IDA ELŐADÁS II.

Bolgár Bence

2014. október 18.

I. Bevezetés

Az adatfúzió célja a prediktív teljesítmény javítása több (heterogén) információforrás felhasználásával. Elvárásaink az adatfúziós módszerekkel szemben:

- Használatával javuljon a prediktív teljesítmény.
- Legyen használható heterogén, akár különböző formátumú (pl. vektoriális, gráfos) adatok esetén is.
- Skálázódjon jól (lineárisan) az információforrások számával.
- Legyen lehetőség szakértői tudás integrálására.
- Legyen automatizált, könnyen használható, felhasználóbarát (pl. paraméterezés kérdése).
- Legyen hatékonyan számítható (pl. párhuzamosítható).
- Álljanak rendelkezésre matematikai-statisztikai garanciák a teljesítményt illetően.
- Kezelje a hiányos ill. zajos adatokat.

A fúzió szintje szerint három kategóriát különböztethetünk meg: adatszintű (alacsony szintű, korai), köztes, döntésszintű (magas szintű, késői) fúzió.

II. Adatszintű fúzió

Adatszintű fúziónál adatok direkt, leírás-szintű kombinációja történik. Ennek legegyszerűbb módja a vektortér-integráció (VSI), amely a vektoriális adatok konkatenációját (egymás után fűzését) jelenti. Ekkor a kombinált adathalmaz

$$\mathcal{X} = \bigoplus_{k=1}^{n} \left\{ \mathcal{X}_k \subset \mathbb{R}^{d_k} \right\}$$

formában írható, ahol $\mathcal{X}_k$ jelöli az egyes adatforrásokat. Az

$$\boldsymbol{x} \in \mathcal{X} \cong \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \subset \mathbb{R}^{d_1 \cdots d_n}$$

kombinált minta a következőképpen áll elő:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_n \end{bmatrix},$$

ahol $\boldsymbol{x}_i \in \mathcal{X}_i$, az $\boldsymbol{x}$ minta reprezentációja az $i.$ adatforrásban. A módszer jellemzői:

- Előnyök: egyszerű, hatékonyan számítható, jól skálázódik, automatizált.
- Hátrányok: nincs lehetőség szakértői tudás integrálására, csak vektoriális adatokra alkalmazható, hiányos adatokat és zajt nem kezel.

III. Köztes fúzió

A köztes fúzió az adatoknak egy átmeneti reprezentációját használja fel. Ebben a szakaszban néhány példát mutatunk a „közös nyelvre” – a teljesség igénye nélkül.

III.A. Valószínűségi fúzió

A valószínűségi fúzióra láttunk példát az előző előadásban (modellek kombinálása naiv bayesi alapon); ide tartoznak még a később tárgyalt Bayes-hálók, mixture modellek stb. Megjegyzendő, hogy ezek közül a legtöbb döntésszintű adatfúzióra is alkalmas.

Előnyök:

- Lehetőség *a priori* ismeretek normatív kombinációjára.
- Bizonytalanság, hiány közvetlen kezelése.
- Kimenet: jól értelmezhető valószínűségi állítások.
- Évtizedes tapasztalatok, évszázadok (!) óta finomított matematikai háttér, elméleti garanciák.
- Egyre hatékonyabb algoritmusok.

Hátrányok:

- Nagy számításigény.
- Általában kedvezőtlen skálázódás.
- Nehézkes az ismeretek transzformációja a valószínűségek nyelvére.

III.B. Hálózatos megközelítések

Rendkívül elterjedt egyes szakterületeken:

- Bioinformatika. A tudás itt több (kapcsolt!) szinten is megjelenik: génszabályozási, fehérje-fehérje interakciós, metabolikus, jelátviteli, betegség–betegség stb. hálózatok. Ezen heterogén tudás fúziója az elmúlt években a bioinformatika egyik legintenzívebben kutatott területévé vált.
- Telekommunikáció. Itt gondolhatunk például az Internet szerkezetének feltárását, hatékony kereső- és ajánlóalgoritmusok implementációját, reklámozási stratégiák kialakítását célzó kutatásokra, amelyeknek gyakran központi eleme az adatfúzió.
- Kapcsolati hálók. A kapcsolati hálók integratív elemzése a legkülönbözőbb területeket érinti: társadalomtudományok, epidemiológia, közgazdaságtan stb.

A hálózatos adatok fúziójára a fenti szakterületeken rengeteg *ad hoc* megoldást dolgoztak ki, amelyek számbavétele szinte lehetetlen vállalkozás – standard, „bevált” módszerek ugyanis egyelőre nem léteznek. Gyakran alkalmazzák a hálózatok valamilyen direkt integrációját (pl. tranzitív lezárás), visszavezetést optimalizációs feladatokra, hálózatillesztést, valószínűség-alapú eljárásokat stb.

Előnyök:

- Intenzív kutatás.
- Stabil matematikai alapok (pl. gráfelmélet) és empirikus eredmények.

Hátrányok:

- Konstrukció nem triviális.
- Szakértői tudás bevitele nem triviális.
- Bizonytalanság és hiány kezelése gyakran nehézkes.

III.C. Kernel fúzió

A Multiple Kernel Learning (MKL) eljárások a hagyományos kernel gépek kiterjesztései, ahol a fúzió a kernel mátrixok szintjén történik. A konkrét módszerek előtt nézzük a közös előnyöket és hátrányokat.

Előnyök:

- Rendszerint jó prediktív teljesítmény.
- Heterogén információforrások jól integrálhatók (amennyiben tudunk hasonlóságokat származtatni).

- Lineáris skálázódás az információforrások (kernelek) számával.
- Lehetőség szakértői tudás integrálására (pl. kernel függvények tervezésével).
- Automatizált algoritmusok.
- Hatékonyan számítható, jól skálázódik nagy mennyiségű adatra is, gyakran párhuzamosítható.
- Matematikai-statisztikai garanciák.

Hátrányok:

- Paraméterezés általában nem triviális.
- Bizonytalanság és hiány kezelése nehézkes.

III.C.1. Statikus kombináció

A kernelek kombinációjának legegyszerűbb módszerei a „statikus" lineáris vagy nemlineáris kombinációk:

- Kernelek uniform átlagolása: $\boldsymbol{K} = \frac{1}{n}\sum_{k=1}^{n} \boldsymbol{K}_k$.
- Kernelek súlyozott átlagolása: $\boldsymbol{K} = \sum_{k=1}^{n} d_k \boldsymbol{K}_k$, ahol $\|\boldsymbol{d}\| = 1$ valamilyen $\|\cdot\|$ normában.
- Kernelek szorzása: $\boldsymbol{K} = \prod_{k=1}^{n} \boldsymbol{K}_k$.
- Kernelek Hadamard-szorzata: $\boldsymbol{K} = \boldsymbol{K_1} \circ \boldsymbol{K_2} \circ \cdots \circ \boldsymbol{K_n}$, ahol $(\boldsymbol{A} \circ \boldsymbol{B})_{ij} = \boldsymbol{A}_{ij}\boldsymbol{B}_{ij}$.

És még számos más eljárás. Egyszerűségük ellenére általában jó prediktív teljesítményt nyújtanak.

III.C.2. Adaptív kombináció

A kernel gép tanítása és az adatfúzió egy lépésben is történhet, ha az optimalizációba a lineáris kombináció súlyainak tanulását is beépítjük. Ennek előnye, hogy a fúzió ekkor „adaptívan", mindig az aktuális feladatot figyelembe véve történik. A hagyományos SVM primál feladatát a következőképpen módosítjuk:

$$\begin{aligned} \min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{d}} \quad & \frac{1}{2}\sum_k \|\boldsymbol{w}_k\|^2 + C\sum_i \xi_i \\ s.t. \quad & y_i\left(\sum_k \boldsymbol{w}_k^T\left(\sqrt{d_k}\phi_k\left(\boldsymbol{x}_i\right)\right) + b\right) \geq 1 - \xi_i, \\ & \|\boldsymbol{d}\|_p = 1, \quad \xi_i \geq 0, \quad d_k \geq 0. \end{aligned}$$

A fenti feladatban a $\boldsymbol{w} \leftarrow \begin{bmatrix} \boldsymbol{w}_1 \\ \vdots \\ \boldsymbol{w}_n \end{bmatrix}$ és a $\phi(\boldsymbol{x}_i) \leftarrow \begin{bmatrix} \sqrt{d_1}\phi_1(\boldsymbol{x}_i) \\ \vdots \\ \sqrt{d_n}\phi_n(\boldsymbol{x}_i) \end{bmatrix}$ helyettesítéssel éltünk. Vegyük észre, hogy ez nem más, mint a vektortér-integráció azzal a különbséggel, hogy az egyes reprezentációk meg vannak szorozva egy $\sqrt{d_k}$ súllyal! A hagyományos SVM-hez nagyon hasonló gondolatmenetet követve, valamint a skalárszorzat tulajdonságait felhasználva a $\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j) = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ összefüggés a következőképen módosul:

$$\begin{bmatrix} \sqrt{d_1}\phi_1(\boldsymbol{x}_i) \\ \vdots \\ \sqrt{d_n}\phi_n(\boldsymbol{x}_i) \end{bmatrix}^T \begin{bmatrix} \sqrt{d_1}\phi_1(\boldsymbol{x}_j) \\ \vdots \\ \sqrt{d_n}\phi_n(\boldsymbol{x}_j) \end{bmatrix} = \sum_k d_k k_k(\boldsymbol{x}_i, \boldsymbol{x}_j),$$

azaz a kernelek lineáris kombinációjához jutunk, ahol az algoritmus egyben az optimális súlyozást is meg fogja adni. Külön figyelmet érdekel a $\|\boldsymbol{d}\|_p = 1$ kényszer – enélkül a d_k súlyokat minden határon túl növelve a célfüggvény tetszőlegesen csökkenthető lenne. Az $\|\boldsymbol{a}\|_p = \left(\sum_l a_l^p\right)^{1/p}$ normát L_p-normának nevezzük, és p különböző értékeire más-más jellegű regularizációt kapunk: $p < 2$ esetén kevés súly fog magas értéket kapni (ritka kombináció, „legjobb" kernelek kiválasztása), nagyobb p esetén egyenletesebb lesz a súlyok eloszlása.

III.C.3. Kernel alignment

Tekintsünk egy hagyományos kétosztályos osztályozási feladatot, azaz legyen $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{P}$, ahol $\boldsymbol{x}_i \in \mathcal{X} \subset \mathbb{R}^d$, $y_i \in \mathcal{Y} = \{-1, +1\}$. Könnyen megmutatható, hogy ekkor az $\boldsymbol{y}\boldsymbol{y}^T$ szorzat egy ideális kernel, amely minden tanítómintát helyesen osztályoz:

$$\begin{aligned} y_i = y_j \quad &\Leftrightarrow \quad (\boldsymbol{y}\boldsymbol{y}^T)_{ij} = 1, \\ y_i \neq y_j \quad &\Leftrightarrow \quad (\boldsymbol{y}\boldsymbol{y}^T)_{ij} = -1. \end{aligned}$$

[Gondolhatunk arra, hogy a kernel függvény felfogható úgy is, mint egy hasonlóságfüggvény]. Legyen most adott n darab különböző kernelünk, $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_n$, amelyek különböző információforrásokból származnak (báziskernelek). Keressük azt a $\boldsymbol{K}$ pozitív szemidefinit kernelt, amely a lehető „legközelebb" van a báziskernelekhez, valamint az ideális kernelhez:

$$\begin{aligned} \min_{\boldsymbol{K}} \quad & L(\boldsymbol{K}, \boldsymbol{K}_1, \ldots, \boldsymbol{K}_n, \boldsymbol{y}\boldsymbol{y}^T) \\ s.t. \quad & \boldsymbol{K} \succeq 0, \end{aligned}$$

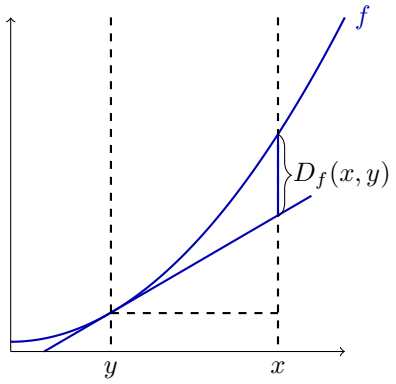
ahol L valamilyen veszteségfüggvény. A továbbiakban az egyszerűség kedvéért egy kernelen mutatjuk be a veszteségfüggvényeket. Mátrix-közelségi problémáknál gyakran használt függvény a Frobenius-norma:

$$\begin{aligned} \min_{\boldsymbol{K}} \quad & \left\| \boldsymbol{K} - \boldsymbol{y}\boldsymbol{y}^T \right\|_F \\ s.t. \quad & \boldsymbol{K} \succeq 0, \end{aligned}$$

ahol $\|A\|_F = \sqrt{\sum_i \sum_j |A_{ij}|^2}$. A Frobenius-norma hátránya, hogy nem garantálja a kapott mátrix pozitív szemidefinit tulajdonságát. Vegyük hát a Frobenius-norma egy általánosítását, amelyet Bregman-divergencia néven ismerünk. A Bregman-divergencia definíciója valós értékű függvényekre:

$$D_f(x, y) := f(x) - f(y) - f'(y)(x - y),$$

ahol $f : \mathbb{R} \to \mathbb{R}$, azaz szemléletesen a következőt jelenti:



Megjegyezzük, hogy az így definiált divergencia általánosságban nem szimmetrikus és nem teljesíti a háromszög-egyenlőtlenséget. A Bregman-divergencia triviálisan kiterjeszthető a többdimenziós esetre, azaz legyen most $f : \mathbb{R}^d \to \mathbb{R}$, és

$$D_f(\boldsymbol{x}, \boldsymbol{y}) := f(\boldsymbol{x}) - f(\boldsymbol{y}) - \langle \nabla f(\boldsymbol{y}), (\boldsymbol{x} - \boldsymbol{y}) \rangle .$$

1. Példa. *Legyen $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^2$. Ekkor $D_f(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{x} \rangle - \langle \boldsymbol{y}, \boldsymbol{y} \rangle - \langle 2\boldsymbol{y}, \boldsymbol{x} - \boldsymbol{y} \rangle = \|\boldsymbol{x} - \boldsymbol{y}\|^2$, ami éppen az euklideszi távolság.*

Végül definiáljuk a Bregman-divergenciát mátrixokra is, $f : \mathbb{R}^{n \times m} \to \mathbb{R}$.

$$D_f(\boldsymbol{X}, \boldsymbol{Y}) := f(\boldsymbol{X}) - f(\boldsymbol{Y}) - \langle \nabla f(\boldsymbol{Y}), (\boldsymbol{X} - \boldsymbol{Y}) \rangle_F ,$$

ahol $\langle \cdot, \cdot \rangle_F$ a Frobenius-szorzat, azaz $\langle \boldsymbol{A}, \boldsymbol{B} \rangle_F = \operatorname{tr}(\boldsymbol{A}\boldsymbol{B}^T) = \sum_i \sum_j A_{ij} B_{ij}$.

2. Példa. *Könnyen belátható, hogy* $f(\boldsymbol{X}) = \|\boldsymbol{X}\|_F^2 \rightsquigarrow D_f(\boldsymbol{X}, \boldsymbol{Y}) = \|\boldsymbol{X} - \boldsymbol{Y}\|_F^2$.

3. Példa. *Legyen most* $f(\boldsymbol{X}) = -\log\det \boldsymbol{X}$. *Ekkor*

$$\begin{aligned} D_f(\boldsymbol{X}, \boldsymbol{Y}) &= -\log\det \boldsymbol{X} + \log\det \boldsymbol{Y} - \left\langle (\det \boldsymbol{Y})^{-1} \operatorname{adj} \boldsymbol{Y}, \boldsymbol{X} - \boldsymbol{Y} \right\rangle_F \\ &= -\log\det \boldsymbol{X}\boldsymbol{Y}^{-1} - n + \operatorname{tr}(\boldsymbol{X}\boldsymbol{Y}^{-1}), \end{aligned}$$

ahol $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{n\times n}$.

A 3. példában látott divergenciát LogDet divergenciának is nevezik. A kernel alignment eljárások szempontjából igen fontosak az alábbi tulajdonságai:

1. Garantálja a pozitív szemidefinit tulajdonságot.
2. Megőrzi a rangot.
3. Első argumentumában konvex.

Az 1. tulajdonság miatt az optimalizációból elhagyhatjuk a $\boldsymbol{K} \succeq 0$ kényszert. A 2. tulajdonság lehetővé teszi az ún. alacsony rangú approximációt, amellyel hatékonyan kezelhetők nagy méretű adathalmazok is, míg a 3. tulajdonság az optimalizációs stratégia kidolgozását könnyíti meg.

IV. Döntésszintű fúzió

A döntésszintű fúzió során a forrásonként külön-külön elvégzett elemzések (pl. osztályozás, regresszió) eredményeit egyesítjük. A korábban vett boosting, MoE, szavazásos eljárások stb. variánsai mind felfoghatók késői adatfúziós eljárásként. Itt csak egy speciális esetet, a sorrendi fúziót fogjuk megvizsgálni.

Előnyök:

- Rugalmas, gyakorlatilag bármi kombinálható; forrásonként akár különböző algoritmusokat is használhatunk.
- Előbbi egyben a szakértői tudás bevitelét is segíti.

Hátrányok:

- Könnyen számításigényessé válhat.
- A fúziót nem (közvetlenül) az adat vezérli, mivel az a döntések szintjén egyáltalán nem jelenik meg.

Sorrendi fúziót gyakran használnak információ-visszakeresési feladatokban, azaz pl. webes keresőmotorokban, ajánló rendszerekben, vagy a bio- és kemoinformatika területén (génprioritizálás, virtuális screening). A sorrendekben gyakran nem csupán egy rendezési reláció áll rendelkezésre, hanem egzakt pontszámok is, amelyeket egyfajta relevanciaként is értelmezhetünk (ti. a lekérdezés szempontjából).

Néhány egyszerű stratégia a sorrendi fúzióra:

- **Sum rank.** Adott entitás rangja a különböző sorrendekben elért rangjainak összege alapján számolható.
- **Sum score.** Adott entitás rangja a különböző sorrendekben található pontszámainak összege alapján számolható.
- **Pareto ranking.** Adott entitás rangja az alapján alakul, hogy hány entitás ért el nála jobb rangot *minden* sorrendben.
- **Rank vote.** Minden sorrend szavaz az első k elemére. A végső sorrend a kapott szavazatok alapján alakul.
- **Parallel selection.** Minden sorrendből kiválasztjuk a soron következő elemet és beillesztjük a végső sorrendbe. Ha olyat választunk, ami már bekerült, akkor helyette a következőt választjuk.

A döntetleneket a sum rank vagy sum score módszerekkel szokás feloldani.

4. Példa. *Tekintsük az alábbi sorrendeket:*

A	*10*	B	*10*	A	*8*
B	*8*	A	*6.5*	D	*7*
C	*7*	C	*3*	B	*7*
D	*6.5*	D	*2*	C	*6*
E	*6*	E	*1*	E	*5*

Az egyes módszerekkel kialakuló kombinált sorrendek:

- *Sum rank:* $A(4)$, $B(6)$, $C(10)$, $D(10)$, $E(15)$.
- *Sum score:* $B(25)$, $A(24.5)$, $C(16)$, $D(15.5)$, $E(12)$.
- *Pareto ranking:* $A(0)$, $B(0)$, $D(1)$, $C(2)$, $E(4)$.
- *Rank vote (*$k = 3$*):* $A(3)$, $B(3)$, $C(2)$, $D(1)$, $E(0)$.
- *Rank vote (*$k = 2$*):* $A(3)$, $B(2)$, $D(1)$, $C(0)$, $E(0)$.
- *Parallel selection (balról jobbra):* A, B, D, C, E.