# Strategy Report for Statistical Mean Reversion Bot

## How I Explored the Dataset

To build a robust trading bot, I began by exploring the historical order book data using tools like `pandas`. Specifically, I:

- Inspected buy and sell order depth snapshots across multiple timestamps.
- Visualized best bid, best ask, and the mid-price to identify stable pricing behavior.
- Monitored the bid-ask spread distribution over time.
- Measured short-term volatility using rolling standard deviation of mid-prices.

From this exploration, I noticed:

- The mid-price tends to mean-revert over short windows.
- The spread fluctuates, but frequently narrows to a tight band.
- Statistically significant deviations from the average mid-price could signal profitable entry/exit points.

## What Inspired the Strategy

This strategy was influenced by classic **mean-reversion** theory and short-term **microstructure-based trading**. Key inspirations include:

- The idea that asset prices often revert to a local average over time.
- The use of standard deviation as a proxy for volatility to avoid reacting to random noise.
- A spread filter to trade only when there's real opportunity, reducing unnecessary churn.

## Core Logic of the Algorithm

The bot executes a form of statistical arbitrage by:

1. Tracking the rolling window of mid-prices (average of best bid and ask).
2. Computing the mean and standard deviation of the mid-prices.
3. Defining upper and lower thresholds:

$$\text{Upper Threshold} = \mu + 1.5\sigma, \quad \text{Lower Threshold} = \mu - 1.5\sigma$$

4. If the current best ask is below the lower threshold, the bot issues a buy order.
5. If the current best bid is above the upper threshold, it sells.
6. Ensures the bot stays within position limits ($\pm 50$).
7. Ignores trades when spread is too tight ($<2$), to reduce noise trading.

## Experiments, Variations, and Insights

During development, I experimented with several variants:

- **Window Size:** Smaller windows (10) were too reactive; larger ones (30+) lagged. A window of 20 provided a balanced response.
- **Standard Deviation Multipliers:** Values below 1.5 led to frequent false signals. 1.5 produced fewer but higher-confidence trades.
- **Spread Threshold:** Requiring a minimum spread of 2 helped avoid noise and slippage losses.
- **Mid-price Storage:** Replacing manual list popping with `collections.deque` improved performance and simplicity.

**Key Insights:**

- Not every price deviation is trade-worthy—adding a volatility buffer is essential.
- Avoiding trades in narrow spreads significantly improved PnL stability.
- Hard position limits keep the strategy safe under rapid fluctuations.

## Conclusion

This statistical mean reversion bot is designed to make informed, risk-aware trades based on short-term deviations from a stable mid-price. It incorporates spread filtering, volatility control, and strict position limits to minimize downside risk and maximize opportunities from mispricings. It performs best in markets where mid-price oscillations are common but spreads are not persistently wide or thin.