

# Mathematics of Finance Project 1

Renee Prager and M.K.

September 2025

# 1 Amortized Loan Formula

Consider a loan of principal  $P$ . The loan is to be repaid in  $T$  years with  $m$  fixed payments per year. The annual interest rate is  $r$  and is compounded periodically based on payment frequency (i.e., per-payment rate is  $r/m$ ). Here we ignore the day-count complications and assume payments are made at the end of each payment period.

## 1.1

Derive the installment amount  $C$  as a function of  $P$ ,  $T$ ,  $m$ , and  $r$ .

In class we found that for an amortized loan, the loan installment equation is

$$C = \frac{Pr}{1 - (1 + r)^{-T}},$$

derived from the present value formula

$$P = C \frac{1 - (1 + r)^{-T}}{r}.$$

Here we are told that the interest per period is  $\frac{r}{m}$  and the total number of payments is  $mT$ , so if we plug them into the equation we get

$$C = \frac{P \left( \frac{r}{m} \right)}{1 - \left( 1 + \frac{r}{m} \right)^{-mT}}.$$

## 1.2

For each payment, calculate the amount that goes into principal and interest, respectively.

The interest portion ( $I_k$ ) is the interest rate times the balance before payment ( $B_{k-1}$ ) for payment  $k$ :

$$I_k = \frac{r}{m} B_{k-1}.$$

The principal portion is the fixed payment at time  $k$  ( $C$ ) minus what was given as interest:

$$P_k = C - I_k = C - \frac{r}{m} B_{k-1}.$$

## 1.3

Write a Python function that takes in arguments  $P$ ,  $T$ ,  $m$ , and  $r$ , and outputs (a)  $C$  (b) the amount that goes into principal and interest for each payment. Python code in Appendix

## 1.4

An adjustable-rate mortgage (ARM) is a home loan with an interest rate that is fixed for an initial period and then fluctuates based on market conditions. With  $P = 1,000,000$ ,  $T = 30$  (years), and  $m = 12$ ,

(a) For the first 5 years the bank asks for an interest rate  $r_1 = 6\%$ , calculate the amount of each payment  $C_1$ .

(b) For the next year the bank asks for an interest rate  $r_2 = 8\%$ , calculate the amount of each payment  $C_2$ .

(c) For the next year the bank asks for an interest rate  $r_3 = 10\%$ , calculate the amount of each payment  $C_3$ .

(d) For the rest of the years the bank asks for an interest rate  $r_4 = 12\%$ , calculate the amount of each payment  $C_4$ .

Given:

$$P = \$1,000,000, \quad T = 30, \quad m = 12$$

**(a) First 5 years: Interest rate = 6%**

$$\frac{r}{m} = \frac{0.06}{12} = 0.005, \quad mT = 30 \cdot 12 = 360$$

$$C = \frac{1,000,000(0.005)}{1 - (1.005)^{-360}} \approx \$5,995$$

**(b) Year 6: Interest rate = 8%**

There are now 25 years left, so

$$\frac{r}{m} = \frac{0.08}{12}, \quad mT = 25 \cdot 12 = 300$$

The loan balance at this point can be found using the equation

$$B = P \frac{(1+i)^n - (1+i)^p}{(1+i)^n - 1},$$

where

$$n = 360, \quad p = 60, \quad i = 0.005.$$

So the balance is

$$B \approx \$930,600.$$

Now we compute the new payment:

$$C = \frac{930,600 \left( \frac{0.08}{12} \right)}{1 - \left( 1 + \frac{0.08}{12} \right)^{-300}} \approx \$7,172.$$

(c) **Year 7: Interest rate = 10%**

$$\frac{r}{m} = \frac{0.10}{12} = 0.008333, \quad mT = 24 \cdot 12 = 288$$

The loan balance is

$$B \approx \$918,529.$$

The new installment is

$$C = \frac{918,529(0.008333)}{1 - (1.008333)^{-288}} \approx \$8,454.$$

(d) **Remaining Time: Interest rate = 12%**

$$\frac{r}{m} = \frac{0.12}{12} = 0.01, \quad mT = 23 \cdot 12 = 276$$

The remaining balance is

$$B \approx \$908,270.$$

Thus the installment for the remaining loan term is

$$C = \frac{908,270(0.01)}{1 - (1.01)^{-276}} \approx \$9,739.$$

## 1.2

1)

**Proposition 1** states that a contingent claim  $C(T)$  can be replicated by investing in a portfolio  $(\theta_1, \theta_2)$  of the risk-free asset and the underlying risky asset. That is,

$$C(T) = \theta_1 A(T) + \theta_2 S(T).$$

Then,

$$C(0) = \theta_1 A(0) + \theta_2 S(0),$$

otherwise an arbitrage opportunity would arise.

Under a simple market model, a similar argument can be made to derive the fair price of a put option  $P(0)$ .

We know that

$$P(T) = \max(K - S(T), 0).$$

If a put option is replicated so that

$$P(T) = \theta_1 A(T) + \theta_2 S(T),$$

then in order to prevent arbitrage, it must be that

$$P(0) = \theta_1 A(0) + \theta_2 S(0).$$

2)

Using the above proposition, we can price a put option with:

$$\begin{aligned}K &= 19.5, & A(0) &= 100, & A(T) &= 105, \\S(0) &= 20, & S_u(T) &= 22, & S_d(T) &= 19.\end{aligned}$$

We replicate the option.

**Upper state:**

$$\begin{aligned}\theta_1 A(T) + \theta_2 S_u(T) &= \max(K - S_u(T), 0) \\ \theta_1(105) + \theta_2(22) &= \max(19.5 - 22, 0) = 0.\end{aligned}$$

**Lower state:**

$$\begin{aligned}\theta_1 A(T) + \theta_2 S_d(T) &= \max(K - S_d(T), 0) \\ \theta_1(105) + \theta_2(19) &= \max(19.5 - 19, 0) = 0.5.\end{aligned}$$

### Solving the System

Solve the up and down equations together:

$$\begin{aligned}-\theta_2(19) + 0.5 &= -\theta_2(22) \\ -3\theta_2 &= 0.5 \quad \Rightarrow \quad \theta_2 = -\frac{1}{6}.\end{aligned}$$

Substitute into the upper-state equation:

$$\begin{aligned}\theta_1(105) + \left(-\frac{1}{6}\right)(22) &= 0 \\ \theta_1 &= \frac{11}{315}.\end{aligned}$$

### Put Option Price

Plug into the pricing equation:

$$\begin{aligned}P(0) &= \theta_1 A(0) + \theta_2 S(0) \\ P(0) &= \left(\frac{11}{315}\right)(100) + \left(-\frac{1}{6}\right)(20) \approx 0.15873.\end{aligned}$$

## A Python Code

```
# -*- coding: utf-8 -*-  
"""project 1 code
```

*Automatically generated by Colab.*

*Original file is located at  
[https://colab.research.google.com/drive/16Kzo32EH\\_lDU18th9kT2j--bPeosGCB4](https://colab.research.google.com/drive/16Kzo32EH_lDU18th9kT2j--bPeosGCB4)*

*Project 1, Question 3*  
"""

```
# P is the principal of the loan  
P=100  
# T is the lifespan of the loan  
T=30  
# r is the annual rate  
r=0.1  
# m is the number of payments per year  
m=12
```

```
#formula for fixed payment (C)  
P * (1-(1+(r/m))**(-1))/((1+r/m)**(-1)-(1+r/m)**(-T*m-1))
```

```
def my_loan_calculator (principal,life_span,payments_per_year,rate):
```

```
    #compute the discount factor for one period  
    discount_factor= (1+rate/payments_per_year)**(-1)
```

```
    #compute payment per period  
    amount_per_payment= principal*(1-discount_factor)/(discount_factor-discount_factor**(life
```

```
    return amount_per_payment
```

```
#calculate each payment using given values  
my_loan_calculator(100,20,12,0.05)
```

```
# Loan Schedule
```

```
def loan_schedule(principal,life_span,payments_per_year,rate):
```

```
    #loan schedule calculated using loan calculator  
    payment_amount=my_loan_calculator(principal,life_span,payments_per_year,rate)
```

```
    #start with initial loan balance  
    balance=principal
```

```

#loop for all payments (here 360)
for i in range(life_span*payments_per_year):

    #calculate interest owed this payment
    accrued_interest=balance*rate/payments_per_year

    #calculate principal portion owed this payment
    principal_paid = payment_amount - accrued_interest

    print(f"At time {i+1}, the balance before payment is {balance}, the principal\
paid is {principal_paid}, the accrued interest is {accrued_interest}")

    #update balance
    balance=balance-(payment_amount-accrued_interest)

    print(f"At time {i+1}, the final balance is {balance}")

    return balance

loan_schedule(100,10,2,0.05)

```