# PyFi Methodology

This document is intended to provide an explanation of the methodology used in the PyFi package. This will hopefully make review easier than trying to read code. This document isn't meant to derive or justify any of the formulas or methods - this information can be found elsewhere.

## 1 Functions

### Present/Future Value of Cash Flows

- **pv(.)** - returns the present value of a series of cash flows.

$$pv = \sum_{t=1}^{n} CF_t * (1 + (apr * dt))^{-t*dt} \tag{1}$$

  A few things should be noted about this formula:

  - $CF$ is the cash flow list that's passed to the function. The code assumes the first cash flow happens at $t = 1$, so $CF_t$ would be `cash_flows[t-1]`.
  - $apr$ is the nominal annual interest rate.
  - $n$ is the total number of cash flow periods. This is equivalent to the length of `cash_flows`.
  - $dt$ is the time, in years, between cash flow periods. Note that $dt * n$ is the total number of years that the cash flows cover.

- **fv(.)** - returns the future value by taking the pv, then bringing it forward in time.

$$fv = pv * (1 + (apr * dt))^{n} \tag{2}$$

### Numerical Approaches

- **irr(.)** - returns the internal rate of return of a series of cash flows. The IRR is the $apr$ that satisfies

$$0 = pv = \sum_{t=1}^{n} CF_t * (1 + (apr * dt))^{-t*dt} \tag{3}$$

  This is achieved numerically by performing the following steps:

  1. Guess an initial apr, and adjust by some factor (initially .1) in one direction until the $pv$ switches signs.

2. When the $pv$ switches signs, divide the adjusting factor by 10 then adjust in the direction that brings the $pv$ closer to 0.

3. Repeat step 2 until $pv = 0$ or the $apr$ is found to 10 decimal places.

Note that at least 1 cash flow must be of another sign than the others, or else $apr = \infty$. All cash flows having the same sign will throw an exception.

## Cash Flow Characteristics

- **macD(.)** - returns the Macaulay duration of a series of cash flows. This is the weighted average maturity of the cash flows.

$$pv_t = CF_t * (1 + (apr * dt))^{-t*dt} \tag{4}$$

$$MacD = \sum_{t=1}^{n} \frac{(t * dt) * pv_t}{pv_t} = \frac{\sum_{t=1}^{n}(t * dt) * pv_t}{pv} \tag{5}$$

- **modD(.)** - returns the Modified duration of a series of cash flows. This is the percentage change in $pv$ for a percentage *point* change in yield, or $apr$. Technically, it is given by

$$ModD = -\frac{1}{pv} * \frac{\partial pv}{\partial(apr * dt)} \tag{6}$$

However, PyFi uses the following convenient relationship to find it:

$$ModD = \frac{MacD}{1 + (apr * dt)} \tag{7}$$

- **convexity(.)** - returns the convexity of a series of cash flows. This is defined as

$$Convexity = \frac{1}{pv} * \frac{\partial^2 pv}{\partial(apr * dt)^2} \tag{8}$$

For PyFi's purposes, with periodic compounding, this formula becomes:

$$Convexity = \frac{\sum_{t=1}^{n} t * (t + 1) * dt^2 * w_t}{(1 + (apr * dt))^2} \tag{9}$$

where

$$w_t = \frac{pv_t}{pv} \tag{10}$$

is the weight of each particular payment in the total $pv$. Note that in the code, the values $t$ and $t + 1$ in (9) are instead $t + 1$ and $t + 2$, due to the list indexing beginning at 0.