Financial Modeling and Financial Statement Analysis in Python

Paras Ahuja

March 28, 2021

Contents

Preface

This book is for anyone who is interested in using Python for financial modeling and financial statement analysis. I have tried my best to make sure that the book is self-contained. Key financial formulas and some background is presented in the book. The purpose of the book is not to be overly financial but to use Python to analyze financial statements.

Part I

Fundamentals of Finance

Chapter 1

Time Value of Money

Time value of money (TVM) is one of the most important concepts in the entirety of financial knowledge. One must understand the ideas behind the TVM in order to master other financial concepts and theorems. This is because TVM shows up in many different contexts. Even though the goal of this book to instil in its reader the knowledge about how Python can be used to make financial calculations more efficient, it is still worth our time to pay attention to some key concepts.

1.1 Future Value

As finance professionals, one of the most basic problems faced by us involves interest rates and requires us to understand the future value of an investment made today. To bring home the point, we start with the basic scenarios and level up.

Our friend comes to us and asks for a $10,000 loan because they are in a dire need of cash and will repay us in one year. We ask our friend about the interest rate that they are willing to pay us because we will have to forgo our current consumption in order to lend them the $10,000. The friend says that 5% interest rate is acceptable to them. We graciously accept their offer. How much will we get back in a year from now?

This is a trivial scenario and most people will have no trouble stating that the they will get back $10,500 in a year. Generally, the money invested today, present value, henceforth referred to as PV, for one year will grow to $(1+r)$, where r is the interest rate (recall that if interest rate is in percent, we divide it by 100). This scenario deals with single period-investment or loan - the idea is same. Mathematically speaking, we can show that our single-period loan of $10,000 will grow as follows:

$$FV = 10,000 \times \left(1 + \frac{5}{100}\right) = 10,500.$$

The problem rises in complexity as the number of years increase. What if the friend said that they need $10,000 for 5 years at 5%? This is a question of multi-period investment. Now we have questions about whether the interest we get is simple interest or compound interest. In simple interest we only get interest on the original amount lent. In comparison, compound interest gives us interest on interest. Our pay-offs are completely different. Here is what our pay-offs look like:

Table 1.1: Future Value of $10,000

Year	PV	Simple Interest	Compound Interest	Total	FV
1	$10,000.00	$500	500		
2	10,500.00	500	500		
3	11,025.00	500	500		
4	11,576.25	500	500		
5	12,155.06	500	500		

In Python we program this as:

```
1  def future_value(pv, rate):
2    fv = pv * (1 + rate/100)
3    return fv
```

Let's dissect this code. In order to declare a function in Python we utilize the reserved word `def` followed by the name of the function. In parentheses we have the parameters on which we perform an action. In `line 2` of our code we perform our calculation, and in `line 3` we `return` the result. Note that the words highlighted in green are reserved words. These words are not available for us to use.