# Week 4 Module 3: Loan Amortization

CSCI E-5a: Programming in R

Let's clear the global computing environment:

```
rm( list = ls() )
```

## Module Overview and Learning Outcomes

Hello! And welcome to Module 3: Amortizing a Loan.

In this module, we'll explore the concept of amortizing a loan, which means to pay off a loan with regular payments.

- In section 1, we'll see how to calculate the amount of the regular payment.
- In section 2, we'll see how this payment amount will pay off the loan balance.
- In section 3, we'll explore how we can use iteration to automate the calculation of an amortization schedule.

When you've completed this module, you should be able to:

- Calculate the payment amount for a loan, given the initial loan amount, the number of payments, and the interest rate.
- Construct an amortization schedule for the repayment of the loan.

There are no new built-in R functions in this module.

All right! Let's start by learning how to calculate the amount of an annual loan payment.

## Section 1: Loan Payment

> **Main Idea:** *We can calculate the amount of the annual loan payment*

In this section, we'll see how to calculate the amount of the regular loan payment.

Suppose Obi takes out a loan for a value of $1,000$ dollars, and agrees to repay the loan by making 3 annual payments, the first of which occurs one year after the loan is issued.

The annual interest rate for this loan is 5%.

How much should the amount of each payment be, in order to fully pay off the loan in 3 payments?

More generally, suppose someone takes out a loan of size $L$, and agrees to pay the loan back by making $n$ annual payments, the first of which occurs one year after the loan is issued.

The annual interest rate for the loan is denoted $r$.

What is the amount of the annual payment?

For our example, we have these values for the variables:

| Variable | Value |
|---|---|
| $L$ | 1,000 |
| $n$ | 3 |
| $r$ | 0.05 |

Remember – when we write a percentage such as "5%", that really means the number 0.05.

From elementary finance, there is a simple formula to calculate the annual payment amount.

First, we calculate a *discount factor*, denoted by $\nu$:

$$\nu = \frac{1}{1+r}$$

Then the annual payment amount, denoted by $P$, is:

$$P = \frac{r \cdot L}{1-\nu^n}$$

Let's do this for our example.

First, we calculate the discount factor:

$$\nu = \frac{1}{1+r} = \frac{1}{1+0.05} = 0.952381$$

Then the annual payment amount, denoted by $P$, is:

$$P = \frac{r \cdot L}{1-\nu^n} = \frac{0.05 \times 1,000}{1-0.952381^3} = 367.21$$

Let's do this in R.

First, we can define some variables:

```
loan.amount <- 1000

number.of.payments <- 3

annual.interest.rate <- 0.05
```

Now we can calculate the discount factor:

```
discount.factor <- 1/(1 + annual.interest.rate)

cat(
    "Discount factor:",
    formatC(
        discount.factor,
        format = "f",
        digits = 6
    )
)
```

```
## Discount factor: 0.952381
```

Now we can calculate the annual payment amount:

```
annual.payment.amount <-
    loan.amount * annual.interest.rate /
    (1 - discount.factor^number.of.payments)

cat(
    "Annual payment amount:",
    formatC(
        annual.payment.amount,
        format = "f",
        digits = 2
    )
)
```

```
## Annual payment amount: 367.21
```

So that's how to calculate the annual loan payment amount.

Now let's see how this payment amount pays off the loan balance.

### Exercise 4.1: Annual Payment Amount

Ashley takes out a loan for $10,000 and pays it back in 3 annual installments, the first one occurring one year after the loan is issued.

The annual interest rate is 4%.

Save the loan amount and the annual interest rate in variables, but call the variables something different from `loan.amount` and `annual.interest.rate`, because we're using those variables for our text example.

Calculate the annual payment amount.

Save this annual payment amount in a variable, but call the variable something different from `annual.payment.amount`, because we're using that for our text example.

**Solution**

## Section 2: Amortizing a Loan

> **Main Idea:** *We can amortize a loan*

In this section, we'll see how this payment amount will pay off the loan balance.

OK, what exactly does it mean to say that if Obi makes 3 annual payments of $367.21 that he will pay off the loan?

Let's work this out by hand.

At the start of the first year, the loan balance is just the loan amount.

```
loan.balance <- loan.amount
```

At the end of the year, the loan balance has accrued 5% interest.

Let's update the loan balance variable at the end of the year before the first payment:

```
loan.balance <-
    loan.balance +
    (annual.interest.rate * loan.balance)

cat(
    "Loan balance at end of year 1, before payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 1, before payment: 1050.00
```

This represents the loan balance at the end of the year *before* Obi makes the first annual payment.

When Obi makes the annual payment at the end of year 1, the loan balance decreases by the amount of the payment:

```
loan.balance <-
    loan.balance - annual.payment.amount

cat(
    "Loan balance at end of year 1, after payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 1, after payment: 682.79
```

The loan balance at the end of year 1 after the payment is made now becomes the loan balance at the beginning of year 2:

```
cat(
    "Loan balance at beginning of year 2:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at beginning of year 2: 682.79
```

This loan balance accrues 5% interest over the duration of the second year.

The loan balance at the end of the second year, before Obi makes the annual payment, is:

```
loan.balance <-
    loan.balance +
    (loan.balance * annual.interest.rate)

cat(
    "Loan balance at end of year 2, before payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 2, before payment: 716.93
```

When Obi makes the annual payment at the end of the second year, the loan balance decreases by the amount of the payment:

```
loan.balance <-
    loan.balance - annual.payment.amount

cat(
    "Loan balance at end of year 2, after payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 2, after payment: 349.72
```

The loan balance at the end of year 2 after the payment is made now becomes the loan balance at the beginning of year 3:

```
cat(
    "Loan balance at beginning of year 3:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at beginning of year 3: 349.72
```

The loan balance at the end of the third year, before Obi makes the annual payment, is:

```
loan.balance <-
    loan.balance +
    (loan.balance * annual.interest.rate)

cat(
    "Loan balance at end of year 3, before payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 3, before payment: 367.21
```

When Obi makes the annual payment at the end of the third year, the loan balance is 0.00 and the loan is paid off:

```
loan.balance <-
    loan.balance - annual.payment.amount

cat(
    "Loan balance at end of year 3, after payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 3, after payment: 0.00
```

So that's what it means to say that Obi will pay off the loan in 3 years with an annual payment of $367.21.

This process of paying off the loan with regular payments is called *amortizing* the loan, and the actual sequence of balances is called the *amortization schedule*.

So that's how the annual loan payment amount amortizes the loan.

Now let's think about how we can use iteration to automate this calculation.

## Exercise 4.2: Amortizing a loan

Ashley takes out a loan for $10,000 and pays it back in 3 annual installments, the first one occurring one year after the loan is issued.

The interest rate is 4%.

Calculate the amortization schedule.

**Solution**

## Section 3: Amortization and Iteration

**Main Idea:** *Let's automate this process*

In this section, we'll explore how we can use iteration to automate the calculation of an amortization schedule.

You might have noticed that this calculation was very repetitive.

The loan balance at the end of the first year becomes the loan balance at the beginning of the second year.

Then the loan balance at the end of the second year, after the payment, becomes the loan balance at the beginning of the third year.

This process continues until the loan is fully paid off.

It's hard to do this calculation using vectorized operations, because it doesn't consist of element-wise or component-wise calculations.

Instead, it consists of continually updating the loan balance, and we perform a sequence of calculations.

Thus, this process lends itself to iteration.

We initialize the loan balance with the initial loan amount.

Then we can construct a `for` loop that iterates over the years of the loan.

The first iteration represents the first year, and we can perform all the updating operations using just the interest rate and the annual payment amount, which we've calculated previously.

The second iteration represents the second year, and again we can peform all the updating operations using just the interest rate and the annual payment amount.

We can continue in this way for the duration of the loan, and at the end of the loan you should have an outstanding loan balance of 0.00.

You'll get to work this out in your problem set.

So that's how to use iteration to automate the process of constructing an amortization schedule.

Now let's review what we've learned in this module.

## Module Review

In this module, we explored the concept of amortizing a loan, which means to pay off a loan with regular payments.

- In section 1, we saw how to calculate the amount of the regular payment.
- In section 2, we saw how this payment amount will pay off the loan balance.
- In section 3, we explored how we can use iteration to automate the calculation of an amortization schedule .

Not that you've completed this module, you should be able to:

- Calculate the payment amount for a loan, given the initial loan amount, the number of payments, and the interest rate.
- Construct an amortization schedule for the repayment of the loan.

There were no new built-in R functions in this module.

All right! That's it for Module 3: Amortizing a Loan.

Now let's move on to Module 4: Moving Average.

# Solutions to the Exercises

## Exercise 4.1: Annual Payment Amount

Ashley takes out a loan for $10,000 and pays it back in 3 annual installments, the first one occurring one year after the loan is issued.

The annual interest rate is 4%.

Save the loan amount and the annual interest rate in variable, but call the variables something different from `loan.amount` and `annual.interest.rate`, because we're using those variables for our text example.

Calculate the annual payment amount.

Save this annual payment amount in a variable, but call the variable something different from `annual.payment.amount`, because we're using that for our text example.

**Solution**

First, we can define some variables:

```
ashley.loan.amount <- 10000

number.of.payments <- 3

ashley.annual.interest.rate <- 0.04
```

Now we can calculate the discount factor:

```
discount.factor <- 1/(1 + annual.interest.rate)

cat(
    "Discount factor:",
    formatC(
        discount.factor,
        format = "f",
        digits = 6
    )
)
```

```
## Discount factor: 0.952381
```

Now we can calculate the annual payment amount:

```
ashley.annual.payment.amount <-
    ashley.loan.amount *
    annual.interest.rate /
    (1 - discount.factor^number.of.payments)

cat(
    "Annual payment amount:",
    formatC(
        ashley.annual.payment.amount,
        format = "f",
        big.mark = ",",
```

```
        digits = 2
    )
)
```

```
## Annual payment amount: 3,672.09
```

### Exercise 4.2: Amortizing a loan

Ashley takes out a loan for $10,000 and pays it back in 3 annual installments, the first one occurring one year after the loan is issued.

The interest rate is 4%.

You've already calculated the annual payment amount for Ashley in Exercise 3.1.

Now construct an amortization schedule to show that this annual payment amount will indeed pay off her initial loan in 3 years.

**Solution**

At the start of the first year, Ashley has a loan for $10,000, with an interest rate of 4.00%.

```
loan.balance <- 1000
```

At the end of the year, the loan balance has accrued 4% interest.

Let's update the loan balance variable:

```
loan.balance <-
    loan.balance +
    (ashley.annual.interest.rate * loan.balance)

cat(
    "Loan balance at end of year 1, before payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 1, before payment: 1040.00
```

This represents the loan balance at the end of the year *before* Ashley makes the first annual payment.

When Ashley makes the annual payment at the end of year 1, the loan balance decreases by the amount of the payment:

```
loan.balance <-
    loan.balance - ashley.annual.payment.amount

cat(
    "Loan balance at end of year 1, after payment:",
    formatC(
        loan.balance,
```

```
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 1, after payment: -2632.09
```

The loan balance at the end of year 1 after the payment is made now becomes the loan balance at the beginning of year 2:

```
cat(
    "Loan balance at beginning of year 2:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at beginning of year 2: -2632.09
```

This loan balance accrues 4% interest over the duration of the second year.

The loan balance at the end of the second year, before Ashley makes the annual payment, is:

```
loan.balance <-
    loan.balance +
    (loan.balance * ashley.annual.interest.rate)

cat(
    "Loan balance at end of year 2, before payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 2, before payment: -2737.37
```

When Obi makes the annual payment at the end of the second year, the loan balance decreases by the amount of the payment:

```
loan.balance <-
    loan.balance - ashley.annual.payment.amount

cat(
    "Loan balance at end of year 2, after payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 2, after payment: -6409.45
```

The loan balance at the end of year 2 after the payment is made now becomes the loan balance at the beginning of year 3:

```
cat(
    "Loan balance at beginning of year 3:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at beginning of year 3: -6409.45
```

The loan balance at the end of the third year, before Ashley makes the annual payment, is:

```
loan.balance <-
    loan.balance +
    (loan.balance * ashley.annual.interest.rate)

cat(
    "Loan balance at end of year 3, before payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 3, before payment: -6665.83
```

When Ashley makes the annual payment at the end of the third year, the loan balance is 0.00 and the loan is paid off:

```
loan.balance <-
    loan.balance - ashley.annual.payment.amount

cat(
    "Loan balance at end of year 3, after payment:",
    formatC(
        loan.balance,
        format = "f",
        digits = 2
    )
)
```

```
## Loan balance at end of year 3, after payment: -10337.92
```