

# **ANALISI ALGORITMICA CON MATLAB**

Di Rizzo Roberto

## **PREMESSA**

Nel vasto campo del trading on line, e più specificatamente nell'Analisi Tecnica, software come Excel e MatLab sono quelli tra i più utilizzati, oltre i software specifici.

Questo lavoro si occupa in maniera particolare dell'analisi algoritmica, prendendo in considerazione uno dei tanti indici utilizzati in questo campo, le medie mobili, delle quali la più comune interpretazione è il confronto tra la media mobile del titolo e il prezzo del titolo stesso.

Verranno sviluppati degli algoritmi in grado di automatizzare tutta una procedura costituita da: calcolo delle medie mobili, successivo confronto con i prezzi dei titoli, segnali di operatività e visualizzazione a video delle medie mobili con maggiori performance in termini di rendimento. Il software utilizzato in questo caso è MatLab(Versione 7.0)<sup>1</sup>.

---

<sup>1</sup> Matrix Laboratory E' un sistema interattivo ed un linguaggio di programmazione per il calcolo tecnico e scientifico. Dotato di potenti istruzioni che rendono molto facili la programmazione.

## **MEDIE MOBILI E LORO OPERATIVITA'**

Le **medie mobili** (Moving Averages) sono sicuramente gli indicatori più utilizzati in analisi tecnica e rappresentano uno strumento insostituibile per depurare i prezzi dagli elementi erratici (rumore di fondo) che possono confondere l'analista.

Lo scopo delle medie mobili non è quello di anticipare i movimenti di prezzo o prevederne le evoluzioni, ma quello di seguire e identificare chiaramente il **trend** in atto.

Esistono diversi tipi di medie mobili che differiscono tra loro per la formula di calcolo utilizzata:

- **media mobile semplice** (*SMA=simple moving average*)
- **media mobile centrata** (*CMA=centered moving average*)
- **media mobile ponderata** (*WMA=weighted moving average*)
- **media mobile esponenziale** (*EMA=exponential moving average*)

Per tutti i tipi di medie mobili vale la seguente considerazione:

Quanto maggiore è il dominio temporale della media mobile tanto più tempo dovrà trascorrere per poter iniziare a rappresentarla.

In questo lavoro le medie mobili prese in considerazione sono quelle semplici, calcolate con l'ausilio di MatLab per un lasso temporale che va da 1 a 100 giorni.

Quello che ci si aspetta da esse, è la loro adattabilità a fornire segnali di vendita o acquisto, per aiutare a disinteressarsi dei momenti di crollo o di impennata dei prezzi, ed a concentrare l'attenzione sui "crossing-points"<sup>2</sup> della media sulla serie dei prezzi, i quali costituiranno segnali di acquisto o di vendita.

L'utilizzo delle medie mobili come trading system è un metodo per individuare quindi il momento ideale di acquisto e vendita e si usa in questo modo:

- si acquista quando il prezzo perfora verso l'alto la propria media e si vende quando la perfora verso il basso; si tratta del più elementare sistema di trading, soggetto a frequenti falsi segnali ed efficace solo nei momenti di tendenza ben definita;

---

<sup>2</sup> Punti nei quali le medie mobili incrociano i prezzi dei titoli dall'alto o dal basso.

## **ALGORITMI PER IL CALCOLO DELLE MEDIE MOBILI E SEGNALI DI OPERATIVITA'**

Il primo passo da fare nella costruzione degli algoritmi con il software MatLab, è il caricamento di una serie di dati da un file formato excel o teso. Le istruzioni necessarie in questo caso sono:

(le istruzioni precedute dal segno %, sono commenti) :

```
%Caricamento dati da file formato excel.
```

```
clear
```

```
S=xlsread('FIAT');
```

Successivamente si procede con il calcolo delle medie mobili da uno a cento giorni, ed all'inserimento delle medie in una matrice di colonne cento, e righe variabili a seconda della lunghezza della serie.

Le istruzioni sono le seguenti:

```
%Separa le colonne
```

```
xA=S(:,1);
```

```
%xA=Apertura.
```

```
xM=S(:,2);
```

```
%xM=Massimo giornaliero.
```

```
xm=S(:,3);
```

```
%xm=Minimo giornaliero.
```

```
xC=S(:,4);
```

```
%xC=Chiusura.
```

```
xV=S(:,5);
```

```
%xV=Volumi scambiati.
```

```
xCA=S(:,5);
```

```
%xCA=Chiusura aggiustata.
```

```
i=1;
```

```

%i=Indice delle righe della matrice.
r=1;
%r=Indice delle colonne della matrice.
MM=[];
%Matrice delle medie mobili.
L=length(xC);
%L=lunghezza del vettore dei valori di chiusura.
for r=1:100
    j=1;
    %j=Indica il valore di partenza dal quale si calcola la media mobile.
    for i=r:L
        MM(i,r)=sum(xC(j:i))/r;
        j=j+1;
    end
end
end

```

Il prossimo passo è quello di leggere le istruzioni ed assegnare ad ogni *corssing point* il rispettivo segnale. Per questo tipo di operazione l'algoritmo è leggermente più complesso ed è formato dalle seguenti istruzioni:

```

%Si inizializzano nuovamente i rispettivi indici.
r=1;
i=1;
V=[];
%Matrice delle vendite.
A=[];
%Matrice degli acquisti.
H=[];
%Matrice dei valori sui quali vi è nessuna operazione.
for r=1:100
    sa=0;
    %E' un valore che tiene conto degli acquisti effettuati prima di ogni
    %vendita, se sa=0 allora significa che la vendita non viene presa in
    %considerazione perchè non vi è il prezzo d'acquisto.
    for i=r+1:L-1
        if xC(i-1)-MM(i-1,r)<0 & xC(i)-MM(i,r)>=0
            A(i,r)=xA(i+1);

```

```
    sa=sum(A(1:i,r));
elseif xC(i-1)-MM(i-1,r)>=0 & xC(i)-MM(i,r)<0 & sa>0
    V(i,r)=xA(i+1);
else
    H(i,r)=xA(i+1);
end
end
end
end
```

## ALGORITMI PER LA CREAZIONE DEGLI UTILI TOTALI E PARZIALI

In questa fase si creano degli algoritmi in grado di calcolare sia gli utili totali provenienti dai segnali di operatività e gli utili parziali.

Le istruzioni utilizzate in questa fase sono:

```
%In questa fase verranno creati degli algoritmi in grado di calcolare gli
%utili relativi ad ogni singolo acquisto e vendita.
%Inizializzazione indici, r=colonne, i =righe.
r=1;
i=1;
v=1;
%L'indice v ci permette di creare una matrice nella quale ci siano solo i
%valori di Vendita senza nessun valore = 0.
LV=length(V);
%LV=Numero delle righe della matrice V(Vendite).
Ven=[];
%Ven=Nuova matrice delle vendite, dove compariranno solo i valori diversi
%da 0.
for r=1:100
    v=1;
    for i=1:LV;
        if V(i,r)~=0;
            Ven(v,r)=V(i,r);
            v=v+1;
        end
    end
end
%Stesso procedimento per la nuova matrice degli acquisti.
%Inizializzazione degli indici.
r=1;
i=1;
a=1;
%L'indice a, ha la stessa funzione dell'indice v visto sopra.
```

```
Aqu=[];
% Aqu=Nuova matrice degli acquisti dove compariranno solo i valori diversi da
% 0.
LA=length(A);
for r=1:100
    a=1;
    for i=1:LA;
        if A(i,r)~=0
            Aqu(a,r)=A(i,r);
            a=a+1;
        end
    end
end

% Inizializzazione indici.
i=1;
r=1;
e=1;
% Assegnamo la Matrice U e il vettore UT.
U=[];
UT=[];
lv=length(Ven);
% Questo algoritmo ci permette di creare una matrice in grado di tenere
% traccia del numero di operazioni che si svolgono seguendo le medie mobili
% da 1 a 100.
for r=1:100
    e=1;
    if Ven(i,r)~=0
        U(e,r)=1;
    else
        U(e,r)=0;
    end
    e=2;
    for i=2:lv
        if Ven(i-1,r)~=0 & Ven(i,r)==0 | Ven(i,r)~=0
            U(e,r)=1;
        else
            U(e,r)=0;
        end
    end
end
```

```
    end
    e=e+1;
end
UT(r)=sum(U(:,r));
%Nel vettore UT, ogni elemento indica il totale delle operazioni
%per ogni media mobile
end
%Inizializzazione
i=1;
r=1;
EqL=[];
%EqL=Matrici degli utili parziali.
%Con questo algoritmo viene creata una matrice, le cui colonne
%rappresentano tutti gli utili parziali derivanti da ogni singola
%operazione
for r=1:100
    for i=1:UT(r)
        EqL(i,r)=Ven(i,r)-Aqu(i,r);
    end
end
G=[];
%Vettore dei Guadagni Totlai.
r=1;
%Indice colonna.
for r=1:100
    G(r)=sum(EqL(:,r));
end
End
```

Finita questa fase, nel wok Space di MatLab abbiamo tutte le informazioni necessarie alla nostra analisi. Quindi il passo successivo sarà quello di visualizzare dei grafici nei quali ci siano le medie mobili con le performance migliori in termini di rendimento.

## ALGORITMI PER I GRAFICI

Come detto sopra, in questa fase verranno costruiti dei grafici sui quali è possibile effettuare delle considerazioni, naturalmente il grafico che viene fuori dovrà essere il risultato di un confronto dettagliato delle informazioni possedute.

Quindi anche per la grafica abbiamo bisogno di algoritmi che individuano le medie mobili con le caratteristiche migliori.

Questa fase nel software MatLab è dettata dalle seguenti istruzioni:

```
p=1;
%Con questo algoritmo, verrà esposto a video il grafico della media mobile
%che ha riportato un utile Totale maggiore; Il num di giorni della media
%mobile in questione;e anche il grafico dell'Equity line della media mobile
%in questione.
for p=1:100
    if G(p)==MXG;
        figure(1);
        plot(xC,'y');
        hold on
        plot(MM(:,p),'r');
        title('Cross-Over between: Close Vs Moving Average with  GREATER TOTAL PROFIT');
        ylabel('Moving Average & Close');
        xlabel('Line Time Observation ');
        axis([100 L 1 MXC+4]);
        disp('La media mobile con Utile Totale maggiore è quella a gg')
        disp(p)
        figure(2);
        plot(EL(:,p),'b');
        hold on
        title('Corresponding Equity Line ');
        ylabel('Profit-Loss');
        axis([0 UT(r) 1 MXC+10]);
```

```

%Le istruzioni "axis" permettono di ottenere dimensioni degli assi

%variabili a seconda della lunghezza dei vettori rappresentati.
%in questo ultimo caso indica che : Asse x va da 0 al numero di
%operazioni effettuate con la media mobile in questione; Asse y va
%da 1 al max valore di chiusura.
end
end
% Una volta visualizzata la media mobile con un utila totale maggiore,
%si analizzala media mobile che genera un utile untiraio maggiore.
%Attrverso questo algoritmo
p=1;
UM=[];
%UM, vettore degli utili unitari massimi pwer ogni media mobile.
for p=1:100
    UM(p)=max(EqL(:,p));
end
M=max(UM);
%M, corrisponde al massimo dei massimi.
p=1;
%Ora con questo algoritmo verrà esposto il grafico della media mobile
%con la quale si ottiene un utile unitario massimo; il num dei giorni delle
%medie mbili con questo valore; e la ripetiva Equity Line.
for p=1:100
    if UM(p)==M
        figure(3);
        plot(MM(:,p),'g');
        hold on
        plot(xC,'y');
        title('Cross-Over between: Close Vs Moving Average with GREATER UNITARY PROFIT');
        axis([100 L 1 MXC+4]);
        ylabel('Moving Average & Close');
        xlabel('Line Time Observation ');
        disp('La media mobile con Utile Unitario Maggiore è quella a gg')
        disp(p)
        figure(4);
        plot(EL(:,p),'b');
        hold on
        title('Corresponding Equity Line ');
    end
end

```

```

        ylabel('Profit-Loss');
        axis([0 UT(r) 1 MXC+10]);
    end
end
%fine parte quinta
%-----
%parte ultima
mm=input('Iserisci la massima perdita percentuale sopportabile,(Basis Point) : ');
%con il comando input è possibile inserire un valore input direttamente
%dall'esecuzione del programma.
% In questo caso verrà chiesto in fase di esecuzione di inserire un valore
%di perdita massimo disposti ad accettare.
GP=[];
%Assegnazione matrice GP
%Questo algoritmo calcola per ogni operazione, il rendimento percentuale.
for r=1:100
    v=1;
    for i=2:UT(r)
        GP(v,r)=(EL(i,r)-EL(i-1,r))/EL(i-1,r);
        v=v+1;
    end
end
r=1;
i=1;
%Questo ciclo for, individui per ogni singola media, i rispettivi
%utili/perdite massime.
for r=1:100
    LM(r)=max(GP(:,r));
    Lm(r)=min(GP(:,r));
end
x=1;
y=0;
%y, in questo caso funge da verio e proprio contatore.
GG=[];
%Assegnazione vettore GG,
%questo algoritmo confronta i valori inseriti dall'input in esecuzione, con
%quelli presenti nelle matrici creati, se rispettano le condizione
%desiderate, allora espone a video qual'è la media mobile che permette di
%avere qualla perdita massima;se invece non esiste un caso per cui valgono

```

```

%le condizioni richieste, espone un messaggio.
for r=1:100
    if Lm(r)>-mm
        GG(x)=LM(r);
        x=x+1;
    else
        y=y+1;
    end
end
BGP=max(GG);
%Questo algoritmo se la ricerca va a buon fine espone rispettivamente:
%la media mobile che soddisfa la richiesta; il suo grafico; e il grafico
%del rispettivo Equity line.
for r=1:100
    if LM(r)==BGP & Lm(r)>-mm y<100;
        disp('Le medie mobili con i parametri desiserati sono quelle a giorni: ');
        disp(r);
        figure(5);
        plot(MM(:,r),'g');
        hold on
        plot(xC,'y');
        title('Cross-Over between: Close Vs Moving Average with corresponding Max Loss');
        axis([100 L 1 MXC+4]);
        ylabel('Moving Average & Close');
        xlabel('Line Time Observation ');
        figure(6);
        plot(EL(:,r),'r');
        hold on

        title('Corresponding Equity Line ');
        ylabel('Profit-Loss');
        axis([0 UT(r) 1 MXC+10]);
    else
        end
end
if y==100;
    disp('Spiacente ma non vi è una perdita inferiore a quella richiesta');
end

```

Come si può notare in questo ultimo script di MatLab, vi è un'istruzione che ci permette di inserire un valore durante l'esecuzione del programma, si tratta della massima perdita che si è disposti ad accettare tra un segnale di operatività e l'altro.

## **CONSIDERAZIONI FINALI**

Partendo dall'obiettivo che si era posto in precedenza, questo lavoro si propone come un ulteriore e alternativo strumento, per automatizzare e rendere immediati calcoli che senza l'ausilio di software come MatLab, risulterebbero molto dispendiosi in termini di tempo, e in molti casi indurrebbero all'errore.

Naturalmente questo lavoro è solo un'introduzione a quello che potrebbe essere un efficiente strumento di automatizzazione dei calcoli per l'analisi tecnica.