Código de Matlab - Financial Portfolio Optimization

```
%Portfolio Optimization using CVX and Barrier Method

%number of variables
n=4;

%variable controling tradeoff betwen return and risk
gama=1;

%generate vector of ones
vec_ones=ones(n,1);

% generate random expected returns of the assets
miu=randn(n,1);

%covariance matrix of the returns of assets in portfolio
temp=rand(n);
cov=temp'*temp;

% solve optimization problem with CVX
tic

cvx_begin

variable w(n);

%maximize cost function
maximize(miu'*w - gama*w'*cov*w);

subject to
(vec_ones')*w == 1; w>=0;

cvx_end;
toc
% time1(i)=toc;

%generate variables for Barrier Method

%generate vector z (initial feasible point)
z=rand(n,1);
z=z/sum(z);
z=z(1:end-1);

%generate vector D
D=eye(n,n);
D(end,:)=-1;
D(:,end)=0;
D=D(1:end,1:end-1);

%generate vector b
b=zeros(n,1);
b(n)=1;

% solve optimization problem using Barrier Method
```

```
%initial t, u and tolerance
t=10;
u=10;
epsb=0.05; %epsilon barrier # epsilon newthon

f0= @(z)(-miu'*(D*z+b)+gama*(D*z+b)'*cov*(D*z+b));

grad_f0= @(z) (-miu'*D+gama*(D*z+b)'*2*cov*D);
grad_phi= @(z) -(z).^-1 + 1/(1-sum(z))*ones(n-1,1);

hess_f0= gama*2*(D'*cov*D);
hess_phi= @(z) diag((z.^2).^-1)+(1/(sum(z)-1)^2)*ones(n-1,n-1);

tic
while(1)

%loop2-newton method for convex functions
c1=10^-6;
beta=0.5;
epsn=0.001;
k=0;
while(k<20)

    %gradiente da função tfo+phi
    grad=t*grad_f0(z)'+grad_phi(z);

    if(norm(grad)<epsn)
        break;
    end

    %hessiana da função tfo+phi
    hess=t*hess_f0+hess_phi(z);

    %descent direction
    d=hess\(-grad);

    alfa=1;

while((t*f0(z+alfa*d)+phi(z+alfa*d,n))>(t*f0(z)+phi(z,n)+alfa*c1*grad'
*d))
    alfa=beta*alfa;
    end

    z=z+alfa*d;
    k=k+1;
end

if((n/t)<epsb)
    break;
end
t=u*t;
end
toc
% time2(i)=toc;

%reconstruct final portfolio z
z(n)=1-sum(z);
```

```
figure(1); clf; % plot solution
subplot(1,3,1); stem(miu,'LineWidth',5);
title('rates of return');
subplot(1,3,2); stem(cov*w,'g','LineWidth',5);
hold on
subplot(1,3,2); stem(cov*z,'r','LineWidth',5);
title('Risk');
subplot(1,3,3); stem(w,'g','LineWidth',5);
title('Portfolio: green-CVX, red-BARRIER');
hold on
subplot(1,3,3); stem(z,'r','LineWidth',5);
```

Função phi.m

```
function [ phi ] = phi(z,n)
phi=0;
 for i=1:n-1
        if(z(i)>=0)
        phi=phi+log(z(i));
        else
            phi=phi-inf;
        end
 end

 if((1-sum(z))>=0)
 phi=-phi-log(1-sum(z));
 else
     phi=inf;
 end

end
```