

IMPORTBLOOM

Stata package to import Bloomberg exported data

Nicola Tommasi
C.I.D.E.
nicola.tommasi@univr.it

1. Il comando importbloom

importbloom è un nuovo comando Stata che permette

- di importare in Stata i dati esportati tramite una particolare procedura che usa le API Bloomberg di Excel
- di trasformarli in serie storica
- di convertire in numeriche le variabili create

I dati devono essere stati esportati tramite la procedura "Creare spreadsheet" -> "Tabella dati storici" e devono rispettare la disposizione-layout mostrata in figura 1.

Figura 1 – Layout dei dati

Notare che non deve essere selezionato "Vedere nome campo". Se i dati esportati presentano questa riga bisogna selezionarla, eliminarla e risalvare il file. A partire dalla cella di partenza ci deve essere la riga con il "Titolo" e subito dopo la riga con "Data" e "Campo" come si vede in figura 2

	A	B	C	D	E	F
1	Data iniziale	CY 2007				
2	Data finale	CY 2020				
3						
4		FR0000120404 ISIN				
5	Dates	ENTERPRISE_VALUE	CF_FREE_CASH_FLOW_FIRM	FREE_CASH_FLOW_EQUITY	CF_FREE_CASH_FLOW	WACC
6	31/12/2007	11997.6209	-180.5602	1198	-249	11.2288
7	31/12/2008	9100.4967	-453.3492	955	-524	9.857
8	31/12/2009	10400.7761	-453.3492	17	-325	9.6987
9	31/12/2010	8525.2385	-453.3492	-1490	44	11.8169
10	30/12/2011	4868.7196	140.0526	-250	83	10.1933
11	31/12/2012	6613.1855	-818.6639	8	-849	8.9421

Figura 2 – Esempio di dati esportati e conformi al comando

2. Sintassi

Per installare il comando, dalla command bar di Stata digitate:

```
net from https://raw.githubusercontent.com/NicolaTommasi8/importbloom/master/
```

Otterrete questo output

...

quindi cliccate su [importbloom](#) e alla pagina successiva su [\(click here to install\)](#).

...

La sintassi del comando è la seguente:

```
importbloom using filename, cellrange([start][:end]) datastart(string) nvar(#)  
    lasttick(string) [sheet("sheetname") from(string) to(string) clear ]
```

dove:

filename è il percorso e il nome del file Excel dei dati esportati

cellrange([start][:end]) specifica il range di celle da importare dal foglio excel. Il range viene specificato usando la notazione di Excel, per esempio `cellrange(A1)` o `cellrange(A1:BC2000)`

datastart(string) specifica la colonna da cui partono i dati. Quindi è la colonna dopo la data, dove iniziano i dati. Forse non serve, perché potrebbe essere la colonna `cellrange+1`

nvar(#) specifica il numero di campi esportati per ciascun titolo

sheet(sheetname) importa il foglio *sheetname*. Come default importa il primo foglio del file Excel.

from(varlist) lista dei campi che devono essere rinominati, tipicamente perché il nome è troppo lungo o perché il nome sarebbe incompatibile con le regole di Stata sui nomi delle variabili (è ancora da vedere bene se farlo, come farlo e dove piazzarlo)

to(varlist) lista dei nuovi nomi da assegnare ai campi specificati in `from(varlist)`

clear elimina i dati in memoria prima di caricare i dati dal file Excel

2.1. Esempi

Nel primo esempio si importano i dati dal foglio "Foglio1" dove ci sono 32 campi per ciascun ticker e l'ultimo di questi ticker è nella colonna DEP del foglio.

```
. importbloom using "data/Vantaggio competitivo e WACC.xlsx", cellrange(A4) ///  
> sheet("Foglio1") datastart(B) nvar(32) lasttick(DEP)
```

```
. summ
```

Variable	Obs	Mean	Std. dev.	Min	Max
-----+					
date	0				
ENTERPRISE~E	1,242	11074.45	22131.1	5.125	275673.4
CF_FREE_CA~M	1,150	464.8148	1048.991	-3928.55	8677.674
FREE_CASH_~Y	1,220	535.3761	1699.12	-5986	21368
CF_FREE_CA~W	1,252	322.9462	957.2912	-5003	8431
-----+					
WACC	1,250	7.00434	2.775853	-25.4714	21.747
WACC_COST_~Y	1,250	9.316001	3.114569	-25.6074	23.8439
WACC_WEIGH~Y	1,250	71.21741	22.55256	1.8608	100
WACC_COST_~T	1,250	1.568361	1.5902	-.9692	8.7885
WACC_WEIGH~T	1,250	28.67527	22.45683	0	98.1392
-----+					
SHORT_AND_~T	1,252	3320.436	7371.259	0	66354
CASH_AND_M~S	1,254	1399.036	3407.764	.083	36060
FCF~R_GROWTH	712	11.17473	23.38947	-72.5478	226.1581

FREE_CASH_~H		850	9.434474	28.7091	-54.3235	310.6351
FCF~L_GROWTH		1,088	88.41307	1820.618	-19456.96	32075.89

HISTORICAL~P		93	227.6864	316.9819	14.3072	1669.34
EBIT		1,254	658.7461	1528.055	-16169	13266
IS_INC_TAX~P		1,254	150.5178	341.3357	-1689	2932
CAPITAL_EX~D		1,252	-541.5659	1569.208	-16797	0
CF_CASH_FR~T		1,252	-729.68	2334.02	-34399	14286

IS_IMPAIR~S		745	96.29379	481.9777	-103	9158
IS_IMPAIR~L		834	101.086	469.6049	-11	4786
IS_IMPAIR_~S		475	36.19235	100.9268	-50	1126
CF_CHNG_NO~P		1,238	-8.101132	397.3753	-2390	8759
RETURN_COM~Y		1,246	8.061317	28.50765	-527.0218	239.998

RETURN_ON_~T		1,248	3.600192	15.08787	-136.1412	429.4941
RETURN_ON_~P		1,221	5.655745	22.31824	-231.2528	139.3538
RETURN_ON_~L		1,221	5.172286	25.98726	-734.1741	131.1622
OPER_MARGIN		1,254	43.76568	2343.848	-11153.5	81623.19
PE_RATIO		93	46.10994	106.7487	.1578	1025

PX_TO_BOOK~O		93	1.471926	1.489012	.2286	6.6377
EV_TO_T12M~A		1,226	53.0463	851.3633	.4938	21008.69
NET_DEBT_T~A		1,236	9.017561	169.2792	-38.8095	4180
ticker		0				

Nel secondo esempio il numero di campi è 13 e l'ultimo ticker è posizionato nella colonna GQO

```
. set maxvar 8000
```

```
. importbloom using "data/us_banks.xlsx", cellrange(A4) sheet("US1") ///
> datastart(B) nvar(13) lasttick(GQ0)
```

```
. summ
```

Variable		Obs	Mean	Std. dev.	Min	Max

date		0				
NET_INT_INC		26,861	94.85882	565.9889	-449.202	12644
IS_ACT_LOA~T		25,901	12.36292	151.3245	-576	16609
BS_TOT_LOAN		26,745	8090.337	46071.2	0	1033521
BS_LOAN_MTG		26,751	7966.754	45301.14	0	1022258

BS_TOTAL_D~S		3,903	457.9059	2468.646	-2.345	37205
BS_DEMAND_~T		26,336	2634.726	16761.92	0	529051
INTEREST_B~O		26,605	6670.138	40938.49	0	996854
BS_CUSTOME~S		26,736	9263.505	57186.44	0	1482479
TOTAL_EQUITY		26,783	1384.682	8689.521	-.385	208079

BS_TIER1_C~O		26,077	13.74289	80.34161	0	12960
BS_NON_PER~T		26,140	104.7876	875.102	0	34573
BS_TOT_ASSET		26,751	12827.44	82475.08	0	1981349
BS_CASH_NE~M		26,738	283.0214	1641.788	0	63904
ticker		0				

Nel file "imp_bloom.do", oltre al codice per replicare i due precedenti esempi, sono riportati ulteriori esempi relativi a dati presenti nel file testXimportbloom.xlsx.

3. Note Finali

`importbloom` usa il comando aggiuntivo `xframeappend` di Roger Newson (Imperial College London, United Kingdom) e include un pezzo di codice scritto da William Matsuoka per convertire l'identificativo delle colonne di Excel, nel corrispondente numero di colonna. Trovate l'articolo e il codice qui: [Putexcel Part II: numofbase26\(\)](#).

Dati esportati con layout differente da quello previsto per il comando `importbloom` dovrebbero essere importabili abbastanza facilmente con poche righe di codice. Se così non fosse, provate a sottoporvi il caso e vediamo cosa si può fare.