academic year 2019

Introduction to Finance with MATLAB

University Paris-Dauphine
Magistère Banque, Finance & Assurance

Gauthier Vermandel[1] *gauthier.vermandel@dauphine.fr*

Lecture 5: Bonds and Interest Rates Valuation

Content of the Lecture

1 Optimization **2**

Objectives of this lecture:

- Understanding optimization method;

[1]Department of Economics, Paris-Dauphine University. Codes available on my website: : http://vermandel.fr/bfa1.

1. Optimization

In the simplest case, an optimization problem consists of maximizing or minimizing a real function by systematically choosing input values from within an allowed set and computing the value of the function. The generalization of optimization theory and techniques to other formulations constitutes a large area of applied mathematics. More generally, optimization includes finding "best available" values of some objective function given a defined domain (or input), including a variety of different types of objective functions and different types of domains. In economics and finance, there are many applications which require optimization methods such as portfolio optimization or optimal policy.

1.1 Quadratic function

Suppose that we have a problem of the form:

$$f(x) = -0.1x^2. \tag{1}$$

What is the value of x that provides the smallest $f(x)$? We could iterate on x randomly, or use optimization methods where search directions are determined by gradients of $f(x)$. Gradients are first order derivatives. Here, we first define the function fun1, and then we use the function fmincon() to minimize function fun1. Fun1 could also be set in an external file.

```
% create a local function
fun1 = @(x) .1*x.^2;
X = -5:5;
% plot the function
plot(X,fun1(X))
x0 = 1;
[Xopt,fval] = fmincon(fun1,x0)
hold on;
        plot(Xopt,fval,'o')
hold off;
```

1.2 Noisy quadratic function

Suppose that we augment the previous problem with some noise induced by a sinus term:

$$f(x) = -0.1x^2 + \sin(x). \tag{2}$$

We run exacty the same code, the following result from the opimization are given by:

```
% create a local function
fun2 = @(x) .1*x.^2+sin(x);
X = -5:5;
% plot the function
plot(X,fun2(X))
```

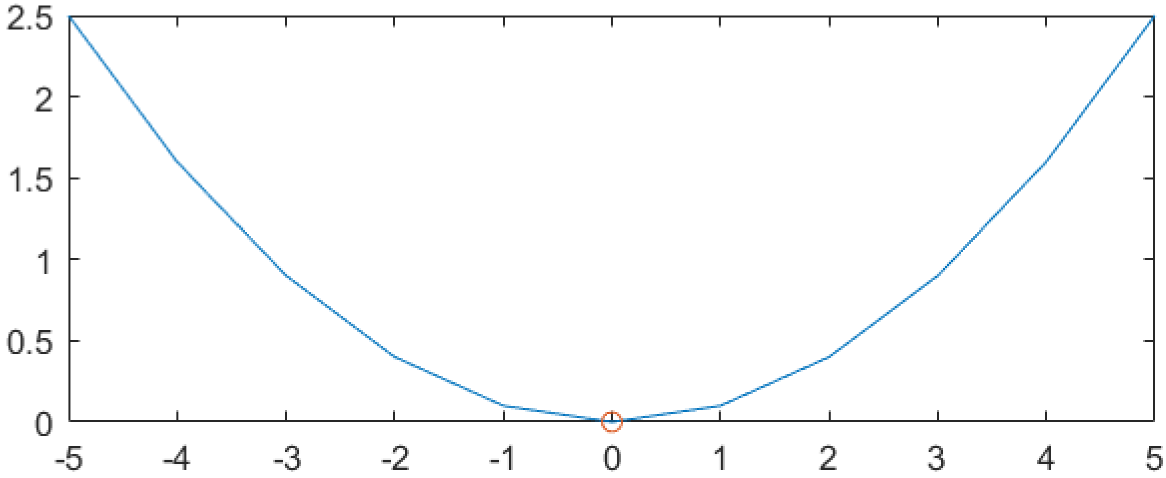

Figure 1: Solution of a quadratic function.

```
x0 = 1;
[Xopt,fval] = fmincon(fun2,x0)
hold on;
        plot(Xopt,fval,'o')
        Y = -5:0.01:5;
        plot(,fun2(Y),':')
hold off;
```

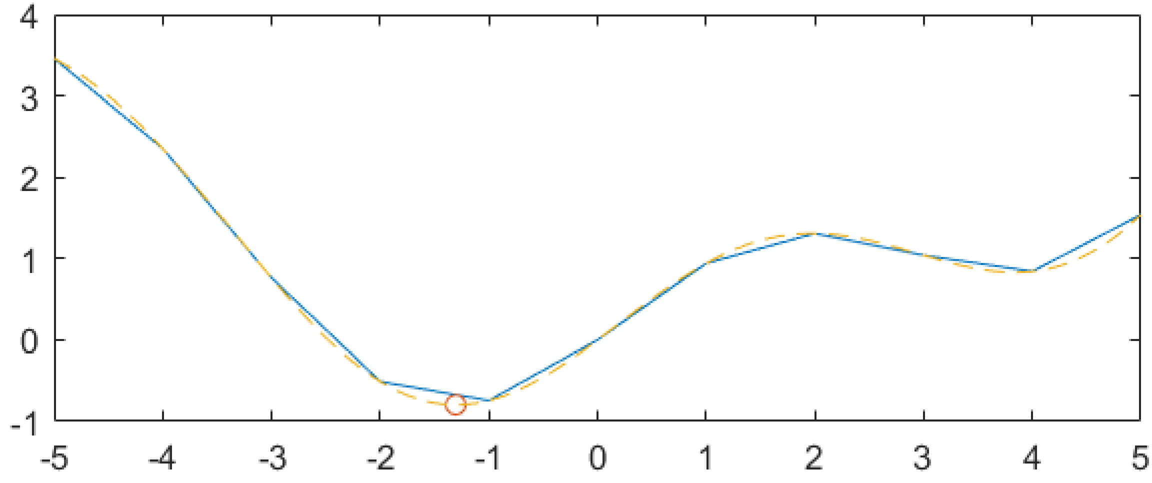

Figure 2: Solution of the noisy quadractic function.

1.3 Bounds restrictions

Suppose that we want to impose restriction on the optimization:

$$\begin{aligned} &\min_x -0.1x^2 + \sin(x) \\ s.t.\ &x > 2 \\ s.t.\ &x < 5 \end{aligned} \tag{3}$$

With bound restrictions, the global extrema is no more the solution of the problem. This can be coded easily in fmincon().

```
% create a local function
fun2 = @(x) .1*x.^2+sin(x);
X = -5:5;
% plot the function
plot(X,fun2(X))
x0 = 1;
[Xopt,fval] = fmincon(fun2,x0,[],[],[],[],2,5)
hold on;
        plot(Xopt,fval,'o')
        Y = -5:0.01:5;
        plot(Y,fun2(Y),'--')
        plot([2 2],[-1 3.5],'k')
        plot([5 5],[-1 3.5],'k')
hold off;
```

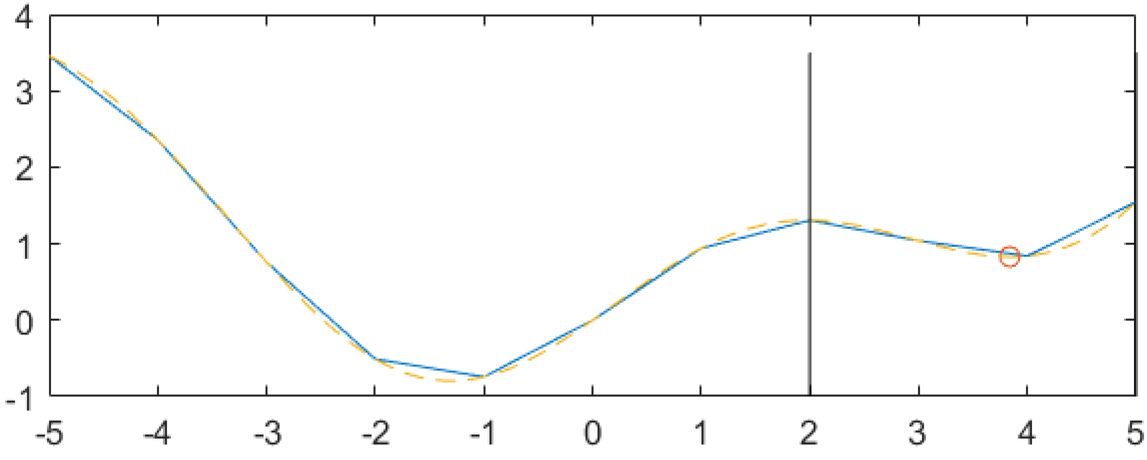

Figure 3: Solution of the noisy quadractic function with bounds restrictions.