Kernel alapú félig-felügyelt tanuló algoritmusok

Bodó Zalán

zbodo@cs.ubbcluj.ro
BBTE, Matematika és Informatika Kar

Az előadás tartalma

Kernel alapú tanulás
- Kernelek
- SVM

Néhány (általános célú) kernel

Félig-felügyelt tanulás

Kernel alapú félig-felügyelt tanulás
- LapSVM
- Szomszédsági kernel
- ISOMAP
- Hierarchikus klaszter kernel
- Átsúlyozó klaszter kernelek

Kernel alapú tanulás

Gépi tanulás

- **felügyelt tanulás**
- felügyelet nélküli tanulás
- megerősítéses tanulás

Felügyelt tanulás

- diszkrét eset: **osztályozás**
- $\widehat{f} : X \to Y$ keresése, amely legjobban közelíti $f : X \to Y$-t; f tanulási példák által van megadva: $(\boldsymbol{x}_i, y_i)$, $\boldsymbol{x}_i \in X \subseteq \mathbb{R}^d$, $y_i \in Y$
  - $|Y| = 2$ (általában $Y = \{-1, 1\}$) – bináris osztályozás
  - $|Y| > 2$, $Y = \{c_1, c_2, \ldots, c_K\}$, K az osztályok száma – többosztályos tanulás
    - $f : X \to Y$ – egycímkés tanulás
    - $f : X \to 2^Y$ – többcímkés tanulás

Kernel alapú tanulás

Gépi tanulás

- **felügyelt tanulás**
- felügyelet nélküli tanulás
- megerősítéses tanulás

Felügyelt tanulás

- diszkrét eset: **osztályozás**
- $\widehat{f} : X \to Y$ keresése, amely legjobban közelíti $f : X \to Y$-t; f tanulási példák által van megadva: $(\boldsymbol{x}_i, y_i)$, $\boldsymbol{x}_i \in X \subseteq \mathbb{R}^d$, $y_i \in Y$
  - $|Y| = 2$ (általában $Y = \{-1, 1\}$) – bináris osztályozás
  - $|Y| > 2$, $Y = \{c_1, c_2, \ldots, c_K\}$, K az osztályok száma – többosztályos tanulás
    - $f : X \to Y$ – egycímkés tanulás
    - $f : X \to 2^Y$ – többcímkés tanulás

Kernelek

- gépi tanulás $\rightarrow$ adatok közötti hasonlóság
- **kernel** $=$ hasonlósági metrika
- kernel: $k(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{y}) \rangle = \phi(\boldsymbol{x})' \phi(\boldsymbol{z})$, ahol $\phi : X \rightarrow \mathcal{H}$-t feature-leképezésnek nevezzük
- James Mercer (1909): "Minden folytonos szimmetrikus pozitív szemidefinit kernel függvény kifejezhető skalárszorzatként egy nagydimenziós térben."
- **kernel-trükk**: adott algoritmus esetén $k(\cdot, \cdot) \rightarrow \widetilde{k}(\cdot, \cdot)$

Egyszerű bináris osztályozó: centroid módszer

- keressük meg a két osztály középpontját (centroidját)
- új pont: ahhoz az osztályhoz rendeljük, melynek középpontjához közelebb esik

$$\boldsymbol{c}_+ = \frac{1}{N_+} \sum_{\boldsymbol{x}_i \in X_+} \boldsymbol{x}_i \quad \text{és} \quad \boldsymbol{c}_- = \frac{1}{N_-} \sum_{\boldsymbol{x}_i \in X_-} \boldsymbol{x}_i$$

Kernelek

- gépi tanulás $\rightarrow$ adatok közötti hasonlóság
- **kernel** $=$ hasonlósági metrika
- kernel: $k(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{y}) \rangle = \phi(\boldsymbol{x})' \phi(\boldsymbol{z})$, ahol $\phi : X \rightarrow \mathcal{H}$-t feature-leképezésnek nevezzük
- James Mercer (1909): "Minden folytonos szimmetrikus pozitív szemidefinit kernel függvény kifejezhető skalárszorzatként egy nagydimenziós térben."
- **kernel-trükk**: adott algoritmus esetén $k(\cdot, \cdot) \rightarrow \widetilde{k}(\cdot, \cdot)$

Egyszerű bináris osztályozó: centroid módszer

- keressük meg a két osztály középpontját (centroidját)
- új pont: ahhoz az osztályhoz rendeljük, melynek középpontjához közelebb esik

$$\boldsymbol{c}_+ = \frac{1}{N_+} \sum_{\boldsymbol{x}_i \in X_+} \boldsymbol{x}_i \quad \text{és} \quad \boldsymbol{c}_- = \frac{1}{N_-} \sum_{\boldsymbol{x}_i \in X_-} \boldsymbol{x}_i$$

$$
\begin{aligned}
f(\boldsymbol{x}) &= \frac{1}{2}\operatorname{sign}\left[\|\boldsymbol{x}-\boldsymbol{c}_-\|^2 - \|\boldsymbol{x}-\boldsymbol{c}_+\|^2\right] \\
&= \operatorname{sign}\left[\boldsymbol{x}'\left(\boldsymbol{c}_+ - \boldsymbol{c}_-\right) + \frac{\|\boldsymbol{c}_-\|^2 - \|\boldsymbol{c}_+\|^2}{2}\right]
\end{aligned}
$$

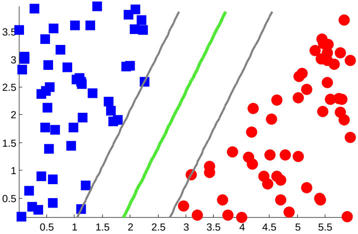
$$
\boldsymbol{w} = \boldsymbol{c}_+ - \boldsymbol{c}_- \quad \text{és} \quad b = \frac{\boldsymbol{c}_-^2 - \boldsymbol{c}_+^2}{2}
$$

$$
\begin{aligned}
f(\boldsymbol{x}) &= \operatorname{sign}\left(\boldsymbol{x}'\boldsymbol{w} + b\right) \\
&= \operatorname{sign}\left(\frac{1}{N_+}\sum_{\boldsymbol{x}_i \in X_+}\langle \boldsymbol{x}, \boldsymbol{x}_i\rangle - \frac{1}{N_-}\sum_{\boldsymbol{x}_i \in X_-}\langle \boldsymbol{x}, \boldsymbol{x}_i\rangle + b\right)
\end{aligned}
$$

SVM

- bináris osztályozó: szeparáló hipersíkot keres **(Boser, Guyon, Vapnik 1992)**
- melyik hipersíkot válasszuk? – amelyik **maximális szegéllyel** választja szét a pozitív és negatív adatokat
- hipersík: $H = \{\boldsymbol{x} \mid \boldsymbol{w}'\boldsymbol{x} + b = 0\}$
- hipersík szegélye: $2/\|\boldsymbol{w}\| \Rightarrow$ minimalizáljuk $\frac{1}{2}\|\boldsymbol{w}\|^2$-et

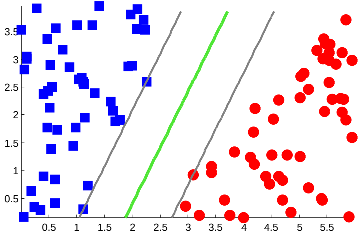
$$\begin{array}{ll} \min & \lambda_A \|\boldsymbol{w}\|^2 + \frac{1}{\ell}\sum_{i=1}^{\ell} \xi_i \\ \text{úgy, hogy} & y_i(\boldsymbol{w}'\boldsymbol{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, \ell \end{array}$$



SVM

- bináris osztályozó: szeparáló hipersíkot keres **(Boser, Guyon, Vapnik 1992)**
- melyik hipersíkot válasszuk? – amelyik **maximális szegéllyel** választja szét a pozitív és negatív adatokat
- hipersík: $H = \{\boldsymbol{x} \mid \boldsymbol{w}'\boldsymbol{x} + b = 0\}$
- hipersík szegélye: $2/\|\boldsymbol{w}\| \Rightarrow$ minimalizáljuk $\frac{1}{2}\|\boldsymbol{w}\|^2$-et

$$\begin{array}{ll} \min & \lambda_A \|\boldsymbol{w}\|^2 + \frac{1}{\ell}\sum_{i=1}^{\ell} \xi_i \\ \text{úgy, hogy} & y_i(\boldsymbol{w}'\boldsymbol{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, \ell \end{array}$$



Nehány (általános célú) kernel

- lineáris:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{x}'\boldsymbol{z}$$

- polinomiális:

$$k(\boldsymbol{x}, \boldsymbol{z}) = (a\boldsymbol{x}'\boldsymbol{z} + b)^c$$

- Gauss típusú:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{z}\|^2}{2\sigma^2}\right)$$

- szigmoid:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \tanh(a\boldsymbol{x}'\boldsymbol{z} + r)$$

Néhány (általános célú) kernel

- lineáris:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{x}'\boldsymbol{z}$$

- polinomiális:

$$k(\boldsymbol{x}, \boldsymbol{z}) = (a\boldsymbol{x}'\boldsymbol{z} + b)^c$$

- Gauss típusú:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{z}\|^2}{2\sigma^2}\right)$$

- szigmoid:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \tanh(a\boldsymbol{x}'\boldsymbol{z} + r)$$

Nehány (általános célú) kernel

- lineáris:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{x}'\boldsymbol{z}$$

- polinomiális:

$$k(\boldsymbol{x}, \boldsymbol{z}) = (a\boldsymbol{x}'\boldsymbol{z} + b)^c$$

- Gauss típusú:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{z}\|^2}{2\sigma^2}\right)$$

- szigmoid:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \tanh(a\boldsymbol{x}'\boldsymbol{z} + r)$$

Nehány (általános célú) kernel

- lineáris:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{x}'\boldsymbol{z}$$

- polinomiális:

$$k(\boldsymbol{x}, \boldsymbol{z}) = (a\boldsymbol{x}'\boldsymbol{z} + b)^c$$

- Gauss típusú:

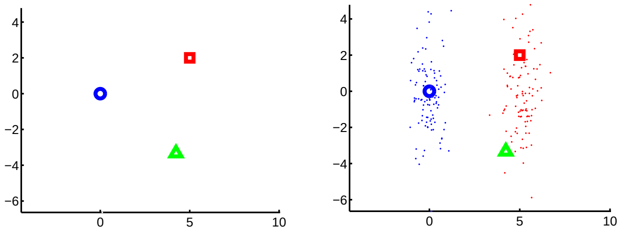
$$k(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{z}\|^2}{2\sigma^2}\right)$$

- szigmoid:

$$k(\boldsymbol{x}, \boldsymbol{z}) = \tanh(a\boldsymbol{x}'\boldsymbol{z} + r)$$

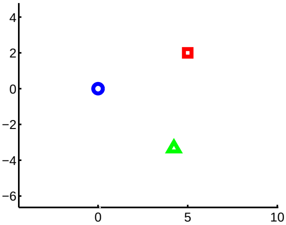
Félig felügyelt tanulás

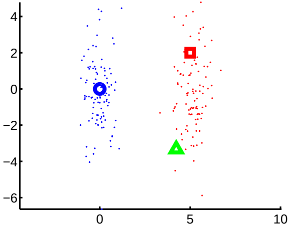
- felügyelt tanulás:
  $D = \{(\boldsymbol{x}_i, y_i) \,|\, \boldsymbol{x}_i \in X \subseteq \mathbb{R}^d, y_i \in \{-1,+1\}, i = 1,\ldots,\ell\}$;
  keressük $\widehat{f} : X \to \{-1,+1\}$-et, amely "egyezik" D-vel
- **félig-felügyelt tanulás**:
  $D = \{(\boldsymbol{x}_i, y_i) \,|\, i = 1,\ldots,\ell\} \cup \{\boldsymbol{x}_j \,|\, j = \ell+1,\ldots,\ell+u\}$,
  $\ell \ll u$, $N = \ell + u$;
  - **induktív**: keressük $\widehat{f} : X \to \{-1,+1\}$-et, amely "egyezik" D-vel + felhasználjuk a D_U által közvetített információt
  - **transzduktív**: keressük $\widehat{f} : D_U \to \{-1,+1\}$-et felhasználva a $D = D_L \cup D_U$ halmazt



Félig felügyelt tanulás

- felügyelt tanulás:
  $D = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in X \subseteq \mathbb{R}^d, y_i \in \{-1, +1\}, i = 1, \ldots, \ell\}$;
  keressük $\widehat{f} : X \to \{-1, +1\}$-et, amely "egyezik" D-vel
- **félig-felügyelt tanulás**:
  $D = \{(\mathbf{x}_i, y_i) \mid i = 1, \ldots, \ell\} \cup \{\mathbf{x}_j \mid j = \ell + 1, \ldots, \ell + u\}$,
  $\ell \ll u$, $N = \ell + u$;
  - **induktív**: keressük $\widehat{f} : X \to \{-1, +1\}$-et, amely "egyezik" D-vel + felhasználjuk a D_U által közvetített információt
  - **transzduktív**: keressük $\widehat{f} : D_U \to \{-1, +1\}$-et felhasználva a $D = D_L \cup D_U$ halmazt





Kernel alapú félig-felügyelt tanulás

- *megváltoztatjuk* a reprezentációs teret, az *összes* adatot figyelembe véve
- felügyelt tanulás + adatfüggő kernelek = **félig-felügyelt tanulás**
- "hagyományos" kernelek: adott $D_1 \neq D_2$ adathamazok esetén, $\mathbf{x}, \mathbf{z} \in D_1 \cap D_2$

$$k(\mathbf{x}, \mathbf{z}) = k(\mathbf{x}, \mathbf{z})$$

- **adatfüggő kernelek**: adott $D_1 \neq D_2$ adathalmazok esetén, $\mathbf{x}, \mathbf{z} \in D_1 \cap D_2$

$$k(\mathbf{x}, \mathbf{z}; D_1) \Bumpeq k(\mathbf{x}, \mathbf{z}; D_2)$$

"$\Bumpeq$" = "nem feltétlenül egyenlő"

Kernel alapú félig-felügyelt tanulás

- *megváltoztatjuk* a reprezentációs teret, az *összes* adatot figyelembe véve
- felügyelt tanulás + adatfüggő kernelek = **félig-felügyelt tanulás**
- "hagyományos" kernelek: adott $D_1 \neq D_2$ adathamazok esetén, $\boldsymbol{x}, \boldsymbol{z} \in D_1 \cap D_2$

$$k(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x}, \boldsymbol{z})$$

- **adatfüggő kernelek**: adott $D_1 \neq D_2$ adathalmazok esetén, $\boldsymbol{x}, \boldsymbol{z} \in D_1 \cap D_2$

$$k(\boldsymbol{x}, \boldsymbol{z}; D_1) \Bumpeq k(\boldsymbol{x}, \boldsymbol{z}; D_2)$$

"$\Bumpeq$" = "nem feltétlenül egyenlő"

LapSVM

- induktív tanulási módszer: SVM + "simasági" feltétel: *ha két pont egy nagy sűrűségű régióban közel található egymáshoz, akkor azok címkéi is hasonlóan kell viselkedjenek*
- új optimalizálási feladat **(Sindhwani, Niyogi, Belkin 2005)**:

$$\min \quad \lambda_A \|\boldsymbol{w}\|^2 + \frac{1}{\ell}\sum_{i=1}^{\ell}\xi_i + \frac{\lambda_I}{N^2}\sum_{i=1}^{N} W_{ij}(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j))^2$$

$$\text{úgy, hogy} \quad y_i(\boldsymbol{w}'\boldsymbol{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, \ell$$

- $\equiv$ megoldani az SVM optimalizálási feladatát az alábbi kernellel:

$$k_{\mathrm{lsvm}}(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{k}_x' \left(\frac{N^2}{2}\frac{\lambda_A}{\lambda_I}\boldsymbol{I} + \boldsymbol{L}^t\boldsymbol{K}\right)^{-1} \boldsymbol{L}\boldsymbol{k}_z$$

LapSVM

- induktív tanulási módszer: SVM + “simasági” feltétel: *ha két pont egy nagy sűrűségű régióban közel található egymáshoz, akkor azok címkéi is hasonlóan kell viselkedjenek*
- új optimalizálási feladat **(Sindhwani, Niyogi, Belkin 2005)**:

$$\min \quad \lambda_A \|\boldsymbol{w}\|^2 + \frac{1}{\ell}\sum_{i=1}^{\ell}\xi_i + \frac{\lambda_I}{N^2}\sum_{i=1}^{N} W_{ij}(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j))^2$$

$$\text{úgy, hogy} \quad y_i(\boldsymbol{w}'\boldsymbol{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, \ell$$

- $\equiv$ megoldani az SVM optimalizálási feladatát az alábbi kernellel:

$$k_{\mathrm{lsvm}}(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{k}_x' \left(\frac{N^2}{2}\frac{\lambda_A}{\lambda_I}\boldsymbol{I} + \boldsymbol{L}^t\boldsymbol{K}\right)^{-1} \boldsymbol{L}\boldsymbol{k}_z$$

LapSVM

- induktív tanulási módszer: SVM + "simasági" feltétel: *ha két pont egy nagy sűrűségű régióban közel található egymáshoz, akkor azok címkéi is hasonlóan kell viselkedjenek*
- új optimalizálási feladat **(Sindhwani, Niyogi, Belkin 2005)**:

$$\begin{array}{ll} \min & \lambda_A \|\boldsymbol{w}\|^2 + \frac{1}{\ell}\sum_{i=1}^{\ell} \xi_i + \frac{\lambda_I}{N^2}\sum_{i=1}^{N} W_{ij}(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j))^2 \\ \text{úgy, hogy} & y_i(\boldsymbol{w}'\boldsymbol{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \dots, \ell \end{array}$$

- $\equiv$ megoldani az SVM optimalizálási feladatát az alábbi kernellel:

$$k_{\mathrm{lsvm}}(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{k}_x' \left(\frac{N^2}{2} \frac{\lambda_A}{\lambda_I} \boldsymbol{I} + \boldsymbol{L}^t \boldsymbol{K} \right)^{-1} \boldsymbol{L}\boldsymbol{k}_z$$

Szomszédsági kernel

- klaszter-feltételen alapszik: *ha két pont azonos klaszterben van, akkor nagy valószínűséggel azonos osztályban is*
- ötlet: minden pontot helyettesítünk a pont szomszédainak átlagával **(Weston, Leslie, Zhou, Elisseeff, Noble 2004)**:

$$\phi_{\mathrm{nbd}}(\boldsymbol{x}) = \frac{1}{|N(\boldsymbol{x})|} \sum_{\boldsymbol{x}' \in N(\boldsymbol{x})} \phi_{\mathrm{b}}(\boldsymbol{x}')$$

  ahol $N(\cdot)$ a szomszédsági függvény
- a kernel így:

$$k_{\mathrm{nbd}}(\boldsymbol{x}, \boldsymbol{z}) = \frac{\sum_{\boldsymbol{x}' \in N(\boldsymbol{x}), \boldsymbol{z}' \in N(\boldsymbol{z})} k(\boldsymbol{x}', \boldsymbol{z}')}{|N(\boldsymbol{x})||N(\boldsymbol{z})|}$$

Szomszédsági kernel

- klaszter-feltételen alapszik: *ha két pont azonos klaszterben van, akkor nagy valószínűséggel azonos osztályban is*
- ötlet: minden pontot helyettesítünk a pont szomszédainak átlagával **(Weston, Leslie, Zhou, Elisseeff, Noble 2004)**:

$$\phi_{\mathrm{nbd}}(\boldsymbol{x}) = \frac{1}{|N(\boldsymbol{x})|} \sum_{\boldsymbol{x}' \in N(\boldsymbol{x})} \phi_{\mathrm{b}}(\boldsymbol{x}')$$

  ahol $N(\cdot)$ a szomszédsági függvény
- a kernel így:

$$k_{\mathrm{nbd}}(\boldsymbol{x}, \boldsymbol{z}) = \frac{\sum_{\boldsymbol{x}' \in N(\boldsymbol{x}), \boldsymbol{z}' \in N(\boldsymbol{z})} k(\boldsymbol{x}', \boldsymbol{z}')}{|N(\boldsymbol{x})||N(\boldsymbol{z})|}$$

ISOMAP

- valójában dimenziócsökkentő módszer **(Tenenbaum, de Silva, Langford 2000)**
- ha az adatok egy manifoldon helyezkednek el, akkor a gráf alapú távolságok (**legrövidebb utak**) megközelítik a manifoldon vett távolságokat
- gráfok: k legközelebbi szomszéd, vagy ε-szomszédsági
- $\Rightarrow$ $\boldsymbol{G}$ távolsági mátrix
- az ISOMAP kernel abba a térbe transzformálja a pontokat, ahol az euklideszi távolságok megegyeznek a gráftávolságokkal
- a kernel:

$$\boldsymbol{K}_{\mathrm{isomap}} = \tau(\boldsymbol{G}) = -\frac{1}{2}\boldsymbol{J}\boldsymbol{G}^{(2)}\boldsymbol{J},$$

  ahol $\boldsymbol{J} = \boldsymbol{I} - \frac{1}{N}\mathbf{1}\mathbf{1}'$ a centrálási mátrix, és a $\tau(G) = -\frac{1}{2}\boldsymbol{J}\boldsymbol{G}^{(2)}\boldsymbol{J}$ operátor a távolságokat skalárszorzattá alakítja

ISOMAP

- valójában dimenziócsökkentő módszer **(Tenenbaum, de Silva, Langford 2000)**
- ha az adatok egy manifoldon helyezkednek el, akkor a gráf alapú távolságok (**legrövidebb utak**) megközelítik a manifoldon vett távolságokat
- gráfok: k legközelebbi szomszéd, vagy ε-szomszédsági
- $\Rightarrow$ $\boldsymbol{G}$ távolsági mátrix
- az ISOMAP kernel abba a térbe transzformálja a pontokat, ahol az euklideszi távolságok megegyeznek a gráftávolságokkal
- a kernel:

$$\boldsymbol{K}_{\text{isomap}} = \tau(\boldsymbol{G}) = -\frac{1}{2}\boldsymbol{J}\boldsymbol{G}^{(2)}\boldsymbol{J},$$

ahol $\boldsymbol{J} = \boldsymbol{I} - \frac{1}{N}\mathbf{1}\mathbf{1}'$ a centrálási mátrix, és a $\tau(G) = -\frac{1}{2}\boldsymbol{J}\boldsymbol{G}^{(2)}\boldsymbol{J}$ operátor a távolságokat skalárszorzattá alakítja

Hierarchikus klaszter kernel

- hierarchikus klaszterezésen alapuló kernelek **(Bodó 2008)**
- hierarchikus klaszterezés:
  - **ponttávolság** mérése: $d(\boldsymbol{x}, \boldsymbol{z})$ (általában euklideszi)
  - **klasztertávolság** mérése: $D(C_1, C_2)$

1. egyszerű láncmódszer (*single linkage*)

$$D(C_1, C_2) = \min \{ d(\boldsymbol{x}, \boldsymbol{z}) \mid \boldsymbol{x} \in C_1, \boldsymbol{z} \in C_2 \}$$

2. teljes láncmódszer (*complete linkage*)

$$D(C_1, C_2) = \max \{ d(\boldsymbol{x}, \boldsymbol{z}) \mid \boldsymbol{x} \in C_1, \boldsymbol{z} \in C_2 \}$$

3. átlagos láncmódszer (*average linkage*)

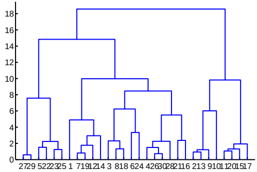
$$D(C_1, C_2) = \frac{1}{|C_1| \cdot |C_2|} \sum_{i=1}^{|C_1|} \sum_{j=1}^{|C_2|} d(\boldsymbol{x}_i, \boldsymbol{z}_j)$$

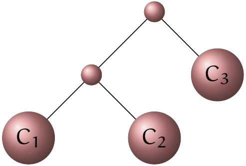
- **utrametricitás**: ha

$$
\begin{aligned}
D(C_1, C_2) &\leq D(C_1, C_3) \\
D(C_1, C_2) &\leq D(C_2, C_3)
\end{aligned}
$$

akkor

$$D(C_1, C_2) \leq D(C_{12}, C_3)$$

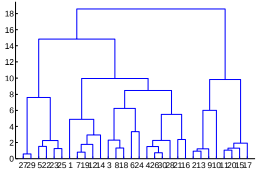




- *Tétel*: ha $\boldsymbol{M}$ utrametrikus, akkor $-\frac{1}{2}\boldsymbol{JMJ}$ egy Gram-mátrix, amely azon vektorok skalárszorzatait tartalmazza, melyek négyzetes euklideszi távolságai az $\boldsymbol{M}$-ben vannak
- kernel:

$$\boldsymbol{K}_{\text{hck}} = -\frac{1}{2}\boldsymbol{JMJ}$$

- kiterjesztés: ISOMAP módszerrel való kombinálás: $d(\cdot, \cdot) \rightarrow$ gráftávolságok; nem összefüggő gráf esetén a gráf összefüggővé alakítása

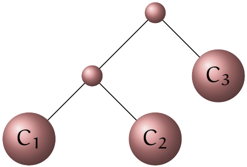


- **utrametricitás**: ha

$$\begin{aligned} D(C_1, C_2) &\leq D(C_1, C_3) \\ D(C_1, C_2) &\leq D(C_2, C_3) \end{aligned}$$

akkor

$$D(C_1, C_2) \leq D(C_{12}, C_3)$$



- *Tétel*: ha $\boldsymbol{M}$ utrametrikus, akkor $-\frac{1}{2}\boldsymbol{JMJ}$ egy Gram-mátrix, amely azon vektorok skalárszorzatait tartalmazza, melyek négyzetes euklideszi távolságai az $\boldsymbol{M}$-ben vannak
- kernel:

$$\boldsymbol{K}_{\mathrm{hck}} = -\frac{1}{2}\boldsymbol{JMJ}$$

- kiterjesztés: ISOMAP módszerrel való kombinálás: $d(\cdot, \cdot) \rightarrow$ gráftávolságok; nem összefüggő gráf esetén a gráf összefüggővé alakítása

Átsúlyozó klaszter kernelek

- az alapkernel átsúlyozása a klaszterbe tartozás alapján **(Bodó, Csató 2010)**
- kernelek kombinációja (néhány tulajdonság):
  - ha $\boldsymbol{K}_1, \boldsymbol{K}_2$ p.szd., akkor $\boldsymbol{K}_1 + \boldsymbol{K}_2$ is p.szd.
  - ha $\boldsymbol{K}$ p.szd., $a > 0$, akkor $a\boldsymbol{K}$ is p.szd.
  - ha $\boldsymbol{K}_1, \boldsymbol{K}_2$ p.szd., akkor $\boldsymbol{K}_1 \odot \boldsymbol{K}_2$ is p.szd., ahol $\odot$ a Hadamard-szorzatot jelöli
- átsúlyozó kernelek:

$$\boldsymbol{K}_{\mathrm{rck}} = \boldsymbol{K}_{\mathrm{rw}} \odot \boldsymbol{K}_{\mathrm{b}}$$

$$k_{\mathrm{rw}}(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{U}_{\cdot\boldsymbol{x}} - \boldsymbol{U}_{\cdot\boldsymbol{z}}\|^2}{2\sigma^2}\right) \tag{1}$$

$$\boldsymbol{K}_{\mathrm{rw}} = \boldsymbol{U}'\boldsymbol{U} + \frac{\alpha}{N}\mathbf{1}\mathbf{1}', \quad \alpha \in [0, 1) \tag{2}$$

$$\boldsymbol{K}_{\mathrm{rw}} = \beta\boldsymbol{U}'\boldsymbol{U} + \frac{1}{N}\mathbf{1}\mathbf{1}', \quad \beta \in (0, \infty) \tag{3}$$

Köszönöm a figyelmet!