ALGORITMI ANALIZE MASIVNIH PODATKOV

DOMEN MONGUS

P01 - Uvod

Vsebina

- definicija masivnih podatkov,
- ključni podatkovni viri,
- analiza trendov razvoja in
- aktualni izzivi na področju


SAY BIG DATA
ONE MORE TIME
memegenerator.net

Definicija

- Tradicionalno 3xV
  - **Volume** (Velikost)
  - **Velocity** (Hitrost)
  - **Variety** (Raznorodnost)

- **Vedno premikajoča tarča!**
  - Masivnih podatkov ni mogoče učinkovito obdelati s tradicionalnimi metodami obdelave podatkov

Definicija

- Definicija s 5xV
  - Volume (Velikost)
  - Velocity (Hitrost)
  - Variety (Raznorodnost)
  - **Veracity** (Verodostojnost)
  - **Value** (Vrednost)

- **Podpora pri odločanju!**
  - Analize in napovedi so fokus obdelave podatkov!

Aplikacije


Poslovna analitika
Socialna Omrežja
...
Masivni podatki
Medicina
Oblikovanje politik
Zavarovalništvo

Tudi mi smo senzorji …

- Kje se srečamo s tehnologijami masivnih podatkov?
  - Ste kdaj prejeli ciljno reklamno sporočilo?
    - Analize obnašanja na spletu
  - Uporabljate kartice popusta?
    - Nakupovalne navade
  - Ste kdaj plačali s kreditno kartico?
    - Zaznava goljufivih bančnih transakcij
  - Ste prejeli popust pri zavarovali avto?
    - Bonusni programi zavarovalnic (primer sledenja)
  - Uporabljate elektriko?
    - Napovedovanje porabe električne energije

Osnovni izrazoslovje

- **Ciljna spremenljivka (ang. target variable)** je spremenljivka, ki jo napovedujemo
- **Razlagalna spremenljivka (ang. explanatory variable)** je vrednost, ki jo uporabljamo za napovedovanje
- **Značilnica (ang. feature)**, je značilna lastnost ali značilen vidik nečesa
- **Izdvajanje značilnic (ang. featuer extraction)** je postopek strukturiranja značilnic za nadaljno rabo
- **Podatkovne transformacije (ang. data transformation)** je preslikava podatkov iz ene oblike/strukture v drugo

Osnovni izrazoslovje

- **Čiščenje podatkov (ang. data cleaning)** je proces zaznave in odstranjevanja manjkajočih ali pokvarjenih podatkov
- **Izbira podatkov (ang. data selection)** je postopek izbiranja ustreznih podatkov za naše analize
- **Agregacija podatkov (ang. data aggregation)**, je združevanje podatkov v skupne strukture
- **Podatkovno zlivanje (ang. data fusion)** je postopek združevanja komplementarnih podatkov
- **Podatkovno bogatenje (ang. data enrichment)** pomeni izboljševanje informacijske vrednosti podatkov skozi njihovo združevanje

Tradicionalen model procesa podatkovne analitike

Izbira podatkov

Čiščenje podatkov

Podatkovne transformacije

Analiza

Vizualizacija
Interpretacija

Masivni podatki spreminjajo svet!

Nekoč

- Zastavi vprašanje
- Izvedi eksperiment
- Zberi podatke
- Analiziraj podatke
- Odgovori na vprašanje

Danes

- Zbiraj podatke
- Zastavi vprašanje
- Strukturiraj podatke
- Analiziraj podatke
- Odgovori na vprašanje

Podatkovni viri

Masivni podatki so nestrukturirani ali delno strukturirani

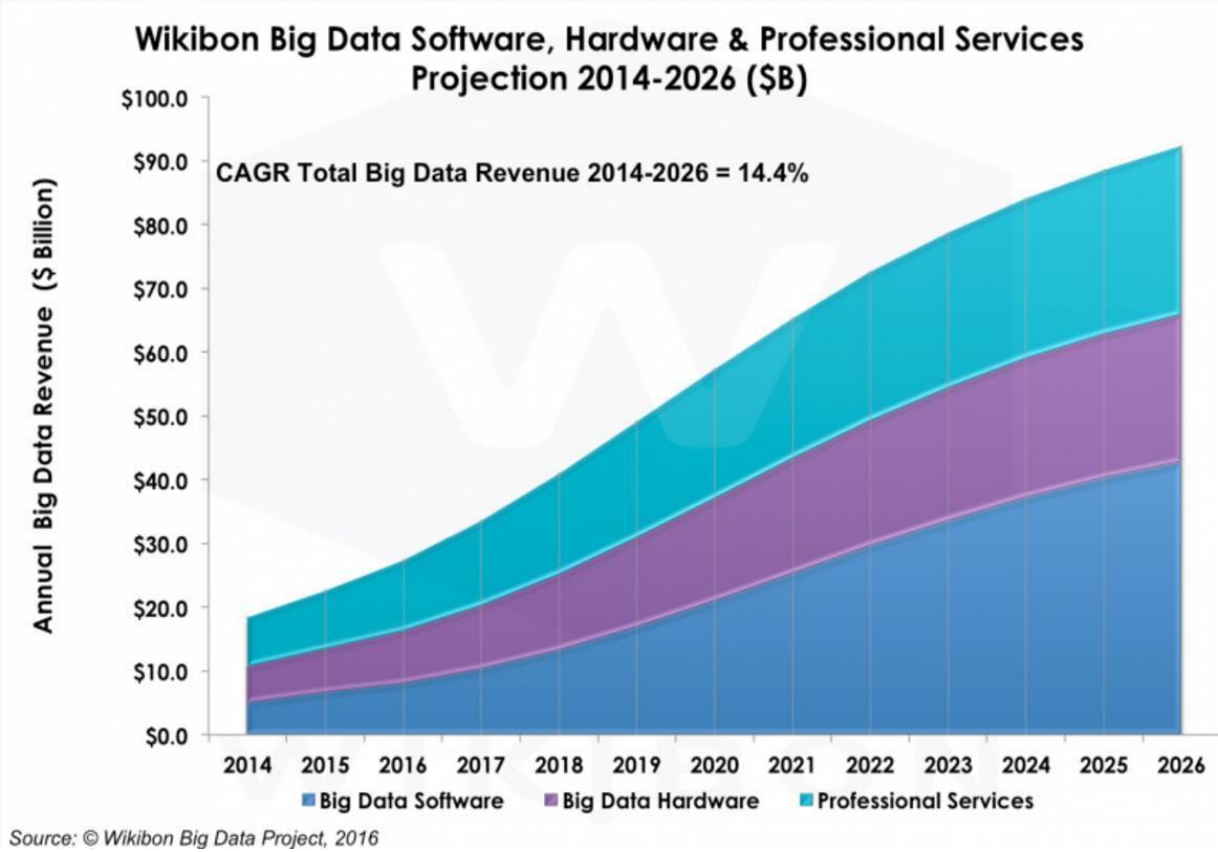
Masivni podatki vsebujejo vrednost

Masivni podatki niso enostavno razumljivi

- Senzorski podatki
- Logi naprav
- Spletne strani
- EMaili
- Socialna omrežja
- Mediji
- Tradicionalni dokumenti
- …

Trenutni trendi

- Ključni uporabniki
  1. Finančni sektor
  2. Proizvodnja
  3. Prodaja
  4. Mediji/zabava
  5. Igralništvo
  6. Zdravstvo
  7. Telekomunikacije
  8. Vladne organizacije



ALGORITMI ANALIZE MASIVNIH PODATKOV

DOMEN MONGUS

P03 – Prilagoditve pomnilniški hierarhiji

Vsebina – V4Volume

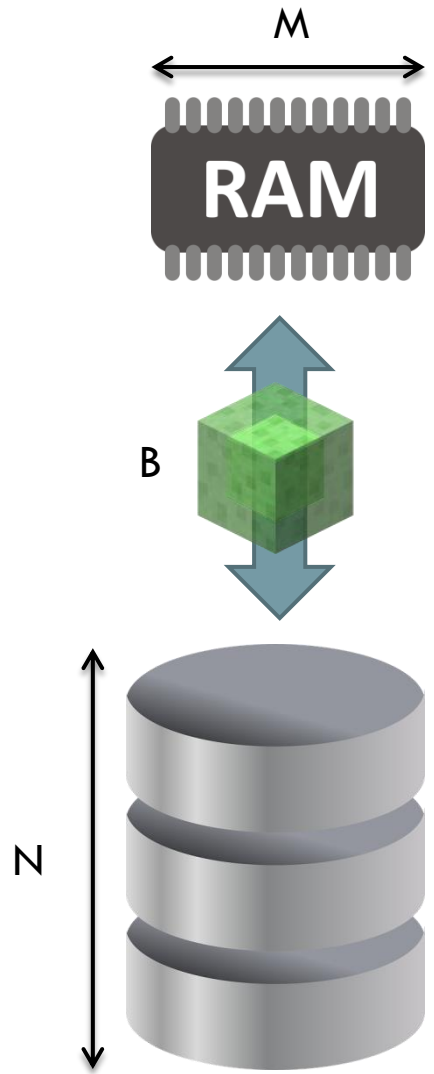
- DAM model (ang. Disk Access Model)
- Predpomnilniško-zavedni algoritmi

Ključne predpostavke:

- Podatki so preveliki za RAM
  - Podatkovne strukture so prevelike za RAM
- Operacije nad podatki so zelo enostavne
  - Časovna zahtevnost odvisna zgolj od števila dostopov do diska

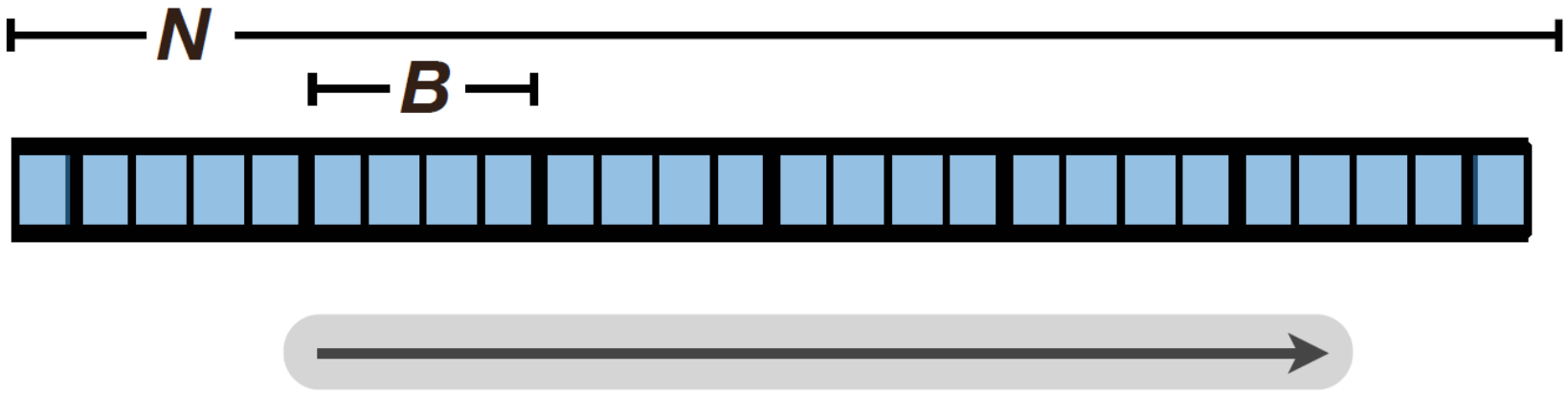
Model dostopov do diska (DAM)

- Velika količina podatkov
  - Podatki se prenašajo v blokih
  - V pomnilnik lahko shranimo nekaj blokov
  - Velikost diska je „neomejena“
- Cilj: Minimizacija prenosa podatkov
  - Parametri:
    - B = velikost bloka
    - M = velikost pomnilnika RAM
    - N = velikost podatkov



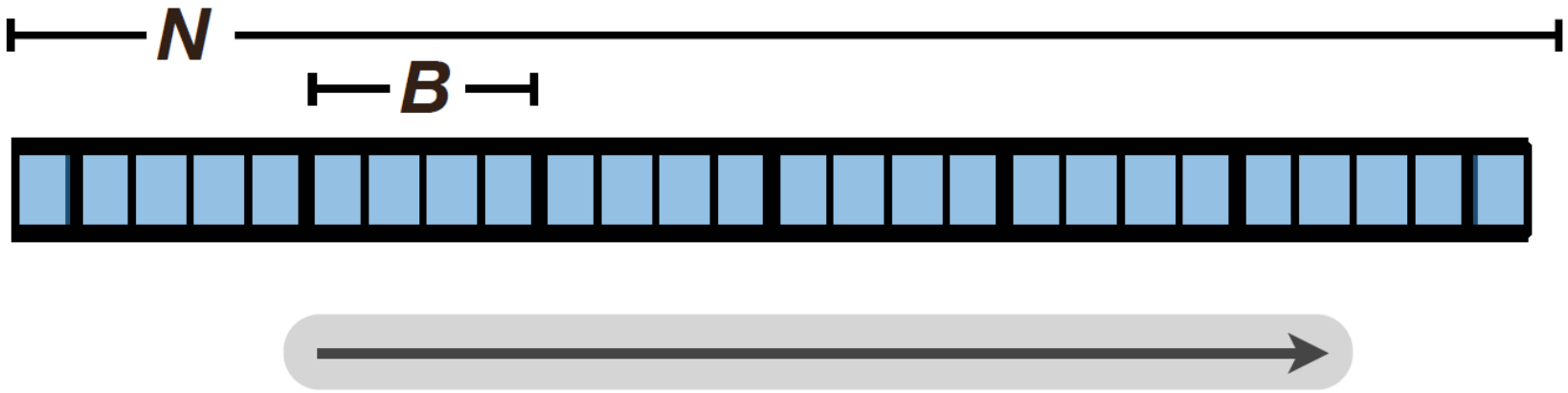
Prebiranje vrstice

- Koliko IO operacij je potrebno za branje?



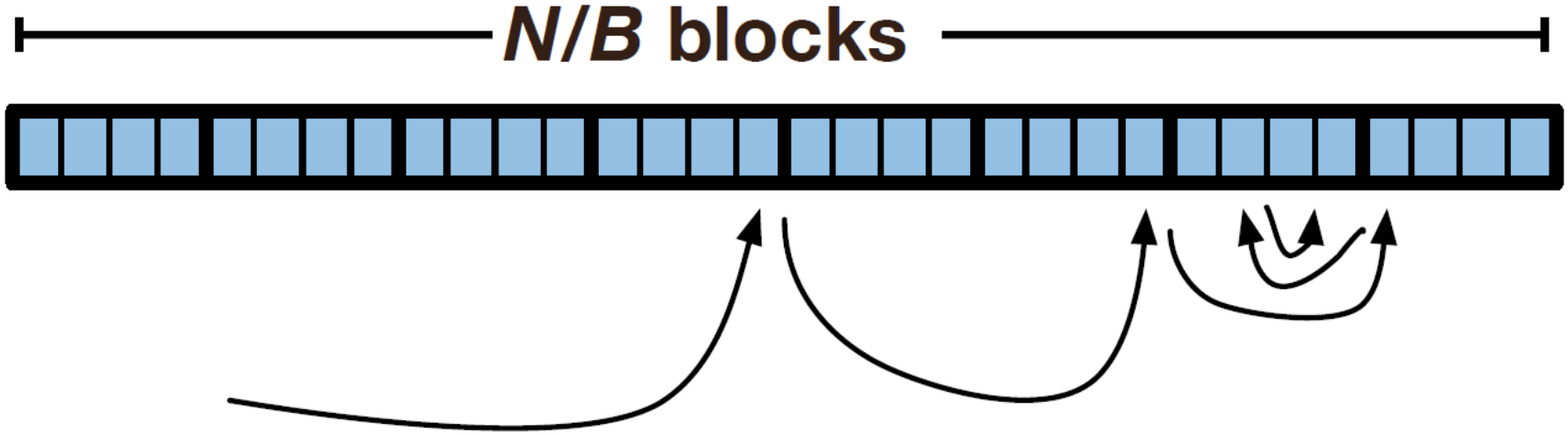
Prebiranje vrstice

- Koliko IO operacij je potrebno za branje?
  - O(N/B)



Koliko IO operacij je potrebno za iskanje elementa?

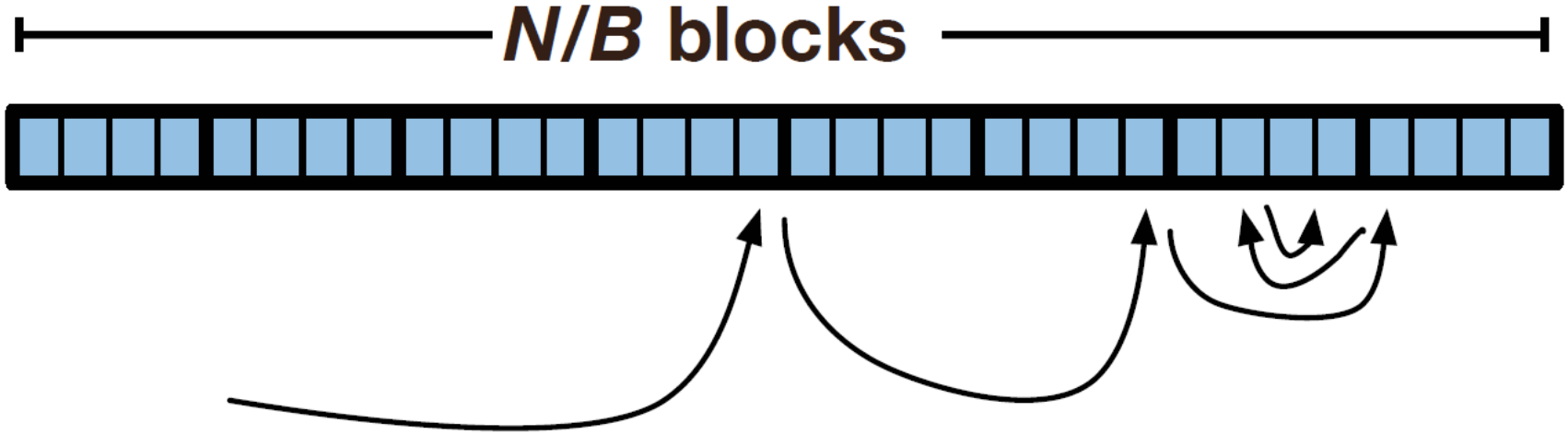
- Najslabši primer?



Koliko IO operacij je potrebno za iskanje elementa?

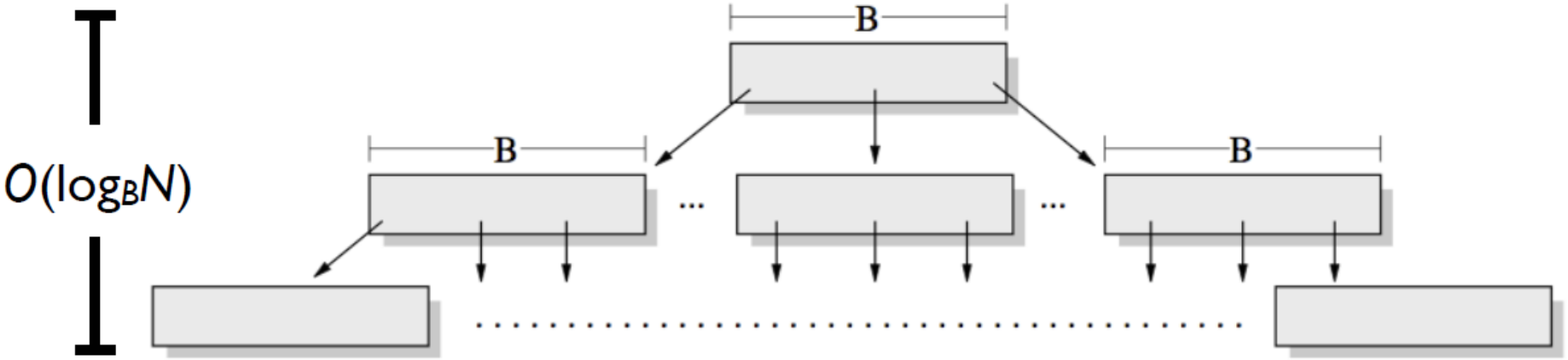
- Najslabši primer?

$$O\left(\log_2 \frac{N}{B}\right) \approx O(\log_2 N)$$



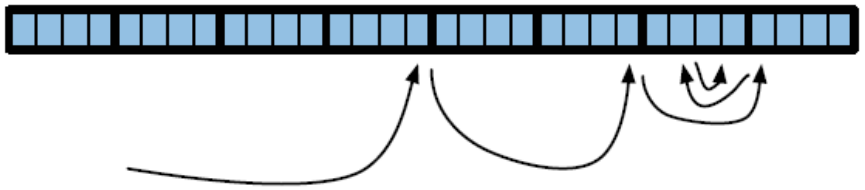
Koliko IO operacij je potrebno za iskanje elementa?

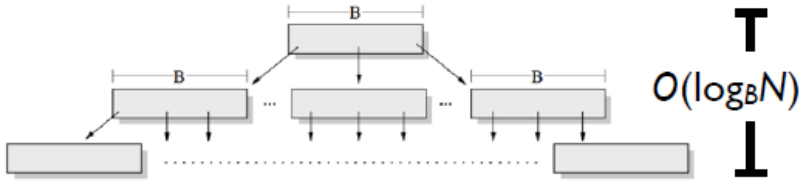
- PRIMER: Binarno drevo



Koliko IO operacij je potrebno za iskanje elementa?

- **Nauk zgodbe:** podatkovna struktura je ključ do učinkovite implementacije algoritma.

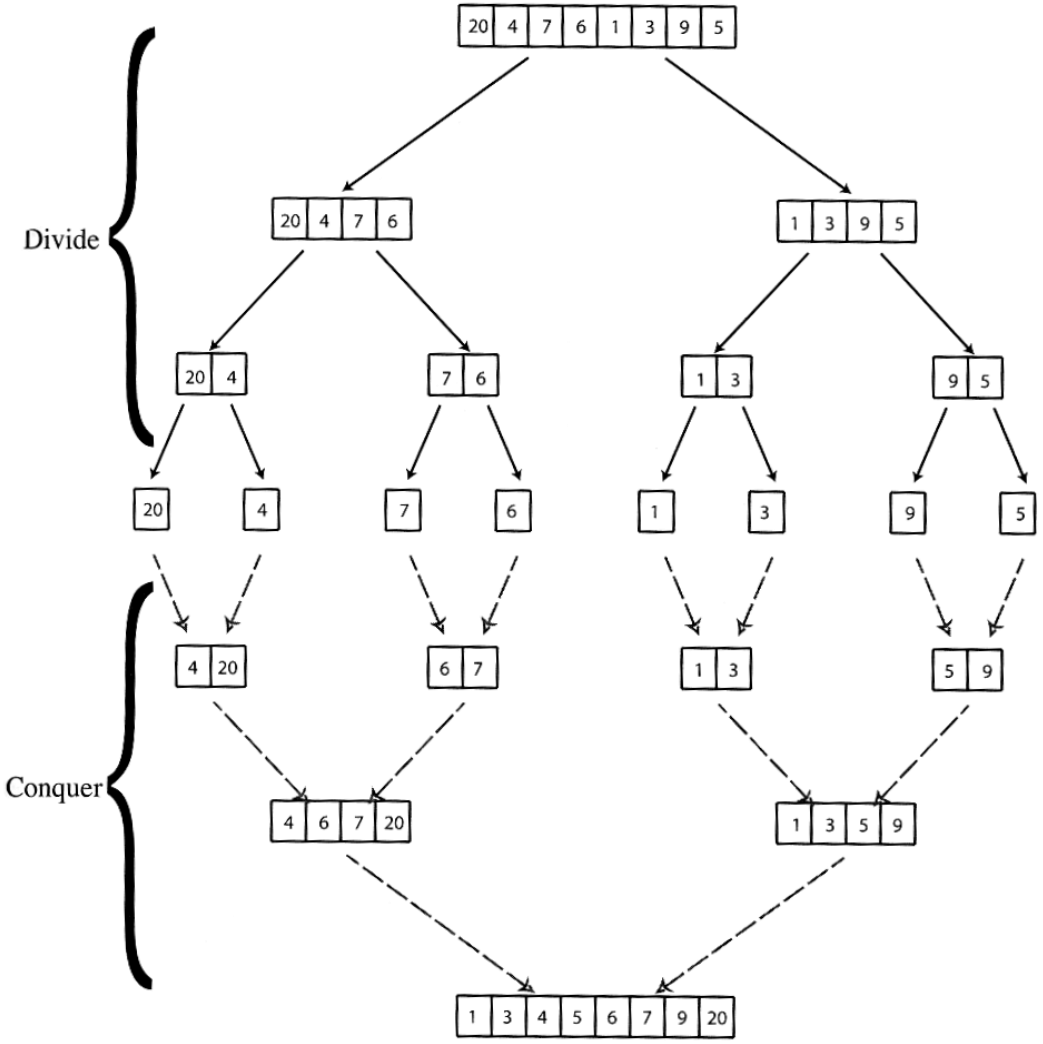




$$O(\log_2 N)$$

$$O(\log_B N) = O\left(\frac{\log_2 N}{\log_2 B}\right)$$

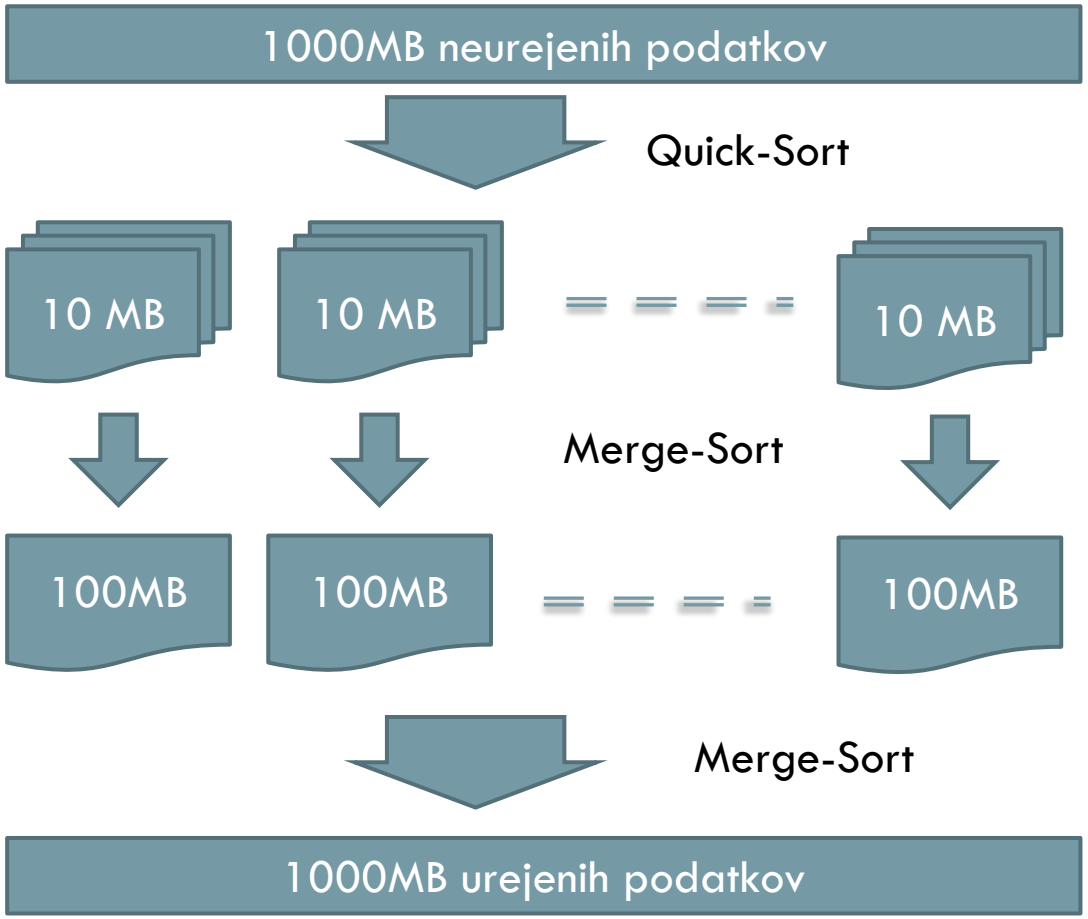
PONOVITEV: Urejanje z zlivanjem



Učinkovito urejanje velikih podatkov

- N = 1000 MB
- M = 10 MB
- B = 1MB

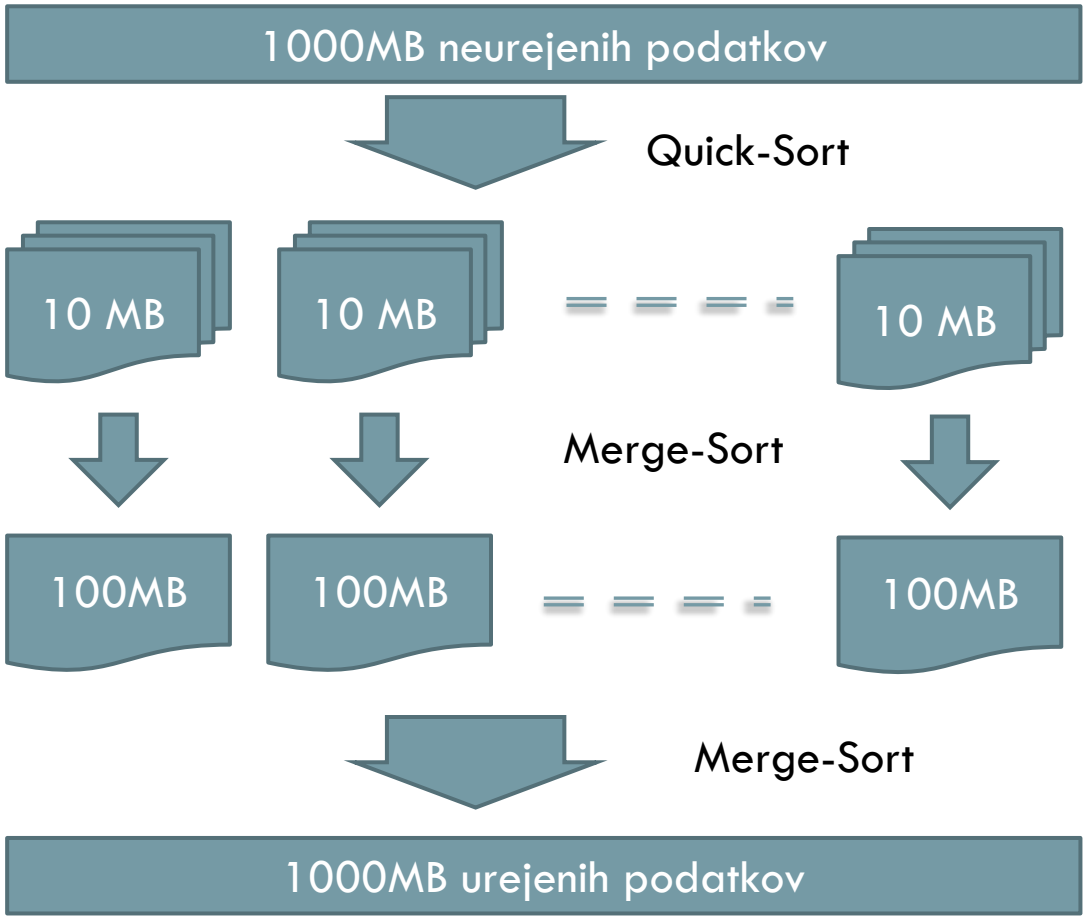
- Zakaj merge v dveh korakih?



Učinkovito urejanje velikih podatkov

- N = 1000 MB
- M = 10 MB
- B = 1MB

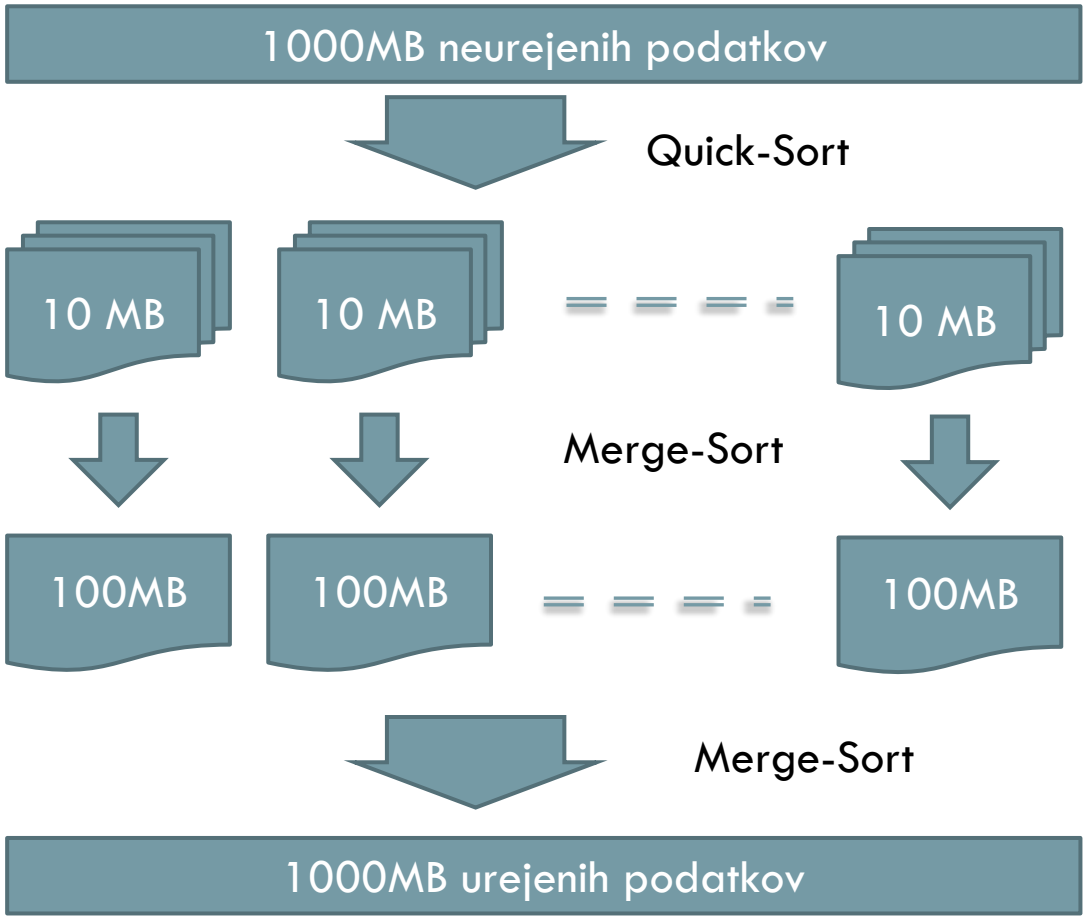
- Zakaj merge v dveh korakih?
  - Vedno beremo 1 blok!



Učinkovito urejanje velikih podatkov

- N = 1000 MB
- M = 10 MB
- B = 1MB

- Koliko IO operacij?



Učinkovito urejanje velikih podatkov

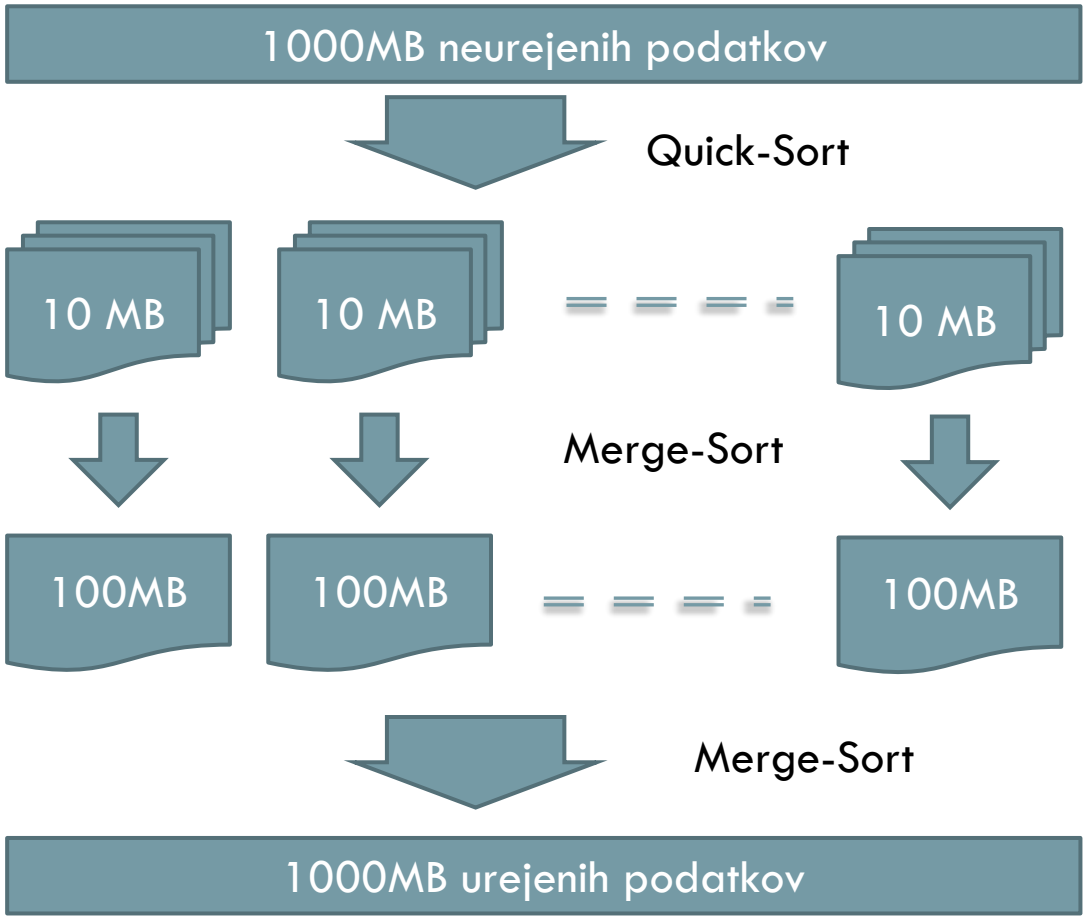
- N = 1000 MB
- M = 10 MB
- B = 1MB

- Koliko IO operacij?

$$O\left(\frac{N}{B}\log_{M/B}\frac{N}{B}\right)$$

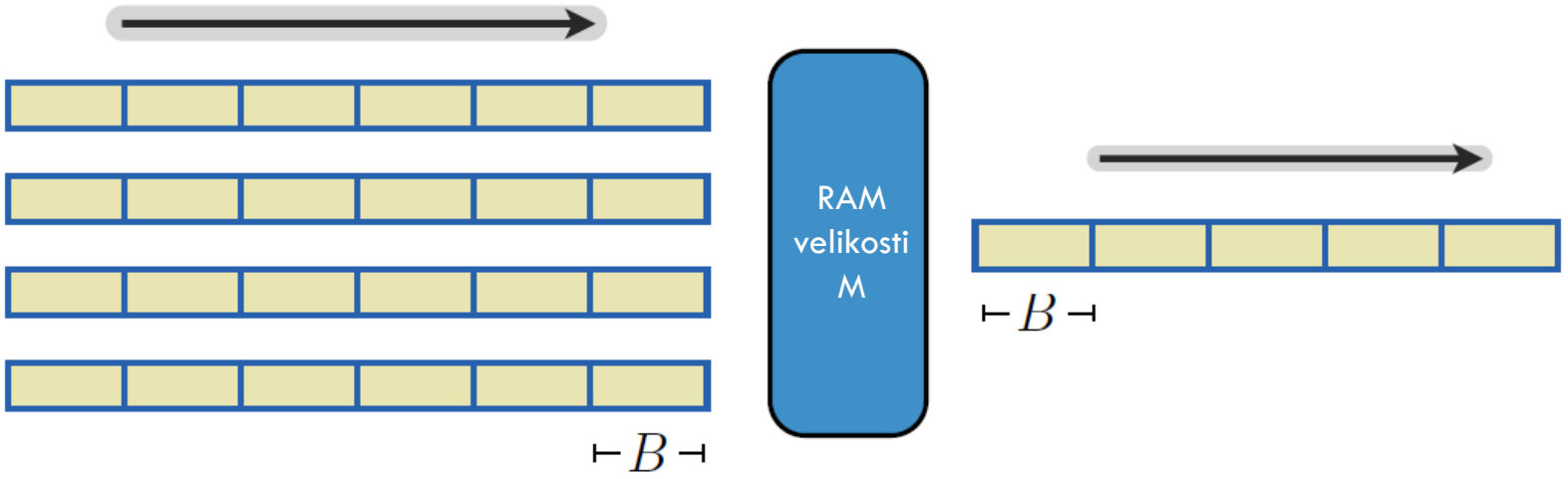
Cena prehoda

Število prehodov



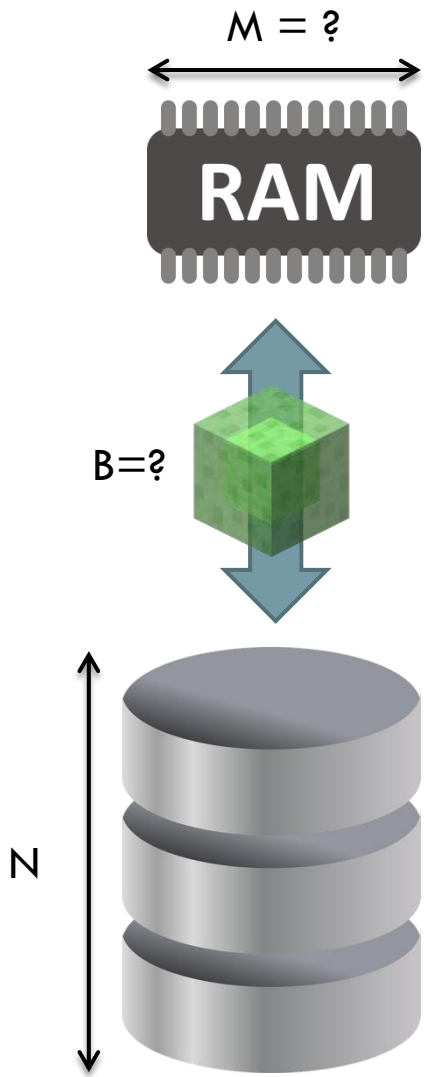
Učinkovito urejanje velikih podatkov

1. Uredi bloke velikosti M
2. Ustvari M/B tokov
3. Zlij tokove



Predpomnilniško nezavedni algoritmi

- Algoritem v naprej ne pozna parametrov B in M
- Cilj enak kot prej:
  - minimizirati število prenosov
- Optimizacija za vse možne M in B
  - Seveda pa ne najhitreje pri vseh M in B
  - Konsistentnost
- Je naš algoritem sortiranja predpomnilnoško zaveden?



Predpomnilniško nezavedni algoritmi

- Ang: Cache-oblivious algorithm
- Def: Algoritmi, ki izkoriščajo pomnilniško hierarhijo brez eksplicitnega znanja o velikosti pomnilnika.

- PRIMER: Transponirana matrika:
  - A velikosti n×m in B velikosti m×n
  - Kako izvedemo operacijo:

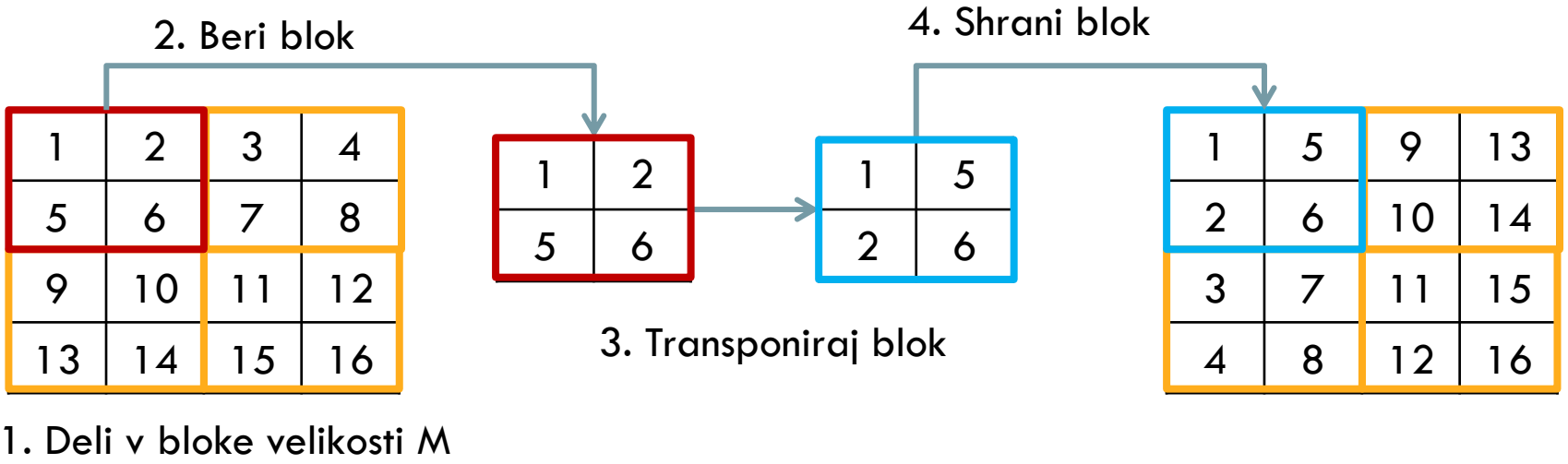
    $B = A^T$?
  - It's all about divide and conquer!

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

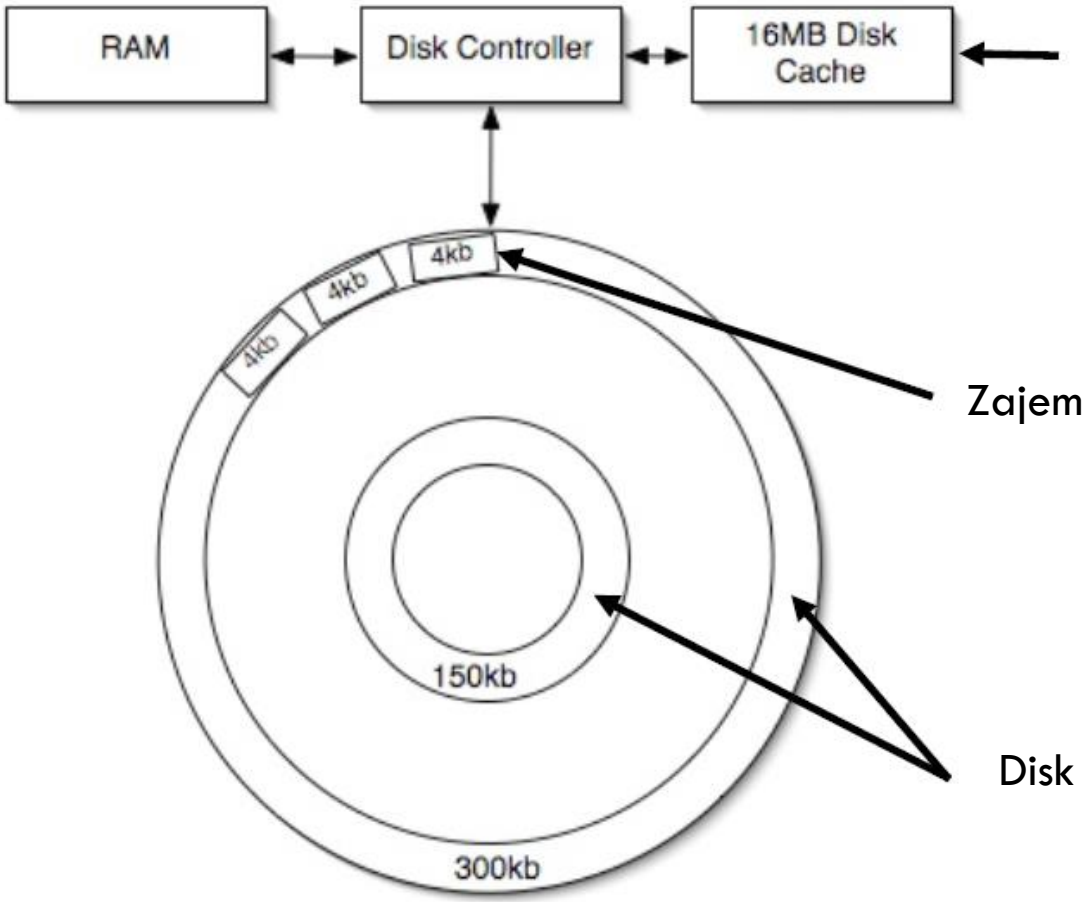
Predpomnilniško nezavedni algoritmi

- Učinkovit pomnilniško zaveden pristop:



- Če to idejo implementiramo rekurzivno, bomo vedno prišli do bloka velikosti M
  - Vse nadaljnje rekurzije ne zahtevajo več branja!

Resnično življenje



Disk zajame sledi v predpomnilnik - okoli 100 sledi

Zajem bloka konstantne velikosti

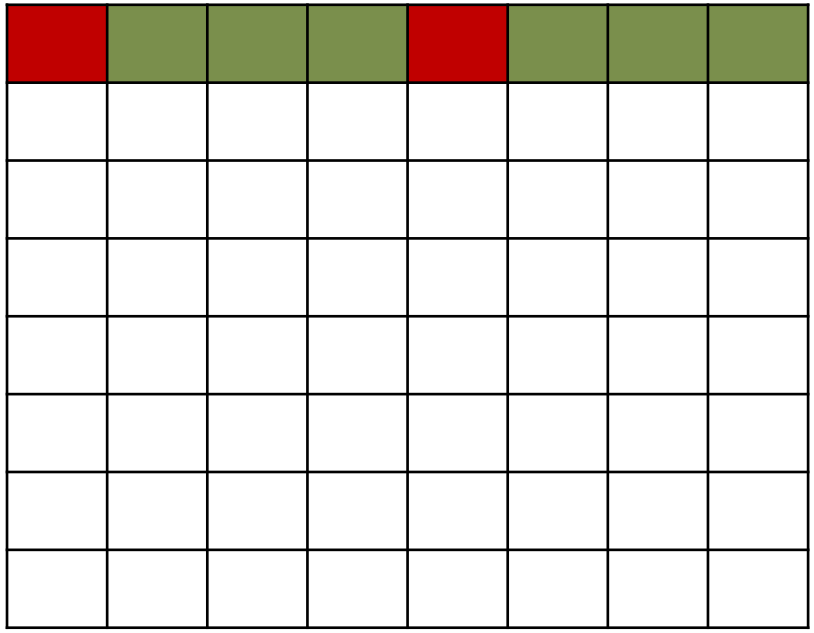
Disk je organiziran v sledi

Branje zaporednih blokov je cca. 10x hitrejše!

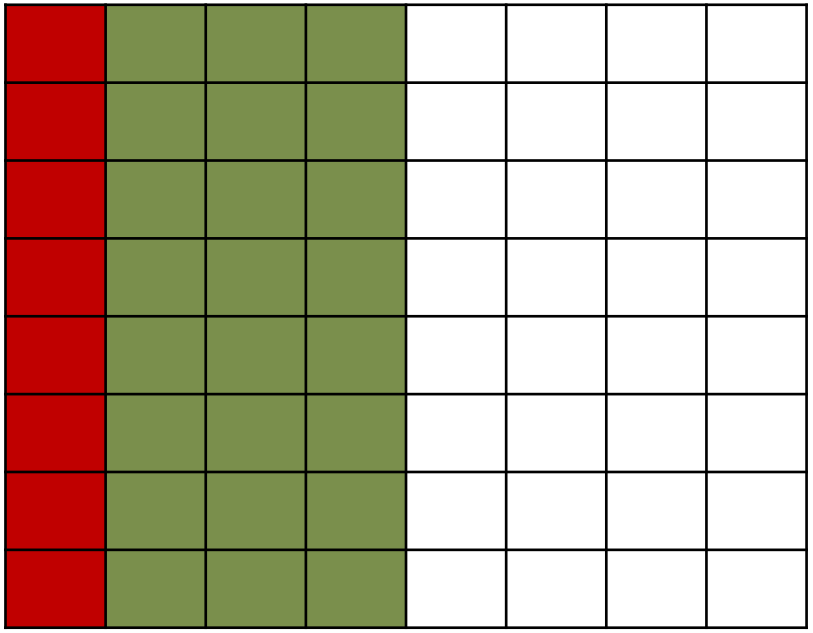
Resnično življenje

Množenje matrik

A

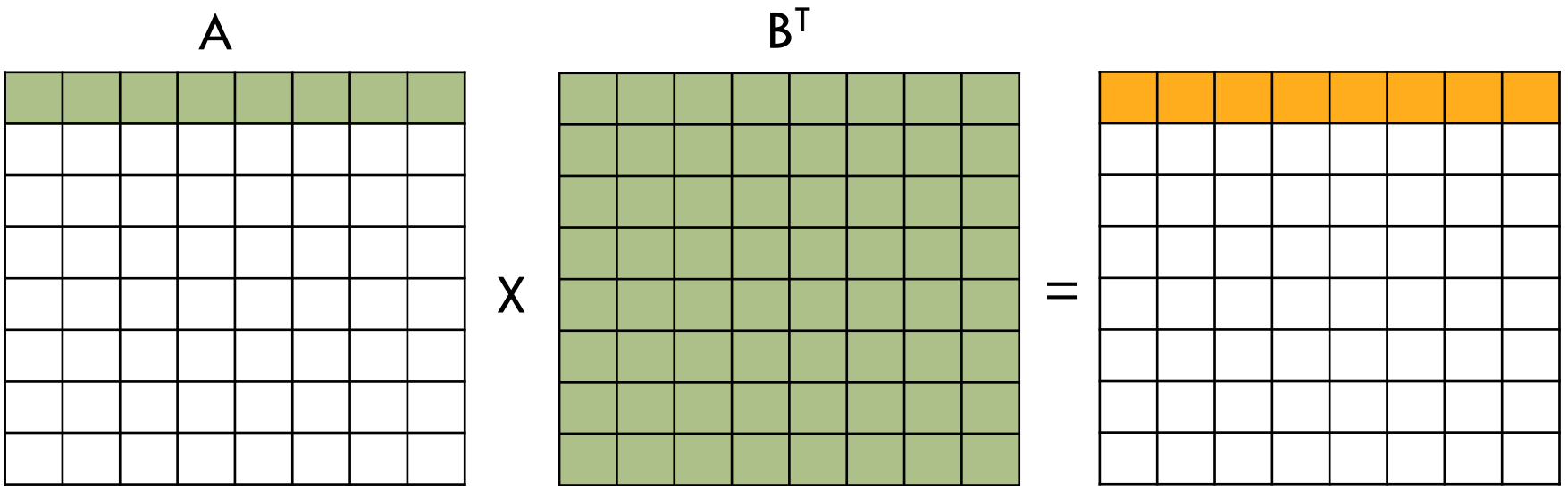


B



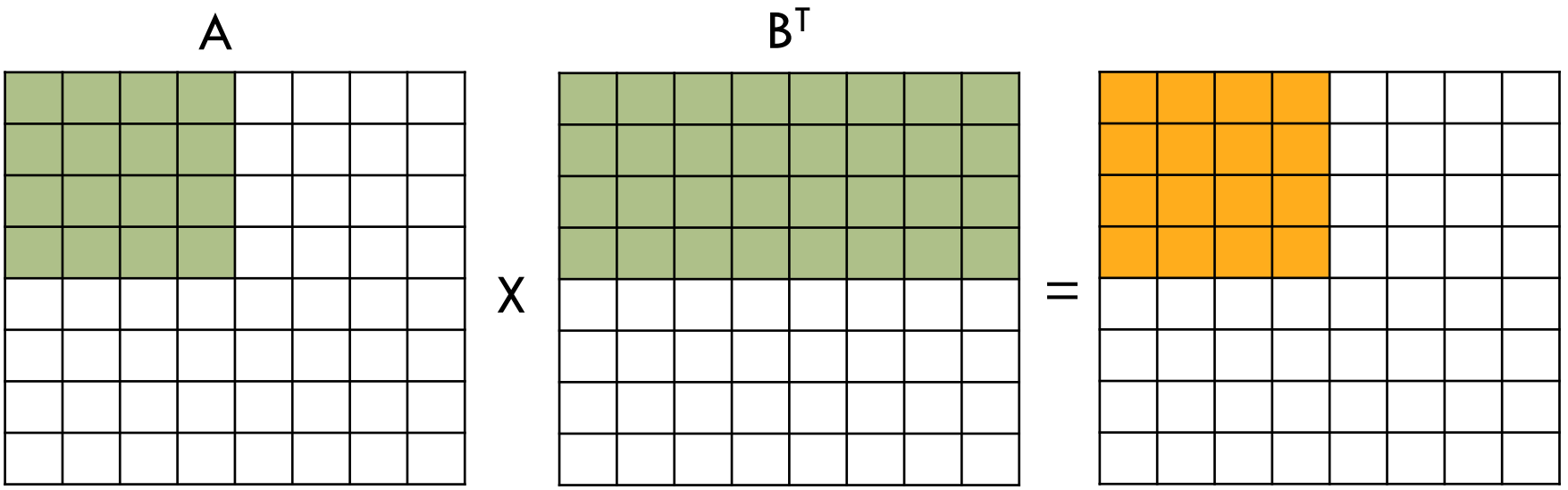
Resnično življenje

Množenje matrik (transponiran B)



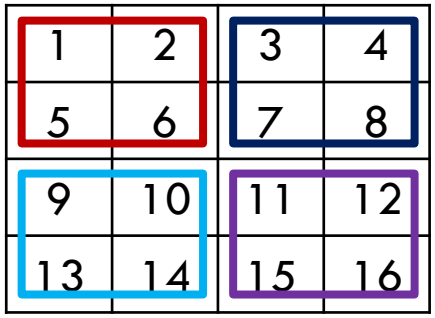
Resnično življenje

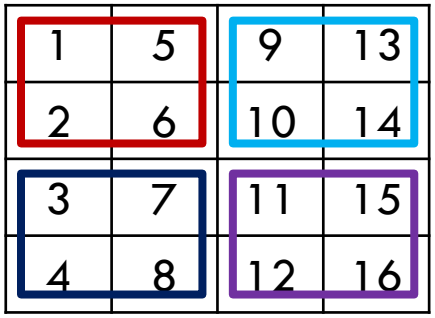
Množenje matrik (transponiran B)

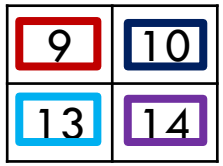


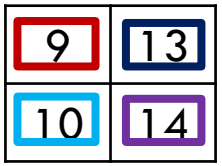
Predpomnilniško nezavedni algoritmi

- Pomnilniško nezaveden pristop:
  - PREDPOSTAVKE PODATKOVNE STRUKTURE:
    - Hrani toliko podatkov, kot jih lahko
    - Ko dostopamo do vrednosti, ki je ne hrani, prebere blok



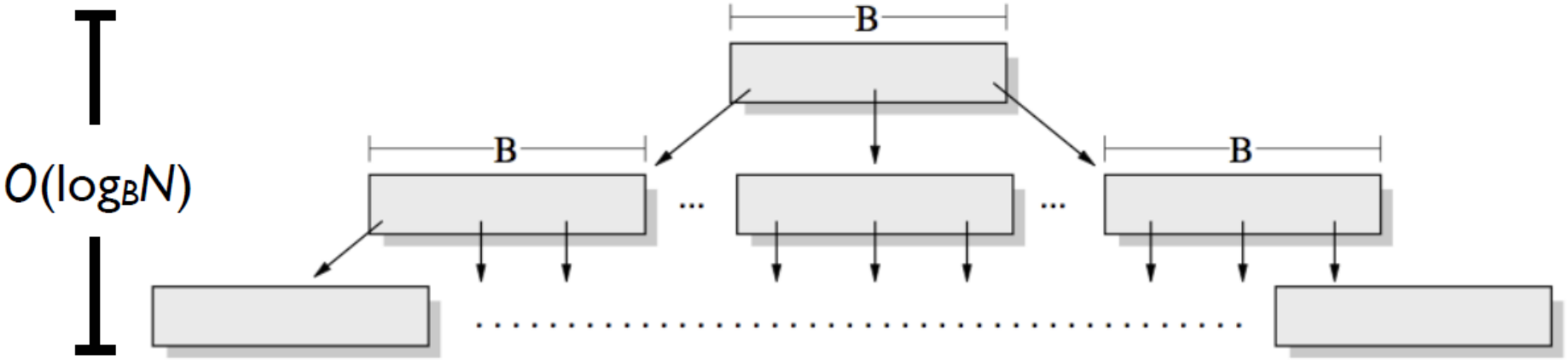






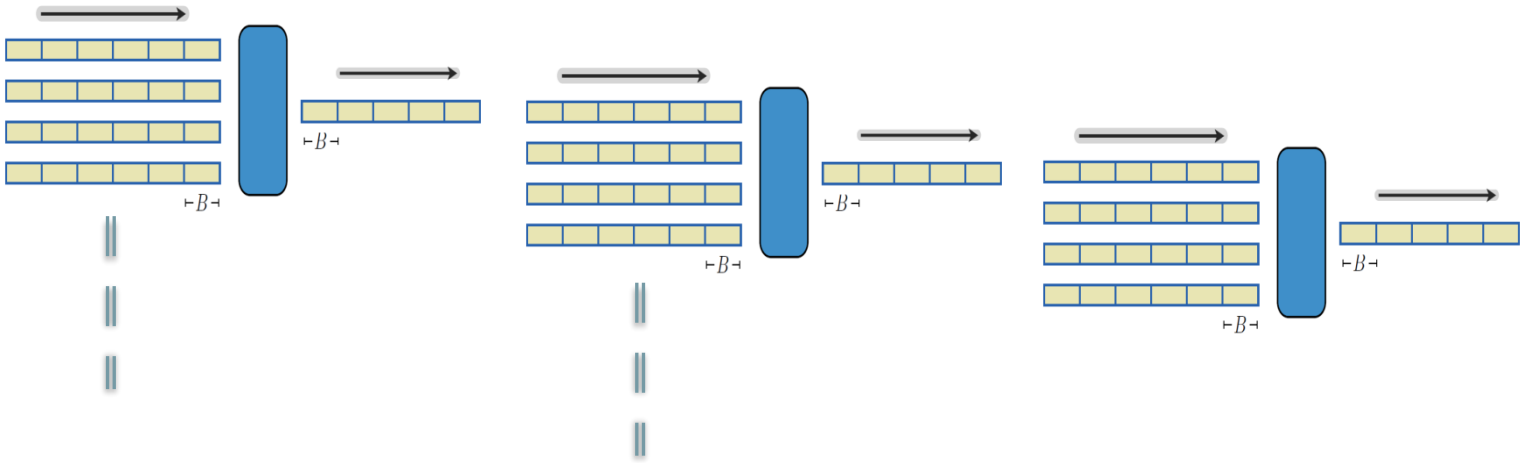
Koliko IO operacij je potrebno za iskanje elementa?

- PRIMER: Binarno drevo
  - Ne glede na velikost bloka, stvar deluje enako!



Lijačno (funnel) zlivanje

- Definicija strukture zlivanja, ki deluje učinkovito neglede na M
  - *k*-lijak – podatkovna struktura, ki zlije k urejenih vhodnih tokov
- Rekurzivno zlivanje: N^(1/3) tokov z N^(2/3) elementov



ALGORITMI ANALIZE MASIVNIH PODATKOV

DOMEN MONGUS

P03 – Opisna analiza

Motivacija

- Analiza nakupov
  - Začetek Big Data analiz
- Apriori algoritem
  - Najbolj citiran članek na področju podatkovnega rudarjenja
- Temeljno vprašanje:
  - Kakšne so navade kupcev?

Motivacija

- Kaj kupuje Homer Simpson poleg plenic?

- Odgovor:
  - Če kupuješ plenice imaš doma verjetno otroka
  - Ker imaš otroka, verjetno ne hodiš dosti v ven
  - Izkaže se, da poleg plenice kupuješ pivo

Motivacija

- Kaj kupuje Homer Simpson poleg plenic?

- Posledica:
  - V trgovinah imamo skupaj pivo in plenice
  - Plenice daš v akcijo in dvigneš ceno piva
  - Lahko damo v akcijo tudi pivo?

Vsebina

- Opisna statistika
- Asociacijska pravila
- Apriori algoritem

Opisna statistika

- Iskanje opisnih (meta) podatkov

- Statistični povzetki:
  - Srednje (pričakovane) vrednosti
  - Spremenljivost (disperzija)

- Običajno izdelamo iz histograma
  - Rešujemo problem volumna

Opisna statistika

- Osnovni podatki:
  - N = Število elementov
  - *Min* n = Najmanjši element
  - *Max* n = Največji element
  - Razpon vrednosti $n \in [Min\ n, Max\ n]$

- Histogram $H[k]$ = *število elementov v košu* k
  - K = *število košev*
  - $k \in [0, K-1]$

Opisna statistika

- Kako določiti število košev K?
  - $K = \frac{Max\ n - Min\ n}{h}$, kjer je *h* velikost koša
  - $K = \sqrt{n}$
  - Normalna porazdelitev običajno

$$k = \lceil \log_2 n \rceil + 1,$$

  - Izboljšava za nenormalno porazdelitev

$$k = 1 + \log_2(n) + \log_2\left(1 + \frac{|g_1|}{\sigma_{g_1}}\right)$$

Opisna statistika

- Povprečje na osnovi histograma

$$\bar{n} = \frac{\sum_{k=0}^{K} k * H[k]}{\sum_{k=0}^{K} H[k]}$$

- Pričakovana vrednost $= arg\ max\ H[k]$

- Kako izračunamo mediano?

- Standardni odklon

$$\bar{n} = \sqrt{\frac{\sum_{k=0}^{K} k * (H[k] - \bar{n})}{\sum_{k=0}^{K} H[k]}}$$

Opisna statistika

- Spremenljivost

$$m_1 = \frac{\sum(x_i - \bar{x})}{N}$$

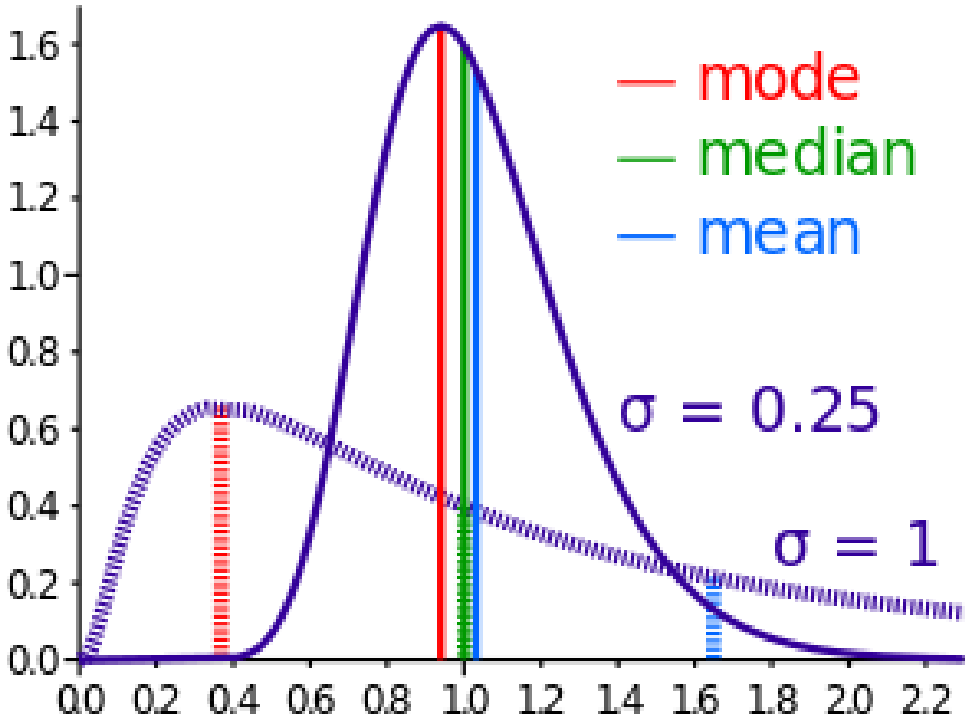
$$m_2 = \frac{\sum(x_i - \bar{x})^2}{N}$$

$$m_3 = \frac{\sum(x_i - \bar{x})^3}{N}$$

$$m_4 = \frac{\sum(x_i - \bar{x})^4}{N}$$

Asimetrija (skewness)

$$b_1 = \frac{m_3}{s^3} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^3}{\left[\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2\right]^{3/2}},$$



Opisna statistika

- Spremenljivost

$$m_1 = \frac{\sum (x_i - \bar{x})}{N}$$

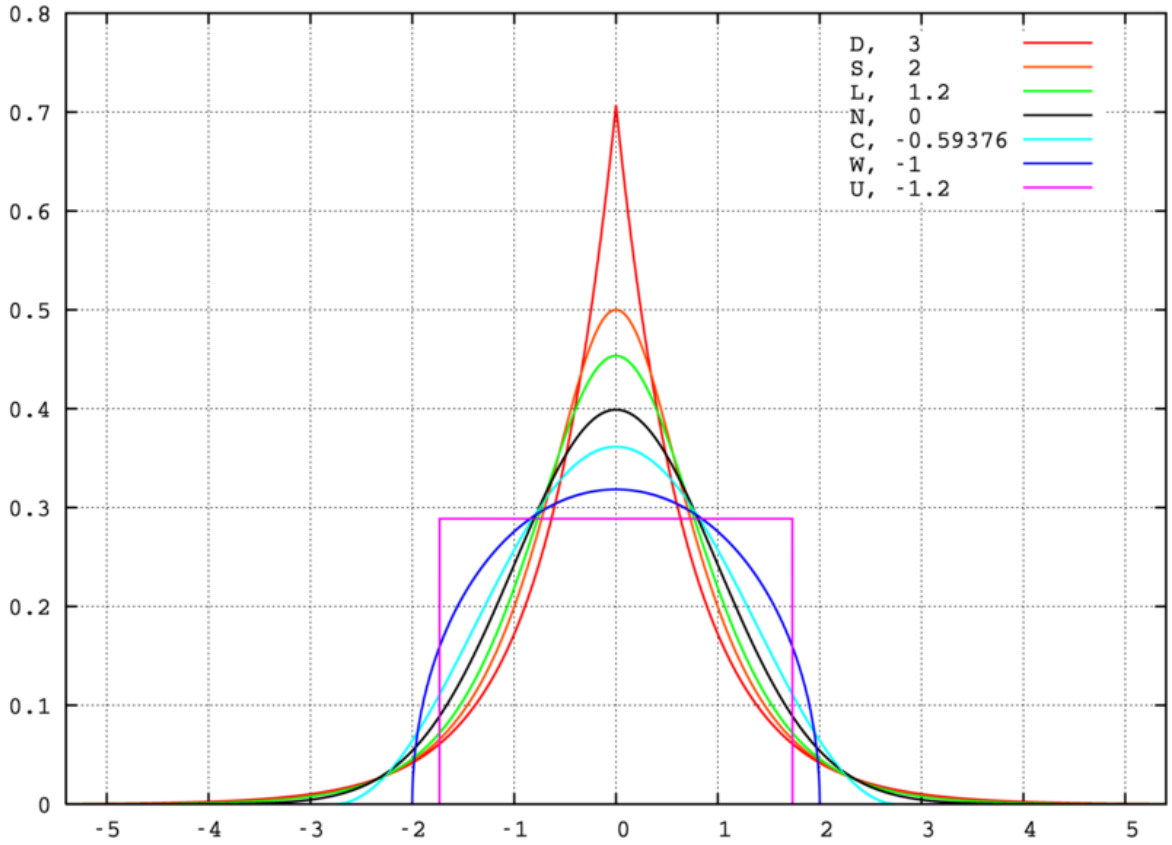
$$m_2 = \frac{\sum (x_i - \bar{x})^2}{N}$$

$$m_3 = \frac{\sum (x_i - \bar{x})^3}{N}$$

$$m_4 = \frac{\sum (x_i - \bar{x})^4}{N}$$

Sploščenost (kurtosis)

$$g_2 = \frac{m_4}{m_2^2} - 3 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2\right)^2} - 3$$



Model trgovine in nakupovalnega vozička

Množica pogosto kupljenih izdelkov

- Velik nabor artiklov
- Velik nabor nakupovalnih vozičkov
  - V vsakem vozičku malo artiklov
- Katere artikli so „pogosto“ kupljeni?
  - **Podorno število** množice *I*

$$sup(I) = \frac{\text{št. } vozičkov\ v\ katerih\ je\ I}{\text{št. } vseh\ vozičkov},$$

  - **Podporna pragovna vrednost** *s* določa množico pogosto kupljenih izdelkov

Primer

- Množica artiklov = {**m**arelice, **č**ešnje, **p**omaranče, **b**reskve, **j**agode}
- Podporna pragovna vrednost s = 33%, približno 3

  B_1={m,c,b} B_2={m,p,j}

  B_3={m, b} B_4={c,j}

  B_5={m,p,b} B_6={m,c,b,j}

  B_7={c,b,j} B_8={b,c}

- Kateri so pogosti artikli:
  - {m}, {c}, {b}, {j},
  - {m,b}, {b,c}, {c,j}

Formalizacija

- Iščemo „if-then“ relacije

  $\{i_1, i_2, \ldots, i_k\} \rightarrow j$

  $I \rightarrow j$, če $= \{i_1, i_2, \ldots, i_k\}$

- Zaupanje:

$$conf(I \rightarrow j) = \frac{\sup(I \cup j)}{\sup(I)} = P(j|I)$$

- Kakšno je zaupneje v pravilo {m,b} → c ?

Formalizacija

- Iščemo „if-then“ relacije

  $\{i_1 , i_2 , \ldots, i_k\} \rightarrow j$

  $I \rightarrow j$, če $= \{i_1 , i_2 , \ldots, i_k\}$

- Zaupanje:

  $$conf(I \rightarrow j) = \frac{\sup(I \cup j)}{\sup(I)} = P(j|I)$$

- Kakšno je zaupneje v pravilo {m,b} → c ?

  $conf(\{m,b\} \rightarrow c) = 50\%$

Naivni Algoritem

- **Zanimajo nas vsa asociativna pravila, ki imajo podporo večjo od *s* in zaupanje večje kot *c***
- Osvnovna ideja, če ima *I* podporo *s*, potem ima pravilo $I \rightarrow j$ podporo, ki je vsaj *cs*. Zato:
  - A vsebuje vse množice, ki imajo podporo vsaj *cs*
  - B vsebuje vse množice, ki imajo vsaj *s* (tako $A \subseteq B$)
    - $B \backslash A$ definira pravila
    - Poiščimo torej tiste množice, ki so v B in jih ni v A ter ugotovimo kateri element jim manjka
- Ne pozabimo, problem je Big Data!

Naivni Algoritem

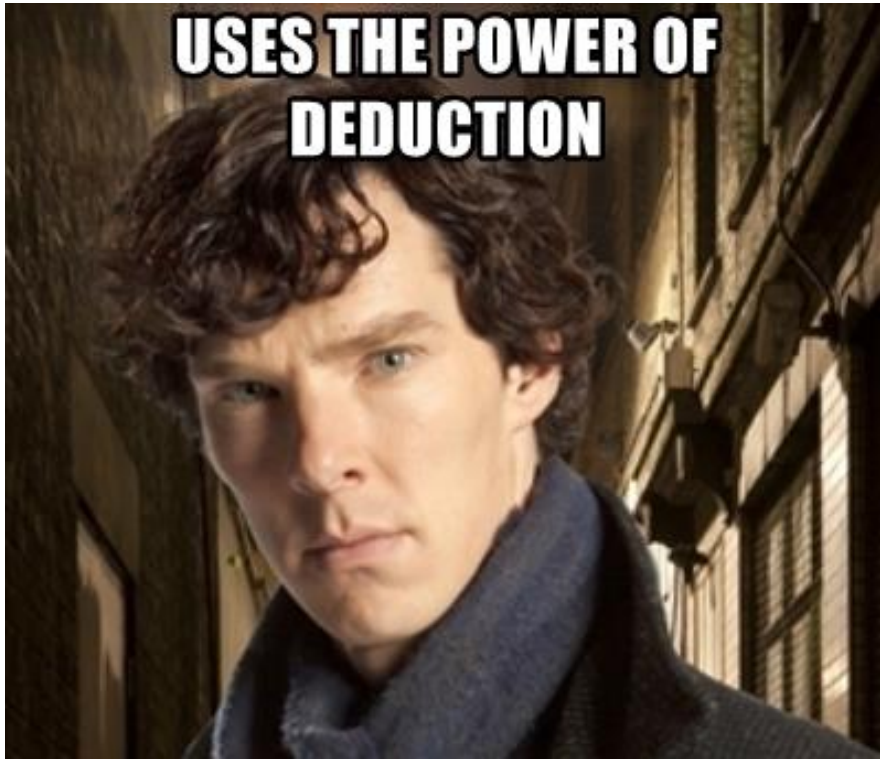
- Problem iskanja asociativnih pravil je enak problemu iskanja pogosto kupljenih artiklov
  - Zahteva uporabo DAM!
- Prešteti moremo vse množice s podporo cs
  - V bistvu histogram
- Štetje zahteva:
  - Izvedbo trikotna matrike
  - Izvedbo seznama parov

Apriorni algoritem

- Množica artiklov ne more biti pogosta, če vse njene podmnožice niso pogoste!
  - Časovna zahtevnost algoritma je enaka podpornemu številu, ki ga iščemo!
- Postopno filtriranje glede na dedukcijo!☺
  - **Monotonost (matematično):**

    $$\forall x, y : x \leq y \Rightarrow f(x) \leq f(y)$$

  - **Praktično:** če se par artiklov pojavi v s nakupovalnih vozičkih, se vsak izmed para artiklov sam pojavi vsaj s-krat.



Apriorni algoritem

- **Z drugimi besedami:** če se artikel ne pojavi v vozičku vsaj *s*-krat, potem se tudi noben izmed njegovih parov ne:
  - **Prehod 1**: Preštejemo število pojavitev vsakega artikla in brišemo vse, ki se ne pojavijo vsaj *s*-krat
    - Generiramo seznam kandidatov parov pogosto kupljenih artiklov
  - **Prehod 2**: Preštejemo kolikokrat se pojavijo pari pogosto kupljenih artiklov (vse filtrirane preskočimo) in zopet filtriramo.
    - Generiramo seznam trojic pogosto kupljenih artiklov.

Apriorni algoritem

- **Generiranje trojic pogosto kupljenih artiklov:**
  - Izberemo par, iz katerega želimo generirati trojice
  - Izberemo drugi par, ki vsebuje vsaj en element, v prvem paru
  - Dobimo tretji element, ki definira trojico.

Primer

- Množica artiklov = {**m**arelice, **č**ešnje, **p**omaranče, **b**reskve, **j**agode}
- Podporna pragovna vrednost s = 3 in c=50%

B_1={m,c,b} B_2={m,p,j}

B_3={m, b} B_4={c,j}

B_5={m,p,b} B_6={m,c,b,j}

B_7={c,b,j} B_8={b,c}

- Korak 1:
  - sup({m})=5
  - sup({c})=5
  - sup({p})=2
  - sup({b})=5
  - sup({j})=4
- Filtriranje
  - {m,c,b,j}

Primer

- Množica artiklov = {**m**arelice, **č**ešnje, **p**omaranče, **b**reskve, **j**agode}
- Podporna pragovna vrednost s = 3 in c=50%

B_1={m,c,b} B_2={m,p,j}

B_3={m, b} B_4={c,j}

B_5={m,p,b} B_6={m,c,b,j}

B_7={c,b,j} B_8={b,c}

- Filtriranje
  - {m,c,b,j}
- Generiramo pare:
  - sup({m,c})=2
  - sup({m,b})=4
  - sup({m,j})=2
  - sup({c,b})=4
  - sup({c,j})=3
  - sup({b,j})=2

Primer

- Množica artiklov = {**m**arelice, **č**ešnje, **p**omaranče, **b**reskve, **j**agode}
- Podporna pragovna vrednost s = 3 in c=50%

B_1={m,c,b} B_2={m,p,j}
B_3={m, b} B_4={c,j}
B_5={m,p,b} B_6={m,c,b,j}
B_7={c,b,j} B_8={b,c}

- Generiramo pare:
  - sup({m,c})=2
  - sup({m,b})=4
  - sup({m,j})=2
  - sup({c,b})=4
  - sup({c,j})=3
  - sup({b,j})=2
- Filtriranje
  - Število parov je 6
  - Prag = 3

Primer

- Množica artiklov = {**m**arelice, **č**ešnje, **p**omaranče, **b**reskve, **j**agode}
- Podporna pragovna vrednost s = 3 in c=50%

B_1={m,c,b} B_2={m,p,j}

B_3={m, b} B_4={c,j}

B_5={m,p,b} B_6={m,c,b,j}

B_7={c,b,j} B_8={b,c}

- Filtriranje
  - sup({m,b})=4
  - sup({c,b})=4
  - sup({c,j})=3
- Izračunamo trojice
  - sup({m,b,c})=2
  - sup({c,b,j})=2

ALGORITMI ANALIZE MASIVNIH PODATKOV

DOMEN MONGUS

P04 – Analiza časovnih vrst

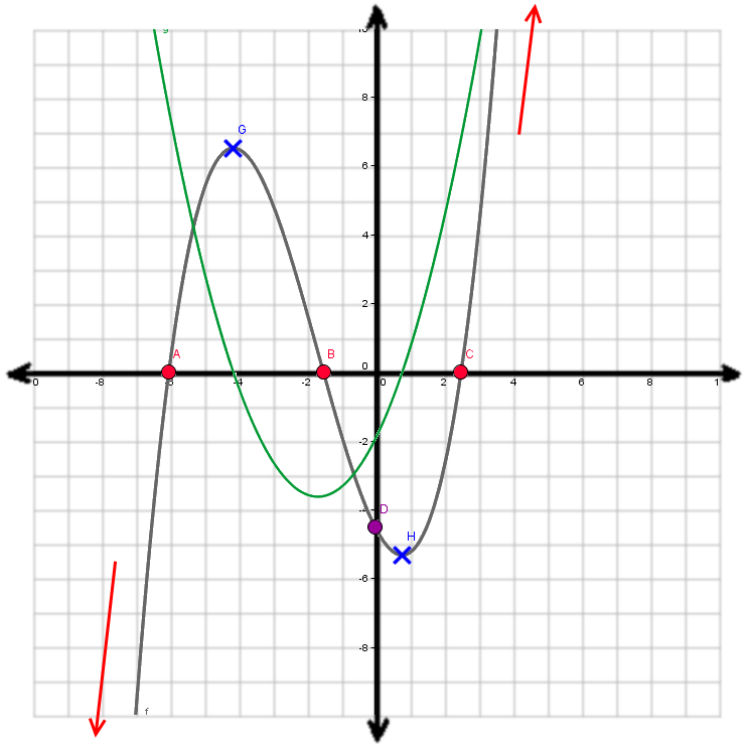
Motivacija - V 4 Velocity

- Časovna vrsta
  - Časovno urejena množica opazovanj

- Aplikacije:
  - Vremenske napovedi (temperatura, vlažnost, …)
  - Finančni trendi (vrednost valut, delnic …)
  - Povpraševanje po dobrinah (nakupi)
  - Medicina (srčni utrip, EEG,…)

Motivacija - V 4 Velocity

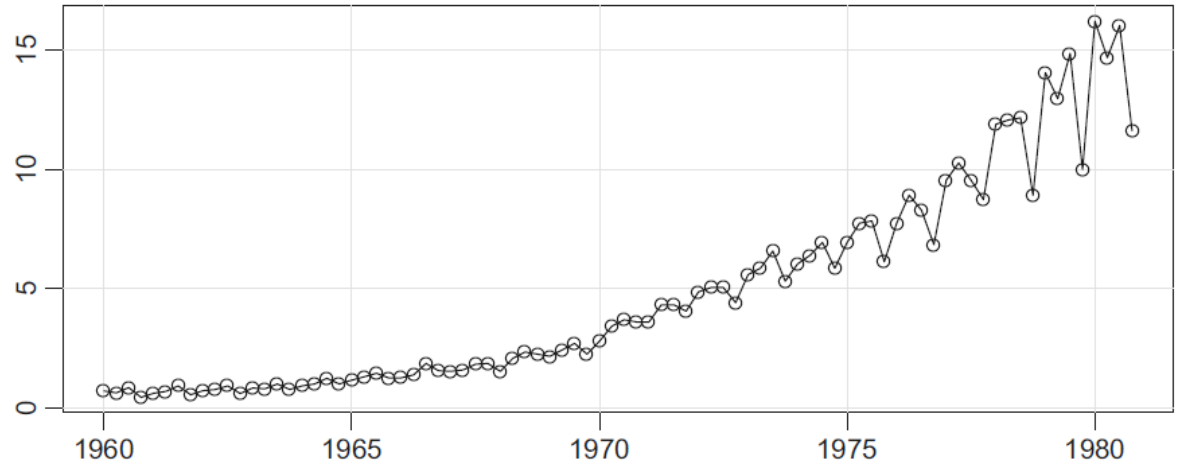
- Časovna vrsta
  - V čem je razlika?
- Regresija (tradicionalno)
  - Ciljna spremenljivka
  - Razlagalne spremenljivke



$$y_i = \beta_0 1 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\top \boldsymbol{\beta} + \varepsilon_i, \qquad i = 1, \ldots, n,$$

Motivacija - V 4 Velocity

- Časovna vrsta
  - Tradicionalna časovna vrsta (četrtletni zaslužki podjetja)



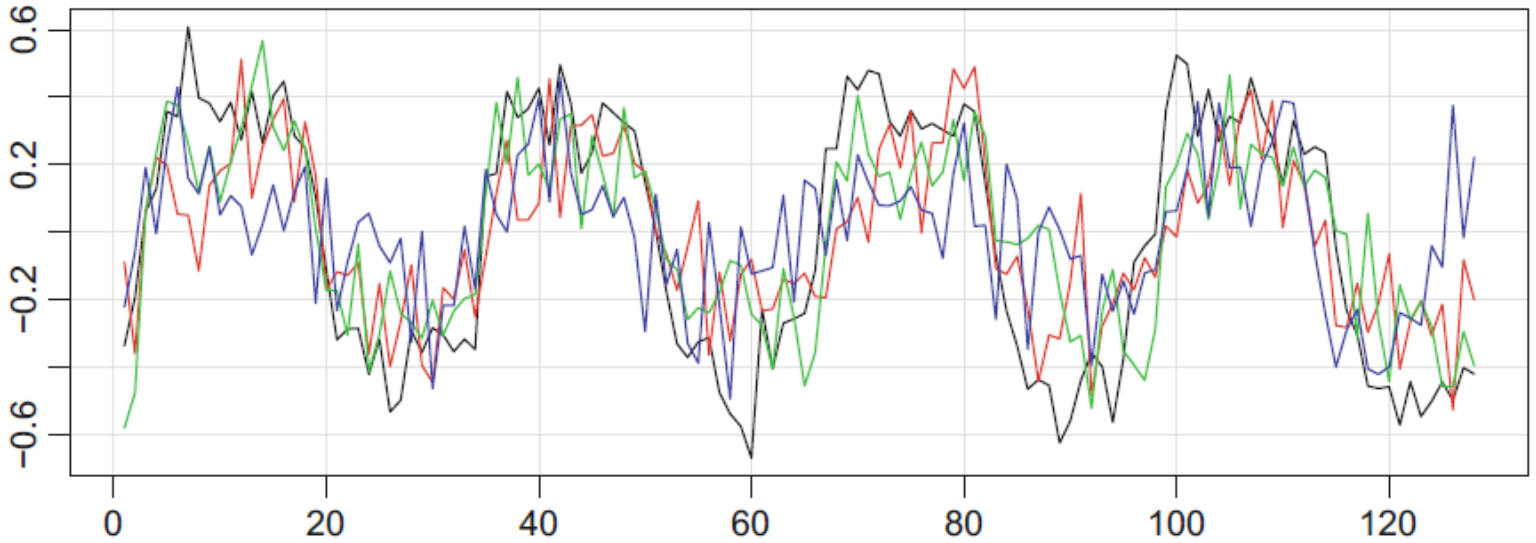
  - Analiza vsebovanih vzorcev za predvidevanje:
    - Trendi, cikli, šum, povezave z zunanjimi okoliščinami …

Vsebina

- Analiza in lastnosti časovnih vrst
- Napovedovanje vrednosti
  - Autokorelacija
  - ARIMA

Narava časovnih vrst

- Obravnavali bomo le **univariantne diskretne časovne vrste**
  - Univariantnost – spremljamo eno samo spremenljivko
  - Meritve izvajamo v enakomernih časovnih korakih
- Notacija
  - Naključna spremenljivka $X = \{x_t\}$, kjer t predstavlja čas
  - $t \in \{1, 2, \ldots, T\}$
- Variabilnost:



Osnovni matematični model

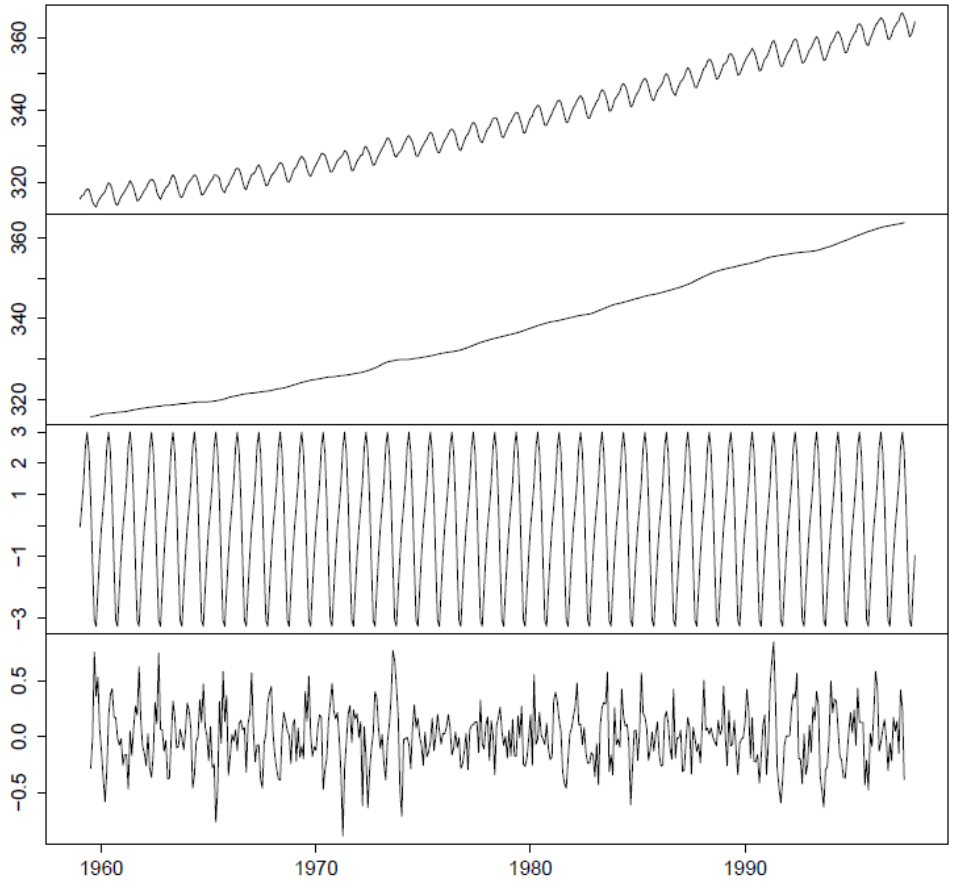
- Notacija
  - Naključna spremenljivka $X = \{x_t\}$, kjer t predstavlja čas
  - $t \in \{1,2, \dots , T\}$

- Naivna različica: $x_t = f(t)$
  - Zaradi visoke stopnje variabilnosti skoraj nikoli ni učinkovit

- Splošni model: $x_t = f(t) + \varepsilon$
  - $f(t)$ – deterministični del, ki sledi časovnim zakonitostim
  - ε – naključni del, ki sledi zakonom verjetnosti

Osnovni matematični model

- Razlogi za variabilnost vrednosti
  - Dekompozicija signala:
  - Trend:
  - Sezonski efekti:
  - Neregularne fluktuacije:



Stacionarna časovna vrsta

- Definicija:
  - Časovna vrsta je stacionarna, kadar je verjetnost pojavitve vsake vrednosti $X = \{x_t\}$ enaka verjetnosti pojavitve vsake vrednosti v drugem časovnem obdobju $X_h = \{x_{t+h}\}$,
  - Taka časovna vrsta je odvisna zgolj od časovne razlike in ne od dejanskega časa!
- Šibko stacionarna:
  - Povprečje je konstanta
- Zakaj je stacionarnost koristna?

Avtokorelacija

- Pearsonov korelacijski koeficient

$$r = r_{xy} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sqrt{(\sum x_i^2 - n\bar{x}^2)}\sqrt{(\sum y_i^2 - n\bar{y}^2)}}.$$

- kjer
  - *n* – število elementov
  - *x*,*y* – spremenljivki

Avtokorelacija

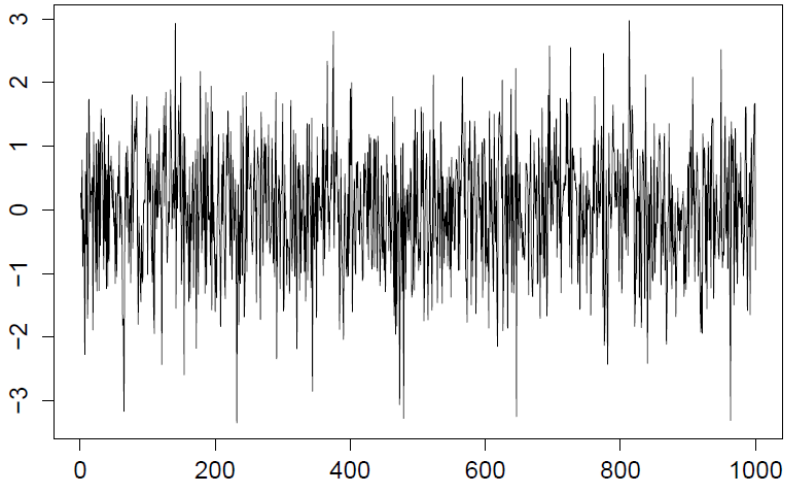
- Definicija
  - Korelacija med signalom $X = \{x_t\}$ in njegovo zakasnjeno kopijo $X_h = \{x_{t+h}\}$
- Naivni pristop k napovedovanju:
  - Poiskati najprimernejši h
  - $x_{t+1} = x_{t+h+1}$
- Določa sezonski efekt oz. periodičnost signala

Tradicionalne časovne vrste

Naključne vrednosti

- Nabor vrednosti iz območja [X,Y]
- Ima konstantno povprečje
- Konstantno varianco
- Je stacionaren

Beli šum (Gaussovo naključje)

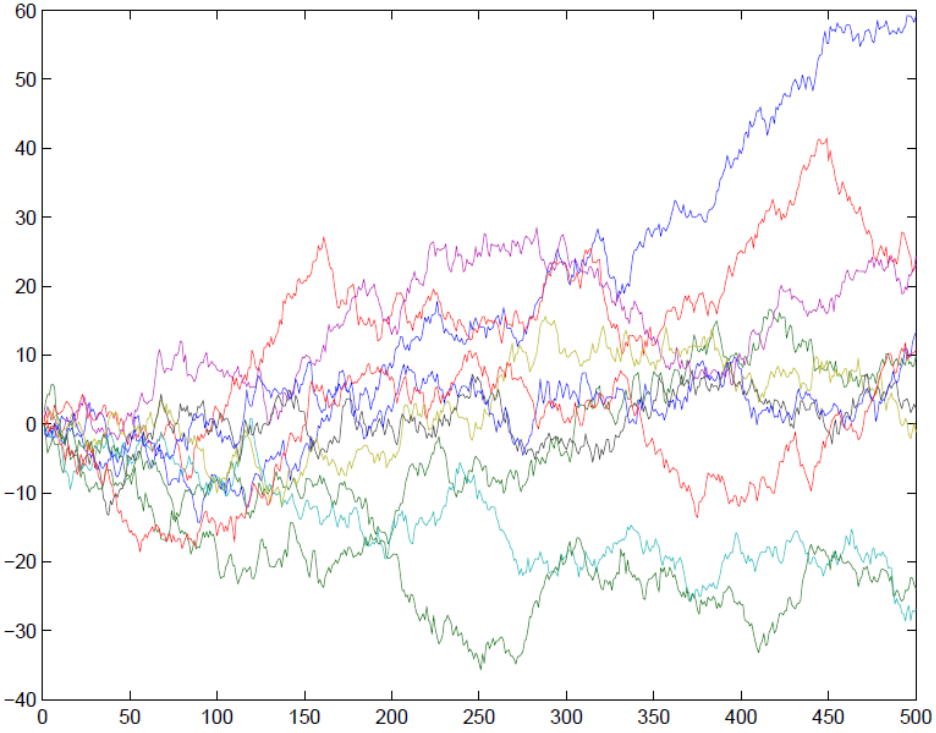


Tradicionalne časovne vrste

Naključni sprehod

- $x_{t+1} = x_t + w_t$, kjer
  - je w_t naključna vrednost
- Povprečje se spreminja
- Tudi varianca se spremenija
- Ni stacionaren

10 naključnih sprehodov:



Tradicionalne časovne vrste

Naključni sprehod

- $x_{t+1} = x_t + w_t$, kjer
  - je w_t naključna vrednost
- Povprečje se spreminja
- Tudi varianca se spremenija
- Ni stacionaren

Diferenciacija

- Odvod naključni sprehod
  - $\Delta x_{t+1} = x_{t+1} - x_t = w_t$
- Ker je w_t povsem naključna vrednost
  - je Δx_{t+1} stacionaren!

Ocena trenda – tradicionalna regresija

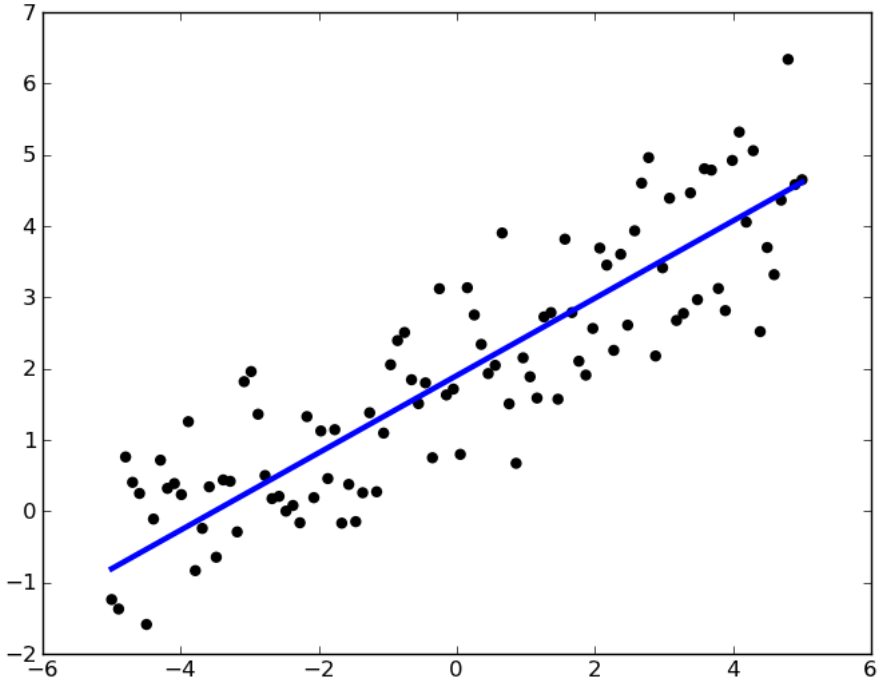
- Definicija
  - Korelacija med signalom $X = \{x_t\}$ in njegovo zakasnjeno kopijo $X_h = \{x_{t+h}\}$
- Naivni pristop k napovedovanju:
  - Poiskati najprimernejši h
  - $x_{t+1} = x_{t+h+1}$
- Takšen pristop lahko uporabimo zgolj nad stacionarno časovno vrsto.

Navadna linearna regresija

- Ocena dolgoročnega trenda
- Minimizacija napake
  - Differencialne enačbe
- Metoda najmanjših kvadratov

  (-) Poudari outlierje

  (+)Enostavna reševanje



Metoda najmanjših kvadratov

- Centriranje podatkov
  - Črta gre skozi koordinatno izhodišče

$$\begin{aligned} y_i &= b_0 + b_1 x_i \\ \overline{y} &= b_0 + b_1 \overline{x} \\ y_i - \overline{y} &= 0 + b_1 (x_i - \overline{x}) \end{aligned}$$

- Splošni model

$$y_i = \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_k x_{i,k} + \varepsilon_i$$

  - *k* koeficientov za *k* parametrov
  - in napaka ε_i
  - V matrični obliki:

$$y_i = [x_{i,1},\ x_{i,2},\ \ldots,\ x_{i,k}] \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \varepsilon_i$$

$$y_i = \underbrace{x_i^T}_{(1\times k)} \underbrace{\beta}_{(k\times 1)} + \varepsilon_i$$

Metoda najmanjših kvadratov

- Če imamo več meritev, lahko izdelamo matriko
  - b predstavlja oceno dejanske vrednosti

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,k} \\ x_{2,1} & x_{2,2} & \dots & x_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \dots & x_{n,k} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

- Generalizirano $\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{e}$
- Velikosti matrik:

$\mathbf{y}$: $n \times 1$

$\mathbf{X}$: $n \times k$

$\mathbf{b}$: $k \times 1$

$\mathbf{e}$: $n \times 1$

Metoda najmanjših kvadratov

- Minimizacija kvadratov napak
- Rešitev: $\frac{f(\mathbf{b})}{\partial \mathbf{b}} = 0$

$$\begin{aligned} f(\mathbf{b}) &= \mathbf{e}^T\mathbf{e} \\ &= (\mathbf{y} - \mathbf{X}\mathbf{b})^T(\mathbf{y} - \mathbf{X}\mathbf{b}) \\ &= \mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{X}\mathbf{b} + \mathbf{b}\mathbf{X}^T\mathbf{X}\mathbf{b} \end{aligned}$$

- Po nekaj napora lahko z diferencialnimi enačbami ugotovimo
  - Ta formula je biblija!

$$\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

- Inverzna matrika:

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}.$$

Metoda najmanjših kvadratov - Primer

- Ne pozabimo: $\mathbf{b} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$
- Vhod:

$$x_{1,\text{original}} = [1,\ 3,\ 4,\ 7,\ 9,\ 9] \qquad x_1 = [-4.5,\ -2.5,\ -1.5,\ 1.5,\ 3.5,\ 3.5]$$

$$x_{2,\text{original}} = [9,\ 9,\ 6,\ 3,\ 1,\ 2] \qquad x_2 = [4,\ 4,\ 1,\ -2,\ -4,\ -3]$$

$$y_{\text{original}} = [3,\ 5,\ 6,\ 8,\ 7,\ 10] \qquad y = [-3.5,\ -1.5,\ -0.5,\ 1.5,\ 0.5,\ 3.5]$$

$$\mathbf{X} = \begin{bmatrix} -4.5 & 4 \\ -2.5 & 4 \\ -1.5 & 1 \\ 1.5 & -2 \\ 3.5 & -4 \\ 3.5 & -3 \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} -3.5 \\ -1.5 \\ -0.5 \\ 1.5 \\ 0.5 \\ 3.5 \end{bmatrix}$$

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} 55.5 & -57.0 \\ -57.0 & 62 \end{bmatrix} \qquad \mathbf{X}^T\mathbf{y} = \begin{bmatrix} 36.5 \\ -36.0 \end{bmatrix}$$

$$\mathbf{X}^T\mathbf{X}^{-1} = \begin{bmatrix} 62 & 57.0 \\ 57.0 & 55.5 \end{bmatrix} \qquad \mathbf{X}^T\mathbf{y} = \begin{bmatrix} 36.5 \\ -36.0 \end{bmatrix}$$

Inverzna matrika

- Ni enostavno!

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}.$$

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & B & C \\ D & E & F \\ G & H & I \end{bmatrix}^{\mathrm{T}} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & D & G \\ B & E & H \\ C & F & I \end{bmatrix}$$

- Bločna inverzija

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \\ -(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{bmatrix}$$

Metoda najmanjših kvadratov - Primer

- Rezultat $b1 = 1.01$ in $b2 = 0.43$?

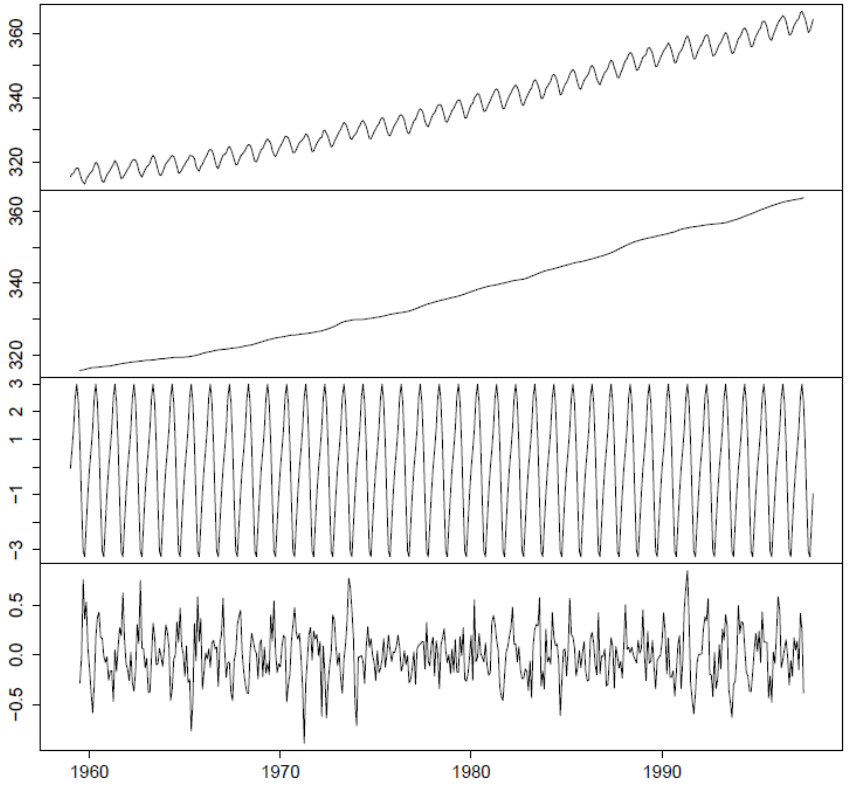
ALGORITMI ANALIZE MASIVNIH PODATKOV

DOMEN MONGUS

P05 – Analiza časovnih vrst

Motivacija - V 4 Velocity

- Časovna vrsta:
  - Beli šum, naključni sprehod,…
- Razlogi za variabilnost vrednosti
  - Dekompozicija signala:
  - Trend:
  - Sezonski efekti:
  - Neregularne fluktuacije



Avtokorelacija in linearna regresija

- Pearsonov korelacijski koeficient

$$r = r_{xy} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sqrt{(\sum x_i^2 - n\bar{x}^2)}\sqrt{(\sum y_i^2 - n\bar{y}^2)}}.$$

- Avtokorelacija je korelacija med signalom $X = \{x_t\}$ in njegovo zakasnjeno kopijo $X_h = \{x_{t+h}\}$

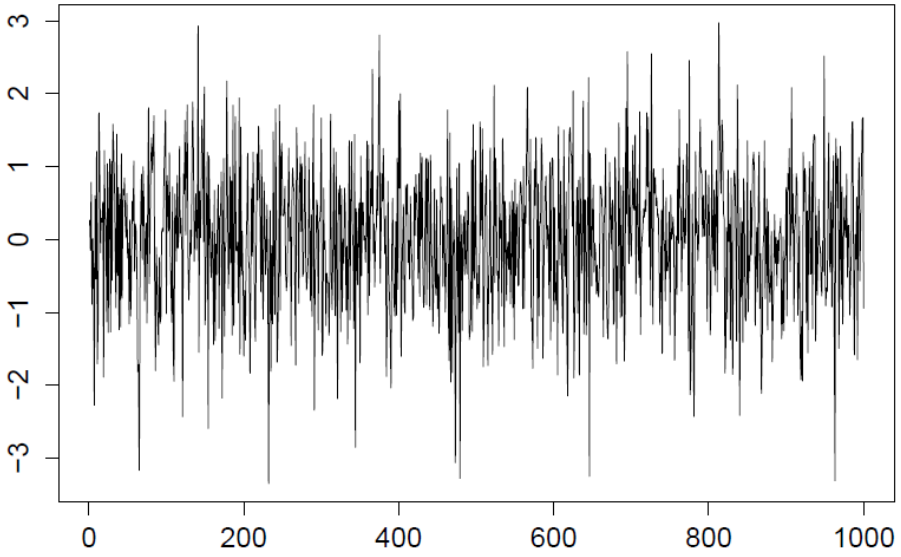
- Generalizirana regresijska enačba $\mathbf{y} = \mathbf{Xb} + \mathbf{e}$
  - Metoda najmanjših kvadratov $\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

Vsebina

- Tradicionalni pristopi, ki napovedovanju vrednosti v časovnih vrstah:
  - Premikajoče povprečje
  - Avtoregresijski model
  - ARIMA
- Načrtovanje napovedovalnih modelov

Beli šum

- Definicija
  - Povprečje = *0*
  - Varianca = *konstanta*
  - Je nekoreliran
- V signalu ne ugotovimo vzorca
  - Množica statističnih testov
- Zaključni kriterij vsake časovne analize.



Nelinearna regresija z metodo najmanjših kvadratov

- Matrična predstavitev metode najmanjših kvadratov v linearnem sistemu:

$$y_i = \beta_1 x_{i,1} + \beta_2 x_{i,2} + \ldots + \beta_k x_{i,k} + \varepsilon_i$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \ldots & x_{1,k} \\ x_{2,1} & x_{2,2} & \ldots & x_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \ldots & x_{n,k} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

- Polinomska regresija:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_m x_i^m + \varepsilon_i \ (i = 1, 2, \ldots, n)$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^m \\ 1 & x_2 & x_2^2 & \ldots & x_2^m \\ 1 & x_3 & x_3^2 & \ldots & x_3^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^m \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_m \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

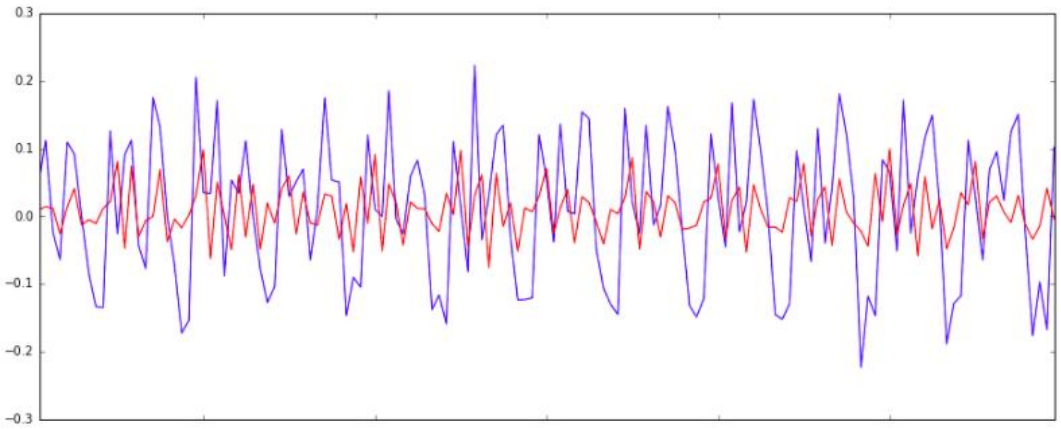
Kako z več spremenljivkami?

Avtoregresijski model

- Notacija *AR(p)*
  - *p* – število preteklih vrednosti, ki jih uporabimo za napoved
- Definicija: $x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + w_t$
  - Rešljivo s tradicionalno linearno regresijo
- Zahteva stacionarne vrednosti!
  - Povprečje = 0
  - Standardna deviacija = konstanta

Premikajoče povprečje

- Ang. moving average
- Notacija: MA(q)
  - q – določa dolžino modela
- Definicija: $X_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}$
  - θ_q - koeficienti (povezani z napakami), ki jih računamo
  - ε_t, ε_{t-1},…, ε_{t-q} - napake prejšnjih napovedih (stacionarni)
  - μ - povprečje zadnjih *q* vrednosti



Reševanje MA modelov

- Napake v napovedovanju so nedoločljive, zato potrebujemo iterativni pristop.

- Množica možnih pristopov:
  - Metoda Yule–Walker
  - Metoda največje verjetnosti (Maximum Likelihood)
    - Newton–Raphsonov in Scoring Algorithmi
  - Iterativni Gauss – Newtonov algoritem

Avtoregresijsko premikajoče povprečje

- Ang. autoregresive moving average
- *Notacija: ARMA(p,q)*
  - *p* – število avtoregresijskih členov
  - q – ševilo členov belega šuma (napak)

- Definicija:

$$x_t = \phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} + w_t + \theta_1 w_{t-1} + \cdots + \theta_q w_{t-q}$$

Integrirana ARMA

- Ang. Avtoregresive Integrated Moving Average
- *Notacija: ARIMA(p,d,q)*
  - *p* in *q* prevzeta iz ARMA
  - *d* – red diferenciacije
- Diferenciacija:
  - Prvi red: $y'_t = y_t - y_{t-1}$
  - Drugi red: $$\begin{aligned} y^*_t &= y'_t - y'_{t-1} \\ &= (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) \\ &= y_t - 2y_{t-1} + y_{t-2} \end{aligned}$$

Višje običajno ne gremo…

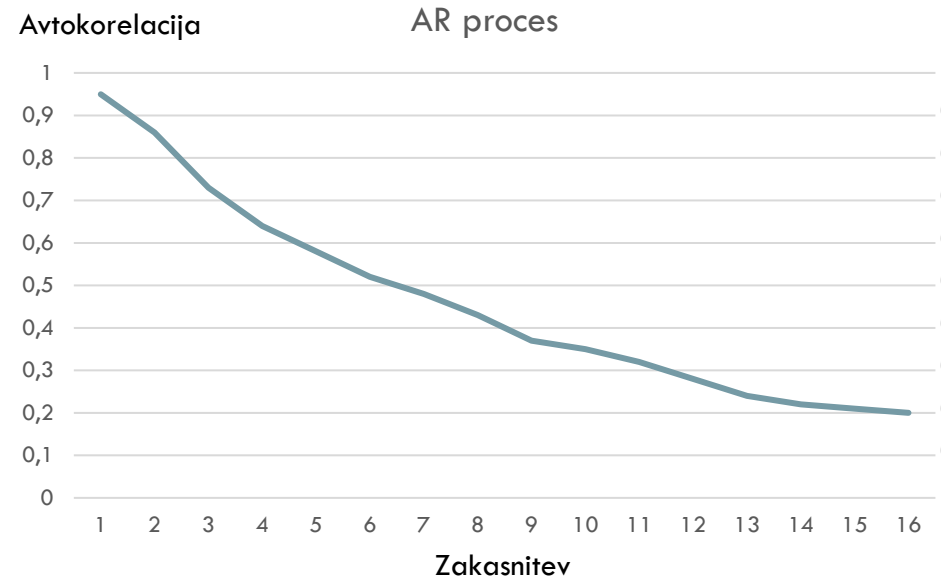
Načrtovanje modelov

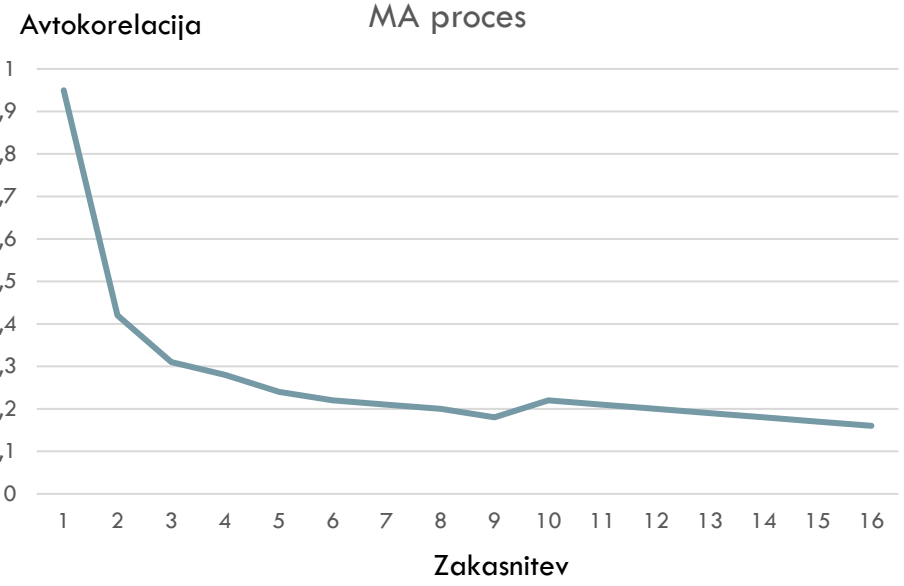
- Metoda Box-Jenkins
  - Identifikacija modela
  - Izvedba modela
  - Diagnostika

- Pred uporabo B-J izvedi:
  - Ali so podatki beli šum?
  - Ali je časovna vrsta stacionarna?
    - Če ni, izvedi diferenciacijo

Izračun korelograma

- Procese modeliramo glede ne vrste, ki jih razberemo na osnovi korelograma:
  - To je graf avtokorelacije glede na zakasnitev
  - Lahko izberemo optimalno zakasnitev?





Delna avtokorelacija

- Korelacija je približek regresije z navadnimi najmanjšimi kvadrati:
  - x: {22,17,16,14,13,10,1215,21,19,18,16,19,20,24}
  - x_1 : {17,16,14,13,10,1215,21,19,18,16,19,20,24, 21}
  - Korelacijski faktor: 0.68132
  - Regresijski koeficient (AR(1) model): 0.69608
    - Vsaka vrednost „ima 69% vpliva na naslednjo vrednost“
- PRIMER AR(3) modela:
  - $X = k_1 x_1 + k_2 x_2 + k_3$
  - Zanima nas koliko točno vpliva $k_2 x_2$ npr. brez $k_1 x_1$
    - *Izkaže se, da je vpliv n-tega zamika enak koeficientu n-tega člena regresije n-zamikov.*

Metoda Box-Jenkins

- Identifikacija modela in optimalne zakasnitve
  - Analiza avtokorelacije za določitev najprimernejše zakasnitve (optimal lag).
  - Upoštevanje delne avtokorelacije
    - Izločanje ekstremnih dogodkov

MODEL	Avtokorelacija	Delna avtokorelacija
AR(p)	Pada počasi proti 0	Takoj upade blizu 0
MA(q)	Takoj upade blizu 0	Pada počasi proti 0
ARMA(p,q)	Pada počasi proti 0	Pada počasi proti 0

Metoda B-J

- Diagnostika:
  - Običajno izračun korena povprečne kvadratne napake (ang. root mean square error)

$$\mathrm{RMSE} = \sqrt{\sum \frac{(y_{pred} - y_{ref})^2}{N}}$$

  - Izris grafa napake in preveri ali so napake res beli šum?