

Egy ML algoritmus életciklusa az OTP Bankban

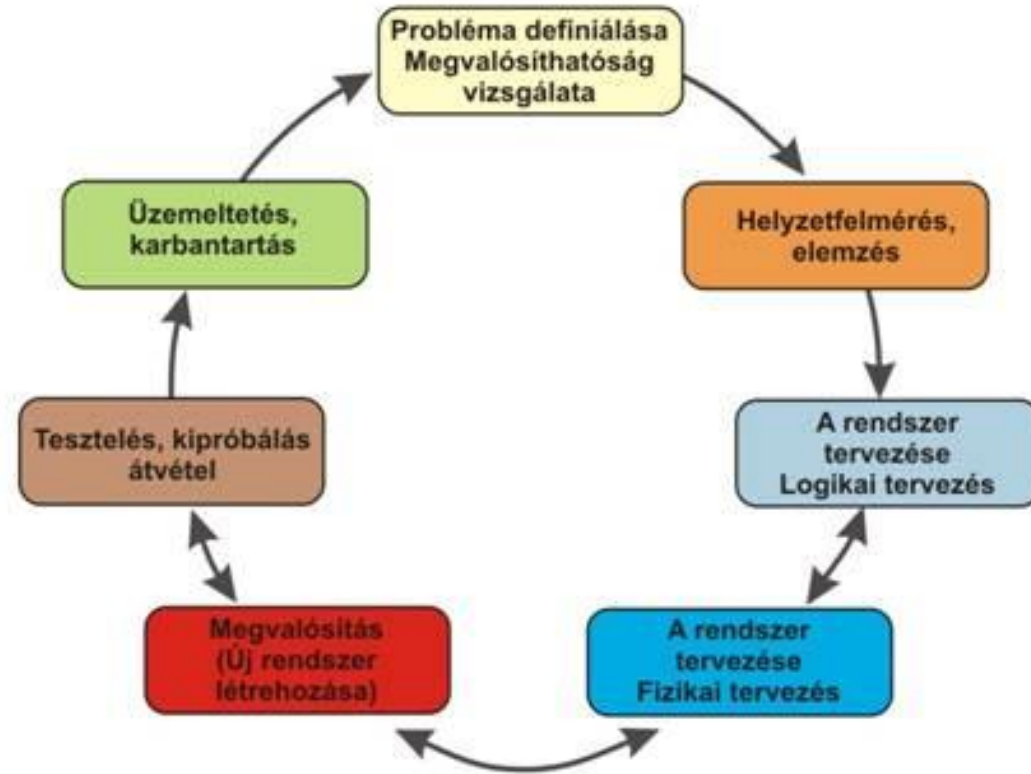
Kuslits Lukács

Data Science tanácsadó, OTP Bank (ML algoritmusok)

2021.04.15.

A modellfejlesztés mint szoftverfejlesztési ciklus

A rendszerfejlesztés életrajza (általános)



Főleg egyszeri megoldások, egyszer használatos elemzések → rendszeres szolgáltatást biztosító, karban tartott, folyamatosan fejlesztett, minőségellenőrzött, adatbiztonsági szempontból is megfelelő ML algoritmusok.

A modellfejlesztés mint szoftverfejlesztési ciklus

Kísérleti fejlesztésektől a nagyvállalatban üzemeltetett megoldásokig:

- Az üzemelés- és bankbiztonság **legalább olyan fontos**, mint a modell pontossága!
- Kööttségek, amikkel a kísérleti jellegű megoldásoknál nem találkozunk

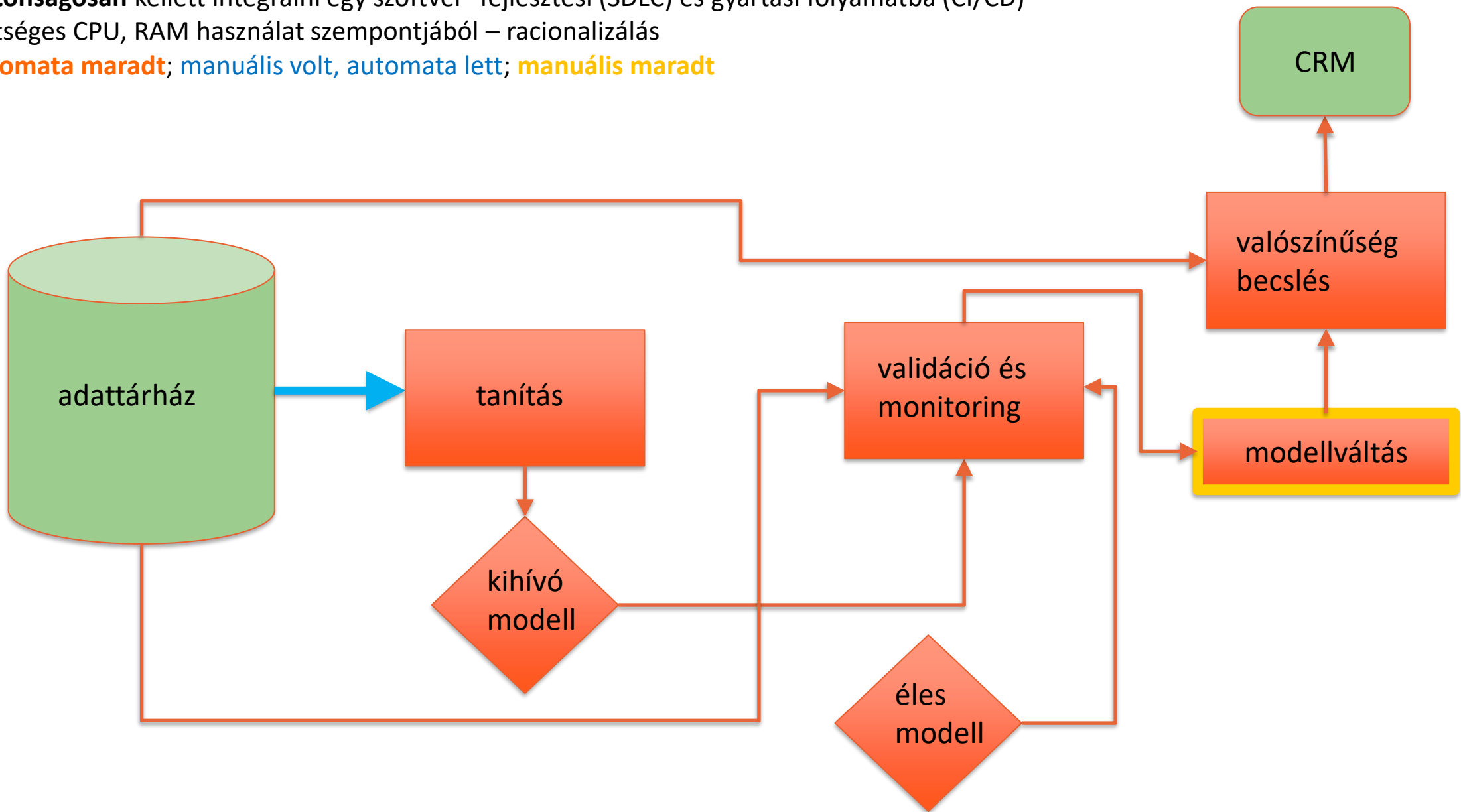
Egy ML modell fejlesztési, üzemelési életciklusának sajátosságai:

- Nyílt forráskód → biztonság?
- Folyamatos rendszerkapcsolatok, jelentős számítás igény
- változnak az adatok → változnia kell a modellnek is
 - magyarázó és magyarázott adatok, az ezek közti a kapcsolatok folyamatosan vagy akár hirtelen megváltozhatnak (*pandémia: a fióki készpénzfelvételek száma 25%-kal csökkent**)
- Az ML modellek fejlesztői: data scientist-ek, adatelemzők, általában NEM INFORMATIKUSOK (mély rendszerintegrációs ismeretek kellenek, data engineer, DevOps szakember kompetenciája)

* <https://www.vg.hu/penzugy/penzugyi-szolgaltatok/felgyorsitotta-a-jarvany-a-penzforgalom-digitalizaciojat-3539953/>

Ajánló modell

- általános ajánlómotor, termékaffinitási valószínűségeket kalkulál
- több tucat táblából használ fel adatokat
- **biztonságosan** kellett integrálni egy szoftver- fejlesztési (SDLC) és gyártási folyamatba (CI/CD)
- költséges CPU, RAM használat szempontjából – racionalizálás
- **automata maradt**; manuális volt, automata lett; **manuális maradt**



Modell fejlesztési ciklus lépései nagy vonalakban

/és azok megvalósítása/

Prototyping	Development	Production
Notebook IDE Vizuális programozás Kísérletezés	CI/CD: /központi build infrastruktúrán/ Model Serving: /in-house, NFS/ Data Pipeline: /ábra, dokumentáció/ Data Governance: /sql, python testing/ Testing: /python unit testing, python integration testing/ Kódminőség ellenőrzés: /linter (static code analysis), quality gate/	Workflow: /in-house/ ML platform: /K8s alapú/ Monitoring: /dashboard/ Biztonsági adatkezelés: /secret management, logkezelő rendszer/

Összefoglalás

- Éles üzleti folyamatokat nem lehet ad-hoc modellekkel kiszolgálni!
- A hagyományos szoftverfejlesztési életciklushoz képest kihívást jelent, hogy open source algoritmusoknak kell stabilan, bankbiztonsági szempontoknak megfelelően üzemelni.
- **A nem ad-hoc modellek biztonsági előírásoknak kell megfeleljenek, integrálni kell őket más alkalmazásokkal, kód szintű strukturális átalakítás szükséges.**
- Magas szintű automatizációt kell biztosítani a modellezési, tesztelési folyamatban.
- **Az SDLC folyamatot fel kell készíteni arra, hogy a megváltozott adatok mindent megváltoztathatnak.**
- **Sok résztvevő közreműködésére van szükség!**

Köszönöm a figyelmet!

Abstract green geometric shapes, including a large quarter-circle, a smaller semi-circle, and a circle, are positioned in the lower-left and center of the slide.

Kérdések?

Abstract green geometric shapes, including a large quarter-circle, a smaller semi-circle, and a circle, are positioned in the lower-left and bottom-center areas of the slide.