```
my = {.06, .08, .11};
V = {{0.0025, -.002, .003}, {-.002, .01, .01}, {.003, .01, .04}};
```

```
MatrixForm[my]
MatrixForm[V]
```

$$\begin{pmatrix} 0.06 \\ 0.08 \\ 0.11 \end{pmatrix}$$

$$\begin{pmatrix} 0.0025 & -0.002 & 0.003 \\ -0.002 & 0.01 & 0.01 \\ 0.003 & 0.01 & 0.04 \end{pmatrix}$$

```
expPortfolioReturn[x_, y_, z_] = {x, y, z}.my
```

$$0.06\,x + 0.08\,y + 0.11\,z$$

```
varPortfolioReturn[x_, y_, z_] = Simplify[{x, y, z}.V.{x, y, z}]
```

$$0.0025\,x^2 - 0.004\,x\,y + 0.01\,y^2 + 0.006\,x\,z + 0.02\,y\,z + 0.04\,z^2$$

```
stdPortfolioReturn[x_, y_, z_] = √varPortfolioReturn[x, y, z]
```

$$\sqrt{0.0025\,x^2 - 0.004\,x\,y + 0.01\,y^2 + 0.006\,x\,z + 0.02\,y\,z + 0.04\,z^2}$$

```
varPortfolioXYs[z_] :=
 (sol =
   FindRoot[
    {x + y + z == 1,
     (criticalReturn - expPortfolioReturn[x, y, z]) /
       stdPortfolioReturn[x, y, z] ==
      Quantile[NormalDistribution[0, 1], varLevel]},
    {x, (1 - z)/2}, {y, (1 - z)/2}]; {x /. sol, y /. sol})
```

```
? PrecisionGoal
```

PrecisionGoal is an option for various numerical operations which specifies how many effective digits of precision should be sought in the final result. ≫

```
criticalReturn = -0.1;
varLevel = 0.05;
```

```
varPortfolioXYs[.5]
```

```
{0.313493, 0.186507}
```

```
varPorts =
 Table[Append[varPortfolioXYs[(z - 1) / 10], (z - 1) / 10],
   {z, 1, 10}] // TableForm
```

```
FindRoot::lstol :
 The line search decreased the step size to within tolerance
   specified by AccuracyGoal and PrecisionGoal but was
   unable to find a sufficient decrease in the merit
   function.  You may need more than MachinePrecision
   digits of working precision to meet these tolerances. >>

FindRoot::lstol :
 The line search decreased the step size to within tolerance
   specified by AccuracyGoal and PrecisionGoal but was
   unable to find a sufficient decrease in the merit
   function.  You may need more than MachinePrecision
   digits of working precision to meet these tolerances. >>

FindRoot::lstol :
 The line search decreased the step size to within tolerance
   specified by AccuracyGoal and PrecisionGoal but was
   unable to find a sufficient decrease in the merit
   function.  You may need more than MachinePrecision
   digits of working precision to meet these tolerances. >>

General::stop :
 Further output of FindRoot::lstol will be suppressed
   during this calculation. >>
```

-0.0868002	1.0868	0
-0.0951012	0.995101	$\frac{1}{10}$
-0.0773615	0.877361	$\frac{1}{5}$
-0.0269695	0.726969	$\frac{3}{10}$
0.0736097	0.52639	$\frac{2}{5}$
0.313493	0.186507	$\frac{1}{2}$
0.442501	-0.0425008	$\frac{3}{5}$
0.385983	-0.0859832	$\frac{7}{10}$
0.326883	-0.126883	$\frac{4}{5}$
0.265349	-0.165349	$\frac{9}{10}$

```
minZInVar = .334
maxZInVar = .5167
varPortfolioXYs[minZInVar]
varPortfolioXYs[maxZInVar]
```

```
0.334
```

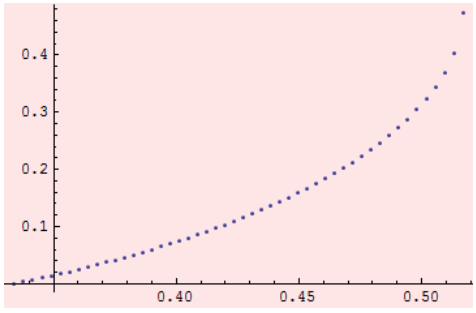
```
0.5167
```

```
{0.000050158, 0.66595}
```

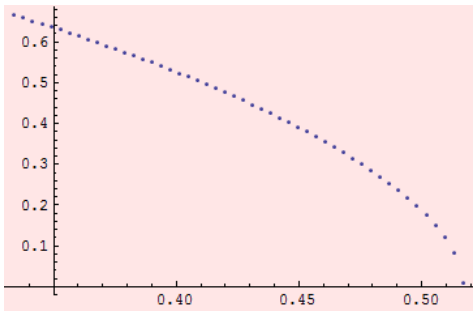
```
{0.47336, 0.00993968}
```

```
nInTable = 50;
varPortfolios = Module[{zIncrementInTable},
    zIncrementInTable = (maxZInVar - minZInVar) / (nInTable - 1);
    Table[
     Append[varPortfolioXYs[minZInVar + (i - 1) zIncrementInTable],
      minZInVar + (i - 1) zIncrementInTable], {i, 1, nInTable}]];
```

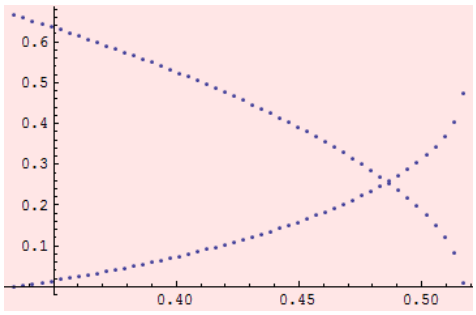
```
xAndzPlot =
 ListPlot[
  Transpose[{Transpose[varPortfolios][[3]],
    Transpose[varPortfolios][[1]]}]]
```




```
yAndzPlot =
 ListPlot[
  Transpose[{Transpose[varPortfolios][[3]],
    Transpose[varPortfolios][[2]]}]]
```

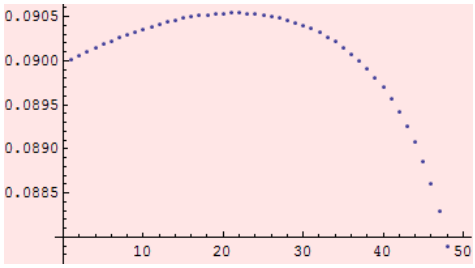



```
Show[xAndzPlot, yAndzPlot]
```

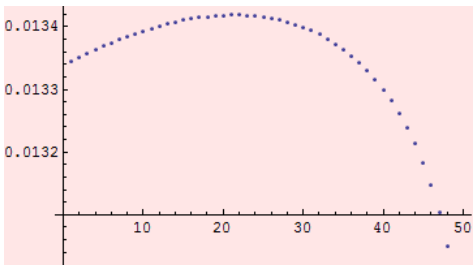



```
meanVarPortfolios = Table[varPortfolios[[i]].my, {i, 1, nInTable}];
varianceVarPortfolios =
  Table[varPortfolios[[i]].V.varPortfolios[[i]], {i, 1, nInTable}];
stdDevVarPortfolios = √varianceVarPortfolios ;
```

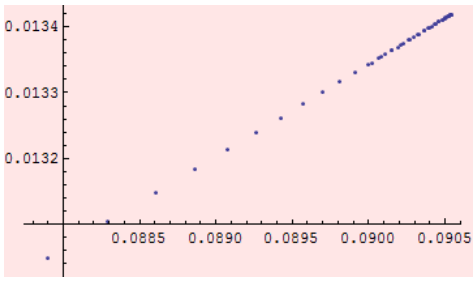
```
ListPlot[meanVarPortfolios]
```




```
ListPlot[varianceVarPortfolios]
```




```
ListPlot[Table[{meanVarPortfolios[[i]], varianceVarPortfolios[[i]]},
  {i, 1, nInTable}]]
```




```
contingentBadReturn[portfolio_] :=
 Module[{expPortfolio, stdDevPortfolio},
  expPortfolio = portfolio.my;
  stdDevPortfolio = √(portfolio.V.portfolio);
  NIntegrate[
    s PDF[NormalDistribution[expPortfolio, stdDevPortfolio], s],
    {s, -∞, criticalReturn}] /
   CDF[NormalDistribution[expPortfolio, stdDevPortfolio],
    criticalReturn]]
```

```
contingentBadReturn[varPortfolios[[1]]]
```

```
-0.148272
```

```
contingentBadReturnsVarPortfolios =
  Table[contingentBadReturn[varPortfolios[[i]]], {i, 1, nInTable}];
```

```
ListPlot[contingentBadReturnsVarPortfolios]
```