



UNIVERSITA' POLITECNICA DELLE MARCHE

FACOLTA' DI INGEGNERIA

Corso di Laurea in Ingegneria Informatica e dell' Automazione

**SVILUPPO E IMPLEMENTAZIONE MATLAB DI UN
SISTEMA DI SUPPORTO ALL'INVESTIMENTO AZIONARIO
"RETAIL"**

**DEVELOPMENT AND IMPLEMENTATION ON MATLAB OF
A "RETAIL" EQUITY INVESTMENT SUPPORT SYSTEM**

Relatore:
Prof. Simone Fiori

Tesi di laurea di:
Aurelio Pacetti

A.A. 2021 / 2022

INDICE

CAPITOLO 1. CONCETTI INTRODUTTIVI.....	4
1.1 Matlab	4
1.2 Discussione generale sugli investimenti azionari.....	5
1.3 Necessità di software di sorveglianza dell'andamento dei titoli in tempo reale in tempi di incrementata volatilità	6
CAPITOLO 2. SPECIFICHE E CARATTERISTICHE DEL SOFTWARE	7
2.1 Spiegazione del programma	7
2.2 I risultati ottenuti con relative analisi	15
CAPITOLO 3. CONCLUSIONI	19
3.1 Considerazioni personali e spunti per il miglioramento	19
SITOGRAFIA	20

*A tutta la mia famiglia.
A mio padre e a mia madre,
a mia sorella,
Agli amici che mi hanno supportato.*

A me, per essere arrivato al mio primo traguardo.

CAPITOLO 1. CONCETTI INTRODUTTIVI

L'obiettivo di tale trattazione è quello di mostrare i risultati ottenuti da un software di supporto all'investimento azionario “*retail*” sviluppato e implementato in *Matlab*.

Nel presente capitolo verranno esposti i concetti chiave necessari alla comprensione del progetto iniziando dall'introduzione alla piattaforma utilizzata per la creazione del sistema di supporto: *Matlab*.

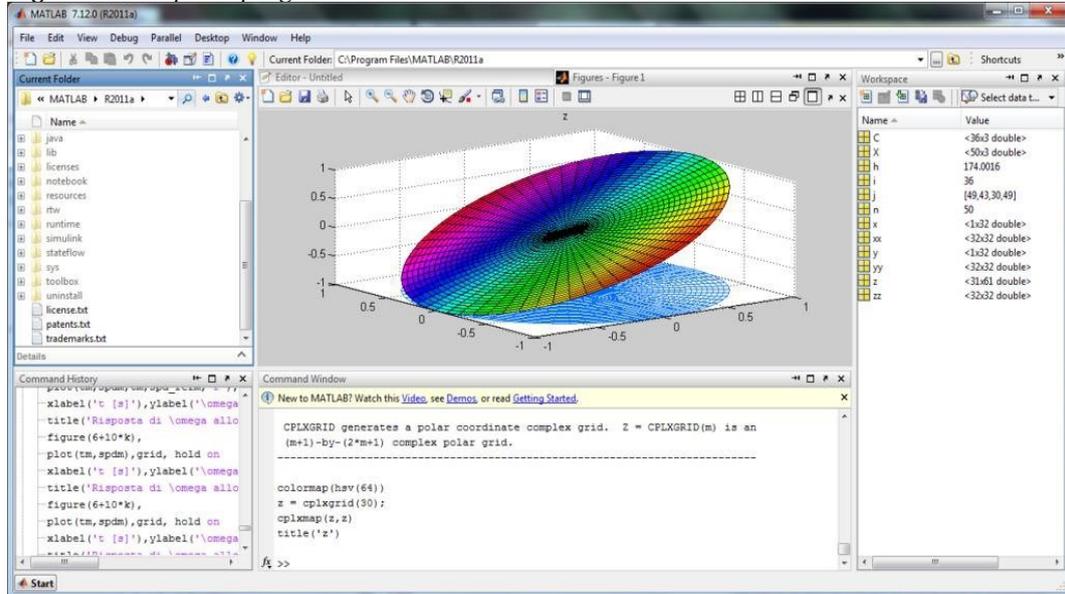
Successivamente verranno illustrate le specifiche e le caratteristiche del software con vari risultati.

1.1 Matlab

Matlab è un ambiente per il calcolo numerico e l'analisi statistica scritto in C, consente di manipolare matrici, visualizzare funzioni e dati, implementare algoritmi, creare interfacce utente, e interfacciarsi con altri programmi.

Nella figura 1 un esempio dell'interfaccia grafica di *Matlab*.

Figura 1. Esempio di programma Matlab



Fonte: Wikipedia

1.2 Discussione generale sugli investimenti azionari

Investire nel mercato azionario per molti è un'attività che spaventa a causa delle oscillazioni dei prezzi e il rischio potenziale di perdere una parte o il valore totale del denaro investito.

Tuttavia comprare azioni e titoli, pur non offrendo alcuna garanzia di guadagno, può accrescere il capitale più rapidamente rispetto ad altri tipi di operazioni

finanziare, offrendo l'opportunità di generare rendimenti in grado di arginare gli effetti dell'inflazione (il denaro fermo nei conti correnti perde valore).

1.3 Necessità di software di sorveglianza dell'andamento dei titoli in tempo reale in tempi di incrementata volatilità

I mercati finanziari possono essere soggetti a periodi di elevata volatilità, dal punto di vista matematico significa che in termini percentuali la distanza del prezzo di un titolo dal suo valore medio in un determinato periodo è significativa.

Di conseguenza certi titoli necessitano un monitoraggio costante.

Il software in questione, dotato di un database/portafoglio utente, è in grado di analizzare contemporaneamente più titoli, riportando graficamente vari dati presi in considerazione per ogni titolo in borsa, dando la possibilità all'utente di ottimizzare i tempi delle proprie ricerche e avere a disposizione tutti i dati necessari per attuare i propri investimenti.

CAPITOLO 2. SPECIFICHE E CARATTERISTICHE DEL SOFTWARE

Tale capitolo descrive la parte fondamentale del progetto, infatti, verrà mostrato il software utilizzato per supportare gli investimenti azionari.

2.1 Spiegazione e avvio del programma

Il programma va a prendere i dati dal sito “Investing.com”, uno dei primi tre siti web finanziari globali che offre quotazioni di mercato, informazioni su azioni, futures, opzioni, analisi, materie prime e un calendario economico.

Successivamente vengono riportati in tempo reale in un grafico attraverso il quale l’utente farà poi le proprie considerazioni sui movimenti e le scelte da fare.

Nella figura 2 viene riportato il database dell’utente con i titoli delle azioni presi in considerazione e il portafoglio dell’utente (con i prezzi medi di carico, il numero delle azioni possedute e il target gain).

Figura 2. Database utente

```
8      %% Inizializzazione dati utente (database utente) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
9      % Indirizzi relativi ai titoli su investing.com e relativi ticker
10     ticker1 = 'UNIR'; % UniEuro
11     ticker2 = 'CEMI'; % Cementir Holding
12     ticker3 = 'TLIT'; % Telecom Italia
13     ticker4 = 'EEMS'; % EEMS Italia
14
15     ticker_vec = {ticker1; ticker2; ticker3; ticker4};
16
17     N = 4;
18
19
20     % Prezzi medi di carico (pmc) e numero azioni possedute
21     pmc_1 = 15.68; n_azioni_1 = 31; tg1 = 50; % Target gain in EUR
22     pmc_2 = 6.4488; n_azioni_2 = 64; tg2 = 50; % Target gain in EUR
23     pmc_3 = 0.2494; n_azioni_3 = 5000; tg3 = 50; % Target gain in EUR
24     pmc_4 = 0.123; n_azioni_4 = 5800; tg4 = 50; % Target gain in EUR
25     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
26
27     pmc_vec = [pmc_1; pmc_2; pmc_3; pmc_4];
28     n_azioni_vec = [n_azioni_1; n_azioni_2; n_azioni_3; n_azioni_4];
29     tg_vec = [tg1; tg2; tg3; tg4];
```

Fonte: progetto Matlab

In seguito, nelle figure 3 e 4 vengono riportate varie inizializzazioni.

Nello specifico ciò che riguarda il grafico (oggetto, pannello e finestra) e diverse variabili necessarie quali i tempi di start e stop del programma.

Si noti nell'immagine seguente, nei commenti in verde a partire da riga 45, una sorta di legenda delle linee presenti nel grafico che vedremo poi nei risultati del programma.

Figura 3. Inizializzazione oggetto grafico e delle variabili utilizzate

```
31      %% Inizializzazione oggetto grafico
32      list_factory = fieldnames(get(groot,'factory'));
33      index_interpreter = find(contains(list_factory,'Interpreter'));
34      for i = 1:length(index_interpreter)
35          default_name = strrep(list_factory{index_interpreter(i)},'factory','default');
36          set(groot, default_name,'latex');
37      end
38      set(groot, 'DefaultLineLineWidth', 1.5);
39      set(groot, 'DefaultAxesFontSize', 12);
40      set(groot, 'DefaultTextFontSize', 10);
41      set(groot, 'DefaultTextInterpreter', 'Latex')
42      set(groot, 'DefaultAxesTickLabelInterpreter', 'Latex');
43
44      %% Inizializzazione
45      % linea nera orizzontale = prezzo medio di carico (in euro);
46      % linea magenta orizzontale = obiettivo da raggiungere in termini di valore
47      % della singola azione (in euro), da leggere sull'asse di sx;
48      % linea rossa orizzontale = zero in corrispondenza della variazione % (asse
49      % di dx)
50
51
52      tnow = datetime('now','Format','HH:mm:ss'); tvec = tnow;
53      tstop = datetime('17:30:00','Format','HH:mm:ss');
54      tstart = datetime('09:00:00','Format','HH:mm:ss');
55      if (tnow > tstop)|| (tnow < tstart), warndlg('Mercato azionario chiuso!'); return; end
56      tnyse = datetime('15:30:00','Format','HH:mm:ss');
57
58      k = 1; % Indice ciclo
59
60      TP = 5; % Tempo di pausa tra interrogazioni differenti (secondi)
```

Fonte: progetto Matlab

Figura 4. Inizializzazione pannello e finestre dei grafici

```
62 % Oggetti grafici
63
64 % Inizializzazione pannello grafico
65 f = figure('Position', [100 80 1400 700]);
66
67 hb_pos_vec = [2.3 1 1 1; 1.3 1 1 1; 2.3 1 1 1; 1.3 1 1 1];
68 hv_pos_vec = [2.7 0.5 1 1; 1.4 0.5 1 1; 2.6 0.015 1 1; 1.35 0.015 1 1];
69
70 for n = 1:N
71     indirizzoPagina{n} = getInvestingPageAddr(ticker_vec{n});
72
73     % Inizializzazione della finestra grafica dei titoli
74     hs_vec(n) = subplot(2,2,n);
75     hp_n1_vec(n) = plot(tnow,nan);
76     hold on; grid on
77     yline(pmc_vec(n),'-k','LineWidth',2)
78     ylabel('Valore in EUR'); title(ticker_vec{n}); xlabel('Orario');
79     tp_vec(n) = (tg_vec(n) + n_azioni_vec(n)*pmc_vec(n))/n_azioni_vec(n); yline(tp_vec(n),'-m','LineWidth',2)
80
81     yyaxis right
82
83     hp_n2_vec(n) = plot(tnow,nan);
84     yline(0,'-r','LineWidth',2); xline(tnyse,'-k','LineWidth',2)
85     ylabel('Variazione %')
86     xlim([tvec tstop])
87     position_vec(n,:) = hs_vec(n).Position;
88     ht_vec(n) = annotation('textbox', position_vec(n,:), 'FitBoxToText', true);
89     hb_vec(n) = annotation('textbox', position_vec(n,:)*diag(hb_pos_vec(n,:)), 'FitBoxToText', true);
90     hv_vec(n) = annotation('textbox', position_vec(n,:)*diag(hv_pos_vec(n,:)), 'FitBoxToText', true);
91
92 end
93 pause(0.5)
94
```

Fonte: progetto Matlab

Una volta inizializzato tutto l'occorrente il programma si avvia andando ad estrarre tutte le informazioni necessarie per poi renderle visibili attraverso i grafici creati.

In particolare, come si può vedere dalla figura 5 a riga 101, vengono estratti i valori della quotazione, del volume, dello scostamento percentuale, del volume medio e il

numero di azioni scambiate nella giornata, per ciascuno dei quattro titoli presi in considerazione.

Successivamente viene calcolato il profitto (che figurerà graficamente in verde se positivo e in rosso se negativo) e vengono rappresentati graficamente tutti i dati analizzati, ripetendo ciclicamente l'operazione ogni cinque secondi, aggiornando di fatto il grafico in tempo reale.

Figura 5. Loop di lettura e rappresentazione dei risultati

```
96 % Loop di lettura e rappresentazione dei risultati
97 disp('Programma avviato!')
98 while tnow < tstop
99     for n = 1:N
100         % Legge le pagine da it.investing.com
101         [g(k,n), percent(k,n), vol(k,n), vol_medio1, azioni1] = getInvestParamWeb(indirizzoPagina{n});
102
103         % Calcola profitto
104         profitto = (q(end,n) - pmc_vec(n))*n_azioni_vec(n);
105
106         % Rappresenta graficamente i dati letti
107         hp_n1_vec(n).XData = tvec; hp_n1_vec(n).YData = q(:,n);
108         hs_vec(n).YAxis(1).Limits = [min([tp_vec(n) pmc_vec(n) q(:,n)'])*0.95 max([tp_vec(n) pmc_vec(n) q(:,n)'])*1.05];
109         hp_n2_vec(n).XData = tvec; hp_n2_vec(n).YData = percent(:,n);
110         hs_vec(n).YAxis(2).Limits = [min(percent(:,n))-1 max(percent(:,n))+1];
111         if profitto >= 0
112             ht_vec(n).BackgroundColor = "green";
113             ht_vec(n).String = ['Profitto lordo: ' num2str(profitto,4) ' EUR'];
114         else
115             ht_vec(n).BackgroundColor = "#D95319";
116             ht_vec(n).String = ['Pendita: ' num2str(-profitto,4) ' EUR'];
117         end
118         hb_vec(n).BackgroundColor = "cyan"; hb_vec(n).String = ['Quotazione: ' num2str(q(end,n),3) ' EUR'];
119         hv_vec(n).BackgroundColor = "yellow"; hv_vec(n).String = ['Volume: ' num2str(CommaFormat(vol(end,n))) ];
120     end
121     % Procede con l'iterazione
122     k = k+1;
123     pause(TP); tnow = datetime('now','Format','HH:mm:ss'); tvec = [tvec; tnow];
124 end
125 disp('Programma giornaliero terminato!')
```

Fonte: progetto Matlab

Nelle successive immagini vengono rappresentate le funzioni utilizzate dal programma.

Figura 6. Funzione che salva il pannello grafico in formato pdf a fine giornata

```
127     %% Salva il pannello grafico in formato PDF alla fine della giornata di contrattazioni
128     SetPDFprint(gcf);
129     figureName = ['Chiusura giornata ', datestr(now, 'dd_mm_yyyy')]; print(figureName, '-dpdf')
```

Fonte: progetto Matlab

Figura 6.1.

```
175     function SetPDFprint(h)
176         set(h, 'PaperUnits','centimeters');
177         set(h, 'Units','centimeters');
178         pos=get(h,'Position');
179         set(h, 'PaperSize', [pos(3) pos(4)]);
180         set(h, 'PaperPositionMode', 'manual');
181         set(h, 'PaperPosition',[0 0 pos(3) pos(4)]);
182     end
```

Fonte: progetto Matlab

Questa funzione mi permette di salvare automaticamente in pdf i grafici del programma al termine dell'esecuzione (ovvero le 17.30, ora in cui chiude la borsa italiana).

Nella figura 8 viene descritta una funzione che separa le migliaia con una virgola, per facilitare la lettura dei volumi nel grafico.

Figura 7. Funzione che separa le migliaia con una virgola

```
184 function commaFormattedString = CommaFormat(value)
185 %=====
186 % Takes a number and inserts commas for the thousands separators.
187 % Split into integer part and fractional part.
188 [integerPart, decimalPart]=strtok(num2str(value),'.');
189 % Reverse the integer-part string.
190 integerPart=integerPart(end:-1:1);
191 % Insert commas every third entry.
192 integerPart=[sscanf(integerPart,'%c',[3,inf])' ...
193             repmat(',',ceil(length(integerPart)/3),1)'];
194 integerPart=integerPart(:)';
195 % Strip off any trailing commas.
196 integerPart=deblank(integerPart(1:(end-1)));
197 % Piece the integer part and fractional part back together again.
198 commaFormattedString = [integerPart(end:-1:1) decimalPart];
199 end
```

Fonte: progetto Matlab

Nelle figure 8 e 8.1 vengono illustrate integralmente le funzioni ausiliare del programma, hanno tutte in comune la funzione “extractBetween” che permette appunto di estrarre dei valori numerici dalla pagina html di un sito web, nel nostro caso “Investing.com”.

Figura 8. Funzioni ausiliarie

```
131 %%% Funzioni ausiliarie
132 function [quot, percent, vol, vol_medio, azioni] = getInvestParamWeb(webpageName)
133 try
134     %---- QUOTAZIONE ----
135
136     pagina = webread(webpageName);
137     qC = extractBetween(pagina, 'instrument-price-last">', '<'); qC = strrep(qC, ',', '.');
138     quot = str2double(qC); % Quotazione attuale in EUR
139
140     %---- VARIAZIONE PERCENTUALE ----
141     signPerc = extractBetween(pagina, 'change-percent"><!-- -->', '<!-- -->'); signPerc = signPerc(1);
142
143     if nnz(signPerc == '+') == 1
144         signPerc = extractBetween(pagina, 'change-percent"><!-- -->', '<!-- -->'); signPerc = signPerc(1);
145         varPerc = extractBetween(pagina, ['change-percent"><!-- -->' signPerc '<!-- -->'], '<!-- -->', '<!-- -->'); varPerc = strrep(varPerc, ',', '.');
146         percent = str2double([signPerc varPerc(1)]); % scostamento percentuale azioni
147     else
148         varPerc = extractBetween(pagina, 'change-percent"><!-- -->', '<!-- -->');
149         varPerc = strrep(varPerc, ',', '.');
150         percent = str2double(varPerc); % scostamento percentuale azioni
151     end
152
153     %---- VOLUME ----
154     qC = extractBetween(pagina, 'volume-value">', '<'); qC = strrep(qC, ',', '.');
155     vol = str2double(qC); % Volume azioni scambiate
156
157     %---- VOLUME MEDIO ----
158     qC = extractBetween(pagina, '"avgVolume" class="font-bold"><span class="key-info_dd-numeric__2cYjc"><span>', '<'); qC = strrep(qC, ',', '.');
159     vol_medio = str2double(qC); % Volume medio azioni
```

Fonte: progetto Matlab

Figura 8.1.

```
161 %---- AZIONI IN CIRCOLAZIONE ----
162 qC = extractBetween(pagina, '"sharesOutstanding" class="font-bold"><span class="key-info_dd-numeric__2cYjc"><span>', '<'); qC = strrep(qC, ',', '.');
163 azioni = str2double(qC); % Azioni in circolazione
164
165 catch
166     warning('Errore temporaneo di lettura dal sito investing.com!')
167     quot = nan;
168     percent = nan;
169     vol = nan;
170     vol_medio = nan;
171     azioni = nan;
172 end
173 end
```

Fonte: progetto Matlab

Infine, in figura 9 abbiamo la funzione che, dato un particolare ticker (la sigla identificativa per ogni titolo in borsa), elabora e restituisce l'indirizzo pagina associato a quel titolo.

Figura 9. Funzione “getInvestingPageAddr”

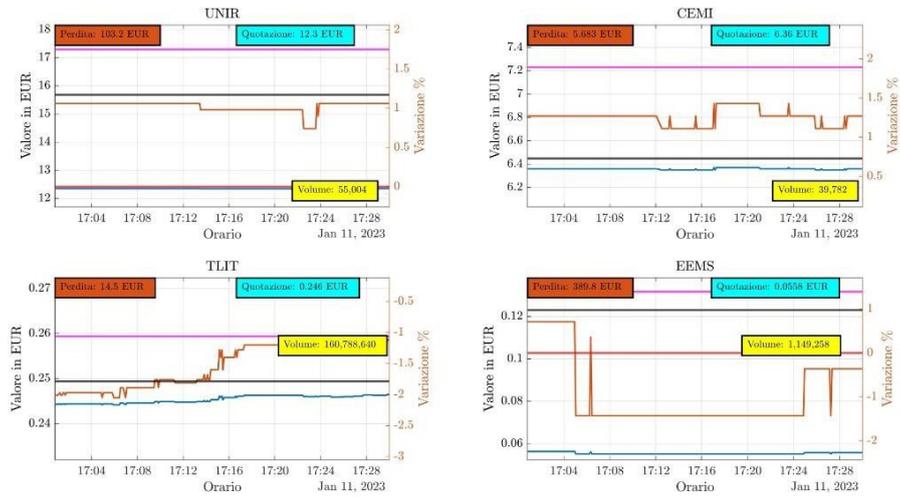
```
203 function webPageAdd = getInvestingPageAddr(ticker)
204
205 website = 'https://it.investing.com/';
206 querySearchTicker = [website 'search/?q=' ticker];
207
208 pagina = webread(querySearchTicker);
209
210 tickerResults = extractBetween(pagina, '<a class="js-inner-all-results-quote-item row" href="/', '>');
211
212
213 webPageAdd = [website tickerResults{1}];
214
215 end
```

Fonte: progetto Matlab

2.2 I risultati ottenuti con relative analisi

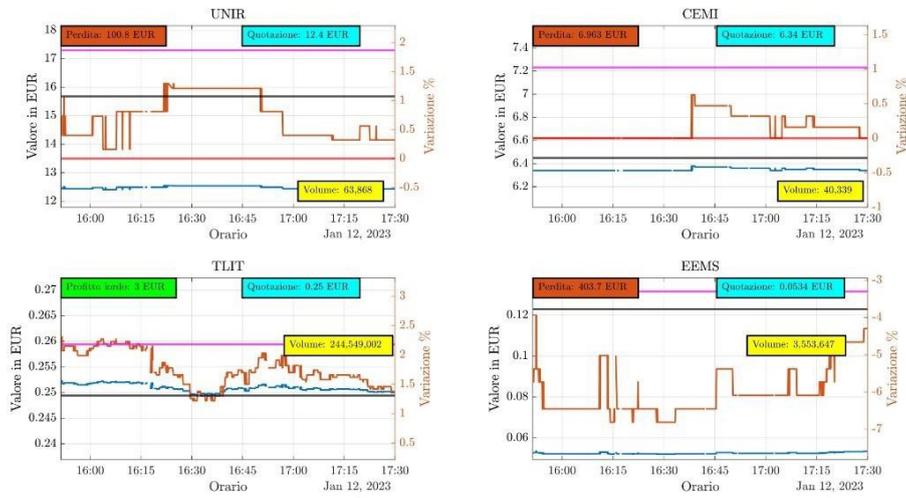
Ecco alcuni dei risultati ottenuti in diverse giornate di analisi.

Figura 10.



Fonte: progetto Matlab

Figura 11.



Fonte: progetto Matlab

Come si può notare all'interno del grafico sono presenti alcuni box testo contenenti alcune informazioni in tempo reale quali la quotazione, il volume delle azioni scambiate e il profitto dell'utente per ogni titolo.

Come accennato precedentemente la linea nera orizzontale rappresenta il prezzo medio di carico (in euro) ovvero "la media dei prezzi unitari di acquisto dei titoli di un portafoglio, comprensivi di commissioni e al netto di oneri, bolli e spese. Rappresenta la base su cui si calcolano utili e perdite."

La linea magenta orizzontale sarebbe l'obiettivo da raggiungere in termini di valore della singola azione(da leggere sull'asse verticale di sx) ; la linea rossa

orizzontale corrisponde allo zero della variazione percentuale (da leggere sull'asse verticale di dx).

Per quanto riguarda le curve del grafico , rispettivamente la rossa rappresenta la variazione percentuale rispetto al valore del titolo all'apertura della borsa in quel giorno; mentre quella blu indica la quotazione in euro del titolo.

CAPITOLO 3. CONCLUSIONI

3.1 Considerazioni personali e spunti per il miglioramento

In conclusione, il presente progetto ha riguardato lo sviluppo di un programma in grado di lavorare con grafici relativi agli andamenti azionari.

Con l'aggiunta di volumi differenziali e la percentuale di incremento o decremento rispetto alla chiusura della seduta precedente, gli investitori avrebbero maggiori informazioni sui titoli da essi monitorati.

Inoltre, con la possibilità di aggiornare il database utente durante l'esecuzione giornaliera, acquistando azioni e di conseguenza modificando pmc e il numero di azioni possedute, i grafici si aggiornerebbero senza dover interrompere l'esecuzione del codice e poi farlo ripartire.

Attraverso l'utilizzo di metodi di analisi dei dati e di algoritmi di machine learning, il programma potenzialmente sarebbe in grado di fornire informazioni utili per prevedere le tendenze future del mercato e supportare maggiormente le decisioni degli investitori tramite analisi grafiche, quantitative e qualitative.

La realizzazione di questo progetto attraverso il machine learning potrebbe rappresentare un passo significativo verso l'utilizzo sempre più diffuso dell'intelligenza artificiale nell'ambito degli investimenti e del trading azionario.

Siamo certi che la tecnologia continuerà a evolversi e a offrire nuove opportunità per migliorare l'efficienza e la precisione delle decisioni degli investitori.

SITOGRAFIA

<https://it.wikipedia.org/wiki/MATLAB>

<https://www.forbes.com/advisor/it/investire/come-comprare-azioni/>

<https://www.bancobpm.it/magazine/glossario/prezzo-medio-di-carico/>