# Gyanúgenerálási rendszerterv banki tranzakciós adatbázisok kapcsán

Szerkesztett: Pitlik László, 2011. december – 2012. január

Társszerzők: Bruder Emese, Domján Csaba, Ruff Ferenc, Szűcs Imre

## fejezet: Időbeli, aggregált riasztások

## Előzmények

A gyanúgenerálás (<http://www.google.hu/search?q=itbszm>) a hasonlóságelemzés egyik speciális ága (<http://miau.gau.hu/miau2009/index.php3?x=e0&string=susp>). Ennek lényege a nem deklaratív gyanú fogalmának bevezetése (<http://miau.gau.hu/miau/153/kdrik_2011.doc>) többrétegű, anti-diszkriminációs (<http://miau.gau.hu/myx-free/index.php3?x=e091>) számításokra alapozva. A nem deklartív gyanú esetén nem kell pontos riasztási küszöbértékeket megadni. A gyanúmomentumok a korábbi élethelyzeteket leíró attribútumokra megfogalmazott irányultságok (pl. minél kevesebb a tranzakciók száma, annál kevesebb az esély a visszaélésre – hiszen nulla tranzakció esetén nem létezhet visszaélés) mentén, egymással versengő erőtereket felépítve keletkeznek optimalizációs eljárások felhasználásával. A gyanúgenerálás eredménye egy relatív skálán kifejezett gyanúmérték, mely hozzárendelésre kerül minden egyes vizsgált időintervallumhoz. Az önellenőrző számítások alapján a gyanúmértékek lehetnek hitelesek és hiteltelenek. Az emberi döntéshozó számára a hitelesen magas gyanúmértékek és a hiteltelen esetek jelentenek további feladatot. A gyanúgenerálás, s az erre épülő riasztás és az adekvát operatív akciók kivitelezése automatizálható.

## Az adatvagyonról

A banki tranzakciók kapcsán a naplóállományokból számos karakterisztika kiolvasható:

* honnan (pénzintézeti azonosító, országkód)
* hová (pénzintézeti azonosító, országkód)
* mikor
* milyen csatornán
* milyen összeg
* milyen pénznemben
* stb. vándorolt.

Mivel egyedi tranzakciókra érvényes gyanúgenerálás (hasonlóan a pontszerű becsléshez) nem lehetséges, így aggregált időszakok (pl. órák) adatait, mint objektumokat (vö. mátrixsorok) kell feltölteni olyan attribútumokkal, melyekre vonatkozóan gyanúsítási irányok fogalmazhatók meg: pl.

* minél kevesebb a tranzakciók száma az adott időszakban, annál kisebb az esélye annak, hogy ebben a halmazba gyanús elem keveredett
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy darabszám (count)
  + technikai alábontás: pénznemenként és/vagy tranzakciós csatornánként és/vagy deklarált (jogszabályilag érzékeny) értékintervallumonként
* minél kisebb a tranzakciók volumene annál kisebb az esély, hogy értelme legyen gyanús tranzakciót katalizálni
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy összeg (sum)
  + technikai alábontás: pénznemenként és/vagy tranzakciós csatornánként
* minél kevesebb a pénzintézetek száma, annál kisebb az esélye, hogy valamelyikben nem tervezett tranzakció történik
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy darabszám (count)
  + technikai alábontás: pénznemenként és/vagy tranzakciós csatornánként és/vagy deklarált (jogszabályilag érzékeny) értékintervallumonként
* minél kevesebb az érintett pénznemek száma, annál kisebb az esélye bonyolult álcázási manővereknek
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy darabszám (count)
  + technikai alábontás: tranzakciós csatornánként és/vagy deklarált (jogszabályilag érzékeny) értékintervallumonként
* minél homogénebb a tranzakciók összegszerűsége, azaz minél kisebb a szórása, annál kevesebb az esélye speciális tartalmú manipulációnak
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy vektor, mely szórására van szükség
  + technikai alábontás: pénznemenként és/vagy tranzakciós csatornánként
* minél kevesebb az érintett tranzakciós csatornák száma, annál kisebb az esélye speciális manipulációk előfordulásának
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy darabszám (count)
  + technikai alábontás: pénznemenként és/vagy tranzakciós csatornánként és/vagy deklarált (jogszabályilag érzékeny) értékintervallumonként
* minél gyakoribbak az egyes honnan-hová párosítások, annál kisebb az esélye speciális manipulációnak
  + kivitelezés: honnan-hová kombinatorikai terek megalkotása
    - országok között
    - pénzintézetek között
  + lekérdezés-sorozat (select distinct),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége, ill. honnan-hová alakzatok egyedi vagy kombinált paraméterei,
  + ahol az idő tetszőlegesen görgethető, ill.
  + a lekérdezés eredményei kombinációnkénti darabszámok (count)
  + feldolgozás: a leggyakoribb kombinációk hibapontértéke nulla, a legritkábbé pl. 100, a többié előfordulással fordított arányban
* minél nagyobb a homogén tranzakciók közötti időeltérés (szórás), annál kisebb az esélye annak, hogy a sok hasonló adat között gyanús bújjon meg
  + homogén tranzakció: azonos pénznem, honnan-hová koordináták, értéknagyság-intervallum,
  + kivitelezés: lekérdezés (select),
  + ahol a szűrési feltétel (where) egy időintervallum eleje/vége,
  + mely tetszőlegesen görgethető, ill.
  + a lekérdezés eredménye egy vektor időkoordinátákkal, melyek szórása kell

A fentiek mintájára tetszőleges további mutatószámok alakítható ki. A fentebb felsorolt egyszerűbb/komplexebb mutatószámok generálására a hardverkapacitások függvényében óránként minimum egyszer, de számítókapacitás függvényében X perces eltolással van szükség. Az adott időintervallumhoz tartozó mutatószámok egy objektum – attribútum - mátrix (OAM) alakját veszik fel automatikusan: ahol a sorok az egyes órák (folyamatosan bővülő sorszámmal), az oszlopok a számított karakterisztikák, az értékek a fentebb jelzett darabszámok, szórások, összegek, stb.

A primer OAM és a (minél/annál-jellegű) irányok alapján a teljes OAM oszloponként sorszámokká (rank) konvertálandó (vö. standardizálás). Ezen sorszámmátrix minden további lépés (black box) inputja.

Az elemzés outputja egy hitelességi státusz és egy gyanúmérték. A hitelességi státusz egymáshoz képest inverz irányú inputok elemzés utáni szimmetria sértéseit detektálja, vagyis az a hiteles gyanú, mely megfelel a tagadás tagadása elvnek. A gyanú mértéke az egyszerre vizsgált objektumok 100 gyanúpont körül megoszlását jelenti: a kisebb értékek a kevésbé, a nagyobb értékek az erőteljesebben gyanús időintervallumokat jelzik.

A gyanúk időbeli dinamikája (vagyis egyre növekvő riasztási szintek egymást követő időintervallumonként) stabilizálhatja a riasztás szükségszerűségét.

## Specialitások

Külön (de a fentiekhez hasonló) elemzési logika alkotható egyes aktőrök magatartásának (speciális tranzakciócsoportok, mint objektumok) figyelésére, melyben a relatív gyanú nehezen „hack”-elhető logikájának követési kísérletei kerülhetnek relatív gyanúként megfogalmazásra…

Az inverz-ellenőrzésekkel hitelesített anti-diszkriminációs számítások mellett minden egyes mutatószám kapcsán norma-értékek kalkulálhatók, melytől a tények eltérései szintén riasztási alapot jelenthetnek (pl. sokkal kevesebb tranzakciónak illett volna lennie, vagy inkább nagyobb arányú forint tranzakciók lettek volna logikusak, stb.).

A gyanúképződés dinamikája, mint addicionális feladat előrejelzésekhez is alkalmas szekunder adatbázis termel ki, mely alapján a gyanú jövőbeli várható alakulása is vélelmezhető…

## Összefoglalás

A nem deklaratív tanulásnak nincs érdemi alternatívája:

* a WizWhy típusú elemzés nem képes relatív gyanú generálására, csak a kombinatorikai tér ismert szegmenseihez tartozó gyakoriságok értelmezésére, vagyis nem képes a gyanú genetikai potenciáljának meghatározására, szimulációs jellegű értelmezésére (<http://miau.gau.hu/miau/150/la150.docx>), a szabályerők esetén nincs érdemi mód önellenőrzésre,
* a matematikai statisztikai megközelítések egyike sem önellenőrző, s ezek nem ismerik az optimalizációs tanulási folyamatokat,
* a neurális hálózatok ismert gyanúk felismerésére taníthatók be, s nem képesek a plátói (gyanú) ideál kezelésére.

## fejezet: Tranzakció-szintű gyanúgenerálás

## Előzmények

A nem rekordszintű, hanem aggregált gyanúgenerálását élesen el kell választani a tranzakciószinten megfogalmazódó elvárásoktól (<http://miau.gau.hu/miau2009/index.php3?x=e0&string=susp>). A tranzakciószintű gyanúgenerálást más szóval „outlier detection”-ként is fel lehet fogni (vö. <http://miau.gau.hu/miau/162/outliers.docx>), mely felveti azonnal a szóba jöhető módszerek összehasonlítását is (l. 3. fejezet). Emellett felvetődik a tranzakciószinten elvárt gyanúk esetén a context free és a naplófájlok tartalmából kiinduló gyanúgenerálás lehetősége is.

### Szakértői szempontok

A potenciális módszerek összevetése kapcsán fontos szempontok:

* igényel-e egy-egy eljárás következményváltozót ahhoz, hogy gyanút tudjon megfogalmazni? (ill. képes-e futni egy-egy eljárás így is, úgy is?)
* kell-e pozitív (és negatív) tanulási minta a gyanúk felismeréséhez?
* skálázható-e egy-egy eljárás tetszőlegesen, azaz tetszőlegesen sok adat esetén is hatékonyan futtatható?
* milyen szerepet tölt be a futtató az elemzésben? (vö. deklaratív szabályok?, plátói ideák? utólagos értelmezés/belemagyarázás?)
* mennyi inputváltozó dolgozható fel egy-egy eljárással?
* milyen skálán ábrázolt inputok dolgozhatók fel egy-egy eljárással (pl. képes-e egy-egy eljárás minden változót kezelni akkor is, ha ezek nominális skálaként kerültek egytől egyig leképezésre?)
* van-e önellenőrző (konzisztencia-) rétege az elemzéseknek? s ha igen, akkor mennyi a nem hiteles adatok aránya?
* képes-e egy-egy eljárás real time futni?
* van-e a real time futásnak inicializálási előfeltétele vagy enélkül is real-time egy-egy elemzési folyamat?
* a context free és a context dependent, ill. az aggregált és a tranzakciószintű gyanúk mutatószámai milyen mértékben térnek el egymástól eljárásonként (pl. szórás csak többelemű halmazok esetén számolható)?
* az aggregált gyanú átváltható-e egyedi gyanúvá (pl. időszakok határainak eltolása, zoom-olás?)
* ha egy eljárás többféle beállítással is képes eredményt adni, ezen eredmények hibridizálását (összevonását) kizárja-e bármi is?
* hálózatelemzésre alkalmas adatvagyonok kialakíthatók-e bármilyen adatvagyonból, ill. fordítva?
* segíti-e a context free gyanúgenerálás új tartalomfüggő gyanúk logikájának felismerését?
* milyen mértékben párhuzamosíthatók egy-egy eljárás számítási lépései?

### Téves riasztás?

A gyanúgenerálás lényege természetesen az egyébként más módon, utólag beigazolódott gyanúk előzetes felismerni tudása. Emellett azonban nem számít az sem téves riasztásnak, ha olyan gyanúk esélyét veti fel egy-egy eljárás, melyek jogi értelemben nem bizonyíthatók be adott esetben (vö. intuitív szakértő vélemények a gyanú-profilképzésben). A módszerek által feltárható gyanúk nem lehetnek kevesebben és nem lehetnek mások, mint amit egy intuitív emberi agy képes feltárni, vagyis a módszereknek képesnek kell lenniük mindenféle anomália-fogalom leképezésére (vö. hasonlóságelemzés = intuíciógenerálás).

### Gyanúfokozatok

A hasonlóságelemzés ilyen formában az alábbi (sajnos nem feltétlenül kihagyás- és átfedés-mentes felosztást biztosító) skálán valahol középtájon helyezkedik el:

* deklaratív szabályok (SQL-vetülete, törvényi, szakértői előírások alapján)
* nagyon "tuti" beválású, egyedi fejlesztésű előrejelző modellek
* hasonlóságelemzés: az emberi intelligencia támogatása, kiegészítése, pótlása (inkl. tévedés joga, megértés, meg nem értés felismerése mellett)
* klasszikus adatbányászati megoldások (ismert gyanúminták reprodukálása)
* outlier detection minden vetülete
* egyszerű plauzibilitás tesztek
* RND tranzakció-választás szúrópróbaszerű esetvizsgálatokhoz...

### A context-free gyanúgenerálás logikai keretei

* a naplóállományok változói esetén ezek distinct (egyedi) értékeinek listáját és az ezek mögött megfigyelhető tranzakciók számosságát illik ismerni
* a nagyon egyedi distinct értékek (pl. egyedi adathiány) önmagukban deklaratív gyanúk lehetőségét vetik fel, hiszen pl. miért hiányozna egy dátum, vagy channel információ?, hacsak nem egy hacker-aktivitás figyelmetlen tartalommanipulációja kapcsán?
* a kis gyakorisággal előforduló anomáliáknak bármi lehet az oka, tehát a riasztás tartalmi megalapozottsága elsődlegesen nem vizsgálható: ha emberi szakértők utólag, számos hasonló matematikai, statisztikai (kombinatorikai) okból megtörtént riasztás után, melyek nem vezettek valós rendkívüli eseményhez és nem is sikerült értelmet lelni a tiszta matematikai formulák mögött, jobb, ha letiltják ezen elvi szinten létezhető, de az üzleti modell szempontjából nem releváns, potenciális gyanúrétegeket...
* amennyiben bármilyen apró rendszertelenség (transzformációs anomália) kerül egy distinct vezérelt gyanúgenerálásba, akkor tömegesen alakulhatnak ki téves riasztások (pl. téves elhatároló jelek/delimiterek megjelenése).
* a hasonlóságelemzés legegyszerűbb (de nominális skálán nem alkalmazható) céltalanság-orientált verziója nem mást, mintha minden egyes kombinatorikailag létezhető irányvektort (2^n, ahol n = a változók számával) rákényszerítünk a teljes adatvagyonra, majd keressük az inverz alakok bevonása mellett a hitelesen nem magyarázható rekordokat…
* a tartalom-független gyanúgenerálás nem zárja ki, hogy olyan részhalmazokra kerüljön egy-egy algoritmus futtatásra, melyek valamilyen közös (nem feltétlenül szakmailag interpretálható) feltételnek megfelelnek (pl. distinct értékek szerinti csoportok).
* amennyiben a rekordszám minimalizálása cél kell, hogy legyen pl. a real-time jelleggel erősítése miatt, úgy a kis pénzértéket érintő rekordokat érdemes kizárni a vizsgálatból, hiszen ha a vizsgálat (ellenőrzés, gyanúgenerálás futásának gépideje) drágább,mint egy potenciális kártérítés, akkor nem célszerű az ellenőrzés elvégzése…
* a tartalom-független gyanúk felismerése esetén az alapszabály a következő: ami egyedi, egyszer, ritkán előforduló paraméterkombináció (vö. azaz nem kis számosságú distinct-csoport), az alapvetően nem lehet gyanús, mert az egyediség oka bármi lehet (vö. multi-kulturalitás, vagy mindenki másként egyforma)…ill. a tömegesen hasonló rekordok azért sem lehetnek gyanúsak, mert tömeges csalás nem létezhet, hiszen ez a rendszer összeomlását kell, hogy jelentse…
* azonban az, ami adott mutatószámok kapcsán egyedi, n-dimenziós paraméterkombináció, az újszerű (alapvetően nem nominális skálán ábrázolt alapadatok esetén képezhető) mutatószámok képzése esetén már nem feltétlenül marad egyedi (pl. a bankkártya fennmaradó egy érvényességi hónapjaival szorzott átutalási/pénzfelvételi összeg)…
* ha egy egyediség sok-paraméteres térben adott, akkor egy szűkített dimenziószám esetén már nem kényszerű ennek fennmaradása…
* egy-egy rekord többféle csoportba is tartozhat, vagyis a részhalmaz-specifikus gyanúk n-rétegben keletkezhetnek…

### Kombinatorika LP-alapon?!

**CSAK nominális skálán ábrázolt változókra érvényes, tartalom-független gyanúgenerálás hasonlóságelemzéssel:**

0. Bármely ismert LOG-mérőszám esetén keresni kell a rel. PICI csoportokat (vö. alacsony distinct-értékek, ami a demo-ban jelenleg a kért 94 rekordot jelenti, ahol mkb\_auth\_\_card\_status = 002. – vö. XLS). Minden rekord bele kell, hogy essen min. 1 darab ilyen csoportba, vagyis a csoportok méretét az határozza meg, hogy minden rekord tartozik-e már valahová. EZ egy előzetes = inicializálási fázis, vagyis ez még nem real-time kihívás...

1. Ha megvannak az elemzésre kijelölt alcsoportok (melyek formálisan lehetnének véletlenül választott részhalmazok, vagy idősávonként kijelöltek, vagy értéknagyság szerint leválogatottak), akkor csoportonként a csoportdistinct-listák alapján minden egyedileg előforduló változó minden egyes önálló értékét 1-től n-ig el kell látni pozitív egész belső kódszámmal, azaz lényegében úgy kell tekinteni minden adatot, mint egy nominális skálán ábrázolt értéket (vö. színek). Akkor is így kell eljárni, ha egyébként az adat rangsor, vagy magasabb szintű skálán ábrázolt tartalommal bír (vö. érték, dátum). Ha a múltban minden fontos distinct-opció előfordult, akkor utána már mehet real-time elemzés is, de ha nem, akkor az inicializálás és a tanulási folyamat-sorok elölről indítandók...

2. Ha megvan az 1...n kódolás minden attribútumra, akkor megvan egyben a tanulási minta is. Ezen a tanulási mintán annyi (+2) elemzést kell elvégezni, ahány mező/változó/mérőszám van benne. A +1 elemzés esetén a következményváltozó monoton, a +2 elemzés esetén a következményváltozó az idő maga (vagyis keressük, hogy a korábbi időszakok jellegzetességei alapján eltelhetett-e annyi idő, amennyi eltlet, vagy fordítva: ennyi idő elteltével ilyen kombinatorikai változás = újszerűség nem logikus, rendszeridegen)... A többi esetben (x\_i=y\_i) minden egyes változó egyszer a többi alapján magyarázandó (Y). A modell egy-egy MCM modell, azaz nincs még lépcső-kritérium (restrikció) sem az LP-kben. Ez nem más, mint nominális skálára adaptált kombinatorika LP-alapon. Az egész rendszer pedig egy automatikus SWOT, ahol az idő és a változók gyanúhatásainak összevetése az önellenőrző réteg, ill. a distinct részhalmazok egymásra hatása a gyanú multiplikátora vagy elnyomó effektusa…

3. Az elemzések az elvárásoknak megfelelően zömmel nulla gyanúmomentummal térnek vissza. A gyanúmomentumokat tartalmazó ábrán célszerű logaritmikus skálát használni a vizuális könnyebbség érdekében.

4. OUTPUTS = minden egyes "pici" csoport egyedi gyanútérképe, ill. a gyanútérképek tranzakciónkénti egybevetése... A végső riasztási mértéket a megrendelő állíthatja be... (vö. szakértői rendszer).

5. Business modell = az inicializálás és betanulás után a következő tranzakciók mindaddig a megtanult paramétertömbök (quasi lépcsős-függvények, helyesebb „kombinatorikai polinomok”) alapján számítható (= robotszakértői rendszer), amíg egy vadonatúj distinct érték egy tranzakcióban meg nem jelenik... Ekkor erre a többletmintázatra a tanulás egy részét újra kell futtatni... Akkor is új tanulás szükséges egy-egy érintett csoportban, ha adott tranzakció (bár nem különbözik értékkészletében a többitől), de azonnali riasztást adna, mert a tanulás során ilyen lépcsőkombinációt nem volt képes felismerni a gyanúgenerálás, vagyis pl. értelmezési intervallumon kívüli visszatérési értéket ad a rendszer (inputhatás-halmozódás, vagy hiány miatt).

Gyanús tehát az a tranzakció, mely esetén a lehető legtöbb változó kódértéke a lehető legkevésbé magyarázható a többi változó és a többi rekord alapján a lehető legtöbb csoportban, de legalább egy tranzakció-csoport alapján. Vagyis minden, ami nagyon egyedi az zömmel nehezen gyanúsítható. Minden, ami egyre kevésbé egyedi, egyre gyanúsabb lehet, ha nem felel meg a többi tranzakció sugallta mintázatoknak. A COCO MCM nem igényel semmilyen emberi inputot (context free = még a változók neve sem feltétlenül kell, minden csak a distinct értéksorok kódolására alapul), s a cél a téves riasztások minimalizálása. A megrendelő/alkalmazó/felhasználó tehát egy újszerű matematikai konstrukció alapján ismerhet fel egyediségeket, érezhet rá ok-okozati összefüggésekre. A tartalomfüggő megoldás esetében már azok a tranzakciók kerülnek nagyító alá, melyek egyedisége minimális: pl. objektumképző lehet a card-id, de lényegében bármi más is... Itt azonban a gyanú fogalmát szakértői szinten kell életre kelteni: pl. gyanús az a TRANZAKCIÓSOROZAT, mely időben egymást gyorsan követő elemekből áll és az értékek egyre csökkennek... Fontos tudni, a fenti context free gyanúgenerálás is képes ezt feltárni, csak nem érthető, nem ismerhető fel az ok-okozatiság automatikusan, miért is gyanús egy-egy tranzakció..., hiszen ha pl. csak az összeg változik, s minden más azonos, akkor ez nem lesz magyarázható semmivel, azaz gyanúvá válik az érintett tranzakciók mindegyike...utólag, ill. bármelyik új tranzakció real-time...

(kék vonal: tranzakció szintű aggregált gyanú, sárga vonal: idő-alapú egyedi gyanú, X-tengely: tranzakció-azonosító, egyben az idő múlása, Y –tengely: gyanúérték logaritmusa)

## fejezet: Módszertani összehasonlító elemzés

## Előzmények

Az összehasonlító elemzések (vö. <http://miau.gau.hu/miau/162/outliers.docx>) egyik alapkérdése, melyik módszerrel milyen kérdésekre milyen minőségű választ, mennyi és milyen erőforrás-felhasználással lehet kapni? Más megfogalmazásban: melyik módszer mit nem tud?

Az alábbiakban A Rapid Miner (Szűcs Imre), a WizWhy (Domján Csaba), az SPSS (Bruder Emese), az R-csomag (Ruff Ferenc) szakértőivel készített interjúk eredményei olvashatók (ill. később a hálózatelemzés témaköre is bevonásra kerül a gyanúgenerálásra való alkalmasság kapcsán). Az oktatás célja az ismeretek objektív és érdek-független megosztása a Hallgatósággal. Ennek érdekében a licence-mentesen és minden tekintetben white boksz-ként rendelkezésre álló hasonlóságelemzés mellett minden egyéb piaci megoldás bemutatása segíti az egyes megoldások előnyösségének és problémáinak beazonosítását a Hallgatóság által saját magukban zajló szintézis keretében.

A szakértők felé feltett kérdés nem volt más, mint: miként nyúlna hozzá egy 1.000.000 (rekord) \*20 (változót) tartalmazó naplóállományhoz context free és a változók ismeretében:

### Rapid Miner (Szűcs Imre)

A gyanús rekordok megtalálását általában 2-féle módon szokás megközelíteni:

* Megkeressük azon változókat (a megadottak között, vagy ezekből levezetve) melyekben kiugró érték-kereséssel a probléma megoldható. Vagyis szakértői beavatkozás szükség ezen elemzés inicializáláshoz. A megoldás során n-dimenziós (n = megmaradt változók száma) keresünk olyan pontokat, melyekre igaz, hogy a környezetükhöz képest jobban eltérőnek mutatkoznak, mint más pontok a saját környezetükhöz képest. A kiugró értéket tehát nem csak egy, hanem akár több változó együttes kezelésével is kereshetjük. Ebben a megközelítésben a problémát visszavezetjük kiugró érték-keresési problémává.
* A gyanúgenerálási feladatot olyan osztályozási problémának is felfoghatjuk, amiben egyetlen egy osztály van. Azok lesznek a gyanús rekordok, melyeknek ebbe az osztályba való tartozási valószínűségük alacsony (vö. pl. anti-diszkriminációs LP-számítások).
* Feltárjuk azon üzleti logikákat, melyek gyanút jeleznek, és ezeket képezzük le megfelelő algoritmusokkal (vö. célirányos, tartalomfüggő gyanúgenerálás).

Rapid Miner specifikus tudnivalók: Outlier detection-re több lehetőség is van valamilyen metrikával a távolságok számítására: ilyen például 1-class SVM implemenetáció.

### WizWhy (Domján Csaba)

A WizRule szoftver annyira célszoftvere az említett problémamegoldásnak, hogy gyakorlatilag az elemezendő fájl bekérése után egyetlen kattintással elindul az automatikus folyamat, amelynek az eredménye a gyanús esetek valószínűség alapján sorba-rendezett listája. Mivel a folyamat automatikus, ezért inkább a „hogyan” kérdés helyett a „mi az előnye” kérdésre érdemes kitérni elsődlegesen:

* A WizRule feltárja egy adatbázisban lévő összes if-then szabályt (még a helyesírási hibákat is kimutatja – vö. nominális skálán ábrázolt, azaz egymástól minőségileg különbözőnek ítélt adatok). Ha belegondolunk abba, hogy már 20 változó esetén is több millió variáció létezik a változók egymás utáni vizsgálatára, így emberi elemzés esetén a „miért éppen azzal kezdjük”, „miért éppen azzal folytatjuk” kérdésekre nem lehet elfogadhatóan válaszolni. Sajnos az emberi agy kiragad néhány mintát (pl. milyen csalások történtek a múltban”) és annak mentén próbál gondolkodni (=próbálkozni). A WizRule nem esik ebbe a hibába, illetve számára nem jelent gondot átvizsgálni annyi variációféleséget, amely egy elemzőnek akár éveket jelente még szoftvertámogatással is.
* Legtöbben a gyanús esetek hallatán szélsőséges, kiugró értékekre asszociálnak, holott ez csak egy része a problémakörnek. Például egy privát bankos ügyfél százmilliós tranzakciókat is bonyolíthat, és mégsem tekinthetjük gyanús esetnek (sőt megsértődne a bankjára, ha egy algoritmus őt tüntetné fel megvizsgálandó esetnek, akár többször is). De az átlagos keresetű ügyfél számláján is megjelenhet egy nagyobb összeg, ha hitelt kapott. Számtalan ilyen eset hozható fel és a teljesség igényével nem lehet felsorolni egy elemzőnek, annak érdekében, hogy a szoftvere ne tévedjen. A WizRule éppen az összes szabály miatt automatikusan képes feltárni, hogy példa okáért, ha az illető privát bankos és százmilliós utalása van, akkor se legyen gyanús, a hitelt kapott ügyfelet se listázza ki ellenőrzésre és minden olyan más esetben is magyarázhatónak minősít kiugró értékeket, amelyre nem is gondol egy szakember. Azaz minden változót minden változóval megvizsgál, és ha talál magyarázatot (pl. az egyedi! „hitelátutalás” szó szerepel a comment mezőben, akkor megmagyarázottnak tekinti (de ezt az egyediséget nem kell megtanítania vele, vagyis nem kell leprogramozni az elvi/kontextus-függő lehetőségeket). Ezáltal csökken pl. a banki dolgozók felesleges munkája (nem kell minden második esetet kidobni a listából látva a megmagyarázhatóságát), a bank presztízsét is megőrzi.
* Nagyon sok csaló tudja jól, hogy a cégek többnyire a kiugró tételeket vizsgálják, ezért „átlagos méretű” csalásokat követnek el (pl. ATM csalások, biztosítási csalások egy része), de ezek szokatlansága feltárható WizRule által. Ezért a kiugró értékek helyett/mellett a „szokatlan” esetek feltárása lenne a fontos, de ez csak akkor valósulhat meg, ha az összes „normális viselkedés” szabályát feltárjuk, és aki nem tartozik ezek közé, az tekinthető legtöbbször gyanús esetnek.
* Nagyon fontos a reprodukálhatóság. A WizRule ugyanazzal az adatbázissal pl. 2 év múlva is ugyanazt az eredményt hozza ki, ellenben ha más módszereknél manuálisan kell vezérelni a szoftvert (mit vizsgáljon meg, mi alapján, mi legyen a küszöbérték, stb.), akkor más eredmény születik minden alkalommal. Ez olyan, mint a lottó: ha minden kombinációt kitölthetnénk, akkor mindig ötösünk lenne, ellenben ha kevés szelvénnyel próbálkozunk, akkor csak néha lesz nyereményünk. A fentiekben már említettem, hogy a WizRule minden if-then szabályt feltár, így nem szerencse dolga, hogy megtalál-e gyanús eseteket, és az sem, hogy ami megmagyarázható, annak a megmagyarázhatóságát feltüntesse, és ne szerepeljen a rekord a gyanús esetek között.
* A gyanús eredmények értelmezése elengedhetetlen lenne (nem mondható, hogy „ezt dobta a gép, valami miatt gyanúsak”). A WizRule megadja a gyanú hátterét is értelmezhető if-then szabályok formájában. Ebből következik, hogy a gyanús esetek exportálhatók és minden igazgatóság megkaphatja a hozzá tartozó eseteket az értelmezéssel együtt.
* Az összehasonlíthatóságra kellene törekedni a módszerek kapcsán. Ha egyik módszernél szükséges x nap konzultáció és nem csak nyers változók kerülnek be szakértői tudás által, úgy az eredmény nem tudható be 100%-ban a módszernek és senki sem tudja a helyes arányt a primer adatok és a szakértői hozzájárulás között. Ha mindez automatikus, akkor 100%-ban a módszer eredménye a gyanús esetek feltárása. Ismerve a tanácsadói díjakat, elmondható, hogy egy napi díjból megvásárolható a WizRule és örökös tanácsadóként működtethető évtizedeken át.
* A gyanús esetek feltárásánál az egyik legfontosabb az új csalásfajták feltárása. A szoftver jöjjön rá elsőként(!) az új típusokra és ne a szakember (utóbbi esetben már rendszerint nagy a baj). Ezért is fontos minden szabály feltárása, feltárni az ismert és még nem ismert szabályokat. A már ismert csalástípusoknál már illene programnak léteznie a prevenció érdekében. Ezért az új típusú esetek felfedezésénél a szakértői tudás nem sokat számít.
* A téves riasztások elkerülése is összefügg a korábbiakkal (azaz ha létezik értelmes magyarázat, akkor azok automatikusan zárják ki a riasztásokat).
* Nem elégséges gyanús eseteket detektálni, nagyon fontos a gyanúk valószínűségét is kalkulálni és ezáltal sorba rendezni a gyanús eseteket, hogy a legsürgősebb esetekkel lehessen kezdeni a prevenciót. A WizRule ezt valósítja meg.
* Az ügyfelek rendszerint sokféle típusba sorolhatók, így a szegmentáció nem tud jó megoldást nyújtani, hiszen a másság miatt is „kilógnak” ügyfelek, de ezek többnyire megmagyarázható esetek. A WizRule a sok-sok szabály által nem követi el azt a tévedést, hogy ha pl. 3 csoportba nem sorolható be valaki, akkor gyanúsnak tekinti. Ha egy cégnél pl. 158 féle típus van, akkor mindegyiket fel tudja tárni és ezt nem kell megtanítani vele.

Természetesen a WizRule esetében is attól függ a sikeresség, hogy a releváns változók rendelkezésre állnak-e (a fent említett példát követve, ha nem szerepel, hogy az adott ügyfél privát bankos, akkor talán a kiugró értékek téves riasztáshoz vezethetnek, de az is lehet, hogy más adatokból mégis megmagyarázhatónak tekinti (pl. csak a privát bankosoknál lehet egy adott számlatípus és annak megnevezése szerepel az adatok között). Egy-egy adatpiac akár 600-700 változót is tartalmazhat, és mivel nem ismerjük, hogy milyen gyanús eseteket akarunk kiválasztani, így a logika elvén sincs mód „valahol elkezdeni”. Ezért fontos az automatikusság, illetve néhány aspektus helyett az összes alapján történő vizsgálat.

WizRule vs. Wizwhy: amennyiben a megtanulható következményváltozó rendelkezésre áll, akkor a WizWhy is hatékonyan bevethető eszköz a gyanúgenerálás során (<http://moneyfromprediction.com/>).

Az egymásra épülő tranzakciók felismerése nem magától értetődő (pl. azonos helyről rövid időn belül azonos kártyára vonatkozó egyre csökkenő pénzfelvételi kísérletek): hiszen ez speciális tartalomfüggő gyanúként értelmezendő.

Távolság-alapúság vs. egy halmazba kényszerítés: A WizRule kombinatorikai alapú, de képes a távolság és egy halmazba tartozást is értelmezni, lévén a szabályalapúság szegmentáló jellegű (vö. egy halmazba tartozás).

Nominális skála vs. magasabb skálák: A WizRule felismeri a magasabb fokú skálák információs többletértékét, vagyis intervallumokat képes képezni szükség esetén.

Újratanulás vs. szimuláció: A WizRule az X+1. rekordra nem alkalmazza a korábbi X-re feltárt szabályokat, hanem újratanul. A WizWhy képes szimulációs jelleggel (vö. szakértői rendszer) működni.

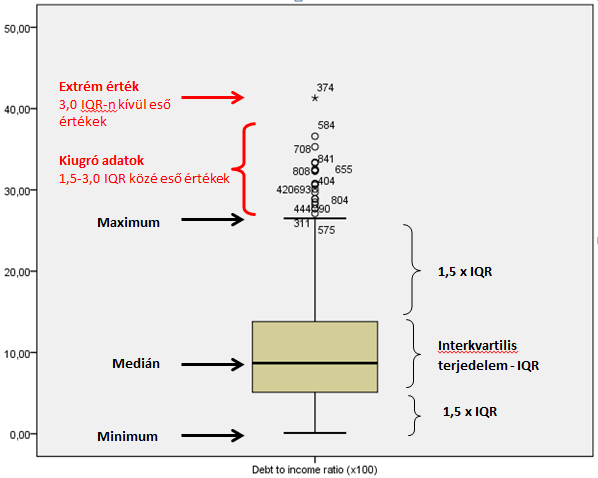
Teljes rekordhalmaz vs. részleges rekordhalmaz: A teljes rekordhalmazon könnyebb a szabályt erősítő minimális rekordszámot teljesíteni. Tehát a context free jelleg hatásossága függ a futtatás alapparamétereitől (minimális esetszám/szabály, max. változószám/szabály). A túl sok szabály között nem vész el a gyanú.

Futásidő: 1\*1.000.000 futtatás munkaideje kb. = 10\*100.000 tanulás. Tehát: ha a több rekord jobb szabályokat képes adni, és a részhalmazokkal nem lehet időt nyerni, akkor az egyben futtatás előnyösebb, kivéve, ha real-time részeredményekre van szükség feltétlenül, vagy a futásbiztonság nem stabil eléggé (áramszünet, stb.).

### SPSS (Bruder Emese)

Az SPSS egy adatbázis kezelő statisztikai program. Természetesen képes kezelni az 1.000.000 rekordos adatmennyiséget, a változók száma is szinte korlátlan, körülbelül 32.000. Nominális, ordinális és metrikus skálán mért adatokkal dolgozik. Létezik egy adatbányászati modul, ami az SPSS Modeler (Clementine) nevet viseli, és létezik egy SPSS Text Analytics for Surveys is, ez szövegek elemzésére lett kifejlesztve. Ezek külön modulként vásárolhatók meg, nem jár az alap szoftverhez.

A kérdés lényeget adó problémára részben talán megoldást adhat az alap modul is: a "mennyi összeget mozgatott az ügyfél" kérdés esetében a kiugró értékek figyelése megoldható egy boxplot diagrammal, vagy egy egyszerű sorba rendezéssel. A boxplot diagram jelzi, hányadik esetnél tapasztalható a kiugró, vagy extrém érték. Ez után az adott eseteket manuálisan kell ellenőriznünk, de szűrhetünk is a kiugró értéket tartalmazó esetekre.



Természetesen a minimum értékhez közelebb eső kiugró adatokat is figyelhetjük, abban az esetben, ha például a kis összegű, tranzakciókat gyanúsnak ítéljük. (például kis összegű, percenkénti tranzakciók) Az SPSS programcsomag kezeli a dátumváltozókat másodperc pontosságig, tehát a gyakori tranzakciók felismerhetők.

Másik lehetőség, hogy a Select Cases menüben kiválogatunk olyan eseteteket, amelyek egy komplex feltételrendszernek tesznek eleget. Ehhez tudnunk kell, hogy mely kombinációkat ítéljuk gyanúsnak. Szakértők segítségével létre kell hozni azt az összefüggésrendszert, amelyek gyanúra adhatnak okot. pl: Select Cases If: honnan="Hungary" & hova="Panama" or "Mauritius" & összeg>1.000.000 USD... Itt tetszőlegesen hosszú algoritmust adhatunk meg matematikai műveletek segítségével. =, ~=, >,<, és, vagy. A kiválogatás után csak a "gyanús" esetek maradnak az adatbázisban, ezekkel dolgozhatunk tovább. A módszer hátránya, hogy csak az általunk betáplált műveletek szerinti gyanús eseteket szűri ki, semmi féleképp sem képes tanulásra, real time gyanúgenerálásra nem használható.

A program segítségével képezhetünk egy olyan komplex mutatórendszert, amely az egyes rekordokban található tulajdonságok alapján az eseteket pontozza az általuk generált gyanú szerint, ehhez akár új változókat is generálhatunk a meglévők felhasználásával. A magas pontszámot kapott esetek lesznek a gyanúsak, az alacsony pontszámmal rendelkező esetek kevésbé gyanúsak. Ebben az esetben is a pontozás feltételrendszerét nekünk kell megadnunk.

### R-csomag (Ruff Ferenc)

Szélsőséges adatok detektálása többváltozós térben az R környezet segítségével( <http://www.r-project.org/>):

1. aq.plot algoritmus (<http://cran.r-project.org/web/packages/mvoutlier/mvoutlier.pdf>)

Alkalmazható, ha az egyes változók arányskálán mérhetők, valamint normál eloszlásúnak tekinthetők. A módszer lényege, hogy a távolságmérésre az ún. robusztus Mahalanobis távolságot használja, melynek négyzete *k* szabadsági fokú Khi négyzet eloszlást követ (*k* a változók számát jelenti.). A mért távolságának eloszlását veti tehát össze az elméleti eloszlásfüggvénnyel, és két különböző kritérium (0.975 ill. módosított kvantilis) esetén is meghatározza a szélsőséges adatokat. Természetesen nem csak grafikusan kapjuk meg az eredményt, hanem egy kétértékű vektor formájában is.

aqplot.tiff

1. sign1 algoritmus (<http://cran.r-project.org/web/packages/mvoutlier/mvoutlier.pdf> )

Ezt az eljárást kimondottan sokváltozós, sok megfigyelési egységet tartalmazó adatbázisok esetére fejlesztették ki. A távolságmérés alapja itt is a Mahalanobis távolság. A dimenziószám csökkentése érdekében a Főkomponens analízist használja, és ebben az új térben számolja a távolságokat és hasonlítja össze őket az elméleti Khi négyzet eloszlással. Az eredményt szintén egy kétértékű vektor formájában adja meg.

Melléklet: Nominális skálán ábrázolt inputok hasonlóságelemzésének értelmezése

A hasonlóságelemzéssel feltárt gyanú rétegekből áll.  
Egy-egy rekord saját rokon-csoportjaihoz képest a vizsgált attribútumok mentén bármely attribútum szempontjából gyanússá válhat (ha értelmezhető lefelé, vagy felfelé jelentősen eltérve). Amennyiben az adatok nominális skálán, azaz alfanumerikus, egyedi jelként kerülnek értelmezésre, akkor nincs lefelé és felfelé értelmezésre lehetőség, hiszen nem is kell tudnia az elemző programnak, mi volt eredetileg az adott attribútum jelentése (vö. context-free jelleg). Amennyiben a jelenségről csak annyit tudunk, hogy számszerűsíthető (pl. amount), akkor sem lehet automatikus eltekinteni azon anomáliáktól, melyek pl. egy átutalás túl kicsinek vélelmeznek.  
Példa: Az amount request attribútum esetén azon gyanútételek kizárása, melyek szerint az átutalt összegnek illett volna sokkal kisebbnek  (pl. fele vagy még ennél is kisebb) lenni, mint a ténylegesen átutalt összeg, lehetséges. Itt lehet ugyanis átlépni a context-dependent logikai szintre: hiszen ezek a gyanúk lehetnek anti-gyanúk, mert nem kell akciót kapcsolni feltétlenül hozzájuk. De az is igaz, hogy pl. valahol egy nagyvállalatnál a sok csak nagy utalás mellett néhány kicsi mégis csak lehet gyanús, mert kilógnak az utalási profilból...s lehet sikkasztási szándék mögötte... Tehát a context-dependent gyanúrétegek felhasználóbarát síkokat nyitnak ezek ki és bekapcsolhatósága révén...  
  
A többrétegű gyanúgenerálás esetén automatikusan szöveges jelentéssé formálható minden egyes gyanúmomentum:  
- ténylegesen nominális változó esetén a standard szöveg nem más, mint az adott jelenséget (pl. location) nem lehetett rekonstruálni a többi tranzakciót jellemző adat alapján, vagyis vélelmezhető, hogy ilyen location esetén ilyen tranzakció nem valószínűsíthető. Noha a tény és a norma numerikus eltérése formálisan nem értelmezhető, mégis igaz az optimalizálás kapcsán, hogy minél nagyobb az eltérés, annál nagyobb a gyanú valószínűsége, hiszen az optimalizáció lényegében restrikciók nélkül bármilyen logikátlan/ alaktalan felületre képes rásimulni...  
- legalább rangsor-skálát feltételezni engedő attribútum-tartalmak esetén a kalkulált norma és a tény eltérése valódi nagyságrendi problémákat enged feltételezni:  
-- pl. adott összeg túl sok, vagyis ilyen tranzakciós környezetben ilyen mértékű pénzmozgás szokatlan (vö. privát erőforrások napi limitet közelítő kiaknázása)  
-- pl. adott összeg túl alacsony, vagyis ilyen tranzakciós környezetben ilyen mértékű pénzmozgás szokatlan (vö. nagyvállalati sikkasztás gyanúja)  
  
Harapófogó effektus: Minden, ami tehát csak egyszer fordul elő, a flexibilis algoritmus által egyedi, véletlen, egyszeri esetként nem lenne automatikusan gyanús, éppen ezért a tanulási mintában sql-oldalon minden egyediség gyanúként szűrendő.  
  
A kiscsoportos tanulás gyorsabb, de kockázatosabb a kevesebb egyidejű minta miatt. A kiscsoportok kialakítása az egyediség egy speciális fuzzy-rétege (hiszen maga a többrétegű gyanú-fogalom is egy fajta fuzzy megközelítés). A kiscsoportok (kis elemszámú, egy-rétegű where-feltétellel leírható speciális sql-halmazok) az egyediség következő fokozatát írják le. A véletlenszerűen választott vagy nagy elemszámú csoportok jelentik a normál működési rendet, ahol semmilyen (gyakorisági alapra visszavezethető) többletinformáció nem áll a gyanúgenerálás rendelkezésére.  
  
Önellenőrző számítások lehetősége: ha egy bankkártyához több, mint egy gyanú kötődik folyamatos működés kapcsán (tehát már a 2. gyanú esetén is), ott a teljes kártyatörténet nagyító alá vehető. Egy kártya-történetben a gyanúsnak minősített pillanatok karakterisztikusan el kell, hogy térjenek a többi történéstől. Abban az esetben, ha a tanulás kihasználta a kiscsoportos (gyakoriság-alapú) többlettudást, akkor sem kell feltétlenül újra tanulni egy olyan tranzakció érdekében, ami egyetlen egy kiscsoport where-feltételének sem felel meg: ilyenkor minden egyes ismert gyanú-paramétertömb (lépcsős függvény), mint egyedi szakértői tudás használható fel az új rekord gyanúmértékének többrétegű levezetésére, ha egyetlen egy nominális skála-kód sem hiányzik sem input sem output oldalon az egyetlen csoportba soroló kódon túl. Hiszen a besorolás alapját jelentő attribútum-opció hiányában felismert gyanú-esély valódi gyanúként értelmeződik. Az adathiány (feltétel-nem-teljesülés) pedig nem hat. Ez nem más, mint egy véletlen minta szimulálása. A rendszerműködtetés során valós döntési kérdés: vajon tanulni gyorsabb-e, vagy a meglévő szakértői tudást lehívni kerül-e kevesebb időbe?  
  
A másik lehetséges önellenőrző számítás az idő-tényező levezetésére tett kísérlet minden egyes tranzakció kapcsán: vagyis a kérdés: Történhetett-e pont akkor a tranzakció?  
Az időanomália-görbe és a tartalom-független többrétegű gyanú-görbe időnézete illik, hogy nagy gyakorisággal együttmozgásokat mutasson.