & \text{Cov}(X_2, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(X_n, X_1) & \text{Cov}(X_n, X_2) & \dots & \text{Var}(X_n) \end{pmatrix} = \mathbb{E}[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{X} - \mathbb{E}[\mathbf{X}])^T]$$

- simetrična i pozitivno semidefinitna
- singularna (tj. nema inverz) ako postoje linearno zavisni retci ili ako  $\sigma_i^2 = 0$
- $\text{Cov}(X_i, X_j) = 0 \Rightarrow$  **dijagonalna** kovarijacijska matrica,  $\Sigma = \text{diag}(\sigma_i^2)$
- $\sigma_i^2 = \sigma \Rightarrow$  **izotropna** kovarijacijska matrica,  $\Sigma = \sigma^2 \mathbf{I}$

- **Nezavisne varijable**  $\Leftrightarrow P(X, Y) = P(X)P(Y)$

- nezavisne varijable su nekorelirane,  $\text{Cov}(X, Y) = \rho_{X,Y} = 0$ , ali obrat ne vrijedi!

### 3 Osnovne vjerojatnosne distribucije

- Diskretna varijabla:
  - Jednodimenzija binarna: **Bernoullijeva razdioba**
  - Jednodimenzija viševrijednosna: **kategorička (multinulijeva) razdioba**
  - Višedimenzija: Konkatenirani vektor binarnih/viševrijednosnih varijabli
- Kontinuirana varijable:
  - Jednodimenzija: **univariatna normalna (Gaussova) razdioba**
  - Višedimenzija: **multivariatna normalna (Gaussova) razdioba**
- **Bernoullijeva razdioba** – binarna s.v.:

$$P(x|\mu) = \mu^x(1-\mu)^{1-x}$$

$$\Rightarrow \mathbb{E}[X] = \mu, \text{Var}(X) = \mu(1-\mu)$$

- **Kategorička (multinulijeva) razdioba** – viševrijednosna diskretna s.v.:

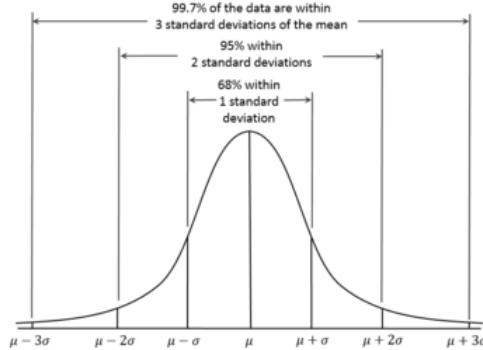
$$P(\mathbf{x}|\boldsymbol{\mu}) = \prod_{k=1}^K \mu_k^{x_k}$$

- $\mathbf{x} = (x_1, x_2, \dots, x_K)^T$  – vektor indikatorskih varijabli (**one-hot encoding**)
- $\boldsymbol{\mu} = (\mu_1, \dots, \mu_K)$  – vjerojatnosti pojedinih vrijednosti,  $\sum_k \mu_k = 1, \mu_k \geq 0$

- **Normalna (Gaussova) razdioba** – kontinuirana vrijednost uz prisustvo **šuma**:

$$p(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$\Rightarrow \mathbb{E}[X] = \mu, \text{Var}(X) = \sigma^2$$

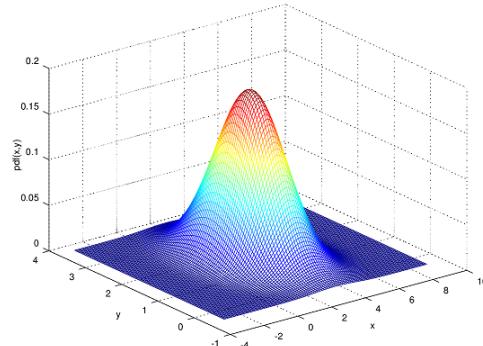


- **Multivariatna (višedimenzijska) normalna (Gaussova) razdioba**:

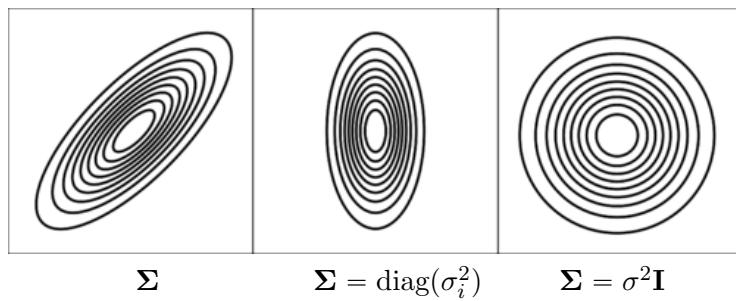
$$p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right)$$

$$\Rightarrow \mathbb{E}[X] = \boldsymbol{\mu}, \text{Cov}(X_i, X_j) = \boldsymbol{\Sigma}_{ij}$$

- značajke su savršeno **multikolinearne**  $\Leftrightarrow \boldsymbol{\Sigma}$  je singularna  $\Leftrightarrow p(\mathbf{x})$  je nedefinirana
- $\Delta^2 = (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})$  – **kvadratna forma**
- $\Delta$  – **Mahalanobisova udaljenost** između  $\mathbf{x}$  i  $\boldsymbol{\mu}$

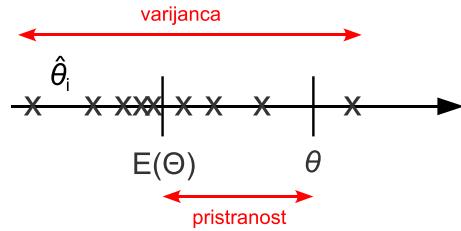


- Kovariacijska matrica određuje izgled Gaussove razdiobe:



## 4 Procjena parametara

- Raspolažemo konačnim i slučajnim (= reprezentativnim) uzorkom iz **populacije**
- Na temelju uzorka **procjenjujemo parametre** modela koji opisuje populaciju
- **Uzorak** – niz s.v.  $(X_1, X_2, \dots, X_N)$  koje su **iid** (identično i nezavisno distribuirane)
- **Statistika** – funkcija slučajnog uzorka,  $\Theta = g(X_1, X_2, \dots, X_N)$
- **Procjenitelj (estimator)** – statistika koja odgovara parametru populacije  $\theta$
- **Procjena (estimacija)** – vrijednost procjenitelja za dani uzorak,  $\hat{\theta} = g(x_1, x_2, \dots, x_N)$
- Procjenitelj je s.v., pa ima svoje očekivanje i varijancu



- **Pristranost (bias)** – razlika između očekivanja procjenitelja i parametra populacije:

$$b(\Theta) = \mathbb{E}[\Theta] - \theta$$

- **Nepristran procjenitelj (unbiased estimator)**  $\Leftrightarrow \mathbb{E}[\Theta] = \theta \Leftrightarrow b(\Theta) = 0$

- Primjeri:

- $\hat{\mu} = \frac{1}{N} \sum_i x^{(i)}$  – nepristran procjenitelj srednje vrijednosti  $\mu$
- $\hat{\sigma}^2 = \frac{1}{N} \sum_{i=1}^N (x^{(i)} - \hat{\mu})^2$  – pristran procjenitelj varijance  $\sigma^2$  (podcjenjuje!)
- $\hat{\sigma}^2 = \frac{1}{N-1} \sum_{i=1}^N (x^{(i)} - \hat{\mu})^2$  – nepristran procjenitelj varijance  $\sigma^2$

- Postupci za izvođenje procjenitelja:

- **Procjenitelj najveće izglednosti** (maximum likelihood estimator, **MLE**)
- **Procjenitelj maximum a posteriori (MAP)**
- **Bayesovski procjenitelj** (bayesian estimator)

- Radit ćemo MLE i MAP

# 14. Procjena parametara II

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

Jan Šnajder, natuknice s predavanja, v1.5

## 1 Funkcija izglednosti

- Skup podataka  $\mathcal{D} = \{\mathbf{x}^{(i)}\}_{i=1}^N$  koji su **iid**; prepostavka:  $\mathbf{x}^{(i)} \sim p(\mathbf{x}|\boldsymbol{\theta})$
- Vjerojatnost uzorka:

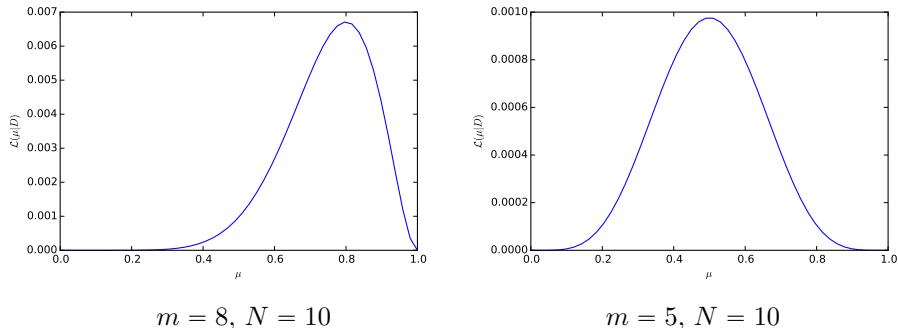
$$p(\mathcal{D}|\boldsymbol{\theta}) = p(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}|\boldsymbol{\theta}) = \prod_{i=1}^N p(\mathbf{x}^{(i)}|\boldsymbol{\theta}) \equiv \mathcal{L}(\boldsymbol{\theta}|\mathcal{D})$$

gdje je  $\mathcal{L} : \boldsymbol{\theta} \mapsto p(\mathcal{D}|\boldsymbol{\theta})$  **funkcija izglednosti**

- $\mathcal{L}(\boldsymbol{\theta}|\mathcal{D})$  – vjerojatnost realizacije uzorka  $\mathcal{D}$ , ako je parametar populacije jednak  $\boldsymbol{\theta}$
- Npr. funkcija izglednosti Bernoullijske varijable –  $m$  pozitivnih ishoda u  $N$  pokusa:

$$\mathcal{L}(\mu|\mathcal{D}) = P(\mathcal{D}|\mu) = P(x^{(1)}, \dots, x^{(N)}|\mu) = \prod_{i=1}^N P(x^{(i)}|\mu) = \mu^m (1-\mu)^{(N-m)}$$

gdje  $m = \sum_i x^{(i)}$



## 2 Procjenitelj MLE

- Prepostavka: uzorak  $\mathcal{D}$  je **najvjerojatniji mogući**, inače ne bi bio izvučen
- Najbolja procjena za  $\boldsymbol{\theta}$  je ona koja maksimizira izglednost  $\mathcal{L}(\boldsymbol{\theta}|\mathcal{D})$

- **Procjenitelj najveće izglednosti** (*maximum likelihood estimator*) – **MLE**:

$$\hat{\boldsymbol{\theta}}_{\text{MLE}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \mathcal{L}(\boldsymbol{\theta} | \mathcal{D})$$

- Radi matematičke jednostavnosti, maksimizirat ćemo **log-izglednost**:

$$\hat{\boldsymbol{\theta}}_{\text{MLE}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} (\ln \mathcal{L}(\boldsymbol{\theta} | \mathcal{D}))$$

- MLE za parametar Bernoullijeve razdiobe:

$$\begin{aligned} \ln \mathcal{L}(\mu | \mathcal{D}) &= \ln \prod_{i=1}^N \mu^{x^{(i)}} (1-\mu)^{1-x^{(i)}} = \sum_{i=1}^N x^{(i)} \ln \mu + \left( N - \sum_{i=1}^N x^{(i)} \right) \ln(1-\mu) \\ \frac{\partial \ln \mathcal{L}}{\partial \mu} &= \frac{1}{\mu} \sum_{i=1}^N x^{(i)} - \frac{1}{1-\mu} \left( N - \sum_{i=1}^N x^{(i)} \right) = 0 \\ \Rightarrow \hat{\mu}_{\text{MLE}} &= \frac{1}{N} \sum_{i=1}^N x^{(i)} = \frac{m}{N} \end{aligned}$$

$\Rightarrow$  **relativna frekvencija** (udio realizacije  $x = 1$ )

- MLE za parametre kategorijalne razdiobe:

$$\ln \mathcal{L}(\boldsymbol{\mu} | \mathcal{D}) = \ln \prod_{i=1}^N P(\mathbf{x}^{(i)} | \boldsymbol{\mu}) = \ln \prod_{i=1}^N \prod_{k=1}^K \mu_k^{x_k^{(i)}} = \sum_{k=1}^K \sum_{i=1}^N x_k^{(i)} \ln \mu_k$$

$\Rightarrow$  maksimizacijom po  $\mu_k$  uz  $\sum_{k=1}^K \mu_k = 1$  metodom **Lagrangeovih multiplikatora**:

$$\hat{\mu}_{k,\text{MLE}} = \frac{1}{N} \sum_{i=1}^N x_k^{(i)} = \frac{N_k}{N}$$

$\Rightarrow$  relativna frekvencija  $k$ -te vrijednosti kategorijalne varijable

- MLE za parametre normalne razdiobe:

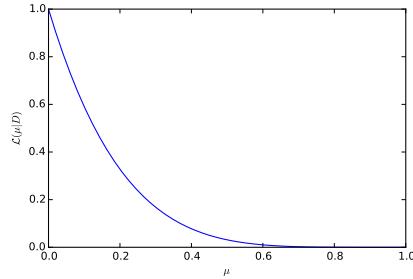
$$\begin{aligned} \ln \mathcal{L}(\mu, \sigma^2 | \mathcal{D}) &= \ln \prod_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma} \exp \left( -\frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right) \\ &= -\frac{N}{2} \ln(2\pi) - N \ln \sigma - \frac{\sum_i (x^{(i)} - \mu)^2}{2\sigma^2} \\ \frac{\partial \ln \mathcal{L}}{\partial \mu} &= 0 \Rightarrow \hat{\mu}_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^N x^{(i)} \\ \frac{\partial \ln \mathcal{L}}{\partial \sigma^2} &= 0 \Rightarrow \hat{\sigma}_{\text{MLE}}^2 = \frac{1}{N} \sum_{i=1}^N (x^{(i)} - \hat{\mu}_{\text{MLE}})^2 \end{aligned}$$

$\Rightarrow$  procjenitelj varijance je pristran (za malen  $N$  preporuča ga se korigirati)

- MLE za parametre multivarijatne normalne razdiobe:

$$\begin{aligned}
\ln \mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\Sigma} | \mathcal{D}) &= \ln \prod_{i=1}^N p(\mathbf{x}^{(i)} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
&= -\frac{nN}{2} \ln(2\pi) - \frac{N}{2} |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{i=1}^N (\mathbf{x}^{(i)} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}) \\
\nabla_{\boldsymbol{\mu}} \ln \mathcal{L} = 0 &\Rightarrow \hat{\boldsymbol{\mu}}_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}^{(i)} \\
\nabla_{\boldsymbol{\Sigma}} \ln \mathcal{L} = 0 &\Rightarrow \hat{\boldsymbol{\Sigma}}_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}_{\text{MLE}})(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}_{\text{MLE}})^T
\end{aligned}$$

- MLE smo već koristili kod izvoda funkcije gubitka za poopćene linearne modele
- Minimizacija empirijske pogreške  $\Leftrightarrow$  MLE procjena za  $\mathbf{w}$  uz odgovarajuću  $p(y|\mathbf{x})$
- MLE je sklon **prenaučenosti** – osobito problematično kada je  $N$  malen
- Npr., bacanje novčića (Bernoullijeva varijabla):  $m = 0, N = 5 \Rightarrow \hat{\mu}_{\text{MLE}} = 0$ :



### 3 Procjenitelj MAP

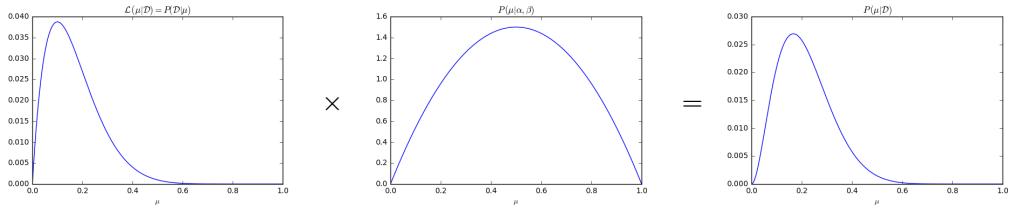
- Želimo kombinirati informacije iz podataka (izglednost  $\boldsymbol{\theta}$ ) s **apriornim znanjem** o  $\boldsymbol{\theta}$
- $p(\boldsymbol{\theta})$  – **apriorna razdioba parametra  $\boldsymbol{\theta}$**  (*parameter prior*)
- **Aposteriorna vjerojatnost parametra  $\boldsymbol{\theta}$**  (Bayesov teorem):

$$p(\boldsymbol{\theta} | \mathcal{D}) = \frac{p(\mathcal{D} | \boldsymbol{\theta}) P(\boldsymbol{\theta})}{p(\mathcal{D})}$$

- Procjenitelj **maksimum a posteriori (MAP)**:

$$\hat{\boldsymbol{\theta}}_{\text{MAP}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} p(\boldsymbol{\theta} | \mathcal{D}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} p(\mathcal{D} | \boldsymbol{\theta}) p(\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \mathcal{L}(\boldsymbol{\theta} | \mathcal{D}) p(\boldsymbol{\theta})$$

- Izglednost  $\times$  Prior  $\propto$  Posterior:



- Rješivo **analitički**, ako  $p(\mathcal{D}|\theta)$  i  $p(\theta)$  odaberemo tako da daju neku standardnu  $p(\theta|\mathcal{D})$
- **Konjugatne distribucije**  $\Leftrightarrow p(\theta|\mathcal{D})$  i  $p(\theta)$  su iste vrste distribucija
- $p(\theta)$  je **konjugatna apriorna distribucija** za  $p(\mathcal{D}|\theta) \Rightarrow p(\theta|\mathcal{D})$  i  $p(\theta)$  su konjugatne
- Svaka  $p(\mathcal{D}|\theta)$  iz **eksponečijalne familije** ima svoju konjugatnu apriornu distribuciju:
  - $p(\mathcal{D}|\theta)$  Bernoullijeva  $\Rightarrow p(\theta)$  beta
  - $p(\mathcal{D}|\theta)$  kategorijkska  $\Rightarrow p(\theta)$  Dirichletova
  - $p(\mathcal{D}|\theta)$  normalna  $\Rightarrow p(\theta)$  normalna
  - $p(\mathcal{D}|\theta)$  multiv. normalna  $\Rightarrow p(\theta)$  multiv. normalna

## 4 Beta-Bernoullijev model

- Konjugatna apriorna distr. za izglednost Bernoullijeve varijable je **beta-distribucija**:
 
$$p(\mu|\alpha, \beta) = \frac{1}{B(\alpha, \beta)} \mu^{\alpha-1} (1-\mu)^{\beta-1}$$

gdje beta-funkcija  $B$  služi za normalizaciju, te  $\alpha > 0$  i  $\beta > 0$

  - $\alpha = \beta = 1 \Leftrightarrow$  uniformna distribucija  $\Rightarrow$  **neinformativna apriorna distribucija**
  - $\alpha > 1, \beta > 1 \Rightarrow$  veća gustoća vjerojatnosti za  $\mu = 0.5$
  - $\alpha > \beta \Rightarrow$  veća gustoća vjerojatnosti za  $\mu \in (0.5, 1)$
  - $\alpha < \beta \Rightarrow$  veća gustoća vjerojatnosti za  $\mu \in (0, 0.5)$
- Maksimizator (mod) beta-distribucije:  $\frac{\alpha-1}{\alpha+\beta-2}$  (za  $\alpha, \beta > 1$ )
- Aposteriorna beta-distribucija:

$$\begin{aligned} p(\mu|\mathcal{D}, \alpha, \beta) &= \mu^m (1-\mu)^{N-m} \frac{1}{B(\alpha, \beta)} \mu^{\alpha-1} (1-\mu)^{\beta-1} \frac{1}{p(\mathcal{D})} \\ &= \mu^{m+\alpha-1} (1-\mu)^{N-m+\beta-1} \frac{1}{B(\alpha, \beta)p(\mathcal{D})} \\ &= \mu^{\alpha'-1} (1-\mu)^{\beta'-1} \frac{1}{B(\alpha', \beta')} \end{aligned}$$

gdje  $\alpha' = m + \alpha$  i  $\beta' = N - m + \beta$

- MAP-procjenitelj odgovara modu aposteriorne beta-distribucije:

$$\hat{\mu}_{\text{MAP}} = \frac{\alpha' - 1}{\alpha' + \beta' - 2} = \frac{m + \alpha - 1}{\alpha + N + \beta - 2}$$

- Za  $N \rightarrow \infty$  procjenom dominiraju podatci; za  $\alpha = \beta = 1$  MAP degenerira u MLE
- MAP provodi **zaglađivanje** (*smoothing*) – preraspoređivanje mase vjerojatnosti
- **Laplaceovo zaglađivanje (Laplace smoothing)** – MAP sa  $\alpha = \beta = 2$ :

$$\hat{\mu}_{\text{MAP}} = \frac{m + 1}{N + 2}$$

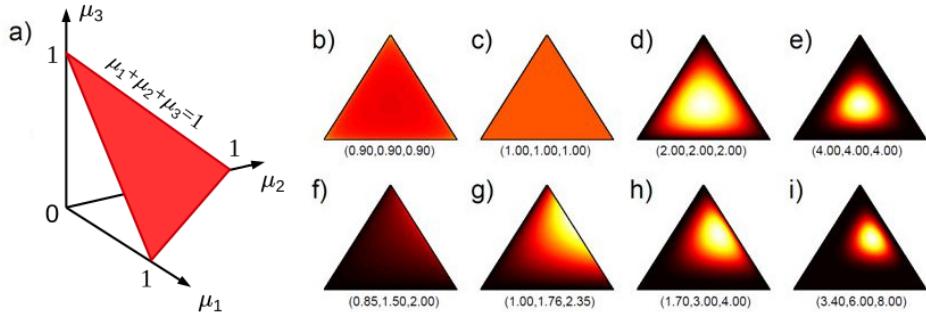
## 5 Dirichlet-kategorijski model

- Konjugatna apriorna distr. za multinomijalnu izglednost je **Dirichletova distribucija**

$$P(\boldsymbol{\mu}|\boldsymbol{\alpha}) = P(\mu_1, \dots, \mu_K | \alpha_1, \dots, \alpha_K) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k=1}^K \mu_k^{\alpha_k - 1}$$

gdje beta-funkcija  $B$  služi za normalizaciju, te  $\alpha_k > 0$

- Dirichletova distribucija je poopćenje beta-distribucije na  $K$  varijabli
- $\mu_k$  leže na  $(K - 1)$ -dimenzijskom **standardnom simpleksu**, tj.  $\sum_{k=1}^K \mu_k = 1$  i  $\mu_k \geq 0$
- Npr., za  $K = 3$ , to je trokut u trodimenzijskome prostoru:



- MAP-procjenitelj odgovara modu Dirichletove distribucije:

$$\hat{\mu}_{k,\text{MAP}} = \frac{\alpha'_k - 1}{\sum_{k=1}^K \alpha'_k - K}$$

gdje  $\alpha'_k = N_k + \alpha_k$  i  $N_k = \sum_i x_k^{(i)}$  (broj nastupanja  $k$ -te vrijednosti)

- Uz  $\alpha_k = 2$ , najvjerojatnija je uniformna distribucija po  $\boldsymbol{\mu}$ , a procjenitelj je:

$$\hat{\mu}_{k,\text{MAP}} = \frac{N_k + 1}{N + K}$$

# 15. Bayesov klasifikator

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

Jan Šnajder, natuknice s predavanja, v1.4

## 1 Pravila vjerojatnosti

- **Pravilo zbroja:**

$$P(x) = \sum_y P(x, y)$$

⇒ **marginalna vjerojatnost** iz **zajedničke vjerojatnosti** (*joint*)

- **Pravilo umnoška:**

$$P(x, y) = P(y|x)P(x) = P(x|y)P(y)$$

- Dva pravila izvedena iz pravila umnoška:

- **Bayesovo pravilo:**

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

- **Pravilo lanca** (*chain rule*):

$$\begin{aligned} P(x_1, \dots, x_n) &= P(x_1)P(x_2|x_1)P(x_3|x_1, x_2) \cdots P(x_n|x_1, \dots, x_{n-1}) \\ &= \prod_{k=1}^n P(x_k|x_1, \dots, x_{k-1}) \end{aligned}$$

⇒ **faktorizacija** zajedničke vjerojatnosti na umnožak **faktora**

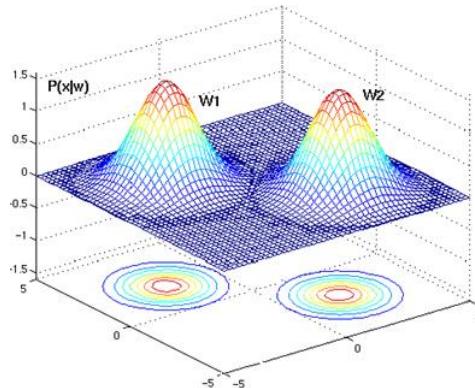
## 2 Bayesov klasifikator

- Model Bayesovog klasifikatora:

$$h_j(\mathbf{x}; \boldsymbol{\theta}) = P(y = j|\mathbf{x}) = \frac{p(\mathbf{x}|y = j)P(y = j)}{\sum_k p(\mathbf{x}|y = k)P(y = k)}$$

- $P(y|\mathbf{x})$  – **aposteriorna vjerojatnost** (*posterior*) klase za zadani primjer
  - $p(\mathbf{x}|y)$  – **izglednost klase** (*class likelihood*) – vjerojatnost primjera u klasi
  - $P(y)$  – **apriorna vjerojatnost klase** (*class prior*)

- Primjer: binarna klasifikacija s Gaussovim gustoćama za izglednosti klasa:



- Faktorizacija  $p(\mathbf{x}, y)$  na  $p(\mathbf{x}|y)P(y)$  omogućava modeliranje složenih distribucija
- Klasifikacija u najvjerojatniju klasu (**MAP-hipoteza**):

$$h(\mathbf{x}) = \operatorname{argmax}_j p(\mathbf{x}|y=j)P(y=j)$$

- Bayesov klasifikator – **parametarski** i **generativni** model

### 3 Generativni modeli

- Modeli modeliraju **zajedničku distribuciju**  $p(\mathbf{x}, y)$
- Na temelju  $p(\mathbf{x}, y)$  računamo  $p(y|\mathbf{x})$  ili neku drugu distribuciju od interesa
- Modeliraju **nastajanje podataka**  $\{(\mathbf{x}^{(i)}, y^{(i)})\}_i$  – tzv. **generativna priča**
- Generativna priča Bayesovog klasifikatora:

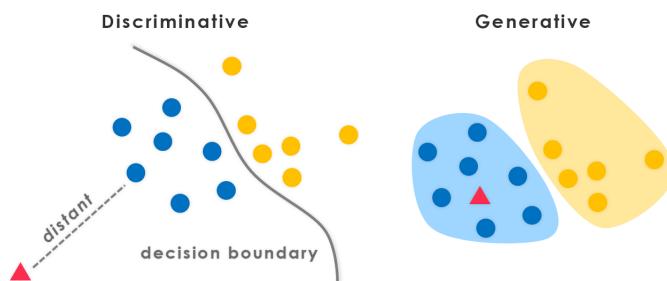
$$P(\mathbf{x}, y) = p(\mathbf{x}|y)P(y)$$

⇒ odabir oznake prema  $P(y)$ , zatim odabir primjera prema  $P(\mathbf{x}|y)$

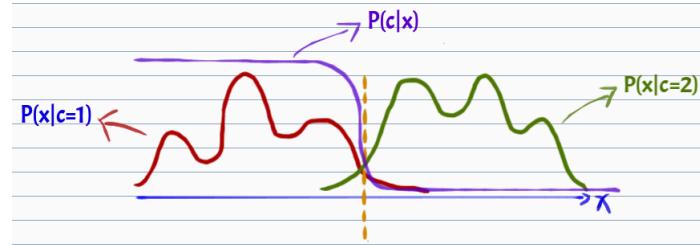
- Složeniji generativni modeli: **Bayesove mreže**, **HMM**, **GMM**, **LDA**
- Usp.: diskriminativni modeli izravno modeliraju  $p(y|\mathbf{x})$ ; npr. logistička regresija:

$$h(\mathbf{x}; \mathbf{w}) = P(y|\mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x})$$

- Diskriminativno vs. generativno:

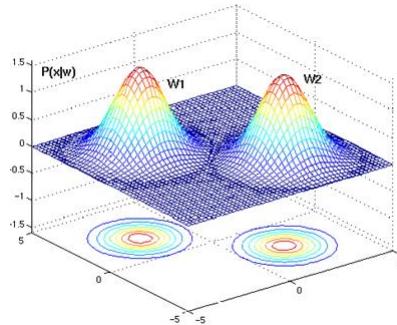


- Prednosti: laka ugradnja stručnog znanja, interpretabilnost/analiza rezultata
- Nedostatci: iziskuju mnogo primjera za učenje, nepotrebna složenost modeliranja
- Primjer: nepotrebna složenost modeliranja zajedničke vjerojatnosti:



## 4 Gaussov Bayesov klasifikator

- Izglednost klase modeliramo **Gaussovom (normalnom) razdiobom**:  $\mathbf{x}|y \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
- $\boldsymbol{\mu}$  predstavlja **prototipni primjer**; primjeri odstupaju od prototipa uslijed **šuma**



- Jednodimenzionalni slučaj:

$$p(x|y = j) = \frac{1}{\sqrt{2\pi}\sigma_j} \exp \left\{ -\frac{(x - \mu_j)^2}{2\sigma_j^2} \right\}$$

- Model (**MAP-hipoteza**):

$$h(x) = \operatorname{argmax}_j p(x, y = j) = \operatorname{argmax}_j p(x|y = j)P(y = j)$$

- Model za klasu  $j$ :

$$h_j(x) = p(x, y = j) = p(x|y = j)P(y = j)$$

- Prelazak u logaritamsku domenu i uklanjanje konstanti:

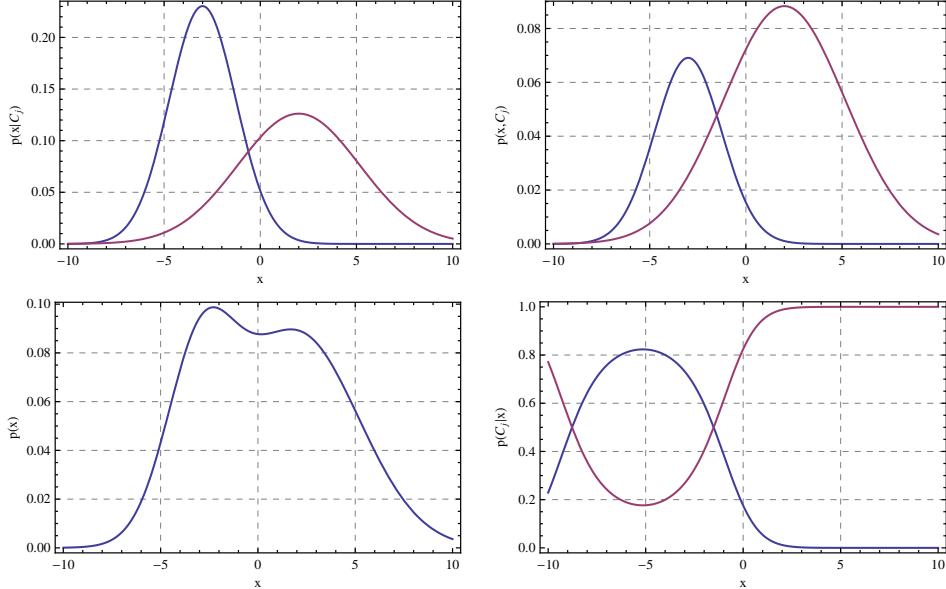
$$\begin{aligned} h_j(x) &= \ln p(x|y = j) + \ln P(y = j) \\ &= -\frac{1}{2} \ln 2\pi - \ln \sigma_j - \frac{(x - \mu_j)^2}{2\sigma_j^2} + \ln P(y = j) \end{aligned}$$

- MLE procjene parametara:

$$\begin{aligned}\hat{\mu}_j &= \frac{1}{N_j} \sum_{i=1}^N \mathbf{1}\{y^{(i)} = j\} x^{(i)} \\ \hat{\sigma}_j^2 &= \frac{1}{N_j} \sum_{i=1}^N \mathbf{1}\{y^{(i)} = j\} (x^{(i)} - \hat{\mu}_j)^2 \\ P(y = j) &= \hat{\mu}'_j = \frac{1}{N} \sum_{j=1}^N \mathbf{1}\{y^{(i)} = j\} = \frac{N_j}{N}\end{aligned}$$

- Primjer:

$$\begin{aligned}p(x|y=1) &\sim \mathcal{N}(-3, 3), P(y=1) = 0.3 \\ p(x|y=2) &\sim \mathcal{N}(2, 10), P(y=2) = 0.7\end{aligned}$$



- Više značajki  $\Rightarrow$  izglednosti modeliramo multivarijatnom normalnom razdiobom:

$$p(\mathbf{x}|y=j) = \frac{1}{(2\pi)^{n/2} |\Sigma_j|^{1/2}} \exp \left( -\frac{1}{2} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_j) \right)$$

- Model za klasu  $j$ :

$$\begin{aligned}h_j(\mathbf{x}) &= \ln p(\mathbf{x}|y=j) + \ln P(y=j) \\ &= -\frac{n}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma_j| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x} - \boldsymbol{\mu}_j) + \ln P(y=j) \\ &\Rightarrow -\frac{1}{2} \ln |\Sigma_j| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x} - \boldsymbol{\mu}_j) + \ln P(y=j)\end{aligned}$$

- MLE procjene parametara:

$$\begin{aligned}\hat{\boldsymbol{\mu}}_j &= \frac{1}{N_j} \sum_{i=1}^N \mathbf{1}\{y^{(i)} = j\} \mathbf{x}^{(i)} \\ \hat{\boldsymbol{\Sigma}}_j &= \frac{1}{N_j} \sum_{i=1}^N \mathbf{1}\{y^{(i)} = j\} (\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}_j)(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}_j)^T \\ \hat{\mu}_j &= \frac{1}{N} \sum_{j=1}^N \mathbf{1}\{y^{(i)} = j\} = \frac{N_j}{N}\end{aligned}$$

- Broj parametara:  $\frac{n}{2}(n+1)K + K \cdot n + K - 1 \Rightarrow \mathcal{O}(n^2)$

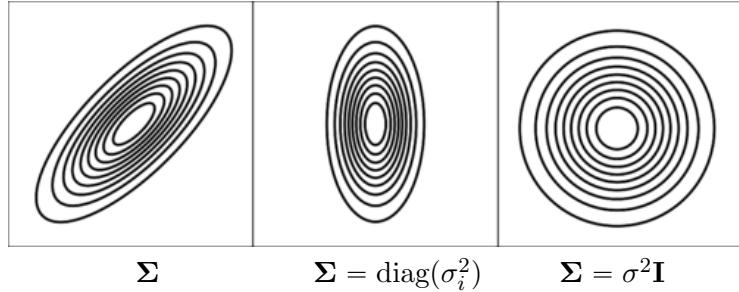
- Granica između dviju klasa:  $h_1(\mathbf{x}) - h_2(\mathbf{x}) = 0$ :

$$\begin{aligned}h_{12}(\mathbf{x}) &= h_1(\mathbf{x}) - h_2(\mathbf{x}) \\ &= -\frac{1}{2} \ln |\boldsymbol{\Sigma}_1| - \frac{1}{2} (\mathbf{x}^T \boldsymbol{\Sigma}_1^{-1} \mathbf{x} - 2\mathbf{x}^T \boldsymbol{\Sigma}_1^{-1} \boldsymbol{\mu}_1 + \boldsymbol{\mu}_1^T \boldsymbol{\Sigma}_1^{-1} \boldsymbol{\mu}_1) + \ln P(y=1) \\ &\quad - \left( -\frac{1}{2} \ln |\boldsymbol{\Sigma}_2| - \frac{1}{2} (\mathbf{x}^T \boldsymbol{\Sigma}_2^{-1} \mathbf{x} - 2\mathbf{x}^T \boldsymbol{\Sigma}_2^{-1} \boldsymbol{\mu}_2 + \boldsymbol{\mu}_2^T \boldsymbol{\Sigma}_2^{-1} \boldsymbol{\mu}_2) + \ln P(y=2) \right) \\ &\quad \dots \mathbf{x}^T (\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1}) \mathbf{x} \dots\end{aligned}$$

$\Rightarrow$  član koji kvadratno ovisi o  $\mathbf{x} \Leftrightarrow$  **nelinearna granica**

## 5 Varijante Gaussovog Bayesovog klasifikatora

- Uvodimo prepostavke na  $\boldsymbol{\Sigma}$  koje pojednostavljaju model

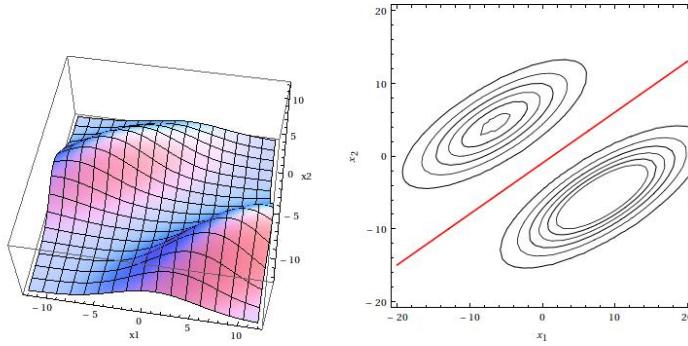


- **Dijeljena kovarijacijska matrica:**  $\hat{\boldsymbol{\Sigma}} = \sum_j \hat{\mu}_j \hat{\boldsymbol{\Sigma}}_j$

- Model za klasu  $j$ :

$$\begin{aligned}h_j(\mathbf{x}) &= \ln p(\mathbf{x}|y=j) + \ln P(y=j) \\ &= -\frac{n}{2} \cancel{\ln 2\pi} - \frac{1}{2} \cancel{\ln |\boldsymbol{\Sigma}|} - \frac{1}{2} (\mathbf{x}^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - 2\mathbf{x}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j + \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j) + \ln P(y=j)\end{aligned}$$

- Model je linearan  $\Rightarrow$  **linearna granica** između klasa
  - Broj parametara:  $\frac{n}{2}(n+1) + nK + K - 1 \Rightarrow \mathcal{O}(n^2)$



- **Dijeljena i dijagonalna kovarijacijska matrica:**  $\Sigma = \text{diag}(\sigma_i^2)$

- Vrijedi  $|\Sigma| = \prod_i \sigma_i^2$  i  $\Sigma^{-1} = \text{diag}(1/\sigma_i^2)$
- Izglednost klase:

$$\begin{aligned}
p(\mathbf{x}|y=j) &= \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_j)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_j)\right) \\
&= \frac{1}{(2\pi)^{n/2} \prod_{i=1}^n \sigma_i} \exp\left(-\frac{1}{2} \sum_{i=1}^n \left(\frac{x_i - \mu_{ij}}{\sigma_i}\right)^2\right) \\
&= \prod_{i=1}^n \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left\{-\frac{1}{2} \left(\frac{x_i - \mu_{ij}}{\sigma_i}\right)^2\right\} \\
&= \prod_{i=1}^n \mathcal{N}(\mu_{ij}, \sigma_i^2) = \prod_{i=1}^n p(x_i|y)
\end{aligned}$$

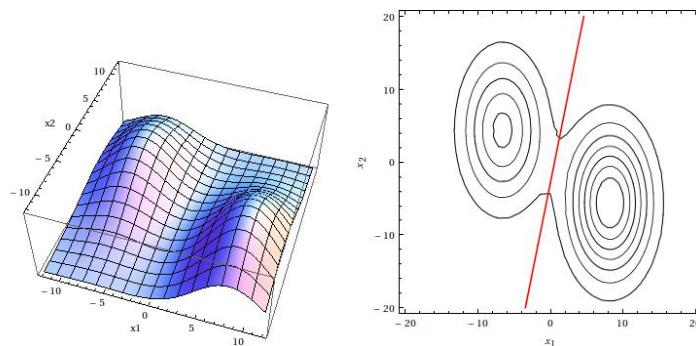
$\Rightarrow$  **uvjetna nezavisnost** značajki  $\Rightarrow$  **Gaussov naivan Bayesov klasifikator**

- $x_k \perp x_j | y \Rightarrow \text{Cov}(x_k|y, x_j|y) = 0 \Leftrightarrow p(\mathbf{x}|y) = \prod_k p(x_k|y)$
- Model za klasu  $j$ :

$$\begin{aligned}
h_j(\mathbf{x}) &= \ln p(\mathbf{x}|y=j) + \ln P(y=j) \\
&= \sum_{i=1}^n \ln \frac{1}{\sqrt{2\pi}\sigma_i} + \sum_{i=1}^n \left(-\frac{1}{2} \left(\frac{x_i - \mu_{ij}}{\sigma_i}\right)^2\right) + \ln P(y=j)
\end{aligned}$$

$\Rightarrow$  **normirana euklidska udaljenost** između primjera  $\mathbf{x}$  i prototipa klase  $\boldsymbol{\mu}_j$

- Broj parametara:  $n + n \cdot K + K - 1 \Rightarrow \mathcal{O}(n)$

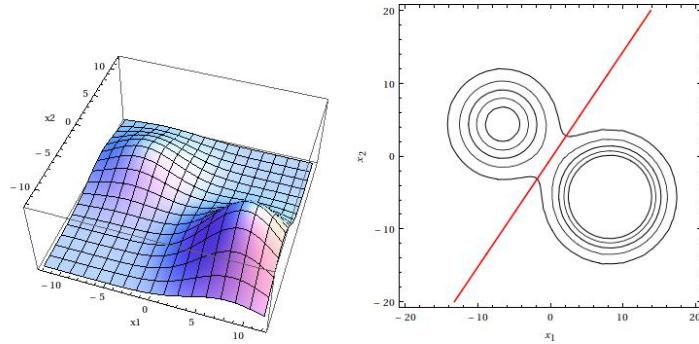


- **Izotropna kovarijacijska matrica:**  $\Sigma = \sigma^2 \mathbf{I}$

– Model za klasu  $j$ :

$$h_j(\mathbf{x}) = -\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu_{ij})^2 + \ln P(y = j)$$

– Broj parametara:  $1 + Kn + K - 1 \Rightarrow \mathcal{O}(n)$



- Druge varijante:

| Pretpostavka                        | Kov. matrica                             | Broj parametara  |
|-------------------------------------|------------------------------------------|------------------|
| Različite, hiperelipsoidi           | $\Sigma_j$                               | $Kn(n+1)/2 + Kn$ |
| Dijeljena, hiperelipsoidi           | $\Sigma$                                 | $n(n+1)/2 + Kn$  |
| Različite, poravnati hiperelipsoidi | $\Sigma_j = \text{diag}(\sigma_{i,j}^2)$ | $2Kn$            |
| Dijeljena, poravnati hiperelipsoidi | $\Sigma = \text{diag}(\sigma_i^2)$       | $n + Kn$         |
| Različite, hipersfere               | $\Sigma_j = \sigma_j^2 \mathbf{I}$       | $K + Kn$         |
| Dijeljena, hipersfere               | $\Sigma = \sigma^2 \mathbf{I}$           | $1 + Kn$         |

- Odabir modela: **unakrsnom provjerom**

# 16. Bayesov klasifikator II

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

Jan Šnajder, natuknice s predavanja, v1.4

## 1 Bayesov klasifikator vs. logistička regresija

- Ideja: pokazati da logistička regresija i Bayesov klasifikator izračunavaju isti  $P(y|\mathbf{x})$
- Model **logističke regresije**:

$$h(\mathbf{x}; \mathbf{w}) = P(y = 1|\mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x})$$

- Aposteriorna vjerojatnost za **kontinuirani Bayesov klasifikator** (za dvije klase):

$$\begin{aligned} P(y = 1|\mathbf{x}) &= \frac{p(\mathbf{x}|y = 1)P(y = 1)}{p(\mathbf{x}|y = 1)P(y = 1) + p(\mathbf{x}|y = 2)P(y = 2)} = \frac{1}{1 + \frac{p(\mathbf{x}|y=2)P(y=2)}{p(\mathbf{x}|y=1)P(y=1)}} = \\ &= \frac{1}{1 + \exp\left(\ln \frac{p(\mathbf{x}|y=2)P(y=2)}{p(\mathbf{x}|y=1)P(y=1)}\right)} = \frac{1}{1 + \exp(-\alpha)} = \sigma(\alpha) \end{aligned}$$

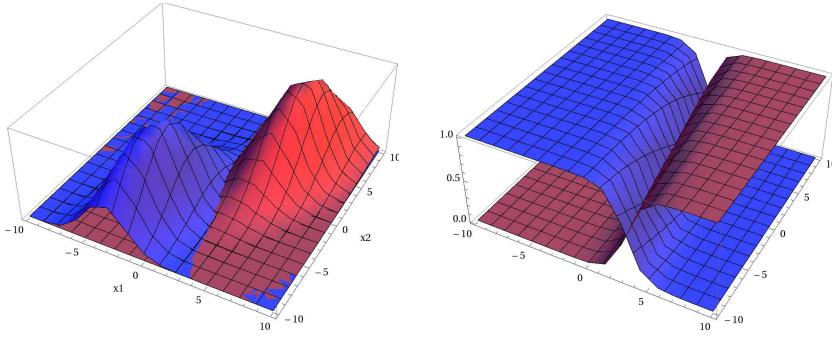
gdje

$$\alpha = \ln \frac{p(\mathbf{x}|y = 1)P(y = 1)}{p(\mathbf{x}|y = 2)P(y = 2)} = \underbrace{\ln p(\mathbf{x}|y = 1)P(y = 1)}_{h_1(\mathbf{x})} - \underbrace{\ln p(\mathbf{x}|y = 2)P(y = 2)}_{h_2(\mathbf{x})}$$

- Možemo li  $\alpha$  prikazati kao linearu kombinaciju težina,  $\alpha = \mathbf{w}^T \mathbf{x}$ ?
- Da, ako prepostavimo **dijeljenu kovarijacijsku matricu**:

$$\begin{aligned} \alpha &= h_1(\mathbf{x}) - h_2(\mathbf{x}) \\ &= \mathbf{x}^T \underbrace{\Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}_{\mathbf{w}} - \underbrace{\frac{1}{2}\boldsymbol{\mu}_1^T \Sigma^{-1} \boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2^T \Sigma^{-1} \boldsymbol{\mu}_2 + \ln \frac{P(y = 1)}{P(y = 2)}}_{w_0} = \mathbf{w}^T \mathbf{x} + w_0 \end{aligned}$$

$\Rightarrow$  logistička regresija istovjetna je Bayesovom klasifikatoru s dijeljenom  $\Sigma$



- Broj parametara:  $\frac{n}{2}(n+1) + 2n + 1$  (Bayes) vs.  $n+1$  (logistička regresija)  
 $\Rightarrow$  diskriminativan model daje istu predikciju, ali s manje parametara

## 2 Naivan Bayesov klasifikator

- $\mathbf{x} = (x_1, \dots, x_n)$  ima  $\prod_{k=1}^n K_k$  mogućih vrijednosti
- $p(\mathbf{x}|y)$  kao kategorička razdioba od  $\mathbf{x}$   $\Rightarrow$  **previše parametara i nema generalizacije**
- Pojednostavljenje uvođenjem **induktivnih prepostavki** u obliku uvjetnih nezavisnosti
- Prepostavka: u svakoj klasi, svaka značajka uvjetno je nezavisna od svih drugih:

$$x_k \perp (x_1, \dots, x_{k-1}) | y \Leftrightarrow P(x_k | x_1, \dots, x_{k-1}, y) = P(x_k | y)$$

- Faktorizacija uz tu prepostavku:

$$P(x_1, \dots, x_n | y) = \prod_{k=1}^n P(x_k | x_1, \dots, x_{k-1}, y) = \prod_{k=1}^n P(x_k | y)$$

- **Naivan Bayesov klasifikator** (*Naïve Bayes classifier*):

$$h(x_1, \dots, x_n) = \operatorname{argmax}_j P(y=j) \prod_{k=1}^n P(x_k | y=j)$$

- Procjena parametara – MLE za  $P(y)$  i MAP za  $P(\mathbf{x}|y)$ :

$$\begin{aligned} P(y=j) &= \hat{\mu}_j = \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{y^{(i)} = j\} = \frac{N_j}{N} \\ P(x_k | y=j) &= \hat{\mu}_{k,j} = \frac{\sum_{i=1}^N \mathbf{1}\{x_k^{(i)} = x_k \wedge y^{(i)} = j\} + 1}{\sum_{i=1}^N \mathbf{1}\{y^{(i)} = j\} + K_k} = \frac{N_{kj} + 1}{N_j + K_k} \end{aligned}$$

gdje je  $N_j$  broj primjera u klasi  $j$ , a  $N_k$  broj vrijednosti značajke  $k$

- Broj parametara:  $\sum_{k=1}^n (K_k - 1) \cdot K + K - 1$
- Prepostavka uvjetne nezavisnosti uglavnom ne vrijedi, no model u praksi radi dobro

### 3 Uvjetna nezavisnost

- **Uvjetna nezavisnost**  $X$  i  $Y$  uz dani  $Z$  – notacija:  $X \perp Y | Z$ :

$$\begin{aligned} P(X, Y | Z) &= P(X|Z)P(Y|Z) \\ P(X|Y, Z) &= P(X|Z) \\ P(Y|X, Z) &= P(Y|Z) \end{aligned}$$

### 4 Polunaivan Bayesov klasifikator

- Ideja: **združiti** (ne faktorizirati) varijable koje nisu uvjetno nezavisne
- Npr., ako  $x_2 \not\perp x_3 | y$ :  
$$P(x_1, x_2, x_3, y) = P(x_1|y)P(x_2|x_3|y)P(x_3|y)P(y)$$

$\Rightarrow$  slabije pretpostavke  $\Leftrightarrow$  složeniji model  $\Leftrightarrow$  više parametara
- Broj mogućih združivanja  $\Leftrightarrow$  broj particija  $n$ -članog skupa  $\Leftrightarrow$  **Bellov broj**  $B_n$
- Previše kombinacija  $\Rightarrow$  **heurističko pretraživanje** na temelju **kriterija združivanja**
- Dva pristupa:
  - **točnost modela** (unakrsna provjera) – algoritam FSSJ
  - **procjena zavisnosti varijabli** – algoritmi TAN i  $k$ -DB

#### Algoritam FSSJ

1. Inicijaliziraj  $X = \emptyset$ . Početna faktorizacija:

$$P(x_1, \dots, x_n, y) = P(x_1) \cdots P(x_n)P(y)$$

$$P(y|x_1, \dots, x_n) = P(y)$$

Klasificiraj primjere iz skupa za provjeru:  $y^* = \operatorname{argmax}_j P(y = j)$

2. Za svaku varijablu  $x_i \notin X$  koja još nije uključena u model, razmotri:

- (a) Uključi  $x_i$  kao uvjetno nezavisnu u odnosu na ostale varijable za danu klasu  $j$
- (b) Uključi  $x_i$  tako da se ona doda u zajednički faktor s nekom već uključenom varijablom

3. Izaberi  $x_i$  i opciju koja minimizira pogrešku generalizacije

4. Ponavljam od koraka (2) do konvergencije pogreške

- **Uzajamna informacija** – zavisnost varijabli kao odstupanje  $P(x, y)$  od  $P(x)P(y)$ :

$$I(x, y) = D_{\text{KL}}(P(x, y) || P(x)P(y)) = \sum_{x,y} P(x, y) \ln \frac{P(x, y)}{P(x)P(y)}$$

$$\Rightarrow I(x, y) = 0 \Leftrightarrow x \perp\!\!\!\perp y, \quad I(x, y) > 0 \Leftrightarrow x \not\perp\!\!\!\perp y$$

- $D_{\text{KL}}(P || Q)$  – **Kullback-Leiblerova divergencija** (odstupanje) distribucije  $P$  od  $Q$ :

$$D_{\text{KL}}(P || Q) = \sum_x P(x) \ln \frac{P(x)}{Q(x)}$$

$$\Rightarrow \text{relativna entropija } P(x) \text{ u odnosu na } Q(x)$$

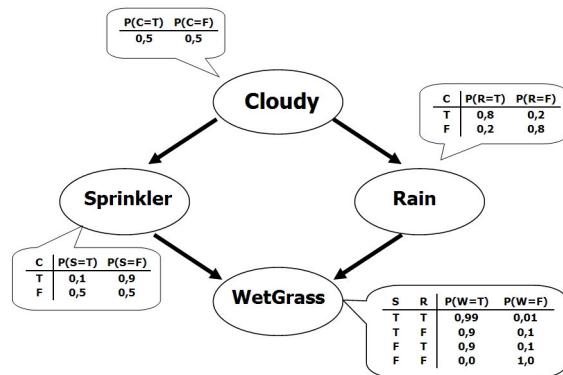
# 17. Probabilistički grafički modeli

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

Jan Šnajder, natuknice s predavanja, v1.4

## 1 Uvod

- **Probabilistički grafički model (PGM)** – sažet zapis zajedničke distrib. pomoću grafa
- Čvorovi grafa su varijable, bridovi su zavisnosti između varijabli

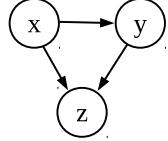


- Svrha: **probabilističko zaključivanje** (određivanje vrijednosti neopažanih varijabli)
- Tri aspekta PGM-a: (1) **reprezentacija**, (2) **zaključivanje** i (3) **učenje**
- Reprezentacija:
  - usmjereni aciklički graf  $\Rightarrow$  **Bayesove mreže**
  - neusmjereni graf  $\Rightarrow$  **Markovljeve mreže**
- Zaključivanje – određivanje vrijednosti nepažanih varijabli na temelju opažanih
- Učenje – procjena parametara ili učenje strukture mreže na temelju podatka
- Mi se fokusiramo na Bayesove mreže

## 2 Bayesove mreže: reprezentacija

- Usmjereni aciklički graf (*directed acyclic graph, DAG*)

- Bridovi povezuju varijablu koja uvjetuje s varijablom koja je uvjetovana
- Npr.,  $p(x, y, z) = p(x)p(y|x)p(z|x, y)$

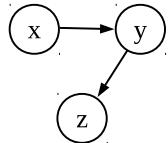


- Bez pretpostavki o uvjetnoj nezavisnosti:

$$\begin{aligned} p(x_1, \dots, x_n) &= p(x_1)p(x_2|x_1)p(x_3|x_1, x_2) \cdots p(x_n|x_1, \dots, x_{n-1}) \\ &= \prod_{k=1}^n p(x_k|x_1, \dots, x_{k-1}) \end{aligned}$$

$\Rightarrow$  pravilo lanca  $\Leftrightarrow$  potpuno povezana Bayesova mreža

- Pretpostavke o uvjetnoj nezavisnosti uklanjaju bridove i pojednostavljaju mrežu
- Npr., ako  $x \perp z | y$ , onda  $p(x, y, z) = p(x)p(y|x)p(z|y)$ :



- Formalno, zajednička distribucija definirana Bayesovom mrežom je:

$$p(\mathbf{x}) = \prod_{k=1}^n p(x_k | \text{pa}(x_k))$$

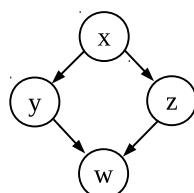
dje  $\text{pa}(x_k)$  označava **čvorove roditelje** čvora  $x_k$

- Čvorovi su poredani u **topološki uredaj** (roditelji dolaze prije djece)
- Svaki DAG ima barem jedan topološki uredaj
- **Uredajno Markovljevo svojstvo** (UMS): svaki čvor  $x_k$  ovisi samo o roditeljima:

$$x_k \perp \text{pred}(x_k) \setminus \text{pa}(x_k) \mid \text{pa}(x_k)$$

dje je  $\text{pred}(x_k)$  skup prethodnika čvora  $x_k$  po topološkom uređaju

- Primjer:  $p(x, y, z, w) = p(x)p(y|x)p(z|x)p(w|y, z)$



- Faktorizacija:

$$\begin{aligned}
 p(x)p(y|x)p(z|x)p(w|y,z) &= p(x,y)p(z|x)p(w|y,z) \\
 y \perp z | x &\Rightarrow p(x,y)p(z|x, \textcolor{red}{y})p(w|y,z) \\
 &= p(x,y,z)p(w|y,z) \\
 x \perp w | y, z &\Rightarrow p(x,y,z)p(w|\textcolor{red}{x},y,z) \\
 &= p(x,y,z,w)
 \end{aligned}$$

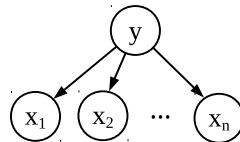
- Uvjetne nezavisnosti proizlaze iz UMS-a:

$$\begin{aligned}
 x_k \perp \text{pred}(x_k) \setminus \text{pa}(x_k) \mid \text{pa}(x_k) \\
 y \perp \{x\} \setminus \{x\} \mid \{x\} \\
 z \perp \{x, y\} \setminus \{x\} \mid \{x\} \Rightarrow y \perp z | x \\
 w \perp \{x, y, z\} \setminus \{y, z\} \mid \{x, y\} \Rightarrow x \perp w | y, z
 \end{aligned}$$

### 3 Primjeri Bayesovih mreža

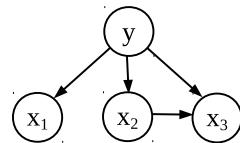
- **Naivan Bayesov klasifikator:**

$$P(\mathbf{x}, y) = P(y) \prod_{i=1}^n P(x_i|y)$$



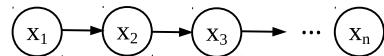
- Polunaivan Bayesov klasifikator. Npr.:

$$P(x_1, x_2, x_3, y) = P(x_1|y)P(x_2|x_3, y)P(y) = P(x_1|y)P(x_2|y)P(x_3|x_2, y)P(y)$$



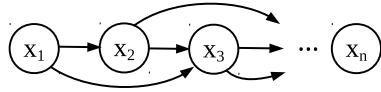
- Markovljev model prvog reda – za modeliranje slijednih podataka (npr., tekst):

$$p(\mathbf{x}) = p(x_1) \prod_{k=2}^n p(x_k|x_{k-1})$$



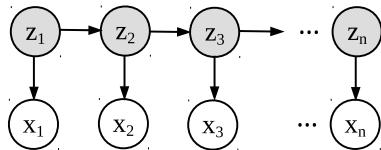
- Markovljev model drugog reda – modelira dulje zavisnosti:

$$p(\mathbf{x}) = p(x_1)p(x_2|x_1) \prod_{k=3}^n p(x_k|x_{k-1}x_{k-2})$$



- Problem: eksplisitno modeliranje duljih zavisnosti povećava složenost modela
- **Skriveni Markovljev model** (*Hidden Markov Model, HMM*):

$$p(\mathbf{x}, \mathbf{z}) = p(z_1)p(x_1|z_1) \prod_{k=2}^n p(z_k|z_{k-1})p(x_k|z_k)$$



⇒ indirektno modelira dulje zavisnosti preko skrivenih varijabli  $\mathbf{z}$

## 4 D-separacija

- Ispitivanje uvjetne nezavisnosti dviju varijabli uz zadane druge varijable
- **D-separacija**: analiziramo povezanost staze u grafu između dva čvora
- Tri pravila: račvanje, lanac, sraz
- (1) **Račvanje**:  $x \leftarrow z \rightarrow y$ 
  - UMS:  $y \perp\!\!\!\perp x | z \Leftrightarrow x \perp\!\!\!\perp y | z$
  - ⇒ ako je varijabla  $z$  opažena, onda su čvorovi odvojeni, inače su povezani
- (2) **Lanac**:  $x \rightarrow z \rightarrow y$ 

$$p(x, y, z) = p(x)p(z|x)p(y|z)$$
  - UMS:  $y \perp\!\!\!\perp x | z \Leftrightarrow x \perp\!\!\!\perp y | z$
  - ⇒ ako je varijabla  $z$  opažena, onda su čvorovi odvojeni, inače su povezani
- (3) **Sraz**:  $x \rightarrow z \leftarrow y$ 

$$p(x, y, z) = p(x)p(y)p(z|x, y)$$
  - UMS:  $y \perp\!\!\!\perp x | \emptyset$
  - ⇒ ako je varijabla  $z$  **neopažena**, onda su čvorovi odvojeni, inače su povezani
- Kod sraza varijable  $x$  i  $y$  se “natječu” za objašnjavanje (uzorkovanje) varijable  $z$
- **Efekt objašnjavanja** (*explaining away*): opažanje  $x$  i  $z$  smanjuje vjerojatnost za  $y$ :

$$p(x|z) \neq p(x|y, z) \Leftrightarrow x \not\perp\!\!\!\perp y | z$$

- Primjer 1: Bacanje dva novčića ( $x, y \in \{0, 1\}$ ) i opažanje njihove sume ( $z = x + y$ )
- Primjer 2:  $x$  – mononukleoza,  $y$  – upala grla,  $z$  – visoka temperatura

### D-separacija čvorova

Raspolažemo skupom varijabli  $E$  koje su opažene.

Za **stazu**  $P$  od čvora  $x$  do čvora  $y$  kažemo da je **d-odvojena (d-separated)** akko vrijedi **barem jedno** od sljedećeg:

- $P$  sadrži **lanac**  $x \rightarrow z \rightarrow y$  ili  $x \leftarrow z \leftarrow y$  i  $z \in E$
- $P$  sadrži **račvanje**  $x \leftarrow z \rightarrow y$  i  $z \in E$
- $P$  sadrži **sraz**  $x \rightarrow z \leftarrow y$  i varijabla  $z$  **nije** u  $E$  i nijedan sljedbenik od  $z$  nije u  $E$

Za **par čvorova**  $x$  i  $y$  kažemo da su čvorovi  $x$  i  $y$  d-separirani za dani  $E$  ako su **sve staze** između ta dva čvora d-separirane za dani  $E$ .

Čvorovi  $x$  i  $y$  su d-separirani za dani  $E$  **akko** su uvjetno nezavisni za dani  $E$ .

- Implementacija: algoritam **Bayesove kuglice** (*Bayes-Ball*)

# 18. Probabilistički grafički modeli II

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

Jan Šnajder, natuknice s predavanja, v1.4

## 1 Zaključivanje

- Primjer trave i prskalice (v. predavanje 17):

$$p(c, s, r, w) = p(c)p(s|c)p(r|c)p(w|s, r)$$

- Upit: Ako je trava mokra ( $w = 1$ ), koja je vjerojatnost kiše ( $r$ ) i prskalice ( $s$ )?

$$\begin{aligned} P(s = 1|w = 1) &= \frac{P(s = 1, w = 1)}{P(w = 1)} \\ &= \frac{\sum_{c,r} P(c, s = 1, r, w = 1)}{\sum_{c,r,s} P(c, s, r, w = 1)} = 0.2781/0.6471 = 0.43 \\ P(r = 1|w = 1) &= \frac{P(r = 1, w = 1)}{P(w = 1)} \\ &= \frac{\sum_{c,s} P(c, s, r = 1, w = 1)}{\sum_{c,r,s} P(c, s, r, w = 1)} = 0.4851/0.6471 = 0.708 \end{aligned}$$

gdje je  $P(w = 1)$  **vjerojatnost dokaza**

- Dvije vrste upita: (1) posteriorni upiti i (2) MAP-upiti
- **Posteriorni upit** je izračun uvjetne vjerojatnosti:

$$p(\mathbf{x}_q|\mathbf{x}_o) = \frac{\sum_{\mathbf{x}_n} p(\mathbf{x}_q, \mathbf{x}_o, \mathbf{x}_n)}{p(\mathbf{x}_o)} = \frac{\sum_{\mathbf{x}_n} p(\mathbf{x}_q, \mathbf{x}_o, \mathbf{x}_n)}{\sum_{\mathbf{x}'_n, \mathbf{x}'_q} p(\mathbf{x}'_q, \mathbf{x}_o, \mathbf{x}'_n)}$$

gdje su  $\mathbf{x}_q$  varijable upita,  $\mathbf{x}_o$  su opažene varijable, a  $\mathbf{x}_n$  varijable smetnje (*nuisance*)

- **MAP-upiti** – najvjerojatnija vrijednost varijabli upita:

$$\mathbf{x}_q^* = \operatorname{argmax}_{\mathbf{x}_q} \sum_{\mathbf{x}_n} p(\mathbf{x}_q, \mathbf{x}_o, \mathbf{x}_n)$$

## 2 Zaključivanje: eliminacija varijabli

- Odgovaranje upita iziskuje konstrukciju  $p(\mathbf{x}_q, \mathbf{x}_o, \mathbf{x}_n)$  pa marginalizaciju/normalizaciju
- Velik  $n \Rightarrow$  **kombinatorna eksplozija**  $\Rightarrow$  NP-složen problem
- Poništava prednost Bayesove mreže (sažet zapis zajedničke distribucije)
- Alternative: **egzaktno zaključivanje** i **približno zaključivanje**
- **Eliminacija varijabli** – egzaktno zaključivanje pomoću dinamičkog programiranja
- **Eliminacija varijabli zbroj-umnožak** – potiskivanje marginalizacije što dublje:

$$\begin{aligned}
p(w) &= \sum_c \sum_s \sum_r p(c, s, r, w) \\
&= \sum_c \sum_s \sum_r p(c)p(s|c)p(r|c)p(w|s, r) \\
&= \sum_s \sum_r p(w|s, r) \underbrace{\sum_c p(c)p(s|c)p(r|c)}_{t_1(s, r)} \\
&= \sum_s \underbrace{\sum_r p(w|s, r)t_1(s, r)}_{t_2(s, w)} \\
&= \sum_s t_2(s, w) \\
&= t_3(w)
\end{aligned}$$

- Varijante algoritma za skriveni Markovljev model (HMM):
  - eliminacija varijabli  $\Rightarrow$  **algoritam naprijed nazad** (*forward-backward algorithm*)
  - MAP-upiti  $\Rightarrow$  **Viterbijev algoritam**
- Za općenite Bayesove mreže eliminacija varijabli je presložena
- Alternativa: približno zaključivanje – **propagacijski algoritmi** i **metode uzorkovanja**

## 3 Zaključivanje: metode uzorkovanja

- Ideja: procjena distribucije na temelju uzorka
- Ako uzorkujemo uzorke  $\mathbf{x} \sim P(\mathbf{x})$ , očekivanje je:

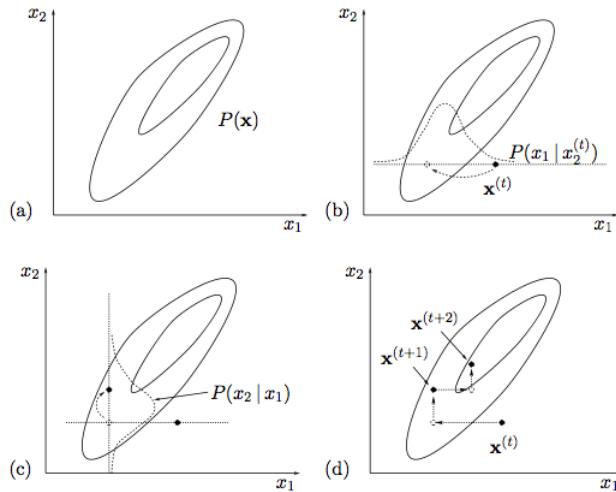
$$P(\mathbf{x} = x) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{\mathbf{x} = x\}$$

- Najjjednostavnija metoda: **unaprijedno uzorkovanje** (*forward sampling*)

– Uzorkovanje varijable za varijablim, prema topološkom uređaju mreže

- Problem: želimo uzorkovati iz uvjetne vjerojatnosti  $P(\mathbf{x}_q | \mathbf{x}_o)$
- **Uzorkovanje s odbijanjem** (*rejection sampling*)
  - Uzorkovanje unaprijed i odbijanje  $\mathbf{x}$  za koje  $\mathbf{x}_o$  nisu na željenim vrijednostima
  - Problem: neučinkovito, osobito ako je vjerojatnost dokaza  $P(\mathbf{x}_o)$  malena
- **Uzorkovanje po važnosti** (*importance sampling*)
  - Postavljanje  $\mathbf{x}_o$  na željene vrijednosti, unaprijedno uzorkovanje i korekcija očekivanja
  - Problem: loša kvaliteta procjene, pogotovo ako su  $\mathbf{x}_o$  pri dnu Bayesove mreže
- **Gibbsovo uzokovanje** (*Gibbs sampling*)
  - Postupak iz porodice **Markov Chain Monte Carlo (MCMC)**
  - Krenuvši od slučajnog vektora  $\mathbf{x}$ , uzorkujemo ciklički varijablu po varijablu

$$\begin{aligned}
 \mathbf{x}^0 &\sim p(x_1^0, x_2^0, x_3^0) \quad \Rightarrow \text{početni vektor (npr., unaprijednim uzorkovanjem)} \\
 x_1^1 &\sim p(x_1 | x_2^0, x_3^0) \\
 x_2^1 &\sim p(x_2 | x_1^1, x_3^0) \\
 x_3^1 &\sim p(x_3 | x_1^1, x_2^1) \quad \Rightarrow \text{vektor } \mathbf{x}^1 = (x_1^1, x_2^1, x_3^1) \\
 x_1^2 &\sim p(x_1 | x_2^1, x_3^1) \\
 x_2^2 &\sim p(x_2 | x_1^2, x_3^1) \\
 x_3^2 &\sim p(x_3 | x_1^2, x_2^2) \quad \Rightarrow \text{vektor } \mathbf{x}^2 = (x_1^2, x_2^2, x_3^2) \\
 &\vdots
 \end{aligned}$$



## 4 Učenje

- PGM-ovi su probabilistički modeli  $\Rightarrow$  učenje se svodi na **procjenu parametara  $\theta$**
- MLE, MAP ili bayesovska procjena

- Log-izglednost za općenitu Bayesovu mrežu:

$$\begin{aligned}
\ln \mathcal{L}(\boldsymbol{\theta} | \mathcal{D}) &= \ln p(\mathcal{D} | \boldsymbol{\theta}) \\
&= \ln p(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)} | \boldsymbol{\theta}) \\
&= \ln \prod_{i=1}^N p(\mathbf{x}^{(i)} | \boldsymbol{\theta}) \\
&= \ln \prod_{i=1}^N \prod_{k=1}^n p(x_k^{(i)} | \text{pa}(x_k^{(i)}), \boldsymbol{\theta}_k) \\
&= \ln \prod_{k=1}^n \prod_{i=1}^N p(x_k^{(i)} | \text{pa}(x_k^{(i)}), \boldsymbol{\theta}_k) \\
&= \sum_{k=1}^n \sum_{i=1}^N \ln p(x_k^{(i)} | \text{pa}(x_k^{(i)}), \boldsymbol{\theta}_k)
\end{aligned}$$

- MLE procjena za  $k$ -ti čvor:

$$\boldsymbol{\theta}_k^* = \underset{\boldsymbol{\theta}_k}{\text{argmax}} \sum_{i=1}^N \ln p(x_k^{(i)} | \text{pa}(x_k^{(i)}), \boldsymbol{\theta}_k)$$

- MAP procjena za  $k$ -ti čvor:

$$\boldsymbol{\theta}_k^* = \underset{\boldsymbol{\theta}_k}{\text{argmax}} \left( \sum_{i=1}^N \ln p(x_k^{(i)} | \text{pa}(x_k^{(i)}), \boldsymbol{\theta}_k) + \ln p(\boldsymbol{\theta}_k) \right)$$

- MAP-procjena za kategorijsku razdiobu (Dirichlet-kategorijski model uz  $\alpha = 2$ ):

$$\begin{aligned}
\hat{\mu}_{k,j,l} &= \frac{N_{kjl} + 1}{N_{kj} + K_k} \\
N_{kjl} &= \sum_{i=1}^N \mathbf{1}\{\mathbf{x}_{\text{pa}(x_k)}^{(i)} = j \wedge x_k^{(i)} = l\} \\
N_{kj} &= \sum_l N_{kjl}
\end{aligned}$$

gdje je  $K_k$  broj mogućih vrijednosti varijable  $x_k$

- Primjer: MAP procjena za čvor  $w$  u mreži s travom i prskalicom (v. predavanje 17):

$$P(w|s, r) = \frac{\sum_{i=1}^N \mathbf{1}\{x_s^{(i)} = s \wedge x_r^{(i)} = r \wedge x_w^{(i)} = w\} + 1}{\sum_{i=1}^N \mathbf{1}\{x_s^{(i)} = s \wedge x_r^{(i)} = r\} + 2}$$

- Modeli sa skrivenim varijablama (npr., HMM, GMM)  $\Rightarrow$  tzv. **nepotpuni podatci**
  - Log-izglednost se ne dekomponira po strukturi grafa  
 $\Rightarrow$  MLE nema rješenje u zatvorenoj formi
    - Učenje pomoću **algoritma maksimizacije očekivanja** ili **gradijentnim usponom**
- **Učenje strukture mreže:**
  - Polunaivan Bayesov klasifikator: v. 4.3.2–4.3.4 u skripti
  - Učenje općenite strukture Bayesove mreže: **algoritam K2**

# 19. Grupiranje

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

Jan Šnajder, natuknice s predavanja, v1.1

## 1 Nenadzirano učenje

- Raspolažemo skupom **neoznačenih primjera** (*unlabeled data*):  $\mathcal{D} = \{\mathbf{x}^{(i)}\}_{i=1}^N$
- Primjerni su neoznačeni jer ih (1) ne znamo označiti ili (2) označavanje je preskupo
- Osnovni zadatci nenadziranog učenja:
  - grupiranje (*clustering*)
  - procjena gustoće (*density estimation*)
  - otkrivanje novih/stršećih vrijednosti (*novelty/outlier detection*)
  - smanjenje dimenzionalnosti (*dimensionality reduction*)
- **Polunadzirano učenje:** većina primjera je neoznačena

## 2 Grupiranje

- Razdjeljivanje primjera u grupe (*clusters*), tako da su **slični** primjeri u istoj grupi
- Nalaženje “prirodnih” (intrinzičnih) grupa u skupu neoznačenih podataka
- Vrste grupiranja: **čvrsto/meko, partijsko/hijerarhijsko**
- Primjene: (1) istraživanje podataka, (2) kompresija, (3) polunadzirano učenje
- Grupiranje primjera / grupiranje značajki / bi-clustering

## 3 Algoritam K-sredina

- Particijsko grupiranje u  $K$  čvrstih grupa ( $K$  je unaprijed određen)
- **Funkcija pogreške** (kriterijska funkcija):

$$J = \sum_{k=1}^K \sum_{i=1}^N b_k^{(i)} \|\mathbf{x}^{(i)} - \boldsymbol{\mu}_k\|^2$$

gdje je  $\boldsymbol{\mu}_k$  centroid  $k$ -te grupe, a  $b_k^{(i)}$  indikatorska varijabla pripadnosti  $\mathbf{x}^{(i)}$  grupi  $k$

- Svaki primjer  $\mathbf{x}^{(i)}$  svrstavamo u grupu s njemu najbližim centroidom  $\boldsymbol{\mu}_k$ :

$$b_k^{(i)} = \begin{cases} 1 & \text{ako } k = \operatorname{argmin}_j \|\mathbf{x}^{(i)} - \boldsymbol{\mu}_j\| \\ 0 & \text{inače} \end{cases}$$

- Tražimo grupiranje  $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K$  koje minimizira pogrešku:  $\operatorname{argmin}_{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K} J$
- Analitička minimizacija nije moguća jer su  $b_k^{(i)}$  i  $\boldsymbol{\mu}_k$  međuvisni
- Alternativa: **iterativna optimizacija**

- Fiksiramo  $\boldsymbol{\mu}_k$  na neke inicijalne vrijednosti
- Pridružimo primjere grupama (izračunamo  $b_k^{(i)}$  za  $i = 1, \dots, N$ )
- Uz fiksne  $b_k^{(i)}$ , minimizacija  $J$  daje formulu za ažuriranje centroida:

$$\nabla_{\boldsymbol{\mu}_k} J = \mathbf{0} \quad \Rightarrow \quad 2 \sum_{i=1}^N b_k^{(i)} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k) = \mathbf{0} \quad \Rightarrow \quad \boldsymbol{\mu}_k = \frac{\sum_i b_k^{(i)} \mathbf{x}^{(i)}}{\sum_i b_k^{(i)}}$$

- Ponavljamo do konvergencije  $\boldsymbol{\mu}_k$  odnosno  $b_k^{(i)}$

### Algoritam K-sredina (k-means algorithm)

```

1: inicijaliziraj centroide $\boldsymbol{\mu}_k$, $k = 1, \dots, K$
2: ponavljam
3: za svaki $\mathbf{x}^{(i)} \in \mathcal{D}$
4: $b_k^{(i)} \leftarrow \begin{cases} 1 & \text{ako } k = \operatorname{argmin}_j \|\mathbf{x}^{(i)} - \boldsymbol{\mu}_j\| \\ 0 & \text{inače} \end{cases}$
5: za svaki $\boldsymbol{\mu}_k$, $k = 1, \dots, K$
6: $\boldsymbol{\mu}_k \leftarrow \sum_{i=1}^N b_k^{(i)} \mathbf{x}^{(i)} / \sum_{i=1}^N b_k^{(i)}$
7: dok $\boldsymbol{\mu}_k$ ne konvergiraju

```

- Vremenska složenost za  $T$  iteracija:  $T(\mathcal{O}(nNK) + \mathcal{O}(nN)) = \mathcal{O}(TnNK)$

- **Konvergencija algoritma:**

- Broj konfiguracija (particija) je konačan i iznosi  $K^N$
- $J$  monotono pada kroz iteracije

$\Rightarrow$  algoritam svaku konfiguraciju posjećuje najviše jednom  $\Rightarrow$  **algoritam konvergira**

- **Optimalnost algoritma:**

- Algoritam **pohlepno pretražuje** konfiguracije te nalazi **lokalni optimum** od  $J$
- Optimalnost rješenja ovisi o odabiru početnih središta

- Pristupi za odabir početnih središta:

- Nasumičan odabir primjera kao centroida
- $K$  slučajnih vektora prirodnih centroida cijelog skupa podataka
- $K$  centroida iz  $K$  segmenata primjera projiciranih na prvu PCA komponentu
- **k-means++**: vjerojatnost odabira primjera  $\mathbf{x}^{(i)}$  kao novog središta  $\boldsymbol{\mu}_{k+1}$ :

$$P(\boldsymbol{\mu}_{k+1} = \mathbf{x}^{(i)} | \mathcal{D}, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k) = \frac{\min_k \|\boldsymbol{\mu}_k - \mathbf{x}^{(i)}\|^2}{\sum_j \min_k \|\boldsymbol{\mu}_k - \mathbf{x}^{(j)}\|^2}$$

$\Rightarrow$  vjerojatnost je proporcionalna kvadratu udaljenosti od već odabranih središta

- Grupiranje treba pokrenuti više puta i uzeti rezultat s najmanjim  $J$

## 4 Algoritam K-medoida

- Algoritma K-sredina: (1) primjeri moraju biti vektori, (2) udaljenost je euklidska
- **Algoritam K-medoida**: poopćenje  $K$ -sredina za općenitu mjeru sličnosti/različitosti
- Prototipi grupe nisu centroidi nego **medoidi** (odabrani primjeri u svakoj grupi)
- Funkcija pogreške:

$$J = \sum_{i=1}^N \sum_{k=1}^K b_k^{(i)} \nu(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k)$$

gdje je  $\nu : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  općenita **mjera različitosti** dvaju primjera

- Tipična izvedba je **algoritam PAM** (*partitioning around medoids*)

### Algoritam PAM

- ```

1:   inicijaliziraj medoide  $\mathcal{M} = \{\boldsymbol{\mu}_k\}_{k=1}^K$  na odabrane  $\mathbf{x}^{(i)}$ 
2:   ponavljaj
3:     za svaki  $\mathbf{x}^{(i)} \in \mathcal{D} \setminus \mathcal{M}$ 
4:        $b_k^{(i)} \leftarrow \begin{cases} 1 & \text{ako } k = \operatorname{argmin}_j \nu(\mathbf{x}^{(i)}, \boldsymbol{\mu}_j) \\ 0 & \text{inače} \end{cases}$ 
5:     za svaki  $\boldsymbol{\mu}_k \in \mathcal{M}$ 
6:        $\boldsymbol{\mu}_k \leftarrow \operatorname{argmin}_{\boldsymbol{\mu}_j \in \mathcal{D} \setminus \mathcal{M} \cup \{\boldsymbol{\mu}_k\}} \sum_i b_k^{(i)} \nu(\mathbf{x}^{(i)}, \boldsymbol{\mu}_j)$ 
7:   dok  $\boldsymbol{\mu}_k$  ne konvergiraju

```

- Složenost za T iteracija: $T(\mathcal{O}(K(N-K)) + \mathcal{O}(K(N-K)^2)) = \mathcal{O}(TK(N-K)^2)$
- Nedostatak algoritma PAM: visoka vremenska složenost

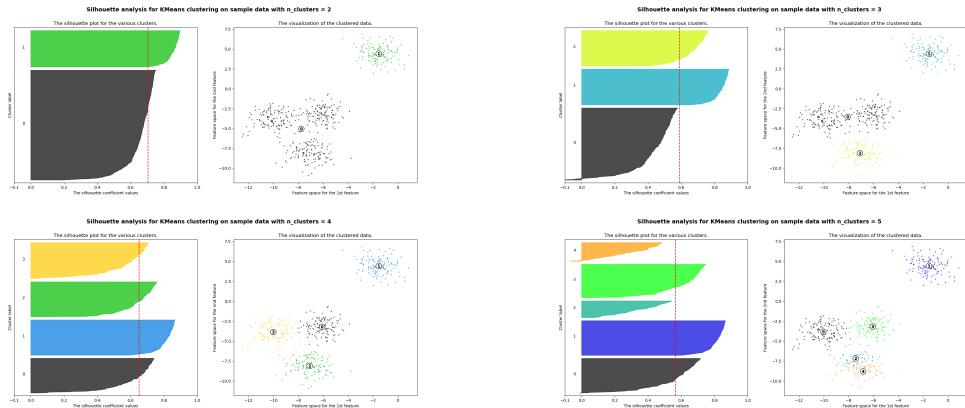
5 Provjera grupa

- **Broj grupa** K koji mnogih je algoritama grupiranja potrebno odrediti unaprijed
- Odabir optimalnog broja grupa dio je **provjere grupiranja** (*cluster validation*)
- J ostvaruje minimum za $K = N \Rightarrow$ nije indikativno za optimalan broj grupa
- Jednostavnije metode za odabir broja grupa:
 - Ručna provjera kvalitete grupa
 - Redukcija dimenzija u 2D-prostor (PCA, MDS, CA, t-SNE) i vizualna provjera
 - Metoda “koljena” (*elbow method*) – nalaženje platoa funkcije $J(K)$
- **Analiza siluete** (*silhouette analysis*):
 - Silueta primjera $\mathbf{x}^{(i)}$:

- Silueta primjera $\mathbf{x}^{(i)}$:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \in [-1, +1]$$

- $a(i)$ i $b(i)$ su prosjek udaljenost od $\mathbf{x}^{(i)}$ do primjera iste odnosno najbliže grupe
- Računamo i grafički prikazujemo $s(i)$ za sve primjere svake grupe
- Loše grupiranje: ispodprosječne siluete nekih grupa i/ili visoka varijanca silueta
- Primjer (scikit-learn):



- **Minimizacija regularizirane funkcije pogreške:**

- Kažnjavanje modela s velikim brojem grupa:

$$K^* = \operatorname{argmin}_K (J(K) + \lambda K)$$

- **Akaikeov kriterij (AIC)** za algoritam K -sredina: $\lambda = 2n$

- **Točnost na podskupu primjera:**

- Raspolažemo označenim podskupom primjera ili parova primjera

- **Randov indeks** – točnost grupiranja na razini parova primjera:

$$R = \frac{a + b}{\binom{N}{2}} \in [0, 1]$$

- a – broj jednakoznačenih parova u istim grupama
- b – broj različito označenih parova u različitim grupama
- Optimalan K je onaj koji maksimizira $R(K)$

20. Grupiranje II

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

Jan Šnajder, natuknice s predavanja, v1.1

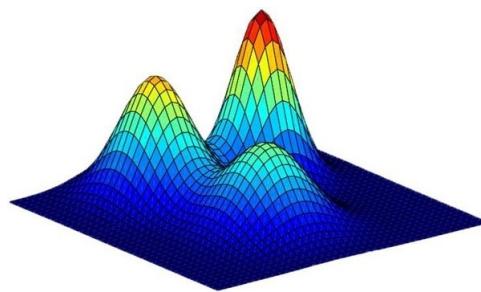
1 Model Gaussove mješavine

- **Model Gaussove mješavine (GMM)** \Rightarrow probabilističko meko partijsko grupiranje
- Poopćenje algoritma K-sredina: umjesto $b_k^{(i)} \in \{0, 1\}$ imamo $h_k^{(i)} \in [0, 1]$
- $h_k^{(i)}$ je **odgovornost** – vjerojatnost da je primjer $\mathbf{x}^{(i)}$ generirala grupa k
- GMM je poseban slučaj **modela miješane gustoće** (*mixture model*)
- Model miješane gustoće je linearna kombinacija K **komponenti**:

$$p(\mathbf{x}) = \sum_{k=1}^K p(\mathbf{x}, y=k) = \sum_{k=1}^K P(y=k)p(\mathbf{x}|y=k) = \sum_{k=1}^K \pi_k p(\mathbf{x}|\boldsymbol{\theta}_k)$$

gdje su π_k **koeficijenti mješavine**, a $p(\mathbf{x}|\boldsymbol{\theta}_k)$ **gustoće komponenti**

- Primjer: model bivarijatne Gaussove mješavine s $K = 3$ grupe:



- Odgovornost možemo izračunati Bayesovim pravilom:

$$h_k^{(i)} = P(y=k|\mathbf{x}^{(i)}) = \frac{P(y=k)p(\mathbf{x}^{(i)}|y=k)}{\sum_j P(y=j)p(\mathbf{x}^{(i)}|y=j)} = \frac{\pi_k p(\mathbf{x}^{(i)}|\boldsymbol{\theta}_k)}{\sum_j \pi_j p(\mathbf{x}^{(i)}|\boldsymbol{\theta}_j)}$$

- Parametri modela su $\boldsymbol{\theta} = \{\pi_k, \boldsymbol{\theta}_k\}_{k=1}^K$, $\boldsymbol{\theta}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Parametre možemo (pokušati) procijeniti metodom MLE

- Log-izglednost parametara modela (tzv. **nepotpuna izglednost**):

$$\ln \mathcal{L}(\boldsymbol{\theta} | \mathcal{D}) = \ln \prod_{i=1}^N p(\mathbf{x}^{(i)}) = \ln \prod_{i=1}^N \sum_{k=1}^K \pi_k p(\mathbf{x}^{(i)} | \boldsymbol{\theta}_k) = \sum_{i=1}^N \ln \sum_{k=1}^K \pi_k p(\mathbf{x}^{(i)} | \boldsymbol{\theta}_k)$$

\Rightarrow ne faktorizira se po komponentama \Rightarrow maksimizacija nema analitičko rješenje

2 Algoritam maksimizacije očekivanja

- Proširenje modela miješane gustoće **latentnim varijablama** (varijable koje ne opažamo)
- Latentna kategorička varijabla $\mathbf{z}^{(i)}$ definira koja je grupa generirala primjer $\mathbf{x}^{(i)}$:

$$\mathbf{z}^{(i)} = (z_1^{(i)}, \dots, z_k^{(i)}, \dots, z_K^{(i)})$$

- Distribucija kategoričke varijable $\mathbf{z}^{(i)}$:

$$P(\mathbf{z}^{(i)} = k) = \prod_{k=1}^K \pi_k^{z_k^{(i)}}$$

- Zajednička gustoća varijabli \mathbf{x} i \mathbf{z} :

$$p(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) = P(\mathbf{z})p(\mathbf{x} | \mathbf{z}, \boldsymbol{\theta}) = \prod_{k=1}^K \pi_k^{z_k} \prod_{k=1}^K p(\mathbf{x} | \boldsymbol{\theta}_k)^{z_k} = \prod_{k=1}^K \pi_k^{z_k} p(\mathbf{x} | \boldsymbol{\theta}_k)^{z_k}$$

\Rightarrow model s **latentnim varijablama** \mathbf{z}

- Log-izglednost parametara modela (tzv. **potpuna izglednost**):

$$\begin{aligned} \ln \mathcal{L}(\boldsymbol{\theta} | \mathcal{D}, \mathbf{Z}) &= \ln \prod_{i=1}^N p(\mathbf{x}^{(i)}, \mathbf{z}^{(i)} | \boldsymbol{\theta}) = \ln \prod_{i=1}^N \prod_{k=1}^K \pi_k^{z_k^{(i)}} p(\mathbf{x} | \boldsymbol{\theta}_k)^{z_k^{(i)}} \\ &= \sum_{i=1}^N \sum_{k=1}^K z_k^{(i)} (\ln \pi_k + \ln p(\mathbf{x}^{(i)} | \boldsymbol{\theta}_k)) \end{aligned}$$

\Rightarrow ako su $\mathbf{z}^{(i)}$ poznate, maksimizacija ove log-izglednosti ima analitičko rješenje

- $\mathbf{z}^{(i)}$ su nepoznate, no možemo izračunati **očekivanje** izglednosti uz fiksirane π_k i $\boldsymbol{\theta}_k$
- Može se pokazati: povećanje očekivanja od $\mathcal{L}(\boldsymbol{\theta} | \mathcal{D}, \mathbf{Z}) \Rightarrow$ povećanje $\mathcal{L}(\boldsymbol{\theta}; \mathcal{D})$
- **Algoritam maksimizacije očekivanja (EM-algoritam)**: iterativna optimizacija $\mathcal{L}(\boldsymbol{\theta}; \mathcal{D})$
- Dva koraka algoritma: E-korak (*expectation*) i M-korak (*maximization*)
- **E-korak**: Izračun očekivanja potpune izglednosti uz fiksirane parametre u iteraciji t :

$$\begin{aligned} \mathcal{Q}(\boldsymbol{\theta} | \boldsymbol{\theta}^{(t)}) &= \mathbb{E}_{\mathbf{Z} | \mathcal{D}, \boldsymbol{\theta}^{(t)}} \left[\sum_{i=1}^N \sum_{k=1}^K z_k^{(i)} (\ln \pi_k + \ln p(\mathbf{x}^{(i)} | \boldsymbol{\theta}_k)) \right] \\ &= \sum_{i=1}^N \sum_{k=1}^K \underbrace{\mathbb{E}[z_k^{(i)} | \mathcal{D}, \boldsymbol{\theta}^{(t)}]}_{= h_k^{(i)}} (\ln \pi_k + \ln p(\mathbf{x}^{(i)} | \boldsymbol{\theta}_k)) \end{aligned}$$

- **M-korak:** Izračun parametara za iteraciju $(t + 1)$ koji maksimiziraju očekivanje:

$$\nabla_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta} | \boldsymbol{\theta}^{(t)}) = 0$$

$$\nabla_{\pi_k} \left(\sum_{i=1}^N \sum_{k=1}^K h_k^{(i)} \ln \pi_k + \lambda \left(\sum_k \pi_k - 1 \right) \right) = 0 \quad \Rightarrow \quad \pi_k^{(t+1)} = \frac{1}{N} \sum_{i=1}^N h_k^{(i)}$$

$$\nabla_{\boldsymbol{\theta}_k} \sum_{i=1}^N h_k^{(i)} \ln p(\mathbf{x}^{(i)} | \boldsymbol{\theta}_k) = 0 \quad \Rightarrow \quad \boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_i h_k^{(i)} \mathbf{x}^{(i)}}{\sum_i h_k^{(i)}}$$

$$\Rightarrow \quad \boldsymbol{\Sigma}_k^{(t+1)} = \frac{\sum_i h_k^{(i)} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k^{(t+1)}) (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k^{(t+1)})^\top}{\sum_i h_k^{(i)}}$$

Algoritam GMM (model GMM + EM-algoritam)

inicijaliziraj parametre $\boldsymbol{\theta} = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$
ponavlja do konvergencije log-izglednosti ili parametara

E-korak:

Za svaki primjer $\mathbf{x}^{(i)} \in \mathcal{D}$ i svaku komponentu $k = 1, \dots, K$:

$$h_k^{(i)} \leftarrow \frac{p(\mathbf{x}^{(i)} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \pi_k}{\sum_{j=1}^K p(\mathbf{x}^{(i)} | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \pi_j}$$

M-korak:

Za svaku komponentu $k = 1, \dots, K$:

$$\boldsymbol{\mu}_k \leftarrow \frac{\sum_i h_k^{(i)} \mathbf{x}^{(i)}}{\sum_i h_k^{(i)}}, \quad \boldsymbol{\Sigma}_k \leftarrow \frac{\sum_i h_k^{(i)} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k) (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k)^\top}{\sum_i h_k^{(i)}}, \quad \pi_k \leftarrow \frac{1}{N} \sum_{i=1}^N h_k^{(i)}$$

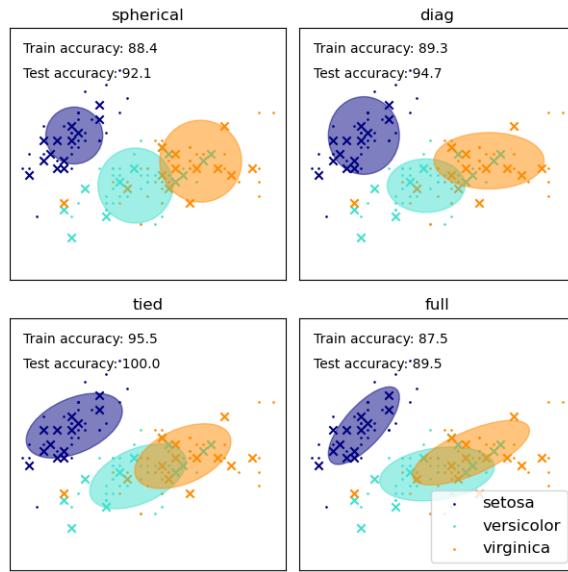
Izračunaj trenutnu vrijednost log-izglednosti

$$\ln \mathcal{L}(\boldsymbol{\theta} | \mathcal{D}) = \sum_{i=1}^N \ln \sum_{k=1}^K \pi_k p(\mathbf{x}^{(i)} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- EM-algoritam konvergira, ali ne nužno u globalni optimum log-izglednosti
- Akaikeov informacijski kriterij (AIC) za odabir optimalnog broja grupa:

$$K^* = \operatorname{argmin}_K (-2 \ln \mathcal{L}(K) + 2q(K))$$

- Moguća pojednostavljenja: dijeljena matrica, dijagonalna ili izotropna matrica Σ



3 Hijerarhijsko grupiranje

- Hijerarhijsko grupiranje producira **dendrogram** – stablasti prikaz hijerarhije grupa
- Provodi se na temelju mjere udaljenosti ili mjere sličnosti/različitosti
- Može biti **aglomerativno** (bottom-up) ili **divizivno** (top-down)
- **Hijerarhijsko aglomerativno grupiranje (HAC)**: iterativno stapa najbliže parove grupa
- **Povezivanje** (*linkage*) – način izračuna udaljenosti između dvije grupe:

- **Jednostruko povezivanje** (*single linkage*)

$$d_{min}(\mathcal{G}_i, \mathcal{G}_j) = \min_{\mathbf{x} \in \mathcal{G}_i, \mathbf{x}' \in \mathcal{G}_j} d(\mathbf{x}, \mathbf{x}')$$

- **Potpuno povezivanje** (*complete linkage*)

$$d_{max}(\mathcal{G}_i, \mathcal{G}_j) = \max_{\mathbf{x} \in \mathcal{G}_i, \mathbf{x}' \in \mathcal{G}_j} d(\mathbf{x}, \mathbf{x}')$$

- **Prosječno povezivanje** (*average linkage*)

$$d_{avg}(\mathcal{G}_i, \mathcal{G}_j) = \frac{1}{N_i N_j} \sum_{\mathbf{x} \in \mathcal{G}_i} \sum_{\mathbf{x}' \in \mathcal{G}_j} d(\mathbf{x}, \mathbf{x}')$$

- **Povezivanje centroida** (*centroid linkage*)

$$d_{cent}(\mathcal{G}_i, \mathcal{G}_j) = \left\| \frac{1}{N_i} \sum_{\mathbf{x} \in \mathcal{G}_i} \mathbf{x} - \frac{1}{N_j} \sum_{\mathbf{x} \in \mathcal{G}_j} \mathbf{x} \right\|$$

Algoritam hijerarhijskog aglomerativnog grupiranja (HAC)

```

1: inicijaliziraj  $K$ ,  $k \leftarrow N$ ,  $\mathcal{G}_i \leftarrow \{\mathbf{x}^{(i)}\}$  za  $i = 1, \dots, N$ 
2: ponavljaj
3:    $k \leftarrow k - 1$ 
4:    $(\mathcal{G}_i, \mathcal{G}_j) \leftarrow \underset{\mathcal{G}_a, \mathcal{G}_b}{\operatorname{argmin}} d(\mathcal{G}_a, \mathcal{G}_b)$ 
5:    $\mathcal{G}_i \leftarrow \mathcal{G}_i \cup \mathcal{G}_j$ 
6: dok je  $k > K$ 

```

- Prostorna složenost: matrica udaljenosti za $\binom{N}{2}$ parova primjera $\Rightarrow \mathcal{O}(N^2)$
- Vremenska složenost: općenito $\mathcal{O}(N^3)$, $\mathcal{O}(N^2 \log N)$ s prioritetsnom listom

21. Vrednovanje modela

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

Jan Šnajder, natuknice s predavanja, v1.3

1 Osnovne mjere vrednovanja

- **Matrica zabune** (*confusion matrix*) – usporeba stvarnih oznaka i predikcija modela

		Stvarno	
		1	0
Model	1	TP	FP
	0	FN	TN

TP – true positives, FP – false positives, FN – false negatives, TN – true negatives

- **Točnost** (*accuracy*) je udio točno klasificiranih primjera u skupu svih primjera:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} = 1 - E(h|\mathcal{D})$$

- Ako je udio klase izrazito neuravnotežen, točnost nije indikativna mjeru

- **Preciznost** (*precision*):

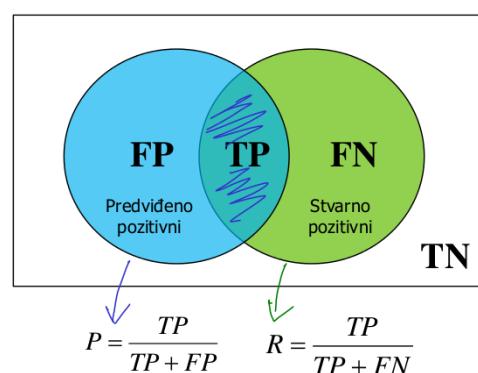
$$P = \frac{TP}{TP + FP}$$

⇒ udio pozitivno klasificiranih primjera u skupu pozitivno klasificiranih primjera

- **Odziv** (*recall, true positive rate, sensitivity*):

$$R = TPR = \frac{TP}{TP + FN}$$

⇒ udio pozitivno klasificiranih primjera u skupu svih pozitivnih primjera



- **Fall-out** (false positive rate)

$$FPR = \frac{FP}{FP + TN}$$

\Rightarrow udio primjera pogrešno proglašenih pozitivnima

- **Specifičnost** (*specificity*):

$$S = \frac{TN}{TN + FP}$$

\Rightarrow udio negativno klasificiranih primjera u skupu svih negativnih primjera

- **Mjera F1** – harmonijska sredina preciznosti i odziva:

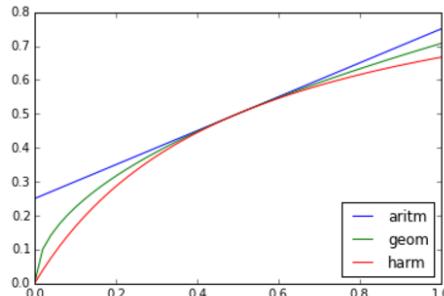
$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} = \frac{2PR}{P + R}$$

- **Mjera F-beta** – poopćenje mjere F1 koje različito naglašava P i R :

$$F_\beta = \frac{(1 + \beta^2)PR}{\beta^2 P + R}$$

$\Rightarrow F_{0.5}$ dvostruko naglašava preciznost, F_2 dvostruko naglašava odziv

- Harmonijska sredina je “najstroža” od triju sredina; npr. za $R = 0.5$ i $P \in [0, 1]$:



- Primjer: $N = 1000$, od čega 100 poz. Ispravno klasificiranih 90 poz. i 650 neg.

		Stvarno	
		1	0
Model	1	90	250
	0	10	650

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} = 0.74$$

$$P = \frac{TP}{TP + FP} = \frac{90}{90 + 250} = 0.265$$

$$R = \frac{TP}{FP + FN} = \frac{90}{90 + 10} = 0.9$$

$$F_1 = \frac{2PR}{P + R} = \frac{2 \cdot 0.265 \cdot 0.9}{0.265 + 0.9} = 0.409$$

2 Višeklasna klasifikacija

- Iz matrice $K \times K$ ($K > 2$) izvodimo matricu 2×2 za svaku klasu j , s elementima:
 - TP_j – j -ti element dijagonale
 - FP_j – zbroj nedijagonalnih elemenata j -tog retka
 - FN_j – zbroj nedijagonalnih elemenata j -tog stupca
 - $\text{TN}_j = N - \text{TP}_j - \text{FP}_j - \text{FN}_j$ – zbroj po elementima izvan retka j i stupca j
- **Makro-prosjek** (M): izračun mjere za svaku klasu pa uprosječivanje kroz klase

$$\text{Acc}^M = \frac{1}{K} \sum_{j=1}^K \text{Acc}_j, \quad P^M = \frac{1}{K} \sum_{j=1}^K P_j, \quad R^M = \frac{1}{K} \sum_{j=1}^K R_j, \quad F_1^M = \frac{1}{K} \sum_{j=1}^K F_{1,j}$$

\Rightarrow jednak utjecaj svih klasa \Rightarrow loš rezultat na manjim klasama narušava mjeru

- **Mikro-prosjek** (μ): zbrajanje matrica pojedinačnih klasa pa izračun mjere

$$\text{TP} = \sum_{j=1}^K \text{TP}_j, \quad \text{FP} = \sum_{j=1}^K \text{FP}_j, \quad \text{FN} = \sum_{j=1}^K \text{FN}_j, \quad \text{TN} = \sum_{j=1}^K \text{TN}_j$$

\Rightarrow vrijedi $\text{FP} = \text{FN} \Rightarrow$ vrijedi $P^\mu = R^\mu = F_1^\mu$

- Vrijedi $\text{Acc}^M = \text{Acc}^\mu$
- Alternativa: neuprosječena točnost – $\text{Acc} = \frac{1}{N} \sum_{j=1}^K \text{TP}_j = P^\mu = R^\mu = F_1^\mu$
- Primjer ($N = 13$, $K = 3$):

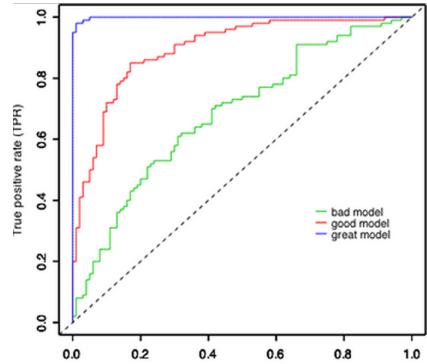
$$\begin{array}{c} \begin{array}{ccc} y = 1 & y = 2 & y = 3 \end{array} \\ \begin{array}{c} y = 1 \\ y = 2 \\ y = 3 \end{array} \left(\begin{array}{ccc} 1 & 1 & 0 \\ 2 & 2 & 3 \\ 0 & 0 & 4 \end{array} \right) \Rightarrow \begin{array}{c} \overbrace{\begin{array}{ccc} y = 1 & y = 2 & y = 3 \end{array}}^{\text{Makro}} \\ \begin{array}{ccc} (1 & 1) & (2 & 5) & (4 & 0) \\ (2 & 9) & (1 & 5) & (3 & 6) \end{array} \end{array} \Rightarrow \begin{array}{c} \overbrace{\begin{array}{cc} 7 & 6 \end{array}}^{\text{zbroj}} \\ \begin{array}{cc} 6 & 20 \end{array} \end{array} \end{array}$$

$$\begin{aligned} \text{Acc}^M &= \frac{1}{3} \left(\frac{10}{13} + \frac{7}{13} + \frac{10}{13} \right) = 0.69 & \text{Acc}^\mu &= \frac{27}{39} = 0.69 \\ P^M &= \frac{1}{3} \left(\frac{1}{2} + \frac{2}{7} + \frac{4}{4} \right) = 0.60 & P^\mu &= \frac{7}{13} = 0.54 \\ R^M &= \frac{1}{3} \left(\frac{1}{3} + \frac{2}{3} + \frac{4}{7} \right) = 0.52 & R^\mu &= \frac{7}{13} = 0.54 \\ F_1^M &= \frac{1}{3} (0.40 + 0.40 + 0.73) = 0.51 & F_1^\mu &= \frac{2P^M R^M}{P^M + R^M} = 0.54 \end{aligned}$$

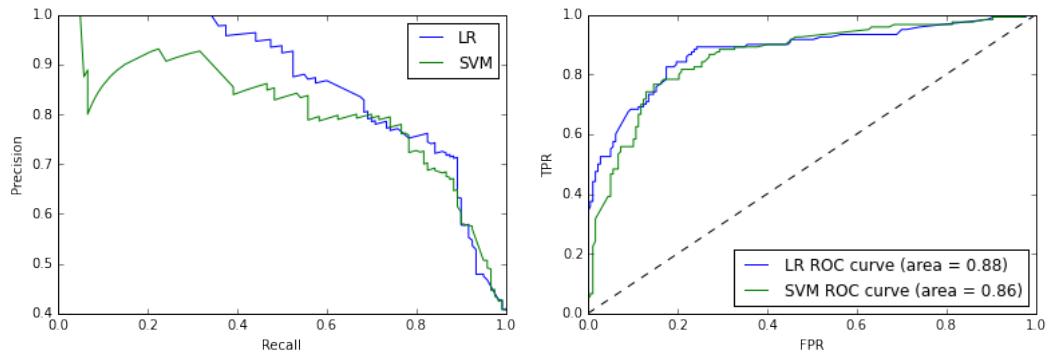
- Tipično (ali ne nužno) $M < \mu$ jer klasifikatori rade lošije na manjim klasama

3 Vrednovanje klasifikatora s pragom

- Ugađanjem klasifikacijskoj praga može se ugađati P i R modela
- **Krivulja preciznost-odziv (P-R)** – preciznost kao funkcija odziva (monotonu opada)
- Agregatna mjera: **prosječna preciznost (AP)** (*average precision*)
- **Krivulja ROC** – odziv kao funkcija od FPR (fall-out)

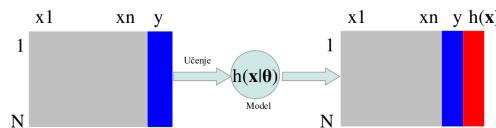


- Nasumična predikcija $\Rightarrow TPR = FPR$, neovisno o udjelu pozitivnih primjera
- Agregatna mjera: **površina ispod ROC krivulje (AUC)** (*area under curve*)
- Najbolji model: (1, 1) za krivulju P-R, (0, 1) za krivulju ROC



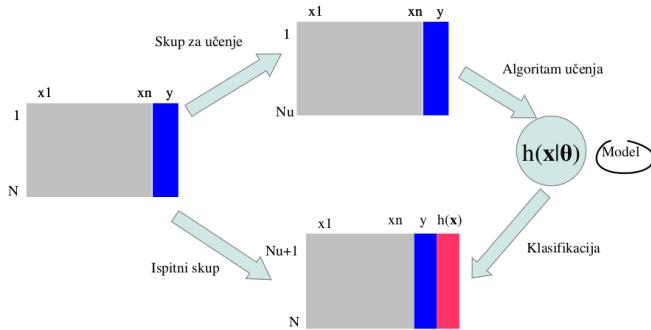
4 Procjena pogreške modela

- Ispitni skup je **slučajan uzorak** \Rightarrow svaka mjera točnosti je funkcija **slučajne varijable**
- Procjena pogreške (točnosti) treba biti **dobra** (nepristrana) i **poštena** (realistična)
- Procjena na skupu za učenje \Rightarrow ne mjerimo pogrešku generalizacije \Rightarrow nepoštено



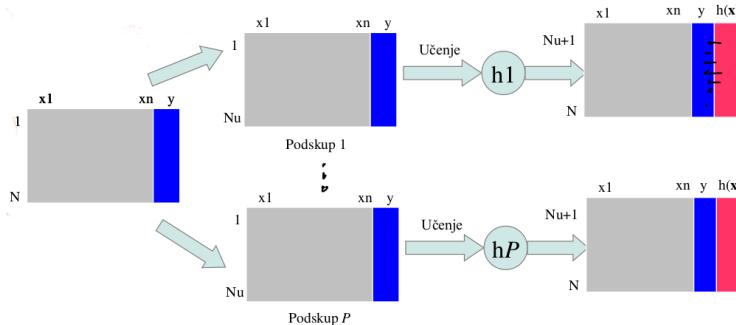
- **Metoda izdvajanja (holdout method)**

- Podjela na skup za učenje i skup za ispitivanje (npr., 70%–30%)
- Prednost: mjerimo pogrešku generalizacije
- Nedostatci: gubitak primjera za učenje, procjena na samo jednom uzorku



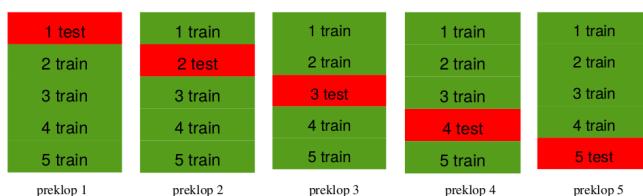
- **Ponovljeno izdvajanje (repeated holdout)**

- Višestruko uzorkovanje skupova za učenje/ispitivanje pa izračun prosjeka mjere
- Prednost: procjena pogreške generalizacije na više uzorka
- Nedostatak: ne kontroliramo koji su primjeri i koliko puta upotrijebljeni



- **k -struka unakrsna provjera (CV) (k -folded cross-validation)**

- Podjela na k **preklopa (folds)** (tipično $k = 5$ ili $k = 10$)
- Učenje na $(k - 1)$ preklopa, ispitivanje na jednom preklopu, ponovljeno k puta
- Prednost: svaki je primjer iskorišten i za učenje i za ispitivanje
- Nedostatak: modeli nisu međusobno nezavisni \Rightarrow visoka varijanca procjene



- **Metoda izdvoji jednoga (LOOCV)** (*leave-one-out cross-validation*)
 - k -struka unakrsna provjera uz $k = N$
 - Prednost: gotovo svi primjeri se koriste za učenje u svakoj iteraciji
 - Nedostatci: računalno skupo, visoka varijanca procjene pogreške
- Procjena pogreške uz **odabir modela**:
 - Podjela na skup za **učenje** ($\mathcal{D}_{\text{train}}$), **provjeru** ($\mathcal{D}_{\text{validate}}$) i **ispitivanje** ($\mathcal{D}_{\text{test}}$)
 - Odabir modela: učenje na $\mathcal{D}_{\text{train}}$ i ispitivanje na $\mathcal{D}_{\text{validate}}$
 - Ispitivanje odabranog modela: učenje na $\mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{validate}}$ i ispitivanje na $\mathcal{D}_{\text{test}}$
- k -struka CV uz odabir modela \Rightarrow **ugniježđena unakrsna provjera** (*nested CV*)

Ugniježđena unakrsna provjera $k \times l$

```

1: podijeli  $\mathcal{D}$  na vanjske preklope  $\mathcal{D}_i$ ,  $i = 1, \dots, k$ 
2: za  $i = 1, \dots, k$  radi: vanjska petlja
3:    $\mathcal{D}_{\text{train}} \leftarrow \mathcal{D} \setminus \mathcal{D}_i$ ,  $\mathcal{D}_{\text{test}} \leftarrow \mathcal{D}_i$ 
4:   za svaku odabranu vrijednost hiperparametra  $\alpha$  radi:
5:     podijeli  $\mathcal{D}_{\text{train}}$  na unutarnje preklope  $\mathcal{D}_j$ ,  $j = 1, \dots, l$  unutarnja petlja
6:     za  $j = 1, \dots, l$  radi:
7:        $\mathcal{D}_{\text{train}'} \leftarrow \mathcal{D}_{\text{train}} \setminus \mathcal{D}_j$ ,  $\mathcal{D}_{\text{validate}} \leftarrow \mathcal{D}_j$ 
8:       nauči model na  $\mathcal{D}_{\text{train}'}$  i ispitaj na  $\mathcal{D}_{\text{validate}}$ 
9:       izračunaj prosjek mjere na  $l$  unutarnjih preklopa
10:      odaberi hiperparametar  $\alpha$  koji maksimizira prosjek mjere
11:      nauči odabrani model na  $\mathcal{D}_{\text{train}}$  i ispitaj na  $\mathcal{D}_{\text{test}}$ 
12:      izračunaj prosjek mjere na  $k$  vanjskih preklopa

```

- Odabir hiperparametara (redak 4) može biti vođen heurističkim pretraživanjem
- Kao optimalan model odabradi onaj koji je najčešće odabran u k vanjskih preklopa
- Paziti da se pri učenju modela koristi isključivo informacija iz skupa za učenje