Final Project of CS 177:The Useful Octave package of Finance

Huimin Jia & Kai Wu

May 7, 2011

Abstract

This article is mainly based on the introduction of a specified Octave package: Financial. There are several parts included. We will first introduce the installation, and then several most useful functions in this package will be listed. After that,two exercises with solutions have been attached so that it will help readers to understand the package more deeply.

1 Introduction

Financial package is a tool used for economic, commercial and stock computing in Octave. It contains many functions such as manipulation and plotting diagrams. Here is the link of the package: http://octave.sourceforge.net/financial/index.html.You still need to install dependencies package Time(version $>=$ 1.0.5) and Miscellaneous(version $>=$ 1.0.6) before install this. And you need an Octave version after 3.2.4.

2 How to install package

First go to http://octave.sourceforge.net/packages.php. By going to the package web page and clicking download. Then start Octave and go to the directory where you placed the downloaded package using the cd command. Then type:
pkg install package_file_name.tar.gz
where package_file_name.tar.gz is the file name of the package you downloaded. Than simply type:
pkg load package_name or **pkg load all**
You can get FAQ about install at link: http://octave.sourceforge.net/FAQ.html#install.

3 Most Useful Functions In the Package

3.1 __fetch_google__

$[data, fields] = {}_{--}fetch\_google{}_{--}(conn, symbol, fromdate, todate, period)$
This function is used to download stock data from finance.google.com. Output $fields$ is a 1*6 cell that lists names of each data fields about stock in order to explain the 6*6 matrix of output $data$. Include: Date, Open, High, Low, Close, Volume.

var $conn$ should input the function $google()$, using to connect to google, get url "http://finance.google.com". var $symbol$ is the stock name. var $fromdate$ and $todate$ is the date datenum(serial date number) for the requested date range. If you enter today's date, you will get yesterday's data. var $period$ allows you to select the period for the data which can be "d"(daily) or "w"(weekly).

3.2 bolling

$\{\} = bolling(asset, samples)$
$\{\} = bolling(asset, samples, alpha)$
$\{\} = bolling(asset, samples, alpha, width)$

$\{[movavg, upperband, lowerband]\} = bolling(asset, samples, alpha, width)$
If no output is requested, function will plot the bollinger bands of the var *asset*. If output is requested, function will return the values for the bollinger bands.

Var *asset* is Volume came from $\_\_fetch\_google\_\_$. If given, var *alpha* is the weighting power of the moving average; 0 (default) is the simple moving average. var *width* is the number of standard deviations to plot above and below the moving average (default: 2).

3.3   highlow

$\{\} = highlow(high, low, close)$
$\{\} = highlow(high, low, close, open)$
$\{\} = highlow(high, low, close, open, color)$
This function is used to plot the var *high*, *low*, and *close* of a security. The var *close* is plotted as a tick to the right, and if var *open* is given and non-empty, it is plotted as a tick to the left. The color can override the default color for the plot.

3.4   movavg

$\{\} = movavg(asset, lead, lag)$
$\{\} = movavg(asset, lead, lag, alpha)$
$\{[short, long]\} = movavg(asset, lead, lag, alpha)$
Calculate the leading and lagging moving average of an asset. If no output is requested the data is plotted. The plots are drawn in the following order: *asset*, *lag*, *lead*. If output is requested, no plot is generated, output *short* and *long* are x and y of the coordinate system.

Var *asset* is Volume came from $\_\_fetch\_google\_\_$, *lead* is the leading stock, *lag* is the lagging stock. If given, var *alpha* is the weighting power of the delay; 0 (default) is the simple moving average, 0.5 would be the square root weighted moving average, 1 would be linear, 2 would be squared, ..., and 'e' is the exponential moving average.

3.5   pointfig

$\{\} = pointfig(asset)$
This function is used for plotting the point figure chart of an asset. In the chart, the upward price movements are plotted as Xs, while downward movements are plotted as Os.

3.6   posvolidx and negvolidx

$\{pvi\} = posvolidx(c, vol, initpvi)$ $\{nvi\} = negvolidx(c, vol, initpvi)$
This function is used for computing the positive(negative) volume index of a security based on its closing price and volume.

3.7   onbalvol

$\{obv\} = posvolidx(c, vol)$
This function is used for computing the on balance volume of a security based on its closing price and volume. The output will be a column vector, and the first number in the output is always 0.

3.8   pmt

$\{\} = pmt(r, n, a, l, m)$
Return the amount of periodic payment necessary to amortize a loan of amount a with interest rate var r in n periods.

The optional argument var l may be used to specify a terminal lump-sum payment. The optional argument var m may be used to specify whether payments are made at the end, "e"(default) or "b"(at the beginning) of each period.

3.9   mirr

$rate = mirr(flow, finrate, reinvestrate)$
Compute the modified internal rate of return. Take periodic cashflows as a vector and the finance rate, Var $finrate$, for negative cash flows and a reinvestment rate, $reinvestrate$, for positive cash flows.

4   Exercise

4.1   Problem 1

Link to finance.google.com and retrieve the datasets of yahoo from Jan.1st,2010 to Mar.15th, 2010. Then give the plotting work as shown below:

[1] Plot the bolling curve of the volume during this time period, using the simple moving average with both upper band and lower band.

[2] Plot the highlow curve for this datasets.

[3] Plot the moving average curve for the closed prices during this time period, giving 3 leading samples and 20 lagging samples.

[4] Plot both upward & downward trend of the closed prices.

4.2   Problem 2

Using the dataset from Exercise 1, do the following computing work:

[1] Compute both the positive volume index and negative volume index of a security based on the closing price and volume, assuming that the starting value of the pvi and nvi is 100. Also, give the on balance volume of a security based on the closing price and volume.

[2] Assuming that one company has borrowed 100,000 from the bank at the rate of 0.025 per month, compute how much it have to pay back monthly to the bank in 1 year. Compare the two conditions which are paid at the beginning of the month and at the end of the month.

[3] A set of data about the cash flow was given from the year 2000 to 2010,which are 30,000 -10,000 50,000, 100,000,-40,000 -80,000 35,000 -50.000 -65,000 15,000. Compute the modified internal rate of return, assuming that the finrate value= 0.05 and the reinvestrate value = 0.03.

5   Solutions

5.1   Solutions for problem 1

$a = datenum(2010, 1, 1); b = datenum(2010, 3, 15);$
$[d, f] = \_\_fetch\_google\_\_(google(), yhoo, a, b, "d");$

[1] $Bolling(d(:, 6), 2, 0, 2)$ (See Chart 1)

[2] $highlow(d(:, 3), d(:, 4), d(:, 5), d(:, 2))$ (See Chart 2)

[3] $movavg(d(:, 5), 3, 20, 0)$ (See Chart 3)

[4] $pointfig(d(:, 5))$ (See Chart 4)
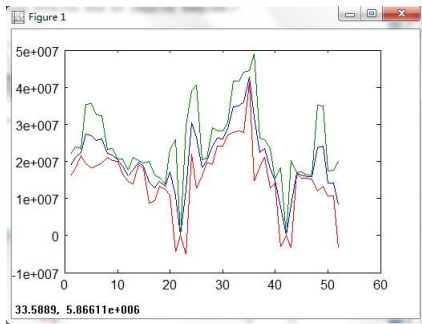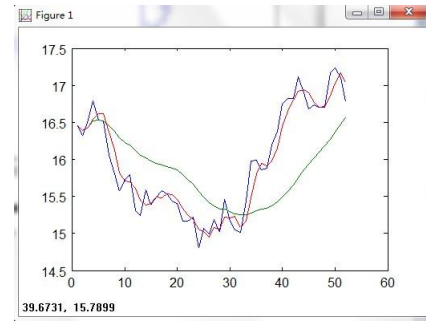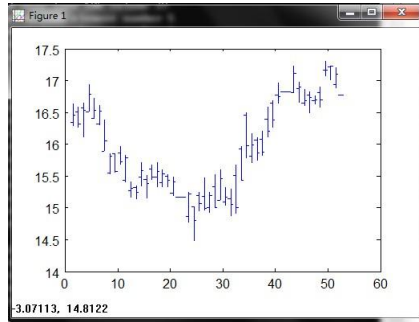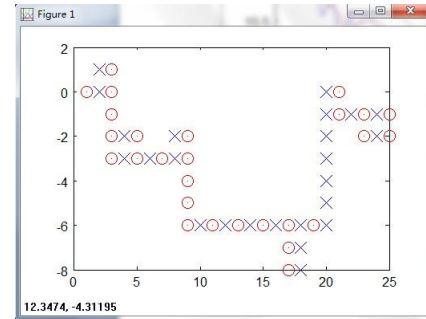
Chart 1

Chart 3

Chart 2

Chart 4

5.2 Solutions for problem 2

[1] posvolidx(d(:,5),d(:,6),100); negvolidx(d(:,5),d(:,6),100); onbalvol(d(:,5),d(:,6))

[2] pmt(0.025,12,100000,b); pmt(0.025,12,100000,e)

[3] mirr([30000 -10000 50000 100000 -40000 -80000 35000 -50000 -65000 15000],0.05,0.03)

6 Conclusion

After the problem-solving process with several functions in financial package, I think readers are able to be more familiar with the use of package. The process of applying them to the financial model curve plotting and computing would be very helpful. However, there are still many other functions not mentioned in this project. We strongly encourage you to check those files and see whether it is useful for your later study life.

References

[1] The octave financial package, available at:
http://octave.sourceforge.net/financial/index.htmll.

[2] The stock data from Google Finance, available at:
http://finance.google.com.