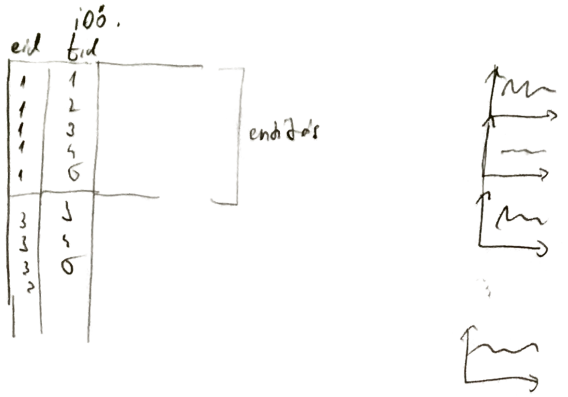
2018.04.17.

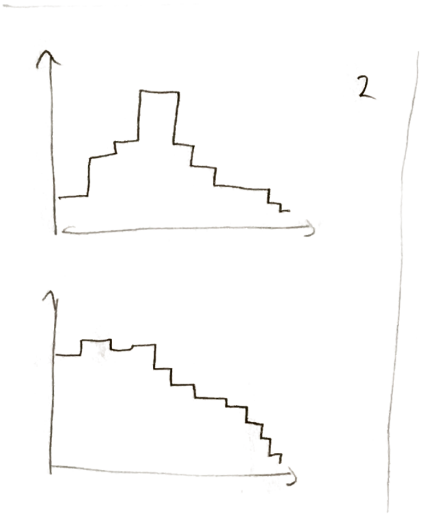
Idősoros adatok

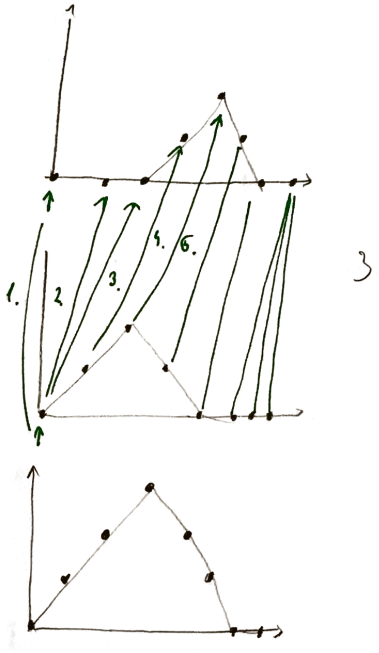
- eddig: UID és adatok
- most: timeID és adatok (szükséges de nem elégséges)
- Δt legyen állandó (általában)
- probléma lehet
  - alul-mintavételezés (túl ritkán vannak)
  - túl-mintavételezés (túl sok egyforma adat egymás után pl.)

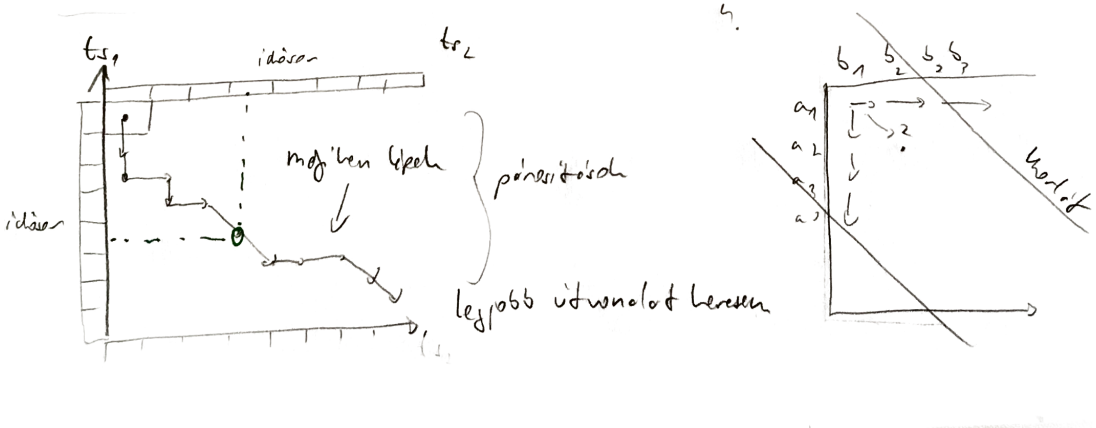
Több idősoron való munka

- 

- egy entitáshoz több adatsor (többféle timestamp-pel)
- nem biztos hogy ugyanolyan idő
- *osztályozás*
  - pl. EKG $\rightarrow$ egy-egy jelsor mennyire jelent veszélyes állapotot a betegre
  - pl. olajfinomítóban nagy tartály - néha berezonál, megvannak az idősoros adatok - meg kell jósolni, mikor fog berezonálni
- *regresszió*
  - pl. focisták lábának mozgását mérik - meg kell becsülni hogy mennyi idő alatt fut le egy távot
    sportanalitika
- *klaszterezés*
  - pl. hasonló időjárás viszonyok keresése
- *anomália-detekció*
  - pl. online pénztárgép forgalom - furcsa mintázat keresése
- megoldás:
  - egy entitáshoz tartozó érétkeket egy targetváltozóra felhasználni NEM JÓ
  - machine learning nem szokott foglalkozni az adatok sorrendjével!
  - **a.** egy adatsorhoz hozzárendelünk jellemző értékeket (átlag, szórás, egyéb statisztikák) (*feature engineering*)
    - ezt csak akkor szokás használni, ha k legközelebbi szomszéd-dal akarunk osztályozni
    - feladat: hogy keresek távolságot
  - *idősorok közötti távolság*
    - két ügyfél bankszámláján lévő összegek
    -



- euklédeszi távolság (legegyszerűbb lehetőség)
- minden érétket négyzetre emelünk vagy vesszük logaritmusát, vagy átlagra normalizáljuk
- ha a mintázat alakja a fontos, de időben lehet hogy el vannak tolva
  - 

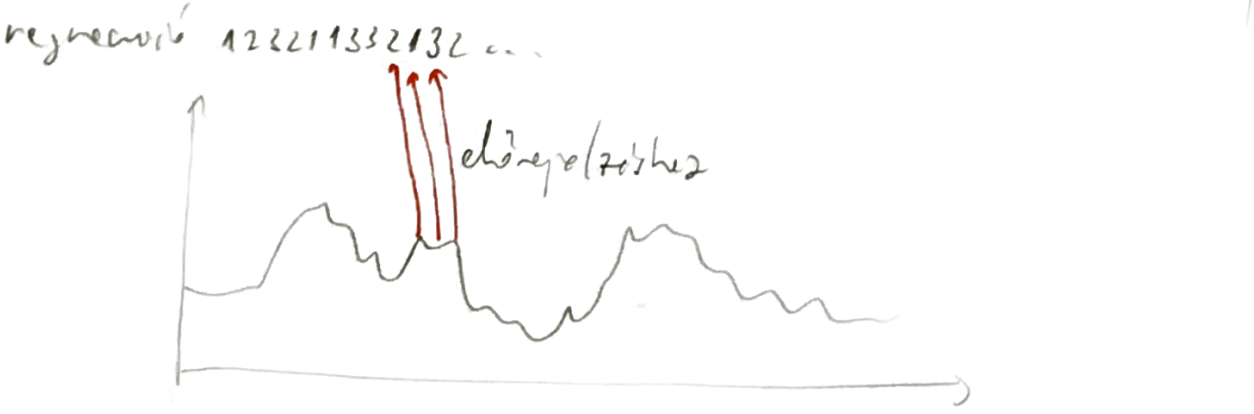
  - *DTW* (Dynamic Time Warping) - *idővetemítés*
    - párosítást keresünk két idősor között
    - azonos (hasonló) értékű párt keresünk mindegyikhez
    - nem keresztezhetik egymást a vonalak! (vagy alul lépünk, vagy föül, vagy mindkettő helyen)
    - páronként a különbség minimális legyen
    - algoritmusa:
      - 

      - legjobb útvonalat keresem a mátrixban

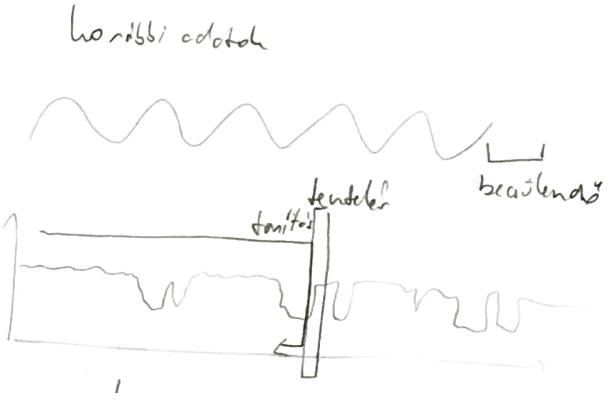
        - nem kell az összes lehetségeset végigvizsgálni (csak jobbra és lefelé haladhatok, és az almátrix optimuma nem függ az előtte lévőktől)
          $ts_1 \mid a_1 \mid a_2 \mid a_3 \mid ...$
          $ts_1 \mid b_1 \mid b_2 \mid b_3 \mid ...$
          $m_{1,1} = d(a_1, b_1)$
          $m_{i,1} = m_{i-1,1} + d(a_i, b_1)$
          $m_{1,j} = m_{1,j-1} + d(a_1, b_j)$
          $m_{i,j} = min \begin{Bmatrix} m_{i-1,j-1} + d(a_i, b_j) \\ m_{i,j-1} + d(a_{i-1}, b_j) \\ m_{i-1,j} + d(a_i, b_{j-1}) \end{Bmatrix}$
        - magyarul: vagy föntről érkezem, vagy balról, vagy átló irányból. Azt választom amelyiknek a legkisebb a kumulatív összege
        - lehet az átlótól való távolságot korlátozni
    - akkor jó, ha kevés adatsor van! - 3-400 adatsor fölött érdemes modellel számolni

Egyetlen idősoron való munka

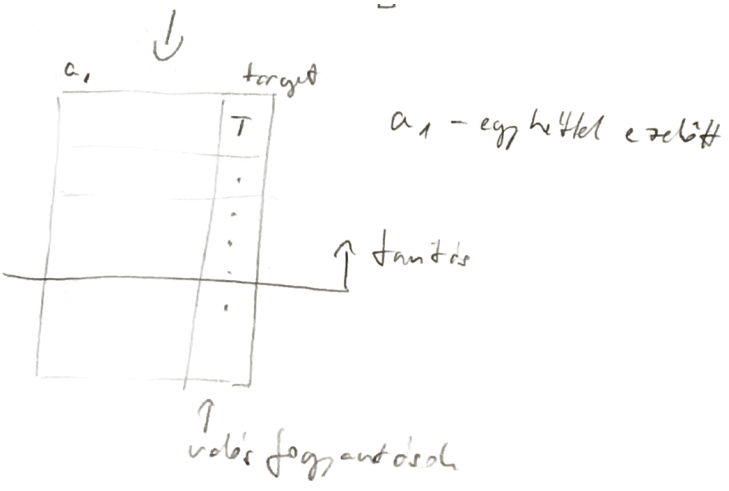
- osztályozás
  - 

  - egy adatsoron időpont csoportokat keresünk, ezt egy osztályba soroljuk
    - idősor szegmentáció
  - pl. telefon gyorsulásérzékelőjének adatai - mit csinálok épp
- klaszterezés
- anomália-detekció
- regresszió
  - 

  - minden időponthoz egy értéket rendelünk
  - előrejelzés
    - minden pillanathoz a következő pillanat értékét rendeljük
    - **Energetikai előrejelzés:**
      - áram kereskedelem - minden következő napra meg kell mondani, hogy a következő nap 15 perces bontásban mennyit fognak fogyasztani (menetrend)
      - áram ára: https://www.hupx.hu/hu/Lapok/hupx.aspx?remsession=1
      -



- Data Understanding:
  - adatok
  - naptáradat (nem mindegy, hogy hétvége vagy ünnepnap stb.)
- input változók - csak 24 óra vagy azelőtti infót tehetünk be paraméterként
  - előző heti azonos nap
  - hőmérséklet
  - stb.
- modell tanítása:
  - (nem szabad véletlen mintát venni!)
  - egy időpont előtti adatok lesznek a tanítók
  - tesztelés csak a következő napon!
  - utána újra tanítok a plusz egy adattal, majd úgy tesztelem a következőt
  - → minden napra egy új modell → modell építési *stratégia* (nem a modellt értékelem, hanem a stratégiát)
  -



- *time window validation* (egy adott időtartam után már a régi adatokat kidobom - mekkora legyen az időtartam?)
- *ARIMA* modell - statisztikában használt - itt nem szokott jó lenni
- *hasznos:* Idősor elemzés előtt három lépés:
  - Trend kiszedése
  - Szezonalitás kiszedése
  - Periodicitás kiszedése