Backtesting Framework Development Report

Phase 1: Mean Reversion System

Executive Summary

Successfully developed a comprehensive backtesting framework for systematic trading strategy evaluation. The framework includes a complete data pipeline processing 23M+ market records, modular strategy components, and detailed performance analytics. Initial testing of the Quick Panic ES mean reversion strategy demonstrates the framework's capabilities while revealing important insights about parameter optimization.

---

1. Data Infrastructure Development

Database Architecture

- **Schema Design**: Created normalized SQLite database with three core tables:
    - symbol: Tradeable instruments (ES, SPX, VIX, VX)
    - price_data: 17.8M historical price records across multiple timeframes
    - market_indicator: 5.4M+ indicator records (sentiment, breadth, positioning)

Data Migration Process

- **Processed File Types**: CSV, TXT, Excel, Parquet formats
- **Handled Complexities**:
    - European number formats (comma decimals)
    - Multiple datetime formats
    - Missing data interpolation
    - Timezone standardization
- **Performance**: Migration completed in ~45 minutes for 23M records using bulk inserts and optimized pragmas

Data Quality Enhancements

- Automated detection of file structures
- Validation of OHLC data integrity

- Preservation of original timestamps
- Merging of the data (to be used for backtesting) is done such that the most granular data is preserved with max data coverage and historical completeness (more details explained in Data_Merging_Explained doc)

---

2. Backtesting Framework Architecture

Modular Design Principles

├── Data Layer (SQL_Db_Migration.py)

├── Processing Layer (fast_merge_1min.py)

├── Signal Generation (Strategy_entry_signals.py)

├── Execution Engine (Strategy_backtesting.py)

└── Configuration (config.json)

Key Framework Features

1. Configuration-Driven Parameters

json

```
{"entry": {
   "es_decline_pct": -1,
   "vix_max": 25,
   "rsi2_min": 10 },
 "exit": { "stop_loss_pct": 1,
   "take_profit_pct": 3}}
```

2. Position Management System

- Prevents overlapping trades
- Tracks entry/exit timing precisely
- Handles partial fills and gaps
- Supports multiple exit conditions

3. Realistic Execution Modeling

- Transaction costs included
- Proper stop/target execution order
- No look-ahead bias
- Slippage considerations

---

3. Performance Analytics Engine

Comprehensive Metrics Calculation

- **Risk Metrics**: Maximum drawdown, consecutive losses, risk-adjusted returns
- **Performance Metrics**: Win rate, profit factor, expectancy, total return
- **Trade Analytics**: Duration analysis, exit reason breakdown, monthly performance
- **Statistical Measures**: Sharpe ratio, average win/loss, win/loss ratio

Automated Reporting

- Real-time performance dashboard during backtesting
- CSV export of all trades with timestamps
- Summary statistics in multiple formats
- Parameter sensitivity analysis

---

4. Initial Strategy Testing Results (Performed over 12 years horizon – 2013-2024)

Quick Panic ES Strategy Overview

The strategy attempts to buy short-term dips in an uptrending market using 12 technical and sentiment filters:

- Technical: ES > 50 SMA, RSI(2) > 10, ES decline from 10-day high
- Volatility: VIX < 25, VX spike < 25%, VIX/VXV ratio < 1.0
- Sentiment: CNN Fear & Greed > 35, NAAIM > 40
- Market Structure: Breadth > 40%, TRIN < 1.5
- Macro: Fed stance ≤ 2, Buffett Indicator < 200%

Parameter Optimization: ES Decline Threshold

Testing revealed significant sensitivity to the decline threshold parameter:

ES Decline	Trades	Win Rate	Profit Factor	Total Return	Max Drawdown
-1%	90	30.0%	1.23	+13.78%	-11.52%
-2%	32	25.0%	0.92	-2.17%	-10.72%
-3%	6	0.0%	0.00	-6.08%	-5.08%
-4%	1	0.0%	0.00	-1.05%	0.00%

Exit Analysis

- **Stop Loss Exits**: 59% of trades hit 1% stop loss
- **Take Profit Exits**: 29% reached 3% target
- **MA Cross Exits**: 12% exited on technical signal

---

5. Key Insights and Observations

1. Over-Parameterization Challenge

The current strategy requires ALL 12 conditions to align simultaneously. This creates several issues:

- **Redundant Filters**: Many indicators likely provide overlapping signals
- **Lagging Indicators**: Fed stance and NAAIM update weekly/monthly, missing intraday opportunities
- **Binary Approach**: All-or-nothing logic ignores partial setups

2. Recommended Filter Prioritization

Based on the backtest data, consider focusing on core filters and removing redundant ones:

Keep (High Impact):

- ES > 50 SMA (trend filter)
- ES decline % (entry trigger)
- VIX level (market regime)
- RSI(2) (oversold condition)

We can consider removing:

- Fed stance (too slow-moving)
- Buffett Indicator (quarterly updates, potentially stale)
- VX spike (may duplicate VIX information)

3. Profit Target Optimization

The original 5% profit target proved unrealistic for mean reversion trades:

- **5% Target Results**: 71 trades, 11.3% win rate, -25% return, 0.56 profit factor
- **3% Target Results**: 90 trades, 30% win rate, +13.8% return, 1.23 profit factor
- **Key Finding**: Winners with 5% target averaged 436 hours (18 days) holding time, indicating the strategy was trying to capture swing moves rather than mean reversion bounces
- **Conclusion**: 3% target aligns better with the intraday/short-term nature of panic dip buying, converting a losing strategy into a profitable one

4. Stop Loss Challenges

With 59% of trades hitting stops, the 1% stop loss appears too tight for the volatility following market declines. The strategy essentially buys into weakness with insufficient room for normal fluctuations.

---

6. Framework Advantages & Next Steps

Framework Strengths

1. **Modularity**: Easy to swap strategies, add indicators, or modify logic
2. **Transparency**: Every trade decision is traceable and auditable
3. **Efficiency**: Processes millions of records in seconds
4. **Extensibility**: Ready for additional asset classes and strategies

Recommended Enhancements

1. Scoring System Implementation Instead of requiring all conditions, implement a weighted scoring approach:

python

```
score = 0
```

```
if ES > SMA_50: score += 3  # Core condition
if VIX < threshold: score += 2  # Important
if NAAIM > 40: score += 1  # Nice to have
# Trade if score >= minimum_threshold
```

---

7. Deliverables Summary

1. **Complete Codebase**: Modular Python framework with configuration files
2. **Documentation**: Setup instructions, code comments, and this report
3. **Analysis Tools**: Performance metrics, trade logs, parameter comparison
4. **Future-Ready**: Framework prepared for further backtesting

---

Conclusion

While the Quick Panic ES strategy shows modest positive returns with a -1% decline threshold, the analysis reveals opportunities for improvement through filter optimization and alternative entry/exit logic. The modular framework design enables rapid iteration and testing of these improvements in future phases.

The infrastructure is now in place for systematic strategy development, providing transparency into what drives performance and enabling data-driven decision making for trading system design.