Lakáshitel vizsgálat
SAS Enterprise Miner 5.3
alkalmazásával

Készítette: Soltész Gábor

solteszgabee@gmail.com

2010.

Tartalomjegyzék

1. Feladat definiálása 4
1.1. Rendelkezésemre álló adatok 4
2. Adatok importálása SAS Enterprise Miner 5.3 programba....4
2.1. SAS Base használatával 4
2.2. Importálás közvetlenül a Miner használatával SAS kód csomópont használatával 6
2.3. Importálás Miner használatával „Merge” csomópont által 7
3. Adatfelosztás (Ismertető) 8
3.1. Tényleges adatfelosztás....9
4. Adatmódosítás....14
4.1. Hiányzó értékek pótlása....14
4.2. A binnelés technikája....24
5. Modellalkotás 25
5.1. Decision tree – Döntési fa 25
5.2. Neural Network / DNNeurál – Neurális háló....27
5.3. AutoNeural....27
6. Kiértékelés 28
6.1. Model comparison – modellek összehasonlítása....28

Ábrajegyzék

1. ábra Adatforrás meghatározása 5
2. ábra Explorer ablak 6
3. ábra Importált SAS adattábla 6
4. ábra Diagram létrehozása 7
5. ábra SAS kód futtatása 7
6. ábra Illesztés alkalmazása 8
7. ábra Illesztési változó meghatározása 8
8. ábra Adatfelosztás 9
9. ábra Célváltozó létrehozása 10
10. ábra Célváltozó 10
11. ábra Célváltozó eloszlása.....10
12. ábra Adatfelosztás megvalósítása.....11
13. ábra Filter beállításai.....11
14. ábra Szűrés meghatározása.....12
15. ábra A célváltozó egyes értékeinek eloszlása.....12
16. ábra Mintavételezés.....13
17. ábra Összefűzés beállításai.....13
18. ábra Megfigyelések megtekintése 14
19. ábra Hiányzó értékek feltárása 14
20. ábra Intervallum változók statisztikai mérőszámai.....15
21. ábra Célváltozó szegmensek szerinti eloszlása.....15
22. ábra Osztályozó változók statisztikai mérőszámai.....15
23. ábra Csere csomópont alkalmazása.....16
24. ábra A pótlás és a csere két változata 16
25. ábra Csere és Pótlás 16
26. ábra Csere csomópont beállításai.....17
27. ábra Csereszerkesztő.....17
28. ábra Nettó jövedelem a csere után.....18
29. ábra A pótlás utáni eredmény 18
30. ábra Grafikon létrehozása 20
31. ábra Két változó felhődiagram.....20
32. ábra Hiányzó értékek pótlása fával.....21
33. ábra Pótlás utáni felhődiagram.....21
34. ábra Adattranszformáció.....22
35. ábra Transzformált változók 22
36. ábra Pótlás a megadott értékkel.....23
37. ábra Folyamatábra a binnelés-sel kiegészítve 24
38. ábra Csoportképzés.....24
39. ábra Döntési fa, neurális háló modellek és összehasonlításuk.....25
40. ábra Döntési fa eredménye 26
41. ábra Neurális háló eredménye 27
42. ábra Kiértékelés - modellek összehasonlítása 28

1. Feladat definiálása

A feladat az jó modell készítése, ami minden háztartás esetén egy jóslást ad számunkra, hogy az adott háztartás rendelkezik e lakáshitellel. A feladatot SAS Enterprise Miner 5.3 alkalmazásával valósítom meg.

1.1. Rendelkezésemre álló adatok

Az adatokat a KSH háztartás és személy állományából válogattam le. Ez az adatállomány már csak az elemzéshez valamilyen mértékben kötődő változókat tartalmazza. A KSH állományok xls kiterjesztéssel rendelkeznek, így ezeket a táblákat elsőként importálni kell SAS adattáblává, mivel a Miner önmagában nem képes importálásra. Az importálás után rendelkezésünkre állnak a KSH adatállományok a „*c:\hitel*” könyvtárban.

2. Adatok importálása SAS Enterprise Miner 5.3 programba

Ebben a pontban az adatállományokat importáljuk a SAS Enterprise Miner 5.3 változatba. A cél, hogy egy projekt létrehozása esetén az adatállományok elérhetőek legyenek. Ennek a megvalósításnak több útja is lehetséges.

2.1. SAS Base használatával

Ebben az esetben elindítjuk a SAS Base-t, majd az alábbi kóddal leválogatom a szükséges változókat. Ezek után végrehajtok egy illesztést és 1 adatállományt (*gen_hazt*) generálok a korábbi állományaimból. Ez az állomány a „*c:\em*” könyvtárban található.

```
/*Könytárak definiálása*/

libname hitel 'c:\hitel';
libname em 'c:\em';

/*Háztartás állomány leválogatása*/

data new_hazt;
 set hitel.hazt(keep =hazon HLETS hlakft hlanm HLBER HLTOR HLKTG HLVIL
HLAKHI HLGAZF HLVIZ HLFUT HLHIT
    HLKTR HKOMP HAERT HMLAK HUD HEPTEL HMEZFOL HNJE HLAKA HLTIP HMOTOKA
HMOTOKB
    HMOTOKC HMOTOVA HMOTOVB HMOTOVC HSTSZGA HSTSZGB HSTSZGC HCTSZGA);
run;

/*Személy állomány leválogatása*/

data new_szem;
 set hitel.szem(keep= hazon sneme szkor scsap scsal SCSAL SBISK1 SBISK3
ESNJ SFEOR SHAZTFO SDOHE SKAVE SGYOR);
run;

/*Illesztés*/

proc sql;
 create table em.gen_hazt as
```

```
    select a.*, b.*
    from
     new_hazt as a
     left join
     new_szem as b
    on a.hazon = b.hazon;
   quit;

data em.gen_hazt;
 set em.gen hazt;
      if hlakft=. then hitel=0;
      else hitel=1;
      label hitel="Történt e hitelfelvétel";
run;
```

Ezek után a leválogatott adatállományt a „*c:\em*” könyvtárban mentettem el ***gen_hazt*** néven.

Elindítom a SAS Enterprise Miner 5.3-t.

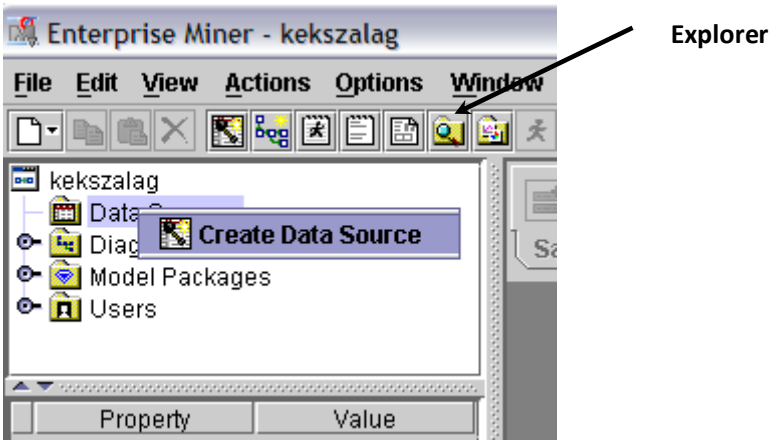
Létre kell hoznunk egy új projektet. Ezt az alábbiak szerint tehetjük meg.

Fájl / New / Project

Meg kell adnunk a projekt nevét és a SAS szervert, amin dolgozni akaFuttatásk.

Esetemben a projekt neve ***hitel*** a szerver pedig a ***SASMain – Logical Workspace server***

A projekt létrehozása után az első legfontosabb dolog a forrásállományok definiálása. Ezt a project panel segítségével egyszerűen megtehetem az *Adatforrásra* (DataSource) ikonra kattintva. (1. ábra)

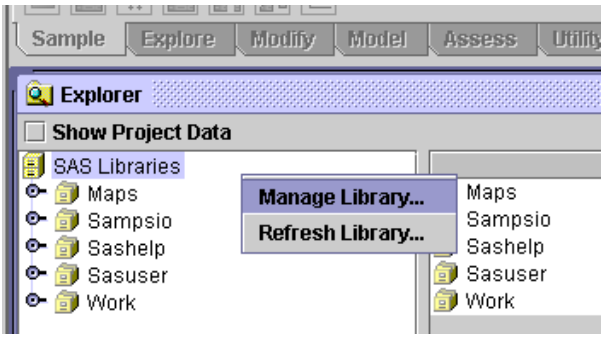


1. ábra Adatforrás meghatározása

Metaadat forrásnak válasszunk SAS táblát.

Következő lépés a kívánt SAS tábla kiválasztása. Azonban itt alapértelmezés szerint csak a programmal telepített és definiált könyvtárak találhatók, így első lépésként egy új könyvtárat kell létrehozni. Ezt a 1. ábra alapján az *Intéző* (***Explorer)*** segítségével tehetjük meg.

Az Explorer ablakban jobb klikk majd a *Könyvtárkezelő* (**Manage Library…) (**2. ábra**)**



2. ábra Explorer ablak

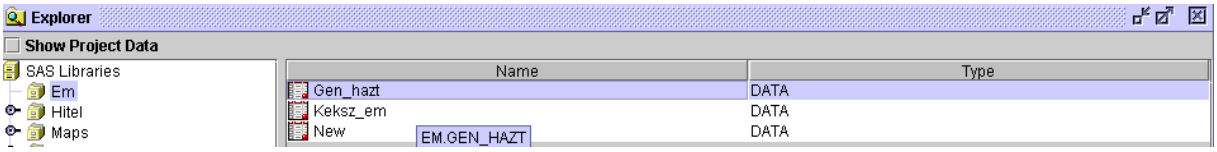
Új könyvtárat akarunk létrehozni. Ehhez definiálnunk kell a könyvtár nevét és az adott könyvtár elérési útját, esetemben ez *EM*, és az elérési út *c:\EM*.

FONTOS, hogy annak a könyvtárnak az elérési útját adjuk meg, amiben a korábban létrehozott SAS állomány található *(gen_hazt.sas7bdat).* Ellenkező esetben csak azokkal a SAS táblákkal fogunk tudni dolgozni, amik az adott könyvtárban találhatók.

Amint létrehoztuk a kívánt könyvtárat ellenőrizzük, hogy a kívánt névvel és tartalommal hoztuk e létre, ehhez először is frissíteni kell a könyvtárstruktúrát.

Klikkeljünk a *Könyvtárfrissítés* (**Refresh Library…)** parancsra. (2. ábra)

Ha mindent jól csináltunk, akkor a következő ábrát kell, hogy kapjuk. (3. ábra)



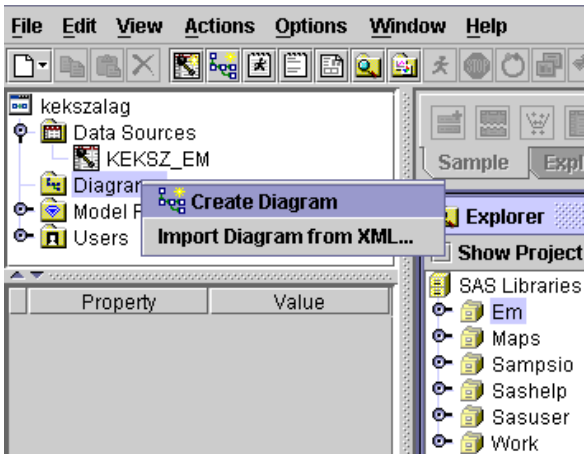
3. ábra Importált SAS adattábla

Ezek után hozzuk létre az adatforrást, amivel dolgozni szeretnénk, ehhez használjuk az *Adatforrás létrehozás* (**Create Data Source)** utasítást használva. (1. ábra) Forrásnak válasszuk ki az *Em* könyvtárban található *Gen_hazt* SAS adattáblát. Ezek után a projekt panel *Adatforrások* pontja alatt megjelenik a kiválasztott tábla.

2.2. Importálás közvetlenül a Miner használatával SAS kód csomópont használatával

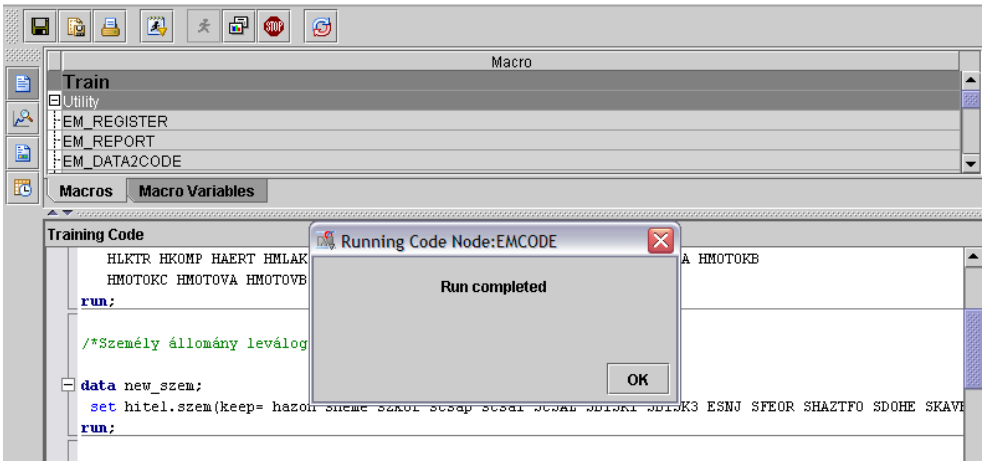
Hozzunk létre egy új projektet az előző pontban leírtaknak megfelelően. Majd hozzunk létre egy diagramot.

Diagramot a Projekt Panel segítségével tudunk létrehozni a *Diagramok* mappára klikkelve, majd a *Diagram létrehozása* parancsot választva.(**Hiba! A hivatkozási forrás nem található.**) A létrehozott diagram neve legyen *hitel.*



4. ábra Diagram létrehozása

A létrehozott diagramon, helyezzünk el egy ***SAS kód*** csomópontot, amit a *„Utility”* fül alatt találunk meg. A bal oldali tulajdonságok panelen válasszuk a *„Kód”* pontot, majd adjuk meg a fent található kódot. Majd jobb klikk a csomóponton és *Futtatás*. (5. ábra)



5. ábra SAS kód futtatása

A kód hatására létrejött az adatállomány a *„c:\em”* könyvtárba *gen_hazt* néven. Következő lépésben a projekthez kell rendelni az adatállományt. Ezt az 1. ábra alapján lehet megvalósítani.

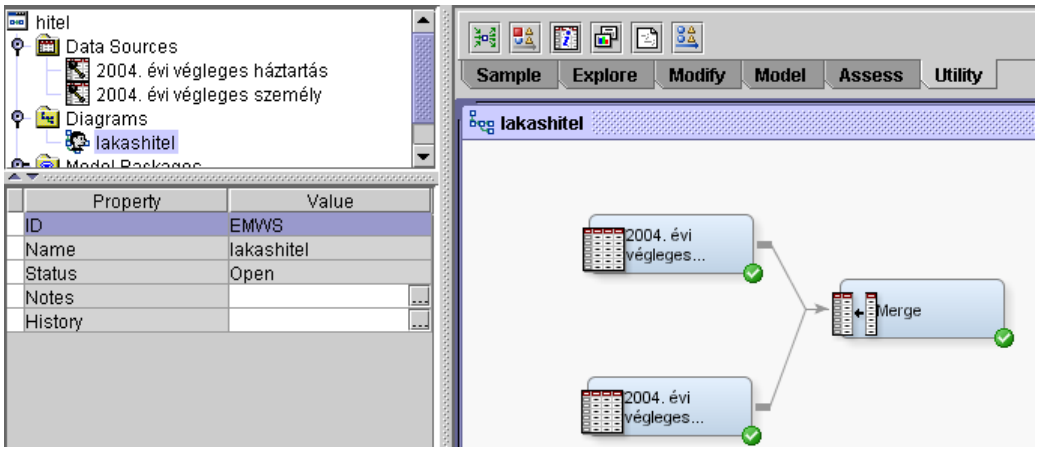
2.3. Importálás Miner használatával „Merge” csomópont által

Kezdeti feltevés, hogy a két KSH adatállomány SAS formátumban a *„c:\hitel”* könyvtárban már a rendelkezésünkre áll.

Indítsuk el a Miner-t, és definiáljuk *„c:\hitel”* elérési úttal a *Hitel* nevű könyvtárat az 1. ábra alapján. Ebben az esetben a definiált könyvtár két darab fájlt fog tartalmazni és nekünk mind a kettőt hozzá kell rendelnünk a projektünk adatforrás könyvtárához. (6. ábra bal felső sarok)

Hozzunk létre egy diagramot „lakáshitel néven” a 4. ábra alapján, majd helyezzünk el a két forrásállományt a munkaterületen és kössünk hozzájuk egy *Illesztés* (**Merge**) csomópontot. (6. ábra)

A forrásállományokon a *Változók szerkesztése* (**Edit Variables…**) segítségével határozzuk meg az a számunkra szükségtelen változókat. Ezt az *Elvetés* (**Drop**) oszlopban adhatjuk meg. A megtartani kívánt változókat a SAS kód „*set*” parancs „*keep=*” utasítása tartalmazza.



6. ábra Illesztés alkalmazása

Az előző módosításokat hajtsuk végre a másik adatállományon is. Természetesen a hozzá tartozó változók megtartásával.

Majd mind a két állomány esetén állítsuk be a *Változók szerkesztése* pont segítségével a HAZON változó *Szabály* (**Role**) szerepét *Input*-ról *ID*-re.

Következő lépésben az *Illesztés* csomópontban, adjuk meg az illesztési változót. Esetünkben ez a változó a *HAZON*.

Name	Merge Role	Overwrite Variable	
HAERT	<none>	Default	Inp
HAZON	By	Default	Inp
HCTSZGA	<none>	Default	Inp

7. ábra Illesztési változó meghatározása

Az *Illesztési szabály* (**MergeRole**) oszlopban állítsuk a HAZON változót *By*-ra. (7. ábra)

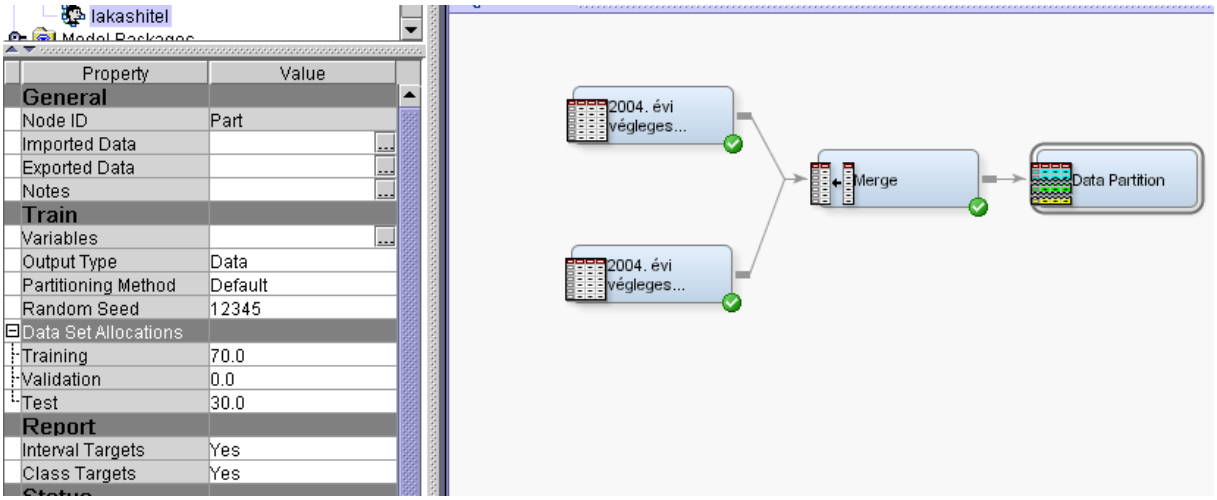
Futtassuk a csomópontot és tekintsük meg az eredményt a bal oldali *exportált adatok* (**Exported Date**) tulajdonság résznél, majd a kapott ablakban válasszuk *a tanuló* (**TRAIN**) adatokat és nyomjuk meg a *Böngészés* (**Browse**) gombot.

3. Adatfelosztás (Ismertető)

A SAS Enterprise Miner-ben az *Adatfelosztás* (**Data Partition**) csomópont hozza létre a tanuláshoz használt tanuló adathalmazt (Train), a tanulás visszacsatolásához használt valódiság ellenőrzése (Validation) halmazt és a tanulás folyamatától szigorúan elkülönített teszt (Test) halmazt.

A Validation halmazt valójában nem minden tanuló algoritmus használja, ezek esetében a Validation halmaz Test halmaz minőségben marad meg. Sajnos jelen pillanatban még nem tudtuk meghatározni, mely modell csomópontok igényelnek Validation halmazt.

Kössünk egy ilyen csomópontot az Illesztés csomópont után. (8. ábra)



8. ábra Adatfelosztás

Az adatfelosztás beállításai között szerepel a megfelelő metódus kiválasztása. A Miner az alábbi metódusok alapján képes felosztani az adatállományt.

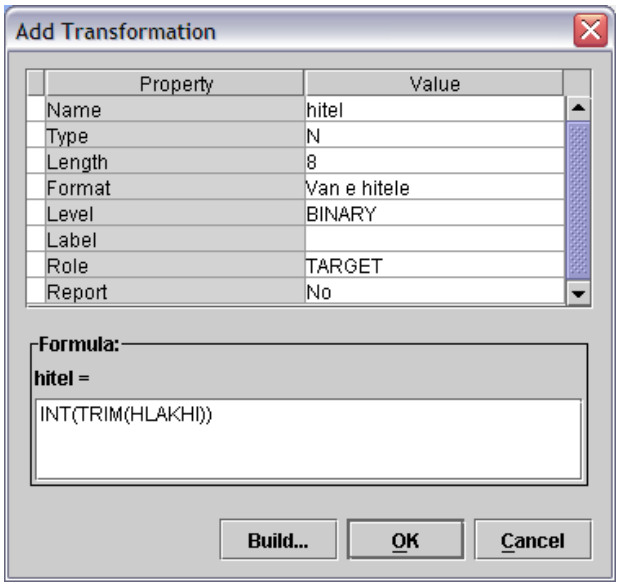
- Véletlen alapú szelekcióval, bináris célváltozónál alapértelmezett.
- Klasztereket alakít ki a megfigyelésekből és ezeken belül alkalmazza a véletlen felosztást. Class célváltozó esetén alapértelmezett, a klaszterezés alapja célváltozó.
- Stratified (rétegzett) esetben előre definiált klasztereken belül alkalmazza véletlen elosztást.

3.1. Tényleges adatfelosztás

Ha megfigyeljük az adatállományunkat, akkor láthatjuk, hogy több mint, 19000 rekordot tartalmaz. Mivel a célunk egy olyan modell megalkotása, mely előre jelzést ad arról, hogy egy ügyfél fog e hitelt felvenni, vagy sem, szükséges egy célváltozó kiválasztása. Itt előrejelzésről van szó és ebben az esetben a legcélszerűbb megoldás, ha a célváltozó bináris (2 értékű) változó, ami azt mondja meg, hogy az ügyfél vett e fel hitelt. Ha megfigyeljük az adatállományunkat, akkor láthatjuk, hogy erre csak egy változóból lehet egyértelműen következtetni. Ennek a változónak a neve: *Mennyi hitelt vettek fel (ezer Ft-ban)? – metaadat azonosító: HLAKFT.* Ennek a változónak a segítségével, lehetőség van egy olyan bináris változót létrehozni, ami megfelelő lesz célváltozónak. Ha az adatállományt a korábban (SAS Base használatával) ismertetett fejezet alapján hoztuk létre, akkor az ott definiált „Hitel” változó teljes mértékben megfelel célváltozónak. Ha azonban illesztéssel alkottuk meg a kiindulási állományunkat, akkor egy változó *transzformációs csomópontot* kell az illesztés után kötni (**Transform Varrriables**), majd a *formula* pontban létre kell hozni egy új változót (Create). (9. ábra)

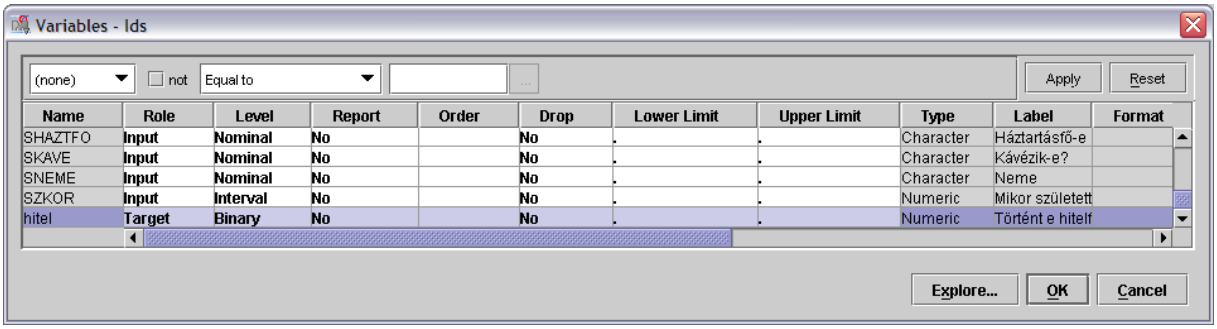
Ezzel a lépéssel létrehoztunk egy „Hitel” bináris célváltozót, amit úgy képeztünk, hogy a „Van e lakáshitel törlesztése” – HLAKHI változóból eltávolítottuk a felesleges karaktereket, majd egész számmá konvertáltuk őket.

Most akármelyik lépéssel is hoztuk létre a forrásállományt, mindegyik esetben rendelkezésünkre kell, hogy álljon egy „Hitel” nevű változó, mely megfelel a fent definiált tulajdonságoknak.



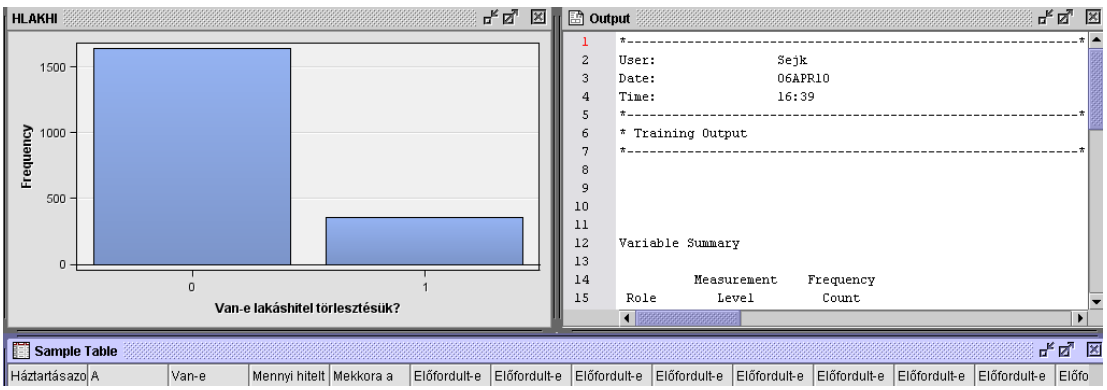
9. ábra Célváltozó létrehozása

A biztonság végett tekintsük, meg a forrásállományunkat, vagy az illesztés esetében a (**Merge**) csomópont és válasszuk a *változó szerkesztés* parancsot. Itt keressük ki a *„hitel”* változó sorát, a beállításoknak azonosnak kell lenni az alábbi ábrán láthatóakkal. (10. ábra)



10. ábra Célváltozó

Ezek után kössünk az illesztés csomópont után egy *grafikus varázsló* (**Graph Explorer)** csomópontot. Válasszuk a változó parancsot a képernyő bal oldalán található beállítások panelen. Ezek után megjelenik a változók listája. Itt a használat (**Use**) szerepet állítsuk **No**-ra, kivétel a célváltozó esetén. Ha megvagyunk futassuk a csomópontot.



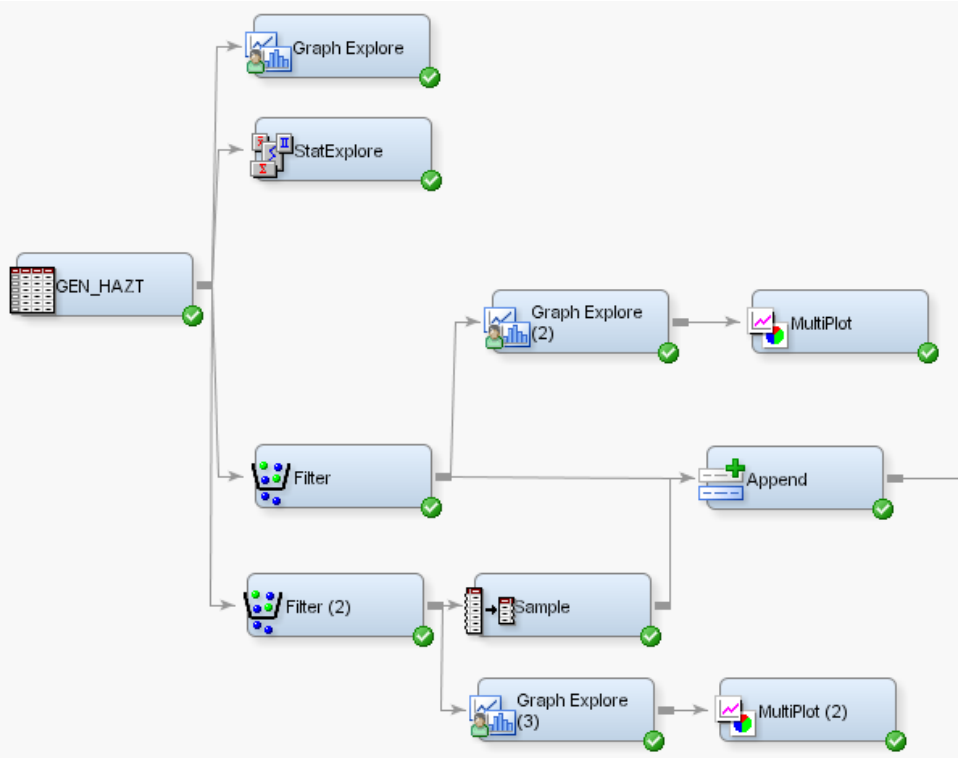
11. ábra Célváltozó eloszlása

Az ábrát megfigyelve szembetűnő lehet, hogy nagyok a különbségek. Ez az eset akkor is igaz, ha a futtatás előtt a beállítások panelen a mintavételezést véletlenre állítjuk és méretnek maximumot adunk meg.

Ebben az esetben több dolgot tehetünk.

- Módosíthatjuk a tanuló algoritmus kiértékelő statisztikáját (másik voltozót választunk)
- Alkalmazunk egy profitmátrixot, így súlyokat rendelünk az egyes esetekhez
- Célesemények túldúsítása (túlmintavételezés)

Ebben a példában viszont egy teljesen más megoldást választunk. Kialakítunk egy rész mintát és ezen tanítjuk a modellünket. Ennek a hátránya abban rejlik, hogy nem fog rendelkezésre állni teszthalmaz, tehát egy másik adatállományra lesz szükség, amin tesztelhetjük a modellünket.



12. ábra Adatfelosztás megvalósítása

A forrásállományunkhoz kapcsoljunk két filter csomópontot. (12. ábra) Ezeknek a célja, hogy leválogassuk a célváltozó két lehetséges értékének a halmazait. Az első filter csomópont szűri a célváltozó „1” értékű sorait.

Első lépésben adjuk meg a szűrés beállításait:

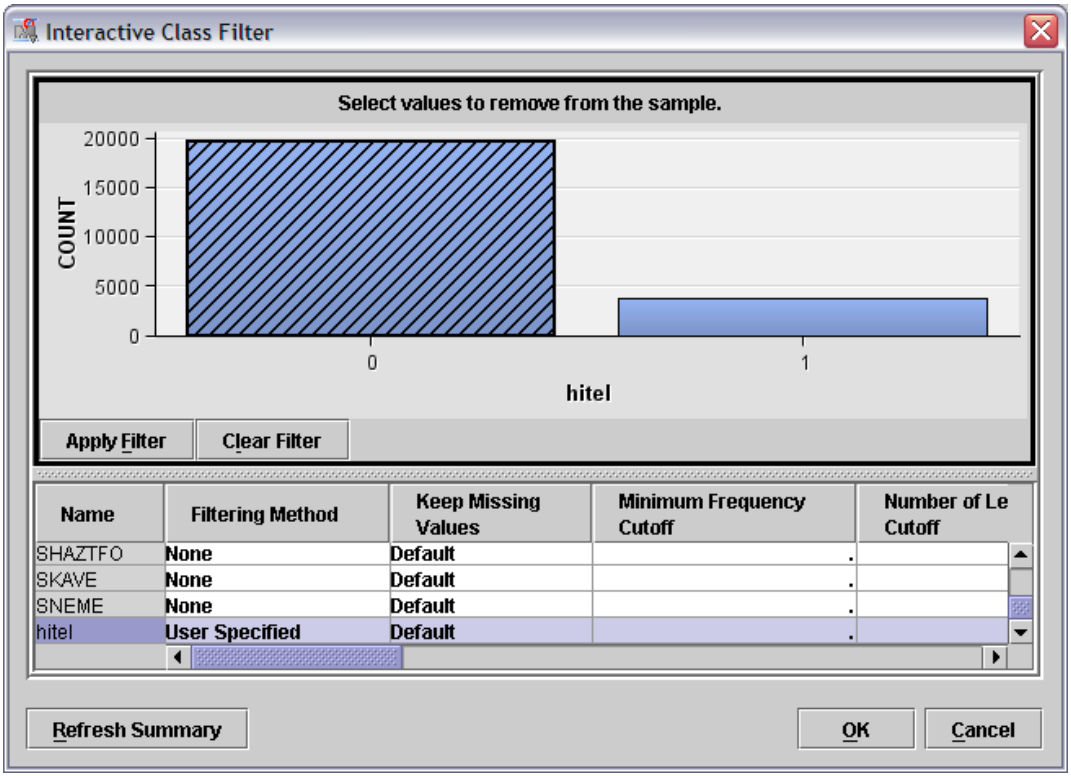
- A teljes adathalmazon végezzük a szűrést
- Mivel kezdetben is kevés „1”-es értékű sor áll a rendelkezésre így megtartjuk a hiányzó értékeket is, hiszen ezek elvetésével tovább csökkentenénk a megfigyelések számát.
- Nem szükséges normalizálni

Property	Value
General	
Node ID	Filter
Imported Data	
Exported Data	
Notes	
Train	
Export Table	Filtered
Tables to Filter	All Data Sets
Class Variables	
Class Variables	
Default Filtering Metho	Rare Values (Percenta
Keep Missing Values	Yes
Normalized Values	No
Minimum Frequency Cu	1
Minimum Cutoff for Peri	0.01
Maximum Number of L	2
Interval Variables	
Interval Variables	
Default Filtering Metho	Standard Deviations fro
Keep Missing Values	Yes
Tuning Parameters	
Score	
Create score code	Yes

13. ábra Filter beállításai

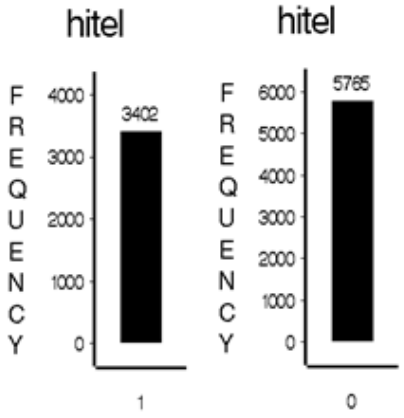
Ezek után válasszuk a beállítások közül a *változók* menüpontot. Ekkor az alábbi képernyő áll a rendelkezésünkre. (14. ábra)

A „*hitel*” változó kivételével az összes változó szűrési metódusát állítsuk **None**-ra. Majd a hitel változó esetén definiáljuk a kritériumokat. (**Generate Summary**) Mivel az 1-es értékeket akarjuk leszűrni, így jelöljük ki a eltávolítani kívánt 0-s oszlopot, majd klikk az elfogadásra. (**Apply Filter**)



14. ábra Szűrés meghatározása

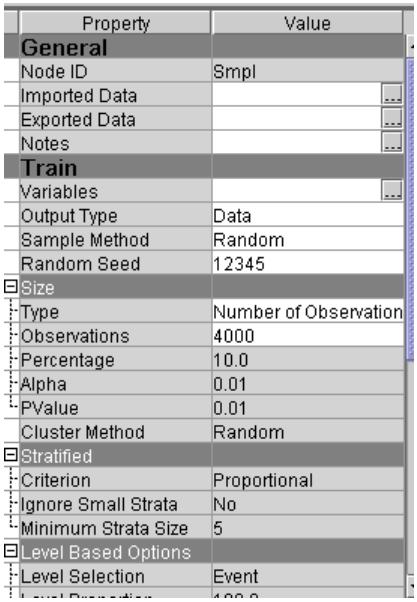
Futtassuk a csomópontot és kössünk utána egy grafikon varázslót (**Graph Explorer**). Ezen a csomóponton is hajtsuk végre a következő beállításokat a bal oldali panel segítségével. Állítsuk a méretet maximumra, majd válasszuk a *változók* menüpontot. A változókat a „*hitel*” kivételével állítsuk nem használtra. Futtassuk a csomópontot. Az áttekinthetőség javítása érdekében kössük egy **MultiPlot** csomópontot. Az eredményeket a (15. ábra bal oldali grafikonja mutatja)



15. ábra A célváltozó egyes értékeinek eloszlása

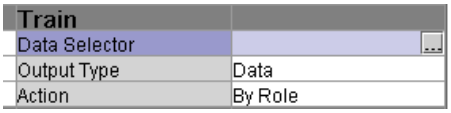
Következő lépésben kössünk a 12. ábra szerint egy másik filter csomópontot. Ezen hajtsuk végre az előző lépéssorozatot, hogy most a 0 értékűeket válogathassuk le. Azzal az eltéréssel, hogy a hiányzó értékeket eldobhatjuk, hiszen még így is magasabb az elemek száma, mint az 1-es értékek esetén. Az eredményeket a (15. ábra jobb oldali grafikonja mutatja).

Most, hogy külön rendelkezésünkre állnak a halmazok eljött az idő, hogy összefésüljük őket. A jó modell elkészítéséhez 8-10 ezer megfigyelés elég és a célváltozó értékeinek sem kell pont azonosnak lenni. Így akár össze is fűzhetnénk a két halmazt, azonban még előtte alkalmazzunk egy mintavételezési csomópontot. (**Sampling**) Alkalmazzuk az alábbi beállításokat.



16. ábra Mintavételezés

A kiindulási mintahalmaz generálásához fűzzük össze a két csomópontot az (**Append**) csomóponttal. A beállításokat a 17. ábra mutatja.

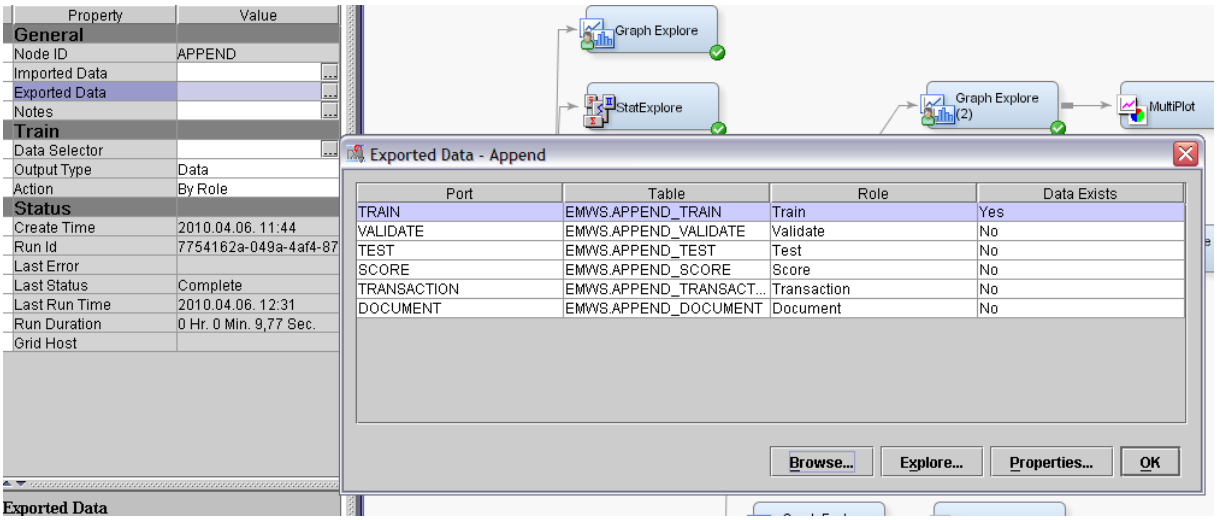


17. ábra Összefűzés beállításai

A generált adatállomány megfigyeléseit az összefűzés csomópont bal oldali beállítások panelján az **Exported Data** menüpont alatt találjuk. Ezt a pontot választva listázásra kerülnek az adatállományok. Jelen esetben az összes megfigyelés a tanuló (**train**) állományt képezi.

A tanuló állományon válasszuk a böngészés gombot (**Browse**) (18. ábra) Ekkor az adatállomány összes megfigyelése megtekinthető.

Észrevehetjük, hogy az összefűzés következtében egy új *_dataobs_* változóval bővült a lista, azonban erre az attribútumra az elemzés szempontjából nincs szükség, így egy elvetés (**drop**) csomópontot az összefűzés után kötve elvethetjük a kívánt változókat. Esetünkben a _dataobs_ változót.



18. ábra Megfigyelések megtekintése

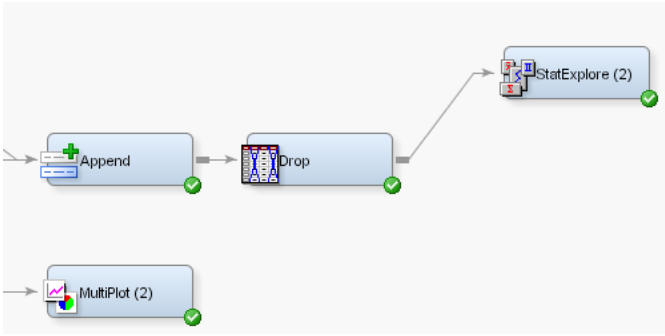
4. Adatmódosítás

A továbbiakban adatmódosítási lépéseket ismertetünk. Ezek lényeges részét képezik a helyes eredmény elérésének.

4.1. Hiányzó értékek pótlása

Az adatelőkészítés vagy módosítás egyik legfontosabb lépése a hiányzó értékek pótlása. Ennek azért van nagy jelentősége, mert a hiányzó értékeket bizonyos tanuló algoritmusok (lineáris, logisztikus regresszió) nem kezelik és így ezeket a megfigyeléseket nem veszik figyelembe, így fontos információkat veszíthetünk.

Annak feltárásában, hogy bizonyos változók mennyi hiányzó értékkel rendelkeznek, a statisztikavarázsló (**StatExplorer**) nagy segítséget jelenthet. Kössük ezt a 19. ábra szerint.



19. ábra Hiányzó értékek feltárása

A csomópont futtatása után az alábbi eredményeket kaptuk. (20. ábra - 22. ábra)

Variable	ROLE	Mean	Std. Deviation	Non Missing	Missing	Minimum	Median	Maximum
ESNJ	INPUT	804802.90	451117.22	6058	1344	-228039	712870	3000000
HAERT	INPUT	9.29	6.44	6901	501	1	8	42
HCTSZGA	INPUT	1.08	0.28	250	7152	1	1	2
HLAKFT	INPUT	1845.24	2207.94	3402	4000	50	1000	12000

20. ábra Intervallum változók statisztikai mérőszámai

Variable	Role	Formatted Value	Frequency Count	Percent
hitel	TARGET	0	4000	54.0394
hitel	TARGET	1	3402	45.9606

21. ábra Célváltozó szegmensek szerinti eloszlása

Variable	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
HAZON	INPUT	3954	0	030352200430551	0.09	032106900030261	0.09
HEPTEL	INPUT	2	0	0	97.54	1	2.46
HKOMP	INPUT	2	0	0	94.46	1	5.54

22. ábra Osztályozó változók statisztikai mérőszámai

Az eredményeket megtekintve láthatjuk, hogy van bőven tennivalónk. A legegyszerűbb módosítás az osztályozó változókat tekintve tehető. Egészen pontosan a *HAZON* változóra vonatkozóan. Ez a változó azonosítókat tartalmaz így minden megfigyelés esetén különböző. Ebből adódóan nincs előrejelző ereje a modellt tekintve. A változó szerepét (**Role**) kell módosítani azonosító (**ID**) típusra a forrásállomány esetén. Segítséget nyújthat a beállításban a 10. ábra.

Tekintsük az intervallum változókat. Az ábrán az első változó a *személy éves nettó jövedelme* (ESNJ). Ez a változó tartalmaz negatív értéket, azonban azt tudjuk, hogy ez a változó nem vehet fel negatív értéket. Továbbá ez a változó számos hiányzó értéket is tartalmaz. Ezeket kezelni kell.

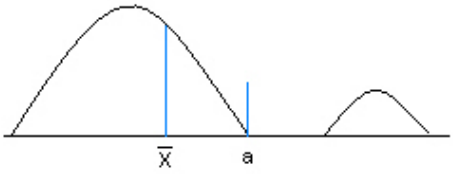
Több lehetőség is adott, hogy előbb a negatív majd a hiányzó értékeket kezeljük vagy fordítva. Ebben a tanulmányban előbb a negatív értékeket helyettesítjük az átlaggal, majd kezeljük a hiányzó értékeket. Ez kevésbé erőforrás igényes, mint a másik lehetőség.

A *Csere* csomópontot (**Replacement**) a legtöbb esetben a *Pótlás* csomóponttal (**Impute**) együtt használják. Ez a csomópont a *Módosítás* eszköztárban található.

Elméleti áttekintés

Funkcióját tekintve a legegyszerűbb magyarázatot a 23. ábra mutatja. Legyen adott egy változónk, mely ilyen eloszlással rendelkezik és tartalmaz hiányzó értékeket. A cél a hiányzó értékek pótlása, de előtte alkalmazni, kell a *Csere* csomópontot. Ez azért fontos, mert így az a-nál nagyobb értékeket nem kell törölnünk, így nem veszítünk egyéb információkat, hanem csak alkalmazzuk a csomópontot és ezeket az értékeket az X átlaggal pótoljuk.

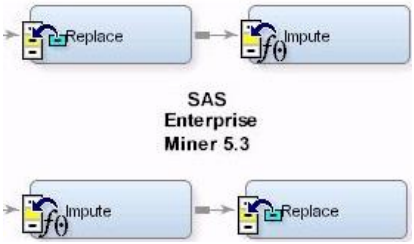
Ezek után már foglalkozhatunk a hiányzó értékek pótlásával, hiszen egy jobban központosult eloszlás áll rendelkezésünkre.



23. ábra Csere csomópont alkalmazása

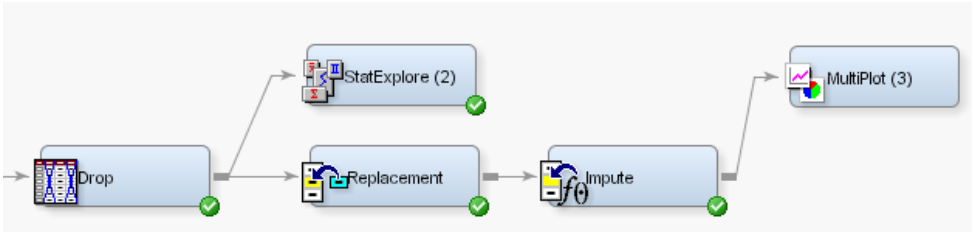
Természetesen az is megoldható, hogy előbb pótoljuk a hiányzó értékeket, majd ezután alkalmazzuk a *Csere* csomópontot. (A kapott eredmény nem azonos az előzővel)

Azt, hogy melyik esetet alkalmazzuk, nagymértékben befolyásolja a változó leíró statisztikai jellemzői.



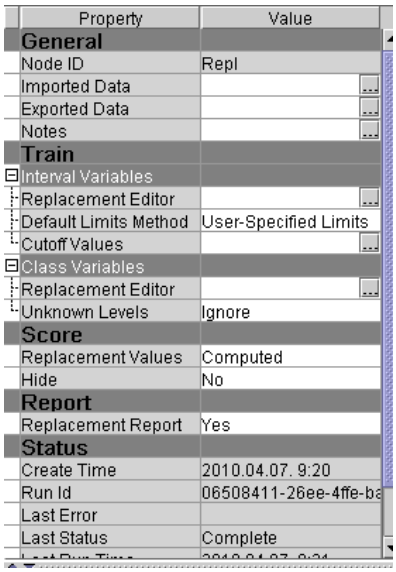
24. ábra A pótlás és a csere két változata

Térjünk vissza a konkrét projektünkhöz és kezeljük az anomáliákat. Ehhez kössük a *Csere* és a *Pótlás* csomópontokat a 25. ábra szerint.



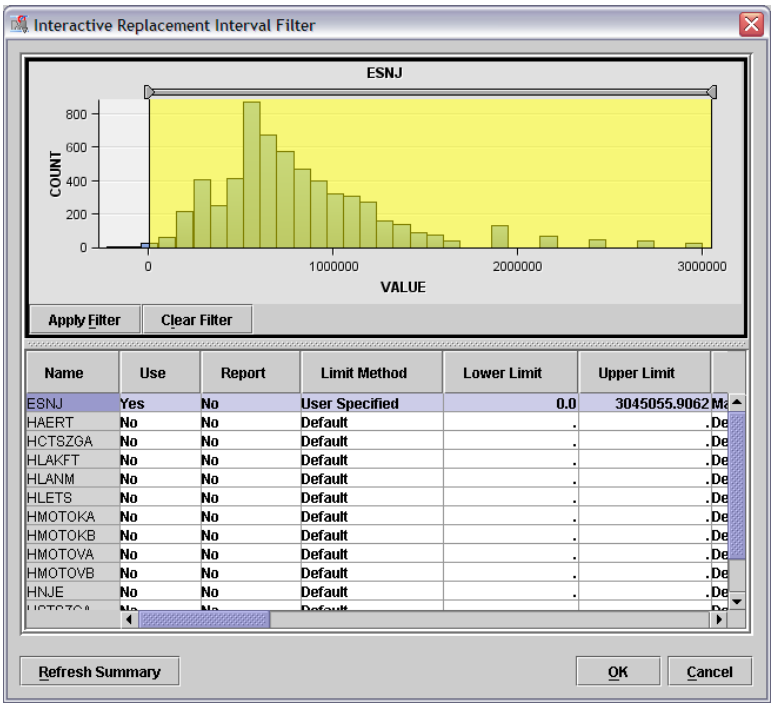
25. ábra Csere és Pótlás

A *Csere* csomópont a nullánál kisebb nettó jövedelemmel rendelkező egyedeket képezi le az általunk megadott tartományba. A beállításokat a bal oldali panelen tehetjük meg. (26. ábra)



26. ábra Csere csomópont beállításai

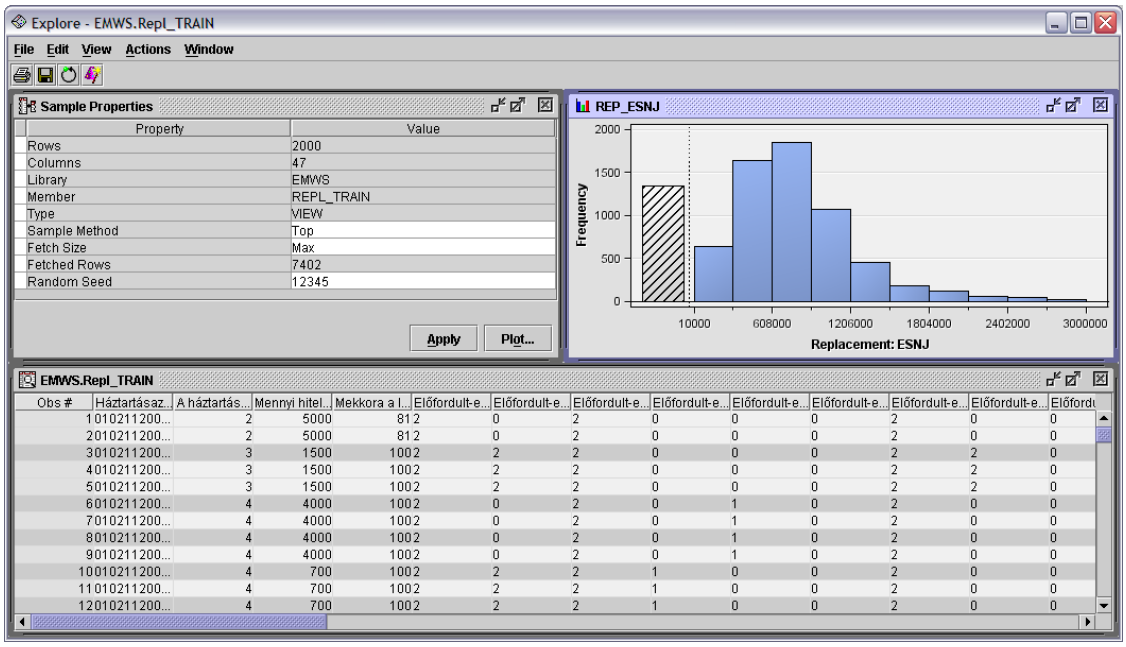
Mivel a változtatni kívánt változó intervallum típusú, így ezeket a beállításokat kell módosítani. Állítsuk be metódusnak: Felhasználó *által definiált határok* (**User-Specified Limits**). Majd válasszuk a *csere szerkesztőt* (**Replacement Editor**).



27. ábra Csereszerkesztő

A fenti ábrát tekintve láthatjuk, hogy az alsó határ a 0 érték, míg a felső az alapértelmezett. Ezeket a beállításokat az ábra tetején található csúszka segítségével is megadhatjuk. Továbbá itt van lehetőség a cseretartomány megadására is. Ezt most nem változtassuk.

Ezek után alkalmazzuk a pótlás csomópontot (**Impute**) a 25. ábra szerint.



28. ábra Nettó jövedelem a csere után

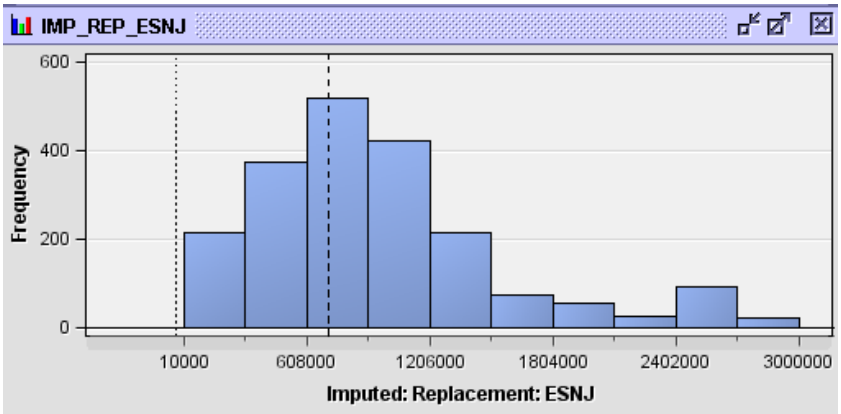
Válasszuk a Pótlás csomópont **Imported Data** menüpontot, hogy megtekintsük a kívánt változó eloszlását. Válasszuk ki a *REP_ESNJ* változót, ez az előző pont módosítása által generált változó. Válasszuk a *Feltárás* (**Explore**) menüpontot. A Miner figyelmeztet minket, hogy csak az első 2000 sor kerül ábrázolásra. Ha ezen változtatni szeretnénk, akkor válasszuk a maximumot méretnek. (28. ábra)

Jó látható, ha az egeret z előbbi grafikon fehér oszlopa fölé mozgatjuk, hogy megjelenik a feltételben, hogy ezek a megfigyelések a hiányzó értékek. Ezeket fogjuk ebben a pontban pótolni.

A pótlás csomópont bal oldali beállításpaneljén adjuk meg az intervallum típusú változók alapmetódusának az eloszlás változatot (**Distribution**), majd válasszuk a változók menüpontot. Itt az előbb már említett *REP_ESNJ* változót engedélyezzük, míg a többit zárjuk ki.

Ezek után futtassuk a csomópontot.

A kapott eredmény megtekintéséhez kössünk egy MultiPlot csomópontot az előző csomópont után és ábrázoljuk az *IMP_REP_ESNJ* változót. A pontos eredmény megtekintéséhez itt is a maximum méretet válasszuk. (29. ábra)



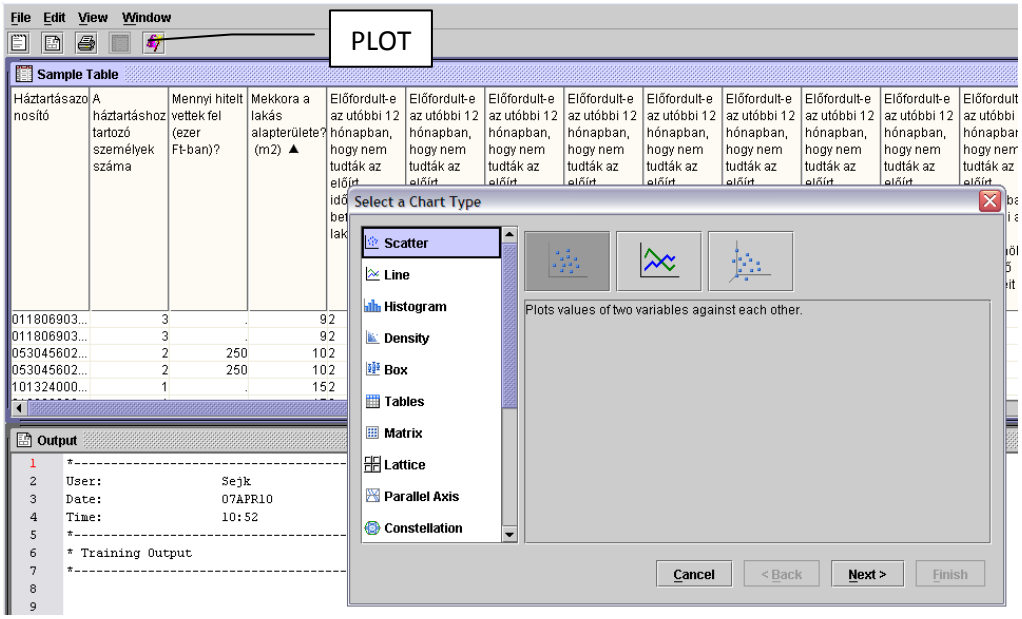
29. ábra A pótlás utáni eredmény

A 20. ábra pontos tanulmányozása esetén észrevehetjük, hogy hasonló probléma adódik a *Háztartások nettó jövedelme HNJE* változó esetén is. Ezen a változón is végezzük el a fenti lépéseket.

Megjegyzés: A HNJE változó a háztartások adatállományból származik és ezek a változók nem tartalmaznak negatív értéket, így a pótlás csomópont (Impute) nem fog végrehajtódni, tehát a generált változó *REP_HNJE* néven fog szerepelni az adatállományban.

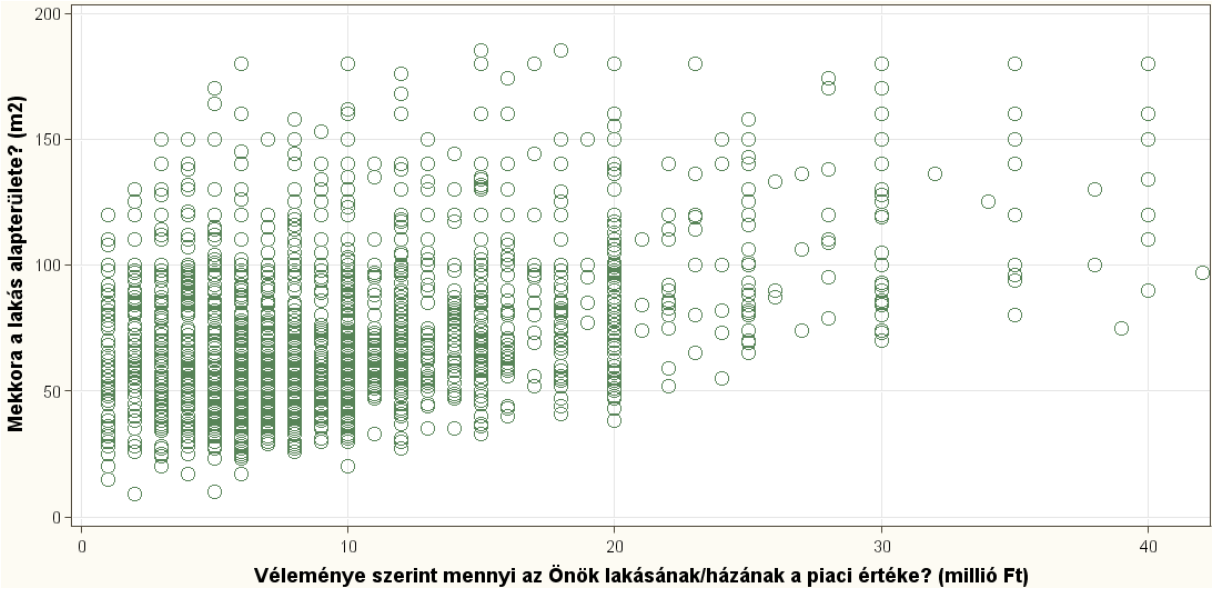
További hiányzó értékek pótlása. Az egyik fontos változó a *HAERT - Véleménye szerint mennyi az Önök lakásának/házának a piaci értéke? (millió Ft)*. Ezeket a hiányzó értékeket a *HLANM - Mekkora a lakás alapterülete? (m2)* segítségével pótolhatjuk.

Alkalmazzunk egy *Grafikon varázslót*, majd a futtatása után válasszuk a **Plot** funkciót és hozzunk létre egy *felhődiagramot* (**Scatter plot**). (*30.* ábra)



30. ábra Grafikon létrehozása

A kapott grafikont a 31. ábra szemlélteti.



31. ábra Két változó felhődiagram

A 31 ábrát tekintve megfigyelhetjük a két változó kapcsolatát. A hiányzó értékeket döntési fával fogjuk pótolni. Az alábbi ábra az (**Impute**) csomópont beállításait mutatja. Válasszuk ki a használni kívánt változókat. A *HAERT* változó lesz a célváltozó. A folyamatnak állítsuk be a fát (**Tree**). Meglehetősen kevés változóra tudunk fát építeni, de itt a cél az eljárás megismerése és a finomság csak másodlagos szempont. Tehát válasszuk ki a HLANM, HLTIP változókat a fa megalkotásához (**Use Tree**).

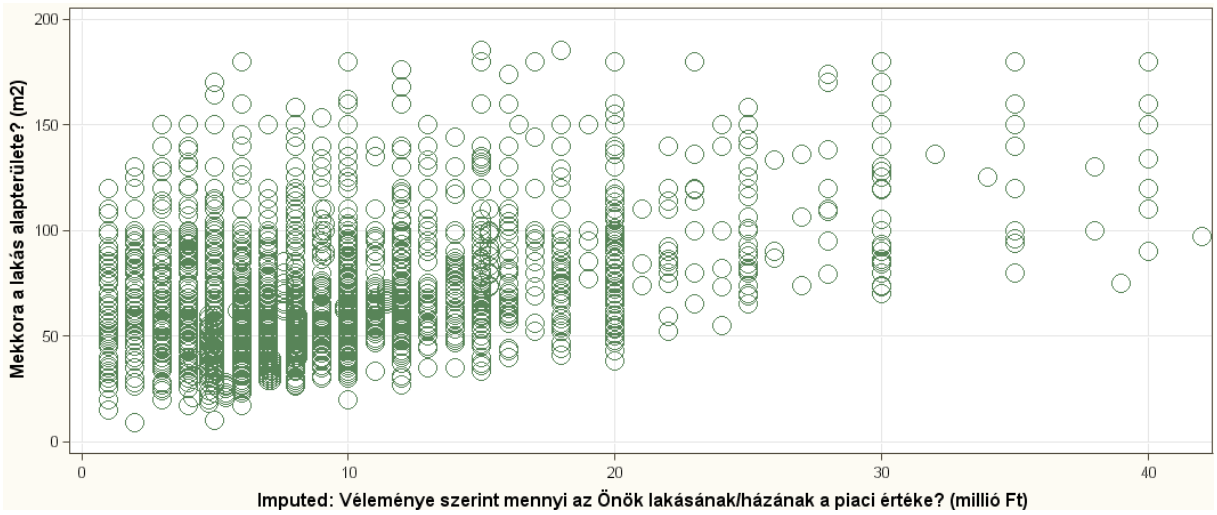
Variables - Impt2

(none) not Equal to

Name	Use	Method	Use Tree	Role	Level	Type
ESNJ	No	Default	Default	Input	Interval	Numeric
HAERT	Yes	Tree	Yes	Input	Interval	Numeric
HCTSZGA	No	Default	Default	Input	Interval	Numeric
HEPTEL	No	Default	Default	Input	Nominal	Character
HKOMP	No	Default	Default	Input	Nominal	Character
HLAKA	No	Default	Default	Input	Nominal	Character
HLAKFT	No	Default	Default	Input	Interval	Numeric
HLANM	Default	None	Yes	Input	Interval	Numeric
HLBER	No	Default	Default	Input	Nominal	Character
HLETS	No	None	Default	Input	Interval	Numeric
HLFUT	No	Default	Default	Input	Nominal	Character
HLGAZF	No	Default	Default	Input	Nominal	Character
HLHIT	No	Default	Default	Input	Nominal	Character
HLKTG	No	Default	Default	Input	Nominal	Character
HLKTR	No	Default	Default	Input	Nominal	Character
HLTIP	Default	Default	Yes	Input	Nominal	Character

32. ábra Hiányzó értékek pótlása fával

Futtassuk a csomópontunkat. Az eredményeket szemléltethetjük grafikonvarázsló segítségével felhődiagramon. (33. ábra)



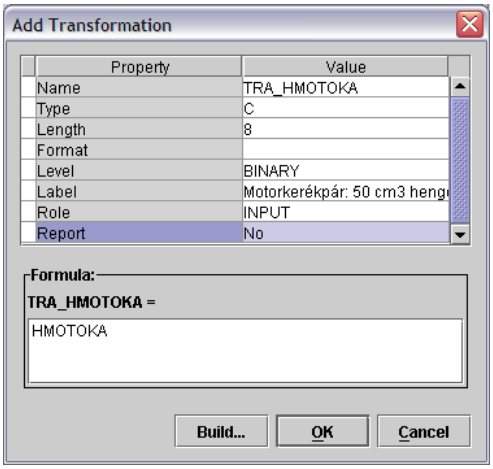
33. ábra Pótlás utáni felhődiagram

Ha az eredményeket táblázat segítségével megfigyeljük, akkor láthatjuk, hogy a hiányzó értékek nem egész számok lettek. Így szükséges ezeket egész értékekre transzformálni. Az adat transzformációt már alkalmaztuk a 3.1 fejezetben, így itt már csak a paramétereket adjuk meg.

Név: TRA_HEART
Címke: Véleménye szerint mennyi az Önök lakásának/házának a piaci értéke? (millió Ft)
Függvény: CEIL(IMP_HAERT)

Következő lépés, hogy az adatállomány tartalmaz olyan változókat, melyek nem lettek kitöltve, mivel ebben az esetben azt jelentik, hogy az adott egyeden nem birtokol a változónak megfelelő tárgyat. Például a HMOTOKA változó azt jelenti, hogy az adott egyednek hány darab 50 cm^3-es motorkerékpárral rendelkezik. Ha ez nincs kitöltve, akkor az ebben az esetben nem hibát, hanem nullát jelent, ezeket kell lekezelnünk.

Ezeket a változókat a lekezelés előtt alakítsuk karakteres bináris változóvá. Ehhez a lépéshez szintén adattarnszformációt kell végrehajtani. A menetét nem közöljük, mivel az megegyezik a korábbiakkal, de a beállításokat az alábbi ábra szemlélteti. (34. ábra)



34. ábra Adattranszformáció

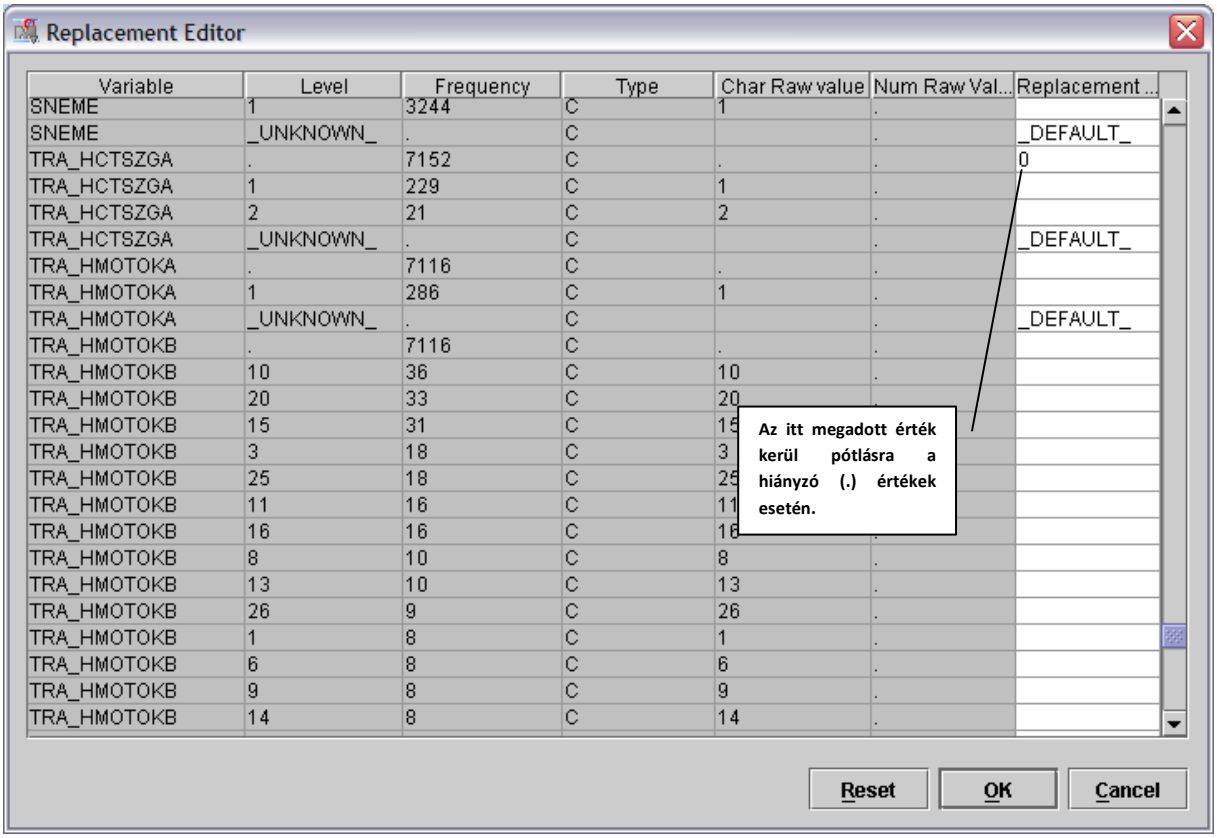
További módosításra szoruló változók, melyeken a fenti átalakításokat kell elvégezni:

- ***HMOTOKA, HMOTOKB, HMOTOKC, HMOTOVA, HMOTOVB, HMOTOVC, HSTSZGA, HSTSZGB, HSTSZGC, HCTSZGA***

NAME	...	...	...	LEVE
TRA_HMOTOKA	C	8		BINARY
TRA_HMOTOKB	C	8		NOMINA
TRA_HMOTOKC	C	8		NOMINA
TRA_HMOTOVA	C	8		BINARY
TRA_HMOTOVB	C	8		NOMINA
TRA_HMOTOVC	C	8		NOMINA
TRA_HSTSZGA	C	8		NOMINA
TRA_HSTSZGB	C	8		NOMINA
TRA_HSTSZGC	C	8		NOMINA
TRA_HCTSZGA	C	8		NOMINA

35. ábra Transzformált változók

Ezek után helyezzünk egy *Pótlás* (**Replace**) csomópontot a transzformációs csomópont után. Majd a pótlás csomópont bal oldali osztályozó változó menüjéből válasszuk a *Pótlás szerkesztőt*.



Variable	Level	Frequency	Type	Char Raw value	Num Raw Val...	Replacement ...
SNEME	1	3244	C	1	.	
SNEME	_UNKNOWN_	.	C		.	_DEFAULT_
TRA_HCTSZGA	.	7152	C	.	.	0
TRA_HCTSZGA	1	229	C	1	.	
TRA_HCTSZGA	2	21	C	2	.	
TRA_HCTSZGA	_UNKNOWN_	.	C		.	_DEFAULT_
TRA_HMOTOKA	.	7116	C	.	.	
TRA_HMOTOKA	1	286	C	1	.	
TRA_HMOTOKA	_UNKNOWN_	.	C		.	_DEFAULT_
TRA_HMOTOKB	.	7116	C	.	.	
TRA_HMOTOKB	10	36	C	10	.	
TRA_HMOTOKB	20	33	C	20	.	
TRA_HMOTOKB	15	31	C	15	.	
TRA_HMOTOKB	3	18	C	3	.	
TRA_HMOTOKB	25	18	C	25	.	
TRA_HMOTOKB	11	16	C	11	.	
TRA_HMOTOKB	16	16	C	16	.	
TRA_HMOTOKB	8	10	C	8	.	
TRA_HMOTOKB	13	10	C	13	.	
TRA_HMOTOKB	26	9	C	26	.	
TRA_HMOTOKB	1	8	C	1	.	
TRA_HMOTOKB	6	8	C	6	.	
TRA_HMOTOKB	9	8	C	9	.	
TRA_HMOTOKB	14	8	C	14	.	

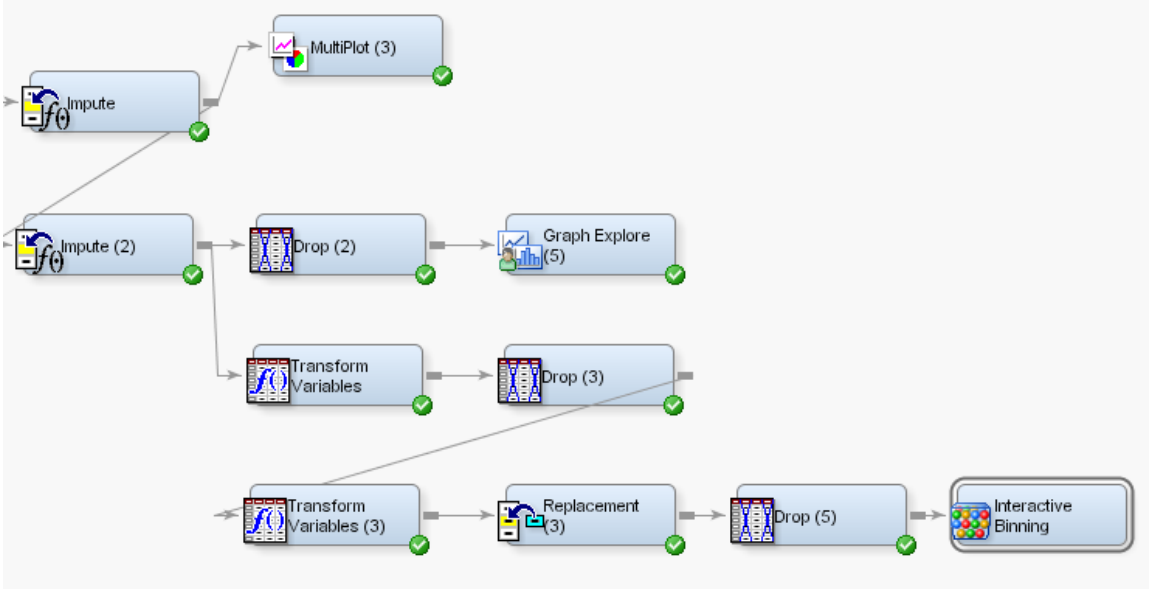
36. ábra Pótlás a megadott értékkel

A létrehozott TRA_ kezdetű változók esetén pótoljuk a hiányzó értékeket (.) nullával. (36. ábra) Ezek után futtassuk a csomópontot.

Az eredmények megtekintése után dobjuk el az eredeti változókat, amikből az új értékeket tarnszformáltuk. Ehhez alkalmazzuk az *Elvetés* (**Drop**) csomópontot. A bal oldali beállítások panelt tekintve megfigyelhetjük, hogy alapértelmezett beállítás szerint a rejtett, visszavont értékek törlésre vannak állítva, ez azért van így mert a Miner azokat az eredeti változókat, amiken bármilyen átalakításokat végeztünk, átállítódnak **Rejected**-re. Így csak annyi a feladatunk, hogy átállítsuk a panel legfelső, *Eltávolítás a táblából*, parancs értékét igenre. Továbbá a változók közül állítsuk be a REP_SFEOR változót eldobásra, és futtassuk a csomópontot.

4.2. A binnelés technikája

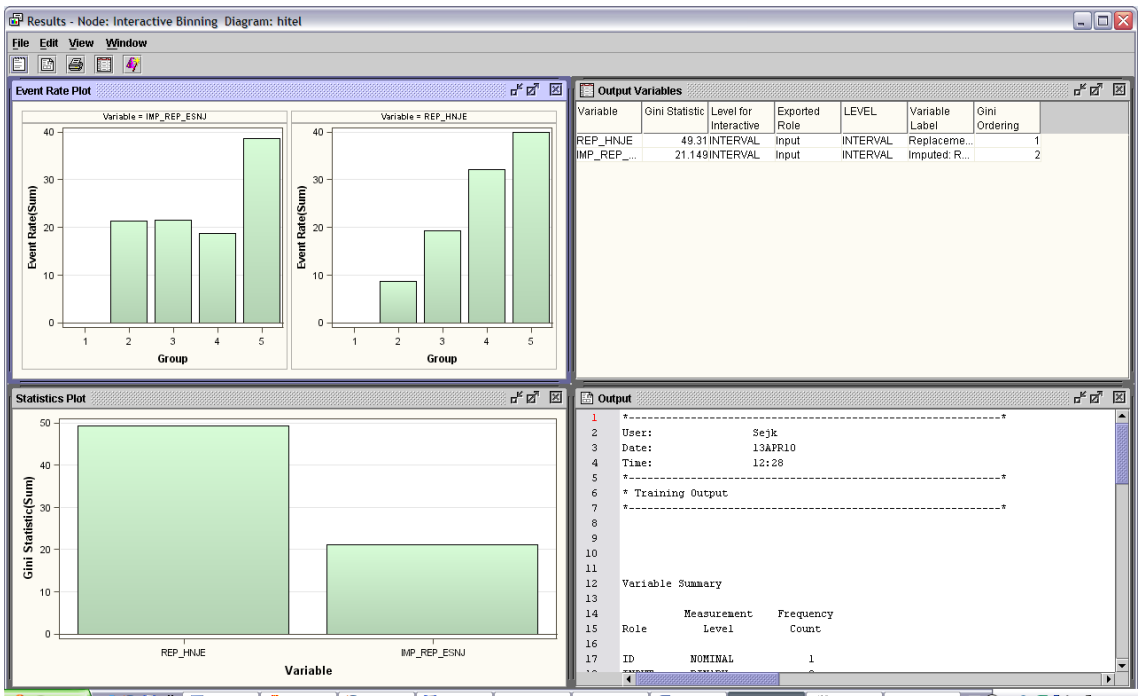
Azt a technikát, amikor intervallum típusú változókból csoportokat képezünk, binnelésnek nevezzük. A Miner esetén egy konkrát csomópont (**Interactive Binning**) megvalósítható ez az átalakítás.



37. ábra Folyamatábra a binnelés-sel kiegészítve

A beállítások előtt ezt a csomópontot futtatni kell.

A beállítások panelen válasszuk a változók menüpontot és válasszuk ki a *REP_HNJE és REP_ESNJ* változókat, amiket fel szeretnénk osztani a panelen megadott számú csoportba (esetemben ez 4). Fontos továbbá megadni egy bináris célváltozót is, ami alapján a Miner elvégzi a műveletet, így a változók menüpontban a REP_hitel célváltozót is ki kell választanunk. (38)



38. ábra Csoportképzés

5. Modellalkotás

A modellépítés célja a bemeneti változók alapján a célváltozót eredményül adó függvény, eljárás meghatározása. Általában akkor alkalmazunk adatbányászati módszert, amikor a transzformáció nem egyértelmű, hanem komplex. Érdemes azonban azt is észben tartani, hogy sok olyan adatbányászati feladatot tűzünk ki, ahol az adathalmaz nem tartalmazza a célváltozó meghatározásához szükséges információkat. Ilyen esetekben a base line módszertől alig lesz jobb a legjobb modell is.

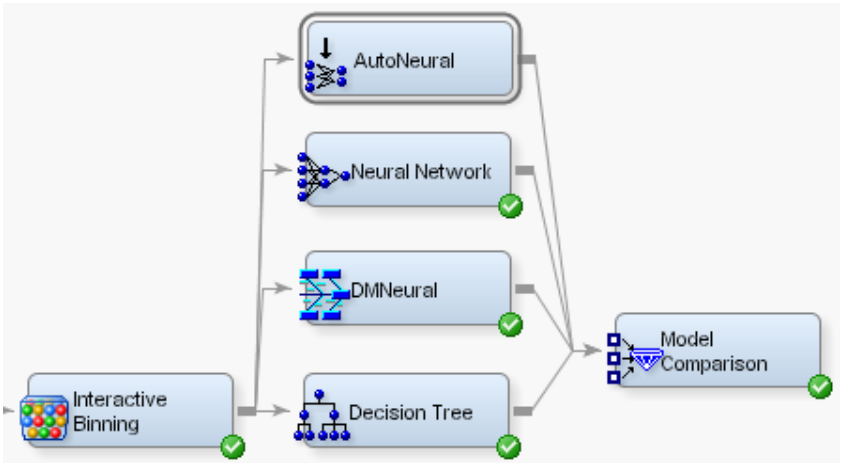
Tapasztalatok azt mutatják, hogy az alapbeállításokkal futtatott modellezéseknél ritkán hoznak sokkal jobb eredményt a további paraméterezések (inkább az új változó transzformációi hoznak ugrásokat jóságban).

5.1. Decision tree – Döntési fa

A döntési fa elsődleges előnye, hogy összefüggései jól átláthatóak, ami segíti együttműködésünket az adatgazdákkal. Bináris fáról beszélünk, ha a fának az elágazásokban kettő ága van.

A SAS döntési fa megoldásának előnye a hiányzó adatok kezelése. Amikor az elágazáshoz tartozó változó üres, akkor más változóhoz tartozó tartalék szabályt alkalmaz. A tartalék szabályok négyszintűek.

Kössünk egy *döntési fa* (**Decision tree**) csomópontot, az előző pontban létrehozott „interaktív binnelés” csomópontunkhoz. (39. ábra)



39. ábra Döntési fa, neurális háló modellek és összehasonlításuk

A csomópontunk baloldali tulajdonság paneljén adjuk meg az alábbi tulajdonságokat.

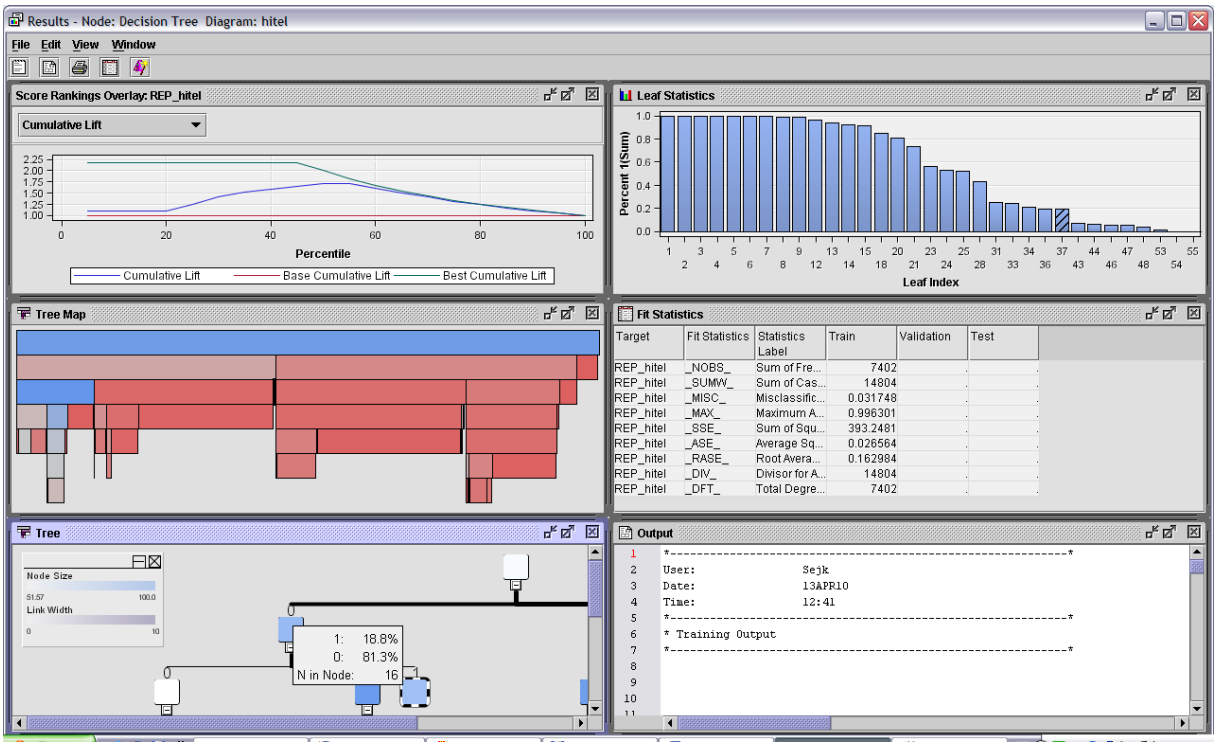
Tulajdonságok →

a. maximum branch: 4, max. négyfele ágazhat el egy node
b. leafe size: 8, minimum 8 megfigyelés legyen egy levélben
c. maximum depth: 10 maximális út az irányított fában
d. Number of Surrogate Rules: 4 hiányzó adat esetén négyszintű tartalék szabálykészlet

Futtassuk csomópontot és tekintsük meg az eredményt.

Az 40. ábra bal oldali ablakai a fa felépítését mutatják. Lentebb képileg is, fentebb az egyes levelekben a célváltozó szerinti megfigyelések számát láthatjuk. Értelemszerűen a levélhez tartozó döntés a levélben nagyobb arányban lévőkhöz típusa lesz.

A Tree Map a Leaf stat.-hoz hasonlóan jeleníti meg nem csak a leveleket, de az összes csomópontot. A téglalap színe a 0/1 arányt tükrözi. Figyeljük meg, a Tree Map-on kijelöltünk egy téglalapot (fölötte ott vannak kis sárga ablakban a számok) és mind a Tree-n, mind a Leaf Stat.-on megjelent a hozzá tartozó rész.



40. ábra Döntési fa eredménye

5.2. Neural Network / DNNeurál – Neurális háló

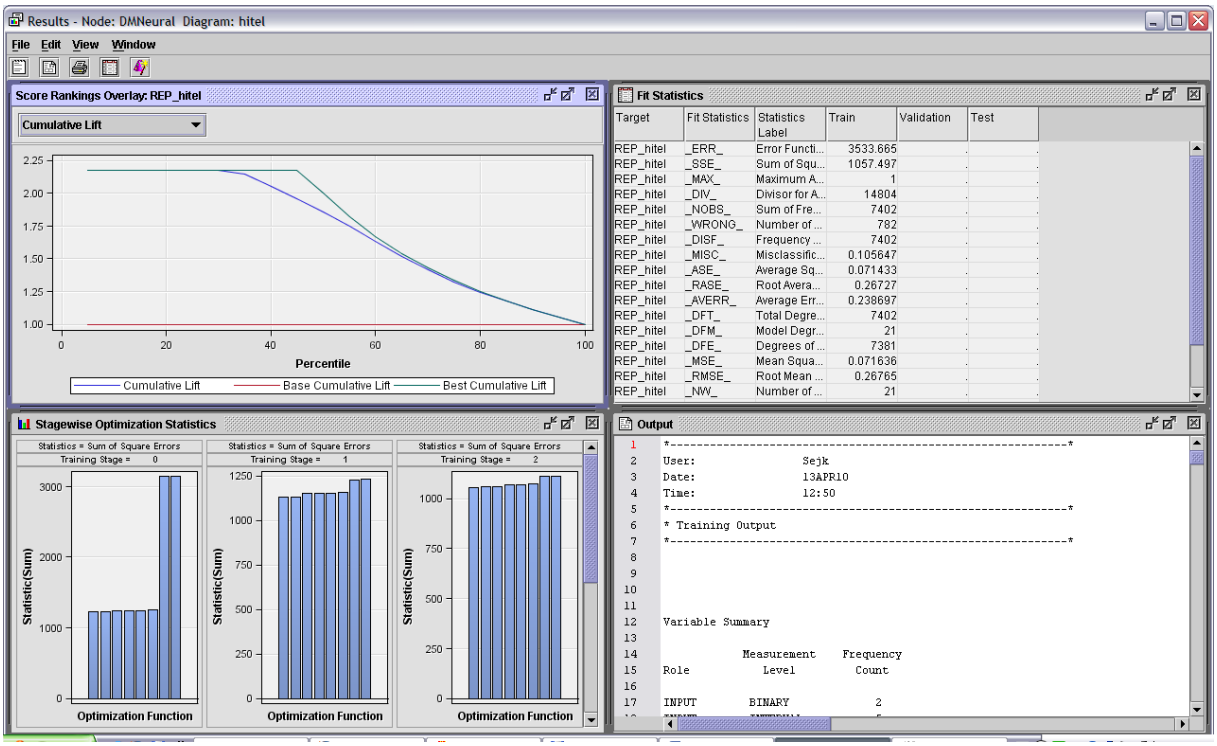
Kössünk egy neurális háló csomópontot a 39. ábra szerint.

A neurális hálózatok hatékonyan kezelnek nem lineáris összefüggéseket, kiemelt alkalmazási területük hitel-kockázat elemzés, direkt marketing és eladás-előrejelzés.

1. Egy Neural network node-ot kapcsoljunk a Transform node után.

2. Tulajdonságok → **Network** és megnyílik a network ablak, itt:
   a. **Direct Connection** property: **Yes.**
   b. **Number of Hidden Units** property: **5.**

3. Futtassuk a node-ot és tekintsük meg az eredményt.



41. ábra Neurális háló eredménye

5.3. AutoNeural

Ez a csomópont gyakorlatilag az előző pontban futtatott csomópontot futtatja többször, miközben optimalizálja annak paramétereit. A nagy számítási igény miatt ezt a csomópontot most egy csökkentett változó halmazon futtatjuk, a Variable selection 48 input változóból 6-ot adott át az AutoNeurálnak. Ennek ellenére is az AutoNeurál 5,5 percig futott, míg a Neurál 1,5 percig a 48 változón.

AutoNeural csomópont tulajdonságok →

Architecture: **Cascade**
Train Action: Search

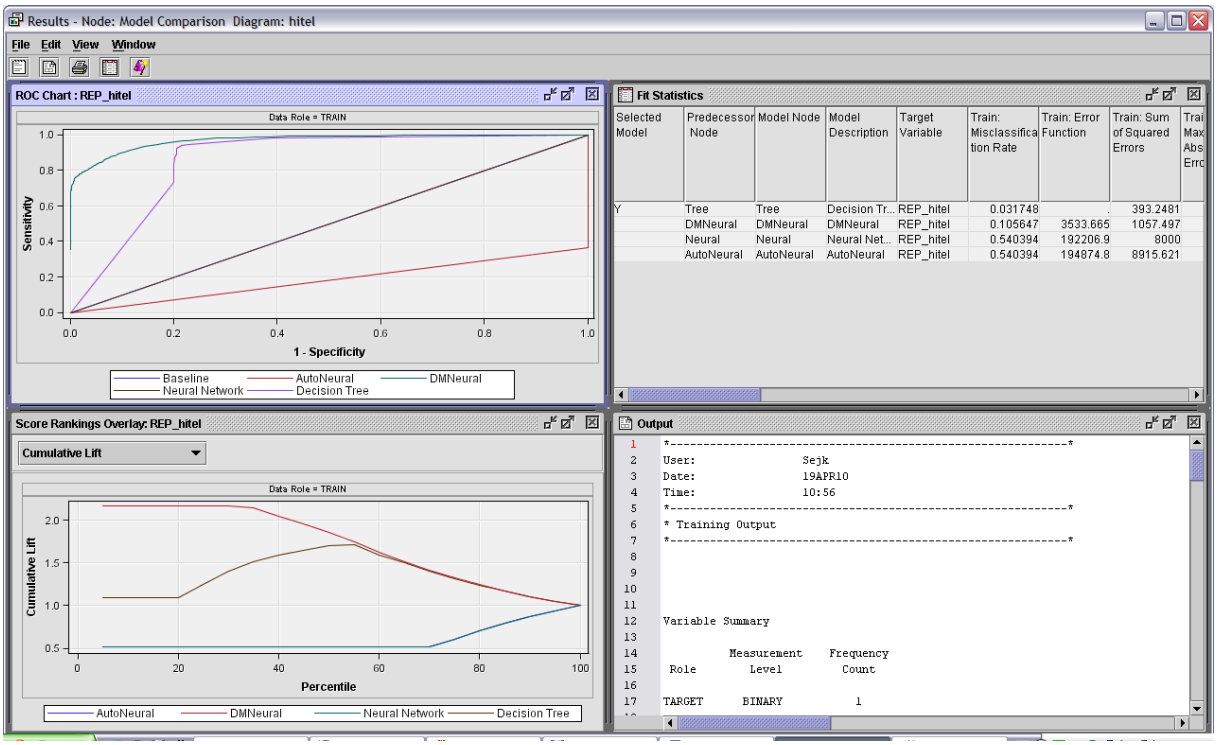
6. Kiértékelés

6.1. Model comparison – modellek összehasonlítása

Az irodalomban sokféle minőségi paraméter található a modellek jóságának értékelésére, a legfontosabb mérőszámok kiszámítására lehetőséget biztosít az MC csomópont. A csomópont kulcs paramétere a Model selection csoportból a Selection Statistic. Ha nincs speciális célunk, vagy tudásunk, akkor hagyjuk ezt az alapértelmezett ROC-on.

1. Az Assess fülről helyezzük az MC csomópontot a diagramra, majd az Interaktív döntési fa kivételével kössük össze az összes modellező csomóponttal.

2. Futtassuk, majd nyissuk meg az eredmény ablakot.



42. ábra Kiértékelés - modellek összehasonlítása

A fenti kép bal felső ablaka az úgynevezett ROC görbe, a kék vonal (egyenes y=x) lenne az eredmény, ha véletlen sorrendbe rakva választanánk a hitelkérők között és figyelnénk a sikerességet. A kék görbétől való eltérés a modellek nyeresége.