Academia de Studii Economice
Facultatea de Cibernetică, Statistică și Informatică Economică
Specializarea: Informatică Economică

Proiect Pachete Software

Profesor coordonator
Conferențiar univ. dr.
Vreja Lucia-Ovidia

Studenți
Rădulescu Theodor
Raicea David-Gabriel

List Of Images

List Of Tables

1. Crearea unui set de date SAS din fișiere externe

Importarea datelor dintr-un fișier .csv extern (în cazul nostru, Samsung.csv) într-un set de date SAS pentru a putea fi utilizat în analizele viitoare.

Informațiile necesare sunt: calea către fișier și cunoașterea structurii fișierului.

Vom folosi **infile** cu opțiunea dsd (delimiter-sensitive data) pentru fișiere CSV și **input** pentru a defini variabilele.

```
data samsung;
        infile '/home/u64223636/ProiectPSW/Samsung.csv' dsd firstobs=2;
        input Date :yymmdd10. Open High Low Close Volume;
        format Date yymmdd10.;
run;
```

Datele vor fi încărcate într-un set denumit “samsung”:

Total rows: 5621 Total columns: 6 — Rows 1-100

	Date	Open	High	Low	Close	Volume
1	2000-01-04	6000	6110	5660	6110	4651.737793
2	2000-01-05	5800	6060	5520	5580	4248.232422
3	2000-01-06	5750	5780	5580	5620	4278.686523
4	2000-01-07	5560	5670	5360	5540	4217.780273
5	2000-01-10	5600	5770	5580	5770	4392.884766
6	2000-01-11	5820	6100	5770	5770	4392.884766
7	2000-01-12	5610	5740	5600	5720	4354.818359
8	2000-01-13	5600	5740	5560	5710	4347.205078
9	2000-01-14	5720	5880	5680	5830	4438.565918
10	2000-01-17	6000	6180	5920	6100	4644.125
11	2000-01-18	6160	6160	5980	6100	4644.125
12	2000-01-19	6000	6040	5960	5960	4537.538574
13	2000-01-20	5860	6040	5820	6040	4598.444824

Figure 1. Setul de date “samsung”.

Pentru a evita pierderea datelor la închiderea sesiunii, am creat un set de date permanent denumit “samsung_perm” folosind:

```
libname proiect '/home/u64223636/ProiectPSW';
data proiect.samsung_perm;
        set samsung;
run;
```

Astfel, vom putea folosi oricând fișierul “samsung_perm.sas7bdat”.

2. Crearea și folosirea de formate definite de utilizator

Vom defini formate cu scopul de a transforma variabile brute în forme ușor interpretabile. Astfel, vom clasifica valorile de închidere (Close) în categorii intuitive, precum Scăzut, Mediu și Ridicat. Variabila numerică Close este necesară.

Acest lucru se realizează prin utilizarea *proc format:*

```
proc format;
            value pret_fmt
            low - <5000 = 'Scăzut'
            5000 - <8000 = 'Mediu'
            8000 - high = 'Ridicat';
run;
data proiect.samsung_perm;
            set proiect.samsung_perm;
            length CategorieClose $10;
   CategorieClose = put(Close, pret_fmt.);
run;
title "Primele obs cu clasificarea pe Close";
proc print data=proiect.samsung_perm(obs=10);
            var Date Close CategorieClose;
run;
```

Primele obs cu clasificarea pe Close

Obs	Date	Close	CategorieClose
1	2000-01-04	6110	Mediu
2	2000-01-05	5580	Mediu
3	2000-01-06	5620	Mediu
4	2000-01-07	5540	Mediu
5	2000-01-10	5770	Mediu
6	2000-01-11	5770	Mediu
7	2000-01-12	5720	Mediu
8	2000-01-13	5710	Mediu
9	2000-01-14	5830	Mediu
10	2000-01-17	6100	Mediu

Figure 2. Observații clasificate

3. Procesarea iterativă și condițională a datelor

În analiza datelor financiare istorice ale companiei Samsung, procesarea condițională este esențială pentru clasificarea și interpretarea evoluțiilor pieței.

Vom grupa observațiile în funcție de anul în care au avut loc prin crearea unei variabile ce va clasifica fiecare observație în intervale de ani.

```
data proiect.samsung_perm;
   set proiect.samsung_perm;
   An = year(Date);
   length GrupAni $ 12;
   if 2000 <= An <= 2005 then GrupAni = '2000–2005';
   else if 2006 <= An <= 2010 then GrupAni = '2006–2010';
   else if 2011 <= An <= 2015 then GrupAni = '2011–2015';
   else if 2016 <= An <= 2020 then GrupAni = '2016–2020';
   else if An >= 2021 then GrupAni = '2021–2025';
run;

proc print data=proiect.samsung_perm(obs=10);
   var Date An GrupAni;
   title "Clasificare pe intervale de ani";
run;
```

Clasificare pe intervale de ani

Obs	Date	An	GrupAni
1	2000-01-04	2000	2000–2005
2	2000-01-05	2000	2000–2005
3	2000-01-06	2000	2000–2005
4	2000-01-07	2000	2000–2005
5	2000-01-10	2000	2000–2005
6	2000-01-11	2000	2000–2005
7	2000-01-12	2000	2000–2005
8	2000-01-13	2000	2000–2005
9	2000-01-14	2000	2000–2005
10	2000-01-17	2000	2000–2005

Figure 3. Clasificarea pe intervale de ani

Dorim să determinăm dacă în fiecare zi prețul a crescut, a scăzut sau a stagnat. S-au folosit valorile din Open și Close, iar metoda a constat în instrucțiuni if și missing() pentru a genera o variabilă Trend.

```
data proiect.samsung_perm;
   set proiect.samsung_perm;
   length Trend $ 10;
   if missing(Open) or missing(Close) then Trend = "Necunoscut";
   else if Close > Open then Trend = "Creștere";
   else if Close < Open then Trend = "Scădere";
   else Trend = "Stagnare";
run;
```

```
proc print data=proiect.samsung_perm(obs=10);
   var Date Open Close Trend;
   title "Etichetare trend zilnic";
run;
```

Etichetare trend zilnic

Obs	Date	Open	Close	Trend
1	2000-01-04	6000	6110	Creștere
2	2000-01-05	5800	5580	Scădere
3	2000-01-06	5750	5620	Scădere
4	2000-01-07	5560	5540	Scădere
5	2000-01-10	5600	5770	Creștere
6	2000-01-11	5820	5770	Scădere
7	2000-01-12	5610	5720	Creștere
8	2000-01-13	5600	5710	Creștere
9	2000-01-14	5720	5830	Creștere
10	2000-01-17	6000	6100	Creștere

Figure 4. Etichetare trend zilnic

Vom evalua cât de mare este volumul zilnic şi îl încadrăm în trei niveluri. Au fost folosite valorile din Volume, iar metoda de calcul a fost select-when cu praguri numerice.

```
data proiect.samsung_perm;
   set proiect.samsung_perm;
   length NivelVolum $ 12;
   select;
      when (Volume < 5000000) NivelVolum = 'Mic';
      when (5000000 <= Volume < 20000000) NivelVolum = 'Mediu';
      when (Volume >= 20000000) NivelVolum = 'Ridicat';
      otherwise NivelVolum = 'Necunoscut';
   end;
run;

proc print data=proiect.samsung_perm(obs=10);
   var Date Volume NivelVolum;
run;
```

Obs	Date	Volume	NivelVolum
1	2000-01-04	4651.74	Mic
2	2000-01-05	4248.23	Mic
3	2000-01-06	4278.69	Mic
4	2000-01-07	4217.78	Mic
5	2000-01-10	4392.88	Mic
6	2000-01-11	4392.88	Mic
7	2000-01-12	4354.82	Mic
8	2000-01-13	4347.21	Mic
9	2000-01-14	4438.57	Mic
10	2000-01-17	4644.13	Mic

Figure 5. Încadrarea volumul de tranzacționare

Simulând o investiție de 10.000 KRW cu 5% dobândă, dorim să aflăm în câți ani se dublează. Am folosit do until și o condiție pe coloana Total.

```
data dublare;
   Total = 10000;
   Dobanda = 0.05;
   An = 0;
   do until (Total >= 20000);
      An + 1;
      Total + Dobanda * Total;
      output;
   end;
   format Total dollar12.2;
run;

proc print data=dublare;
run;
```

Obs	Total	Dobanda	An
1	$10,500.00	0.05	1
2	$11,025.00	0.05	2
3	$11,576.25	0.05	3
4	$12,155.06	0.05	4
5	$12,762.82	0.05	5
6	$13,400.96	0.05	6
7	$14,071.00	0.05	7
8	$14,774.55	0.05	8
9	$15,513.28	0.05	9
10	$16,288.95	0.05	10
11	$17,103.39	0.05	11
12	$17,958.56	0.05	12
13	$18,856.49	0.05	13
14	$19,799.32	0.05	14
15	$20,789.28	0.05	15

Figure 6. Simulare investiție

4. Crearea de subseturi de date

Dorim să extragem doar înregistrările din anul 2020. Au fost folosite datele din Date, iar metoda de filtrare s-a bazat pe funcția year() și instrucțiunea if.

```
data samsung_2020;
   set proiect.samsung_perm;
   if year(Date) = 2020;
run;

proc print data=samsung_2020(obs=10);
run;
```

Obs	Date	Open	High	Low	Close	Volume	CategorieClose	An	GrupAni	Trend	NivelVolum
1	2020-01-02	55500	56000	55000	55200	51557.59	Ridicat	2020	2016–2020	Scădere	Mic
2	2020-01-03	56000	56600	54900	55500	51837.80	Ridicat	2020	2016–2020	Scădere	Mic
3	2020-01-06	54900	55600	54600	55500	51837.80	Ridicat	2020	2016–2020	Creștere	Mic
4	2020-01-07	55700	56400	55600	55800	52118.01	Ridicat	2020	2016–2020	Creștere	Mic
5	2020-01-08	56200	57400	55900	56800	53052.02	Ridicat	2020	2016–2020	Creștere	Mic
6	2020-01-09	58400	58600	57400	58600	54733.24	Ridicat	2020	2016–2020	Creștere	Mic
7	2020-01-10	58800	59700	58300	59500	55573.85	Ridicat	2020	2016–2020	Creștere	Mic
8	2020-01-13	59600	60000	59100	60000	56040.87	Ridicat	2020	2016–2020	Creștere	Mic
9	2020-01-14	60400	61000	59900	60000	56040.87	Ridicat	2020	2016–2020	Scădere	Mic
10	2020-01-15	59500	59600	58900	59000	55106.85	Ridicat	2020	2016–2020	Scădere	Mic

Figure 7. Observațiile din 2020

Extragem observațiile din anul 2021 în care prețul de închidere a depășit 7000. Au fost folosite Date și Close, iar metoda aplicată a fost where în cadrul proc print.

```
proc print data=proiect.samsung_perm;
   where year(Date) = 2021 and Close > 7000;
   var Date Close;
run;
```

Obs	Date	Close
5281	2021-01-04	83000
5282	2021-01-05	83900
5283	2021-01-06	82200
5284	2021-01-07	82900
5285	2021-01-08	88800
5286	2021-01-11	91000
5287	2021-01-12	90600
5288	2021-01-13	89700
5289	2021-01-14	89700
5290	2021-01-15	88000
5291	2021-01-18	85000
5292	2021-01-19	87000
5293	2021-01-20	87200
5294	2021-01-21	88100
5295	2021-01-22	86800

Figure 8. Observațiile din anul 2021 în care prețul de închidere a depășit 7000

Pentru a lucra doar cu variabilele importante pentru vizualizare, au fost selectate Date, Open, Close și Volume, iar metoda a fost keep= în instrucțiunea set.

```
data samsung_trimmed;
   set proiect.samsung_perm(keep=Date Open Close Volume);
run;

proc print data=samsung_trimmed(obs=10);
run;
```

Obs	Date	Open	Close	Volume
1	2000-01-04	6000	6110	4651.74
2	2000-01-05	5800	5580	4248.23
3	2000-01-06	5750	5620	4278.69
4	2000-01-07	5560	5540	4217.78
5	2000-01-10	5600	5770	4392.88
6	2000-01-11	5820	5770	4392.88
7	2000-01-12	5610	5720	4354.82
8	2000-01-13	5600	5710	4347.21
9	2000-01-14	5720	5830	4438.57
10	2000-01-17	6000	6100	4644.13

Figure 9. Alegerea variabilelor importante

5. Utilizarea de funcții SAS

Vom calcula media dintre Open și Close, iar apoi o vom rotunji. Pentru asta, am folosit funcțiile mean și round.

```
data proiect.samsung_perm;
   set proiect.samsung_perm;
   PretMediu = mean(Open, Close);
   PretRotunjit = round(Close, 100);
run;

proc print data=proiect.samsung_perm(obs=10);
   var Date Open Close PretMediu PretRotunjit;
run;
```

Obs	Date	Open	Close	PretMediu	PretRotunjit
1	2000-01-04	6000	6110	6055	6100
2	2000-01-05	5800	5580	5690	5600
3	2000-01-06	5750	5620	5685	5600
4	2000-01-07	5560	5540	5550	5500
5	2000-01-10	5600	5770	5685	5800
6	2000-01-11	5820	5770	5795	5800
7	2000-01-12	5610	5720	5665	5700
8	2000-01-13	5600	5710	5655	5700
9	2000-01-14	5720	5830	5775	5800
10	2000-01-17	6000	6100	6050	6100

Figure 10. Prețul mediu și rotunjit

Pentru a extrage luna și ziua pentru analiza sezonieră, am folosit funcțiile month și day aplicate pe coloana Date.

```
data proiect.samsung_perm;
   set proiect.samsung_perm;
   Luna = month(Date);
   Zi = day(Date);
run;

proc print data=proiect.samsung_perm(obs=10);
   var Date Luna Zi;
run;
```

Obs	Date	Luna	Zi
1	2000-01-04	1	4
2	2000-01-05	1	5
3	2000-01-06	1	6
4	2000-01-07	1	7
5	2000-01-10	1	10
6	2000-01-11	1	11
7	2000-01-12	1	12
8	2000-01-13	1	13
9	2000-01-14	1	14
10	2000-01-17	1	17

Figure 11. Extragerea lunii și a zilei

Dorim să măsurăm cât de mare a fost variația de preț într-o zi raportată la prețul de deschidere, construind astfel un indicator de volatilitate. Apoi, clasificăm această volatilitate în „Scăzută”, „Moderată” și „Ridicată”.

```
data proiect.samsung_perm;
   set proiect.samsung_perm;

   if Open > 0 then VolRel = round((High - Low) / Open, 0.001);
   else VolRel = .;

   length TipVolatilitate $10;
   select;
      when (VolRel < 0.01) TipVolatilitate = "Scăzută";
      when (0.01 <= VolRel < 0.03) TipVolatilitate = "Moderată";
      when (VolRel >= 0.03) TipVolatilitate = "Ridicată";
      otherwise TipVolatilitate = "Necunoscut";
   end;
run;

proc print data=proiect.samsung_perm(obs=10);
   var Date Open High Low VolRel TipVolatilitate;
   title "Calculul volatilității zilnice și clasificarea ei";
run;
```

Calculul volatilității zilnice și clasificarea ei

Obs	Date	Open	High	Low	VolRel	TipVolatilitate
1	2000-01-04	6000	6110	5660	0.075	Ridicată
2	2000-01-05	5800	6060	5520	0.093	Ridicată
3	2000-01-06	5750	5780	5580	0.035	Ridicată
4	2000-01-07	5560	5670	5360	0.056	Ridicată
5	2000-01-10	5600	5770	5580	0.034	Ridicată
6	2000-01-11	5820	6100	5770	0.057	Ridicată
7	2000-01-12	5610	5740	5600	0.025	Moderată
8	2000-01-13	5600	5740	5560	0.032	Ridicată
9	2000-01-14	5720	5880	5680	0.035	Ridicată
10	2000-01-17	6000	6180	5920	0.043	Ridicată

Figure 12. Calculul volatilitatii zilnice și clasificarea ei

6. Combinarea seturilor de date

Presupunem că avem două fișiere cu date Samsung pe două perioade diferite. Vom împărți fișierul tău Samsung.csv în două părți și le vom concatena. La concatenare adăugăm o variabilă nouă, Pret, care este calculată în funcție de prețul de închidere.

```
data samsung1 samsung2;
   set proiect.samsung_perm;
   if _N_ <= 100 then output samsung1; /*_N_ este o variabilă automată în SAS care reține numărul */
   else output samsung2;   /*observației curente. Cu acest IF punem primele 100 de observații să fie */
   run;                    /*puse în samsung1 */

data samsung_total;
   set samsung1 samsung2;
   if Close < 5000 then Pret = 0;
   else if Close < 8000 then Pret = 60;
   else Pret = 25;
run;

proc print data=samsung_total (obs=10);
   title 'Concatenarea a doua subseturi Samsung si calculul pretului';
run;
```

Concatenarea a doua subseturi Samsung si calculul pretului

Obs	Date	Open	High	Low	Close	Volume	CategorieClose	Pret
1	2000-01-04	6000	6110	5660	6110	4651.74	Mediu	60
2	2000-01-05	5800	6060	5520	5580	4248.23	Mediu	60
3	2000-01-06	5750	5780	5580	5620	4278.69	Mediu	60
4	2000-01-07	5560	5670	5360	5540	4217.78	Mediu	60
5	2000-01-10	5600	5770	5580	5770	4392.88	Mediu	60
6	2000-01-11	5820	6100	5770	5770	4392.88	Mediu	60
7	2000-01-12	5610	5740	5600	5720	4354.82	Mediu	60
8	2000-01-13	5600	5740	5560	5710	4347.21	Mediu	60
9	2000-01-14	5720	5880	5680	5830	4438.57	Mediu	60
10	2000-01-17	6000	6180	5920	6100	4644.13	Mediu	60

Figure 13. Concatenarea a două subseturi și calculul prețului

Pornind de la setul de date permanent proiect.samsung_perm, , care conține informații bursiere despre acțiunile companiei Samsung (inclusiv prețul de închidere – Close), dorim să redenumim variabila Close în PretFinal.

Această redenumire reflectă mai bine scopul analizei ulterioare, unde prețul de închidere este interpretat drept prețul final relevant pentru decizii de tranzacționare.

```
data samsung_redenumit;
   set proiect.samsung_perm (rename=(Close=PretFinal));
run;

proc print data=samsung_redenumit(obs=10);
   title 'Redenumirea variabilei Close in PretFinal';
run;
```

Redenumirea variabilei Close in PretFinal

Obs	Date	Open	High	Low	PretFinal	Volume	CategorieClose
1	2000-01-04	6000	6110	5660	6110	4651.74	Mediu
2	2000-01-05	5800	6060	5520	5580	4248.23	Mediu
3	2000-01-06	5750	5780	5580	5620	4278.69	Mediu
4	2000-01-07	5560	5670	5360	5540	4217.78	Mediu
5	2000-01-10	5600	5770	5580	5770	4392.88	Mediu
6	2000-01-11	5820	6100	5770	5770	4392.88	Mediu
7	2000-01-12	5610	5740	5600	5720	4354.82	Mediu
8	2000-01-13	5600	5740	5560	5710	4347.21	Mediu
9	2000-01-14	5720	5880	5680	5830	4438.57	Mediu
10	2000-01-17	6000	6180	5920	6100	4644.13	Mediu

Figure 14. Redenumirea variabilei Close în PretFinal

Setul de date original proiect.samsung_perm, care conține valorile de tranzacționare zilnică pentru acțiunile Samsung, a fost împărțit anterior în două subseturi: samsung1 și samsung2. Aceste subseturi conțin observații diferite, dar structura lor este identică.

Pentru a interclasa aceste seturi, trebuie să le sortăm mai întâi după variabila Date. Apoi, folosind clauza BY, le combinăm astfel încât înregistrările să fie ordonate cronologic în noul set de date rezultat.

```
proc sort data=samsung1; by Date; run;
proc sort data=samsung2; by Date; run;
data samsung_interclasat;
   set samsung1 samsung2;
   by Date;
run;
```

```
proc print data=samsung_interclasat (obs=10);
   title 'Interclasarea datelor Samsung dupa Data';
run;
```

Interclasarea datelor Samsung dupa Data

Obs	Date	Open	High	Low	Close	Volume	CategorieClose
1	2000-01-04	6000	6110	5660	6110	4651.74	Mediu
2	2000-01-05	5800	6060	5520	5580	4248.23	Mediu
3	2000-01-06	5750	5780	5580	5620	4278.69	Mediu
4	2000-01-07	5560	5670	5360	5540	4217.78	Mediu
5	2000-01-10	5600	5770	5580	5770	4392.88	Mediu
6	2000-01-11	5820	6100	5770	5770	4392.88	Mediu
7	2000-01-12	5610	5740	5600	5720	4354.82	Mediu
8	2000-01-13	5600	5740	5560	5710	4347.21	Mediu
9	2000-01-14	5720	5880	5680	5830	4438.57	Mediu
10	2000-01-17	6000	6180	5920	6100	4644.13	Mediu

Figure 15. Interclasarea datelor Samsung dupa Data

Fuziunea unu-la-unu este aplicabilă atunci când avem **câte o singură observație corespunzătoare pentru fiecare valoare a unei variabile cheie în ambele seturi**. Este esențial ca aceste seturi de date să fie **sortate în prealabil** după variabila comună, altfel SAS va returna o eroare sau va combina incorect datele.

În acest caz, vom crea un nou set de date care calculează volatilitatea zilnică (diferența dintre cel mai mare și cel mai mic preț al zilei: High - Low), și îl vom fuziona cu setul original pe baza variabilei Date. Astfel, fiecare înregistrare din setul final va conține, pe lângă datele bursiere standard, și o nouă variabilă calculată – Volatilitate.

```
data info_volatilitate;
   set proiect.samsung_perm (keep=Date Close);
   Volatilitate = High - Low;
run;

proc sort data=info_volatilitate; by Date; run;
proc sort data=proiect.samsung_perm; by Date; run;

data samsung_fuziune;
   merge proiect.samsung_perm info_volatilitate;
   by Date;
run;

proc print data=samsung_fuziune (obs=10);
   title 'Fuziunea unu-la-unu pe baza Date';
run;
```

Fuziunea unu-la-unu pe baza Date

Obs	Date	Open	High	Low	Close	Volume	CategorieClose	Volatilitate
1	2000-01-04	6000	.	.	6110	4651.74	Mediu	.
2	2000-01-05	5800	.	.	5580	4248.23	Mediu	.
3	2000-01-06	5750	.	.	5620	4278.69	Mediu	.
4	2000-01-07	5560	.	.	5540	4217.78	Mediu	.
5	2000-01-10	5600	.	.	5770	4392.88	Mediu	.
6	2000-01-11	5820	.	.	5770	4392.88	Mediu	.
7	2000-01-12	5610	.	.	5720	4354.82	Mediu	.
8	2000-01-13	5600	.	.	5710	4347.21	Mediu	.
9	2000-01-14	5720	.	.	5830	4438.57	Mediu	.
10	2000-01-17	6000	.	.	6100	4644.13	Mediu	.

Figure 16. Fuziunea unu la unu pe baza Date

Ne dorim să îmbogățim setul de date Samsung cu o coloană care conține reduceri de preț în funcție de volumul tranzacționat.

Pentru ca fuziunea să fie posibilă, trebuie să existe o **variabilă comună** cu valori identice în ambele seturi de date. Deoarece în setul original samsung_perm, volumul (Volume) are valori continue (ex. 13.245.000), iar în tabelul de reduceri (volume_red) avem doar câteva praguri fixe (ex. 1.000.000, 5.000.000 etc.), a fost necesar să construim o versiune modificată a setului Samsung, în care valorile de Volume au fost rotunjite sau grupate pentru a corespunde celor din tabelul de reduceri.

```
/* Setul auxiliar cu reduceri definite pentru volume fixe */
data volume_red;
input Volume VolumeReducere;
   datalines;
1000000  0.02
5000000  0.05
10000000 0.10
15000000 0.15
;
run;
/* Rotunjirea volumului în setul Samsung pentru a permite fuziunea */
data samsung_std;
   set proiect.samsung_perm;


   if Volume < 3000000 then Volume = 1000000;
   else if Volume < 7500000 then Volume = 5000000;
   else if Volume < 12500000 then Volume = 10000000;
   else Volume = 15000000;
run;
/* Sortarea ambelor seturi de date după variabila comună */
proc sort data=samsung_std; by Volume; run;
proc sort data=volume_red; by Volume; run;
```

```
/* Fuziune reală pe baza variabilei Volume */
data samsung_merge;
   merge samsung_std(in=a) volume_red(in=b);
   by Volume;
   if a;
   PretRedus = round(Close * (1 - VolumeReducere), 0.01);
run;

proc print data=samsung_merge(obs=10);
   title 'MERGE Samsung cu Volume Reduceri pe Volume rotunjit';
run;
```

MERGE Samsung cu Volume Reduceri pe Volume rotunjit

Obs	Date	Open	High	Low	Close	Volume	CategorieClose	VolumeReducere	PretRedus
1	2000-01-04	6000	6110	5660	6110	1000000	Mediu	0.02	5987.8
2	2000-01-05	5800	6060	5520	5580	1000000	Mediu	0.02	5468.4
3	2000-01-06	5750	5780	5580	5620	1000000	Mediu	0.02	5507.6
4	2000-01-07	5560	5670	5360	5540	1000000	Mediu	0.02	5429.2
5	2000-01-10	5600	5770	5580	5770	1000000	Mediu	0.02	5654.6
6	2000-01-11	5820	6100	5770	5770	1000000	Mediu	0.02	5654.6
7	2000-01-12	5610	5740	5600	5720	1000000	Mediu	0.02	5605.6
8	2000-01-13	5600	5740	5560	5710	1000000	Mediu	0.02	5595.8
9	2000-01-14	5720	5880	5680	5830	1000000	Mediu	0.02	5713.4
10	2000-01-17	6000	6180	5920	6100	1000000	Mediu	0.02	5978.0

Figure 17. Setul de date Samsung cu Volume Reduceri cu ajutorul MERGE

Presupunem că dorim să comparăm datele despre acțiunile Samsung din două perioade diferite: una anterioară (samsung_vechi) și una recentă (samsung_nou). Scopul este să observăm care înregistrări apar doar într-unul din seturi și care sunt comune.

Pentru aceasta, folosim opțiunea IN= pentru a marca dacă o observație provine din fiecare dintre cele două surse. Rezultatul este afișat cu ajutorul instrucțiunii PUT, deoarece variabilele definite prin IN= sunt variabile temporare și nu pot fi afișate direct prin PROC PRINT.

```
Date=2000-01-04 inVechi=1 inNou=0 Close=6110
Date=2000-01-05 inVechi=1 inNou=0 Close=5580
Date=2000-01-06 inVechi=1 inNou=0 Close=5620
Date=2000-01-07 inVechi=1 inNou=0 Close=5540
Date=2000-01-10 inVechi=1 inNou=0 Close=5770
Date=2000-01-11 inVechi=1 inNou=0 Close=5770
Date=2000-01-12 inVechi=1 inNou=0 Close=5720
Date=2000-01-13 inVechi=1 inNou=0 Close=5710
Date=2000-01-14 inVechi=1 inNou=0 Close=5830
Date=2000-01-17 inVechi=1 inNou=0 Close=6100
Date=2000-01-18 inVechi=1 inNou=0 Close=6100
Date=2000-01-19 inVechi=1 inNou=0 Close=5960
Date=2000-01-20 inVechi=1 inNou=0 Close=6040
Date=2000-01-21 inVechi=1 inNou=0 Close=5880
Date=2000-01-24 inVechi=1 inNou=0 Close=5700
Date=2000-01-25 inVechi=1 inNou=0 Close=5440
Date=2000-01-26 inVechi=1 inNou=0 Close=5480
Date=2000-01-27 inVechi=1 inNou=0 Close=5520
Date=2000-01-28 inVechi=1 inNou=0 Close=5820
Date=2000-01-31 inVechi=1 inNou=0 Close=5580
Date=2000-02-01 inVechi=1 inNou=0 Close=5320
```

Figure 18. Secțiune din rezultatul compararea datelor cu ajutorul IN=

7. Proceduri specifice SQL

Pentru a exemplifica utilizarea procedurilor specifice SQL am introdus un nou set de date EWY, ce este un ETF din Coreea de Sud care contine mai multe actiuni la firme Coreene precum Samsung (cu o proportie de aproximativ 25%), aceste ETF este reprezentat cu în moneda USD.

Se realizează o joncțiune internă (INNER JOIN) între tabelele Samsung și EWY, păstrând doar înregistrările care au aceeași dată (date) în ambele tabele. Astfel, sunt comparate valorile acțiunilor Samsung și ale ETF-ului EWY pentru zilele în care există date disponibile simultan.

```
PROC SQL;
CREATE TABLE work.inner_join AS
SELECT * FROM sams_renamed AS s
INNER JOIN ewy_renamed AS e ON s.date = e.date;
QUIT;
```

	Date	open_s	high_s	low_s	Close	volume_s	price	open_e	high_e	low_e	volume_e	change_pct
1	2000-05-15	6290	6530	6220	6440	4902.977539	19.44	19.44	19.44	19.44	5.00K	-1.27%
2	2000-05-16	6510	6800	6510	6800	5177.057129	19.81	19.81	19.81	19.81	0.30K	1.90%
3	2000-05-17	7000	7340	6920	6920	5268.416504	19.44	19.44	19.44	19.44	0.40K	-1.87%
4	2000-05-18	6740	7000	6680	6800	5177.057129	19.25	19.31	19.31	19.25	0.80K	-0.98%
5	2000-05-19	6740	6900	6600	6900	5253.189453	19.31	19.31	19.31	19.31	18.00K	0.31%
6	2000-05-22	6720	6830	6620	6670	5078.083008	17.44	17.94	17.94	17.44	70.50K	-9.68%
7	2000-05-23	6500	6700	6350	6380	4857.297852	17.88	17.94	18.12	17.88	51.30K	2.52%
8	2000-05-24	6160	6370	5920	6200	4720.257324	17.69	17.69	17.69	17.56	10.60K	-1.06%
9	2000-05-25	6330	6380	5930	6000	4567.991699	17.88	18.12	18.31	17.88	44.30K	1.07%
10	2000-05-26	5900	5990	5600	5600	4263.460938	16.94	16.94	17.06	16.94	21.00K	-5.26%
11	2000-05-30	5460	5460	5460	5460	4156.872559	17.94	18.12	18.12	17.94	32.10K	5.90%
12	2000-05-31	6000	6290	5960	6160	4689.805664	19.25	19.12	19.25	19.12	55.00K	7.30%
13	2000-06-01	6200	6410	6120	6300	4796.390625	19.19	19.31	19.38	19.19	11.50K	-0.31%
14	2000-06-02	6690	6900	6510	6620	5040.017578	20.5	20.19	20.5	20.19	10.40K	6.83%
15	2000-06-05	6920	6980	6680	6740	5131.377441	21	21.19	21.19	21	3.10K	2.44%
16	2000-06-06	6740	6740	6740	6740	5131.377441	20.75	21.12	21.12	20.75	10.20K	-1.19%
17	2000-06-07	6600	6980	6510	6800	5177.057129	21.25	21.5	21.5	20.5	44.10K	2.41%

Figure 19. Tabel INNER JOIN Samsung EWY

Se realizeaza o joncțiune LEFT JOIN pastrandu-se toate datele din tabelul Samsung,diferent dacă există corespondență în EWY. Dacă pentru o anumită zi nu există date în EWY, câmpurile asociate vor fi completate cu valori lipsă (missing).

```
PROC SQL;
CREATE TABLE work.left_join AS
SELECT * FROM sams_renamed AS s
LEFT JOIN ewy_renamed  AS e
ON s.date = e.date;
QUIT;
```

	Date	open_s	high_s	low_s	Close	volume_s	price	open_e	high_e	low_e	volume_e	change_pct
101	2000-05-23	6500	6700	6350	6380	4857.297852	17.88	17.94	18.12	17.88	51.30K	2.52%
102	2000-05-24	6160	6370	5920	6200	4720.257324	17.69	17.69	17.69	17.56	10.60K	-1.06%
103	2000-05-25	6330	6380	5930	6000	4567.991699	17.88	18.12	18.31	17.88	44.30K	1.07%
104	2000-05-26	5900	5990	5600	5600	4263.460938	16.94	16.94	17.06	16.94	21.00K	-5.26%
105	2000-05-29	5260	5680	5240	5460	4156.872559	.	.	.	.	.	.
106	2000-05-30	5460	5460	5460	5460	4156.872559	17.94	18.12	18.12	17.94	32.10K	5.90%
107	2000-05-31	6000	6290	5960	6160	4689.805664	19.25	19.12	19.25	19.12	55.00K	7.30%
108	2000-06-01	6200	6410	6120	6300	4796.390625	19.19	19.31	19.38	19.19	11.50K	-0.31%
109	2000-06-02	6690	6900	6510	6620	5040.017578	20.5	20.19	20.5	20.19	10.40K	6.83%
110	2000-06-05	6920	6980	6680	6740	5131.377441	21	21.19	21.19	21	3.10K	2.44%
111	2000-06-06	6740	6740	6740	6740	5131.377441	20.75	21.12	21.12	20.75	10.20K	-1.19%
112	2000-06-07	6600	6980	6510	6800	5177.057129	21.25	21.5	21.5	20.5	44.10K	2.41%
113	2000-06-08	6660	6960	6530	6530	4971.497559	20.25	20.44	20.44	20.25	12.00K	-4.71%
114	2000-06-09	6600	7040	6530	7020	5344.551758	22.12	22.12	22.12	22.12	1.30K	9.23%
115	2000-06-12	7200	7280	7030	7190	5473.976563	22	22.56	22.56	22	12.10K	-0.54%
116	2000-06-13	7060	7100	6870	7060	5375.004883	21.12	21	21.31	21	14.50K	-4.00%

Figure 20. Tabel LEFTJOIN Samsung EWY

În momentul realizării unui FULL JOIN se combină toate datele din ambele tabele. Sunt incluse atât zilele care apar doar în Samsung, cât şi cele care apar doar în EWY, rezultând un set complet de date pe întreaga perioadă.

```
PROC SQL;
CREATE TABLE work.full_join AS
SELECT * FROM sams_renamed AS s
FULL JOIN ewy_renamed AS e
ON s.date = e.date;
QUIT;
```

Total rows: 5839 Total columns: 12 Rows 201-300

	Date	open_s	high_s	low_s	Close	volume_s	price	open_e	high_e	low_e	volume_e	chang
201	2000-10-10	3700	3810	3590	3660	2786.475586	15.31	15.44	15.44	15.31	0.50K	2.07%
202	2000-10-11	3450	3460	3210	3220	2451.48877	14.56	14.5	14.56	14.5	5.00K	-4.90%
203	2000-10-12	3140	3280	3140	3140	2390.582031	13.75	14	14	13.5	2.20K	-5.56%
204	2000-10-13	3000	3090	2830	3030	2306.835449	14	13.69	14	13.69	71.90K	1.82%
205	2000-10-16	3030	3030	3030	3030	2306.835449	14	14.06	14.06	14	0.70K	0.00%
206	2000-10-17	2980	3080	2730	2740	2086.049561	13.31	13.5	13.5	13.31	28.80K	-4.93%
207	2000-10-18	2540	2760	2420	2730	2078.436279	13.25	13.31	13.44	13.25	15.80K	-0.45%
208	2000-10-19	2730	3050	2690	2900	2207.863037	13.69	13.44	13.69	13.44	13.30K	3.32%
209	2000-10-20	3240	3330	3190	3330	2535.234863	14.56	14.5	14.62	14.5	5.00K	6.36%
210	2000-10-23	3490	3520	3160	3200	2436.263184	14.44	14.44	14.5	14	6.90K	-0.82%
211	2000-10-24	3180	3420	3100	3340	2542.848633	14.44	14.25	14.44	14.06	20.70K	0.00%
212	2000-10-25	3200	3260	3130	3210	2443.875488	13.88	14.38	14.38	13.88	35.70K	-3.88%
213	2000-10-26	2930	3030	2740	2890	2200.249512	13.69	13.75	14	13.69	2.80K	-1.37%
214	2000-10-27	2890	2890	2890	2890	2200.249512	13.5	13.56	13.62	13.25	5.00K	-1.39%
215	2000-10-30	2800	2880	2720	2750	2093.663086	13.19	13.69	13.69	13.19	20.80K	-2.30%
216	2000-10-31	2750	2850	2630	2850	2169.795654	13.69	13.62	13.75	13.56	4.00K	3.79%

Figure 21. Tabel FULL JOIN Samsung EWY

Determinarea preţurilor maxime şi minime lunare se realizează pentru fiecare lună din fiecare an, valorile maxime şi minime ale preţurilor de închidere (close pentru Samsung şi price pentru EWY). Acest tip de agregare ajută la analiza volatilităţii lunare.

```
PROC SQL;
SELECT YEAR(date) AS an, MONTH(date) AS luna,
    MAX(close) AS max_samsung, MIN(close) AS min_samsung
FROM proiect.samsung_perm
GROUP BY calculated an, calculated luna;
QUIT;

/* Preţ maxim şi minim lunar pentru EWY */
PROC SQL;
SELECT YEAR(date) AS an, MONTH(date) AS luna,
    MAX(price) AS max_ewy, MIN(price) AS min_ewy
FROM proiect.ewy
GROUP BY calculated an, calculated luna;
QUIT;
```

an	luna	max_samsung	min_samsung
2000	1	6110	5440
2000	2	5760	4800
2000	3	7660	5120
2000	4	7300	5400
2000	5	6920	5460
2000	6	7640	6300
2000	7	7760	5730
2000	8	6460	5470
2000	9	5540	3800
2000	10	3940	2730
2000	11	3780	3100
2000	12	3780	3100
2001	1	4600	3160
2001	2	4290	3740
2001	3	4340	3610
2001	4	4700	3640
2001	5	4700	4210
2001	6	4470	3720
2001	7	3940	3340
2001	8	3990	3660
2001	9	3930	2810
2001	10	3710	2810
2001	11	4620	3580
2001	12	5670	4490

an	luna	max_ewy	min_ewy
2000	5	19.81	16.94
2000	6	22.12	19.19
2000	7	22	17.94
2000	8	20.31	17.38
2000	9	18.81	14.75
2000	10	15.88	13.19
2000	11	15.19	12.62
2000	12	14.12	12
2001	1	15.94	12.44
2001	2	15.27	13.77
2001	3	14	12
2001	4	14.01	11.01
2001	5	15.36	13.87
2001	6	15.07	13.81
2001	7	14.26	12.3
2001	8	14.23	13.2
2001	9	13.53	10.81
2001	10	13.23	11.07
2001	11	17.27	13.49
2001	12	18.36	16.33

Figure 22. Preț maxim și minim lunar pentru Samsung și EWY

Se calculează media lunară a prețului de închidere pentru Samsung și EWY. Anul și luna sunt extrase direct din variabila DATE folosind funcțiile YEAR() și MONTH(). Gruparea se face pe baza acestor valori, permițând o analiză lunară sincronizată între cele două surse financiare.

```
PROC SQL;
CREATE TABLE work.medie_lunara AS
SELECT YEAR(s.date) AS an, MONTH(s.date) AS luna,
    MEAN(s.close) AS medie_samsung,
    MEAN(e.price) AS medie_ewy
FROM proiect.samsung_perm s
INNER JOIN proiect.ewy e
ON s.date = e.date
GROUP BY calculated an, calculated luna;
QUIT;
```

	an	luna	medie_samsung	medie_ewy
1	2000	5	6360.8333333	18.5225
2	2000	6	7011.3636364	20.684090909
3	2000	7	7047.5	20.589
4	2000	8	6099.0909091	19.034545455
5	2000	9	4518	16.579
6	2000	10	3270.4545455	14.378181818
7	2000	11	3335.7142857	14.285714286
8	2000	12	3339	13.041
9	2001	1	4065.7142857	14.531904762
10	2001	2	4029.4736842	14.578421053
11	2001	3	3916.6666667	12.945238095
12	2001	4	4115.5	12.552
13	2001	5	4493.6363636	14.62
14	2001	6	4073.8095238	14.504761905

Figure 23. Preț mediu lunar Samsung și EWY

Se caută zilele în care acțiunile Samsung au înregistrat o creștere (close > open), iar ETF-ul EWY a scăzut (valoare negativă în change_pct). Pentru a face posibilă filtrarea numerică, câmpul change_pct (stocat ca text cu %) este convertit într-o valoare numerică (change_num) prin funcția INPUT după eliminarea simbolului % cu SUBSTR.

```
DATA proiect.ewy_clean;
   SET proiect.ewy;
   change_num = INPUT(SUBSTR(change_pct, 1, LENGTH(change_pct)-1), BEST.);
RUN;

PROC SQL;
CREATE TABLE work.zile_opuse AS
SELECT s.date, s.close, s.open, e.price, e.change_pct
FROM proiect.samsung_perm s
JOIN proiect.ewy_clean e ON s.date = e.date
WHERE s.close > s.open AND change_num < 0;
QUIT;
```

	Date	Close	Open	price	change_pct
1	2020-02-19	60200	59800	61.07	-0.02%
2	2020-02-21	59200	58800	58.37	-1.42%
3	2020-02-25	57900	56200	55.29	-0.05%
4	2020-03-05	57800	57600	56.32	-2.19%
5	2020-06-18	52300	52200	57.24	-0.31%
6	2020-06-19	52900	52600	56.75	-0.86%
7	2020-06-24	52900	51900	57.5	-0.36%
8	2020-06-26	53300	52800	57.07	-0.57%
9	2020-07-24	54200	54000	58.91	-0.14%
10	2020-09-03	56400	55600	63.3	-1.06%
11	2020-09-21	59200	59100	65.47	-0.86%
12	2020-09-23	58600	58400	63.43	-1.58%
13	2020-09-24	57800	57700	62.8	-0.99%
14	2020-10-12	60400	60000	67.99	-0.51%

Figure 24. Zile cu creștere Samsung & scădere EWY

Se selectează doar lunile ianuarie, februarie și martie, apoi se calculează media valorii de închidere pentru fiecare an. Se obține o imagine de ansamblu asupra performanței din primul trimestru.

```
DATA work.q1;
   SET sams.samsung;
   IF MONTH(date) IN (1, 2, 3);
RUN;
```

```
PROC SQL;
SELECT YEAR(date) AS an,
      MEAN(close) AS medie_Q1
FROM work.q1
GROUP BY calculated an;
QUIT;
```

an	medie_Q1
2000	5677.188
2001	3990
2002	6579.375
2003	6022.656
2004	10587.54
2005	9899.375
2006	13393.02
2007	11732.9
2008	11340.66
2009	10082.46
2010	15840
2011	18655.33
2012	22920.97
2013	29840
2014	25983.28
2015	27821.67
2016	23910.33
2017	39141.94
2018	49073.11
2019	43960.17
2020	55891.94
2021	83886.67
2022	73403.45

Figure 25. – Medie trimestru Q1 Samsung

Se identifică înregistrările în care volumul de tranzacționare este zero și se înlocuiesc cu valoarea lipsă (.), conform convenției SAS. Acest pas este important pentru evitarea erorilor în analizele ulterioare.

```
DATA work.samsung_clean;
   SET proiect.samsung_perm;
   IF volume = 0 THEN volume = .;
RUN;
```

	Date	Open	High	Low	Close	Volume
1401	2005-05-17	9800	9860	9670	9790	7453.441406
1402	2005-05-18	9890	9900	9780	9790	7453.441406
1403	2005-05-19	9860	10020	9850	9970	.
1404	2005-05-20	9990	10040	9900	9970	7590.47998
1405	2005-05-23	9920	9970	9890	9960	7582.866211
1406	2005-05-24	9930	9970	9860	9900	7537.187988
1407	2005-05-25	9870	9930	9670	9690	7377.304688
1408	2005-05-26	9630	9700	9590	9640	7339.237305
1409	2005-05-27	9800	9860	9730	9800	7461.052246
1410	2005-05-30	9800	9880	9800	9870	7514.348633
1411	2005-05-31	9850	9880	9730	9780	7445.827637

Figure 26. Remedierea înregistrărilor cu volum 0

Se adaugă o coloană nouă care calculează diferența procentuală dintre prețul de închidere Samsung și prețul ETF-ului EWY pentru aceeași zi. Formula aplicată este: ((close - price) / price) * 100
Această valoare permite comparația directă între cele două instrumente.

```
PROC SQL;
CREATE TABLE work.diferente_pct AS
SELECT s.date, s.close, e.price,
    ((s.close - e.price) / e.price) * 100 AS pct_diff
FROM proiect.samsung_perm s
JOIN proiect.ewy e ON s.date = e.date;
QUIT;
```

	Date	Close	price	pct_diff
1	2000-05-15	6440	19.44	33027.572016
2	2000-05-16	6800	19.81	34226.09793
3	2000-05-17	6920	19.44	35496.707819
4	2000-05-18	6800	19.25	35224.675325
5	2000-05-19	6900	19.31	35632.780943
6	2000-05-22	6670	17.44	38145.412844
7	2000-05-23	6380	17.88	35582.326622
8	2000-05-24	6200	17.69	34948.049746
9	2000-05-25	6000	17.88	33457.04698
10	2000-05-26	5600	16.94	32957.85124
11	2000-05-30	5460	17.94	30334.782609
12	2000-05-31	6160	19.25	31900

Figure 27. Diferența procentuală Samsung vs EWY

Se extrag, pentru fiecare lună din fiecare an, cele mai mari variații procentuale înregistrate de ETF-ul EWY (col. change_pct). Această analiză permite evidențierea celor mai volatile perioade din punct de vedere al performanței pieței.

```
PROC SQL;
SELECT YEAR(date) AS an, MONTH(date) AS luna,
    MAX(change_num) AS max_var_pct
FROM proiect.ewy_clean
GROUP BY calculated an, calculated luna;
QUIT;
```

an	luna	max_var_pct
2000	5	7.3
2000	6	9.23
2000	7	3.29
2000	8	6.44
2000	9	5.03
2000	10	6.36
2000	11	4.59
2000	12	6.51
2001	1	11.01
2001	2	3.23
2001	3	5.64

Figure 28. Identificarea lunilor cu variații maxime EWY

8. Prelucrarea datelor prin crearea de rapoarte şi aplicarea de analize statistice

Afişăm un raport detaliat privind evoluția valorii Close în fiecare an.

Au fost folosite datele Date şi Close, iar ca metode de calcul s-au aplicat proc sort pentru sortare şi proc print cu by, sum şi etichete personalizate.

```
data samsung_raport;
   set proiect.samsung_perm;
   An = year(Date);
run;

proc sort data=samsung_raport;
   by An;
run;

proc print data=samsung_raport label sumlabel='Total #byval(An)' grandtotal_label='Total' noobs;
   by An;
   var Date Open Close;
   sum Close;
   format Date date9.;
   label Date = 'Data'
         Open = 'Deschidere'
         Close = 'Inchidere';
   title "Raport anual: evoluția valorii de închidere";
run;
```

Raport anual: evoluția valorii de închidere

An=2000

Data	Deschidere	Inchidere
04JAN2000	6000	6110
05JAN2000	5800	5580
06JAN2000	5750	5620
07JAN2000	5560	5540
10JAN2000	5600	5770
11JAN2000	5820	5770
12JAN2000	5610	5720
13JAN2000	5600	5710
14JAN2000	5720	5830
17JAN2000	6000	6100
18JAN2000	6160	6100
19JAN2000	6000	5960
20JAN2000	5860	6040

Figure 29. Raport anual: evoluția valorii de închidere

Dorim să analizăm distribuția valorii de închidere a acțiunilor pentru a detecta medii, abateri şi valori extreme.

Au fost folosite datele Close, iar ca metode procedura proc univariate cu plot, histogram, şi nextrobs.

```
proc univariate data=proiect.samsung_perm plot;
    var Close;
    histogram Close;
    id Date;
    title "Statistici descriptive și distribuție pentru prețul de închidere";
run;

proc univariate data=proiect.samsung_perm nextrval=5 nextrobs=0;
    var Close;
    id Date;
    title "Valori extreme distincte pentru prețul de închidere";
run;
```

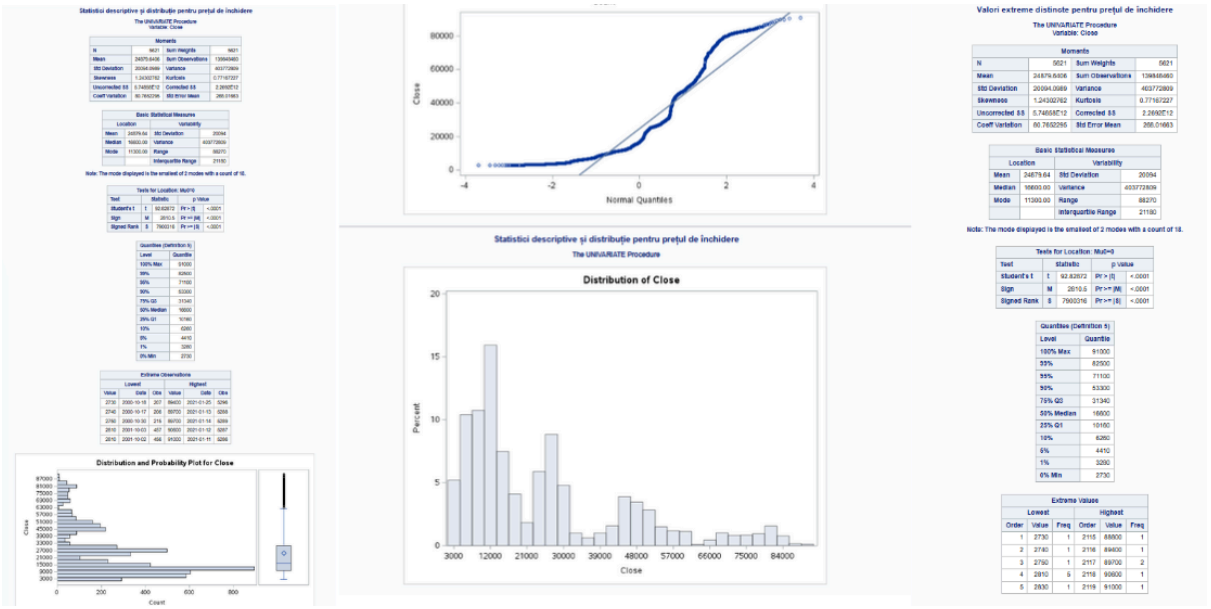


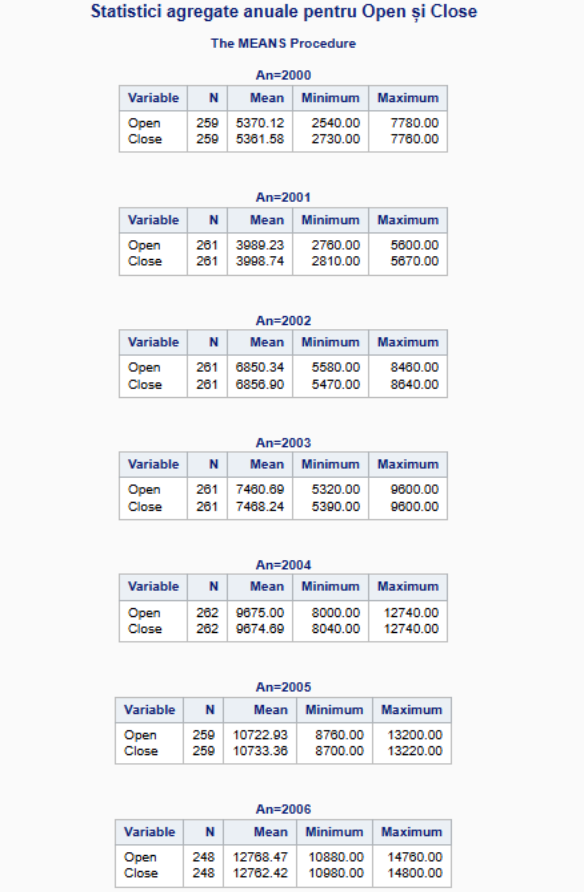
Figure 30. Analiza distribuției valorii de închidere

În vederea generării statisticilor agregate pentru fiecare an (medie, minim, maxim), au fost folosite coloanele Date, Open, Close, iar metodele au fost proc means cu by și var.

```
data samsung_an;
    set proiect.samsung_perm;
    An = year(Date);
run;

proc sort data=samsung_an;
    by An;
run;

proc means data=samsung_an n mean min max;
    by An;
    var Open Close;
    title "Statistici agregate anuale pentru Open și Close";
run;
```



Statistici agregate anuale pentru Open și Close

The MEANS Procedure

An=2000

Variable	N	Mean	Minimum	Maximum
Open	259	5370.12	2540.00	7780.00
Close	259	5361.58	2730.00	7760.00

An=2001

Variable	N	Mean	Minimum	Maximum
Open	261	3989.23	2760.00	5600.00
Close	261	3998.74	2810.00	5670.00

An=2002

Variable	N	Mean	Minimum	Maximum
Open	261	6850.34	5580.00	8460.00
Close	261	6856.90	5470.00	8640.00

An=2003

Variable	N	Mean	Minimum	Maximum
Open	261	7460.69	5320.00	9600.00
Close	261	7468.24	5390.00	9600.00

An=2004

Variable	N	Mean	Minimum	Maximum
Open	262	9675.00	8000.00	12740.00
Close	262	9674.69	8040.00	12740.00

An=2005

Variable	N	Mean	Minimum	Maximum
Open	259	10722.93	8760.00	13200.00
Close	259	10733.36	8700.00	13220.00

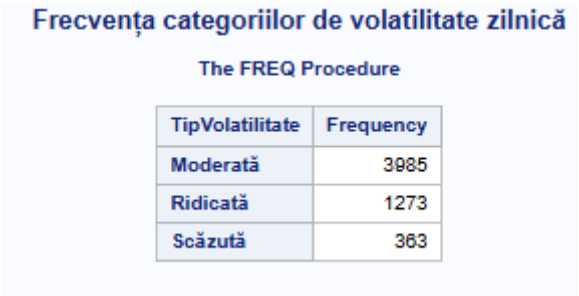
An=2006

Variable	N	Mean	Minimum	Maximum
Open	248	12768.47	10880.00	14760.00
Close	248	12762.42	10980.00	14800.00

Figure 31. Statistici agregate anuale pentru Open și Close

Pentru a analiza câte zile au fost marcate de o volatilitate scăzută, moderată sau ridicată, vom folosi coloanele create în exemplele precedente (VolRel,TipVolatilitate). Acest lucru va fi realizat prin intemediul metodei proc freq.

```
proc freq data=proiect.samsung_perm;
   tables TipVolatilitate / nocum nopercent;
   title "Frecvența categoriilor de volatilitate zilnică";
run;
```



Frecvența categoriilor de volatilitate zilnică

The FREQ Procedure

TipVolatilitate	Frequency
Moderată	3985
Ridicată	1273
Scăzută	363

Figure 32. Frecvența categoriilor de volatilitate zilnică

Dorim să evaluăm dacă există corelație între prețul de închidere (Close) și volumul tranzacționat (Volume).

Au fost folosite variabilele Close și Volume, iar ca metodă de calcul a fost aplicată procedura corr cu specificarea expresă a variabilelor în var și with.

```
proc corr data=proiect.samsung_perm;
   var Volume;
   with Close;
   title "Corelația dintre prețul de închidere și volumul tranzacționat";
run;
```

Corelația dintre prețul de închidere și volumul tranzacționat

The CORR Procedure

1 With Variables:	Close
1 Variables:	Volume

Simple Statistics

Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Close	5621	24880	20094	139848460	2730	91000
Volume	5621	21729	19740	122141076	2078	88908

Pearson Correlation Coefficients, N = 5621
Prob > |r| under H0: Rho=0

	Volume
Close	0.99738 <.0001

Figure 33. Corelația dintre prețul de închidere și volumul tranzacționat

Valoarea coeficientului Pearson este foarte apropiată de 1, ceea ce indică o corelație pozitivă foarte puternică între volumul tranzacționat (Volume) și prețul de închidere (Close).
Cu alte cuvinte, în zilele cu volum mai mare, prețul de închidere tinde să fie mai mare.

În scopul verificării dacă volumul tranzacționat poate fi un predictor pentru prețul de închidere, vol folosi analiza de regresie liniară prin utilizarea Volume ca variabilă independentă și Close dependentă prin procedura reg.

```
proc reg data=proiect.samsung_perm;
   model Close = Volume;
   title "Regresie liniară: estimarea prețului Close în funcție de Volume";
run;
quit;
```

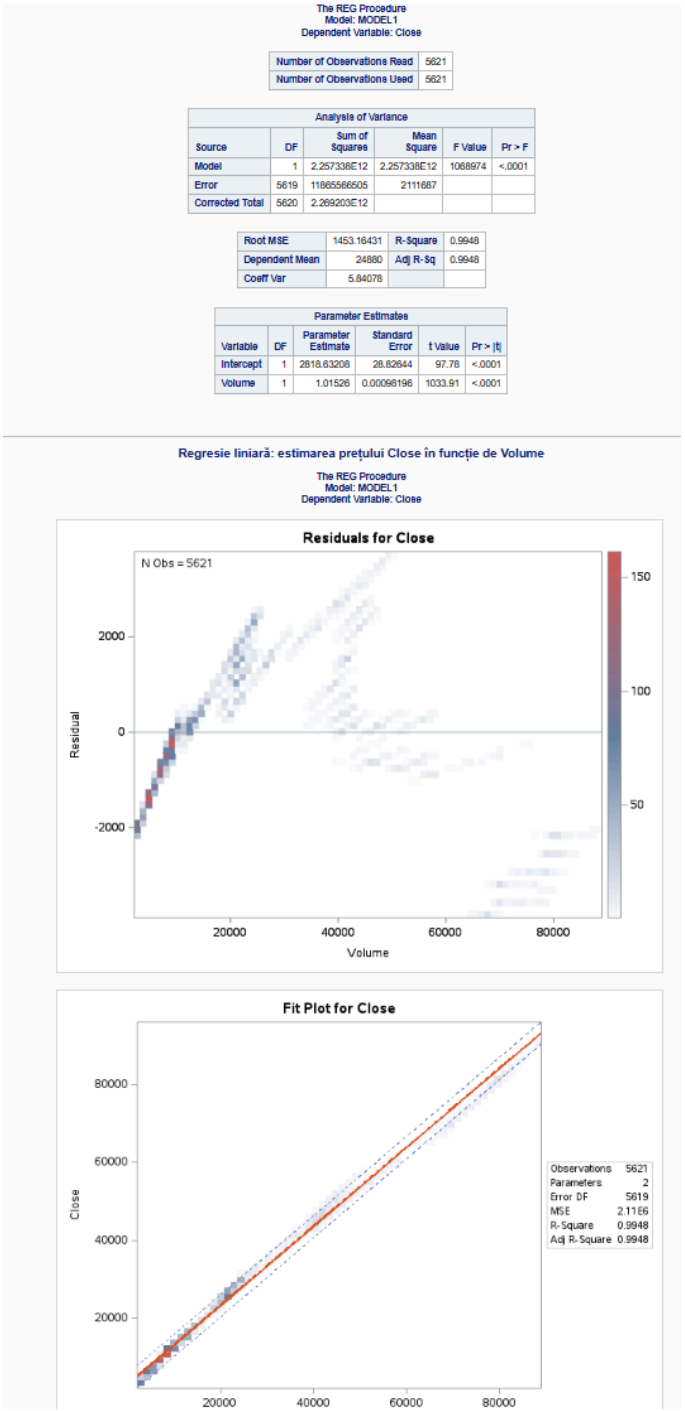


Figure 34. Regresie liniară: estimarea prețului Close în funcție de Volume

Modelul de regresie liniară indică o relație foarte puternică între volumul tranzacționat și prețul de închidere (R pătrat = 0.9948). Fiecare unitate în plus la Volume determină, în medie, o creștere cu 1.10 în Close, coeficientul fiind semnificativ statistic.

9. Generarea de grafice

Pentru a vizualiza cum a evoluat prețul Close pe parcursul anilor, au fost folosite Date și Close, iar ca metodă de calcul proc gplot cu simboluri și interpolare liniară.

```
symbol value=dot i=join color=blue width=1;
title "Evolutia in timp a pretului de inchidere";

proc gplot data=proiect.samsung_perm;
   plot Close*Date;
run;
quit;
```

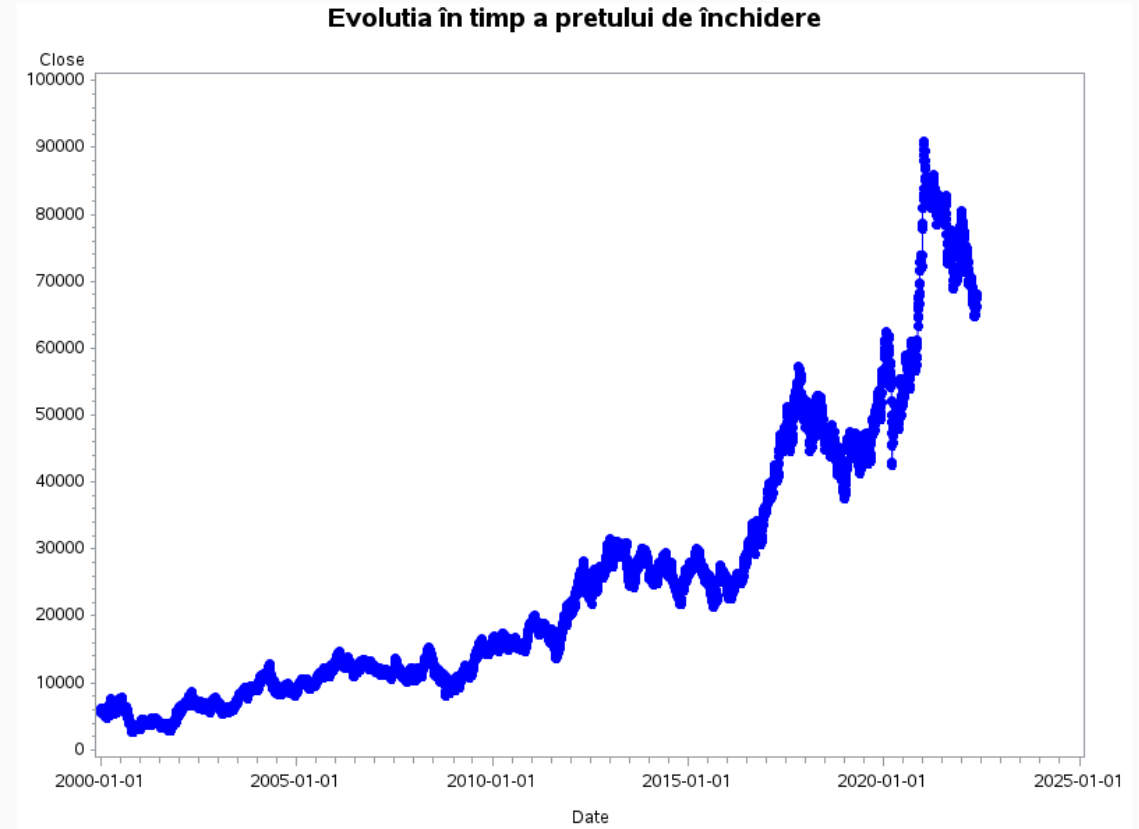


Figure 35. Evoluția în timp a prețului de închidere

Dorim să vizualizăm distribuția volumului tranzacționat zilnic. Au fost folosite datele Volume, iar metoda aleasă a fost proc univariate cu histogram.

```
proc univariate data=proiect.samsung_perm noprint;
   var Volume;
   histogram Volume;
   title "Distribuția volumului tranzacționat";
run;
```

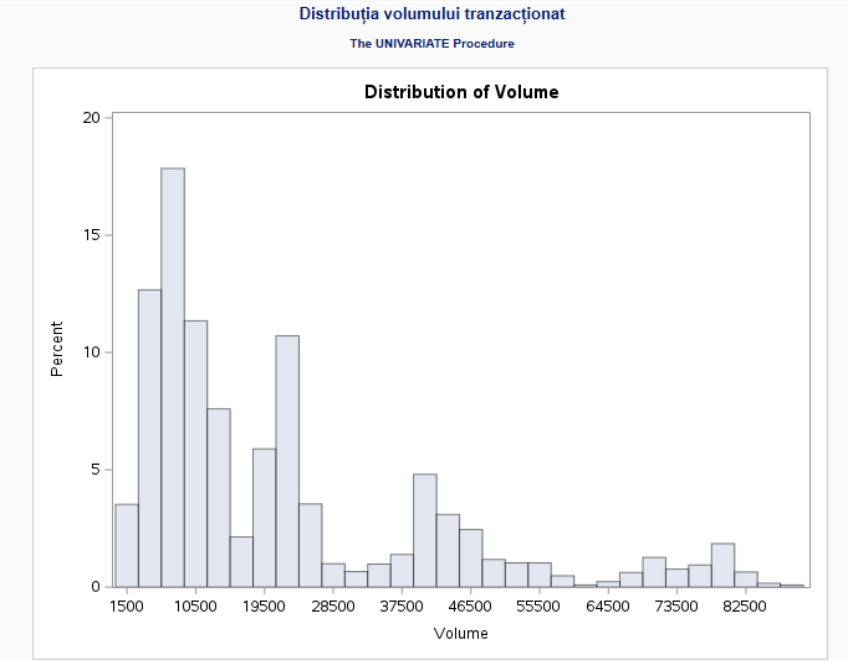


Figure 36. Distribuția volumului de tranzacționat

Vom afișă totalul volumului tranzacționat pentru fiecare an într-un grafic cu bare. Au fost folosite year(Date) și Volume, cu proc gchart și opțiuni sumvar, type.

```
data vol_per_an;
   set proiect.samsung_perm;
   An = year(Date);
run;

title "Volumul tranzacționat pe ani – grafic cu bare";
pattern value=solid;

proc gchart data=vol_per_an;
   vbar An / sumvar=Volume type=sum;
run;
quit;
```

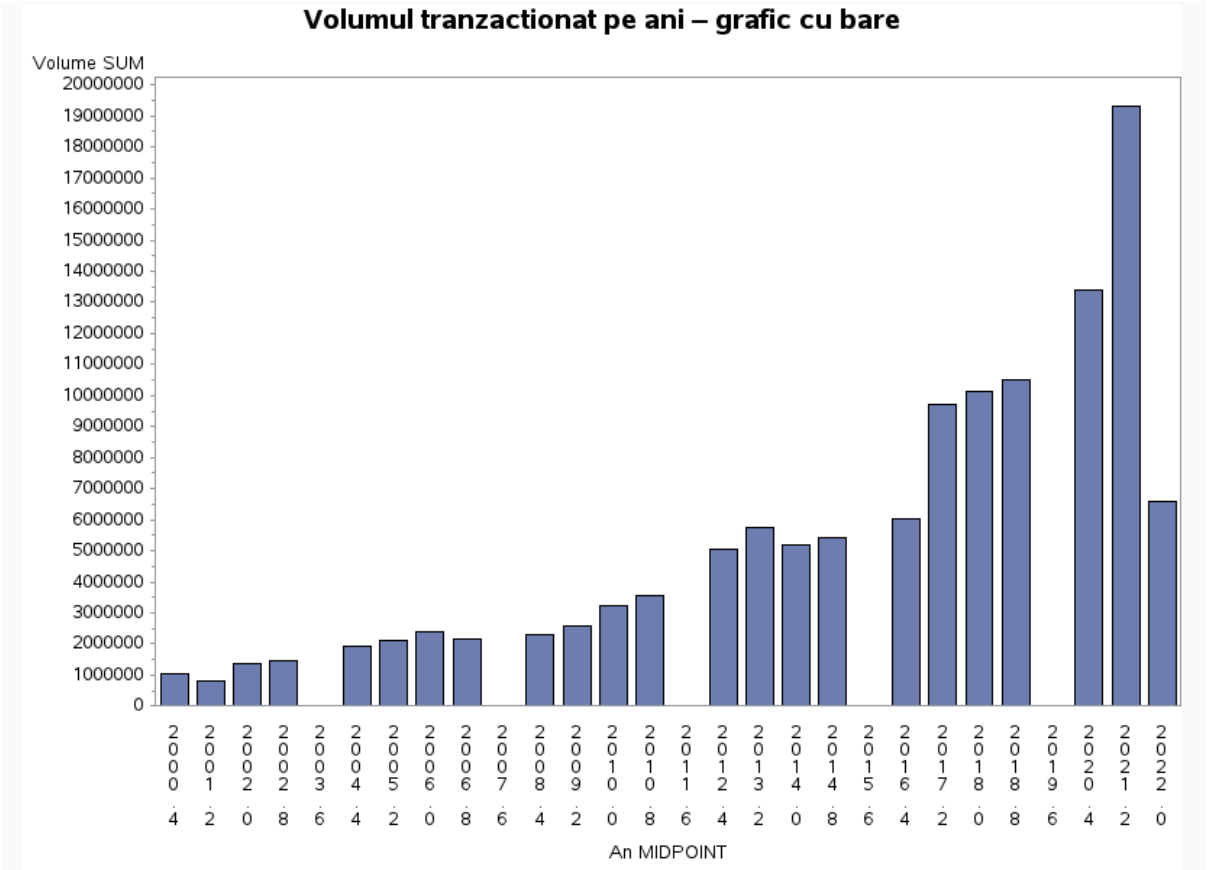


Figure 37. Volumul tranzacționat pe ani

10. Utilizarea SAS ML

În contextul investițiilor bursiere, anticiparea variației prețului acțiunilor este esențială pentru luarea unor decizii informate. Pe baza datelor din fișierul propus, se poate construi un model de învățare automată care să prezică dacă în ziua următoare prețul acțiunii va crește sau nu.

Această problemă poate fi formulată ca o clasificare binară, în care variabila țintă (Target) va avea valoarea:

- 1 dacă prețul de închidere a doua zi (Close_t+1) este mai mare decât cel din ziua curentă (Close_t);

- 0 în caz contrar.

Pentru evitarea conflictelor ce pot fi cauzate de exercițiile anterioare, vom citi din nou setul de date, lucrând cu el din biblioteca work. La început, vom crea variabila țintă.

```
data work.samsung_target;
   set work.samsung;
   Close_t = Close;
   Close_t_plus1 = lag(Close);
   Target = (Close_t_plus1 > Close_t);
run;

data work.samsung_target;
   set work.samsung_target;
   if _N_ > 1;
run;
```

În plus, vom crea variabile cu valoare predictivă, și anume volatilitatea în timpul zilei, dacă acțiunea a crescut în timpul zilei și volatilitatea relativă la prețul de deschidere.

```
data work.samsung_features;
   set work.samsung_target;

   DailyRange = High - Low;
   PriceChange = Close - Open;
   Volatility = (High - Low) / Open;

   format DailyRange PriceChange Volatility 8.4;
run;
```

Cum nu avem valori lipsă, putem trece la standardizarea valorilor, dat fiind faptul că un astfel de algoritm este sensibil la magnitudine.

```
      /* remove la target pt scalare */
      data work.samsung_for_scaling;
         set work.samsung_features;
         retain Target;
      run;
      proc stdize data=work.samsung_for_scaling
                  out=work.scaled_vars
                  method=range;
         var Open High Low Close Volume DailyRange PriceChange Volatility;
      run;
      data work.samsung_scaled;
         merge work.scaled_vars work.samsung_features(keep=Target);
      run;
data work.samsung_scaled;
   set work.samsung_scaled;
   Adj_Close = 'Adj Close'n;
   drop 'Adj Close'n;
run;
```

Până acum, setul de date este de forma:

Obs	Date	Open	High	Low	Close	Adj Close	Volume	Close_t	Close_t_plus1	Target	DailyRange	PriceChange	Volatility
1	2000-01-05	0.0371467639	0.0350914504	0.0355994488	0.0322873003	4248.232422	0.4547696617	5580	6110	1	0.0740	0.3609	0.6033
2	2000-01-06	0.0365770283	0.032113994	0.0362884704	0.0327404554	4278.686523	0.3312121304	5620	5580	0	0.0274	0.3709	0.2254
3	2000-01-07	0.0344120328	0.030944279	0.0337620579	0.0318341452	4217.780273	0.2454404287	5540	5620	1	0.0425	0.3832	0.3613
4	2000-01-10	0.0348678213	0.0320076563	0.0362884704	0.034439787	4392.884766	0.285479402	5770	5540	0	0.0260	0.4045	0.2198
5	2000-01-11	0.0373746582	0.0355168014	0.0384703721	0.034439787	4392.884766	0.363821819	5770	5770	0	0.0452	0.3799	0.3674
6	2000-01-12	0.0349817685	0.0316886431	0.0365181442	0.0338733432	4354.818359	0.17793746	5720	5770	1	0.0192	0.3978	0.1617
7	2000-01-13	0.0348678213	0.0316886431	0.0360587965	0.0337600544	4347.205078	0.250829705	5710	5720	1	0.0247	0.3978	0.2083
8	2000-01-14	0.0362351869	0.0331773713	0.0374368397	0.0351195197	4438.565918	0.3006728983	5830	5710	0	0.0274	0.3978	0.2266
9	2000-01-17	0.0394257065	0.0363675032	0.040192926	0.0381783165	4644.125	0.3867186311	6100	5830	0	0.0356	0.3966	0.2808
10	2000-01-18	0.0412488605	0.0361548277	0.0408819476	0.0381783165	4644.125	0.2756142861	6100	6100	0	0.0247	0.3788	0.1893

Figure 38. Setul de date ML

În continuare, vom împărți setul în două subseturi: de antrenament (70%) și de tesare (30%).

```
%let train_prop = 0.7;

proc surveyselect data=work.samsung_scaled out=work.samsung_split outall
  method=srs
  rate=&train_prop
  seed=123;
run;

data work.train work.test;
   set work.samsung_split;
   if selected then output work.train;
   else output work.test;
run;
```

The SURVEYSELECT Procedure

Selection Method	Simple Random Sampling

Input Data Set	SAMSUNG_SCALED
Random Number Seed	123
Sampling Rate	0.7
Sample Size	3934
Selection Probability	0.7
Sampling Weight	0
Output Data Set	SAMSUNG_SPLIT

Figure 39. Împărțirea setului de date

Vom stoca variabilele independente într-o macrovariabilă:

```
proc sql noprint;
  select name into :indepVars separated by ' '
  from dictionary.columns
  where libname='WORK' and memname='SAMSUNG_SCALED'
       and name not in ('Target' 'Close_t' 'Close_t_plus1' 'selected');
quit;

%put &indepVars;
```

Astfel, datele sunt pregătite pentru a antrena, pentru început, un model de regresie logistică. Acesta va prezice probabilitatea ca Target să fie 1.

```
proc logistic data=work.train descending outmodel=work.logitmodel;
  model Target(event='1') = &indepVars;
  score data=work.test out=work.test_pred_logit outroc=roc_logit;
run;
```

The LOGISTIC Procedure

Model Information	
Data Set	WORK.TRAIN
Response Variable	Target
Number of Response Levels	2
Model	binary logit
Optimization Technique	Fisher's scoring

Number of Observations Read	3934
Number of Observations Used	3934

Response Profile		
Ordered Value	Target	Total Frequency
1	1	1818
2	0	2116

Probability modeled is Target=1.

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	5433.087	3984.554
SC	5439.364	4041.051
-2 Log L	5431.087	3966.554

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	1464.5327	8	<.0001
Score	917.2773	8	<.0001
Wald	687.3251	8	<.0001

Figure 40. Output regresie logistica 1

DailyRange =	0.04658 * Intercept + 12.8822 * High - 11.9288 * Low
PriceChange =	0.4067 * Intercept - 9.80559 * Open + 9.86257 * Close

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-4.3132	0.9229	21.8406	<.0001
Date	1	0.000164	0.000054	9.1503	0.0025
Open	1	346.0	23.6599	213.8433	<.0001
High	1	129.9	24.9330	27.1475	<.0001
Low	1	157.8	33.5209	22.1487	<.0001
Close	1	-629.3	29.3402	459.9986	<.0001
Volume	1	0.3126	0.6743	0.2150	0.6429
DailyRange	0	0	.	.	.
PriceChange	0	0	.	.	.
Volatility	1	1.9229	0.8165	5.5461	0.0185
Adj_Close	1	-0.00003	0.000041	0.5148	0.4732

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Date	1.000	1.000	1.000
Open	>999.999	>999.999	>999.999
High	>999.999	>999.999	>999.999
Low	>999.999	>999.999	>999.999
Close	<0.001	<0.001	<0.001
Volume	1.367	0.365	5.126
Volatility	6.840	1.381	33.891
Adj_Close	1.000	1.000	1.000

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	85.3	Somers' D	0.707
Percent Discordant	14.7	Gamma	0.707
Percent Tied	0.0	Tau-a	0.351
Pairs	3846888	c	0.853

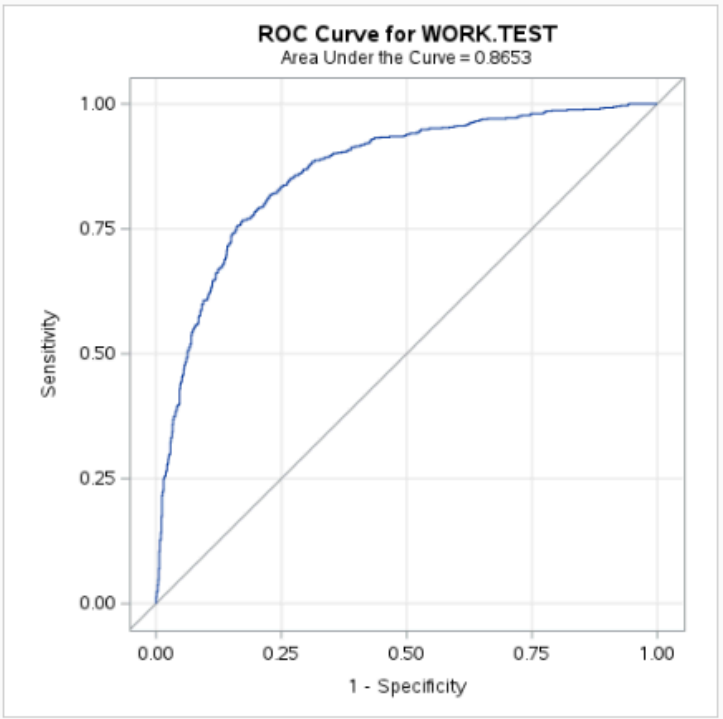


Figure 41. Output regresie logistica 2

Variabilele legate de preț (Open, High, Low, Close) au o influență majoră și semnificativă asupra probabilității ca prețul să crească în ziua următoare. Adj_Close nu aduce valoare adăugată modelului.

c = 0.853 (AUC ROC) => Modelul are foarte bună capacitate de discriminare (85.3%) și clasifică corect în majoritatea cazurilor.

În continuare, am convertit probabilitățile în clase binare, afișând matricea de confuzie pentru cele două praguri (0.3 și 0.5).

```
data work.test_pred_logit;
   set work.test_pred_logit;
   pred_class_05 = (P_1 > 0.5);
   pred_class_03 = (P_1 > 0.3);
run;

title 'Matrice de confuzie – Regresie logistică, prag 0.5';
proc freq data=work.test_pred_logit;
   tables Target * pred_class_05 / nocol;
run;

title 'Matrice de confuzie – Regresie logistică, prag 0.3';
proc freq data=work.test_pred_logit;
   tables Target * pred_class_03 / nocol;
run;
```

Matrice de confuzie – Regresie logistică, prag 0.5

The FREQ Procedure

Frequency Percent Row Pct	Table of Target by pred_class_05		
	pred_class_05		
Target	0	1	Total
0	772 45.79 85.02	136 8.07 14.98	908 53.86
1	208 12.34 26.74	570 33.81 73.26	778 46.14
Total	980 58.13	706 41.87	1686 100.00

Matrice de confuzie – Regresie logistică, prag 0.3

The FREQ Procedure

Frequency Percent Row Pct	Table of Target by pred_class_03		
	pred_class_03		
Target	0	1	Total
0	455 26.99 50.11	453 26.87 49.89	908 53.86
1	48 2.85 6.17	730 43.30 93.83	778 46.14
Total	503 29.83	1183 70.17	1686 100.00

Figure 42. Matricea de confuzie

Pentru pragul de precizie de 0.5, am creat manual matricea de confuzie pentru afișarea unui raport de evaluare mai intuitiv.

```
data evaluare;
   set work.test_pred_logit;

   TP = (Target = 1 and pred_class_05 = 1);
   TN = (Target = 0 and pred_class_05 = 0);
   FP = (Target = 0 and pred_class_05 = 1);
   FN = (Target = 1 and pred_class_05 = 0);
run;

proc means data=evaluare sum noprint;
   var TP TN FP FN;
   output out=confuzie_sum sum=TP TN FP FN;
run;


data evaluare_finala;
   set confuzie_sum;

   Total = TP + TN + FP + FN;
   Accuracy = (TP + TN) / Total;
   Precision = TP / (TP + FP);
   Recall = TP / (TP + FN);
   F1 = 2 * (Precision * Recall) / (Precision + Recall);
run;

title "Raport de evaluare a modelului logistic (prag 0.5)";
proc print data=evaluare_finala label noobs;
   var TP TN FP FN Accuracy Precision Recall F1;
   label
      TP = "True Positive"
      TN = "True Negative"
      FP = "False Positive"
      FN = "False Negative"
      Accuracy = "Acuratețe"
      Precision = "Precizie"
      Recall = "Recall (Sensibilitate)"
      F1 = "F1-Score";
run;
```

Raport de evaluare a modelului logistic (prag 0.5)

True Positive	True Negative	False Positive	False Negative	Acuratețe	Precizie	Recall (Sensibilitate)	F1-Score
570	772	136	208	0.79597	0.80737	0.73265	0.76819

Figure 43. Raport de evaluare a modelului logistic

Modelul de regresie logistică antrenat pentru a prezice creșterea zilnică a prețului acțiunilor Samsung a obținut rezultate foarte bune la pragul de decizie standard de 0.5. Cu o acuratețe de aproximativ 80%, modelul a fost capabil să clasifice corect majoritatea cazurilor. De asemenea, a prezentat o precizie ridicată (0.807) și un recall bun (0.733), reușind să detecteze majoritatea zilelor reale de creștere. Scorul F1 (0.768) reflectă un echilibru solid între precizie și sensibilitate, ceea ce face ca acest model să fie eficient.