

# Introduction

Kevin Crotty  
BUSI 448: Investments

# Investments Introduction

# Course topics (1)

- Financial Markets Overview
  - Stocks + bonds focus
  - Historical performance
  - Markets structures & trading
  - Arbitrage & its limits
  - Leverage
  - Short-selling

# Course topics (2)

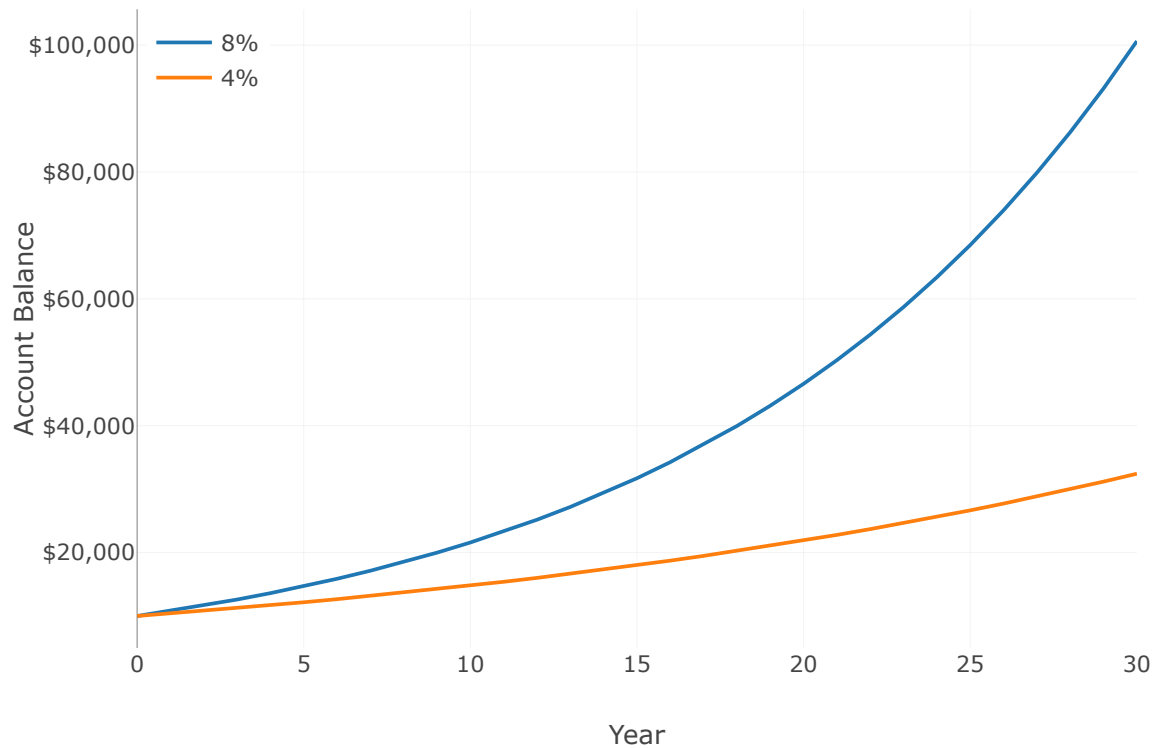
- Optimal Portfolios
- Equity pricing models
- Debt pricing models
- Taxes
- Asset manager performance evaluation

# Why study investments?

- An intellectually interesting field.
- A potentially lucrative field.
- Every one of us will (or maybe have already) face savings problems where we need an understanding of investments.
- Rice University itself faces this problem!
  - Rice Management Company

# Why study expected returns?

**Compounding:** twice the rate turns into much more than twice the money if invested over enough time!



# Learn Investments website

Some of the course will use the [Rice Business Learn Investments Dashboard](#).

The pages demonstrate many concepts we will cover.

- Can be used to solve problems.
- Should be used to help understand concepts and develop intuition.

# Rice Business Learn Investments Dashboard

Winner of the 2023 Financial Management Association Innovation in Teaching Award

Enter



Welcome to the dashboard for learning and teaching investments and related topics, including options and futures, fixed income, portfolio management, and introductory finance. Created at the [Jones Graduate School of Business](#) at [Rice University](#).

This website contains over 80 pages in 8 different sections. Each section includes an overview that describes the pages in that section. The pages present interactive tables and figures to illustrate investment concepts. Many pages pull data from online



# Textbooks

- Recommended readings available in the syllabus from Investments text by Bodie, Kane, and Marcus.

# Some tools we'll develop

- Analyzing data
  - basic summarization
  - regression analysis
- Constrained optimization
- Monte carlo analysis (simulation)

We will use Python as our toolkit.

# Our toolkit: Python

Why python?

- Python is open-source, so you'll be able to use it without cost down the road in your careers.
- Python allows us to study data and models that would be cumbersome to evaluate in Excel.

# Python resources

- I use both Jupyter notebooks (.ipynb) and the Spyder IDE to run Python files (.py), depending on what I'm working on.
- We will use Jupyter notebooks to run Python on Google Colab (or at mybinder.org).
- Alternatively, you are welcome to install Python on your machines and run everything yourself.
  - In general, I will not be troubleshooting Python installations
- Good python reference: [Python for Data Analysis](#)

# Google Colab

- In class calculations will be done through Google Colab
- Links to notebooks for each day are available through Canvas
- We will start out today with straight-forward finance concepts to get on the same page.

# Course binder

- In class calculations can also be done through mybinder.org.
- Course binder link

# Bond Pricing

# Quick review

Bond pricing inputs:

- CR: annual coupon rate
- YTM: annual yield to maturity
- NOP: payments per year (also # of compounding periods)
- PRINCIPAL: face value of the bond (amt borrowed)
- MATURITY: time-to-maturity (either in periods or years)

What does the timeline of bond CFs look like?



# Bond pricing function

Zero coupon bond:

$$P = \frac{FACE}{(1 + YTM)^T}$$

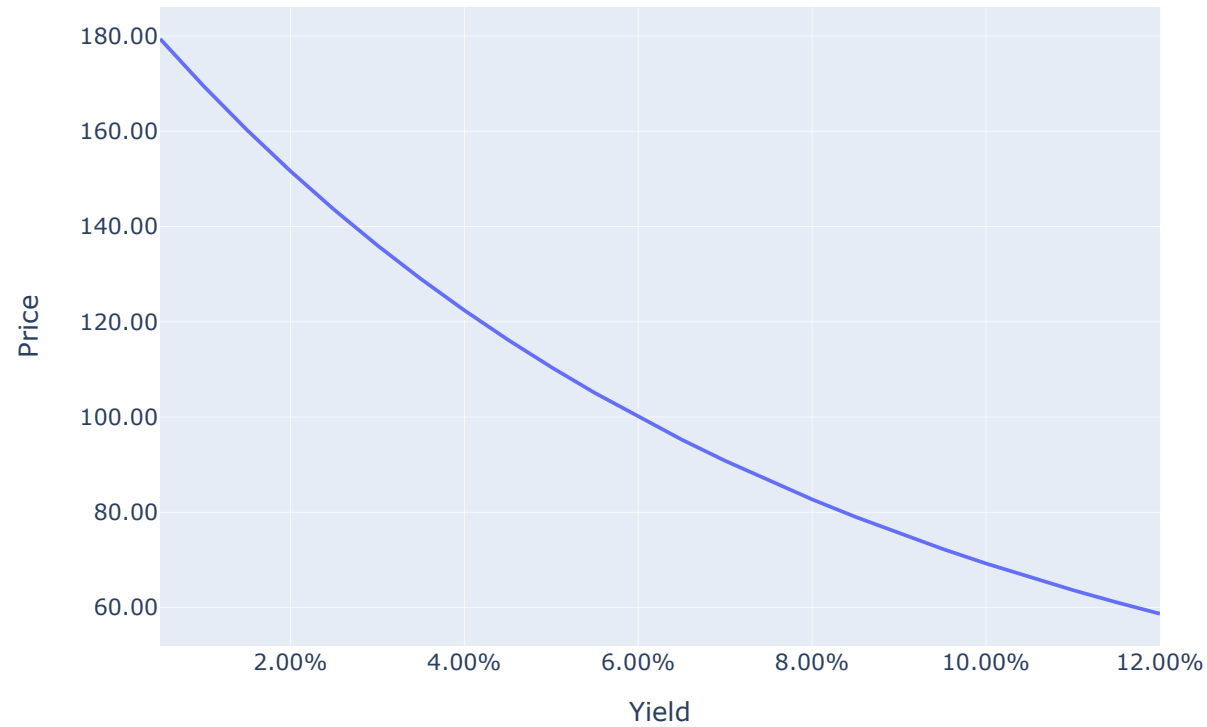
Annual coupon bond:

$$P = \sum_{t=1}^T \frac{COUPON}{(1 + YTM)^t} + \frac{FACE}{(1 + YTM)^T}$$

- YTM: rate that sets the price of the bond equal to PV(future CFs)

# Bond pricing function

Bond prices move inversely with yield!



# Bond pricing - multiple payments per year

$$P = \sum_{t=1}^T \frac{C}{(1 + DR)^t} + \frac{FACE}{(1 + DR)^T}$$

where:

- $DR = \frac{YTM}{NOP} =$  periodic discount rate
- $C = \frac{CR \cdot FACE}{NOP} =$  periodic coupon payment (\$)
- $T = MATURITY \cdot NOP =$  total number of payments (compounding periods)

# Pricing a Bond in Python

3 methods:

- Write out and discount each cash flow
- Use `npf.pv` function
  - custom function to take annual inputs
- Use analytical annuity formula

$$P = \frac{C}{DR} \left( 1 - \frac{1}{(1 + DR)^T} \right) + \frac{FACE}{(1 + DR)^T}$$

# For next time: Saving for Retirement

