Université Nice SOPHIA ANTIPOLIS

POLYTECH NICE-SOPHIA

APPLIED MATHEMATICS AND MODELLING

# Microcredit models and Yunus equation

Project report

June 2010

**Authors:** Léo AUGÉ – Aurore LEBRUN – Anaïs PIOZIN

**Project supervisor:** Marc DIENER

**In association with the thesis student:** Pheakdei MAUK

# TABLE OF CONTENTS

**INTRODUCTION .......... 2**

**I- YUNUS EQUATION .......... 3**

1. CAPITALIZATION .......... 3
2. YUNUS MODEL .......... 4

**II- MODELLING WITH LATENESS .......... 5**

1. BUILDING OF THE MODEL .......... 5
2. LIMITS OF THIS MODEL .......... 5
3. RATES CALCULATION .......... 6
   *a) Effective average rate .......... 6*
   *b) Rate for a geometric distribution .......... 7*
4. DEFAULT OF A LOAN .......... 10
5. COMPARISON WITH ANOTHER MODEL .......... 12

**III- MODELLING WITH CORRELATED LATENESS .......... 13**

1. BUILDING OF THE MODEL .......... 13
   *a) Model with negative correlation .......... 14*
   *b) Model with positive correlation .......... 14*
2. RESULTS .......... 15
   *a) Positive correlation results .......... 15*
   *b) Negative correlation results .......... 16*
3. ANALYSIS OF THE RESULTS .......... 16

**IV- CONSUMER CREDIT .......... 17**

1. CONSUMER CREDIT MODEL .......... 17
2. SIMULATION .......... 17
3. CETELEM MODEL .......... 18

**CONCLUSION .......... 20**

**REFERENCES .......... 21**

**ANNEXE .......... 221**

# INTRODUCTION

Microcredit began with Muhammad Yunus in 1974 in Bangladesh after he met some women who needed only $27 to develop their bamboo stool business. All the banks refused to lend them money because they thought they would never paid-back. Ashamed of this situation, Yunus decided to loan them money from his own pocket and created the famous Grameen Bank. With this new “social business”, Yunus didn’t want to make profits but only help people to generate incomes and enable them to exit poverty. For that, he and the Grameen Bank received the Nobel Peace Prize in 2006.

Through this story, we could understand that microcredit means the loan of very small amount to people who can’t access to traditional lends. The working is simple: borrowers have to pay back frequently (generally every week) small refunds during a short time (a year) with high interests.

Nowadays, microcredit is developed in many countries around the world: there are more than 10,000 Micro Finance Institutes. It represents a business of €50 billion each year for 500 millions of borrowers. Of course, poor countries are mainly concerned about microcredits. They represent 83% of the microcredit economy. But since 2008, new kinds of countries have developed microcredits. This is the case of the United States, where 12.6% of the population lives below the poverty line.

Because of microcredit success (indeed almost every borrower refund their due), consumer credit companies begin to be interested on this business. But we can be skeptical about their intentions.

In order to study microcredits mathematically speaking and find what the applied rates for different situations are, we first took an interest in what is named the Yunus model. It is a model described in his book, ”A world without poverty”, and applied in Bangladesh. Then we decided to establish and study two models a little bit more complex, which take more parameters as the lateness in payments.

And finally, in order to compare the microcredit and the consumer credit system, we built a consumer credit model. It allowed us to find the applied rates in the case where borrowers are late in their paybacks.

# I- Yunus equation

## 1. Capitalization

To begin, we will make a brief introduction to capitalization in order to well understand the subject. The main idea of capitalization is that one euro today isn't worth one euro tomorrow but it will be worth more.

Thus, an amount $B_0$ invested during a period $\delta t$ is increased of an interest $B_0.\rho\ > 0$ and will be worth $B_0 + B_0.\rho = B_0.(1+\rho)$ after this period.

We talk about *simple interests* when the accrued interest is the same at each period. So, after n periods, $T =\ n\delta t$, it would be worth $B_0(1+n\rho)$.

But, generally, we consider that the increased interest during one period will be increased during the next periods: it is the formula of *compound interests*. So, the amount $B_0$will be worth:

- $B_{\delta t} = B_0(1+\rho)$ after one period,
- $B_{2.\delta t} = [B_0(1+\rho)](1+\rho) = B_0(1+\rho)^2$ after two periods,
  $\vdots$
- $B_{n.\delta t} = B_T = B_0(1+\rho)^n$ after n periods.

If we denote $t$ the successive instants multiples of $\delta t, t\ \in \{0, \delta t, 2\delta t, \dots, n\delta t = T\}$, and *r*, called the *continuous interest rate*, the real number such that:

$$e^{r\delta t} = 1 + \rho$$

We can rewrite the equation:

$$B_{k.\delta t} = B_t = B_0(1+\rho)^k$$

$$\Leftrightarrow B_t =\ B_0 e^{r.k\delta t}$$

$$B_t =\ B_0 e^{rt}$$

So if we choose $B_0\ = 1€$, this equation says that one euro will be worth $e^{rt}$ after a time t. It equally means that to have one euro on the date t, we need $e^{-rt}$ (that is to say a little bit less than 1 euro) today:

$$1 \longrightarrow\ e^{rt} \quad \Rightarrow \quad e^{-rt} \longrightarrow 1$$

That is what we named *capitalization*: the fact that money gains value by accumulating interests.

## 2. Yunus model

In this project, we decided to study the microcredit example of Muhammad Yunus given in his book "A world without poverty".

We considered a loan of 1000 BDT (Bangladesh Taka) and assumed that the refund requested is 22 BDT per week during 50 weeks. Let *r* be the continuous interest rate. As seen on the first part, the 22 BDT refunded after one week are worth $22.e^{\frac{-r}{52}}$, the 22 BDT refunded after two weeks are worth $22.e^{\frac{-2r}{52}}$… and so on.

So we obtain this equation:

$$1000 = 22.e^{\frac{-r}{52}} + 22.e^{\frac{-2r}{52}} + \cdots + 22.e^{\frac{-50r}{52}}$$

$$\Leftrightarrow \quad 1000 = \sum_{n=1}^{50} 22\, e^{-r.\frac{n}{52}}$$

If we denote $q$ such that $q = e^{-r}/_{52}$, we obtain:

$$1000 = 22.\sum_{n=1}^{50} q^n$$

$$\Leftrightarrow 1000 = 22.\frac{q.(1-q^{50})}{1-q}$$

$$\Leftrightarrow \quad 22q^{51} - 1022q + 1000 = 0$$

This equation is unsolvable by hand, that is why we used Scilab. This software can find all the solutions of this polynomial (it means at most 51). So we just had to pick the real one between 0 and 1 to find the solution that interested us. Thus we found q=0.9962107.

$$q = e^{-r}/_{52}$$

$$\Rightarrow r = -52 * \log q$$

$$\Rightarrow r = 19{,}74175 \ldots \%$$

So the interest rate applied by the Grameen bank is about 20%, that is quite a high rate but we supposed here that borrowers are never late in their paybacks. That is not what happened in real life because borrowers can have lots of economic or personal problems, especially in poor countries. That is why we decided to study a microcredit model which considers that borrowers can be late in some paybacks.

# II- Modelling with lateness

In this part, we decided to model the same system of microcredit but with a risk of lateness. The person who receives the loan has a probability not to pay back on time. However, we supposed that lateness on a given refund would not have any impact on the following paybacks.

So, we still supposed that someone loan 1000 BDT to another person and that the borrower must pay back each week 22 BDT during 50 weeks. But this time, the person who loaned the money can be late on several paybacks. Given that microcredits are intended to very poor people, we can't ask for lateness penalties. So the duration of the reimbursement is not 50 weeks anymore, but it will be varying according to the payback possibility of the person who need a loan.

## 1. Building of the model

Let $X_n$be the random variable which describes the taken time to reimburse on the n[th] week. Thus: $\forall n \in \{1,2,\dots,50\}$, $X_n \in \mathbb{N}^*$. Given that we suppose that what happens a given week won't influence what happens the next weeks, we can say that the random variables $(X_n)_{n\in\{1,2,\dots,50\}}$are independent.

By denoting *p* the payback probability on a given week, thus we can say that a person who borrows some money has a chance of p to reimburse his loan at the end of the first week, a chance (1-p).p to reimburse his loan at the end of the second week (because 1-p is the probability not to pay back on a given week), etc.

Thus, the probabilities associated to each value are: (w is the number of week taken to pay back on the n[th] week)

| w | 1 | 2 | 3 | | k |
|---|---|---|---|---|---|
| $\boldsymbol{P(X_n = \mathrm{w})}$ | $p$ | $(1-p).p$ | $(1-p)^2.p$ | $...$ | $(1-p)^{k-1}.p$ |

So we can say that each $X_n$follows a geometric law:

$$X_n \backsim \mathcal{G}^*(p) \Leftrightarrow \ P(X_n = \mathrm{k}) = (1-p)^{k-1}.p$$

If denote (as in the first part) $t_1, t_2, \dots, t_{50}$ the payback moment, we obtain:

$$t_n = t_{n-1} + X_n$$

## 2. Limits of this model

The main problem with this model is the independence of the random variables. Indeed we supposed that the lateness on a given week don't have any effects on the following weeks. But in real life, if someone has an accident, an illness or an economic problem, he won't be able to pay back on time. This problem could have an influence on the following paybacks. The person can still be sick or handicapped by this accident.

## 3. Rates calculation

We will now study rates with two different methods.

The first one, named the *effective average rate*, is in fact the expected value of the Yunus equation. And the second one is the mean of calculated rates for lots of examples of geometric distributions.

We are going to study these two values in order to see if the second tend to the first one.

### a) Effective average rate

Let $\bar{r}$ be the effective average rate. This number satisfies the Yunus equation in expected value:

$$1000 = E(\sum_{n=1}^{50} 22\, e^{-\bar{r}.t_n})$$

$$\Leftrightarrow\ 1000 = E(\sum_{n=1}^{50} 22\, e^{-\frac{\bar{r}}{52}.(X_1+X_2+\cdots+X_{50})})$$

$$\Leftrightarrow\ 1000 = 22.\sum_{n=1}^{50} E(\,e^{-\frac{\bar{r}}{52}.X_1}.\,e^{-\frac{\bar{r}}{52}.X_2}\ldots e^{-\frac{\bar{r}}{52}.X_{50}}) \qquad (*)$$

Each $X_i$ is supposed to be independent and follows the same geometric law $X_n \backsim \mathcal{G}^*(p)$ :

$$(*) \Leftrightarrow\ 1000 = 22.\sum_{n=1}^{50} E\left[\left(e^{-\frac{\bar{r}}{52}}\right)^n\right] \Leftrightarrow\ 22.\sum_{n=1}^{50} E\left(e^{-\frac{\bar{r}}{52}}\right)^n$$

We now denote $\bar{q} = e^{-\bar{r}/52}$:

$$(*) \Leftrightarrow\ 1000 = 22.\sum_{n=1}^{50} E(\bar{q}^n)$$

We will calculate $E(\bar{q}^n)$:

$$E(\bar{q}^n) = \sum_{k=1}^{\infty} \bar{q}^k.(1-p)^{k-1}p = \frac{p}{1-p}.\sum_{k=1}^{\infty}\left(\bar{q}(1-p)\right)^k = \frac{p.\bar{q}}{1-\bar{q}(1-p)}$$

$$(*) \ \Leftrightarrow\ 1000 = 22\sum_{n=1}^{50} y^n \qquad \text{with } y = \frac{p.e^{-\bar{r}/52}}{1-e^{-\bar{r}/52}.(1-p)}$$

We denote by y the solution of the Yunus equation (saw on the first part), so we can easily have its value.

From the expression of y, we deduce the expression of $\overline{r}$ :

$$\bar{r} = 52.\log\left(\frac{p + y(1-p)}{y}\right) = 52.\log\left(\frac{p}{y} + (1-p)\right) = 52.\log\left(1 + p.(\frac{1}{y} - 1)\right)$$

We can now plot $\overline{r}$ as a function of the payback probability *p*.

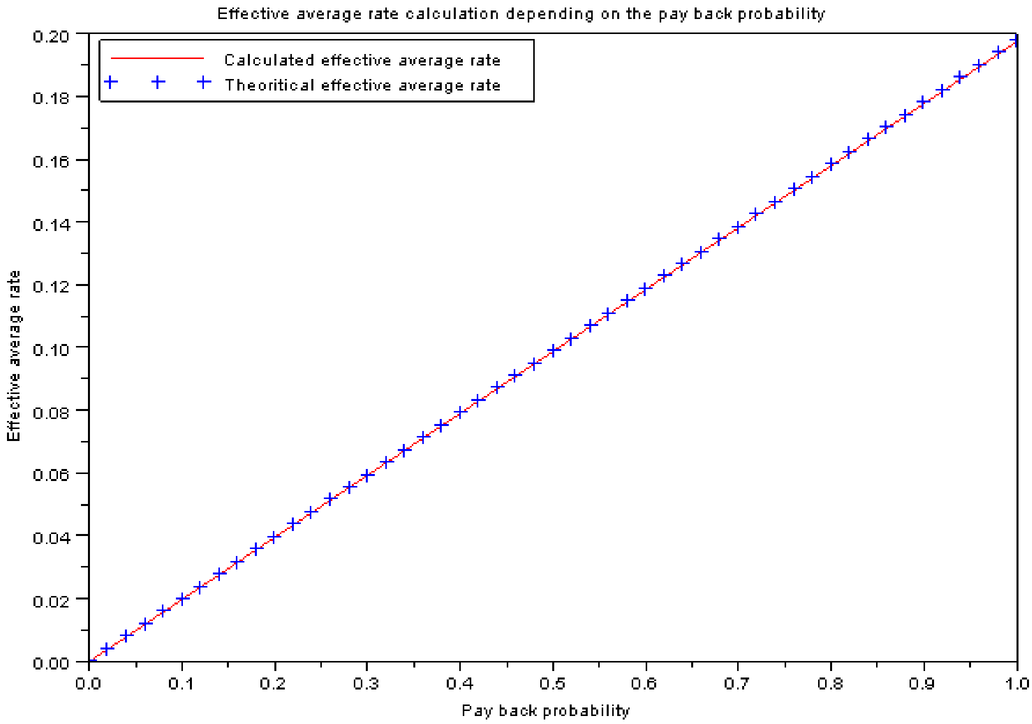



FIGURE 1 : EFFECTIVE AVERAGE RATE IN FUNCTION OF THE PAYBACK PROBABILITY

**Remark:** $\log(1 + X) \approx X$ (when x is near to zero) so we can make an approximation of $\bar{r}$ with $\bar{r} = 52.p(\frac{1}{y} - 1)$. That is why the graph $p \longmapsto \bar{r}(p)$ looks like a straight line.

## b) Rate for a geometric distribution

It should be reminded that we want now to calculate the rate for someone who pays back according to a geometric distribution. To do that, it is essential to be able to simulate a geometric distribution. Then we have to rewrite the equation which governs our new model. And finally we will be able to simulate lots of geometric distributions (which represent the payback times), find the associated rates of each distribution and calculate the average rate of these distributions.

## Simulate a geometric distribution

Most of computing tools are able to give a random value equally distributed between 0 and 1. Our goal is to have value on ℕ distributed according to a geometric law… So we needed a function which can do that.

We are able to have a random number between 0 and 1. Moreover the sum of the probabilities of a geometric law is 1. So we will divide the interval [0;1] into an infinite number of intervals whose lengths correspond to the probabilities of a geometric law.

Thus we will generate a random number between 0 and 1 and look in which interval it is.

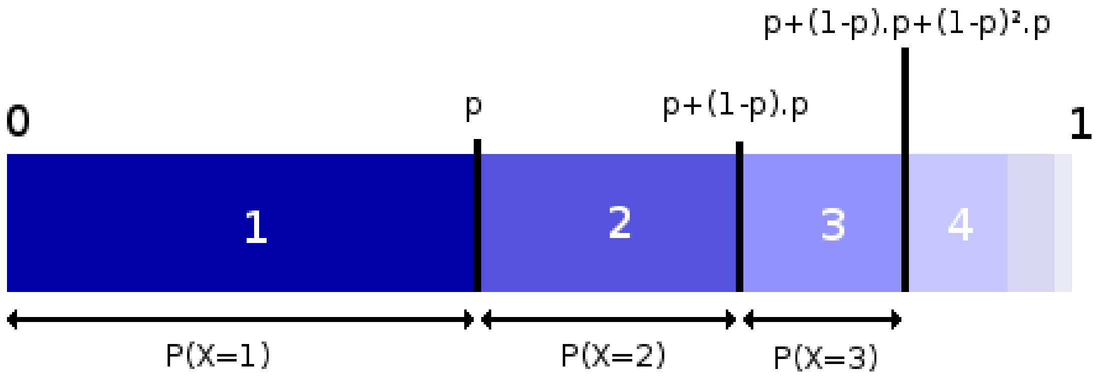



FIGURE 2 : WANTED INTERVAL FOR A GEOMETRIC DISTRIBUTION

This solution can work but it can be costly to compute it like that. A calculation can help us to be more efficient. Indeed we can remark that:

$$\begin{cases} 0 \le x \le p \\ p + p(1-p) + \cdots + p(1-p)^{k-1} \le x \le p + p(1-p) + \cdots + p(1-p)^k, \quad \forall k \in \mathbb{N}^* \end{cases}$$

$$\Leftrightarrow \begin{cases} 0 \le x \le p \\ p \sum_{i=0}^{k-1} (1-p)^i \le x \le p \sum_{i=0}^{k} (1-p)^i, \quad \forall k \in \mathbb{N}^* \end{cases}$$

We will take an interest on the second equation. We want to find the term k+1 to have a number distributed according to the geometric law with parameter p:

$$p \sum_{i=0}^{k-1} (1-p)^i \le x \le p \sum_{i=0}^{k} (1-p)^i, \quad \forall k \in \mathbb{N}^*$$

$$\Leftrightarrow p \frac{1-(1-p)^k}{1-(1-p)} = 1-(1-p)^k \le x \le p \frac{1-(1-p)^{k+1}}{1-(1-p)} = 1-(1-p)^{k+1}$$

$$\Leftrightarrow \mathrm{k} \le \log_{(1-p)}(1-x) = \frac{\log(1-x)}{\log(1-p)} \le k+1$$

So we just need to take the upper integer of $\frac{\log(1-x)}{\log(1-p)}$ to find the wanted distribution. It is now easier and faster to simulate a geometric distribution.

## Adaptation of the Yunus equation

It should be reminded that the Yunus equation is:

$$1000 = \sum_{n=1}^{50} 22\, e^{-\frac{r}{52}t_n}$$

Here we have $t_n = \frac{x_1+x_2+\cdots+x_n}{52}$, with $(x_1, x_2, \dots, x_{50})$ a geometric distribution:

$$\Leftrightarrow\ 1000 = 22e^{-\frac{r}{52}x_1} + 22e^{-\frac{r}{52}(x_1+x_2)} + \cdots + 22e^{-\frac{r}{52}(x_1+x_2+\cdots+x_{50})}$$

$$\Leftrightarrow\ 1000 = 22\left(q^{x_1} + q^{x_1+x_2} + \cdots + q^{(x_1+x_2+\cdots+x_{50})}\right), \qquad with\ q = e^{-\frac{r}{52}}$$

$$\Leftrightarrow\ 1000 = 22\left(q^{x_1}\left(1 + q^{x_2}\left(1 + q^{x_3}\left(1 + \cdots\left(1 + q^{x_{50}}\right)\right)\right)\right)\right)$$

With Scilab, we can implement a function to solve this equation and find the rate value for this distribution.

## Application to lots of distributions

We will now simulate lots of geometric distributions with parameter p=0.8, calculate the rate for each distribution and finally calculate the average of these rates:

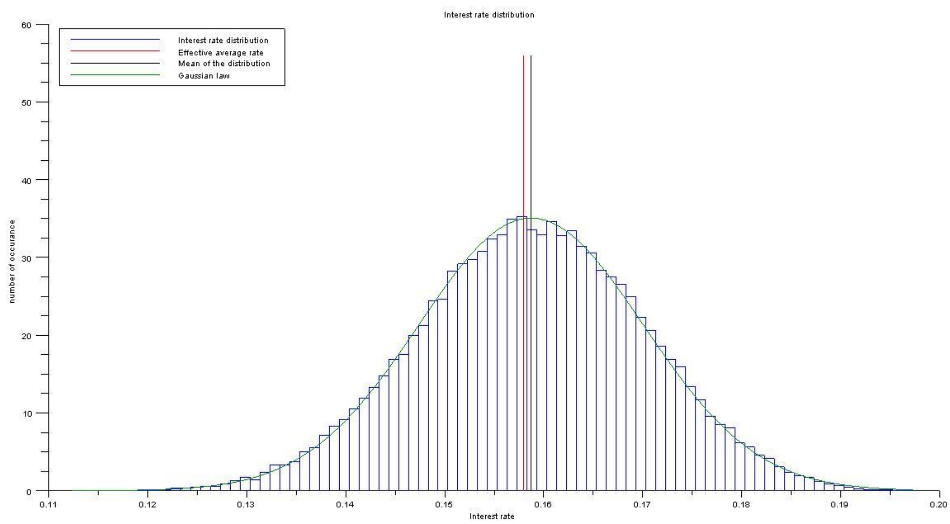



FIGURE 3 : INTEREST RATES FOR 50 000 GEOMETRIC DISTRIBUTIONS

We can see on this graph that the average of the calculated rates (the black line) is quite similar to the effective average rate (the red line). We can also see that the interest rates seem to follow a normal law. We can test if the distribution follows a normal law with normality tests.

### Normality studying

In order to check if the rates are distributed according to a normal law we could make normality tests on our data. But these tests are too long to be led in this three-week project. So we just checked two properties of the Normal law:

- We first calculated with Scilab the third moment ($\mu_3$) of the obtained data:
$$\mu_3 = 0.00405 \dots$$
The third moment of a Normal law is equal to zero, we cannot conclude on our result with this test.

- We also calculated (with Scilab) the "kurtosis" ($\beta_2$)of the obtained data:
$$\beta_2 = \frac{\mu_4}{\sigma^4} = 1.02 \dots$$
For a normal law, the Kurtosis must be equal to three (more or less). So we can say that our distribution don't follow the normal law.

## 4. Default of a loan

### Default calculation

We say that there is a *d-default* if $Max\{X_1, X_2, \dots, X_n\} \geq d$. It means that we set a kind of tolerance limit *d*, and there is a d-default when someone doesn't pay back one time before the d[th] week.

Thus, if we denote $\pi_d$ the probability of a d-default for a given d, with p the payback probability, we obtain:

$$\pi_d(p) = P(Max\{X_1, X_2, \dots, X_n\} \geq d) = P(X_1 \geq d \cup X_2 \geq d \cup \dots \cup X_n \geq d)$$
$$= P(X_1 \geq d) + P(X_2 \geq d) + \dots + P(X_n \geq d)$$
$$= 1 - P(X_i < d), \qquad \forall i \in \mathbb{N}^*$$
$$= 1 - [p + p(1-p) + p(1-p)^2 + \dots + p(1-p)^{d-1}]$$
$$= 1 - p.\sum_{k=0}^{d-1}(1-p)^k = 1 - p.\frac{1-(1-p)^d}{1-(1-p)} = 1 - [1 - (1-p)^d]$$
$$= (1-p)^d$$

$$\pi_d(p) = (1-p)^d$$

Then, we can plot $\pi_d$ as a function of p:

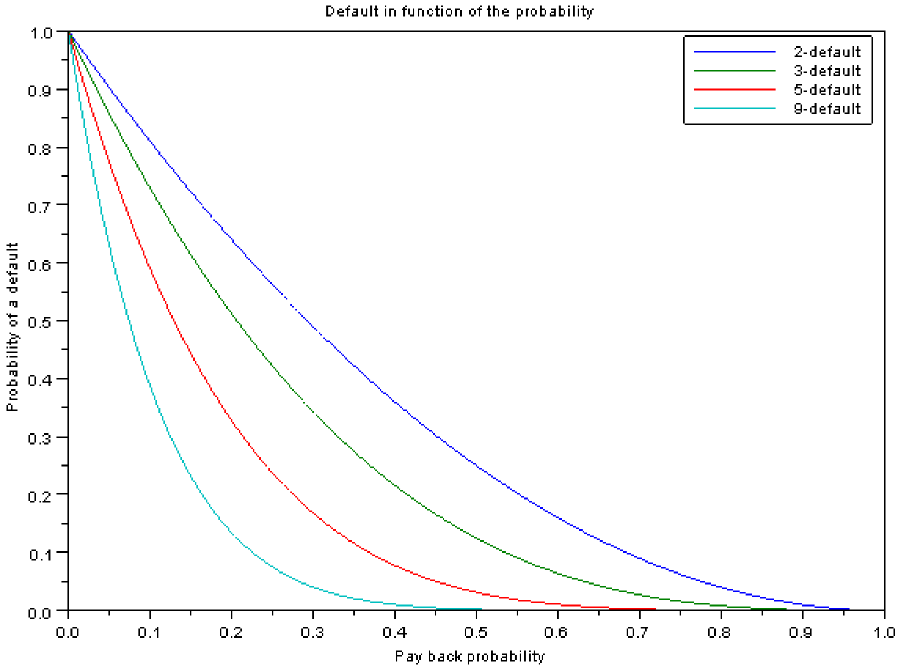



FIGURE 4: DEFAULT IN FUNCTION OF THE PAYBACK PROBABILITY

We can see on the graph that the default probability in function of the payback probability is a decreasing function. Moreover, the lower the tolerance level is (it means the smaller d is), the less decreasing is the function.

## D-default utilization

We will now suppose that we want a probability of d-default of 5% for a given d, it means: $\pi_d(p) = 5\%$ where p is the probability of payback. Our goal is to find the payback probability corresponding to this equation to see in which way the tolerance level has an effect on the probability of a d-default.

So we have:

$$\pi_d(p) = (1-p)^d$$

$$\Leftrightarrow \quad p = 1 - \sqrt[d]{\pi_d(p)}$$

We can calculate easily the corresponding payback probability, and equally its related effective average rate, for example:

- $d = 2 \implies p = 77.64\ \%\ and\ \overline{r} = 15.33\ \%.$
- $d = 3 \implies p = 63.16\ \%\ and\ \overline{r} = 12.48\ \%.$
- $d = 5 \implies p = 45.07\ \%\ and\ \overline{r} = 8.91\ \%.$

So we can see that the more tolerant the loaner is, the smaller is the rate.

# 5. Comparison with another model

We decided to compare the Yunus model with a quite similar model. The Cambodian student who will study this subject for the next two years tell us that in Cambodia, people are used to payback every two weeks. So we assume now that borrowers will reimburse 44BDT every two weeks.

In this case, the interest rate is 19.34997%. If we compare it to the rate when people pay back every week (19.74175%), you can see that it is a little bite inferior. This can be explained by the fact that one euro today is not worth one euro tomorrow. Money lent to people don't yield a profit for the lender during two weeks (whereas one week, in the previous model) because he hasn't got it. So the lender "loose" money during this time. That is why, the interest rate is inferior. To pay-back at the same interest rate, every two weeks, the borrower should refund 44,084 BDT every two week. The rate is 19.74324%.

# III- Modelling with correlated lateness

The model which uses geometric random variables is not accurate because we can't say if at a time a person has a problem on his pay-back, that won't have an influence on his next pay-back. That is why we introduced a correlation between two successive variables. In other words, that means the following variable depends on the previous one.

## 1. Building of the model

We built our model as a sum of a variable distributed according to a geometric law and a variable depending on the previous one. We initialised the first variable with its geometric part: it can't depend on something that happens before.

In our model, the correlation is a ratio of the pay-back probability. We thought that the higher the probability to pay-back is, the less high the correlation between two variables is. Indeed, if the probability to pay-bay is high, people will refund their dues approximately each week: the correlation is near to zero. The contrary is equally true: if the probability to pay-back is low, people will refund their dues with lateness and the correlation his high. That is why the correlated term is a ratio of (1-p).

Another property of the correlation is that it can be positive or negative. You have to remind that a microcredit is used in poor countries. So, we can easily imagine at least two reasons for an inability to pay back:

- We can first imagine that the lender is very sick and can't go to his work. Because he doesn't work, he won't have money this week. The next week, when he will have money, he would prefer to spend it to feed his family than to refund his loan. In this case, there is a positive correlation.
- Then we can imagine that the lender sell bamboo stools. During two weeks, his business is very bad, he won't be able to pay is loan for two weeks, but on the third week his business is very great, he will be able to refund his loan. In this case, there is a negative correlation.

Then, in order to be consistent with the reality and don't have very long time of refunds, we divided the correlation term by ten. We chose ten by trial and error method. If we denote by *a* the correlation term, we got:

$$a = \frac{(1-p)}{10}$$

## a) Model with negative correlation

According to what we said before, mathematically speaking, the model can be written:

$$\begin{cases} x_1 = \varepsilon_1 \\ x_i = I\left(1 + \varepsilon_i - \dfrac{(1-p)}{10}(x_{i-1} - 1)\right) \end{cases}$$

where $(x_i)_{1 \le i \le n}$ is the vector which indicate the number of weeks between two refunds. $(\varepsilon_i)_{1 \le i \le n}$ is a vector of variables distributed according to the geometric law. and I is the function integer part.

We were obliged to use the integer part because $(x_i)_{1 \le i \le n}$ is a vector of integer and $(1 + \varepsilon_i \pm (1-p).(x_{i-1} - 1)$ was not an integer type.

We also added 1 (in red) to be sure the number of week between two refunds is at least one: it is not consistent to reality a person refund before one week. And, we took away 1 (in green) in order not to make our time of refunds explode.

## b) Model with positive correlation

In the case of positive correlation, we don't need any more the plus 1 to be sure the number of week between two refunds is at least one. So, the mathematical model can be written:

$$\begin{cases} x_1 = \varepsilon_1 \\ x_i = I\left(\varepsilon_i + \dfrac{(1-p)}{10}(x_{i-1} - 1)\right) \end{cases}$$

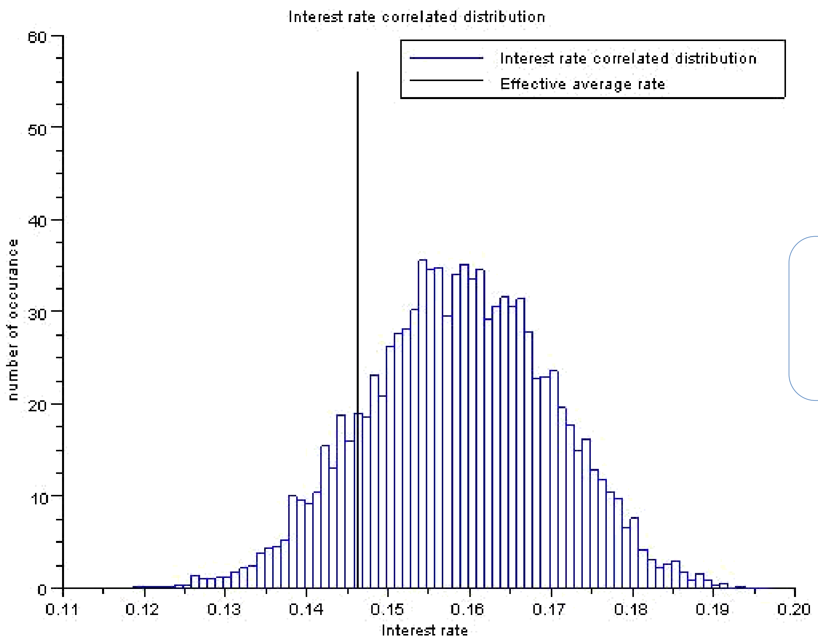
where $(x_i)_{1 \le i \le n}$ is the vector which indicate the number of weeks between two refunds. $(\varepsilon_i)_{1 \le i \le n}$ is a vector of variables distributed according to the geometric law. and $I$ is the function integer part.

## 2. Results

We simulated lots of correlated distributions with parameter p=0.8, calculated the rate for each distribution and finally calculated the average of these rates.

We got these following histograms for 5,000 simulations with the same parameters values we used on part II. We weren't able to get results for more simulations. Scilab bugs: it has problems to find solutions for very high degree polynomials.

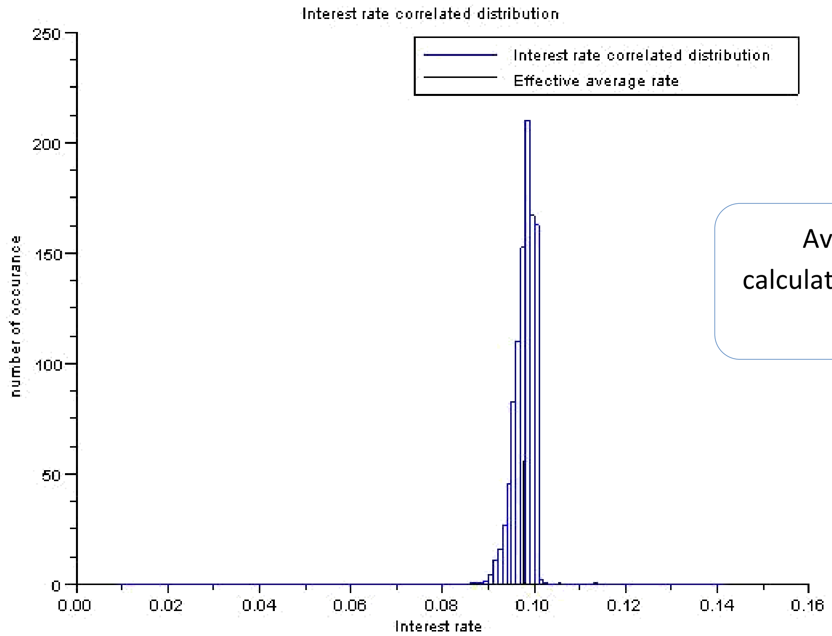
### a) Positive correlation results




Average of the calculated interest rates: 15.87454%

FIGURE 5 : INTEREST RATES FOR 5,000 POSITIVELY CORRELATED DISTRIBUTIONS

## b) Negative correlation results




Average of the calculated interest rates: 9.7864%

FIGURE 6 : INTEREST RATES FOR 5,000 NEGATIVELY CORRELATED DISTRIBUTIONS

# 3. Analysis of the results

In the case of positive correlation, the average of the calculated interest rates is higher than in the case of negative correlation. This result seems very normal, because in the case of positive correlation, people will spend more time to refund money. And, because one euro today is not worth one euro tomorrow, the lender makes a rate inferior to the initial one and vice versa.

Now, if we compare the average of the calculated interest rates in case of positive correlation with the average of the calculated interest rates for a simple geometric distribution with lateness (15.86641%), we can see that they are quite the same.

Whereas, if we do the same thing but with the negative correlation, the two average rates are very different from each other. It is not necessarily a bad model given that we don't have access to statistics that show what really happen with microcredits.

# IV- Consumer credit

In this last part, we wanted to compare consumer credit with microcredits. To do that, we had to build a consumer credit model.

## 1. Consumer credit model

In this model, we assumed that borrowers would have lateness penalties. Indeed, each time he couldn't payback one week, he had to pay back his due the following week increased by 4% (in addition to the normal payback).

To compare the results with the results obtained with the Yunus model, we assumed that borrowers loaned €1000 and had to refund €22 each month.

So the amounts to be refund are:

$$\begin{cases} p_1 = 22, & if\ the\ borrower\ don'thave\ lateness \\ p_2 = 22 + 22 + 22 * 4\%, & if\ the\ borrower\ is\ one\ week\ late \\ p_3 = 22 + 22 + 22 + 22 * 4\% * 2 + 22 * (4\%)^2, & if\ the\ borrower\ is\ two\ weeks\ late \\ \vdots & \vdots \\ p_n = 22.\left(n + \sum_{k=1}^{n-1} (n-k).(4\%)^k\right), & if\ the\ borrower\ is\ n\ weeks\ late\ (n \neq 1) \end{cases}$$

## 2. Simulation

Firstly, we decided to calculate the applied rates assuming that the borrower didn't be late in any payback. We found a rate of 19.74%, it means the same as the one found with the Yunus model in the part I. This result is completely coherent because the borrower hadn't any lateness in his payback, so he hadn't to pay any lateness penalty.

Then we computed with Scilab lots of geometric distributions and plotted as before the histogram of the calculated rates:

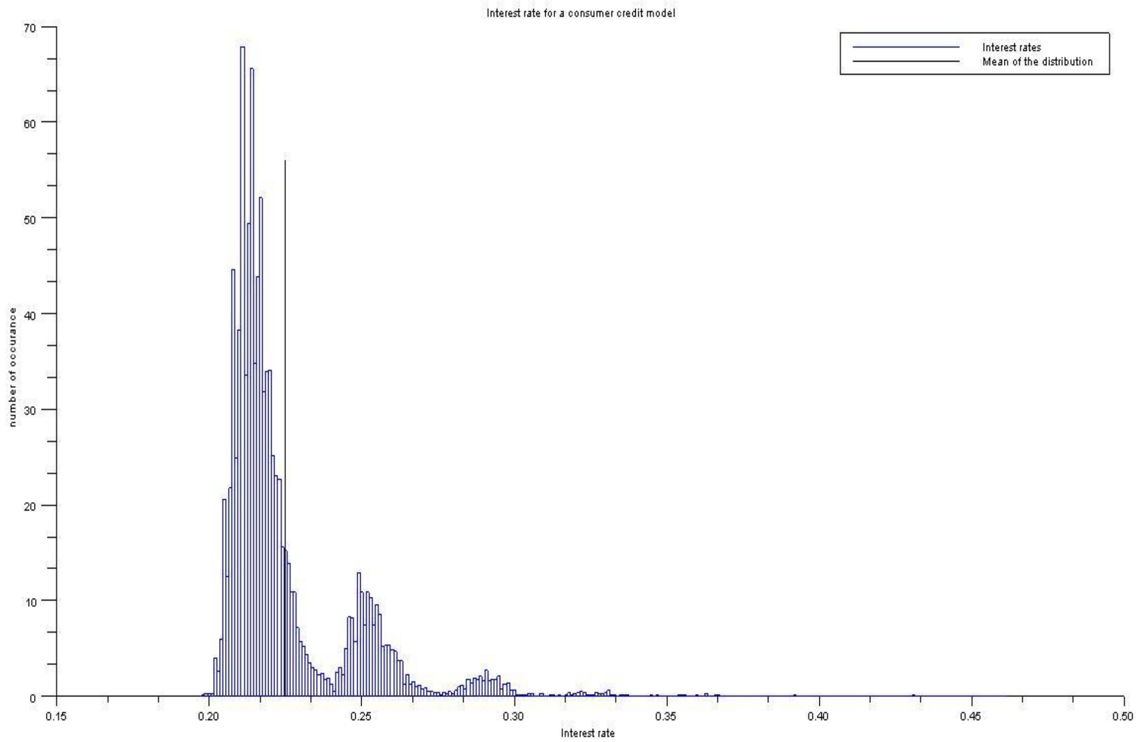



FIGURE 7 : INTEREST RATES FOR A CONSUMER CREDIT MODEL

We can see on this graph that the average of the calculated rate (the black line) is higher than the rate applied for a person who is never late. Moreover we can see that interest rates can reach more than 35% for bad debtors.

But this model is not a quite realistic consumer credit system, because the interests are very high (even if borrowers were not late in any payback). So, to be as realistic as we could, we ask information to a famous consumer credit organisation: Cetelem.

## 3. Cetelem model

Cetelem suggest loaning us €1000 and we had to refund €103 each month. We equally increase the payback probability because borrowers must refund each month (no more each week), so it is quite rare that someone can't refund his due for more than 3 months. Otherwise the interest would reach values unrealistic (as 50% and more...).

As before we first calculated the applied interest rate in the case where there is no lateness. We found a rate of 6.48%. The consumer credit organisation announced a rate of 6.5%.

Then we interested on simulating cases where borrowers can be late on some paybacks. We use, one again, a geometric distribution to describe this model. As usual we plotted our results on a histogram:

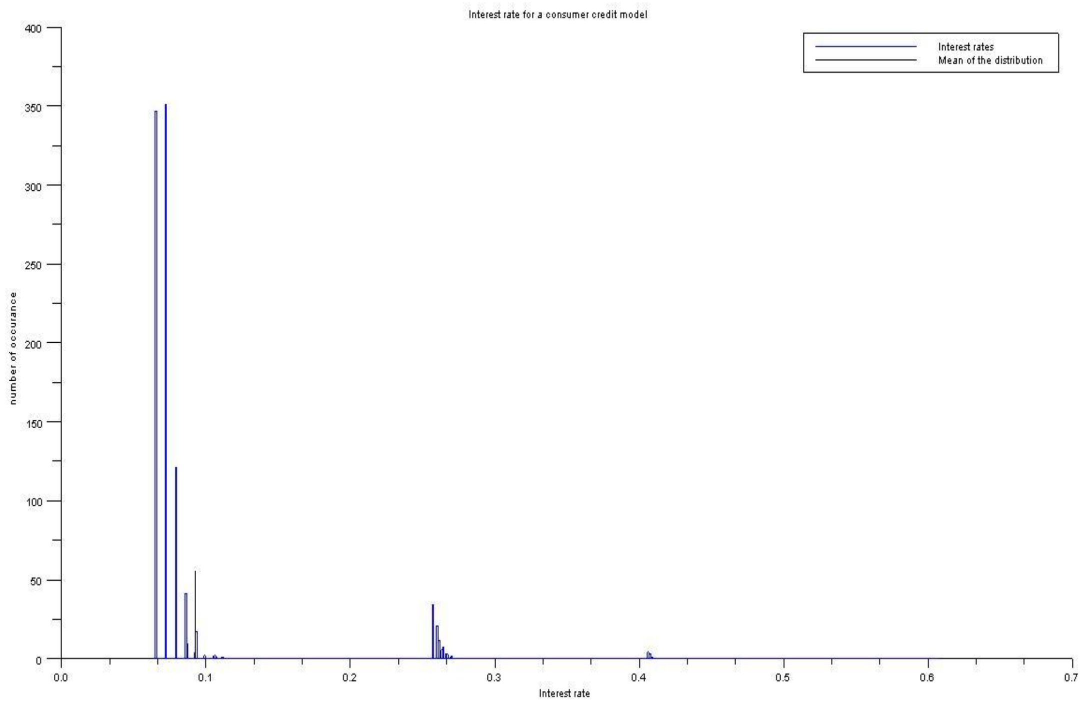



FIGURE 8 : INTEREST RATES FOR THE CETELEM MODEL

On this graph we can see that results are split into three distinct parts.

- In the first one (between 6.5% and 12%) where there are the majority of people: it represent those who payback on time or with some few lateness their loan.
- On the second part (around 26%), there are the people who have problems to refund their loan. You can see that their interest rate become to be high.
- On the third part (near to 40%), there are the people who have unfortunately serious problem to refund their loan. You have to notice that they have to refund approximately one and a half sum of what they landed which is huge. Those people are certainly on financial troubles (because they can't reimburse) and credit consumer companies dragged them in more poverty they already are by charged so high rates.

This spreading out shows interest rate for consumer credits are not linear: as soon as you have lateness on your refunds those companies increase their rate considerably.

# CONCLUSION

Through this project, we have seen a main characteristic of microcredits. In comparison with credit interest rates charged in developed countries, microcredits apply very high interest rates: in France, they are approximately of 3% in contrast to the 20% in the Yunus model.

But this interest rate is lower when people are late in their payments. These are the case of the geometric distribution, the negative and positive correlation. But the rate varies with each situation: it is lower on the model of lateness with positive correlation than on the geometric distribution or on the negative correlation. Indeed, it varies with the number of weeks in total for paying off the entire loan. The more weeks to refund is, the lower is the rate.

This phenomenon can be explained thanks to the fact that one euro today isn't worth a euro tomorrow. So, according to capitalization, if a person doesn't reimburse on time, his loaner will lose money: that is why the loaner charge a lower rate.

Nevertheless, we can say globally that the rates are very high. They aren't inferior to 9%.

If consumer credit companies wanted to do microcredit with the same rules they do in developed countries. The average interest rate really charged will increase. It will be approximately of 22.5%. This is not a good thing for them because they are already poor. We don't have to forget that microcredit has a social dimension: it was created to help people to exit poverty, not for making profits on them and to impoverish them more than they already are. So, if consumer credit companies wanted to really do microcredit they have to change their rules.

To conclude, this project was a great opportunity to link our programming skills to the study of a micro-finance model. Moreover, not only this project was very interesting for all of us, but also it could be very useful for other people who wanted to use our Scilab code later. Besides, it was really instructive for us to work in collaboration with Mr. Diener and Mauk Pheakdei, and we deeply thank them for having provided assistance and good advice.

# REFERENCES

**Books**

M.Yunus, A. Jolis, *Vers un monde sans pauvreté*, J.-C. Lattès (1997)

M.Yunus, *A world without poverty*, Public Affairs (2008)

**Lectures**

M. Diener, F. Diener, *Modèles mathématiques discrets pour la finance et l'assurance*, 2010

J.F. Collet, *Probabilités et Statistiques 2*, 2009

**Online Sources**

*Wikipedia*, free encyclopaedia, [http://en.wikipedia.org]

G. Labrune, *Le microcrédit : une opportunité économique et sociale ?*, Economical, social and environemental concil. [http://lesrapports.ladocumentationfrancaise.fr/BRP/104000271/0000.pdf].

# ANNEXE

```
/////////////////////////////////
//        Model data        //
/////////////////////////////////
clf();
somme = 1000; //loaned amount
remboursement = 22; // amount to pay back each week
n = 50; // number of pay back
e=52; // sample (week, month,year,...)
p = 0.8; //pay back probability

//////////////////////////////////////////////////////////
//        Calculation of the polynom solutions        //
//////////////////////////////////////////////////////////
funcprot(0);
// Calculate all the polynom solutions with the Yunus model
function [sols]=solutions()
  q=poly(0,"q");
  [sols]=roots(remboursement*q^(n+1)-(somme+remboursement)*q+somme); //
endfunction
//Test for this function
sols=solutions();
// Function that select the right solution among all the solutions (it is the real one between 0 and 1)
// sols is the solutions vector
function[solution]=solutionChoisie(sols)
  solution=[];
  imgSols = imag(sols); //imaginary part of the solutions
  for i=1:length(imgSols),
```

```
  if ( imgSols(i)==0 & real(sols(i))>0 & real(sols(i))<1-0.00000001 ) then //pick the real solution
between 0 and 1
        solution=[solution,sols(i)];
  end,
 end,
 solution = real(solution);
endfunction,


//Test for this function
sol=solutionChoisie(sols);
///////////////////////////////////////
//        Rate calculation        //
///////////////////////////////////////
// Calculation of the rate (case where there is no lateness)
function [r]=taux()
 sols = solutions();
 solution = solutionChoisie(sols);
 r = -e*log(solution);
endfunction
//Test for this function
r = taux();
//disp() print the message or the element between the parentesis
disp("*****   Rate calculation example  *****");
disp("Applied rate (case where there is no lateness)");
disp(r);
```

```
/////////////////////////////////////////////////////
//        Effective average rate calculation       //
/////////////////////////////////////////////////////
// Calculate the effective average rate in two different ways
// The paramater p is the payback probability
function[r,r2]=taux_effectif_moyen(p)
  sols = solutions();
  y=solutionChoisie(sols);
  r=e*log((p+y*(1-p))/y);
  r2=e*p*(1/y-1);
endfunction
//Plotting of the effective average rate according to the payback probability
p1=[0:0.02:1];
[taux,r2]=taux_effectif_moyen(p1);
xset("window",0) //window in which the graph will be plot
xtitle("Effective average rate calculation depending on the payback probability"); //title of the window
xlabel("Payback probability"); // legend of the x axis
ylabel("Effective average rate"); //legend of the y axis
plot(p1,taux,'r'); // function which plot the calculated taux effectif moyen
plot(p1,r2,"+"); // function which plot the theoric taux effectif moyen
legend("Calculated effective average rate","Theoritical effective average rate",2); //legend of the graph
////////////////////////////////////////////////////////////
//        Simulation of a geometric distribution         //
////////////////////////////////////////////////////////////
// simulate a value according to the geometric law
function x = randGeometrique()
  y = rand(); // rand()=function which simulate equally a number between 0 and 1
  if (y<=p) then x = 1;
```

```
  else
    k = log(1-y)/log(1-p);
    x = ceil(k);
  end,
endfunction,
// simulate a vector of value chosen according to the geometric law
// the parameter nb is the length of the wanted vector
function x = vecteurGeom(nb)
  x=[];
  for i=1:nb
    x(i) = randGeometrique(),
  end,
endfunction,
//////////////////////////////////////////////////////////////////////
//        Rate calculation with a geometric distribution        //
//////////////////////////////////////////////////////////////////////
// function which calculate the rate for a geometric distribution
// the parameter x is a vecteur sistributed according to the geometric law
// (use the function vecteurGeom to get an x)
function [r]=tauxSim(x)
  q=poly(0,"q"); //declaration of q as a variable
  y=q^x(length(x));
  for i=length(x)-1:-1:1
    y=q^x(i)*(1+y);
  end
  y=remboursement*y-somme;
  sols = roots(y);
  solution = solutionChoisie(sols);
  r = -e*log(solution);
endfunction
```

```
// Test of the previous function
disp("");
disp("*****   Distribution example   *****");
vect = vecteurGeom(n)';
s=sum(vect);
disp("Number of weeks to pay back");
disp(s);
rSim = tauxSim(vect);
disp("Interest rate for this simulation : ");
disp(rSim);
//////////////////////////////////////////////////////////////////////
//         Histogram plotting for geometric simulations         //
//////////////////////////////////////////////////////////////////////
//calculate lots of geometric distribution and the associated rate
t=[];
for i=1:1000
  x=vecteurGeom(n);
  r=tauxSim(x);
  t=[t,r];
end
// Plotting of these rates with an histogram
// (Figure n°1)
disp("");
disp("*****   Lots of distribution simulation   *****");
xset("window",1);
clf();
classe=min(t):0.001:max(t);
rb=taux_effectif_moyen(p)
disp("Effective average rate  : ");
disp(rb);
```

```
effectiveAverageRate=rb*ones(n,1);

y=0:1:n-1;

meanAverage = mean(t)*ones(n,1);

disp("Average of the calculated interest rate  : ");

disp(mean(t));

xtitle("Interest rate distribution")

xlabel("Interest rate");

ylabel("number of occurance");

histplot(classe,t, style=2);

plot(effectiveAverageRate,y,'r');

plot(meanAverage,y,"black");

ecartT = sqrt(variance(t));

normal = 1/(ecartT*sqrt(2*%pi))*exp((-1/2)*((classe-mean(t))/ecartT)^2);

plot(classe, normal,color=13);

legend("Interest rate distribution","Effective average rate","Mean of the distribution","Gaussian
law",2);

 ///////////////////////////////////////////////////////

 //        Testing of normal law properties        //

///////////////////////////////////////////////////////

disp("Third moment (must be equal to 0 to be a normal law) : ");

disp(moment(t,3));

disp("Kurtosis (must be equal to 3 to be a normal law) : ");

disp(moment(t,4)/(moment(t,2)^2));

//////////////////////////////////////////////////////////////////////

//        Calculation of the probability of a d-default          //

//////////////////////////////////////////////////////////////////////

// Calculate the probabilty to have a d-default

// The parameter p is the payback probability

// The parameter d is the "d" of "d-default"

function pi = default(prob,d)
```

```
  pi = (1-prob)^d;
endfunction
// Plotting of the probability to have a default according to the payback probability
// (Figure n°2)
pVar=[0:0.001:1];
xset("window",2);
clf();
xtitle("Default in function of the probability");
xlabel("Pay back probability");
ylabel("Probability of a default");
// uncomment the following line to have the 5% level on the graph
//plot(pVar,[default(pVar,2)' default(pVar,3)' default(pVar,5)' default(pVar,9)'
0.05*ones(1,length(pVar))']);
plot(pVar,[default(pVar,2)' default(pVar,3)' default(pVar,5)' default(pVar,9)']);
legend("2-default","3-default","5-default","9-default");
//xs2gif(2,"DefaultProbability")
////////////////////////////////////////////////////////////////////////////////
//        Calculation of the pay back probabilty to have a default of 5%        //
////////////////////////////////////////////////////////////////////////////////
// Calculate the payback probability to have a probability of def% to have a d-default
// The parameter def is the wanted d-default probability
// The parameter d is the "d" of "d-default"
function prob = payBack(def,d)
  prob = 1-(def)^(1/d)
endfunction,
disp("");
disp("*****   D-default   *****");
prob = payBack(0.05,5)
disp("Pay back probability to have a 2-default of 5% : ");
disp(prob)
```

```
rateDefault = taux_effectif_moyen(prob)
disp("Effective average rate for this probability : ");
disp(rateDefault);
////////////////////////////////
////        Cetelem        //
//////////////////////////////
remboursementC = 103; //amount to pay back every month
sommeC = 1000;  //loaned amount
period = 12; //period to pay back :month
p=0.7 // probability to pay back on time
//function to simulate rate in the case of Cetelem (consumer credit companie)
// When you don't refund on time, they charge a penality rate on what you forget to pay.
// So when you want to refund, you have to pay what you forget to pay plus the penalities.
function [r]=reportSC()
 q=poly(0,"q");
 t = 0.01; //penality for not paying back on times
 x=vecteurGeom(period);
 y=0;
 refundedWeeks = 0;
 i=1;
 //Building of the polynomial
 while (refundedWeeks<10), // 10 because we wanted to refund during 10 month no more
  if (x(i)==1) then
   y = y+remboursementC*q^(sum(x(1:i)));
  else
   rembR = remboursementC;
   for k=1:(x(i)-1)
    rembR = remboursementC + (1+t)*rembR;
   end,
   y = y+rembR*q^(sum(x(1:i)));
```

```
    end
    refundedWeeks = refundedWeeks + x(i);
    i = i+1;
  end
  y=y-sommeC;
  //find the rate
  sols=roots(y);
  solution = solutionChoisie(sols);
  r = -period*log(solution);
endfunction
//Test for this function
disp("");
disp("*****   Consumer credit   *****");
cetelem=reportSC();
disp("Calculated rate");
disp(cetelem);
//////////////////////////////////////////////////////////////////////
//        Histogram plotting for consumer credit model          //
//////////////////////////////////////////////////////////////////////
//function calculate several rates in case of Cetelem
consumerRate=[];
for i=1:10000
  consumerRate = [consumerRate,reportSC()];
end,
//Building of a Histogramm in the case of Cetelem
xset("window",4);
clf();
classe=min(consumerRate):0.001:max(consumerRate);
rb=taux_effectif_moyen(p)
effectiveAverageRate=rb*ones(1.15*n,1);
```

```
yc=0:1:1.15*(n-1);
meanAverage= mean(consumerRate)*ones(1.15*n,1);
disp("Average of the calculated interest rates  : ");
disp(mean(consumerRate));
xtitle("Interest rate for a consumer credit model")
xlabel("Interest rate");
ylabel("number of occurance");
histplot(classe,consumerRate, style=2);
plot(meanAverage,yc,"black");
legend("Interest rates","Mean of the distribution",1);
 x=vecteurG
```