Adaptív dinamikus szegmentálás idősorok indexeléséhez

IPM-08irAREAE kurzus cikkfeldolgozás

Balassi Márton[1]
Englert Péter[1]
Tömösy Péter[1]

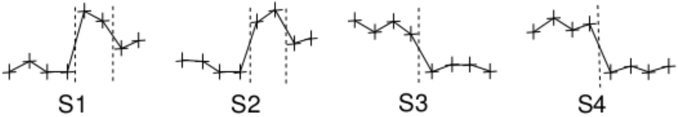
[1] Eötvös Loránd Tudományegyetem – Informatikai Kar

2013. november 20.

Tartalom

Tartalom

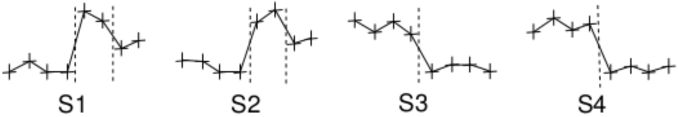
Feladat



Idősorok hatékony szegmentálása

- Adatsorok hasonlósági keresése
- Dinamikus, adaptív indexelés
- Empirikus eredmények

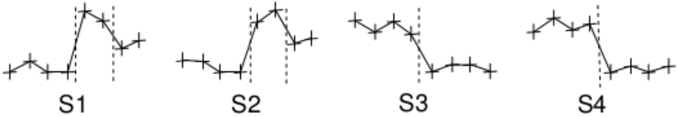
Feladat



Idősorok hatékony szegmentálása

- Adatsorok hasonlósági keresése
- Dinamikus, adaptív indexelés
- Empirikus eredmények

Feladat



Idősorok hatékony szegmentálása

- Adatsorok hasonlósági keresése
- Dinamikus, adaptív indexelés
- Empirikus eredmények

Motiváció

Széles alkalmazási terület

- Orvosi alkalmazás
- Tőzsde
- Szenzorhálózatok
- ...

Motiváció

Széles alkalmazási terület

- Orvosi alkalmazás
- Tőzsde
- Szenzorhálózatok
- ...

Motiváció

Széles alkalmazási terület

- Orvosi alkalmazás
- Tőzsde
- Szenzorhálózatok
- ...

Motiváció

Széles alkalmazási terület

- Orvosi alkalmazás
- Tőzsde
- Szenzorhálózatok
- ...

Motiváció

Széles alkalmazási terület

- Orvosi alkalmazás
- Tőzsde
- Szenzorhálózatok
- ...

Javítási ötlet

- Eddig globálisan indexelték az idősorhalmazokat
- Nem használtak felülről korlátozó függvényeket

Motiváció

Széles alkalmazási terület

- Orvosi alkalmazás
- Tőzsde
- Szenzorhálózatok
- ...

Javítási ötlet

- Eddig globálisan indexelték az idősorhalmazokat
- Nem használtak felülről korlátozó függvényeket

Definíciók

Idősor

A továbbiakban legyen adott n pozitív egész szám, valamint $X = (x_1, x_2, \ldots, x_n)$ és $Y = (y_1, y_2, \ldots, y_n)$ n hosszú, valós számokból álló idősorok. Ezek halmaza legyen $\mathcal{TS}$.

Definíciók

Idősor

A továbbiakban legyen adott n pozitív egész szám, valamint $X = (x_1, x_2, \ldots, x_n)$ és $Y = (y_1, y_2, \ldots, y_n)$ n hosszú, valós számokból álló idősorok. Ezek halmaza legyen $\mathcal{TS}$.

Távolság

X és Y euklideszi távolsága $D(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$.

Definíciók

Távolság

X és Y euklideszi távolsága $D(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$.

Teljes illesztés

Legyen adott $Q \in \mathcal{TS}$ és $0 < \epsilon \in \mathbb{R}$. Keressük azokat az S idősorokat, melyekre $D(Q, S) \leq \epsilon$. Ha a hasonlósági mérték az euklideszi távolság, az idősorok pedig azonos hosszúságúak, ezt teljes illesztésnek nevezzük.

Korábbi eredmények

Alulról korlátozó dimenziócsökkentés

Létezik $D_{LB}(\cdot, \cdot)$ függvény, melyre ha $S_i(i \in \{1, 2\})$ idősor reprezentációja $\tilde{S}_i(i \in \{1, 2\})$, akkor $D_{LB}(\tilde{S}_1, \tilde{S}_2) \leq D(S_1, S_2)$.

Korábbi eredmények

Alulról korlátozó dimenziócsökkentés

Létezik $D_{LB}(\cdot, \cdot)$ függvény, melyre ha $S_i (i \in \{1, 2\})$ idősor reprezentációja $\tilde{S}_i (i \in \{1, 2\})$, akkor $D_{LB}(\tilde{S}_1, \tilde{S}_2) \leq D(S_1, S_2)$. Ekkor ha $D_{LB}(\tilde{Q}, \tilde{S}) > \epsilon$, akkor az S által reprezentált idősorokat már nem kell vizsgálnunk, ugyanis azok tényleges távolsága Q-tól biztos, hogy ϵ-nál nagyobb.

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

Korábbi eredmények

Felhasznált módszerek

- Dimenziócsökkentésre:
  - diszkrét Fourier-transzformációt
  - szinguláris felbontás
  - diszkrét wavelet transzformáció
  - ...
- Indexelésre:
  - R-fa
  - SAX
  - iSAX

A cikk fő eredményei

Eredmények

- Felülről korlátozó dimenziócsökkentés
- DSTree
- Hisztogramkészítés

A cikk fő eredményei

Eredmények

- Felülről korlátozó dimenziócsökkentés
- DSTree
- Hisztogramkészítés

A cikk fő eredményei

Eredmények

- Felülről korlátozó dimenziócsökkentés
- DSTree
- Hisztogramkészítés

Tartalom

Adaptív szakaszonként konstans approximáció

APCA

Egy X idősort $(X_1, X_2, \ldots, X_m)$ szegmenseire bontjuk, ahol $m \leq n$ és $\forall j \in \{1, \ldots, m\} : X_j = (x_{r_{j-1}+1}, \ldots, x_{r_j})$ valamely $0 = r_0 < r_1 < \cdots < r_m = n$ indexsorozatra.

Adaptív szakaszonként konstans approximáció

APCA

Egy X idősort $(X_1, X_2, \ldots, X_m)$ szegmenseire bontjuk, ahol $m \leq n$ és $\forall j \in \{1, \ldots, m\} : X_j = (x_{r_{j-1}+1}, \ldots, x_{r_j})$ valamely $0 = r_0 < r_1 < \cdots < r_m = n$ indexsorozatra.
Egy X_j szegmens reprezentációja a (μ_j, r_j) pár lesz, ahol $\mu_j = \frac{\sum_{k=r_{j-1}+1}^{r_j} s_k}{r_j - r_{j-1}}$ a szegmens értékeinek átlaga. X közelítő reprezentációja tehát $\tilde{X} = ((\mu_1, r_1), \ldots, (\mu_m, r_m))$.

Adaptív szakaszonként konstans approximáció

Illeszkedés

Azt mondjuk, hogy $\tilde{X}$ és $\tilde{Y}$ APCA reprezentációk illeszkednek, ha az előző bekezdés jelöléseivel az m és $r_0, \ldots, r_m$ értékeik megegyeznek, vagyis:

$$\tilde{X} = ((\mu_1^X, r_1), \ldots, (\mu_m^X, r_m)),$$

$$\tilde{Y} = ((\mu_1^Y, r_1), \ldots, (\mu_m^Y, r_m)).$$

Adaptív szakaszonként konstans approximáció

Lemma

Adott két egyforma hosszú idősor, X és Y, valamint illeszkedő APCA reprezentációik, $\tilde{X} = ((\mu_1^X, r_1), \ldots, (\mu_m^X, r_m))$ és $\tilde{Y} = ((\mu_1^Y, r_1), \ldots, (\mu_m^Y, r_m))$.

Adaptív szakaszonként konstans approximáció

Lemma

Adott két egyforma hosszú idősor, X és Y, valamint illeszkedő APCA reprezentációik, $\tilde{X} = ((\mu_1^X, r_1), \ldots, (\mu_m^X, r_m))$ és $\tilde{Y} = ((\mu_1^Y, r_1), \ldots, (\mu_m^Y, r_m))$.
Ekkor:

$$D(X, Y) \geq \sqrt{\sum_{i=1}^{m} (r_i - r_{i-1})(\mu_i^X - \mu_i^Y)^2}$$

Bővített adaptív szakaszonként konstans approximáció

EAPCA

A szegmensek reprezentációját az értékek szórásával egészíti ki, és ennek segítségével egy pontosabb alsó korlátot, valamint egy felső korlátot is definiál.

Bővített adaptív szakaszonként konstans approximáció

EAPCA

A szegmensek reprezentációját az értékek szórásával egészíti ki, és ennek segítségével egy pontosabb alsó korlátot, valamint egy felső korlátot is definiál.

$$\tilde{X} = ((\mu_1, \sigma_1, r_1), \ldots, (\mu_m, \sigma_m, r_m)), \text{ ahol}$$

$$\sigma_i = \sqrt{\frac{\sum_{j=r_{i-1}+1}^{r_i} s_j^2}{r_i - r_{i-1}} - \left(\frac{\sum_{j=r_{i-1}+1}^{r_i} s_j}{r_i - r_{i-1}}\right)^2}$$

értékek a reprezentált szegmens értékeinek szórása.

Bővített adaptív szakaszonként konstans approximáció

Tétel

Adott két egyforma hosszú idősor, X és Y, valamint illeszkedő EAPCA reprezentációik, $\tilde{X} = ((\mu_1^X, \sigma_1^X, r_1), \ldots, (\mu_m^X, \sigma_m^X, r_m))$ *és* $\tilde{Y} = ((\mu_1^Y, \sigma_1^Y, r_1), \ldots, (\mu_m^Y, \sigma_m^Y, r_m))$.

Bővített adaptív szakaszonként konstans approximáció

Tétel

Adott két egyforma hosszú idősor, X és Y, valamint illeszkedő EAPCA reprezentációik, $\tilde{X} = ((\mu_1^X, \sigma_1^X, r_1), \ldots, (\mu_m^X, \sigma_m^X, r_m))$ és $\tilde{Y} = ((\mu_1^Y, \sigma_1^Y, r_1), \ldots, (\mu_m^Y, \sigma_m^Y, r_m))$. Ekkor:

$$D(X, Y) \geq \sqrt{\sum_{i=1}^{m} (r_i - r_{i-1})[(\mu_i^X - \mu_i^Y)^2 + (\sigma_i^X - \sigma_i^Y)^2]},$$

$$D(X, Y) \leq \sqrt{\sum_{i=1}^{m} (r_i - r_{i-1})[(\mu_i^X - \mu_i^Y)^2 + (\sigma_i^X + \sigma_i^Y)^2]}.$$

Korlátok idősorok halmazaitól vett távolságra

Feladat

X idősor és $Y_1, \ldots, Y_l$ egyforma hosszú idősorok esetén a $\min_{1 \leq j \leq l} D(X, Y_j)$ távolságot szeretnénk alulról becsülni.

Korlátok idősorok halmazaitól vett távolságra

Jelölés

Legyen $X, Y_1, \ldots, Y_l$ mind egyforma hosszú, továbbá legyenek

$$\tilde{X} = ((\mu_1^X, \sigma_1^X, r_1), \ldots, (\mu_m^X, \sigma_m^X, r_m)),$$

$$\tilde{Y}_1 = ((\mu_1^{Y_1}, \sigma_1^{Y_1}, r_1), \ldots, (\mu_m^{Y_1}, \sigma_m^{Y_1}, r_m)),$$

$$\ldots,$$

$$\tilde{Y}_l = ((\mu_1^{Y_l}, \sigma_1^{Y_l}, r_1), \ldots, (\mu_m^{Y_l}, \sigma_m^{Y_l}, r_m))$$

az $X, Y_1, \ldots, Y_l$ idősorok illeszkedő EAPCA reprezentációi.

Korlátok idősorok halmazaitól vett távolságra

Jelölés

Jelölje az $\tilde{Y}_1, \ldots, \tilde{Y}_l$ halmazban i. szegmens minimális és maximális átlagát $\mu_i^{min} = \min_{1 \leq j \leq l} \mu_i^{Y_j}$ és $\mu_i^{max} = \max_{1 \leq j \leq l} \mu_i^{Y_j}$, szórását pedig $\sigma_i^{min} = \min_{1 \leq j \leq l} \sigma_i^{Y_j}$ és $\sigma_i^{max} = \max_{1 \leq j \leq l} \sigma_i^{Y_j}$.

Korlátok idősorok halmazaitól vett távolságra

Tétel

Adott $X, Y_1, \ldots, Y_l$ egyforma hosszú idősorok és illeszkedő $\tilde{X}, \tilde{Y}_1, \ldots, \tilde{Y}_l$ EAPCA reprezentációik. Ekkor:

Korlátok idősorok halmazaitól vett távolságra

Tétel

$$\min_{1\leq j\leq l} D(X, Y_j) \geq \sqrt{\sum_{i=1}^{m} (r_i - r_{i-1})(LB_i^{\mu} + LB_i^{\sigma})^2},$$

$$\max_{1\leq j\leq l} D(X, Y_j) \leq \sqrt{\sum_{i=1}^{m} (r_i - r_{i-1})(UB_i^{\mu} + UB_i^{\sigma})^2}.$$

Korlátok idősorok halmazaitól vett távolságra

Tétel

$$\min_{1\leq j\leq l} D(X, Y_j) \geq \sqrt{\sum_{i=1}^{m}(r_i - r_{i-1})(LB_i^{\mu} + LB_i^{\sigma})^2},$$

$$\max_{1\leq j\leq l} D(X, Y_j) \leq \sqrt{\sum_{i=1}^{m}(r_i - r_{i-1})(UB_i^{\mu} + UB_i^{\sigma})^2}.$$

Ahol $LB_i^{\mu}, LB_i^{\sigma}, UB_i^{\mu}, UB_i^{\sigma}$ *elemi függvényei a* $\{\mu_i;\ \sigma_i\}^{\{min;\, max;\, X\}}$ *halmaz elemeinek.*

Korlátok idősorok halmazaitól vett távolságra

Következmény

Ha az illeszkedő EAPCA reprezentációval adott idősorok halmazától vett távolságot szeretnénk alulról becsülni, ahhoz elég a szegmensek átlagainak és szórásainak minimumát és maximumát számon tartanunk.

DSTree

Reprezentáció

Az EAPCA szegmenseket meghatározza jobb végpontjuk, vagyis $(r_1, \ldots, r_m)$, ahol $0 < r_1 < r_2 < \cdots < r_m = n$.

DSTree

Reprezentáció

Az EAPCA szegmenseket meghatározza jobb végpontjuk, vagyis $(r_1, \ldots, r_m)$, ahol $0 < r_1 < r_2 < \cdots < r_m = n$.

Finomítás

Adott $SG_1 = (r_1, \ldots, r_m)$ és $SG_2 = (r'_1, \ldots, r'_{m'})$ szegmentálásokra SG_2 egylépéses finomítása SG_1-nek, ha $m' = m + 1$, és $\exists i_0 : 1 \leq i_0 < m$, hogy $1 \leq i \leq i_0$ esetén $r_i = r'_i$, $i_0 < i$ esetén pedig $r_i = r'_{i+1}$. Jelölése: $SG_1 \prec^1 SG_2$. A reláció tranzitív lezártja a finomítás reláció.

DSTree

Konstrukció

Bináris fával dolgozunk, egy csúcs a részfájába tartozó idősorok egy indexe.

DSTree

Konstrukció

Bináris fával dolgozunk, egy csúcs a részfájába tartozó idősorok egy indexe. Minden csúcsban tároljuk a következő információkat:

- A csúcs által meghatározott részfában található idősorok C számát.
- Az $SG = (r_1, \ldots, r_m)$ szegmentálását a csúcs által indexelt idősoroknak.
- A felső és alsó korlát számításához felhasználható aggregált $Z = (z_1, \ldots, z_m)$ információt, ahol $z_i = (\mu_i^{min}, \mu_i^{max}, \sigma_i^{min}, \sigma_i^{max})$.

DSTree

Konstrukció

Bináris fával dolgozunk, egy csúcs a részfájába tartozó idősorok egy indexe. Minden csúcsban tároljuk a következő információkat:

- A csúcs által meghatározott részfában található idősorok C számát.
- Az $SG = (r_1, \ldots, r_m)$ szegmentálását a csúcs által indexelt idősoroknak.
- A felső és alsó korlát számításához felhasználható aggregált $Z = (z_1, \ldots, z_m)$ információt, ahol $z_i = (\mu_i^{min}, \mu_i^{max}, \sigma_i^{min}, \sigma_i^{max})$.

DSTree

Konstrukció

Bináris fával dolgozunk, egy csúcs a részfájába tartozó idősorok egy indexe. Minden csúcsban tároljuk a következő információkat:

- A csúcs által meghatározott részfában található idősorok C számát.
- Az $SG = (r_1, \ldots, r_m)$ szegmentálását a csúcs által indexelt idősoroknak.
- A felső és alsó korlát számításához felhasználható aggregált $Z = (z_1, \ldots, z_m)$ információt, ahol $z_i = (\mu_i^{min}, \mu_i^{max}, \sigma_i^{min}, \sigma_i^{max})$.

DSTree

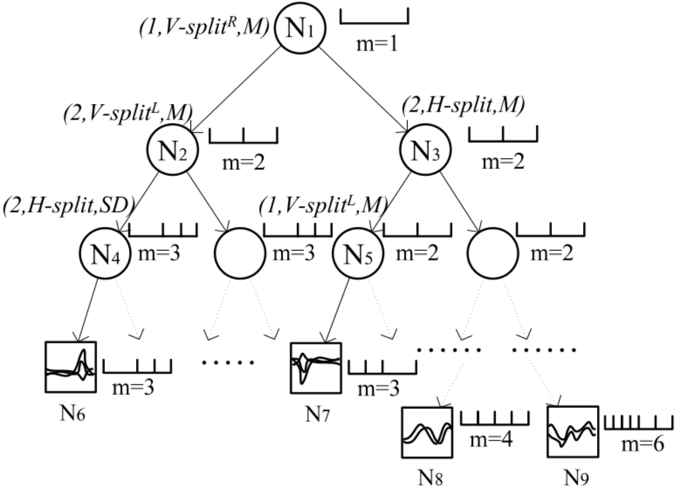
Konstrukció

Bináris fával dolgozunk, egy csúcs a részfájába tartozó idősorok egy indexe. Minden csúcsban tároljuk a következő információkat:

- A csúcs által meghatározott részfában található idősorok C számát.
- Az $SG = (r_1, \ldots, r_m)$ szegmentálását a csúcs által indexelt idősoroknak.
- A felső és alsó korlát számításához felhasználható aggregált $Z = (z_1, \ldots, z_m)$ információt, ahol $z_i = (\mu_i^{min}, \mu_i^{max}, \sigma_i^{min}, \sigma_i^{max})$.

A levelek tárolnak egy mutatót egy legfeljebb ψ (levélkapacitás) idősort tároló fájlra, a belső csúcsok tárolják a hasítási stratégiát.

Példa DSTree-re



Beszúrás DSTree-be

Algorithm 1 $N.Insert(X)$: N is a node, X is a time series

update Z in node N according to X;
if N is a leaf node **then**
    if $C < \psi$ **then** ▷ N has space to hold X
        Append X to data file pointed by N, $C = C + 1$;
    else ▷ $C == \psi$, no space in N to hold X
        Append X to data file pointed by N, $C = C + 1$;
        $SP = BestSplit()$;
        Create two children nodes for N;
        for each time series Y in N **do**
            $N' = N.routeToChild(Y, SP)$; $N'.insert(Y)$;
        end for
    end if
else
    $N' = N.routeToChild(X, SP)$; $N'.insert(X)$;
end if

Hasítási startégiák

Vízszintes hasítás (H-split)

Függőleges hasítás (V-split)

Hasítási startégiák

Vízszintes hasítás (H-split)

A szegmentálás nem változik, az utódok szegmentálása ugyanaz marad, csak kétfelé osztjuk őket.

Függőleges hasítás (V-split)

A szegmentálás finomodik, a szülő csúcs szegmentálásának ugyanazt az egylépéses finomítását fogja tartalmazni mindkét utód.

Műveletek

Hasonlósági keresés

Hisztogram készítés

Műveletek

Hasonlósági keresés

A heurisztikus hasonlósági keresés a beszúrás mintájára megnézi, hogy melyik levélbe kerülne a lekérdezés Q idősora, majd az ebben a levélben tárolt minden idősorra kiszámítja annak Q-tól vett távolságát, és visszaadja a legközelebbit.

Hisztogram készítés

A csúcsokban a Q-tól vett távolságra számított alsó és felső korlát, valamint a csúcs által indexelt idősorok száma alapján, azok összesítésével tudunk becslést adni adott távolság-intervallumon belül levő idősorok számára.

Tartalom

Tesztspecifikáció

Alternatív módszerek

- Piecewise Aggregate Approximation (PAA) & R-fa
- iSAX2.0

Tesztspecifikáció

Alternatív módszerek

- Piecewise Aggregate Approximation (PAA) & R-fa
- iSAX2.0

Tesztspecifikáció

Alternatív módszerek

- Piecewise Aggregate Approximation (PAA) & R-fa
- iSAX2.0

Tesztadatok

- Szintetikus
- Valódi

Tesztspecifikáció

Alternatív módszerek

- Piecewise Aggregate Approximation (PAA) & R-fa
- iSAX2.0

Tesztadatok

- Szintetikus
- Valódi

Tesztspecifikáció

Inputgenerálás

- Véletlen bolyongás, a $[-5,5]$-ből egyenletesen véletlen kezdőponttal, valamint a $[0,2]$-ből egyenletesen véletlen lépéshosszal.
- Normális eloszlással generálódnak az idősor pontjai. A középpont a $[-5,5]$-ből, szórása pedig a $[0,2]$-ből kerül egyenletesen véletlenül kiválasztásra.
- Az előző módszerrel generálunk legalább 3, legfeljebb 10 idősort, majd ezeket konkatenáljuk.
- Több szinuszfüggvény összekeveréséből mintavételezéssel. A függvények periódusa a $[2,10]$-ből, amplitúdója a $[2,10]$-ből, átlaga pedig a $[-5,5]$-ből egyenletesen véletlen.

Tesztspecifikáció

Inputgenerálás

- Véletlen bolyongás, a $[-5, 5]$-ből egyenletesen véletlen kezdőponttal, valamint a $[0, 2]$-ből egyenletesen véletlen lépéshosszal.
- Normális eloszlással generálódnak az idősor pontjai. A középpont a $[-5, 5]$-ből, szórása pedig a $[0, 2]$-ből kerül egyenletesen véletlenül kiválasztásra.
- Az előző módszerrel generálunk legalább 3, legfeljebb 10 idősort, majd ezeket konkatenáljuk.
- Több szinuszfüggvény összekeveréséből mintavételezéssel. A függvények periódusa a $[2, 10]$-ből, amplitúdója a $[2, 10]$-ből, átlaga pedig a $[-5, 5]$-ből egyenletesen véletlen.

Tesztspecifikáció

Inputgenerálás

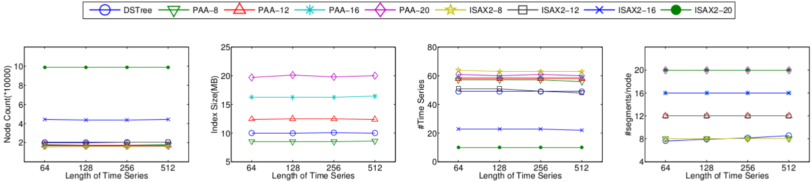
- Véletlen bolyongás, a $[-5, 5]$-ből egyenletesen véletlen kezdőponttal, valamint a $[0, 2]$-ből egyenletesen véletlen lépéshosszal.
- Normális eloszlással generálódnak az idősor pontjai. A középpont a $[-5, 5]$-ből, szórása pedig a $[0, 2]$-ből kerül egyenletesen véletlenül kiválasztásra.
- Az előző módszerrel generálunk legalább 3, legfeljebb 10 idősort, majd ezeket konkatenáljuk.
- Több szinuszfüggvény összekeveréséből mintavételezéssel. A függvények periódusa a $[2, 10]$-ből, amplitúdója a $[2, 10]$-ből, átlaga pedig a $[-5, 5]$-ből egyenletesen véletlen.

Tesztspecifikáció

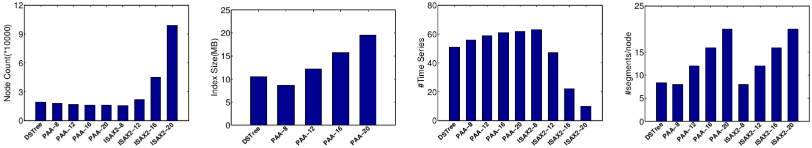
Inputgenerálás

- Véletlen bolyongás, a $[-5, 5]$-ből egyenletesen véletlen kezdőponttal, valamint a $[0, 2]$-ből egyenletesen véletlen lépéshosszal.
- Normális eloszlással generálódnak az idősor pontjai. A középpont a $[-5, 5]$-ből, szórása pedig a $[0, 2]$-ből kerül egyenletesen véletlenül kiválasztásra.
- Az előző módszerrel generálunk legalább 3, legfeljebb 10 idősort, majd ezeket konkatenáljuk.
- Több szinuszfüggvény összekeveréséből mintavételezéssel. A függvények periódusa a $[2, 10]$-ből, amplitúdója a $[2, 10]$-ből, átlaga pedig a $[-5, 5]$-ből egyenletesen véletlen.

Indexméret összehasonlítása a szintetikus (felül) és valódi (alul) adathalmazokon

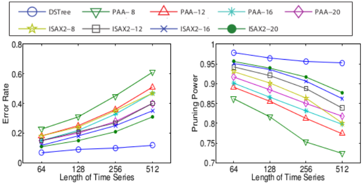


(a) Number of nodes (b) Index size (c) Average #ts per leaf (d) # segments/node



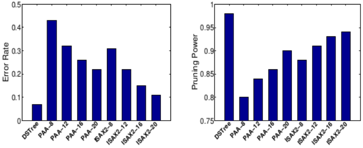
(a) Number of nodes (b) Index size (c) Average #ts per leaf (d) # segments/node

Keresés hatékonysága szintetikus (felül) és valódi (alul) adathalmazokon



(a) Heuristic search error rate

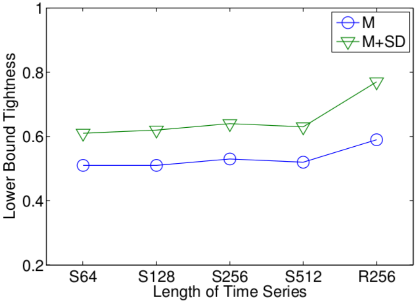
(b) Exact search pruning power



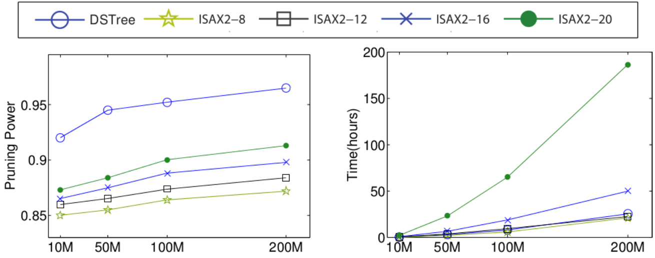
(a) Heuristic search error rate

(b) Exact search pruning power

APCA (M) és EAPCA (M+SD) alsó korlát értékek összehasonlítása

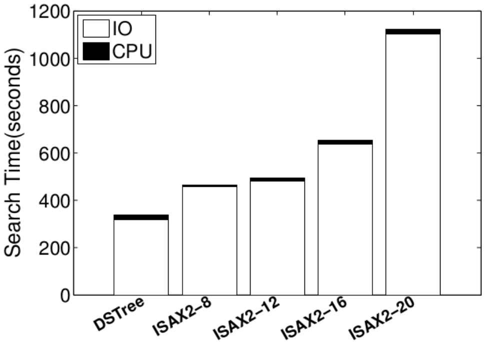


Skálázhatóság összehasonlítása



(a) Pruning power (b) Building time

Keresési idő összehasonlítása



Implementáció

DSTree implementáció

A cikk alapján megvalósítottuk az adatszerkezetet és műveleteit.

- Feltöltünk egy DSTree-t idősorokkal
- Keresünk közöttük egy új idősorhoz hasonlót

Implementáció

DSTree implementáció

A cikk alapján megvalósítottuk az adatszerkezetet és műveleteit. Illusztráljuk ezt a következő példán:

- Feltöltünk egy DSTree-t idősorokkal
- Keresünk közöttük egy új idősorhoz hasonlót

Implementáció

DSTree implementáció

A cikk alapján megvalósítottuk az adatszerkezetet és műveleteit. Illusztráljuk ezt a következő példán:

- Feltöltünk egy DSTree-t idősorokkal
- Keresünk közöttük egy új idősorhoz hasonlót

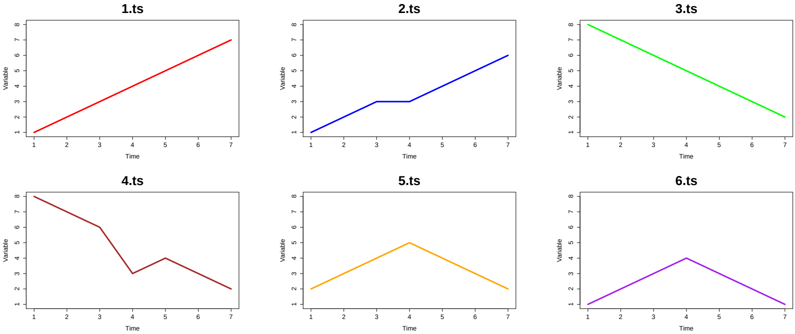
Implementáció

DSTree implementáció

A cikk alapján megvalósítottuk az adatszerkezetet és műveleteit. Illusztráljuk ezt a következő példán:

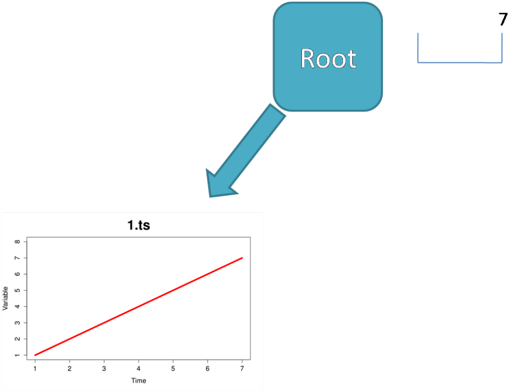
- Feltöltünk egy DSTree-t idősorokkal
- Keresünk közöttük egy új idősorhoz hasonlót

Idősorok

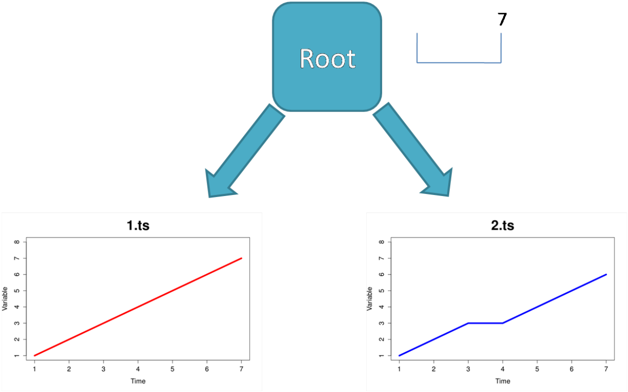


DSTree felépítése

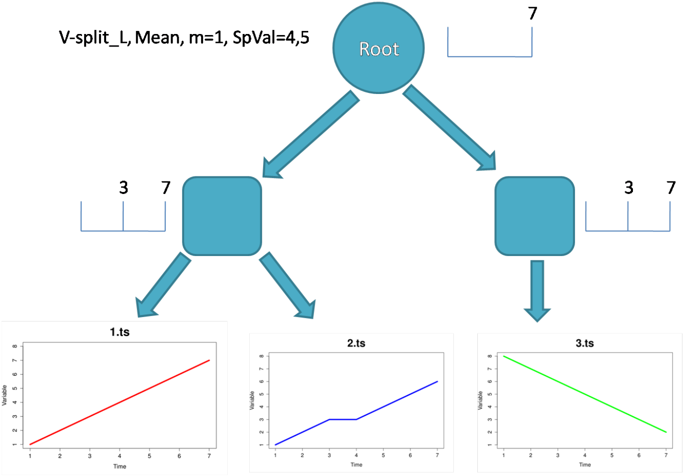
DSTree felépítése



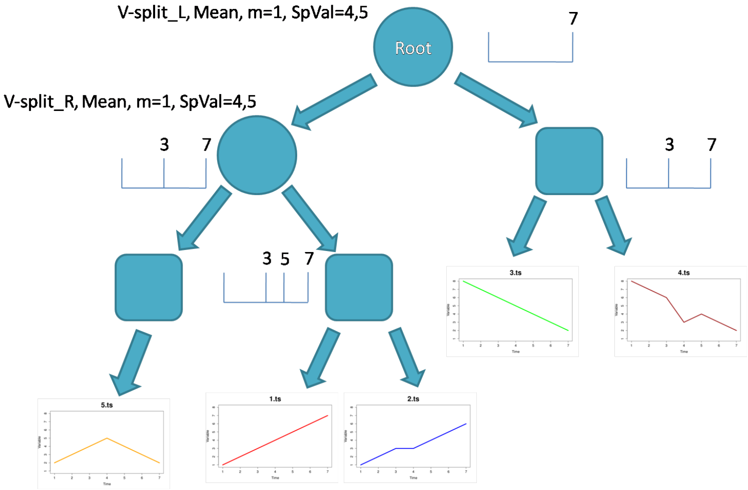
DSTree felépítése



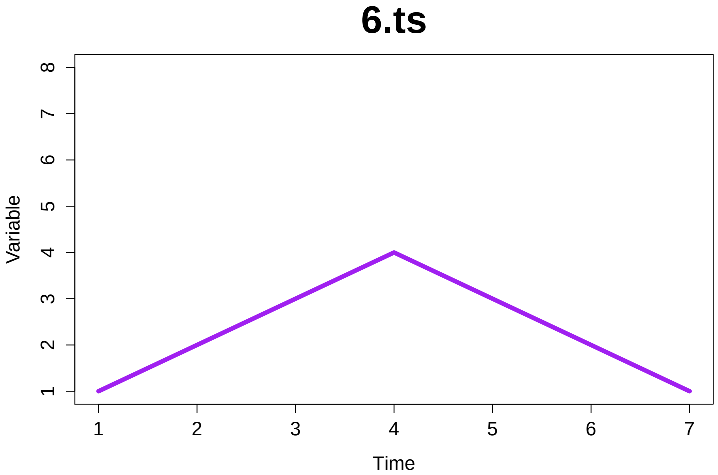
DSTree felépítése



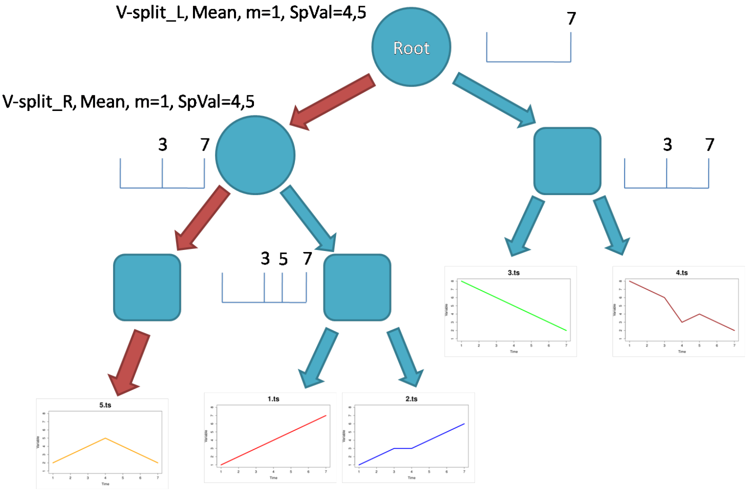
DSTree felépítése



Heurisztikus keresés



Heurisztikus keresés



Tartalom

Összefoglalás

Útravaló

- Adaptív, dinamikus szegmentálással jelentősen javítani lehet a keresési időn
- Érdemes lehet áttérni EAPCA reprezentációra
- DSTree-t fontos tovább vizsgálni
- Standard tesztadatok hiánya

Összefoglalás

Útravaló

- Adaptív, dinamikus szegmentálással jelentősen javítani lehet a keresési időn
- Érdemes lehet áttérni EAPCA reprezentációra
- DSTree-t fontos tovább vizsgálni
- Standard tesztadatok hiánya

Összefoglalás

Útravaló

- Adaptív, dinamikus szegmentálással jelentősen javítani lehet a keresési időn
- Érdemes lehet áttérni EAPCA reprezentációra
- DSTree-t fontos tovább vizsgálni
- Standard tesztadatok hiánya

Összefoglalás

Útravaló

- Adaptív, dinamikus szegmentálással jelentősen javítani lehet a keresési időn
- Érdemes lehet áttérni EAPCA reprezentációra
- DSTree-t fontos tovább vizsgálni
- Standard tesztadatok hiánya

Köszönetnyilvánítás

Köszönjük a figyelmet!