

Eötvös Loránd Kollégium Informatika műhely 2013. 03. 26.

A felhőkön túl – FuturICT teljesen elosztott adatbányászat Ormándi Róbert



FuturICT



- The ultimate goal of FuturICT is to understand and manage complex, global, socially interactive systems, with a focus on sustainability and resilience.
- A FuturICT projekt végső célja komplex, globális, társadalmilag beágyazott rendszerek megértése és irányítása, különös tekintettel a fenntarthatóságra és a regenerálódás képességére



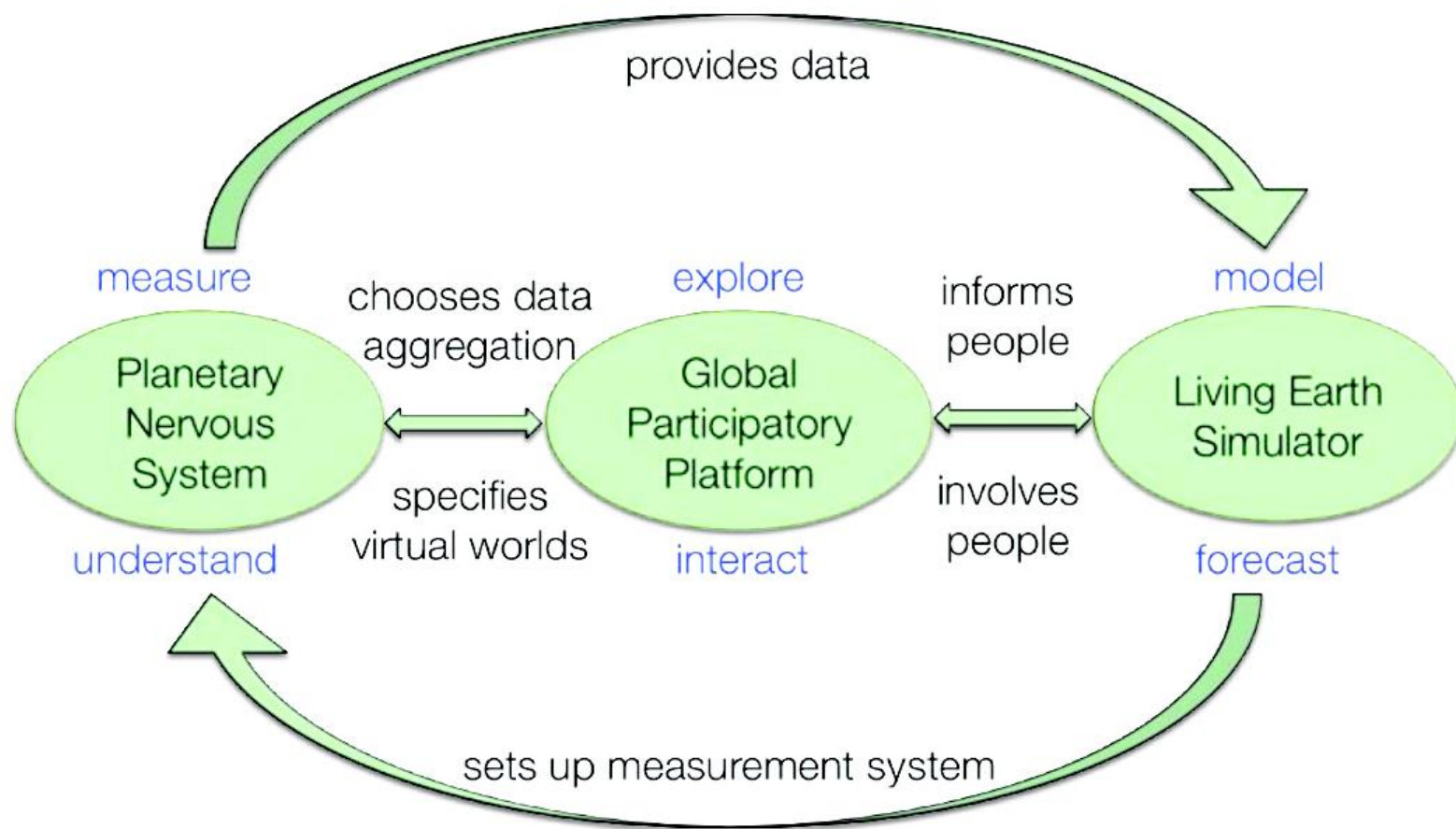
Motiváció



- Információs és kommunikációs technológia
 - Példa nélküli adatmennyiség az emberekről és környezetükről
 - Óriási számítási kapacitás
- Komplex rendszerek és hálózatok tudománya
 - Óriási önszervező rendszerek
 - Kapcsolati hálózatok szerepe, kialakulása
- E két komponenst házasítjuk össze
 - Társadalom, klíma, járványok, stb. megértése



FuturICT Platform



Résztvevők



- SZTE
- BME
- ELTE
- Közép-európai Egyetem
- SZTAKI
- Corvinus
- KFKI
- MTA TTK

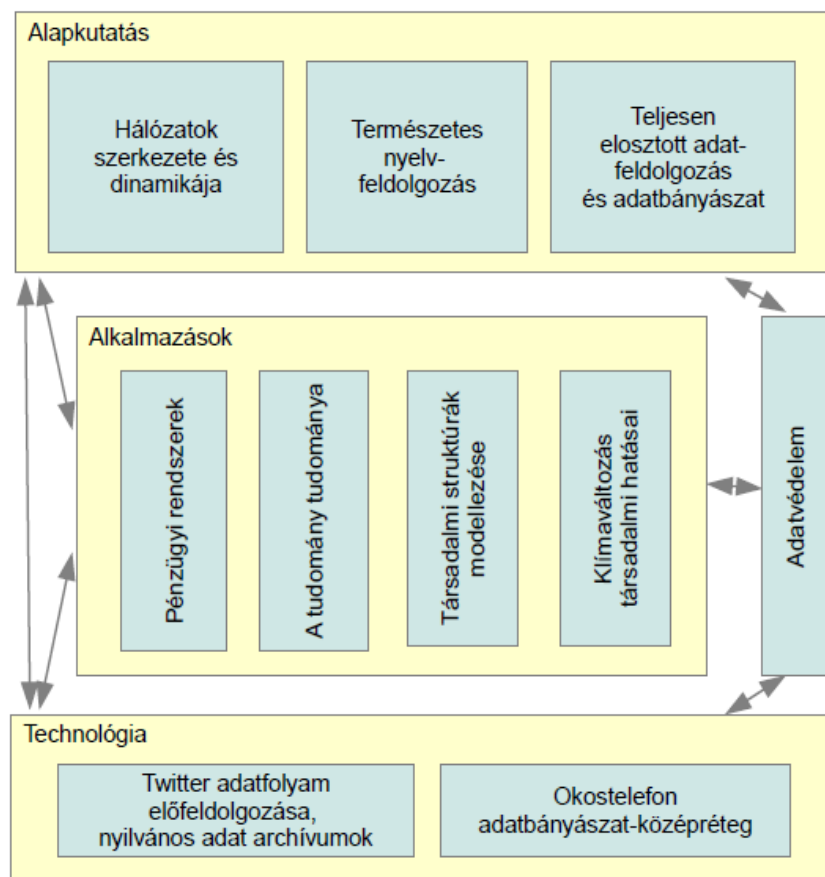


TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

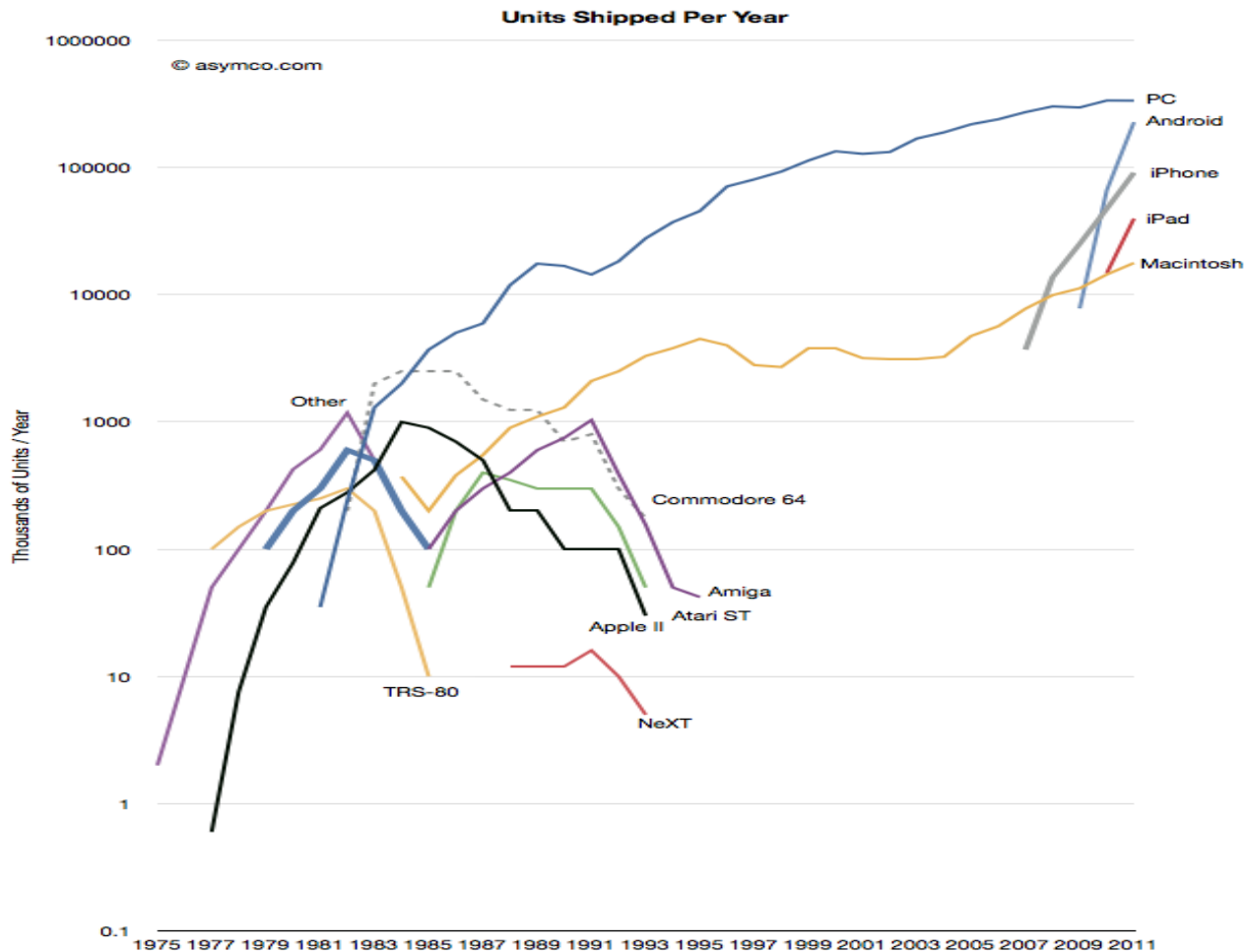
Alprojektek



- Magyar hub erősségei
 - Hálózatelmélet és alkalmazásai
 - “planetary nervous system” azaz adatkinyerés és -bányászat és elosztott adatfeldolgozás
 - Adatvédelem, etika
- Az alprojektek ezekre épülnek

Teljesen elosztott adatbányászat alprojekt

Big Data jelenség

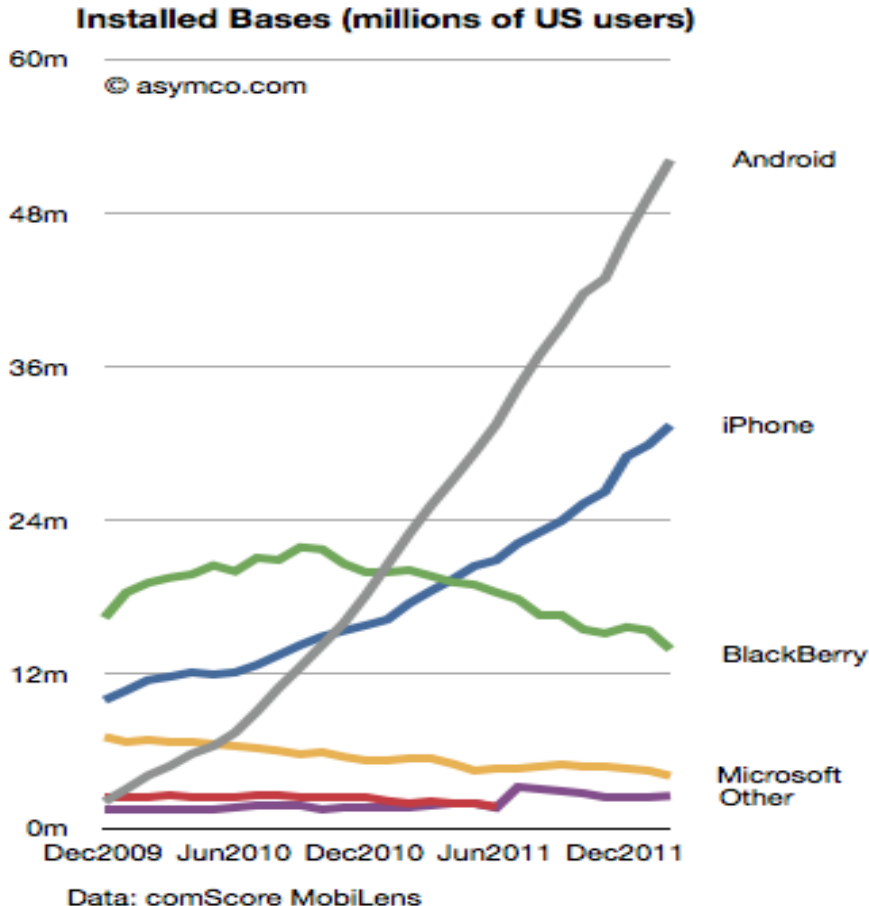


TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

Big Data jelenség



- Exponenciális növekedés a(z):
 - Okos eszközök használatában, és
 - A szenzor és egyéb "kontextuális" adatok mennyiségében
- Remek lehetőség az adatbányászatra (ismert és széleskörben használt)

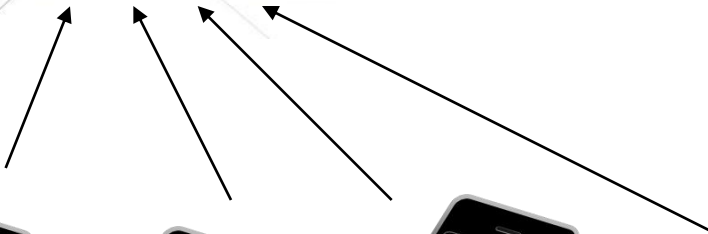
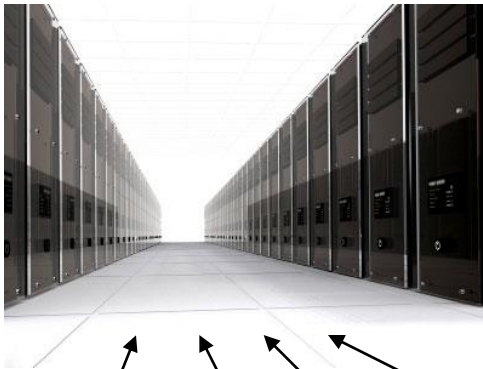
Jelenlegi rendszerek



- Jelenleg:

- Profit orientált cégek használják (korlátozott elérés):

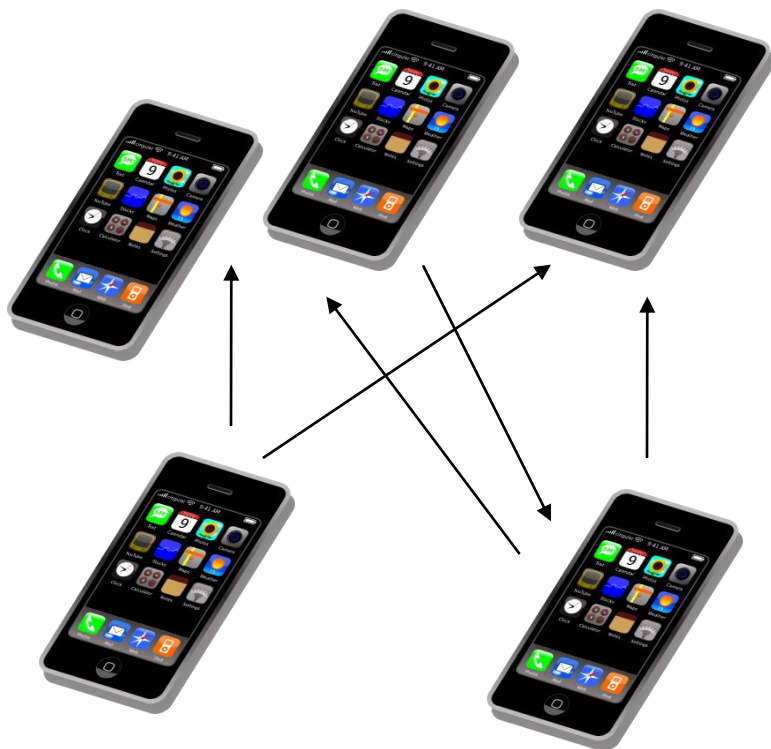
- Üzleti döntéseik támogatására: data-driven management (hipermárketek)
- Meglévő alkalmazásai "okosítására": Google news, Google Now, Gmail SPAM filtering, Evernote tag classification, Siri, Shazam,
- Üzleti modell optimalizálás



Végcél



- Teljesen elosztott (infrastruktúra nélküli) → nincs központi kontrol
- Nyílt rendszer → nem profit, hanem (köz)haszon orientált
- Adatvédelmi szempontok szemelőtt tartása → nincs harmadik fél, nincs mögöttes érdek
- Határtalan skálázódás
- Okos, megbízható szolgáltatások



Milyen környezetben?



- Rendszermodell:
 - Nagy számú csomopont, üzenetküldés alapú kommunikáció
 - Címzés egyedi azonosítókkal
 - NAT
 - Asszinkron kommunikáció
 - Tetszőleges csomag késés és csomagvesztés
 - Felhasználók tetszőlegesen és gyors ki- és belépéshetnek (churn)



Mindezt hogyan?



- Overlay hálózatok
 - T-Man, Newscast
- Teljesen elosztott, adatbányászat
 - Átlagolás példa
 - Gossip Learning Framework - P2Pegasos (merge)
- Nyílt (opensource) megvalósítás
- Csábító usecase-ek (bootstrap)
 - Teljesen elosztott Google Now



Overlay Management



- Overlay: virtuális hálózat (gráf) ami mentén az eszközök kommunikálnak
- Peer sampling service: felsőbb rétegek számára szomszédok biztosítása
- Lokális kommunikáció-Globális tulajdonságok biztosítása
 - Newscast: uniform sampling a hálózathoz
 - T-Man: k legközelebbi szomszédsági gráf, mint fixpont



Overlay Management



- Gossip skeleton (Newscast és T-Man):
- View – node halmaz karbantartása
- Aktív viselkedés: véletlen szomszédnak a view elküldése
- Passzív: üzenet fogadása esetén, nézetek uniója és aktív nézet frissítése

Algorithm 2.1 Skeleton of NEWSCAST and T-MAN protocols

```
1:  $n \leftarrow \text{initNeighbors}()$ 
2: loop
3:    $\text{wait}(\Delta)$ 
4:    $p \leftarrow \text{selectPeer}(n)$ 
5:    $\text{send}(n \cup \text{currNode}()) \text{ to } p$ 
6: end loop
7: procedure ONRECEIVENODES( $nodes$ )
8:    $n \leftarrow \text{updateNeighbors}(n \cup nodes)$ 
9: end procedure
```



Overlay Management

- Newscast:

- Descriptor-okban időbélyeg
- Frissítés után a legfrissebek megtartása
- Uniform random véletlen szomszéd választás

→ Folyamatos keverés (uniform) a hálózat “friss” node-jai felett

Algorithm 2.2 The NEWSCAST protocol

```
1: procedure SELECTPEER(nodes)
2:   node  $\leftarrow$  uniRand(nodes)
3:   return node
4: end procedure

5: procedure UPDATENEIGHBORS(nodes)
6:   sn  $\leftarrow$  sort nodes by field ts
7:   return topK(sn)
8: end procedure

9: procedure CURRNODE()
10:  d  $\leftarrow$  initDesc()
11:  d.ts  $\leftarrow$  current time
12:  return d
13: end procedure
```

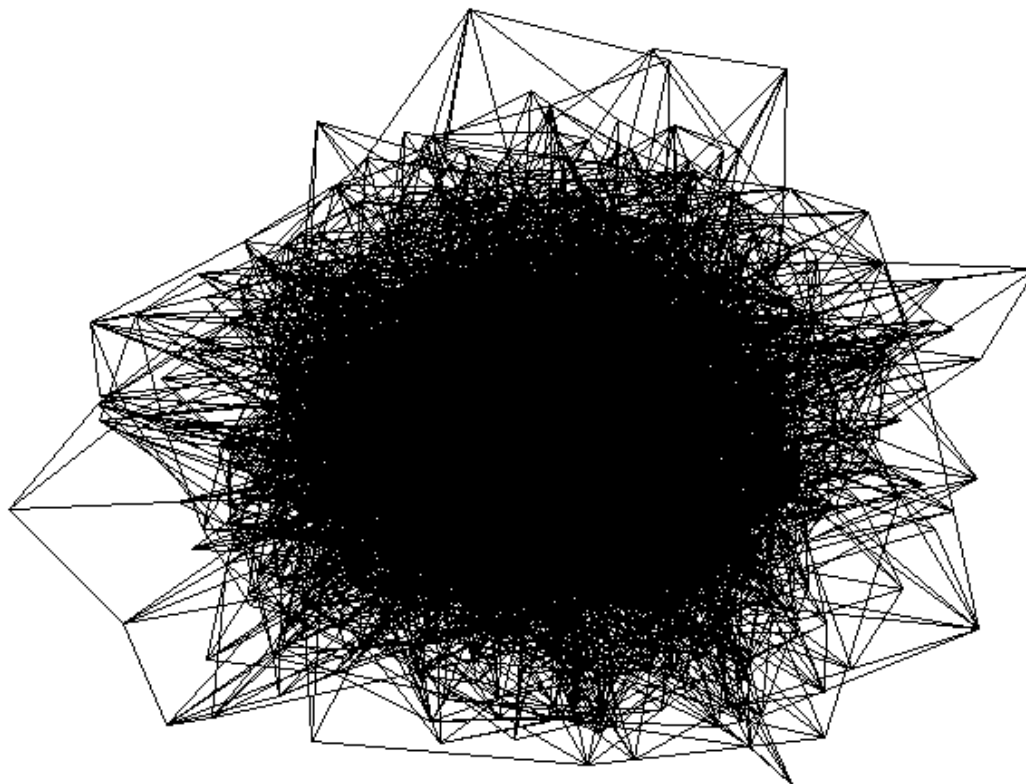
Overlay Management



- T-Man:
 - Nagyon hasonló protokoll
 - Legyen adott egy s : node descriptor x node descriptor $\rightarrow R$ hasonlósági függvény
 - A node descriptorokba helyezzünk el a hasonlóság számításához szükséges plussz információkat
 - Csináljuk ugyanazt mint a Newscast, de a rendezésnél az aktuális node-hoz való hasonlóság szerint csökkenő sorrendet vegyünk!
 - Node választásnál válasszuk a view-ból a hozzánk leghasonlóbb szomszédot
- A hasonlósági gráf lesz a hálózat fixpontja, $\lg(n)$ lépésben!



T-Man - példa



TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

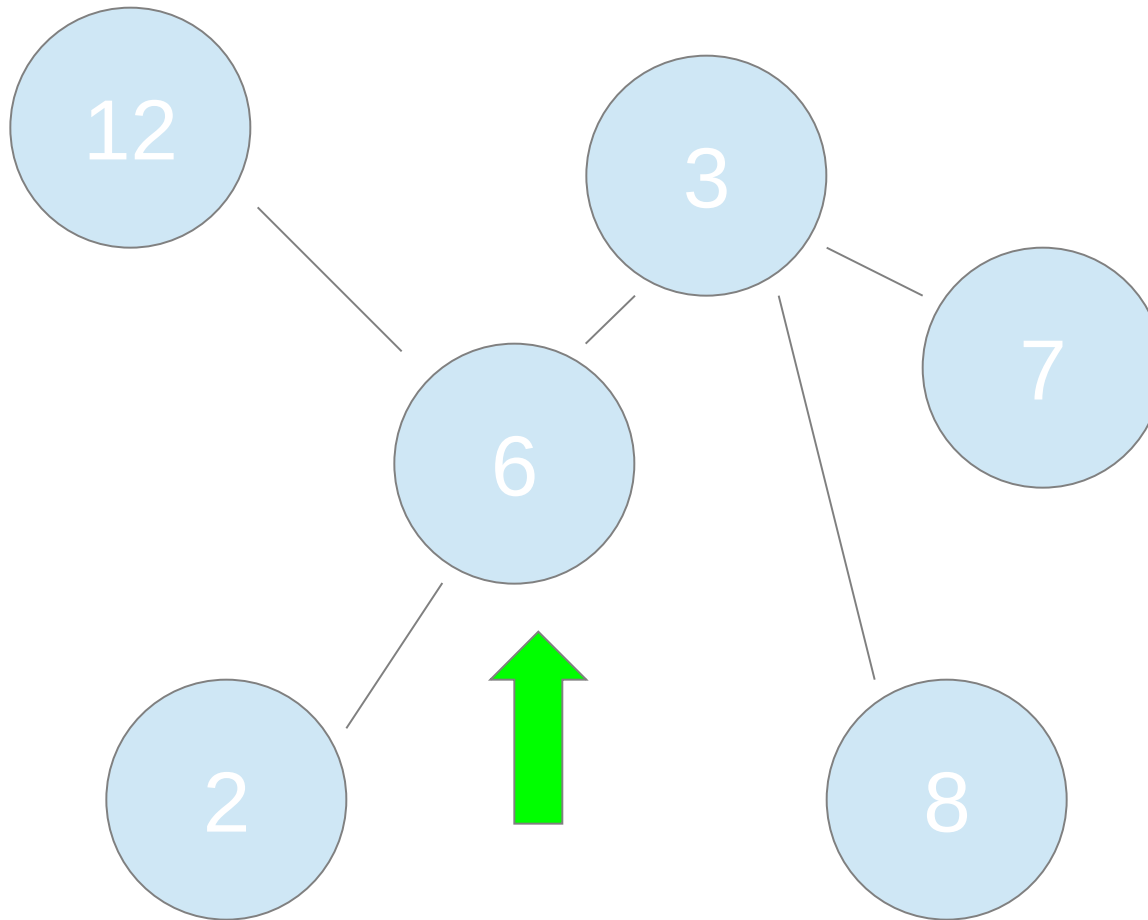
Adatbányászat



- Adatbányászati protokollok futtatása
 - Teljesen elosztott módon
 - Az overlay hálózat fölötti kommunikáció használatával
 - Infrastruktúra mentesen (végtelen skálázódással)
- Rendszermodell:
 - Horizontálisan partícionált a adatok
 - Privát tartalom → Privacy → Feltevés: ne hagyja el az adat az eszközt!



Átlagolás - példa

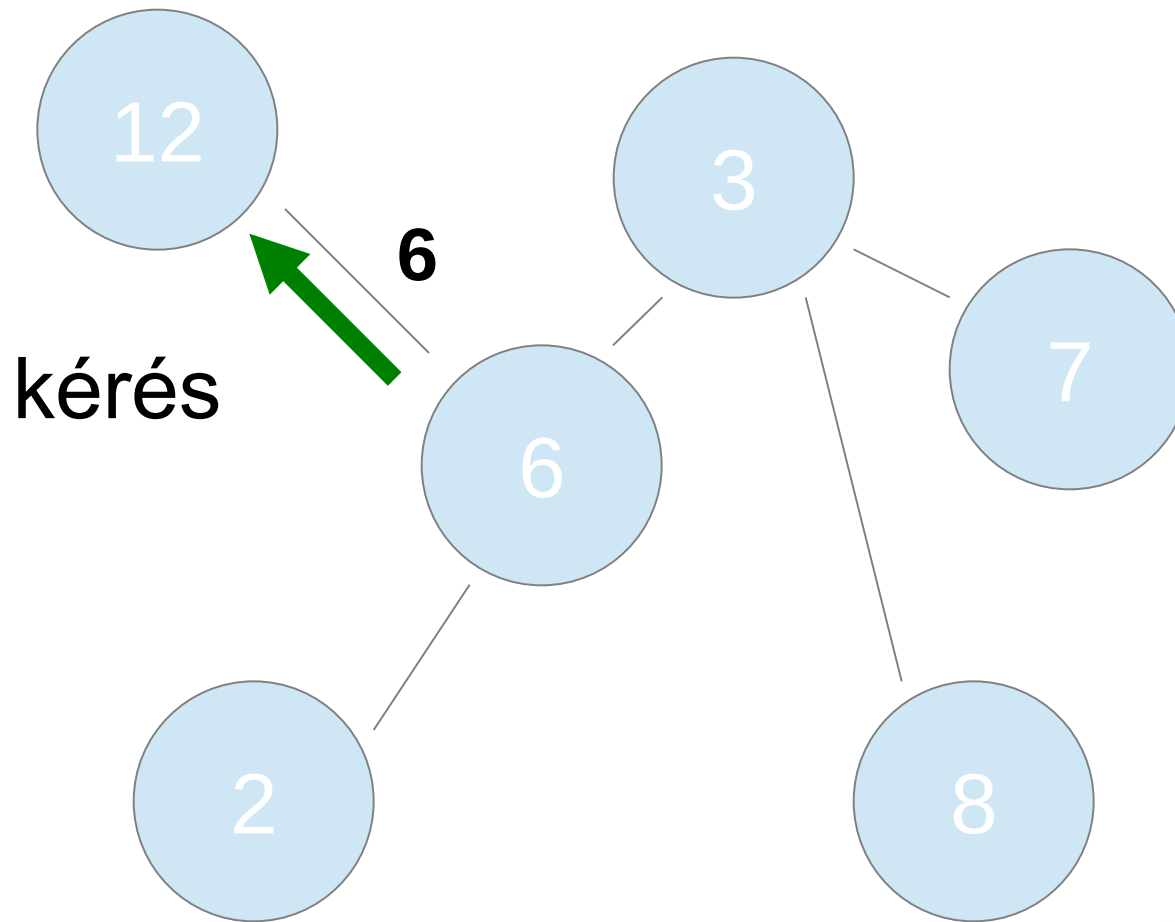


TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

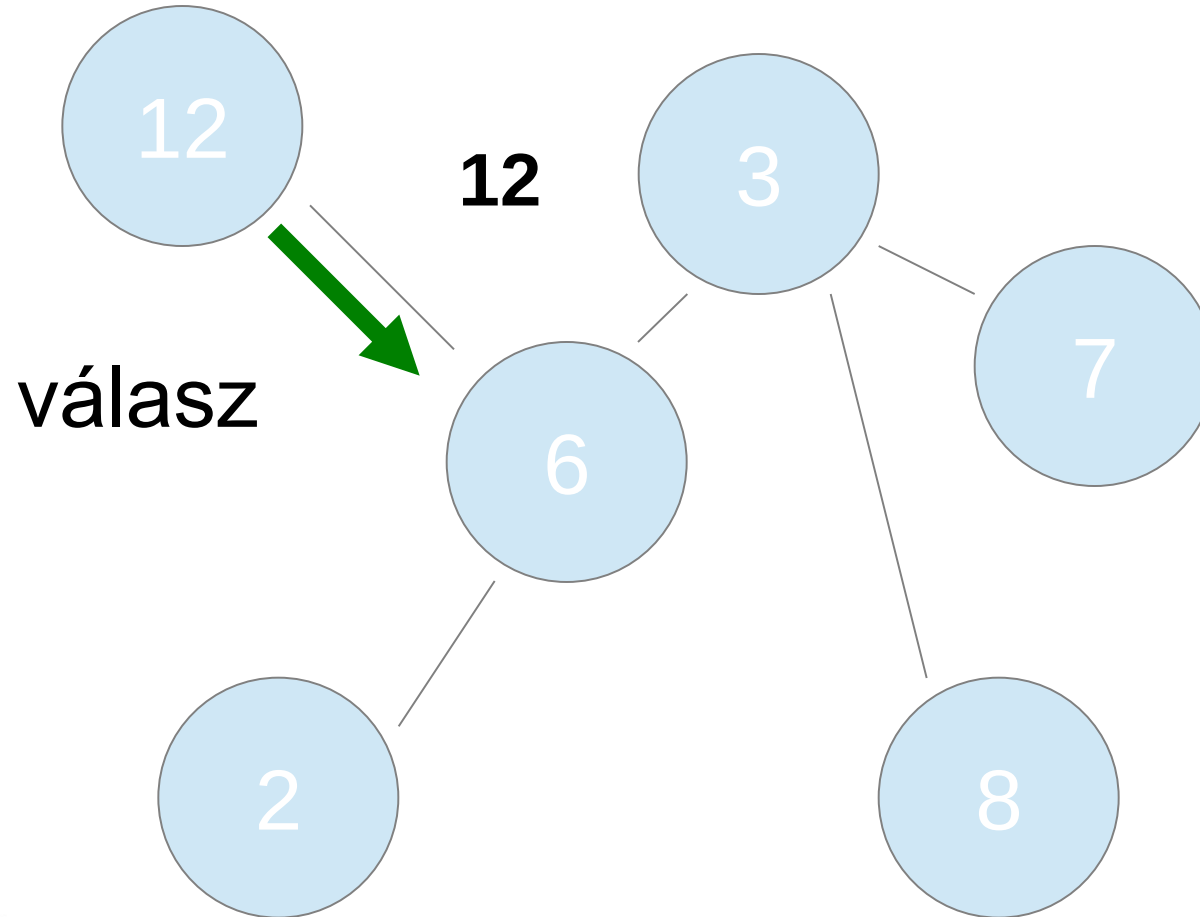
Átlagolás - példa



TÁMOP-4.2.2/B-10/1-2010-0012 projekt



Átlagolás - példa

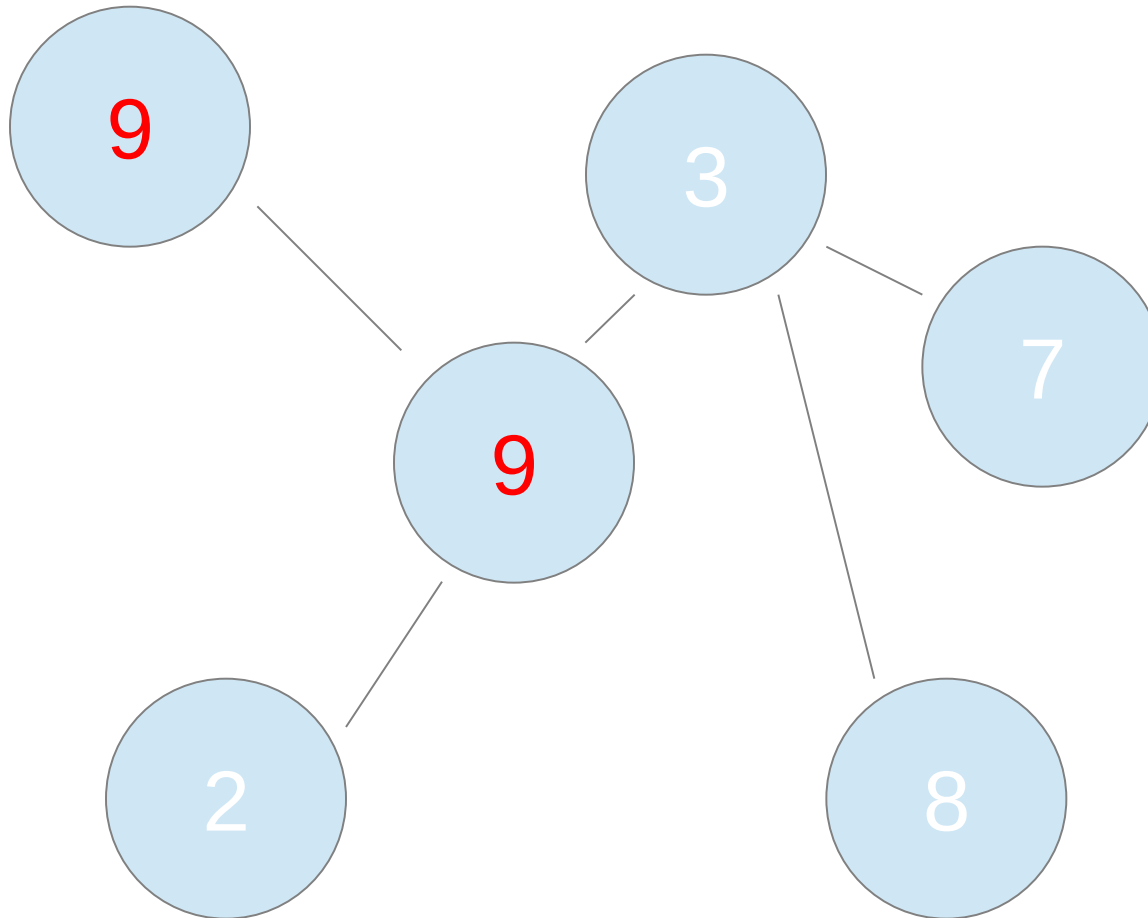


TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

Átlagolás - példa



$$(12+6)/2 = 9$$



TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

Adatbányászat



- A fenti algoritmus egy példa:
 - Nem aszinkron, nem robosztus csomagvesztésre
 - Javítások léteznek, pl. PushSum
 - Szfiztikáltabb számítások is megvalósíthatók:
 - aszinkron,
 - hibatűrő (csomagvesztés, késés, churn) és
 - privacy (adat nem hagyja el az eszközt) módon
- Gossip Learning Framework (GoLF)



GoLF - osztályozás



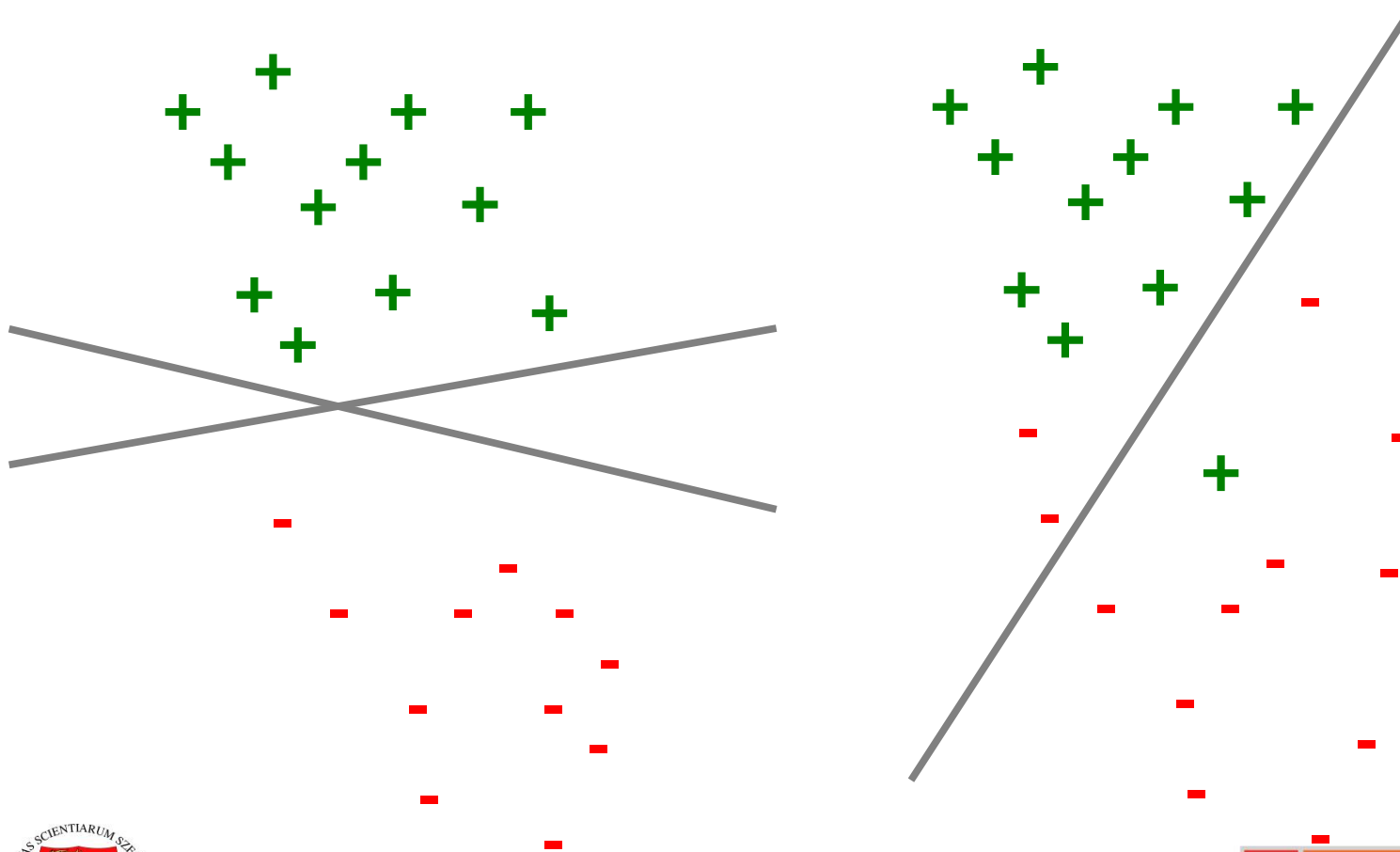
- Felügyelt osztályozási probléma:
 - Adott: (x_i, y_i) párok halmaza a node-okon
 - Cél: építsünk egy f paraméterezett modellt, hogy $f(x)=y$ teljesüljön (nem csak a tanítópéldákra!)
 - Megvalósítás: gyakran valamilyen célfüggvény minimalizálásával:
 - Logisztikus/Lineáris regresszió,
 - ANN,
 - SVM, stb...



GoLF – lineáris oszt.



SZÉCHENYI TERV



TÁMOP-4.2.2/B-10/1-2010-0012 projekt



A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

GoLF - SGD



- Tfh. egy osztályozási hiba adott
- A hiba grádiense a paraméter függvényében
- A teljes grádiens frissítés (teljes adatbázis ismerete szükséges)
- Grádiens lépés egy tanítópéldával (sztochasztikus változat; uniform mintavételezés kritikus)

$$Err(w) = \sum_{i=1}^n Err(w, x_i)$$

$$\frac{\partial Err(w)}{\partial w} = \sum_{i=1}^n \frac{\partial Err(w, x_i)}{\partial w}$$

$$w(t+1) = w(t) - \alpha(t) \sum_{i=1}^n \frac{\partial Err(w, x_i)}{\partial w}$$

$$w(t+1) = w(t) - \alpha(t) \frac{\partial Err(w, x_i)}{\partial w}$$



GoLF



- GoLF:
 - Alkalmazzunk online tanulókat (pl. sztochasztikus gradiens)
 - Amelyek modell (kezdeményei) véletlen sétákat tesznek a hálózatban ($\sim$ független)
 - A csomópontoknál lévő adatokkal frissítik magukat (pl. sztochasztikus gradiens lépés)
 - Lokális vagy globális kombinációs módszerekkel kombinálják a modelleket



GoLF

Algorithm 1 Gossip Learning Scheme

```
1: initModel()
2: loop
3:   wait( $\Delta$ )
4:    $p \leftarrow \text{selectPeer}()$ 
5:   currentModel  $\leftarrow \text{createModel}()$ 
6:   send currentModel to  $p$ 
7: end loop
8:
9: procedure ONRECEIVEMODEL( $m$ )
10:   modelQueue.add( $m$ )
11: end procedure
12:
13: procedure CREATEMODELRW
14:    $m \leftarrow \text{modelQueue.first}()$ 
15:   update( $m$ )
16:   return  $m$ 
17: end procedure
18:
19: procedure CREATEMODEL MU
20:    $m_1 \leftarrow \text{modelQueue.first}()$ 
21:    $m_2 \leftarrow \text{modelQueue.second}()$ 
22:    $m \leftarrow \text{merge}(m_1, m_2)$ 
23:   update( $m$ )
24:   return  $m$ 
25: end procedure
```

GoLF

- Predikció:

- Lokális: egy modell alapján
- Szavaztatott: több modell többségi döntése mentén

```
1: procedure PREDICT( $x$ )
2:    $w \leftarrow currentModel$ 
3:   return sign( $\langle w, x \rangle$ )
4: end procedure

5: procedure VOTEDPREDICT( $x$ )
6:   pRatio  $\leftarrow$  0
7:   for  $m \in modelQueue$  do
8:     if sign( $\langle m.w, x \rangle$ )  $\geq$  0 then
9:       pRatio  $\leftarrow$  pRatio + 1
10:    end if
11:  end for
12:  return sign(pRatio/modelQueue.size()–0.5)
13: end procedure
```

GoLF - Kiértékelés



- Kiértékelés:
 - P2Pegasos – egy SVM tanuló használatával
 - Számos benchmark adatbázison kimérve
 - Több adatbázison
 - Extrém hiba scenario-k felhasználásával
 - 50%-os csomagvesztés
 - 10 ciklus hossznyi üzenet késés
 - Realisztikus churn modell kritikus)



GoLF - Kiértékelés



SZÉCHENYI TERV

	Iris1	Iris2	Iris3	Reuters	SpamBase	Malicious10
Training set size	90	90	90	2000	4140	2155622
Test set size	10	10	10	600	461	240508
Number of features	4	4	4	9947	57	10
Classlabel ratio	50/50	50/50	50/50	1300/1300	1813/2788	792145/1603985
Pegasos 20000 iter.	0	0	0	0.025	0.111	0.080 (0.081)
Pegasos 1000 iter.	0	0	0.4	0.057	0.137	0.095 (0.060)
SVMLight	0	0	0.1	0.027	0.074	0.056 (–)

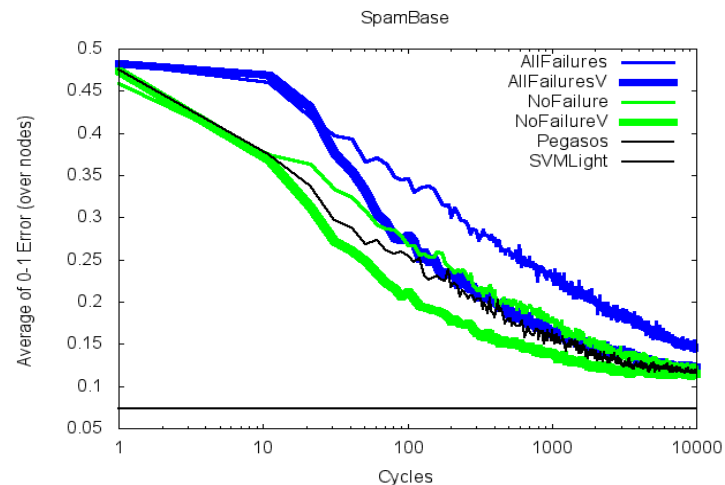
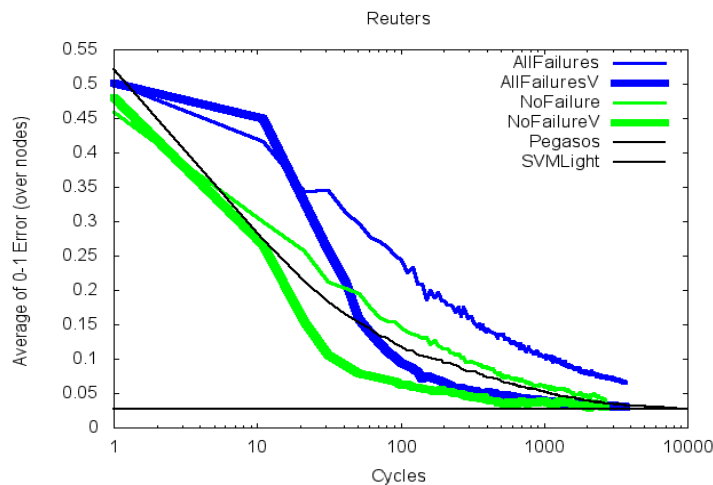
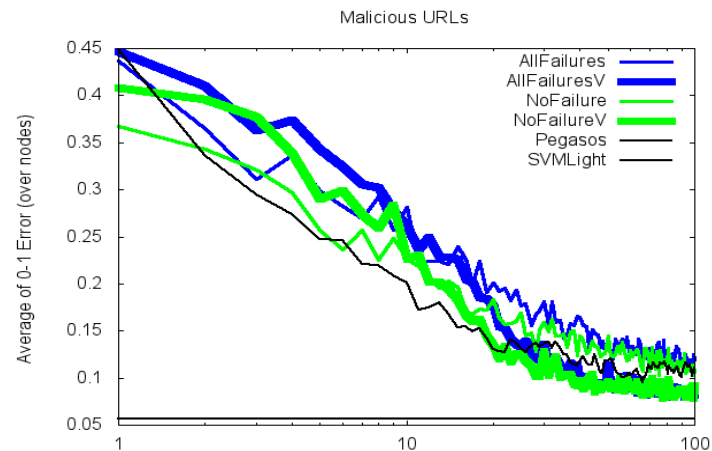
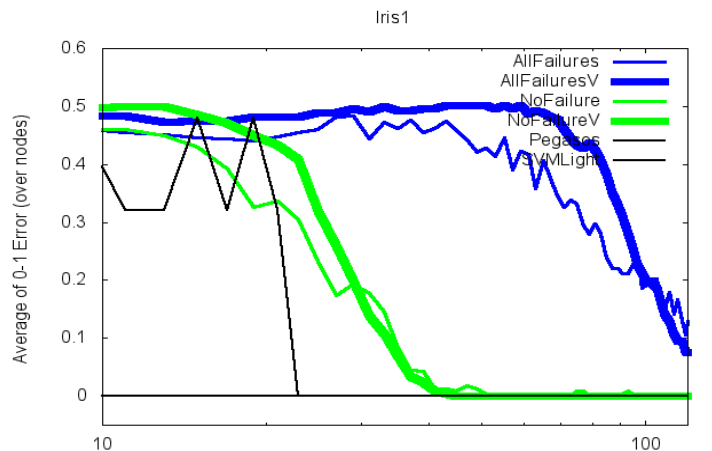


TÁMOP-4.2.2/B-10/1-2010-0012 projekt



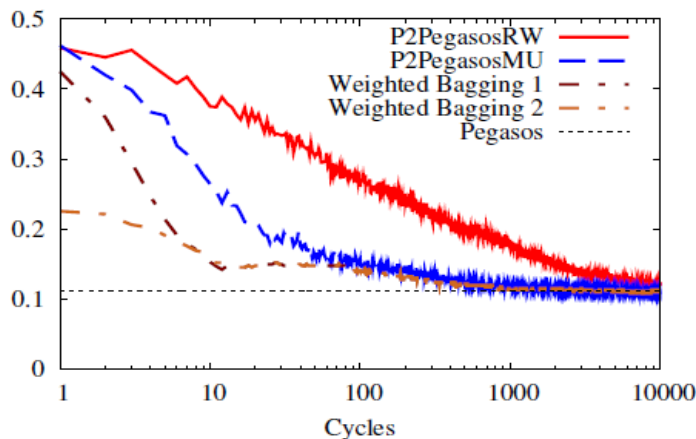
A projekt az Európai Unió támogatásával,
az Európai Szociális Alap
társfinanszírozásával valósul meg.

GoLF - Eredmények

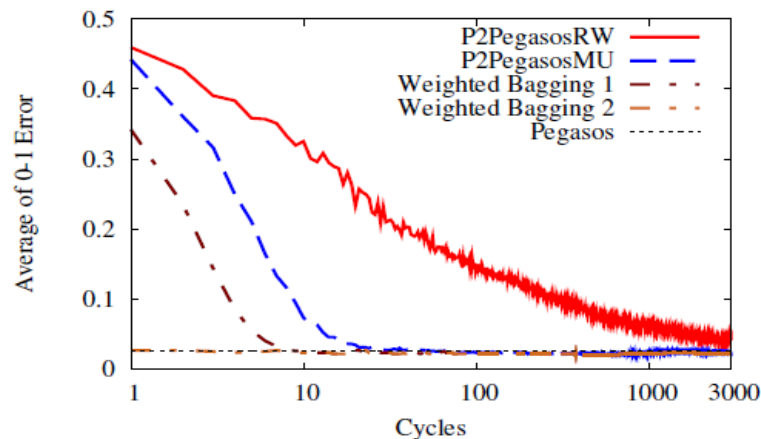


GoLF - Eredmények

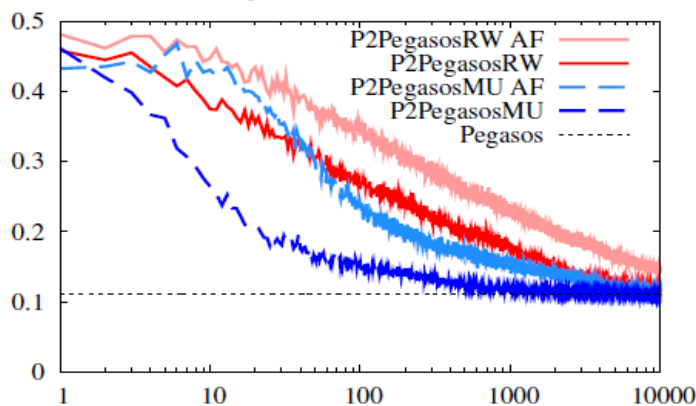
SpamBase No Failure



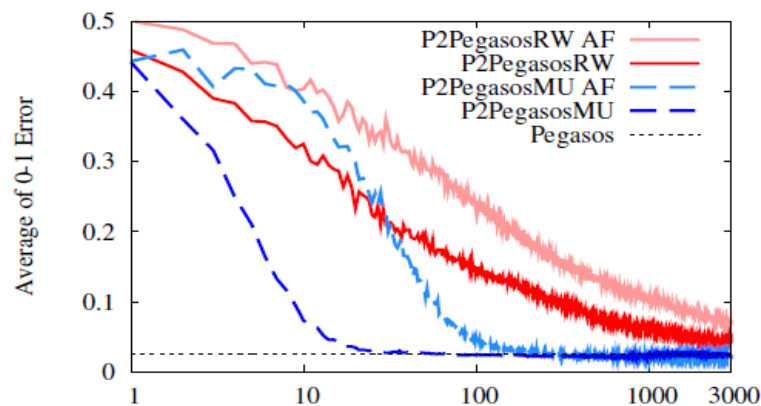
Reuters No Failure



SpamBase with Failures



Reuters with Failures



GoLF – Továbbiak

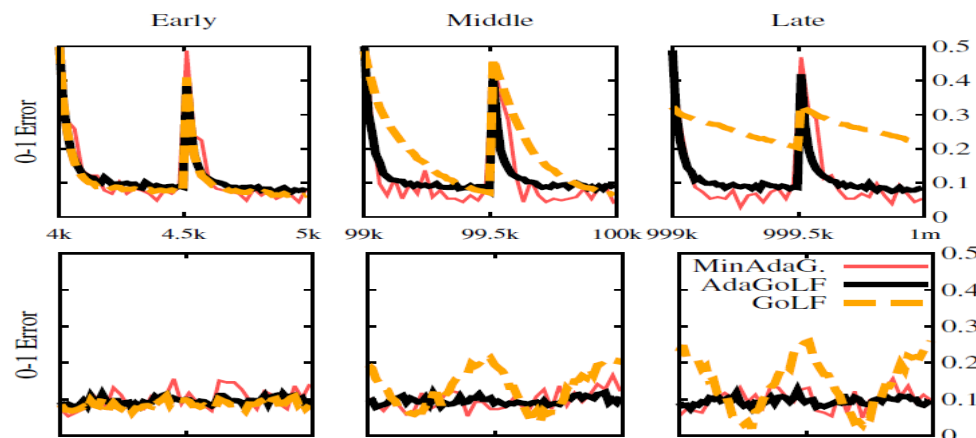


Számos algoritmus megvalósítva:

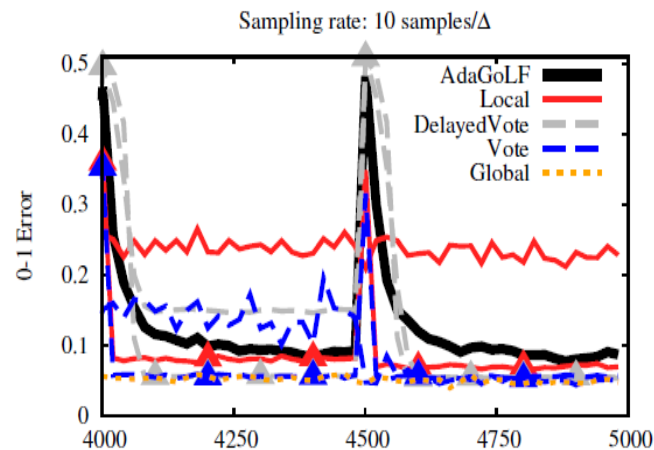
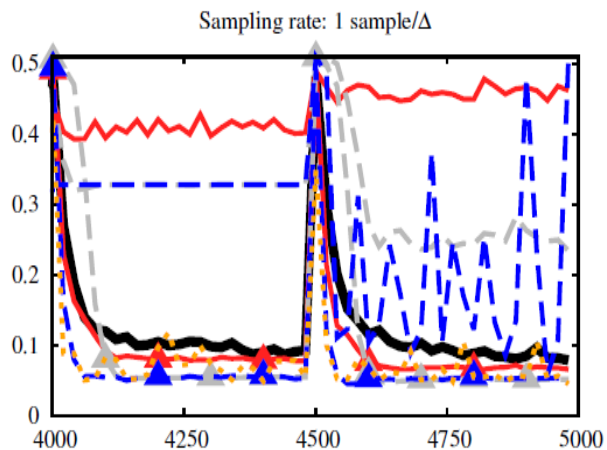
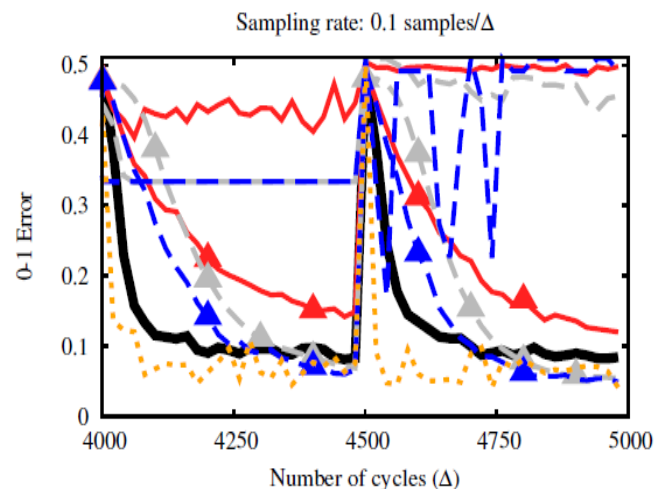
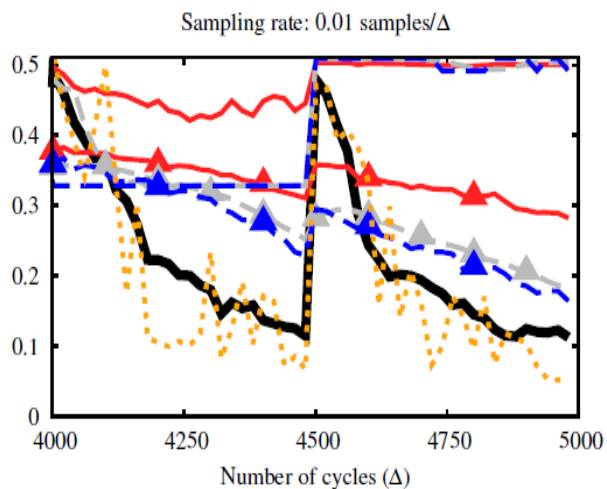
ANN, SVM, LogReg, K-Means, GMM, LowRank Decomp.,
Boosting

Drift detekció kezelésére történő kiterjesztés

Bandita modellekre való alkalmazás



GoLF – Drift kezelés



GoLF - Papers



- Róbert Ormándi, István Hegedűs, and Márk Jelasity. Asynchronous peer-to-peer data mining with stochastic gradient descent. In Emmanuel Jeannot, Raymond Namyst, and Jean Roman, editors, Euro-Par 2011, volume 6852 of Lecture Notes in Computer Science, pages 528–540. Springer-Verlag, 2011.
- Róbert Ormándi, István Hegedűs, and Márk Jelasity. Gossip learning with linear models on fully distributed data. Concurrency and Computation: Practice and Experience, 2012. to appear.
- István Hegedűs, Róbert Busa-Fekete, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Peer-to-peer multi-class boosting. In Euro-Par 2012, volume 7484 of Lecture Notes in Computer Science, pages 389-400. Springer-Verlag, 2012.
- István Hegedűs, Róbert Ormándi, and Márk Jelasity. Gossip-based learning under drifting concepts in fully distributed networks. In Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2012), pages 79–88. IEEE Computer Society, 2012

