Long Short Portfolio Optimisation: Developed a regime-conditioned long–short equity strategy using Fama–French factors, classifying macro regimes as Inflationary, Recessionary, Goldilocks, and Stagflation based on economic indicators. Applied regime-specific factor weighting to rank stocks, long top 5 and short bottom 5, maintaining market-neutral exposure and evaluating performance across regimes.

Options Payoff Diagram Constructor: Built an interactive tool to visualise payoff profiles for custom options strategies, supporting long/short positions across calls and puts with adjustable strike prices, premiums, and quantities.

Multimodal Earnings Intelligence Strategy: Engineered a quantitative framework combining NLP sentiment extraction from earnings call transcripts with GARCH-X volatility modeling to predict post-announcement price drifts and IV crushes. Integrated FinBERT-derived linguistic features (sentiment divergence and uncertainty markers) with options-implied data to execute event-driven straddles, optimizing risk-adjusted returns during high-volatility windows. - use both "hard" data (revenue, EPS) and "soft" data (CEO tone, analyst skepticism)

Roll-down Optimizer: Yield curve is currently steep - roll-down strategy is top recommendation (buying a bond and holding it as it moves closer to maturity, sliding down to lower yield = higher price). Goal: Calculate the theoretical P&L of holding a 5-year bond position vs. a 2-year bond position over a 6-month horizon, assuming the yield curve shape stays exactly the same. Identifies the most profitable point along an upward sloping yield curve to invest in.

How does intraday realized volatility influence equity market liquidity, as measured by bid–ask spreads and Amihud illiquidity, and how does this variation impact transaction costs for medium-sized trades (0.5–1% of stock's daily volume) in liquid U.S. stocks?

Does a volatility risk premium exist in short-dated at-the-money equity index options, measured as the difference between implied volatility (IV) and subsequent realized volatility (RV)? Is this premium statistically significantly positive overall and across different market volatility regimes (low, medium, high) defined using the VIX index?

This project, often called **"Riding the Yield Curve,"** is a classic strategy used by bond fund managers to squeeze extra profit out of a stable market. It is less about predicting the future and more about mathematically exploiting the current "shape" of interest rate markets.

---

1. The Aim: Why do this?

In a "normal" or **steep yield curve**, long-term interest rates are higher than short-term rates (because investors demand more "rent" for locking their money away for longer).

- **The Problem:** If you just buy a bond and wait, you collect interest. But can you make more than just the interest?
- **The Goal:** To prove that by buying a longer-term bond (like a 5-year) and selling it after 6 months, you can make more money than if you had just bought a 6-month bond. This extra profit comes from the **Price Increase** as the bond "slides" down the curve.

2. The Concepts: Time vs. Price

To understand this, you need to know one rule: **When yields go down, bond prices go up.**

- **The "Slide":** Imagine a 5-year bond pays 5%. As time passes, that bond becomes a 4.5-year bond. If the curve is steep, 4.5-year bonds might only pay 4%.
- **The Profit:** Because your bond is still paying the "old" higher rate (5%) but is now valued at the "new" lower rate (4%), the bond's market price **jumps up**. You get your interest (coupon) **PLUS** the capital gain from the price jump.

3. The Technology: What you need to build it

Since this is a mathematical optimization project, you need tools for interpolation and calculation.

Component	**Technology**	**Why?**
Data Fetching	pandas_datareader (FRED)	To get the current Treasury Yield Curve (1m, 2y, 5y, 10y rates).
Curve Fitting	SciPy (Cubic Splines)	To "fill in the gaps." If you have 2y and 5y rates, you need a smooth line to find the exact rate for 4.5 years.
Bond Pricing	NumPy-Financial	To calculate the exact "Present Value" (Price) of the bond at different points in time.
Visualization	Matplotlib	To plot the "Roll-Down Profile"—showing which part of the curve has the steepest "slope" and thus the most profit.

4. The Strategy: How you trade it

You are essentially performing a **Scenario Analysis**.

- **Step A:** Look at the current curve. Calculate the price of a 5-year bond today.
- **Step B:** "Fast-forward" 6 months. Assume the curve hasn't moved. Your bond is now a 4.5-year bond. Calculate its new, higher price at the 4.5-year yield.
- **Step C:** Compare the total return (Coupon + Price Change) of the 5-year bond vs. a 2-year bond.
- **The Result:** You identify the **"Sweet Spot"**—the point on the curve where the slope is steepest, giving you the maximum "slide" profit for the least amount of time.

Maturity	Yield
1 year	2.0%
2 year	2.5%
5 year	3.5%
10 year	4.2%

Step 1️⃣ Pick a bond to buy

We buy a **5-year bond** with:

- Face value = **$100**
- Coupon = **3.6% per year** (= $3.60)
- Maturity = **5 years**

Because:

- Coupon = yield
  ➡️ The bond trades at **par price = $100**

Step 2️⃣ Hold the bond for 1 year

After **1 year**:

- The bond is no longer a 5-year bond
- It is now a **4-year bond**
- It still pays the same $3.60 coupon

⚠️ Key assumption:

> The **yield curve shape stays exactly the same**

So:

- 4-year yield ≈ **3.0%** (between 3-year and 5-year)

Step 3️⃣ Re-price the bond after 1 year

Now the market wants a **3.0% yield** on 4-year bonds.

But your bond pays **$3.60 per year**.

To make its yield fall from 3.6% → 3.0%:

- Its **price must go up**

Let's compute roughly:

New price (simplified)

- Annual coupon = $3.60
- Market yield = 3.0%

A rough price approximation:

Price≈3.600.03=120\text{Price} \approx \frac{3.60}{0.03} = 120Price≈0.033.60=120

(Exact pricing would discount all cash flows, but this is enough for intuition.)

➡️ New bond price ≈ **$106–108 range** (realistically)

Data Intake (Input): You feed the system the current "raw" interest rates (yields) for various maturities (e.g., 1M, 2Y, 5Y, 10Y) fetched from a source like FRED or the US Treasury.

Curve Fitting (Interpolation): Since the market doesn't provide a rate for every single day, you use a **Cubic Spline** to draw a smooth, continuous "slide" between the data points. This allows you to find the exact yield for any arbitrary time (like 4.5 years).

Aging Simulation (Process): You select a bond (e.g., a 5-year) and calculate its price today. Then, you "fast-forward" 6 months. Using your smooth curve, you find the yield for a 4.5-year bond. Because the curve is steep, this yield is lower, which mathematically **pushes the bond's price up**.

Total Return Calculation: You sum the **Interest** (coupons) earned over those 6 months and the **Capital Gain** (the price jump from rolling down the curve). You do this for every maturity point on the curve.

Risk Stress Test: You calculate the **Breakeven Yield Shift**—the "cushion" that tells you how much market rates can rise before your profit disappears.

Optimal Selection (Output): The system identifies the **"Sweet Spot"** (the maturity with the highest total return) and generates a report recommending exactly which bond to buy to "ride the curve" most efficiently.

- Read research papers related to this
- Setup github - uv venv