# TTA-45046 Financial Engineering
# Project Work 2017

April 7, 2017

## 1 Introduction

The idea of the project work is to learn to price structured derivatives. The structured product to be priced is an actual contract offered by Pohjola in 2017.

The programming environment is MATLAB, Python, or R. You must write your code by yourself. **Comment** your code so that others can easily get the idea what your code is doing.

The project work must be returned no later than **21.05.2017** via **Moodle**.

The maximum points of the work is 8.[1] In order to get the maximum 8 points, you must return your solution in time with no errors. Factual errors, coding errors, missing functionality and very poor coding style will reduce the points according to their severity. If there are some major errors in your code which prevent accepting your work, it is possible to correct them according to the feedback given by Milla Siikanen and get it accepted with lowered points. After you have submitted your work, you may be interviewed about it (about ten minutes) by Juho Kanniainen.

The pricing method is **antithetic Monte-Carlo**.

## 2 Objective

The task is to price a structured contract called *OP NOKIA AUTOCALL 3/2017* issued by Pohjola Bank[2]. You should answer the question about the fair price of the contract. You should also study how changes in the parameter setting affect the fair price of the contract (sensitivity analysis).

[1] The maximum points of the exam is 18.

[2] Here is a link to Finnish documentation, however it is not needed to conduct the project work and, in case there are inconsistencies, follow instructions in this Project Work 2017 assignment.

# 3 The Contract

The contract is based on Nokia stock:

- Nokia Oyj, NASDAQ OMX Helsinki

The *Effective Date* $t_0$ of the contract is 15 March 2017 and the contract expires on 3 April 2020 ($T$). For simplicity we can assume that we are making our valuation on the effective date (and that the last prices of our historical stock data are the prices at $t_0$).

## 3.1 Terms of the contract

The trader who invests in the contract pays on the Effective Date $t_0$ amount

$$\text{NA} \times \text{RI},$$

where NA is Notional Amount and RI is the rate of issue. This is the price of the contract. When you determine the fair price (the price at which there is no arbitrage opportunities), you can actually determine the fair rate of issue RI.

The contract may be terminated early given the conditions described later in this section. The amount the investor receives at maturity $T$ (if there is no early termination) is also described later in this section.

### 3.1.1 Early termination

There are five Observation Dates $\tau_i$ before the final observation date:

- $i = 1$: $14^{th}$ September 2017
- $i = 2$: $13^{th}$ March 2018
- $i = 3$: $14^{th}$ September 2018
- $i = 4$: $15^{th}$ March 2019
- $i = 5$: $16^{th}$ September 2019

We mark the observed stock price on the Effective date $t_0$ by $S^*_{t_0}$ and the observed stock price on Observation day $\tau_i$ by $S^*_{\tau_i}$. If on any observation date $\tau_i$

$$\frac{S^*_{\tau_i}}{S^*_{t_0}} - 1 \geq 0,$$

the investor receives amount according to following formula:

$$\text{NA} \times (1 + i \times \text{FR}),$$

where FR is the fixed rate and $i$ is the ordinal of the observation date.

The payment is done on the corresponding payment day $t_{\text{payment},i}$. The payment dates are

- payment,$i = 1$: $3^{rd}$ October 2017
- payment,$i = 2$: $3^{rd}$ April 2018
- payment,$i = 3$: $3^{rd}$ October 2018
- payment,$i = 4$: $3^{rd}$ April 2019
- payment,$i = 5$: $3^{rd}$ October 2019

After the payment, **the contract is terminated and no further payments will be made.**

### 3.1.2 Payment at the maturity

If the there has been no early termination, the final payment depends on the stock price on the Final Observation date $\tau_6 = 17^{th}$ March 2020.

For given barrier level $B$, there are three possible cases

**1:** $S^*_{\tau_6}/S^*_{t_0} - 1 \geq 0$

In this case the amount investor receives is:

$$\text{NA} \times (1 + 6 \times \text{FR}).$$

**2:** $0 > S^*_{\tau_6}/S^*_{t_0} - 1 \geq B$

In this case the amount investor receives is NA.

**3:** $S^*_{\tau_6}/S^*_{t_0} - 1 < B$

In this case the amount investor receives is

$$\frac{S^*_{\tau_6}}{S^*_{t_0}} \times \text{NA}.$$

In all three above described cases, the payment is made at maturity date $T$.

## 4 Model

### 4.1 Stock price model

In this work we assume that the price of stock $S$ follows geometric Brownian motion:

$$dS_t = \mu S_t dt + \sigma S_t dW_t,$$

where $\mu \in \mathbb{R}$ is a constant instantaneous expected return, $\sigma > 0$ is a constant volatility, $S_0 > 0$ the initial stock price, and $W_t$ is a Wiener process.

Under the risk neutral probability measure the price of the stock follows the process

$$dS_t = rS_t dt + \sigma S_t dW_t^Q,$$

where $r$ is the risk free interest rate and $W_t^Q$ is the Wiener process under the risk neutral probability measure. For the sake of simplicity we assume that the risk free rate is positive and constant.

The stock price from time $t$ to time $t+\Delta t$ can be simulated under the risk-neutral measure as

$$S_{t+\Delta t} = S_t \exp\left\{ \left( r - \frac{1}{2}\sigma^2 \right) \Delta t + \sigma \sqrt{\Delta t} \epsilon_t \right\}, \tag{1}$$

where $\epsilon_t$ is a normal random variable for the stock at time $t$. Using Eq. (1) we can simulate the distribution of $S_{\tau_i}$, and at the end, evaluate the discounted mean of the pay-off.

### 4.2 Parameters

Use the following values of the parameters:

- Notional Amount, NA $= 1\,000$ EUR,
- Fixed Rate, FR $= 8.4\%$
- Barrier, $B = -30\%$
- The risk-free interest rate, $r = 0.1\%$
- Number of simulations: $n = 1e6$
- Time to Observations Date $\tau_i$ (in years from $t_0$):
  - $\tau_1 = 6/12 - 1/252$
  - $\tau_2 = 1 - 2/252$
  - $\tau_3 = 1 + 6/12 - 1/252$
  - $\tau_4 = 2$
  - $\tau_5 = 2 + 6/12 + 1/252$
  - $\tau_6 = 3 + 2/252$ (final observation date)
- Time to payment date $t_{payment,i}$ (in years from $t_0$):
  - $t_{payment,1} = 6/12 + 12/252$
  - $t_{payment,2} = t_{payment,1} + 6/12 = 1 + 12/252$
  - $t_{payment,3} = t_{payment,2} + 6/12 = 1 + 6/12 + 12/252$
  - $t_{payment,4} = t_{payment,3} + 6/12 = 2 + 12/252$
  - $t_{payment,5} = t_{payment,4} + 6/12 = 2 + 6/12 + 12/252$
  - $T = t_{payment,6} = t_{payment,5} + 6/12 = 3 + 12/252$ (Time to Maturity)

In addition you have to determine volatility $\sigma$. To estimate the value of $\sigma$ go to e.g. Nasdaq OMX Nordic web page and load closing price data for the stock for one year ending to $t_0$.

### 4.3 Dividends

We assume that the underlying stock will pay dividends in a similar manner as during the past years. So, as the first step, find information on e.g. from Kauppalehti web page or company's web pages about the dividends paid by the respective company during the previous years and make reasonable estimates for future dividends the stocks will pay during the contract. One option is to make the dividend amount proportional to the current stock price.

Since the barrier is checked twice a year, we cannot subtract from the original spot price the sum of the present values of all the future dividends. The correct procedure is as follows.

1. At the beginning, the original spot price is reduced only by the amount of the present values of the possible dividend paid before the first observation day $\tau_1$. If we denote modified spot price with $S_{t_0}$ and the original spot price with $S^*_{t_0}$ we can write the following equation

$$S_{t_0} = S^*_{t_0} - e^{-r(Tdiv_{t_0;\tau_1} - t_0)} \times Q_{t_0;\tau_1} \tag{2}$$

where $Q_{t_0;\tau_1}$ is the first dividend paid at time $Tdiv_{t_0;\tau_1}$ (which is between $t_0$ and$\tau_1$) and $r$ is the constant risk-free interest rate and $t_0$ is the date of your valuation (Effective date).

2. At the beginning of the period between the first and second Observation dates (*after* checking the conditions for early termination), the end price at the first Observation date $\tau_1$ is modified by reducing the amount of the discounted values of the possible dividends paid between $\tau_1$ and $\tau_2$. If we denote the simulated price of the first Observastion date $\tau_1$ with $S^*_{\tau_1}$ and modified price with $S_{\tau_1}$ and we can write the following equation

$$S_{\tau_1} = S^*_{\tau_1} - e^{-r(Tdiv_{\tau_1;\tau_2} - \tau_1)} \times Q_{\tau_1;\tau_2} \tag{3}$$

The possible dividends of the following periods are reduced with similar manner from the end prices of the previous Observation date.

Note that it does not make sense here if the adjusted stock price is negative due to a too large dividend estimate!

### 4.4 Estimating the volatility

Volatility can be estimated using the historical time series data. The estimation goes as follows:

- Load the closing price data from e.g. Nasdaq OMX Nordic web page for the stock for one year ending to $t_0$.
- Calculate the daily log-returns
$$u_{t_i} = \ln\left(\frac{S_{t_i}}{S_{t_{i-1}}}\right)$$
for all $i = 1, 2, ..., m$, where $m$ is the number of observations.
- Use Matlab's `std` command to calculate the standard deviation of the daily log-returns.
- The unbiased estimate of the volatility is obtained by dividing the standard deviation by the square root of the length of the time interval between the observations:
$$\hat{\sigma} = \frac{\mathrm{Std}(u)}{\sqrt{\Delta t}}$$
For daily observations, $\Delta t = 1/252$.

# 5 Recommendations how to proceed with your coding

## 5.1 Phase 1: Make a plan

Every good coding work start with planning. Write down, what your code should do. What is the first task, what the second and so on. How you arrange the data? Do you use vectors, 2-D or 3-D arrays for them? Visualize how data will be located in the vectors and matrices!

## 5.2 Phase 2: Implement the basic functionality with artificial parameters

You do not need to read the time series data or do any estimation at this phase, but instead you can use the following artificial values:

- Volatility $\sigma = 0.3$
- Initial stock price $S^*_{t_0} = 1$
- For simplicity, at this first phase, you can assume no dividends.
- For other parameters, you can use the real ones.

You can start the actual coding with implementing the core functionality, the stock price process and payoff calculation. Write code that implements Equation 1 to simulate the stock prices for the Observation Dates. After your stock price simulation works, add the payoff-function (early termination and final payment) and calculate the present value of the payoff. Remember to use **antithetic Monte Carlo** method for the simulations!

### 5.2.1 Informal Review

It is possible to get comments about your code after you have done the phase 2. Any errors found (and fixed) at this phase will not lower your points. You can also ask comments about your code during the remaining weekly exercises.

## 5.3 Phase 3: Add all the required features

Next you will need the historical stock price data that you can import to excel from e.g. Nasdaq OMX Nordic web page. You can e.g. read the excel-file by using `xlsread` command or you can save the excel file as .csv and read it with `textscan` function. Then you need to estimate the volatility from the time series data.

At this phase, you also need to make your own (reasonable) estimate about the future dividends the stock will pay during this contract. You can e.g. write an excel-file containing your estimated dividends and then read it into MATLAB, or another option is to make the dividend amount proportional to the current stock price, and save the proportions of the stock price that are paid as dividend instead of the absolute values.

## 5.4 Phase 4: Add sensitivity analysis

Before going to sensitivity analysis, your code should be working properly. Now you need to make your code to run a few times with different parameter values for volatility.

Use values $\sigma^{\text{high}} = 1.2\sigma$ and $\sigma^{\text{low}} = 0.8\sigma$. The easiest way to do this might be that you convert the code you have written until now to a function, which takes $\sigma$ as an input the parameter. In order to call the function you created with the different parameter values, you need to write a new .m-file. Maybe you also need to change a little bit your old code to accept the parameter as input. If you plan this in advance in your plan in phase 1, there should not be any big task at this phase to make this change. Add some comments to your conclusion about how these experiments change the value of the contract.

### 5.5 Some practical hints for MATLAB coding

- When you start coding, use small number of paths, less than 10, so that you can easily check that you read and write your vectors and matrices correctly.
- Pressing Crtl-C (in command window) stops MATLAB, if the execution seems to take too long or the code is e.g. in an infinite loop.
- Use breakpoints to debug your code. You can add a breakpoint by pressing the "-"-sign at the left-hand border of the editor on the line. Run the code by pressing the green arrow, and the code execution stops at that line. You can then check in the command window the values of your parameters.
- You can measure the processing time by inserting `tic` command before and `toc` command after the code block you want to measure.
- Use descriptive variable names. E.g. `Payoff` is much better than e.g. `p` for a variable which stores the "Payoff".

### 5.6 Fair play

Every student has to write their own code. It is ok, though, to discuss your implementation and possible problems with others. Share thoughts, not code! Be prepared to orally present your project to the staff of this course.

### 5.7 Your conclusion

You can give your own opinion about this contract, based on the results you've got during this work. You can just tell why this is such a contract you'd like to enter into? How risky is this contract i.e. is there big variance in payoff? Be brief - max 10 sentences, just give your view of this contract! (No extra points from this.)

### 5.8 Final Phase: Submit your work by 21.05.2017

You should return your work via Moodle. Submit only one zip-file, which contains all the necessary files your code needs to run and your brief conclusion. **Well-commented** code that produces clear output (plus conclusion) is sufficient and no extra report is needed. Name the zip-file as XXXXXX_ProjectWork2017.zip, where XXXXXX is your student ID. Although the deadline for submission is rather late, we encourage you to do the work before the exam, as doing this has shown a positive correlation with exam points earlier.