Fixed Income Basics

Jianing(Jay) Yao

May 31, 2025

Abstract: fixed income is a detail-driven domain, where even small nuances can have significant implications. While it can sometimes feel tedious, a solid understanding of the terminology is essential for effective communication and accurate analysis. This document provides an overview of commonly used fixed income terms and market jargon to help build that foundation.

1 Day Count and Accrual Calculation

In fixed income, the accumulation of interest over time is a fundamental concept. To quantify this, we require a decimal representation that reflects the fraction of a year elapsed between two dates. Specifically, given T and T', we denote accrual factor by $\tau(T, T'; \mathcal{C})$,

$$\begin{aligned}\tau(T, T'; \mathcal{C}) &= \left.\frac{\textit{Days Elapsed}}{\textit{Year Basis}}\right| \textit{Subject to convention } \mathcal{C} \\ &:= Acc\left(T, T'; \mathcal{C} = \left\{\textit{Business Convention}, \textit{Holiday Convention}, \textit{Accrual Basis}\right\}\right)\end{aligned}$$

We elaborate the calculation of denominator and numerator in this section.

1.1 Numerator – Day Count

Calculating the number of days between dates T and T' can be complicated by the presence of non-business days. The definition of a business day varies by region and depends on the local holiday calendar, which we refer to as the *holiday convention*. For example, July 4th is Independence Day – a holiday in the United State – but it is a regular business day in Canada.

When a date falls on a non-business day (such as a weekend or a public holiday), we need a rule to adjust it to the nearest valid business day. This rule is defined by the *business day convention*, which determines how the date should be rolled. Common business day conventions include:

- *Following (F):* If a date falls on a non-business day, it is **moved forward** to the next business day.

- *Modified Following (MF):* Similar to "Following," but with one key exception: if moving forward takes the date into the next calendar month, the date is instead **moved backward** to the preceding business day.
- *Preceding (P):* If the date falls on a non-business day, it is moved **backward** to the previous business day.

1.2 Denominator – Year Basis

The *accrual basis* defines how time is counted when calculating the denominator of accrual factor. Here are some common accrual bases,

- *Actual/360:* Uses actual days in the period over 360 days. Common in money markets (e.g., LIBOR).
- *Actual/365:* Actual days over 365. Common in UK markets.
- *30/360:* Assumes each month has 30 days, year has 360 days. Common in corporate bonds.

2 Interest Rate

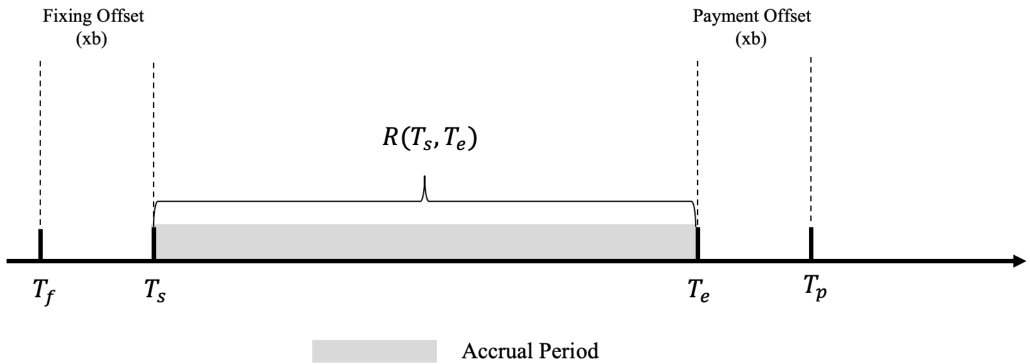
Interest rates can broadly be classified into two main categories: 1) simple interest rates; 2) compounded interest rates. We will discuss each in detail below. Before that, it's useful to understand a few key concepts that apply in both contexts:

- *Spot Rate:* It refers to the current market interest rate for immediate settlement—typically for a transaction that settles "on the spot," meaning today or within a standard settlement period (usually $T+0$ to $T+2$, depending on market convention).
- *Fixing:* It is a published reference rate, typically calculated at a specific time each business day, based on actual or indicative market transactions.
- *Forward Rate:* it is the interest rate applicable to a future period, derived from current market data (such as spot rates or futures prices or swaps). It is not directly observed in the market but inferred from the term structure of interest rates.

While the spot rate and the fixing may coincide at the moment the fixing is published, they serve different purposes. The spot rate reflects the current tradable market rate, whereas the fixing is an official benchmark used for valuation and settlement. Furthermore, fixings are available for both the current day (once published) and historically, whereas the spot rate refers specifically to the present market rate.

2.1 Simple Interest Rates

A *simple interest rate* is an interest rate applied to the principal amount over a specified period, without compounding. This means that interest accrues linearly over time and is not reinvested during the accrual period. To be precise, in *Figure 2.1*,



We have

- T_s denote the accrual start date,
- T_e denote the accrual end date, and
- T_f denote the fixing date

Notice that T_f is usually defined through *fixing offset* from T_s. We denote by $R(T_f; T_s, T_e)$ the annualized rate at which interest is accrued over the accrual period $[T_s, T_e]$. Typically, we would like to value an interest payment associated with R. Although the rate is *fixed-in-advance*, the payment timing can vary:

- *Pay-In-Advance:* settle a payment immediately, and its present value is,

$$V_{\textit{pay-in-adv}}(t_0) = df(t_0, T_s)\tau(T_s, T_e)R(T_f = t_0; T_s, T_e) \tag{1}$$

  In this case, the payment offset is based on T_f rather than T_e. And we assumed that the payment offset equals to the fixing offset; therefore, a discounting factor is required for the period $[0, T_s = T_p]$.

- *Pay-In-Arrear:* settle a payment after/at T_e, which is a natural way to settle simple interest rates payment. If $T_p \geq T_e$, we call it *payment delay.* The present value is calculated as

$$V_{\textit{pay-in-arrear}}(t_0) = df(t_0, T_p)\tau(T_s, T_e)R(T_f = t_0; T_s, T_e) \tag{2}$$

Suppose in the pay-in-advance case, the payment offset is $0B$, then the PV is really just τR. In this case, R is the spot rate[1]. If payment time is not immediate, then in either case, R is the historical fixing as of date T_f because it is crystallized at payment time.

[1] It is equal to the fixing rate if the very moment it settles is the time daily fixing is published.

2.2 Compounded Interest Rates

Compounded interest rates are more complex than simple interest rates because they depend on a series of sampled observed (e.g., daily rates, weekly rates) between T_s and T_e, as well as the method used to aggregate these rates into a single term rate.

Without loss of generality, let us assume the compounding frequency is daily, e.g., Risk-Free-Rate (RFRs). The major compounding types are

$$\begin{aligned} \textit{Geometric Compounding: } R_c &= \frac{1}{\tau}\left(\prod_{i=0}^{N-1}(1+\tau_i R_i) - 1\right) \\ \textit{Arithmetic Compounding: } R_a &= \frac{1}{\tau}\sum_{i=0}^{N-1}\tau_i R_i \end{aligned} \tag{3}$$

where τ_i is the daily accrued. It is obvious that both R_c and R_a cannot be fixed-in-advance, as all daily fixings have to be determined in order to fix the compounding rate between $[T_s, T_e]$. Therefore, such compounding interest rates is by definition fixed-in-arrear. Be cautious with terminology, if a product says *compounded RFR*, it means geometric compounding, unless explicitly stated otherwise.

Compared to LIBOR, compounding with RFRs offers significantly greater flexibility[2]. Most notably, RFR-based terms can be customized to match virtually any accrual period, whereas LIBOR was limited to a fixed set of standard tenors (e.g., 1M, 3M, 6M). Apart from that, there are couple of nuances for RFR-based terms.

- *Standard RFR Rate:* in *figure 1*, the accrual period, observation and calculation period coincides (just like LIBOR), but fixing time is at T_e, and it also allows a payment delay from $T_p \geq T_e$.
- *Look-back:* a look-back shifts the observation period for the daily rates backward in time by a fixed number of days, while keeping the accrual period fixed. In *figure 2*, the fixings used to calculate R starts from $T_s - 1$ and stops at $T_e - 1$.
- *Rates Cutoff:* a rate cutoff stops updating the compounded rate a few days before the end of the accrual period. For the remaining days, the last available rate is used repeatedly. In *figure 3*, we applied rates-cutoff = 1B, which means the last fixing copies from the previous day.

[2] Here, LIBOR and RFR are used as representative examples of simple and compounded interest rate conventions, respectively.

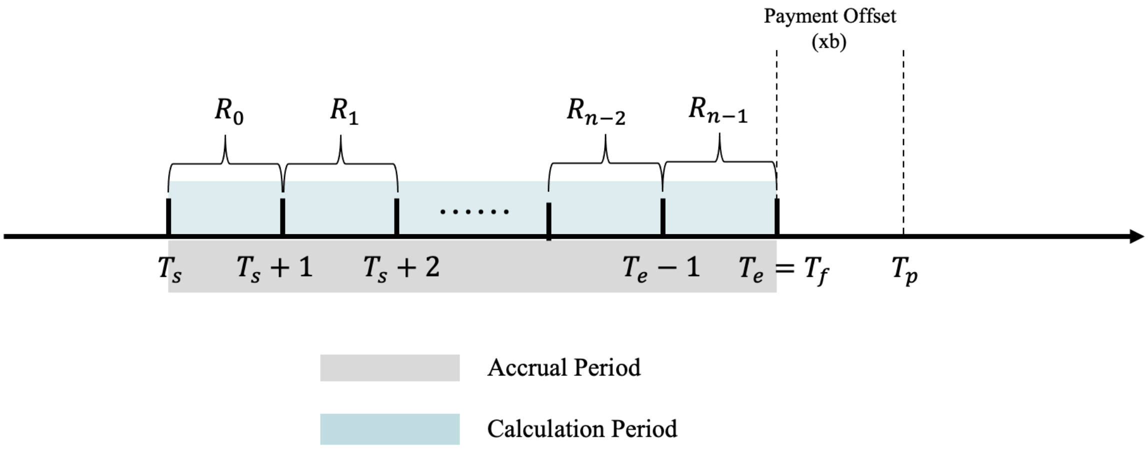


Figure 1: Standard

3 Implementation

3.1 Dates Utilities

Implement two APIs:

- Add a period (tenor) on a given date following certain business day/holiday convention:

  `addPeriod`(*date, tenor, business convention, holiday convention*)

- Given two dates, calculate function τ subject to $\mathcal{C}$,

  `accrued`(*date 1, date 2, accrual basis, business convention, holiday convention*)

You are encouraged to leverage the functionality provided by QuantLib

`https://quantlib-python-docs.readthedocs.io/en/latest/dates.html#ql.Calendar.holidayList`

You may wrap the relevant QuantLib APIs to match the signatures above for easier integration and abstraction.

3.2 Yield Curve Model

Implement a simple yield curve model (similar to what you have done in the class). Specifically, the following are required (and you're free to add more):

- *Constructor:* the yield curve class should be constructed from the following inputs:
  - value date: it has to be a date object

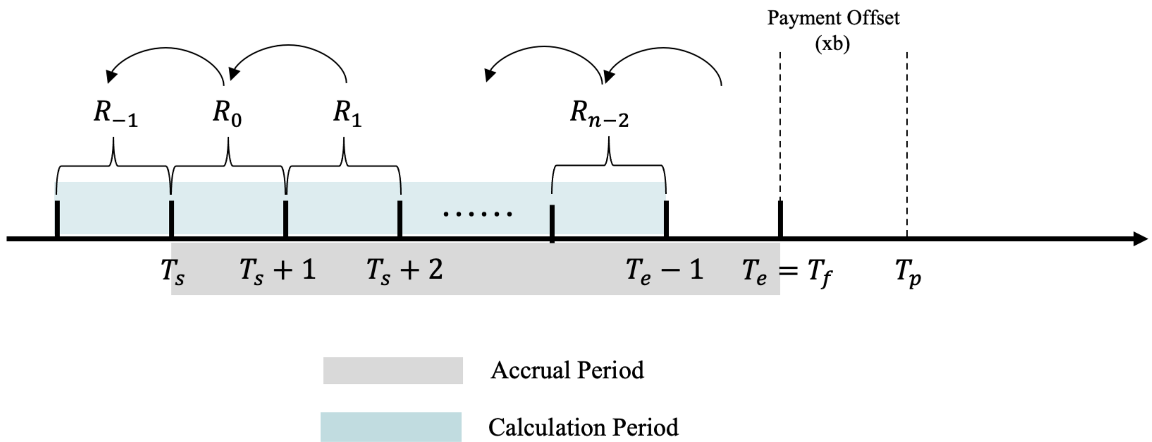


Figure 2: Look-back

- market data (dataframe): a set of indices (e.g., RFR/LIBOR/FedFunds) allows multi-discounting. Given an index, a set of pillar points t_i's, which can be either dates or tenors (or a mixture), a set of instantaneous forward rate r_i, where each r_i is piecewise constant on $[t_{i-1}, t_i]$.

These should be registered as class member variables.

- *DiscountFactor:* given an index, a date/tenor (t) greater than value date, this member function should return a discount factor $df(0, t)$.

You shall also use the date functionality implemented in the last section.

3.3 Product

Implement two classes, one for FRA and one for Swap. Use FRA as an example,

- *Constructor:* FRA should be constructed from below inputs:
  - *Expiration Date(date):* the expiration date (T_f) of an FRA contract;
  - *Termination Date (term/date):* the termination date or tenor of the underlying rate;
  - *PayOrRec:* a payer swap or a receiver swap;
  - *Notional:* the notional amount;
  - *Fixed Rate:* the fixed rate to be exchanged;
  - *Floating Leg Index:* e.g., LIBOR, SOFR, etc;
  - *Payment Offset (optional):* the payment offset (notice, LIBOR is diff from RFR);
  - *Payment Offset Business Convention (optional):* the business day convention used to calculate payment date;

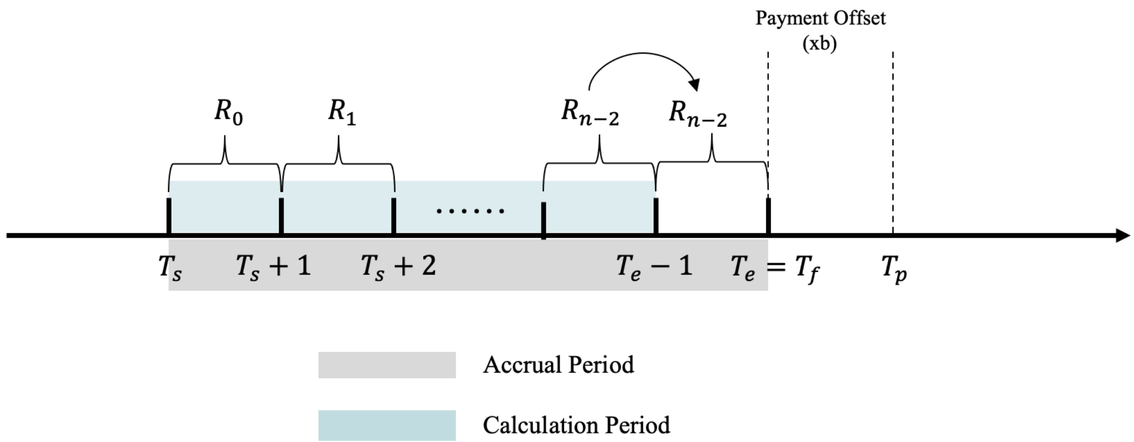


Figure 3: Rates Cutoff

- *Payment Offset Holiday Convention (optional):* the holiday convention used to calculate payment date;
- *RFR Nuances (optional):* only applicable to RFR products.

These members should be registered as the attributes of the product.

- *Accessors:* implement simple accessors to retrieve above attributes. For istance,

```
def fixedRate(self):  return self.fixedRate;
def indexFixingDate(self):  return self.fixingDate;
def indexStartDate(self):  return self.startDate;
def indexEndDate(self):  return self.endDate;
```

Notice, you need to implement an index registry, which allows you to look up fixing offset, business day convention, and holiday convention. For instance, given the expiry of FRA, you will need the conventions from the index to calculate T_s. Also, if you're given a tenor, e.g., $3M$, you will need the conventions to calculate T_e.

3.4 Valuation Engine

Valuation engine is the place where model and product meets. The basic elements of a valuation engine class is:

- *Constructor:* for any product, the valuation engine should be constructed from:
  - *Model:* the model to value the product;
  - *Product:* the product to be valued;

- *ValuationConfiguration (dict):* a dictionary specifies the extra info needed for valuation, e.g., which index used for discounting.

- *CalculateValue:* calculate pv;

- *ParRateSpread:* calculate the par rate;

3.5 Abstraction

The best practice is to implement abstract classes for model, product, valuation engine. The principle is:

- You cannot instantiate an abstract class;

- You shall implement common functions in the base class (abstract class);

- You shall implement specific features in the derived class.