Testing the strategy: How backtesting shapes systematic fixed income performance

numerix

When numbers matter

Contents

1. Introduction

Over the past decade, systematic fixed income (FI) strategies – applying quantitative, rules-based models to bond portfolios – have transitioned from niche experiments to mainstream portfolio techniques.

Several developments have enabled this rise. In particular, the electronification of bond trading (wider electronic and algorithmic trading adoption), improved data availability (e.g., trade reporting and indices), and advances in computing power[1]. With greater transparency and data, investors can now construct robust historical simulations for fixed income strategies where previously it was impractical.

However, systematic fixed income strategies are only as good as the research and backtesting behind them. Fixed income markets pose unique challenges for backtesting. Sparse historical data relative to equities, over-the-counter instruments with infrequent trading, corporate actions like mergers and bond calls, and the possibility of defaults are some of the main challenges.

Without careful handling, backtests may give misleading results. For an institutional investor – whether a hedge fund seeking alpha or an insurer managing asset risk – a thoroughly vetted backtest is crucial before allocating capital. Backtesting provides a window into how a strategy might have performed historically, helping to evaluate its profitability, risk profile, and robustness across market regimes. It allows strategists to refine ideas, statistically validate signals, and avoid costly real-world failures.

"Fixed income markets pose unique challenges for backtesting. Sparse historical data relative to equities, over-the-counter instruments with infrequent trading, corporate actions like mergers and bond calls, and the possibility of defaults are some of the main challenges."

[1] State Street. Insights. The Case for Systematic Investing in Credit, 18 August, 2023

2. The importance of backtesting in systematic fixed income strategies

Backtesting is the cornerstone of systematic investing. It involves applying a strategy's rules to historical market data to simulate how it would have performed. For systematic fixed income strategies, backtesting serves several critical purposes.

> "Before risking real capital, investors can see how a fixed income trading idea (e.g., a factor model or algorithmic trade) would have fared historically."

Idea evaluation

Before risking real capital, investors can see how a fixed income trading idea (e.g., a factor model or algorithmic trade) would have fared historically. This helps filter out strategies that do not look promising. Historical backtests of quantitative signals in fixed income markets have suggested that factor-based approaches can drive outperformance over benchmarks[2]. If a value or momentum signal in corporate bonds never produced positive excess returns historically, it is unlikely to suddenly work in live trading.

Understanding performance drivers

A backtest can decompose where returns come from. For example, it can determine if credit spread compression, roll-down, or default avoidance are the main profit-and-loss (P&L) drivers. It allows strategists to identify which market conditions favor or hurt the strategy (such as how it did in stress periods like 2008 vs. calmer periods).

This contextual understanding is vital for portfolio managers and asset allocators to trust a systematic strategy. A backtest, for example, might reveal that a high-yield bond strategy derives most returns from avoiding defaults and performing well in recoveries, but lags during liquidity crises. These are insights one can only get from historical simulation.

> "Simulating a strategy across history provides a rich sample of potential outcomes, enabling risk analysis."

Risk assessment

Simulating a strategy across history provides a rich sample of potential outcomes, enabling risk analysis. Metrics like maximum drawdown, volatility, and worst-case loss can be estimated from backtest results. Particularly in credit, tail risks (e.g., sudden default spikes or liquidity freezes) are a major concern.

Backtesting through past crises (e.g., 2008 global financial crisis, 2020 pandemic liquidity crunch) tests how the strategy might weather such events. It also exposes any concentration risks. For instance, it might reveal that a strategy inadvertently loads up on one sector or issuer historically.

[2] State Street. Insights. The Case for Systematic Investing in Credit, 18 August, 2023

Parameter tuning and strategy refinement

By examining historical outcomes, quant analysts can tweak strategy parameters (within reason) to improve performance or robustness. For instance, finding the optimal look-back window for a momentum signal or the ideal threshold for a credit spread filter often involves iterative backtesting.

The key is to avoid overfitting (discussed later). Parameters should ideally be chosen based on economic rationale and then confirmed via backtest, rather than blindly maximizing historical profit.

Demonstrating efficacy to stakeholders

For institutional adoption, a strategy must be backed by evidence. Investors, risk managers, and oversight committees typically require seeing backtested performance statistics before green-lighting a new systematic fixed income fund or strategy. A well-documented backtest showing strong Sharpe ratios, consistency across sub-periods, and low correlation to existing strategies can make a compelling case. In fact, systematic fixed income managers often present backtested results to illustrate that their signals (e.g., value, momentum, quality) would have added alpha historically[3].

Relative value nature of fixed income markets

Fixed income trading is largely driven by relative value – the idea of exploiting price discrepancies between related bonds, or rates, rather than making outright bets on interest rates. In practice, most FI strategies are spread trades, meaning that they involve a long position in one instrument and a short position in another, aiming to profit from the spread (or difference) between them.

Because FI trades revolve around spreads, the choice of the benchmark curve against which spreads are measured is critically important. The benchmark curve defines the 'value-neutral' state – in essence, the baseline of fair value yields from which any deviation (spread) is considered.

> "When building a systematic FI strategy (especially in credit), accounting for this relative value orientation is paramount."

When building a systematic FI strategy (especially in credit), accounting for this relative value orientation is paramount. A backtest must ensure that it captures the true spread dynamics and not accidentally loads up on outright interest rate risk.

3. Challenges in historical simulation for fixed income markets

Backtesting a fixed income strategy is often more complex than doing so for equities or other asset classes. Fixed income instruments have idiosyncrasies that can complicate historical analysis.

Limited data and the need for proxies

One fundamental challenge is the limited length and depth of historical data in fixed income markets. Corporate bonds and credit default swaps (CDS) have not been traded electronically for nearly as long

[3] State Street. Insights. The Case for Systematic Investing in Credit, 18 August, 2023

or as transparently as equities. Reliable price and total return indices for corporate bonds only have only existed in many cases in recent decades.

For example, the commonly used Barclays US Corporate Bond Index data begins in the late 1980s. High-yield indices started around the mid-1980s[4]. In contrast, equity data often spans a century or more[5]. This poses a problem. Many credit strategies need to be tested over a variety of market cycles (e.g., inflationary periods, recessions, crises). How can one backtest far back in time if the instrument or index did not exist yet?

The solution is often to use proxies or synthetic extended time series. Researchers will append or splice together different datasets to create a longer history. For instance, Asvanunt and Richardson (2016) constructed a reliable long-run series for US corporate bond returns by combining the Ibbotson Associates long-term corporate bond series (starting in 1926) with the modern Barclays index (from 1988 onward)[6].

They adjusted the older series by removing duration-matched Treasury returns to isolate the credit excess returns, yielding a synthetic history of the "credit risk premium" back to 1936[7]. This allowed analysis of fixed income returns over nearly 90 years, encompassing the Great Depression, World War II, and multiple default cycles – far beyond the live index data.

Another approach to extend history is using related market proxies. For example, if one is testing a strategy on corporate bond spreads but data starts in 2000, one might use older data on credit spreads inferred from bond yield indices or even from Moody's corporate yield averages (which go back many decades)[8].

As a proxy for high-yield corporate bonds before the 1980s, one might use "Baa-rated" bond series, or a combination of Treasury yields plus a constant spread to approximate corporate yield behavior[9]. While imperfect, these proxies can capture the broad pattern of credit returns in earlier periods.

> "The solution is often to use proxies or synthetic extended time series. Researchers will append or splice together different datasets to create a longer history."

[4] Pricing of Corporate Bonds: Evidence From a Century-Long Cross-Section. Mohammad Ghaderi, Sebastien Plante, Nikolai Roussanov, Sang Byung Seo. 29 April 2025.

[5] Historical Returns on Stocks, Bonds and Bills: 1928-2024. Historical Returns for the US, January 2025

[6] AQR. Systematic Credit Investing, April Frieda, CFA and Scott Richardson, Ph.D.

[7] AQR. Systematic Credit Investing, April Frieda, CFA and Scott Richardson, Ph.D.

[8] Federal Reserve Bank of St. Louis. September 2025.

[9] U.S. Department of the Treasury.

In some cases, synthetic data generation is used. This can involve statistical or simulation techniques to create additional history. For example, one might simulate additional return paths by bootstrapping or by modelling credit spread dynamics and generating hypothetical past data. Care must be taken, as synthetic data may not capture all real-world nuances. However, it can be a useful supplement.

A practical method highlighted in quantitative forums[10] is to prepend index returns to an ETF's history to extend it backward. For example, to backtest a strategy using a credit ETF that launched in 2010, one could splice the relevant index's earlier returns (with adjustments for fees) onto the ETF's pre-2010 "history"[11]. This gives a proxy history as if the ETF existed before its inception[12].

> "It is important to acknowledge the limitations when using proxies or stitched datasets. Different eras had different market structures."

It is important to acknowledge the limitations when using proxies or stitched datasets. Different eras had different market structures, e.g., bond trading in the 1970s was very different from today. Data might be of lower frequency or quality (annual or monthly yields vs. daily prices). This notwithstanding, having an indicative long-term backtest is often better than none[13], as it allows testing a strategy across varied regimes. When using extended series, one should document how it was constructed and perhaps test the strategy on the "live" portion separately to ensure consistency.

Accounting for corporate actions and defaults

Another major challenge in fixed income backtesting is handling corporate actions and credit events that occur over time. In equity backtests, one must adjust for stock splits, dividends and mergers. In credit, the analogs are different but equally important.

Issuer mergers and corporate actions

Bonds are obligations of specific corporate entities. Over a backtest spanning many years, companies may undergo mergers, acquisitions, spin-offs, or name changes. The result is that a bond's issuer (its "credit parent") can effectively change or cease to exist. This complicates tracking the performance of a given bond or issuer over time. For example, if Company A is acquired by Company B, the bonds of A might become obligations of B. A 'naïve' backtest might lose track of those bonds or double count them if not careful.

Solutions include using databases that maintain historical issuer mappings (so that, say, pre-merger bonds are attributed to the post-merger entity for analytical purposes), or simplifying analysis at the bond level without aggregating by issuer (although index membership changes must then be handled).

Bond maturities and calls

Unlike stocks, bonds have finite maturities. If you are simulating a strategy that holds a particular bond, what happens when that bond matures or is called early by the issuer? The backtest needs rules for reinvestment. A common approach is to assume that when a bond matures, the proceeds are rolled into a new bond that meets the strategy's criteria at that time.

[10] StackExchange. Quantitative Finance. Backtesting with Stock Indices, how does one deal with it?

[11] Quantstart. Generating Synthetic Histories for Backtesting Tactical Asset Allocation Strategies

[12] Quantstart. Generating Synthetic Histories for Backtesting Tactical Asset Allocation Strategies

[13] Quantstart. Generating Synthetic Histories for Backtesting Tactical Asset Allocation Strategies

For callable bonds, if history shows a bond was called (redeemed) on a certain date, the backtest should reflect the cashflow from the call (usually at par or a slight premium) and then reinvest. Ignoring these actions would overstate performance (for instance, assuming a high-coupon bond kept paying beyond its call date). Thus, historical corporate actions like calls or tenders must trigger appropriate transactions in the simulation.

> “Perhaps most crucial in fixed income strategies is accounting for defaults. If a strategy invested in a bond that defaulted (or a company that went bankrupt), the backtest must reflect the loss.”

Default and bankruptcy events

Perhaps most crucial in fixed income strategies is accounting for defaults. If a strategy invested in a bond that defaulted (or a company that went bankrupt), the backtest must reflect the loss. This includes applying a recovery rate (e.g., if a bond defaults, investors might recover 40¢ on the dollar in liquidation). A robust backtest cannot simply drop defaulted bonds from the dataset (that would introduce survivorship bias – more on this later). Instead, it should realize the negative return on the default date (or over the default workout period).

For example, a backtest of a high-yield bond portfolio should show a hit in 2001-2002 from WorldCom and other defaults, and in 2008-2009 from the spike in default rates. If those events do not appear, the backtest is likely not including the impact of defaults properly. Index total return series inherently account for defaults.

When bonds default and exit the index at a recovery value, the index sees a drop. But in a custom strategy backtest, one needs to manually implement this (i.e., remove the defaulted instrument at recovery value and reinvest remaining funds).

Accounting for defaults also means ensuring the universe of bonds in the backtest includes those that defaulted or were distressed. A common mistake is survivorship bias – using today’s list of bonds or issuers and backfilling their history. That misses companies that existed in the past but later defaulted or disappeared. For a credit factor strategy, this bias can be acute.

Many “fallen angels” (investment-grade companies that have defaulted and thus lost their investment-grade rating) would be excluded if one only looks at surviving issuers. The result would be an overly positive backtest. The best practice is to use point-in-time universes (i.e., at each date in history) and know exactly which bonds were outstanding and tradable at that time, including those that later defaulted. Commercial datasets and index histories can provide this information.

Handling these corporate actions and credit events typically requires clean data and careful simulation logic. One might maintain lists of historical corporate actions. If a bond CUSIP changed ticker due to a merger, map it; if it was called, have the call date and price. This is often where credit backtesting can become quite technical. Some modern backtesting platforms and data vendors offer tools that handle many corporate actions. If one is doing it in-house, it may involve stitching multiple data sources (e.g., rating agency default databases for default dates and recoveries, and corporate actions from news sources and other specialist sources).

4. Best practices for reliable backtesting

Designing a backtest that truly reflects how a systematic strategy would have performed requires discipline and care. Following these guidelines helps prevent the backtest from being overly optimistic or disconnected from reality.

Data quality and pre-processing

The foundation of any backtest is the quality of historical data. The saying, "garbage in, garbage out," could not be more appropriate. Before running simulations, one must prepare the data. This typically involves a few procedures.

Accurate price and return histories

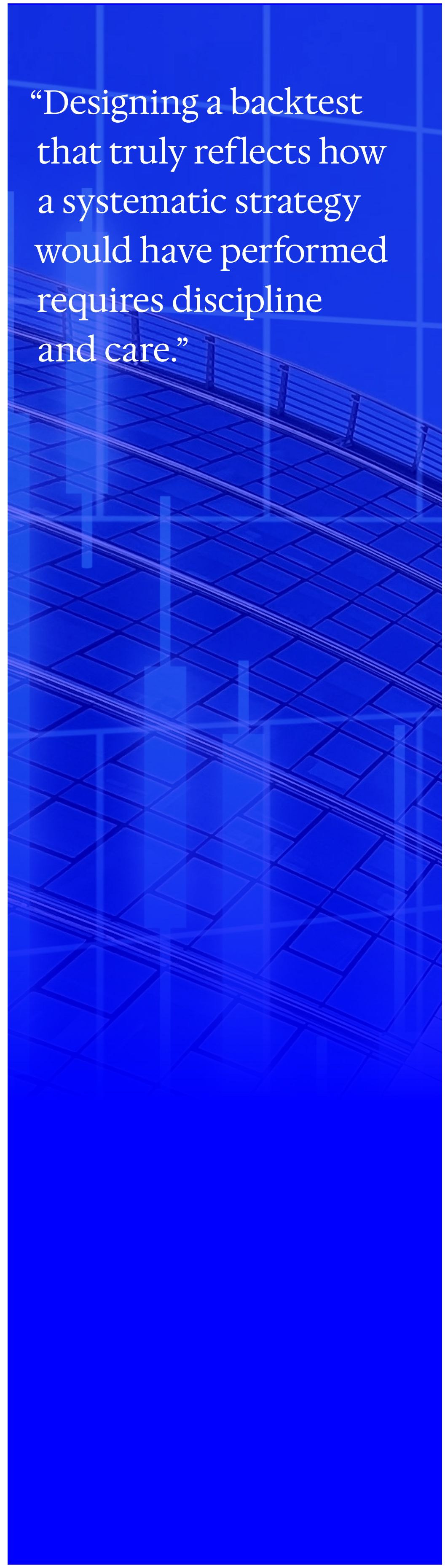
Use reliable sources for bond prices, yields, or total returns. In corporate bonds, trade pricing can be sparse as many bonds do not trade daily. One may need to use evaluated prices to fill gaps. It is important to understand how those prices are determined and to ensure they are point-in-time (i.e., using information only available then, avoiding hindsight). Interpolation or smoothing by data providers is common so one needs to be aware of it. Whenever possible, use data that reflects actual market quotes or transactions and note any periods of illiquidity where prices might be stale.

Corporate actions and adjustments

Adjust prices for events like coupon payments (for total return calculations), calls, and defaults. If a bond paid a coupon, the backtest should either use total return series or add coupon cashflows to the P&L. If a bond defaulted, mark the price down to recovery, ensuring these adjustments maintains the realism of returns. For indices or portfolios, consider that indices typically drop defaulted bonds at a certain point. Your custom backtest should align with a consistent methodology.

Synchronize data frequencies

Fixed income systematic strategies might use multiple data types (e.g., bond prices, equity prices of issuers, macro variables) in signals. Make sure the data frequencies are aligned or appropriately handled. If mixing daily and monthly data, decide on a backtest frequency (say monthly) and sample or aggregate



accordingly to avoid any look-ahead issues. Also, be mindful of different market holiday calendars, time zones, and snapshot timing. A bond's price as of 3pm vs. an equity price at 4pm could introduce slight mismatches.

Survivorship bias elimination

Use historical constituent lists for your universe. For example, if backtesting an investment-grade corporate strategy, get the index membership for each month of the history (or use a broad database of all bonds issued, including those that dropped out due to default). This ensures that defaulted or withdrawn bonds remain in historical datasets up to their exit. As a rule, never use today's index membership or a current list of issuers as the basis for past periods[14].

Point-in-time fundamentals and ratings

If the strategy involves fundamental data (like balance sheet ratios or credit ratings), ensure those are point-in-time. For example, if using credit ratings as an input, use the rating as it was at that date in the past, not the upgraded/downgraded version from the future. Similarly, if using financial ratios, use data that was available at that quarter in history. This avoids look-ahead bias, where future information sneaks into the backtest and inflates performance.

> "Data cleaning can be tedious, but it is vital. Many quantitative strategy failures can be traced back to data problems or biases that were not caught in backtesting."

Why data quality is critical

Data cleaning can be tedious, but it is vital. Many quantitative strategy failures can be traced back to data problems or biases that were not caught in backtesting. For instance, suppose a backtest of a bond strategy showed no losses around defaults. This might be a red flag that default data was missing, and those bonds were simply omitted. Rigorous data validation, including checking basic statistics, plotting time series for sanity, and comparing against known index returns, should be a step before trusting any backtest output.

Out-of-sample validation

Once a strategy is developed, it must be tested on data not used during its creation. This is known as out-of-sample testing, and it is essential for assessing robustness.

Train/test split

One common approach is to split the historical data into two segments: an in-sample period (perhaps older history) used to develop and calibrate the strategy, and an out-of-sample period (more recent history) reserved to test it. For example, use data from 2000–2015 to design the strategy, then see how it would have done in 2016–2025 (which was not "seen" during development). The out-of-sample performance gives a better indication of real-world results. If a strategy performs dramatically worse in the out-of-sample, it is a sign it may have been overfit or reliant on patterns that changed.

[14] Quantified Strategies. 'Survivorship Bias in Trading (How To Avoid It) – Backtesting, Trading and Investing, Oddmund Groette, January 4, 2025

Rolling or walk-forward analysis

This is a more sophisticated validation where you repeatedly simulate how the strategy would be updated and perform through time. For instance, calibrate the model on 2000–2010 data, test on 2011–2012; then recalibrate on 2000–2012, test on 2013–2014, and so on. This walk-forward method mimics how a strategy would actually be re-trained with expanding data and used prospectively. It provides multiple out-of-sample periods and a distribution of performance results.

Cross-validation

Borrowed from machine learning, one can use techniques like k-fold cross-validation for strategies with many parameters. Cross-validation is a technique used to evaluate how well a model 'generalizes' to unseen data. The most common form is k-fold cross-validation, where the dataset is randomly split into k subsets (folds). The model is trained on k-1 folds and tested on the remaining fold, repeating this process k times.

However, in a finance time series context, one must be careful to preserve the time order (so typically one uses expanding windows or skip-sample validation rather than random shuffles).

> "Out-of-sample testing is crucial because it checks the strategy's generalization. It is easy to fool oneself with a great in-sample backtest."

Why out-of-sample validation is so important

Out-of-sample testing is crucial because it checks the strategy's generalization. It is easy to fool oneself with a great in-sample backtest. This phenomenon is sometimes flippantly called "backtest overfitting bingo", where any strategy can be massaged to look good on past data. The true measure is how it fares on data not used during its creation.

When presenting strategy results, the best practice is to show a clear separation between in-sample and out-of-sample testing.

It is worth noting that even out-of-sample testing can be abused if not careful. If you test multiple strategies and each time the out-of-sample fails, and you then tweak the model, the final "out-of-sample" is no longer truly out-of-sample because you indirectly learned from it. This is sometimes called "peeking" at out-of-sample data. The remedy is to treat one portion as a final test that you do not touch until you are ready to do a one-time evaluation.

In practice, some systematic fixed income investors also do paper trading or live simulations once the backtest stage is passed. This is effectively an out-of-sample test in real market conditions but without real capital (or with a small amount). This can further validate that the strategy works with live data feeds, actual executions, and so on.

Incorporating transaction costs and liquidity

A common pitfall in backtesting is to simulate trades as if they incur no costs and face no liquidity constraints. This can be especially misleading in fixed income markets, where bid-ask spreads can be significant and not all bonds can be traded in large quantities without impact. To make a backtest faithful to real trading, one should incorporate the following concepts.

"A common pitfall in backtesting is to simulate trades as if they incur no costs and face no liquidity constraints. This can be especially misleading in fixed income markets, where bid-ask spreads can be significant."

Bid-ask spreads

When a strategy "trades" in the backtest, assume it crosses the bid-ask spread unless there is a specific reason otherwise. For example, if going long a corporate bond, the purchase price should be a bit higher than the mid (to reflect paying the ask) and selling should be at a bit below mid (hitting the bid). If historical bid-ask data is not available for the specific bond, one can estimate it based on bond characteristics (e.g., wider for high-yield or older off-the-run bonds, narrower for on-the-run or high-liquidity issues). Ignoring spreads can dramatically overstate returns, particularly for strategies with high turnover.

Market impact and capacity

Consider the size of the trades, relative to market volume. Fixed income markets can be shallow for certain bonds. If a strategy entails rotating a US$1 billion portfolio into a new set of bonds every month, could the market absorb that easily? If not, one might need to adjust performance for market impact (the price moves caused by your own trading). This is complex to model, but a rule-of-thumb is to haircut expected returns if turnover is high and focused on less liquid names. Alternatively, constrain the backtest to trade only up to a certain percentage of each bond's outstanding amount or trading volume.

Liquidity constraints

A backtest might need to ensure that certain bonds are off-limits if they are too illiquid. For instance, some systematic credit strategies only trade bonds above a certain issuance size (e.g., over US$300 million outstanding) or those in major indices, to ensure liquidity. If your strategy is supposed to be institution-friendly, embedding such constraints in the backtest aligns it with what could actually be done. During crisis periods, liquidity can dry up. One might simulate crises by widening assumed spreads or delaying trade execution in the backtest (to mimic that you could not trade immediately at a reasonable price).

Turnover and transaction cost limits

It is wise to track how much turnover the strategy is generating (annual turnover percentage) and apply an estimated cost per trade. For corporate bonds, trading costs might include not just the spread but any commissions or price impact. You can impose a flat cost, such as 20-50 bps (0.2-0.5%) of the bond notional per trade in the backtest, as a rough estimate and see how it affects returns[15]. Many successful systematic fixed income strategies keep turnover moderate to avoid excessive costs. If a backtest's edge evaporates after applying even modest costs, that is a warning sign the strategy might not survive real trading[16].

5. Avoiding backtesting pitfalls

Even with high-quality data and well-reasoned assumptions, backtesting results can be misleading if not approached with care. Quantitative researchers must be vigilant against certain biases and methodological errors that can make a strategy appear robust in historical simulations but fail in reality.

Below are key pitfalls to watch for and how to mitigate them.

> "Overfitting is the bane of quantitative strategy development. It occurs when a model is too complex or too tuned to historical quirks, effectively 'learning' noise rather than true underlying patterns."

Overfitting and data mining

Overfitting is the bane of quantitative strategy development. It occurs when a model is too complex or too tuned to historical quirks, effectively "learning" noise rather than true underlying patterns. An overfitted strategy will show excellent performance on past data but will likely disappoint going forward. In the context of backtesting, overfitting often arises from data mining. This involves trying many ideas or parameter variations and cherry-picking the one that had the best historical performance.

Modern research tools and rich datasets make it easy to test thousands of hypotheses on past market data. Without discipline, researchers might inadvertently engage in a form of multiple testing. For example, they may test 50 different signals (value, momentum, profitability, leverage) on corporate bond returns and then focus only on the few that worked historically. Those few may have worked by chance alone. Some researchers repeatedly test with the same historical data and simply pick the factor definition that presents the best backtest, effectively engaging in data mining, which is not a sound way to achieve good future performance[17]. In other words, if you torture the data long enough, it will confess to something. However, that something could be a false pattern.

One tell-tale sign of overfitting is extremely complex logic or too many parameters in the strategy. For instance, a credit strategy that uses 12 different signals with specific weightings and non-linear rules

[15] BondWave Trade Insights – Volume 22. Year End Trading Cost Review, Paul Daley, MD, Fixed Income Lab, 30 January, 2025.
[16] How (not) to backtest: 5 pitfalls when testing systematic investment strategies. Medium. 31 July, 2019.
[17] Fidelity International. A bottom-up approach to factor investing in corporate bonds, 3 June 2019, Joe Hanmer, Global Head of Quant

might fit history perfectly, but it is likely over-optimized. Simplicity tends to generalize better. In designing systematic strategies, it is wise to follow Occam's razor[18]. Use as few parameters as necessary and ensure each has economic justification.

Another best practice to combat overfitting is to reduce the degrees of freedom available to the researcher. This can mean setting aside an out-of-sample test (discussed above), using techniques like cross-validation, or applying penalties for model complexity.

Perhaps the simplest guard is honesty about how many variations were tested. Admitting that dozens of ideas were thrown at the wall is not necessarily bad – this is part of research – but then one must take the in-sample performance with a grain of salt. It may be overfit if it was the best among many trial runs.

To illustrate overfitting, consider the following example. Suppose researchers generated 1000 random hypothetical trading strategies and selected the single best-performing one in the historical sample. Unsurprisingly, that "best" strategy had stellar performance in-sample. However, when they simulated its performance on new data, it completely fizzled out.

As the strategy was chosen for its stellar in-sample performance but failed to sustain gains out-of-sample, its apparent "edge" was merely luck. In a backtest, always suspect strategies that look too good to be true – they may have been tuned to past noise.

Overfitting can be mitigated by keeping the strategy development process disciplined. Use theory-driven hypotheses (do not just fish blindly), limit the number of optimizations, and test the strategy on fresh data not used in development.

> "The cure is careful data management and double-checking the backtest logic from a timeline perspective."

Look-ahead bias and survivorship bias

These biases were touched on earlier in the data section. Both look-ahead and survivorship biases are essentially data-handling errors that give unrealistically good backtest outcomes. The cure is careful data management and double-checking the backtest logic from a timeline perspective. Step through a specific historical date in the simulation and verify that all inputs truly reflect only what was known at that time and that the investment universe includes everything it should.

[18] Occam's Razor is a philosophical and problem-solving principle that suggests: "The simplest explanation is usually the best one." More formally, it states that when faced with competing hypotheses or explanations, the one with the fewest assumptions should be selected, unless more complex explanations are necessary to account for the evidence. .

6. Build confidence with backtesting

Rigorously backtesting systematic fixed income strategies – especially in credit markets – is both an art and a science. It requires marrying financial insight with careful data science. For investors, the stakes are high. A strategy that merely looks good on paper but is based on overfit or biased analysis can lead to real losses. Conversely, a strategy that proves itself through exhaustive historical testing and validation can become a reliable source of alpha in the portfolio.

The overarching lesson is that careful backtesting builds confidence. When a strategy has survived harsh historical scenarios, passed stringent validation testing, and retained its edge even out-of-sample, traders and portfolio managers can approach live trading with greater conviction. They know the strategy has been through the proverbial fire – multiple credit cycles, bull and bear markets, liquidity crunches – and has emerged with its efficacy intact. This confidence is not just academic; it translates into sticking with the strategy during the inevitable periods of underperformance, adjusting position sizes wisely, and having the resolve to deploy capital when the model indicates an opportunity.

Systematic fixed income strategies, by their nature, challenge the status quo of discretionary investing. They claim that data-driven, repeatable processes can exploit inefficiencies in fixed income markets. The only way to substantiate that claim is through rigorous historical evidence. A well-executed backtest is thus the bedrock of systematic fixed income investing – it separates genuine alpha generation from "curve-fit" mirages.

As systematic fixed income investing continues to grow, practitioners must hold their strategies to high standards of backtesting discipline. By doing so, they not only protect themselves from costly errors but also contribute to a more robust and innovative investment landscape. With the right approach, backtesting becomes more than a perfunctory exercise. It becomes a powerful tool for learning from the past, refining strategies in the present, and gaining a competitive edge for the future of fixed income investing.

> "The overarching lesson is that careful backtesting builds confidence. When a strategy has survived harsh historical scenarios, passed stringent validation testing, and retained its edge even out-of-sample, traders and portfolio managers can approach live trading with greater conviction."

Ready to elevate your business?

Get started →

More than 750 clients and 90 partners across 52 countries rely on Numerix analytics. Headquartered in New York with offices around the world, Numerix brings together unparalleled expertise combined with local market understanding.

Numerix LLC Corporate Headquarters

100 Park Avenue
15th Floor
New York, NY 10017
Tel: +1.212.302.2220
Fax: +1.212.302.6934
www.numerix.com