مثال:

با اســـتفاده از داده‌های واقعی شـــاخص کل بورس اوراق بهادار تهران میانگین بازدهی و انحراف معیار آن را در افق‌های زمانی مختلف مقایسه کنید؟

پاسخ:

در برنامه زیر، میانگین و انحراف معیار بازدهی در افق‌های زمانی مختلف محاسبه شده است.

MATLAB®

```
clc
clear
load DailyTotalIndex
```
→ داره‌ادی صفر.

```
m10=(1:10:1000)';
m20=(1:20:1000)';
m50=(1:50:1000)';
mean1=mean(price2ret(DailyTotalIndex));
mean10=mean(price2ret(DailyTotalIndex(m10)));
mean20=mean(price2ret(DailyTotalIndex(m20)));
mean50=mean(price2ret(DailyTotalIndex(m50)));
std1=std(price2ret(DailyTotalIndex));
std10=std(price2ret(DailyTotalIndex(m10)));
std20=std(price2ret(DailyTotalIndex(m20)));
std50=std(price2ret(DailyTotalIndex(m50)));
results=[mean1  mean10  mean20  mean50;std1  std10
std20 std50];
```

نام فایل: MeanAndStd

توضیحات:

mean1، mean10، mean20 و mean50: به ترتیب میانگین بازدهی یک، ده، بیست و پنجاه روزه.

std1، std10، std20 و std50: به ترتیب انحراف معیار بازدهی یک، ده، بیست و پنجاه روزه.

نکات:

- در این برنامه پس از پاک‌کردن صفحه و Work Space به ترتیب با دو دستور clc و clear، داده‌های مربوط به شاخص کل که در فایل داده DailyTotalIndex ذخیره شده است، بارگذاری شده و بعد از آن، به محاسبه شماره روزهایی که باید در استخراج بازدهی ۱۰، ۲۰ و ۵۰ روز مدنظر قرار گیرند، پرداخته شده است (از دستور m10=(1:10:1000)' تا دستور m50=(1:50:1000)').
- محاسبه میانگین ۱ روزه تا ۵۰ روزه با دستورات ...=mean1 الی ...=mean50 انجام شده است.
- محاسبه انحراف معیار یک روزه تا ۵۰ روزه با دستورات ...=std1 الی ...=std50 انجام شده است.

برنامه ۱. مقایسه بازدهی و انحراف معیار با افق‌های زمانی مختلف.

با اجرای برنامه بالا برای داده‌های روزانه شاخص کل بورس اوراق بهادار که در فایل داده با نام DailyTotalIndex ذخیره شده است به نتایج زیر دست خواهیم یافت:

---	یک روزه (درصد)	۱۰ روزه (درصد)	۲۰ روزه (درصد)	۵۰ روزه (درصد)
میانگین	0.18	1.84	3.61	9.73
انحراف معیار	0.81	4.08	6.24	10.69

```
clc
clear
n1=20;
n2=100;
n3=500;
t=1;
t1=0:1/n1:t;
t2=0:1/n2:t;
t3=0:1/n3:t;
S0=0;
S1=cumsum([S0;((2*(rand(n1,1)>.5)-1)*sqrt(t/n1))]);
S2=cumsum([S0;((2*(rand(n2,1)>.5)-1)*sqrt(t/n2))]);
S3=cumsum([S0;((2*(rand(n3,1)>.5)-1)*sqrt(t/n3))]);
plot(t1,S1,t2,S2,t3,S3,t1,zeros(n1+1,1), 'k')
```

شرط اولیه صفر

شیر ۱+

خط ۱-

نام فایل: RandomWalkLimit

توضیحات:

n1، n2 و n3: تعداد پرتاب سکه در زمان محدود t،

S1، S2 و S3: دارایی تجمعی پس از انجام پرتاب،

نکات:

- عبارت ...=t1 تا ...=t3 برای تقسیم کل زمان به زیربخش‌هایی با طول متفاوت است.
- عبارت rand(n1,1)>.5، یک متغیر تصادفی با توزیع یکنواخت بین صفر و یک را با عدد ۰.۵ مقایسه‌کرده و نتیجه‌ای برابر صفر یا یک خواهد داشت (با توجه به قضیه حد مرکزی، در تعداد دفعات بالا انتظار است در نیمی از موارد این نتیجه صفر و در نیمی دیگر برابر یک باشد). این نتیجه صفر یا یک، با ضرب در عدد دو شدن و کسر عدد یک از آن، به نتیجه یک یا منفی یک تبدیل می‌شود که نشان‌دهنده برد یا باخت است و سپس در مقدار برد یا باخت در هر مرتبه ((sqrt(t/n1 نیز ضرب می‌شود.
- با گرفتن جمع تجمعی (cumsum) از میزان برد یا باخت، روند مقدار دارایی نهایی استخراج می‌شود.

```
function S=BMSim(T,N,dt)
dS=sqrt(dt)*randn(T-1,N);
S=cumsum([zeros(1,N);dS]);
```

نام فایل: BMSim

توضیحات:

S: حرکت براونی،

T: طول هر مسیر،

N: تعداد مسیر،

dt: گام زمانی،

نکات:

- عبارت sqrt(dt)*randn(T-1,N) نشان‌دهنده تغییرات S یعنی dS است.
- با دستور ...S=cumsum یعنی با تجمیع همهٔ تغییرات، مقدار نهایی S استخراج می‌شود.

می‌توان آزمون

طول ماتریس

را صفر بذار بقیه را از ترن بگیر.

>>S = BMSim(100,20,1);

>>

```
function S=GBMSim(T,N,dt,Mu,Sigma,S0)
S=zeros(T,N);
S(1,:)=S0;
for i=2:T
    dS=Mu*dt*S(i-1,:)+Sigma*S(i-
1,:).*randn(1,N)*sqrt(dt);
    S(i,:)=S(i-1,:)+dS;
end
```

نام فایل: GBMSim

توضیحات:

S: قیمت شبیه‌سازی شده سهم،

T: طول هر مسیر،

N: تعداد مسیر،

dt: گام زمانی،

Mu: جمله رانش،

Sigma: نوسانات،

S0: قیمت اولیه.

نکات:

- ابتدا برای جلوگیری از بزرگ‌شدن یک ماتریس در حلقه تکرار، ماتریس اولیه با اندازه موردنظر با دستور S=zeros(T,N) ایجاد شده است. سپس مقادیر سطر اول این ماتریس و در همه ستون‌ها، مقدار S0 قرار داده شده است.
- در حلقه for، تغییرات قیمت با دستور ...=dS ایجادشده و این تغییرات با دستور ...=S(i,:) با مقادیر قبلی S تجمیع شده است.

مثال:

تأثیر جمله رانش و نوســـانات در فرایند قیمت ســـهم در مدل حرکت براونی هندسی را تحلیل کنید.

پاسخ:

برای پاسخ به این سؤال، در اسکریپت زیر برنامه حرکت براونی هندسی با چهار مقدار مختلف رانش و نوســـانات برای مثال ۱۰ مســـیر هر یک به طول ۱۰۰ و گام زمانی ۱ و با قیمت اولیه فرضی ۱۰۰ دلار اجرا کرده و نتایج را ترسیم می‌کنیم:

MATLAB®

```
S1=GBMSim(100,10,1,.01,.05,100);
S2=GBMSim(100,10,1,.01,.02,100);
S3=GBMSim(100,10,1,.001,.002,100);
S4=GBMSim(100,10,1,-.01,.02,100);
subplot(2,2,1)
plot(S1)
subplot(2,2,2)
plot(S2)
```

نام فایل: MuSigmaGBM

توضیحات:

S: قیمت شبیه‌سازی شده سهم،

T: طول هر مسیر،

N: تعداد مسیر،

dt: گام زمانی،

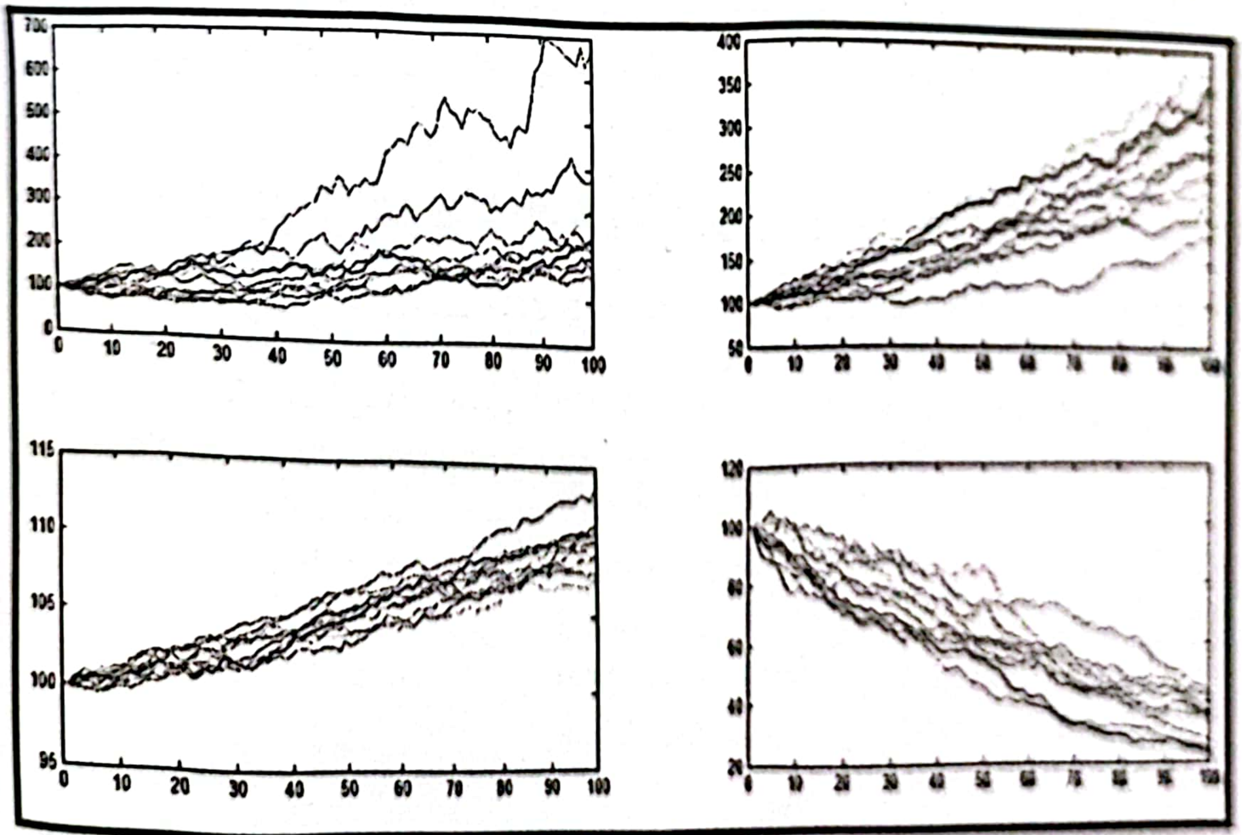
Mu: جمله رانش،

Sigma: نوسانات،

S0: قیمت اولیه.

برنامه ۵. شبیه‌سازی حرکت براونی هندسی با مقادیر مختلف پارامتر رانش و نوسانات.

به این ترتیب چهار نمودار زیر ترسیم می‌شود که به ترتیب دو نمودار بالایی از چپ به راست برای $\mu = 0.01$ و $\sigma = 0.05$، سپس $\mu = 0.01$ و $\sigma = 0.02$ بوده و به همین صورت در دو نمودار پایینی مربوط به حالات $\mu = 0.001$ و $\sigma = 0.002$ و سپس $\mu = -0.01$ و $\sigma = 0.02$ می‌باشد.



شکل ۵. شبیه‌سازی حرکت براونی هندسی با مقادیر مختلف پارامتر رانش و نوسانات.

```
function [RetVaR SSVaR SCVaR r]=ParVaR(S,p,T)
r=zeros(k-T,1);
for i=1:k-T
    r(i,1)=log(S(i+T,1)/S(i,1));
end
Mu=mean(r);
Sigma=std(r);
RetVaR=norminv(p,Mu,Sigma);
S0=S(end,1);
SSVaR=-S0*(RetVaR);
SCVaR=-S0*(exp(RetVaR)-1);
```

نام فایل: ParVar

توضیحات:

RetVaR: ارزش در معرض خطر (درصدی)،

SSVaR: ارزش در معرض خطر (مقداری) به روش ساده،

SCVaR: ارزش در معرض خطر (مقداری) به روش مرکب پیوسته،

r: بازدهی،

S: قیمت،

T: دوره،

p: احتمال مورد نظر،

Mu: میانگین بازدهی،

Sigma: انحراف معیار بازدهی.

نکات:

- در حلقه for، بازدهی لگاریتمی T روزه محاسبه شده است.
- دو دستور ...=Mu و ...=Sigma مربوط به محاسبه میانگین و انحراف معیار بازدهی است تا جایگزین میانگین و انحراف معیار توزیع نرمال ایجادشده در گام بعد قرار گیرد.
- با دستور ...=RetVaR، با فرض توزیع نرمال و با پارامترهای محاسبه‌شده در گام قبل، مقدار عددی که 1-p درصد داده‌ها از آن بیشتر است با استفاده از تابع معکوس نرمال (norminv) استخراج شده و براساس آن مقدار ارزش در معرض خطر به دو روش ساده و پیوسته محاسبه شده است.

```
function [RetVaR SSVaR SCVaR]=HistVaR(S,p,T)
k=size(S,1);
r=zeros(k-T,1);
for i=1:k-T
    r(i,1)=log(S(i+T,1)/S(i,1));
end
RetVaR=quantile(r,p);
S0=S(end,1);
SSVaR=-S0*(RetVaR);
SCVaR=-S0*(exp(RetVaR)-1);
```

نام فایل: HistVar

توضیحات:

RetVaR: ارزش در معرض خطر (درصدی)،

SSVaR: ارزش در معرض خطر (مقداری) به روش ساده،

SCVaR: ارزش در معرض خطر (مقداری) به روش مرکب پیوسته،

r: بازدهی،

S: قیمت،

T: دوره،

p: احتمال موردنظر.

نکات:

- این برنامه تقریباً شبیه برنامه قبلی برای محاسبه ارزش در معرض خطر به روش پارامتریک است و تنها تفاوت آن در استفاده از دستور quantile برای محاسبه چندک به‌عنوان مقدار ارزش در معرض خطر به روش تاریخی است.

برنامه ۷. محاسبه ارزش در معرض خطر به روش تاریخی.

مثال:

با استفاده از داده‌های روزانه شاخص کل بورس اوراق بهادار تهران، ارزش در معرض خطر روزانه با سطح خطای یک درصد را به دو روش پارامتریک و تاریخی محاسبه کنید. در صورت عدم برابری پاسخ این دو روش، علت آن را تحلیل کنید.

پاسخ:

برای پاسخ به این سؤال از داده‌های روزانه شاخص استفاده‌کرده و مقادیر ارزش در معرض خطر روزلنه با سطح خطای یک درصد را به دو روش پارامتریک و تاریخی محاسبه می‌کنیم (استفاده از نماد ~ به معنی عدم نیاز به این خروجی است):

```
>> load DailyTotalIndex
>> S= DailyTotalIndex;p=0.01;T=1;
>> [RetVaR SSVaR SCVaR ~]=ParVaR(S,p,T)
RetVaR =
   -0.0170
sSVaR =
  1.3172e+003
sCVaR =
  1.3061e+003
>> [RetVaR SSVaR SCVaR ~]=HistVaR(S,p,T)
RetVaR =
   -0.0188
sSVaR =
  1.4599e+003
sCVaR =
  1.4463e+003
```

همانطور که مشاهده می‌شود مقدار ارزش در معرض بازدهی خطر پارامتریک برابر ۱/۷- درصد می‌باشد. درنتیجه براساس روش پارامتریک بیشترین درصد کاهش شاخص در یک روز و با احتمال ۹۹ درصد، برابر ۱/۷- درصد خواهد بود. این رقم براساس روش تاریخی برابر ۱/۸۸- درصد است. برای بررسی علت اختلاف این دو رقم در برنامه زیر به مقایسه توزیع بازدهی تاریخی شاخص و توزیع نرمال از طریق نمودار توزیع فراوانی و آزمون جارگ-برا (از طریق دستور jbtest) پرداخته شده است

MATLAB®

```
clear
clc
load DailyTotalIndex
r=price2ret(DailyTotalIndex);
k=size(r,1);
Mu=mean(r);
Sigma=std(r);
```

```
[N1,X]=hist(r,50);
N2=normpdf(X,Mu,Sigma);
bar(X,N1,'y')
hold on
plot(X,N2,'LineWidth',3)
[H,p]=jbtest(r);
```

نام فایل: CompareDist

توضیحات:

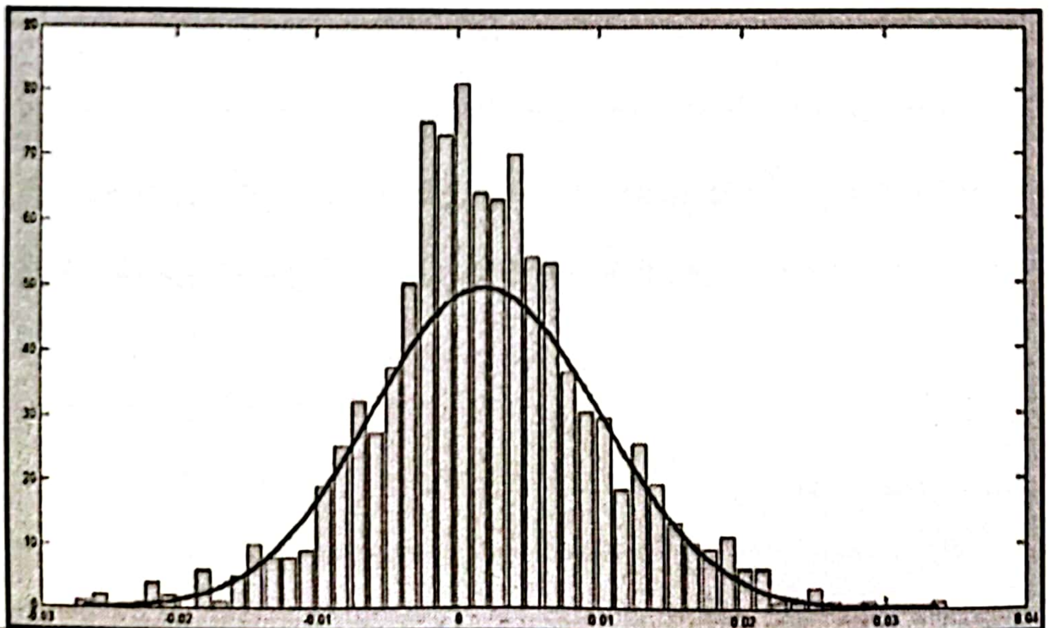
r: بازدهی روزانه،
Mu: میانگین بازدهی،
Sigma: انحراف معیار بازدهی،
N1: فراوانی بازدهی در هر بازه،
N2: مقدار تابع چگالی احتمال توزیع نرمال،
H: نتایج آزمون فرض توزیع جارگ- برا،
p: مقدار آماره p آزمون جارگ- برا.

نکات:

- با استفاده از دستور [N1,X]=hist(r,50)، با تقسیم مقادیر بازدهی به ۵۰ بازه (خروجی گرفته‌شده در متغیر X)، فراوانی در هر بازه استخراج شده است (خروجی گرفته‌شده در متغیر N1).
- دستور N2=normpdf(X,Mu,Sigma) این امکان را می‌دهد که مقدار تابع چگالی توزیع نرمال را با میانگین و انحراف معیار دلخواه استخراج کنیم.
- دستور bar(X,N1,'y') مقادیر فراوانی واقعی را به شکل ستونی به رنگ زرد (با توجه به ورودی y که نشانگر رنگ زرد است) ترسیم کرده و دستور plot(X,N2,'LineWidth',3) نیز مقادیر تابع چگالی نرمال را به شکل خطی با ضخامت برابر ۳ واحد ترسیم می‌کند. همچنین برای قرارگیری دو نمودار بر روی یکدیگر، از hold on استفاده شده است.
- آزمون نرمال‌بودن یا نبودن جارگ-برا با استفاده از دستور [H,p]=jbtest(r) استفاده شده است.

برنامه ۸. مقایسه توزیع بازدهی واقعی با توزیع نرمال.

با اجرای این برنامه، نمودار زیر ترسیم خواهد شد:



شکل ۶. مقایسه توزیع بازدهی تاریخی شاخص و توزیع نرمال.

در این شکل نمودار میله‌ای نشان‌دهنده توزیع بازدهی واقعی بوده و منحنی خطی بیانگر توزیع نرمال است. همانطور که مشاهده می‌شود در قسمت‌های مرکزی توزیع، شاهد کشیدگی بیشتر بازدهی واقعی نسبت به بازدهی نرمال بوده و همچنین در دنباله‌های آن فراوانی بازدهی واقعی نسبت به توزیع نرمال بیشتر است (پدیده پهن‌دنبالگی).[1] بررسی دقیق‌تر این موضوع از طریق آزمون جارک- برا امکان‌پذیر است. در این آزمون، چولگی و کشیدگی توزیع بازدهی واقعی با مقادیر متناظر آن در توزیع نرمال مقایسه می‌شود:

```
>> [H p]
ans =
     1.0000     0.0010
```

خروجی اول (H) نشان‌دهنده عدم رد فرض یک دال بر عدم توزیع نرمال در سطح خطای پیش‌فرض (۵ درصد) است؛ بدین‌روی، بر اساس این آزمون بازدهی واقعی روزانه شاخص توزیع نرمال ندارد. این نتیجه با مقایسه مقدار آماره p (برابر ۰/۰۰۱) با سطح ۵ درصد نیز به‌دست می‌آید.

بنابراین دلیل اصلی تفاوت مقادیر ارزش در معرض خطر دو روش، عدم توزیع نرمال بازدهی (فرض اصلی روش پارامتریک مورد استفاده) است.

```
function [RetVaR SSVaR SCVaR Simr]=MCSimVaR(S,p,T)
r=price2ret(S);
Mu=mean(r);
Sigma=std(r);
S0=S(end,1);
SimS=GBMSim(T+1,10000,1,Mu,Sigma,S0);
SimST=SimS(end,:)';
Simr=log(SimST/S0);
RetVaR=quantile(Simr,p);
SSVaR=-S0*(RetVaR);
SCVaR=-S0*(exp(RetVaR)-1);
```

نام فایل: MCSimVar

توضیحات:

RetVaR: ارزش در معرض خطر (درصدی)،

SSVaR: ارزش در معرض خطر (مقداری) به روش ساده،

SCVaR: ارزش در معرض خطر (مقداری) به روش مرکب پیوسته،

Simr: بازدهی شبیه‌سازی شده،

S: قیمت،

T: دوره،

p: احتمال مورد نظر.

نکات:

- ابتدا با محاسبه میانگین و انحراف معیار بازدهی برای ورود به تابع شبیه‌سازی حرکت براونی هندسی، قیمت با دستور ...SimS=GBMSim براساس این فرایند شبیه‌سازی می‌شود.
- با دستور SimST=SimS(end,:)' آخرین قیمت‌های شبیه‌سازی شده استخراج‌شده و با دستور ...=Simr بازدهی لگاریتمی محاسبه می‌شود.
- مقدار ارزش در معرض خطر با دستور RetVaR=quantile(Simr,p) و دو دستور بعدی محاسبه می‌شود.

برنامه ۹. محاسبه ارزش در معرض خطر به روش شبیه‌سازی مونت‌کارلو براساس حرکت براونی هندسی.

با استفاده از داده‌های روزانه شاخص مقادیر ارزش در معرض خطر درصدی و قیمتی را با افق زمانی یک روزه، با سطح خطای یک‌درصد را به روش شبیه‌سازی مونت‌کارلو محاسبه می‌کنیم:

```
>> load DailyTotalIndex;
>> S=DailyTotalIndex;p=0.01;T=1;
>> [RetVaR SSVaR SCVaR ~]=MCSimVaR(S,p,1)
RetVaR =
    -0.0167
SSVaR =
   1.2989e+003
SCVaR =
   1.2881e+003
```

```
function [ParCVaR HistCVaR MCSimCVaR]=CVaR(S,p,T)
[~ , ~, ~, Parr]=ParVaR(S,p,T);
[HistRetVaR , ~, Histr]=HistVaR(S,p,T);
[MCSimRetVaR , ~, ~,MCSimr]=MCSimVaR(S,p,T);
ParCVaR=-(mean(Parr)+std(Parr)*normpdf(norminv(p))/p);
HistCVaR=mean(Histr(Histr<HistRetVaR));
```

نام فایل: Cvar

توضیحات:

ParCVaR: ارزش در معرض خطر شرطی (درصدی) به روش پارامتریک،

HistCVaR: ارزش در معرض خطر شرطی (درصدی) به روش تاریخی،

MCSimCVaR: ارزش در معرض خطر شرطی (درصدی) به روش شبیه‌سازی،

S: قیمت،

T: دوره،

p: احتمال مورد نظر،

نکات:

- در ابتدا با استفاده از سه تابع قبلی، مقادیر ارزش در معرض خطر به سه روش محاسبه شده است.
- ارزش در معرض خطر شرطی به روش پارامتریک براساس موارد تشریح شده در متن با دستور ParCVaR=... محاسبه شده است.
- برای محاسبه ارزش در معرض خطر شرطی به دو روش دیگر نیز، میانگین مواردی که بازدهی کمتر از ارزش در معرض خطر به روش مشابه است، محاسبه شده است.

برنامه ۱۰. محاسبه ارزش در معرض خطر شرطی.

دوباره با استفاده از داده‌های روزانه شاخص، مقادیر ارزش در معرض خطر شرطی درصدی با افق زمانی یک روزه و با سطح خطای یک درصد را محاسبه می‌کنیم:

```
>> load DailyTotalIndex;
>> S=DailyTotalIndex;p=0.01;T=1;
>> [ParCVaR HistCVaR MCSimCVaR]=CVaR(S,p,T)
ParCVaR =
   -0.0233
HistCVaR =
   -0.0227
MCSimCVaR =
   -0.0227
```

برای بررسی بیشتر موضوع مقادیر محاسبه‌شده ارزش در معرض خطر (درصدی) و خطر شرطی (درصدی) را در جدول زیر دوباره ارائه می‌کنیم:

روش محاسبه	ارزش در معرض خطر	ارزش در معرض خطرشرطی
پارامتریک	-0.0170	-0.0233
تاریخی	-0.0188	-0.0227
شبیه‌سازی	-0.0167	-0.0227

```
function
[S,Volat]=NGARCHSim(T,N,dt,Mu,Omega,Alpha,Beta,Gamma,S
0)
S=ones(T,N)*S0;
Volat0=Omega/(1 -Alpha-Beta);
Volat=ones(T,N)*Volat0;
for t=2:T
    S(t,:)=S(t-1,:)+Mu*S(t-1,:)*dt+sqrt(Volat(t-
1,:)).*S(t-1,:).*randn(1,N)*sqrt(dt);
    Volat(t,:)=Omega+Alpha.*Volat(t-1,:)+...
        Beta.*(sqrt(Volat(t-1,:)*dt).*randn(1,N)-
Gamma.*sqrt(Volat(t-1,:))).^2;
end
```

نام فایل: NGARCHSim

توضیحات:

S: قیمت شبیه‌سازی شده،

Volat: نوسانات (σ_t^2) شبیه‌سازی شده،

T: طول هر مسیر،

N: تعداد مسیر،

dt: گام زمانی،

Mu: جمله رانش (μ)،

Omega: پارامتر ω،

Alpha: پارامتر α،

Beta: پارامتر β،

Gamma: پارامتر γ،

S0: قیمت اولیه.

نکات:

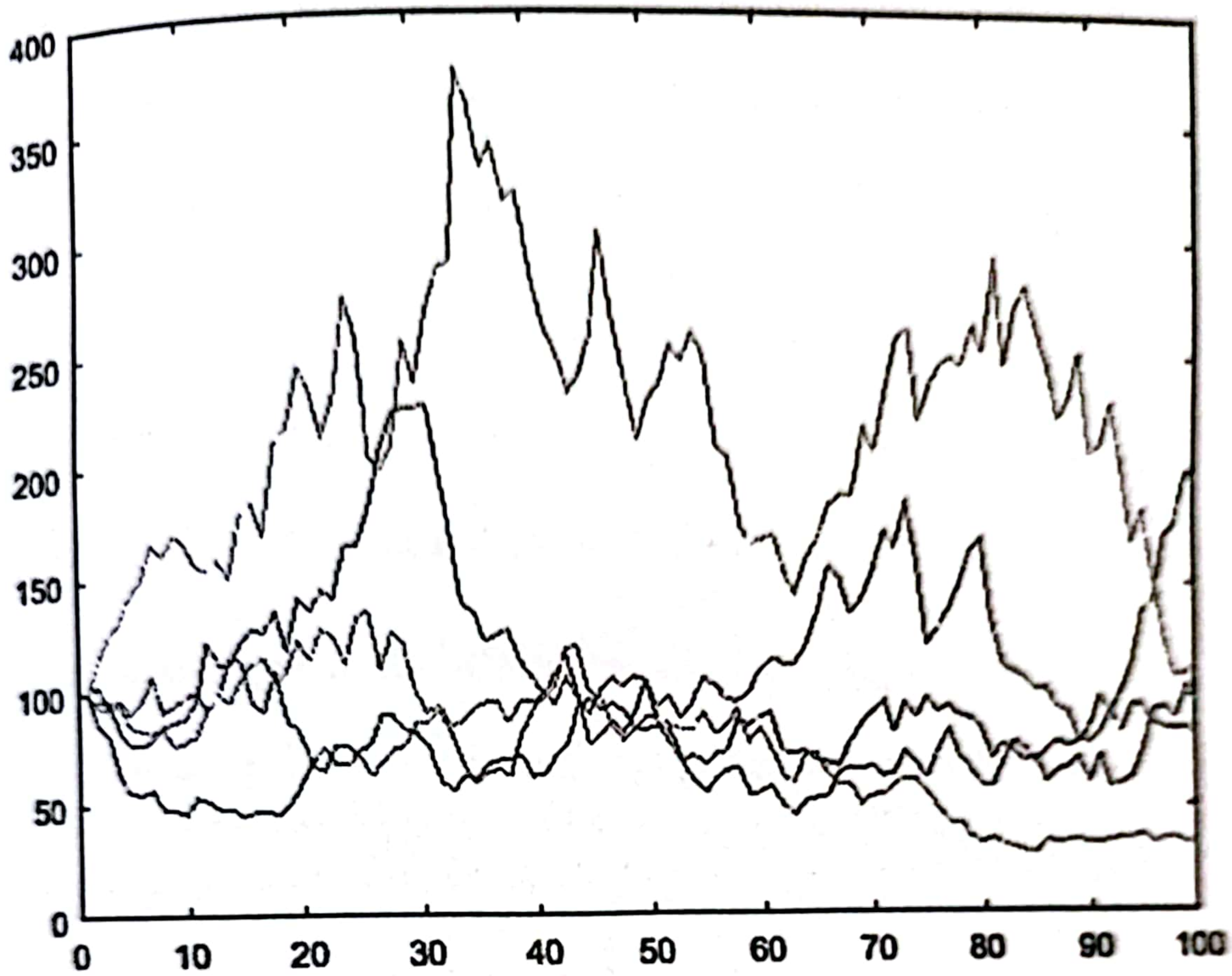
- مقدار اولیه نوسانات از رابطه ...=Volat0 به‌دست آمده و این مقدار در نخستین دور از حلقه تکرار استفاده شده است.
- در حلقه تکرار، در هر دفعه هم مقدار S (با دستور ...=(S(t,:) و هم مقدار نوسانات (با دستور ...=(Volat(t,:) محاسبه شده و در تکرار دو بعد، استفاده می‌شود.

برنامه ۱۱. شبیه‌سازی مدل گارچ غیرخطی.

در برنامه بالا برای استخراج مقدار اولیه نوسانات از مقدار $\omega/(1-\alpha-\beta)$ استفاده شده است. درصورت استفاده برنامه بالا برای ایجاد ۵ مسیر به طول ۱۰۰ به‌صورت زیر:

```
>>
[S,Volat]=NGARCHSim(100,5,1,.01,.01,.005,.01,.03,100);
>> plot(S)
```

به شکلی مشابه زیر برای قیمت دست خواهیم یافت:



شکل ۸. شبیه‌سازی نوسانات بر اساس مدل NGARCH.

```
function
[S,Volat]=HestonSim(T,N,dt,Mu,Gamma,Theta,Kappa,Rho,S0
,Volat0)
S=ones(T,N)*S0;
Volat=ones(T,N)*Volat0;
for t=2:T
    Phi1=randn(1,N);
    Phi3=randn(1,N);
    Phi2=Rho*Phi1+sqrt(1-Rho^2)*Phi3;
    dZ1=Phi1*sqrt(dt);
    dZ2=Phi2*sqrt(dt);
    Volat(t,:)=Volat(t-1,:)+(-Gamma*(Volat(t-1,:)-
Theta))*dt+Kappa*sqrt(Volat(t-1,:)).*dZ2;
    Volat(t,Volat(t,:)<0)=abs(Volat(t,Volat(t,:)<0));
    S(t,:)=S(t-1,:)+Mu*S(t-1,:)*dt+sqrt(Volat(t-
1,:)).*S(t-1,:).*dZ1;
```

نام فایل: HestonSim

توضیحات:

S: قیمت شبیه‌سازی شده،

Volat: نوسانات (σ_t^2) شبیه‌سازی شده،

T: طول هر مسیر،

N: تعداد مسیر،

dt: گام زمانی،

Mu: جمله رانش (μ)،

Gamma: پارامتر γ،

Theta: پارامتر θ،

Kappa: پارامتر κ،

Rho: پارامتر ρ،

S0: قیمت اولیه،

Volat0: نوسانات اولیه.

نکات:

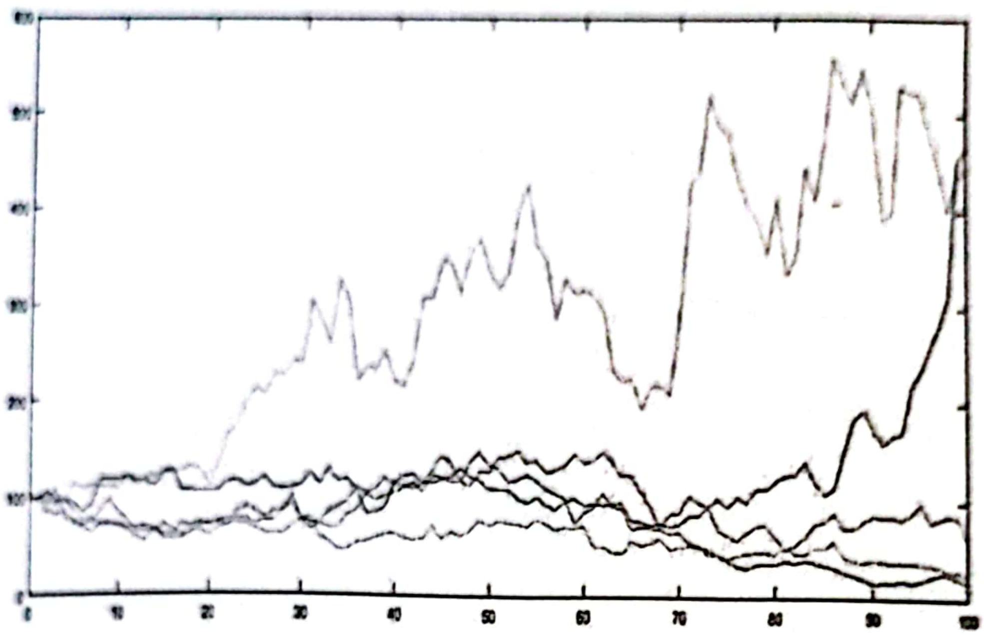
- در حلقه تکرار، مقدار dZ1 و dZ2 طوری محاسبه می‌شود که دارای ضریب همبستگی Rho باشند.
- با دستور ...=Volat(t,Volat(t,:)<0) مقادیر منفی نوسانات با قدر مطلق آن مقادیر جایگزین می‌شوند.

برنامه ۱۲. شبیه‌سازی مدل هستون.

حال می‌توان از برنامه بالا برای ایجاد ۵ مسیر به طول ۱۰۰ استفاده کرد.

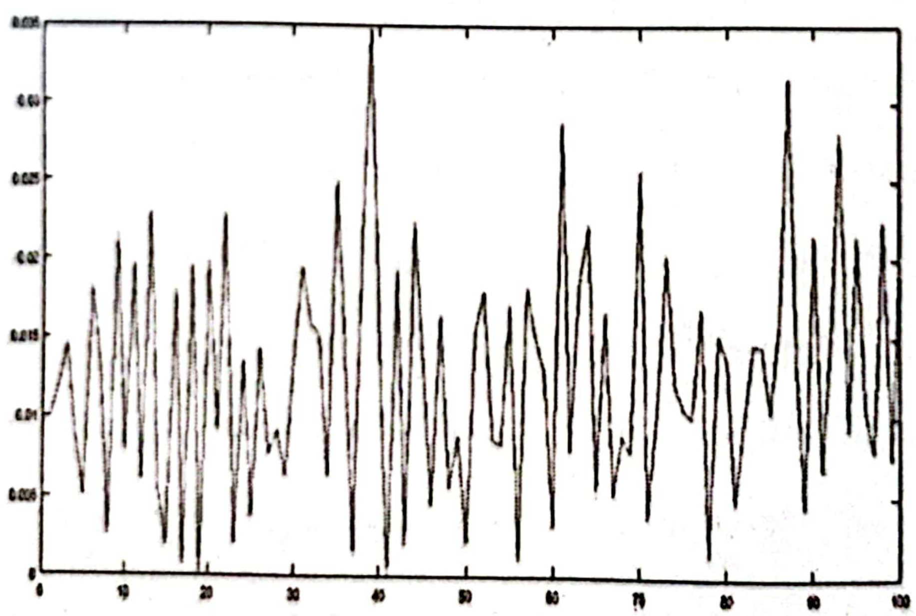
```
>> [S,Volat]=HestonSim(100,5,1,0,2,.01,.1,0,100,.01);
>> plot(S)
```

که به شکلی مشابه شکل زیر برای مقدار قیمت دست خواهیم یافت:



شکل ۹. شبیه‌سازی قیمت بر اساس مدل هستون.

در شکل زیر نیز نوسانات مربوط به یکی از مسیرهای انتخاب‌شده شکل بالا به تصویر کشیده شده است:



شکل ۱۰. شبیه‌سازی نوسانات بر اساس مدل هستون.

MATLAB®

```
function
S=JGBMSim(T,N,dt,Mu,Sigma,Lambda,Muy,Sigmay,S0)
r=zeros(T-1,N);
```

```
for i=1:N
    for t=1:size(r,1)
        n=poissrnd(Lambda*dt);
        if n~=0
            J=lognrnd(Muy,Sigmay,n,1);
        else
            J=1;
        end
        SumlogJ=sum(log(J));
        r(t,i)=(Mu-
0.05*Sigma^2)*dt+Sigma*sqrt(dt)*randn(1)+SumlogJ;
    end
end
S=ret2price(r,S0);
```

نام فایل: JGBMSim

توضیحات:

S: قیمت شبیه‌سازی شده سهم،

T: طول هر مسیر،

N: تعداد مسیر،

dt: گام زمانی،

Mu: جمله رانش (μ)،

Sigma: نوسانات (σ^2)،

Lambda: پارامتر λ ،

Muy: میانگین مقدار جهش،

Sigmay: نوسانات مقدار جهش،

S0: قیمت اولیه،

نکات:

- برای شبیه‌سازی مدل مرتون از دو حلقه داخل هم استفاده شده است تا N مسیر هر یک به طول T ایجاد شود.
- دستور n=poissrnd(Lambda*dt) برای ایجاد متغیر تصادفی با توزیع پواسون استفاده شده است.
- با استفاده از شرط if، در حالاتی که متغیر n غیرصفر است (n~=0)، با دستور ...J=lognrnd متغیر تصادفی J با توزیع لگاریتم- نرمال ایجاد می‌شود.
- پس از شبیه‌سازی مقادیر بازدهی، با دستور S=ret2price(r,S0)، این مقادیر بازدهی به قیمت تبدیل می‌شوند.