6.4 Il problema di ottimizzazione del portafoglio

Seguendo l'approccio di Markowitz, esaminiamo tre problemi di ottimizzazione del portafoglio. I dati necessari sono:

1. N titoli rischiosi;
2. la media dei rendimenti $m = (m^1, m^2, ..., m^n) \in \mathbb{R}^{n \times 1}$;
3. la matrice di varianza-covarianza $V \in \mathbb{R}^{n \times n}$.

La soluzione del problema è data da un vettore $z = (z^1, z^2, ..., z^n) \in \mathbb{R}^n$ che mostra come distribuire il capitale unitario 1 per comprare azioni degli n titoli rischiosi. Il rendimento del portafoglio è dato da:

$$m^1 \cdot z^1 + m^2 \cdot z^2 + ... + m^n \cdot z^n = (m, z). \tag{6.8}$$

Il rischio è calcolato attraverso la matrice varianza covarianza: (z, Vz) ed indica lo scostamento dalla media.

6.5 Primo problema della selezione del portafoglio

Supponiamo di effettuare l'investimento soltanto se il profitto supera il profitto minimo $\overline{\pi}$. Consideriamo il capitale unitario. Questa ipotesi è valida soltanto supponendo che l'investimento ottimale non dipende dal capitale effettivamente investito, quindi possiamo concentrarci sulle percentuali del capitale unitario. Il primo problema di ottimizzazione richiede di trovare la migliore distribuzione del nostro capitale unitario in asset diversi $z = (z^1, z^2, ..., z^n) \in \mathbb{R}^n$ in modo tale da minimizzare il rischio, mentre il rendimento del portafoglio deve rimanere maggiore od uguale a $\overline{\pi}$. La prima osservazione è che z contiene percentuali del capitale unitario, pertanto la somma delle componenti di z deve essere uguale ad uno. Introducendo il vettore $e = (1, 1, ..., 1) \in \mathbb{R}^n$, possiamo utilizzare la notazione per il prodotto scalare per rappresentare la somma delle componenti del vettore z nel seguente modo:

$$z^1 + z^2 + ... + z^n = 1 \cdot z^1 + 1 \cdot z^2 + ... + 1 \cdot z^n = (e, z). \tag{6.9}$$

Pertanto la richiesta di utilizzare tutto il capitale unitario può essere scritta come $(e, z) = 1$. La richiesta che ogni componente di z sia maggiore od uguale di zero viene indicata da $z \geq 0$, quindi il vettore 0 è il lower bound. Questa condizione implica che non possono essere effettuate vendite allo scoperto. La richiesta che il rendimento totale del portafoglio sia maggiore di $\overline{\pi}$ si scrive nel seguente modo:

$$(m, z) \geq \overline{\pi}. \tag{6.10}$$

Infine, il problema viene scritto, nel suo complesso, nel seguente modo:

$$\begin{array}{l} min_z(z, Vz) \\ tale\ che \\ (m, z) \geq \overline{\pi} \\ (e, z) = 1 \\ z \geq 0 \end{array} \quad . \qquad (6.11)$$

Esaminiamo ora il problema per scrivere le istruzioni di MATLAB necessarie per la sua risoluzione. Innanzitutto, si tratta di un problema di ottimizzazione vincolata. In secondo luogo, si può osservare che tutti i vincoli sono lineari. Esaminiamo ora il vincolo $(m, z) \geq \overline{\pi}$. Si tratta di una disuguaglianza lineare, che può essere scritta come $-(m, z) \leq -\overline{\pi}$. Date le variabili di MATLAB m e z, il prodotto scalare è dato da $m' * z$. Ricordando che in caso di vincoli di disuguaglianza $Az \leq B$ si ha che $A = m'$ e $B = p$.
Esaminiamo ora $(e, z) = 1$. Si tratta di un vincolo di uguaglianza lineare, quindi rientra nella forma $Aeqz \leq Beq$, con $Aeq = e'$ e $Beq = [1]$.
Il lower bound $z \geq 0$ può essere rappresentato ponendo $LB = [0; 0; ...; 0]$. La lunghezza di LB dipende dai dati del problema. Supponiamo di assegnare come punto iniziale $z_0 = (\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n})$. Lo schema generale per la risoluzione del problema della selezione del portafoglio diventa (Listato 6.12, 6.13):

Listato 6.12. Listato del file *portafoglio1.m*.

```
function portafoglio1
m= QUI VANNO INSERITI I VALORI DI m;
z0=(1/length(m))*ones(length(m),1);
A=-m';
p= QUI VA INSERITO IL VALORE DI p
B=[-p];
Aeq= ones(1,length(m));
Beq=[1];
LB=zeros(length(m),1);
UB=[ ];
options=optimset('LargeScale','off');
[x,fval,exitflag,output]=fmincon(@fo,z0,A,B,Aeq,Beq,LB,UB,[ ],
options)
```

Listato 6.13. Listato del file *fo.m*.

```
function f=fo(z)
V= QUI VA INSERITA LA MATRICE DI VARIANZA-COVARIANZA
f=z' * (V*z);
```

Esempio 6.13 Date le serie temporali dei titoli Apulia, Az, Azimut, Aureo, dal 29 marzo 1999 fino al 24 settembre 2001, campionate settimanalmente, (la lunghezza totale di ciascuna serie è uguale a 134) si ha che la matrice varianza-covarianza è

$$\begin{array}{lll} 0.000704 & 0.000848 & 0.000707 \\ 0.000848 & 0.00108 & 0.000856 \\ 0.000707 & 0.000856 & 0.000728 \end{array} \qquad (6.12)$$

ed i rendimenti sono: $m = (0.001324, -0.00015, 0.001797)^T$. Risolvere il problema della ottimizzazione del portafoglio.

Listato 6.14. Listato del file *portafoglio1.m*.

```
function portafoglio1
m=[0.001326; -0.00015; 0.001797];                    %N componenti
z0=(1/length(m))*ones(length(m),1);
A=-m';
p=0.001;
B=[-p];
Aeq= ones(1,length(m));
Beq=[1];
LB=zeros(length(m),1);
UB=[ ];
options=optimset('LargeScale','off');
[x,fval,exitflag,output]=fmincon(@fo,z0,A,B,Aeq,Beq,LB,UB,[ ],
options)
```

Listato 6.15. Listato del file *fo.m*.

```
function f=fo(z)
V=[0.000704 0.000848 0.000707
   0.000848 0.00108  0.000856
   0.000707 0.000856 0.000707];
f=z' * (V*z);
```

6.6 Secondo problema della selezione del portafoglio

In questo caso, l'obbiettivo è contemporaneamente di massimizzare il profitto e di minimizzare il rischio. Pertanto, la funzione obbiettivo è $f(z) = -(m, z) + \frac{a}{2}(z, Vz)$. La constante a è un coefficiente che esprime la sensibilità del rischio, e dipende dal profilo del rischio. Pertanto, il problema di ottimizzazione diventa:

$$\begin{array}{l} min_z - (m, z) + \frac{a}{2}(z, Vz) \\ tale\ che \\ (e, z) = 1 \\ z \geq 0 \end{array} \qquad . \tag{6.13}$$

Si tratta di un problema di ottimizzazione quadratica con vincoli soltanto lineari. Questo permette di utilizzare anche l'istruzione `quadprog`.

Esempio 6.14 Con riferimento allo stesso dataset dell'Esempio 6.13, le istruzioni di MATLAB per risolvere il problema (6.13) sono le seguenti:

Listato 6.16. Listato del file *portafoglio2.m*.

```
function portafoglio2
z0=(1/length(m))*ones(length(m),1);
A=[ ];
```

```
B=[ ];
Aeq= ones(1,length(m));
Beq=[1];
LB=zeros(length(m),1);
UB=[ ];
options=optimset('LargeScale','off');
[x,fval,exitflag,output]=fmincon(@fo,z0,A,B,Aeq,Beq,LB,UB,[ ],
options)
```

Listato 6.17. Listato del file *fo.m.*

```
function f=fo(z)
m=[0.001326; -0.00015; 0.001797];                %N componenti
V=[0.000704 0.000848 0.000707
0.000848 0.00108  0.000856
0.000707 0.000856 0.000707];
f=-m'*z+z' * (V*z);
```

6.7 Terzo problema della selezione del portafoglio

In questo problema, l'obbiettivo è la massimizzazione del profitto, mentre il rischio deve essere al di sotto di un livello fissato $\overline{V}$. Pertanto, la funzione obbiettivo è $f(z) = -(m, z)$, ed il vincolo è dato da $(z, Vz) \leq Vs$, quindi i vincoli non sono più lineari. Il problema di ottimizzazione diventa:

$$\begin{array}{l} min_z -(m, z) \\ tale\ che \\ (z, Vz) \leq \overline{V} \\ (e, z) = 1 \\ z \geq 0 \end{array} . \tag{6.14}$$

Listato 6.18. Listato del file *portafoglio3.m.*

```
function portafoglio3
n=3;
z0=(1/n)*ones(n,1);
A=[ ];
B=[ ];
Aeq= ones(1,n);
Beq=[1];
LB=zeros(n,1);
UB=[ ];
options=optimset('LargeScale','off');
[x,fval,exitflag,output]=fmincon(@fo,z0,A,B,Aeq,Beq,LB,UB,@nonlcon,
options)
```

Listato 6.19. Listato del file *fo.m.*

```
function f=fo(z)
m=[0.001326; -0.00015; 0.001797];
f=-m' * z;
```

Listato 6.20. Listato del file *nonlcon.m*.

```
function  [C, Ceq]=nonlcon(z)
V=[0.000704 0.000848 0.000707
   0.000848 0.00108  0.000856
   0.000707 0.000856 0.000707];
C=z'*V*z-Vs;
Ceq=[];
```

6.8 Costruzione della frontiera efficiente

La frontiera efficiente riporta la soluzione ottimale per i vari livelli ammissibili di rischio. MATLAB ha un'istruzione specifica per il calcolo della frontiera efficiente (in Figura 6.1).

```
>> ExpReturn = [0.1 0.2 0.15];
>> ExpCovariance = [0.005   -0.010    0.004
                   -0.010    0.040   -0.002
                    0.004   -0.002    0.023];
>> NumPorts = 20;
>> portopt(ExpReturn, ExpCovariance, NumPorts)
```

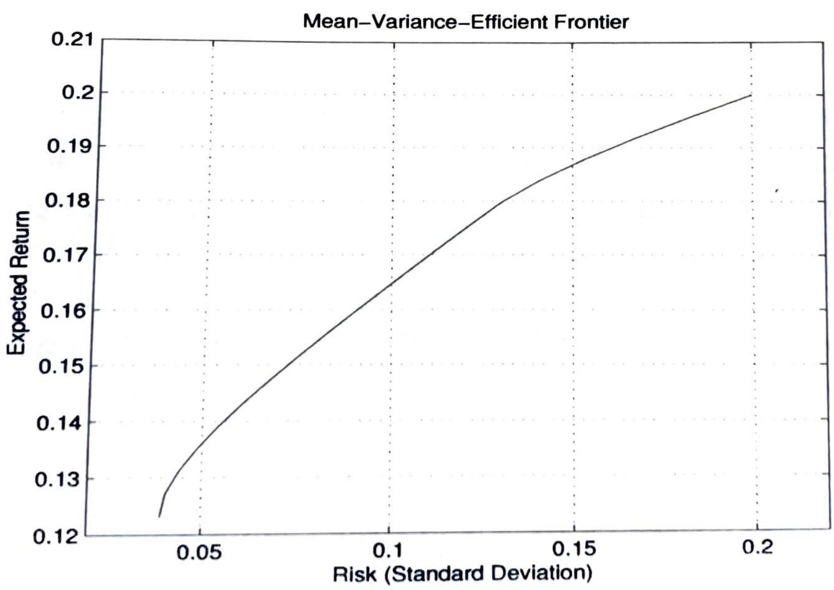

Figura 6.1. Frontiera efficiente.

6.9 Ulteriore applicazione su dati del mercato azionario

Questa sezione mostra come applicare i modelli di selezione del portafoglio su dati dei mercati azionari. Il file esempio.xls scaricabile dal materiale supplementare disponibile riporta su ogni colonna le quotazioni giornaliere dei titoli che compongono l'indice EU50 dallo 01/01/2014 al 31/12/2015 dei titoli che compongono l'indice EU50: ABB LTD N, AIR LIQUIDE, ALLIANZ, ANHEUSER-BUSCH INBEV, ASTRAZENECA, AXA, BARCLAYS, BASF, BAYER, BBV.ARGENTARIA, BANCO SANTANDER, BG GROUP, BNP PARIBAS, BP, BRITISH AMERICAN TOBACCO, BT GROUP, RICHEMONT N, CREDIT SUISSE GROUP N, DAIMLER, DEUTSCHE BANK, DEUTSCHE TELEKOM, DIAGEO, ENI, GLAXOSMITHKLINE, HSBC HDG. (ORD $0.50), IMPERIAL BRANDS, ING GROEP, INTESA SANPAOLO, LLOYDS BANKING GROUP, LVMH, NATIONAL GRID, NESTLE 'R', NOVARTIS 'R', NOVO NORDISK 'B', PRUDENTIAL, RECKITT BENCKISER GROUP, RIO TINTO, ROCHE HOLDING, ROYAL DUTCH, SHELL A, SANOFI, SAP, SCHNEIDER ELECTRIC SE, SIEMENS TELEFONICA, TOTAL, UBS GROUP, UNILEVER CERTS., UNILEVER (UK), VODAFONE GROUP, ZURICH INSURANCE GROUP. Sulle righe sono riportate le date. Occorre innanzitutto caricare il file tramite l'istruzione `A=load('dati.xls')`. In questo modo, `A` è una matrice con 522 righe e 50 colonne: ciascuna colonna contiene le quotazioni di un singolo titolo in date diverse.

Osservazione 6.15 Talvolta l'istruzione non riesce a caricare correttamente il file per via di incompatibilità nel formato. In questo caso, la soluzione più semplice consiste nel copiare i dati, senza etichette sulle colonne e senza le date sulle righe, in un file di puro testo. Se si sta utilizzando il formato di Excel che prevede la virgola come separatore dei decimali, occorre modificarlo nel punto. Questo si può ottenere anche tramite l'istruzione 'sostituisci tutto' nel file di testo già copiato. Una volta preparato il file di testo, dall'interfaccia di MATLAB bisogna eseguire l'istruzione

```
>> A=load('dati.txt')
```

(file a disposizione su internet: dati.txt, dati.xls, Dati010114a311215).

A partire dalle quotazione possiamo calcolare i rendimenti logaritmici nel seguente modo:

```
>> n=size(A,1);
>> R=log(A(2:n,:)./A(1:n-1,:));
```

La matrice `R` avrà quindi 521 righe. Ciascuna colonna contiene i rendimenti di un singolo titolo in date diverse. La media e la matrice varianza-covarianza si ottengono tramite le istruzioni:

```
>> m=mean(R)
>> V=cov(R)
```

In questo modo m è già un vettore riga e non sarà necessario trasporlo per assegnarlo alla matrice dei vincoli lineari A. A questo punto abbiamo tutti gli elementi per impostare i problemi della selezione del portafoglio.

6.9.1 Applicazione: primo modello

Il valore della variabile p è scelto arbitrariamente uguale a 0.001 e può essere liberamente cambiato.

Listato 6.21. Listato del file *portafoglio1.m.*

```
function portafoglio1
A=load('dati.txt');
n=size(A,1);
R=log(A(2:n,:)./A(1:n-1,:));
m=mean(R);
V=cov(R);
z0=(1/length(m))*ones(length(m),1);
A=-m;
p= 0.001;
B=[-p];
Aeq= ones(1,length(m));
Beq=[1];
LB=zeros(length(m),1);
UB=[ ];
options=optimset('LargeScale','off');
[x,fval,exitflag,output]=fmincon(@fo,z0,A,B,Aeq,Beq,LB,UB,[ ],
options)
```

Listato 6.22. Listato del file *fo.m.*

```
function f=fo(z)
A=load('dati.txt');
n=size(A,1);
R=log(A(2:n,:)./A(1:n-1,:));
V=cov(R);
f=z' * (V*z);
```

Si osserva però che questo modo di calcolare V nella funzione *fo.m* è molto inefficiente, perchè ogni volta si ricaricano i dati. Un modo più rapido dal punto di vista della esecuzione delle istruzioni consiste nel copiare direttamente i valori di V nella funzione obbiettivo, invece di caricarli, così come è stato fatto nell'esempio su tre titoli. In questo modo però il programma risulterà molto lungo da stampare, in quanto andranno ricopiate tutte le 521 righe e 50 colonne. Si ottiene il seguente portafoglio: $x = (0,0,0,0.1176,0.0182,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0.044,0,0,0,0,0.1436,0,0.2292,0,0,0,0,0,0.3409,0,0.0455,0,0,0,0,0,0,0,0,0,0,0.0610,0,0,0)$. Le componenti con valori significativi sono le numero 4,5, 21, 26, 28, 34, 36, che corrispondono ai valori 0.1176 0.0182 0.0440 0.1436 0.2292 0.3409 0.0455. Dalla legenda dei titoli si ha quindi che il minimo del rischio, mantenendo il rendimento maggiore o uguale di 0.001, si ottiene investendo la quota 0.11176 in Anheuser-Busch Inbev, 0.0182 in Astrazeneca, 0.044 in Deutsche Telecom, 0.1436 in Imperal brands, 0.2292 in Intesa San Paolo, 0.3409 in Novo Nordisk B, 0.455 in Reckitt Benckiser Group.

Osservazione 6.16 La somma dei valori riportati in `x` non è esattamente 1, ma 0.9390. I valori residui sono contenuti nei decimali dopo la quarta cifra su tutte le componenti del portafoglio, anche quelle in cui il valore mostrato è 0, in quanto la visualizzazione è stata limitata alle prime 4 cifre dopo il punto decimale.

Osservazione 6.17 I titoli selezionati di fatto corrispondono ai titoli con rendimento maggiore, in quanto il massimo rendimento sui titoli è 0.0013. Fissando un diverso valore di `p` si ottiene un portafoglio diverso, per esempio con `p=` 0.0007 si ottiene il portafoglio: `x=` $(0, 0.0050, 0.0365, 0.0768, 0.0430, 0.0343, 0, 0, 0.0113, 0, 0, 0, 0, 0, 0.0173, 0.0335, 0, 0, 0.0317, 0, 0.0517, 0, 0, 0, 0, 0.0858, 0.0311, 0.1148, 0, 0.0271, 0.0227, 0.0124, 0.0286, 0.1531, 0.0124, 0.0523, 0, 0.0051, 0, 0, 0.0207, 0, 0, 0, 0, 0.0149, 0.0575, 0.0203, 0, 0)$ che presenta una diversificazione maggiore degli investimenti in quanto viene ammesso un inferiore rendimento del portafoglio.

Osservazione 6.18 Dalla stampa di `exitflag` si può vedere che l'algoritmo utilizzato è 'medium-scale: SQP, Quasi-Newton, line-search' e sono state effettuate due iterazioni.

Osservazione 6.19 Il valore della funzione, ovvero il rischio, è 0.00014. Decidere se questo valore è alto o basso dipende dall'investitore.

6.9.2 Applicazione: secondo modello

Listato 6.23. Listato del file *portafoglio2.m.*

```
function z=portafoglio2
Dati=load('dati.txt');
p=size(Dati,2);
z0=(1/p)*ones(p,1);
A=[ ];
B=[ ];
Aeq= ones(1,p);
Beq=[1];
LB=zeros(p,1);
UB=[ ];
options=optimset('LargeScale','off');
[z,fval,exitflag,output]=
fmincon(@fo2,z0,A,B,Aeq,Beq,LB,UB,[ ],options)
```

Listato 6.24. Listato del file *fo2.m.*

```
function f=fo2(z)
Dati=load('dati.txt');
n=size(Dati,1);
```

```
R=log(Dati(2:n,:)./Dati(1:n-1,:));
m=mean(R);
V=cov(R);
f=-m*z+z' * (V*z);
```

La soluzione è z= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1498, 0, 0.2092, 0, 0, 0, 0, 0, 0.6410, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), la funzione obbiettivo è f= −0.001. L'algoritmo impiega 38 iterazioni. Le componenti non nulle sono la 26, 28 e 34, con valori 0.1498, 0.2092, 0.6410 che corrispondono a Inbev, Imperal brands, Intesa San Paolo. Di fatto, con i parametri inseriti, Intesa San Paolo raccoglie quote prima disperse su altri titoli.

6.9.3 Applicazione: terzo modello

Immettendo i nuovi dati, il programma relativo al terzo modello risulta modificato nel seguente modo:

Listato 6.25. Listato del file *portafoglio3.m*.

```
function z=portafoglio3
Dati=load('dati.txt');
m=size(Dati,2);
z0=(1/m)*ones(m,1);
A=[ ];
B=[ ];
Aeq= ones(1,m);
Beq=[1];
LB=zeros(m,1);
UB=[ ];
options=optimset('LargeScale','off');
[z,fval,exitflag,output]=fmincon(@fo3,z0,A,B,Aeq,Beq,LB,UB,
@nonlcon1,options)
```

Listato 6.26. Listato del file *fo3.m*.

```
function f=fo3(z)
Dati=load('dati.txt');
n=size(Dati,1);
R=log(Dati(2:n,:)./Dati(1:n-1,:));
m=mean(R);
V=cov(R);
f=-m* z;
disp(f)
```

Listato 6.27. Listato del file *nonlcon1.m*.

```
function [C, Ceq]=nonlcon1(z)
Dati=load('dati.txt');
n=size(Dati,1);
R=log(Dati(2:n,:)./Dati(1:n-1,:));
m=mean(R);
V=cov(R);
Vs=6.0e-004;
C=z'*V*z-Vs;
C=[];
Ceq=[];
```

Valgono le stesse osservazioni di prima rispetto alla somma delle quote di investimento, all'algoritmo usato e rispetto al copiare direttamente i valori di m e V nel programma piuttosto che richiamare ogni volta i dati. Il risultato è: z== $(0,0,0,0.108,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0.1473,0,$ $0.2771,0,0,0,0,0,0.4464,0,0,0,0,0,0,0,0,0,0,0,0,0.212,0,0,0)$. Il valore della funzione obbiettivo, ovvero del rendimento, è -0.0011. La variabile `output` riporta 28 iterazioni. Il valore massimo di rischio accettato è $6.0e-004$. Le componenti con valori significativi sono le 4 26, 28, 34, 47, che si possono trovare rapidamente con il comando `find(z)`. Le quote corrispondenti sono 0.1080, 0.1473, 0.2771, 0.4464, 0.0212, che corrispondono rispettivamente ai titoli Le componenti non nulle corrispondono ai titoli Anheuser-Busch Inbev, Imperal brands, Intesa San Paolo, UNILEVER CERTS. Questo portafoglio risulta differente dal precedente proprio perchè rischio e rendimento sono trattati in maniera diversa. In questo esempio, anche ponendo la soglia di rischio uguale al risultato del modello precedente, 0.00014, si arriva allo stesso risultato.

6.10 Frontiera efficiente

La frontiera efficiente è mostrata in Figura 6.2.

```
>> Dati=load('dati.txt');
>> n=size(Dati,1);
>> R=log(Dati(2:n,:)./Dati(1:n-1,:));
>> ExpReturn = mean(R);
>> ExpCovariance = cov(R);
>> NumPorts = 20;
>> portopt(ExpReturn, ExpCovariance, NumPorts)
```

6.10.1 Confronto su dati successivi

Testiamo adesso la stabilità del portafoglio nel periodo successivo. Prepariamo quindi i file dati2.xls e dati2.txt ed eseguiamo i programmi già illustrati per il primo, secondo e terzo modello, ma caricando i nuovi file di dati. In tutti e tre i programmi precedenti l'unica istruzione da cambiare è

```
>> Dati=load('dati.txt');
```

che diventa

```
>> Dati=load('dati2.txt');
```

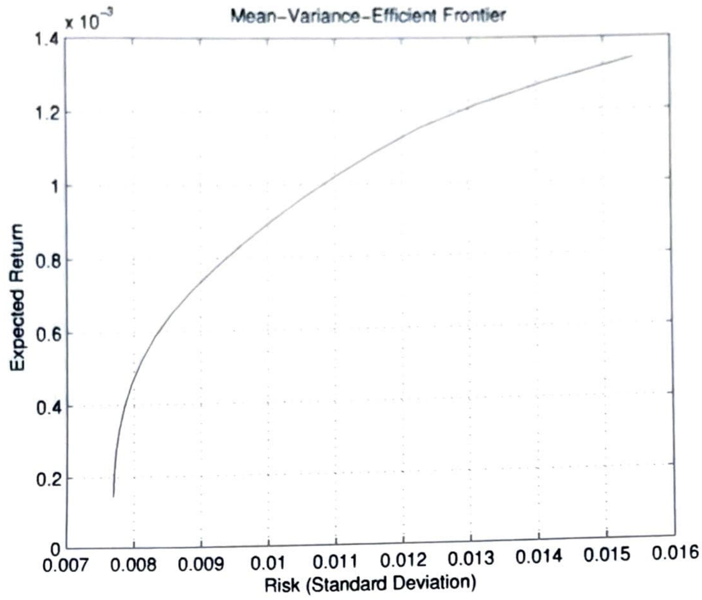

Figura 6.2. Frontiera efficiente, dati dallo 01/01/2014 al 31/12/2015.

Esempio 6.20 Ripetiamo l'applicazione mostrata sui dati dal 01/01/2016 allo 11/02/2016.

Si ottengono i seguenti risultati.
Nel primo modello: il valore della funzione obbiettivo è molto vicina a zero, essendo 3.7340e-004; l'algoritmo termina in 2 iterazioni, le componenti non nulle sono 12, 26, 31, che corrispondono ai titoli BG Group, Imperial Brands e National Grids con quote, rispettivamente, 0.8330, 0.1459, 0.0211. Questo vuol dire che i valori rischio/rendimento di BG Group nei primi due mesi del 2016 sono nettamente migliori rispetto agli altri titoli.
Nel secondo modello: il valore della funzione obbiettivo è -5.4611e-004, l'algoritmo termina in 22 iterazioni, le componenti non nulle sono la 12, 26, 31, che, come prima, che corrispondono ai titoli BG Group, Imperial Brands e National Grids con quote, rispettivamente, 0.7232, 0.1941, 0.0827, leggermente diverse rispetto a quanto ottenuto nel primo modello. Nel terzo modello: l'algoritmo termina in 29 iterazioni, il valore della funzione obbiettivo è -0.0012 e tutta la quota viene indicata per BG Group. Questo risultato per il portafoglio non corrisponde ad una buona diversificazione, per cui è opportuno diminuire il valore di `Vs`, il massimo rischio ammissibile.
Riguardo alla frontiera efficiente, si ha il grafico in Figura 6.3.