

# Fundamental financial concepts

INTRODUCTION TO FINANCIAL CONCEPTS IN PYTHON



**Dakota Wixom**

Quantitative Finance Analyst

# Course objectives

- The Time Value of Money
- Compound Interest
- Discounting and Projecting Cash Flows
- Making Rational Economic Decisions
- Mortgage Structures
- Interest and Equity
- The Cost of Capital
- Wealth Accumulation

# Calculating Return on Investment (% Gain)

$$\text{Return (\% Gain)} = \frac{v_{t_2} - v_{t_1}}{v_{t_1}} = r$$

- $v_{t_1}$ : The initial value of the investment at time
- $v_{t_2}$ : The final value of the investment at time

# Example

- You invest \$10,000 at time = year 1
- At time = 2, your investment is worth \$11,000

$$\frac{\$11,000 - \$10,000}{\$10,000} * 100 = 10\% \text{ annual return (gain) on y}$$

# Calculating Return on Investment (Dollar Value)

$$v_{t_2} = v_{t_1} * (1 + r)$$

- $v_{t_1}$ : The initial value of the investment at time
- $v_{t_2}$ : The final value of the investment at time
- $r$ : The rate of return of the investment per period t

# Example

- Annual rate of return = 10% = 10/100
- You invest \$10,000 at time = year 1

$$\$10,000 * \left(1 + \frac{10}{100}\right) = \$11,000$$

# Cumulative growth (or depreciation)

- $r$ : The investment's expected rate of return (growth rate)
- $t$ : The lifespan of the investment (time)
- $v_{t_0}$ : The initial value of the investment at time 0

$$\text{Investment Value} = v_{t_0} * (1 + r)^t$$

If the growth rate  $r$  is negative, the investment's value will depreciate (shrink) over time.

# Discount factors

$$df = \frac{1}{(1 + r)^t}$$

$$v = fv * df$$

- $df$ : Discount factor
- $r$ : The rate of depreciation per period  $t$
- $t$ : Time periods
- $v$ : Initial value of the investment
- $fv$ : Future value of the investment



# Compound interest

$$\text{Investment Value} = v_{t_0} * \left(1 + \frac{r}{c}\right)^{t*c}$$

- $r$ : The investment's annual expected rate of return (growth rate)
- $t$ : The lifespan of the investment
- $v_{t_0}$ : The initial value of the investment at time 0
- $c$ : The number of compounding periods per year

# The power of compounding returns

Consider a \$1,000 investment with a 10% annual return, compounded quarterly (every 3 months, 4 times per year):

$$\$1,000 * \left(1 + \frac{0.10}{4}\right)^{1*4} = \$1,103.81$$

Compare this with no compounding:

$$\$1,000 * \left(1 + \frac{0.10}{1}\right)^{1*1} = \$1,100.00$$

Notice the extra *\$3.81* due to the quarterly compounding?

# Exponential growth

Compounded Quarterly Over 30 Years:

$$\$1,000 * \left(1 + \frac{0.10}{4}\right)^{30*4} = \$19,358.15$$

Compounded Annually Over 30 Years:

$$\$1,000 * \left(1 + \frac{0.10}{1}\right)^{30*1} = \$17,449.40$$

Compounding quarterly generates an extra *\$1,908.75 over 30 years*

# Let's practice!

INTRODUCTION TO FINANCIAL CONCEPTS IN PYTHON

# Present and future value

INTRODUCTION TO FINANCIAL CONCEPTS IN PYTHON



**Dakota Wixom**  
Quantitative Finance Analyst

# The non-static value of money

## Situation 1

- **Option A:** \$100 in your pocket today
- **Option B:** \$100 in your pocket tomorrow

## Situation 2

- **Option A:** \$10,000 dollars in your pocket today
- **Option B:** \$10,500 dollars in your pocket one year from now

# Time is money

## Your Options

- **A:** Take the \$10,000, stash it in the bank at 1% interest per year, risk free
- **B:** Invest the \$10,000 in the stock market and earn an average 8% per year
- **C:** Wait 1 year, take the \$10,500 instead

# Comparing future values

- **A:**  $10,000 * (1 + 0.01) = 10,100$  future dollars
- **B:**  $10,000 * (1 + 0.08) = 10,800$  future dollars
- **C:** 10,500 future dollars



# Present value in Python

Calculate the present value of \$100 received 3 years from now at a 1.0% inflation rate.

```
import numpy as np  
np.pv(rate=0.01, nper=3, pmt=0, fv=100)
```

```
-97.05
```

# Future value in Python

Calculate the future value of \$100 invested for 3 years at a 5.0% average annual rate of return.

```
import numpy as np  
np.fv(rate=0.05, nper=3, pmt=0, pv=-100)
```

```
115.76
```

# Let's practice!

INTRODUCTION TO FINANCIAL CONCEPTS IN PYTHON

# Net present value and cash flows

INTRODUCTION TO FINANCIAL CONCEPTS IN PYTHON



**Dakota Wixom**  
Quantitative Finance Analyst

# Cash flows

**Cash flows** are a series of gains or losses from an investment over time.

Year	Project 1 Cash Flows	Project 2 Cash Flows
0	-\$100	\$100
1	\$100	\$100
2	\$125	-\$100
3	\$150	\$200
4	\$175	\$300

Assume a 3% discount rate

Year	Cash Flows	Formula	Present Value
0	-\$100	<code>pv(rate=0.03, nper=0, pmt=0, fv=-100)</code>	-100
1	\$100	<code>pv(rate=0.03, nper=1, pmt=0, fv=100)</code>	97.09
2	\$125	<code>pv(rate=0.03, nper=2, pmt=0, fv=125)</code>	117.82
3	\$150	<code>pv(rate=0.03, nper=3, pmt=0, fv=150)</code>	137.27
4	\$175	<code>pv(rate=0.03, nper=4, pmt=0, fv=175)</code>	155.49

Sum of all present values = 407.67

# Arrays in NumPy

## Example:

```
import numpy as np  
array_1 = np.array([100, 200, 300])  
print(array_1*2)
```

```
[200 400 600]
```

# Net Present Value

## Project 1

```
import numpy as np
np.npv(rate=0.03, values=np.array([-100, 100, 125, 150, 175]))
```

407.67

## Project 2

```
import numpy as np
np.npv(rate=0.03, values=np.array([100, 100, -100, 200, 300]))
```

552.40



# Let's practice!

INTRODUCTION TO FINANCIAL CONCEPTS IN PYTHON