Part I: Determining the Optimal Fraction of Capital Risked in a Trading Strategy using the Kelly Criterion *(based on the author's IB Mathematics AA HL Internal Assessment)*

Part II: Blackbox: Testing an Estimated Edge in Forward Trading.

Table of Contents

1. Introduction and aim……...3
2. Mathematical foundations……..3
3. Deriving the Kelly fraction……..4
4. Simulation using historical SPY data……...5
5. Applying the model to the simulated trade sequence……...6
6. Threshold win-count analysis……7
7. Leverage implication of stop-loss sizing……10
8. Volatility Condition……11
9. Reflection and Uncertainty……..17
10. Part I Conclusion……..……21
11. From a known edge to an estimated one……..22
12. The Blackbox system and data……..23
13. Experiment A | Finding and validating an edge……23
14. Experiment B Reconciling the backtest with the forward test……25
15. Experiment C | Volatility-adaptive rules……26
16. Experiment D | Kelly optimisation for realised R-multiples……27
17. Discussion and Limitations……..30
18. Combined Conclusion……...30
19. Bibliography……...31
20. Appendix……32

1. Introduction and aim

In trading and quantitative finance, one of the most important questions is not only whether a strategy has an edge—some type of advantage—but how much capital should be risked when that supposed edge appears to exist. My fascination with trading and finance arose from my parents' careers in finance. Both of them worked at Lehman Brothers when it famously went bankrupt during the Global Financial Crisis in 2008 (Wiggins et al., 2014), and they, as a result, have made undoubtedly sure that I am cognizant of risk's significance in finance. As I have grown up and developed into an aspiring trader, I have become increasingly interested in this problem, as I have repeatedly seen trading discussions focus inordinately on finding entries, while treating position risk sizing as an afterthought—much to my parents' distaste. Mathematically, however, the size of the position can actually dominate final wealth outcomes. A trader with an overwhelming positive edge can still lose money if they overbet, while a trader who risks too little may barely capitalise on the edge at all (MacLean et al., 2010).

The aim of this exploration is to model long-run logarithmic growth of a fixed-rule SPY (S&P 500, exchange-traded fund tracking the performance of the top 500 public companies in the US) trading strategy as a function of the fraction of capital risked per trade, locate the fraction that maximises that model using differentiation, and evaluate whether the theoretical optimum is practically feasible once sampling uncertainty in the win probability, the leverage implied by a tight stop-loss and the stability of the edge across volatility conditions are taken into account.

I first build the wealth model and derive the Kelly fraction from expected logarithmic growth. I then apply it to simulated trade outcomes from historical SPY (S&P 500) data, compare no-leverage growth with theoretical Kelly growth, and investigate whether the number of observed wins is sufficient for profitability. Finally, I reflect mathematically on the model's limitations, particularly the uncertainty in the estimated win rate and the effective leverage implied by tight stop-losses.

Part II (sections 11-18) extends this exploration. It replaces the random-direction system with Blackbox, an automated trading system and forward-tested on OANDA demo accounts, and asks whether an edge estimated from historical data survives forward testing, and what the Kelly criterion implies when it does not.

2. Mathematical foundations

Before deriving the Kelly Criterion, I define the variables and assumptions used in the model. I assume each completed trade has only two possible outcomes: a win or a loss. A win—price hitting the take profit target—earns R times the amount risked, while a loss—price reaching stop-loss level—loses the amount risked. This simplifies the trading problem into a repeated probabilistic model, which makes it possible to analyse long-term growth mathematically.

The key variable is f, the fraction of current wealth risked on each trade. It is important that f represents capital risked, not necessarily the amount of capital allocated to the position. This distinction becomes important later when I analyse the effect of stop-loss distance and effective leverage. The win

probability is denoted by p, so the probability of a loss is $q = 1 - p$. The reward-to-risk ratio is denoted by R. In my simulation, $R = 1.5$ because the take-profit distance is 0.375% and the stop-loss distance is 0.25%.

The reason logarithms are used is that wealth changes multiplicatively over repeated trades. A sequence of percentage gains and losses compounds through multiplication, while logarithms convert these multiplicative changes into sums. This allows the average growth rate per trade to be analysed using a single function, G(f), rather than by tracking every possible order of wins and losses individually.

Symbol	Meaning
W_0	initial wealth
W_n	wealth after n completed trades
f	fraction of current wealth risked on each trade
p	probability of a winning trade
$q = 1 - p$	probability of a losing trade
R	reward-to-risk ratio
k	number of winning trades out of n
SL	stop-loss distance as a decimal fraction of price

If a trade wins, the wealth is multiplied by $1 + fR$. If a trade loses, the wealth is multiplied by $1 - f$. Therefore, after n trades with k wins and $n - k$ losses, the wealth is:

$$W_n = W_0(1 + fR)^k(1 - f)^{(n-k)} \quad (1)$$

This model is multiplicative rather than additive. This matters because percentage gains and losses do not cancel symmetrically: losing 50% and then gaining 50% leaves only 75% of the original wealth (Thorp, 2006). Therefore, long-run growth should be measured using logarithms of wealth, not simple arithmetic averages.

$$ln(\frac{W_n}{W_0}) = k\, ln(1 + fR) + (n - k)\, ln(1 - f) \quad (2)$$

Dividing by n gives average log-growth per trade. If the number of trades is large, by the Law of Large Numbers, the proportion of wins $\frac{k}{n}$ approaches p, so the expected log-growth per trade is:

$$G(f) = p\, ln(1 + fR) + (1 - p)\, ln(1 - f) \quad (3)$$

Thus, maximising expected log-growth is appropriate because it maximises the long-run geometric growth rate rather than the arithmetic average return of a single trade (Breiman, 1961; Latané, 1959).

3. Deriving the Kelly fraction

To find the value of f that maximises long-term growth, I differentiate the expected log-growth function from equation (3), while assuming p and R are constant. Because $G(f)$ is differentiable, an interior maximum can only occur at a stationary point, so I solve $G'(f) = 0$ and use $G''(f)$ to confirm it is a maximum.

$$\frac{d}{df}[G(f) = p\, ln(1 + fR) + (1 - p)ln(1 - f)]$$

$$G'(f) = \frac{pR}{(1 + fR)} - \frac{(1 - p)}{(1 - f)} \quad (4)$$

At the maximum, the derivative equals zero:

$$\frac{pR}{(1 + fR)} = \frac{(1 - p)}{(1 - f)}$$

$$pR(1 - f) = (1 - p)(1 + fR)$$
$$pR - pRf = (1 - p) + (1 - p)fR$$
$$pR - (1 - p) = fR[p + (1 - p)]$$
$$pR - (1 - p) = fR$$

$$f^* = \frac{pR - (1 - p)}{R} = \frac{p(R + 1) - 1}{R} \quad (5)$$

This is the Kelly fraction (Kelly, 1956; Rotando & Thorp, 1992) for a binary win/loss trade with reward-to-risk ratio R.

To confirm that this critical point is a maximum, I differentiate again to get the second derivative:

$$G''(f) = \frac{-pR^2}{(1+fR)^2} - \frac{1-p}{(1-f)^2} \quad (6)$$

For $0 < f < 1$, both terms in equation (6) are negative, as $p > 0, R > 0$, so, $G''(f) < 0$, and $G(f)$ is concave down. Thus, the critical point in equation (5) is a maximum (Stewart, 2015).

4. Simulation using historical SPY data

I used historical SPY 5-minute OHLC candle data from 2026-02-19 to 2026-05-14, covering 60 US trading days and 4,663 candles during regular market hours (Yahoo Finance, 2026). To avoid building a directional trading strategy into the model, the trade direction was random: for each selected entry candle, a value between 0 and 1 was generated, below 0.5 produced a long trade, while a value of 0.5 or above produced a short trade. Building the simulation myself let me test position sizing separately from entry selection, rather than relying on a standardised backtest. The full data-retrieval and simulation code is in Appendix F. This exploration used AI assistance (Anthropic, 2026) to scaffold the Python simulation, to compute several resulting metrics in the table below, and to render the figures in the paper. All modelling choices are my own: the trading rules, the 0.25% stop-loss and 0.375% take-profit, the random long/short 200-bucket entry selection, the first-touch win/loss rule, the random seed, and the ATR and Parkinson volatility measures all below. I derived the mathematics, verified every numerical result independently, and interpreted all outputs. The AI assistance used in Part II is described separately in Section 12.5 and Appendix I.

Entry candles were selected using a 200-bucket method with random seed 42, making the simulation reproducible. The resulting figures are summarised in Table 1. The stop-loss was set at 0.25% from the entry price and the take-profit at 0.375%, so the reward-to-risk ratio was fixed at $R = 1.5$. A trade was recorded as a win if the take-profit was touched first, and as a loss if the stop-loss was touched first. If neither level was touched within 78 candles, approximately one trading session, the trade was marked unresolved and excluded from the calculation of p and R (Pardo, 2008).

Simulation results

Table 1. Simulation summary results. Source: author's calculations. Full breakdown in Appendix A

Metric	Value	Interpretation
Total simulated trades	183	17 of the 200 selected candidates were blocked by active unresolved trades
Completed trades	178	Only wins and losses enter p and R
Wins / losses	80 / 98	Used to estimate win probability

Unresolved / ambiguous	5 / 0	Excluded from p and R
Empirical win rate $\hat{p}$	0.4494	Calculated from $\frac{80}{178}$
Reward-to-risk ratio R	1.5	Fixed by $\frac{0.375\%}{0.25\%}$
Breakeven p^*	0.4000	$p^* = \frac{1}{1+R}$
Kelly fraction f^*	0.0824	Optimal theoretical risk fraction

The empirical win rate was:

$$\hat{p} = \frac{80}{178} = 0.4494 \quad (7)$$

Since the breakeven win rate is:

$$p = \frac{1}{1+R} = \frac{1}{2.5} = 0.4000 \quad (8)$$

the observed win rate is above the breakeven threshold. Substituting $\hat{p} = 0.4494$ and $R = 1.5$ into equation (5):

$$f^* = \frac{[0.4494(2.5)-1]}{1.5} = 0.0824 \quad (9)$$

Therefore, under the model, the Kelly-optimal fraction is approximately 8.24% of current capital risked per trade. This maximum is shown in Figure 1. The full Kelly-parameter breakdown is in Appendix B.

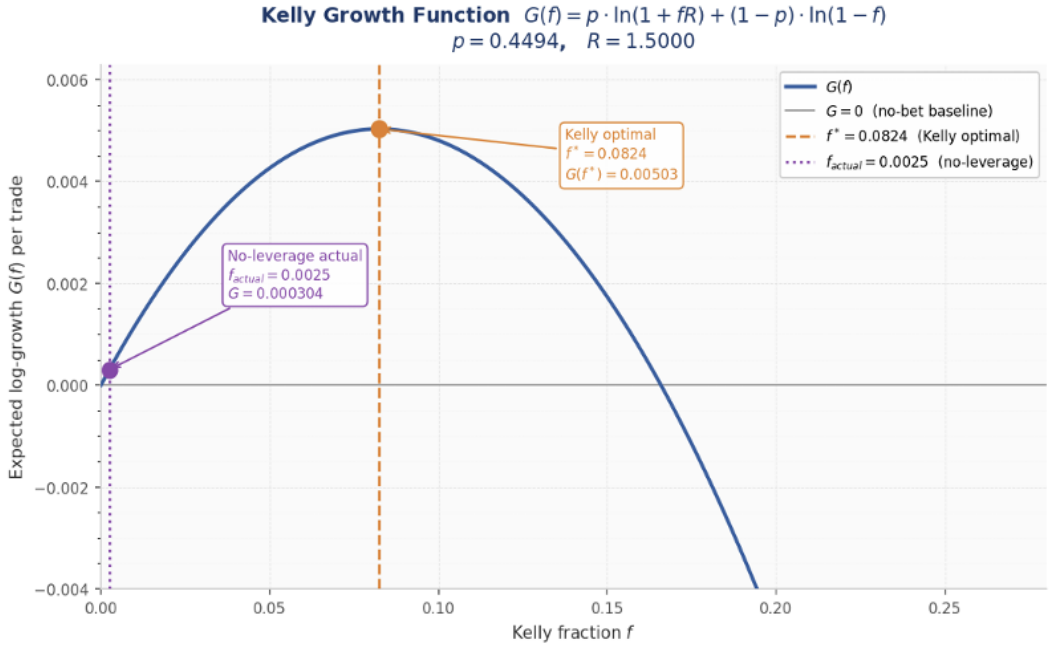

Figure 1. Expected log-growth G(f) using the empirical win rate and R = 1.5. The maximum occurs at f* = 0.0824. Source: author's calculations

5. Applying the model to the simulated trade sequence

The same sequence of 80 wins and 98 losses can be replayed using different fractions f. This separates the trade outcomes from the position-sizing rule. In the realistic no-leverage account path, a loss reduces the account by 0.25%, and a win increases it by 0.375%:

Win: $W_{n+1} = W_n(1.00375)$

Loss: $W_{n+1} = W_n(0.9975)$

Using this no-leverage model, the account grew from $100,000 to approximately $105,562.41, a gain of about 5.56% over the 178 completed trades. The full account path is in Appendix C.

The theoretical Kelly wealth curves instead replay the same win/loss sequence using $f = \frac{f*}{2}, f = f*$, and $f = 2f*$. These are not the same as the no-leverage simulation. They ask a different mathematical question: what would have happened if the same trade sequence had been sized using half Kelly, full Kelly, or double Kelly (MacLean et al., 2011)?

Full Kelly win: $W_{n+1} = W_n(1 + 0.0824 \times 1.5) = W_n(1.1236)$

Full Kelly loss: $W_{n+1} = W_n(1 - 0.0824) = W_n(0.9176)$

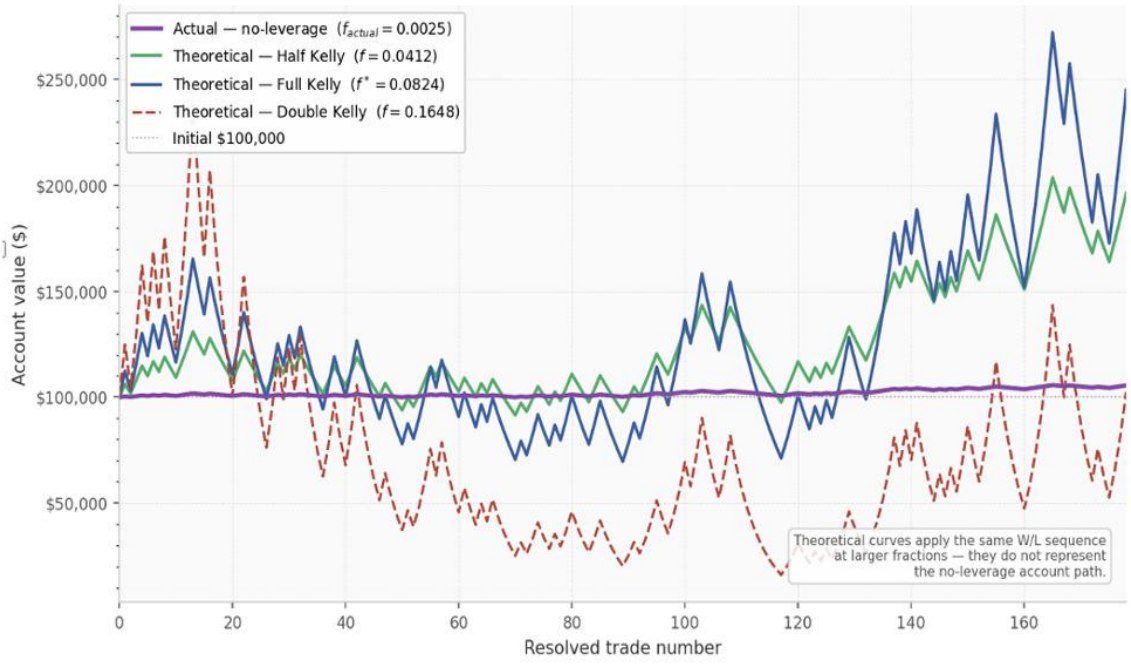

Figure 2. Theoretical wealth curves using half Kelly, full Kelly, and double Kelly on the same resolved trade sequence. Source: author's calculations

Figure 2 compares the realistic no-leverage account path with theoretical Kelly reinvestment paths. The no-leverage path uses the actual SPY price movement of 0.25% loss or 0.375% gain per resolved trade. The Kelly curves replay the same win/loss sequence using fractional risk levels $\frac{f^*}{2}, f^*, and\ 2f^*$, so they are theoretical sizing comparisons rather than directly implemented no-leverage trades. This comparison is important because full Kelly is not simply a larger version of the no-leverage model. The no-leverage path uses $f = 0.0025$, while full Kelly uses $f = 0.0824$. Overbetting occurs when f is pushed beyond the value that maximises G(f). In the wealth curves, double Kelly deliberately tests this risk by using $2f^* = 0.1648$, so each loss removes 16.48% of the account. This makes the curve far more volatile and demonstrates that higher risk does not always mean higher long-term growth (MacLean et al., 2010).

6. Threshold win-count analysis

To test how strong the simulated result really was, I derived the minimum number of wins needed for wealth to finish above its starting value after n trades. Starting with equation (1), profitability requires:

$$W_n > W_0$$

Substituting equation (1) for W_n yields:

$$W_0(1 + fR)^k(1 - f)^{(n-k)} > W_0$$

$$(1 + fR)^k(1 - f)^{(n-k)} > 1$$

Taking natural logarithms of both sides is valid because $ln(x)$ is increasing for $x > 0$.

$$k\, ln(1 + fR) + (n - k)\, ln(1 - f) > 0$$

$$k[ln(1 + fR) - ln(1 - f)] > -n\, ln(1 - f)$$

$$k > \frac{-n\, ln(1 - f)}{[ln(1 + fR) - ln(1 - f)]} \qquad (10)$$

Using $n = 178$, $f = 0.0824$, and $R = 1.5$ gives:

$$k > 75.58$$

Therefore, at least 76 wins are required for the full Kelly strategy to finish above the starting wealth. The simulation had 80 wins, so it exceeded this threshold by only 4 wins. This is a small margin, suggesting that the result is profitable under the model but not overwhelmingly secure.

Self-consistency of the Kelly fraction

A useful check is whether the Kelly fraction itself guarantees that the assumed win probability is above the profitability threshold for that fraction. If f^* is the maximiser and $f^* > 0$, then the maximum log-growth must be positive:

$$p\, ln(1 + f^*R) + (1 - p)\, ln(1 - f^*) > 0$$

$$p[ln(1 + f^*R) - ln(1 - f^*)] > -ln(1 - f^*)$$

$$p > -\frac{ln(1 - f^*)}{[ln(1 + f^*R) - ln(1 - f^*)]} \qquad (11)$$

The expression above is exactly the minimum win proportion required for profitability from equation (10). Therefore, whenever Kelly recommends a positive fraction, the input win probability must exceed the minimum threshold for growth. This proves that the Kelly formula is internally consistent: it does not recommend a positive bet unless the win probability is mathematically sufficient to make growth possible (Rotando & Thorp, 1992).

Implementing the numbers from the trade simulation ($f = 0.0824$, and $R = 1.5$) yields:

$$p > -\frac{ln(1 - 0.0824)}{[ln(1 + 0.0824(1.5)) - ln(1 - 0.0824)]}$$

$$p > 0.425$$

Thus, with the trade simulation's conditions, the win probability or proportion must be above ~42.5% for profitability, or wealth growth. This is different to the 40% breakeven at which the optimal fraction itself first becomes positive, because 8.24% is calibrated to $\hat{p} = 0.4494$.

6.1 Binomial interpretation of the number of wins

The threshold result above gives the number of wins required for profitability. However, the actual number of wins in a fixed number of trades is itself random. In our case, if X represents the number of winning trades in n completed trades, then X can be modelled by a binomial distribution (Grinstead & Snell, 1997):

$$X \sim B(n, p)$$

$$P(X = k) = \binom{n}{k} p^k(1 - p)^{(n-k)} \qquad (11a)$$

This connects directly to the wealth model because each possible value of k produces a different final wealth:

$$W_n = W_0(1 + fR)^k(1 - f)^{(n-k)}$$

For my simulation, n = 178 and $\hat{p}$ = 0.4494. Therefore, the expected number of wins is:

$$E(X) = np = 178(0.4494) = 80.0$$

This matches the observed 80 wins. However, the variance of a binomial random variable is:

$$Var(X) = np(1 - p) = 178(0.4494)(0.5506) \approx 44.0$$

so the standard deviation is:

$$\sigma_X = \sqrt{44} \approx 6.63\ wins$$

This means that, even with the same underlying probability, the observed number of wins could naturally vary by several trades. Since the full Kelly profitability threshold was 76 wins and the simulation produced 80 wins, the margin above the threshold was only about four wins, which is less than one standard deviation. This supports the reflection that the positive result should be interpreted cautiously rather than as proof of a stable trading edge.

By summing the binomial probabilities of exactly 76 wins, then exactly 77, then 78, up to 178, beyond the margin itself, the binomial model gives the probability of clearing the threshold:

$$P(X \geq 76) = \sum_{k=76}^{178} \binom{178}{k} (0.4494)^k (0.5506)^{(178-k)} \approx 0.750$$

So even if the estimated win rate were exactly correct, full Kelly still carries roughly a one-in-four chance of finishing below its starting value; half Kelly, needing only 74 wins, using Half Kelly, raises the probability of profit to

$$P(X \geq 74) \approx 0.836$$

Both assume $p = 0.4494$ is the true win probability, so given the uncertainty in 9.1 even 0.75 is hopeful.

6.2 Consecutive-loss drawdown check

A final way to interpret risk is to ask how many consecutive losses would reduce the account below half of its starting value. With a fixed risk fraction f, after m consecutive losses:

$$W_m = W_0(1 - f)^m$$

Setting $W_m = 0.5W_0$ gives:

$$W_0(1 - f)^m = 0.5W_0$$
$$(1 - f)^m = 0.5$$

$$m = \frac{\ln(0.5)}{\ln(1-f)} \qquad (11b)$$

For full Kelly, $f = 0.0824$:

$$m = \frac{\ln(0.5)}{\ln(1-0.0824)} \approx 8.05$$

So approximately nine consecutive losses (as it is above 8) would reduce the account below half of its starting value (MacLean et al., 2010). This does not mean that nine losses are likely in a row, but it shows why a strategy with positive expected log-growth can still involve severe drawdown risk—a further reason to size below full Kelly, since at half Kelly the same halving would require roughly seventeen consecutive losses. Also, although multiplication means the final product depends mainly on the set of win and loss multipliers, the order of trades still matters for the wealth path and drawdowns. A sequence of early losses can reduce capital substantially before later wins occur, even if the final number of wins is unchanged. Connecting to section 8.1, for a sequence of multipliers M_i:

$$W_n = W_0 \prod_{i=1}^{n} M_i$$

Where:

$$M_i = \begin{cases} 1 + f_i R, \, if \; trade \; i \; is \; a \; win \\ 1 - f_i, \; if \; trade \; i \; is \; a \; loss \end{cases}$$

This gives the final wealth (reference section 8.1 for further context).

Because multiplication is commutative, if the set of multipliers is exactly the same, then the final product is unchanged by reordering. But drawdown depends on order. So, wealth after t trades:

$$W_t = W_0 \prod_{i=1}^{t} M_i$$

And maximum drawdown:

$$D_{max} = max_{1 \le t \le n}(1 - \frac{W_t}{max_{1 \le j \le t} W_j})$$

This depends on the order, as W_t changes after each trade. This matters because drawdown is based on intermediate wealth values, not only the final result.

7. Leverage implication of stop-loss sizing

The most important insight I reached during the investigation was that the Kelly fraction is a fraction of capital risked, not a fraction of capital allocated to the position (Thorp, 2006). This distinction is minor in gambling, where the stake is usually the amount at risk, but it is crucial in trading because a stop-loss can be much smaller than the account value.

If a trader risks fraction f of account value and uses a stop-loss distance SL, then the required notional exposure relative to account value is:

$$\text{Effective exposure} = \frac{f}{SL} \qquad (12)$$

In my simulation, the stop-loss distance was 0.25%, or 0.0025 as a decimal. Therefore:

$$\text{No leverage implementation: } \frac{0.0025}{0.0025} = 1:1$$

$$\text{Original 1\% fixed-risk idea: } \frac{0.0100}{0.0025} = 4:1$$

$$\text{Full Kelly implementation: } \frac{0.0824}{0.0025} = 32.96:1$$

This means that although full Kelly was mathematically optimal under the model, implementing it with a 0.25% stop-loss would require about 33:1 effective exposure. This is practically unrealistic for most traders. More importantly, it shows that leverage does not need to be artificially added to the Kelly Criterion: in trading, leverage can emerge naturally from the relationship between the chosen risk fraction and the stop-loss distance, and is therefore, internally implicated in the Kelly Criterion's construction (Thorp, 2006). This relationship can also be rearranged to find the minimum stop-loss distance required to keep effective exposure below a chosen leverage bound L_{max}:

$$SL_{\min} = \frac{f_{\text{Kelly}}}{L_{\max}}$$

For example, if a trader wanted to keep effective exposure below 10: 1, then:

$$SL_{\min} = \frac{0.0824}{10} = 0.00824 = 0.824\%$$

If the maximum allowed exposure were 30:1, then:

$$SL_{\min} = \frac{0.0824}{30} = 0.00275 = 0.275\%$$

This shows that the Kelly fraction, although derived without explicitly mentioning leverage, creates practical constraints on stop-loss calibration and feasible exposure.

Practical responses to this may include exposure caps, fractional Kelly use, or both.

8. Volatility condition

Because the stop-loss and take-profit distances must be appropriate for the timescale of the data, I used ATR-based volatility buckets to check whether the win rate varied under different volatility conditions. ATR is a realised volatility measure calculated from high-low ranges and previous closes (Wilder, 1978). It is not implied volatility, which would come from options prices—a type of financial derivative (Hull, 2018). ATR was more suitable here because the simulation itself uses historical candle ranges. The ATR at entry was calculated for each position entry—the first and last five completed trades are shown in Table 2. The volatility condition was characterized as either low, medium, or high depending on the ATR value when the position was entered. Win rates across the three ATR conditions are shown in Figure 3.

Table 2. Abridged simulation output; full 183-trade log in Appendix E

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
1	2026-02-19	09:40	SHORT	0.908581	684.6100	686.3215	682.0427	682.0427	Win	0.3750	$100'000.00	$100'375.00	0.000000	N/A
2	2026-02-19	13:20	SHORT	0.699707	682.3750	684.0809	679.8161	684.0809	Loss	-0.2500	$100'375.00	$100'124.06	0.940700	High
3	2026-02-20	09:30	LONG	0.265870	683.3200	681.6117	685.8824	685.8824	Win	0.3750	$100'124.06	$100'499.53	0.796882	Med
4	2026-02-20	10:05	SHORT	0.969176	687.0900	688.8077	684.5134	684.5134	Win	0.3750	$100'499.53	$100'876.40	1.322045	High
5	2026-02-20	11:25	SHORT	0.778751	686.1496	687.8650	683.5765	687.8650	Loss	-0.2500	$100'876.40	$100'624.21	1.327916	High
...	*trades 6–178 omitted — full log in Appendix E*													...
179	2026-05-12	11:30	SHORT	0.875005	732.8700	734.7022	730.1217	734.7022	Loss	-0.2500	$104'907.56	$104'645.30	0.814402	Med
180	2026-05-12	13:25	SHORT	0.851132	734.3500	736.1859	731.5962	736.1859	Loss	-0.2500	$104'645.30	$104'383.68	0.675633	Med
181	2026-05-12	14:30	LONG	0.043475	736.1200	734.2797	738.8804	738.8804	Win	0.3750	$104'383.68	$104'775.12	0.589056	Low
182	2026-05-13	10:55	LONG	0.181498	738.8300	736.9829	741.6006	741.6006	Win	0.3750	$104'775.12	$105'168.03	0.684330	Med
183	2026-05-13	12:25	LONG	0.236745	741.7250	739.8707	744.5065	744.5065	Win	0.3750	$105'168.03	$105'562.41	0.572573	Low

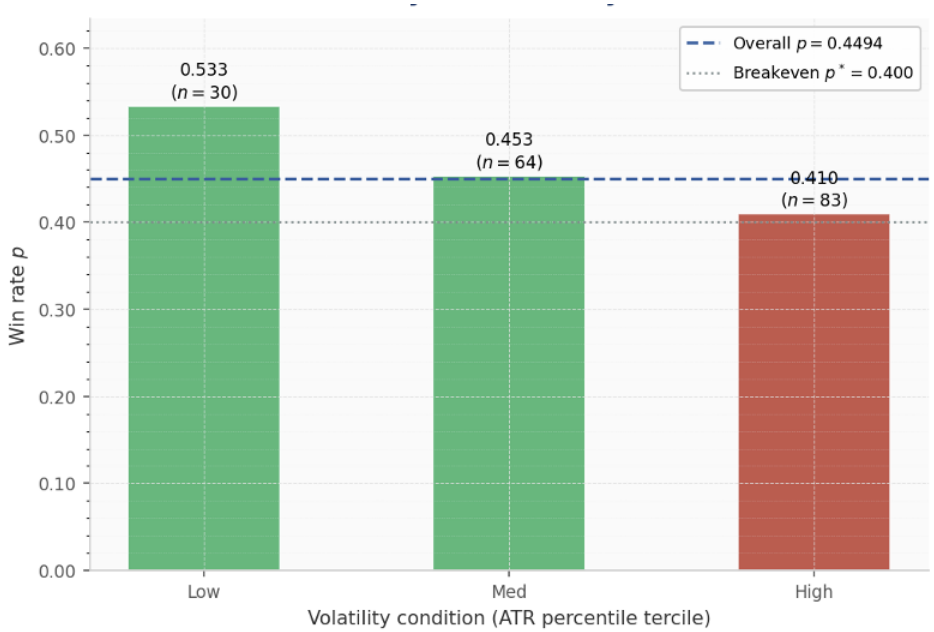

Figure 3. Win rate by ATR volatility condition. Low and medium ATR conditions outperformed the overall win rate, while high ATR conditions underperformed. Source: author's calculations

The ATR-14 cutoffs were the 33rd and 67th percentiles of all non-zero ATR values across the 4,663-candle dataset, ensuring roughly equal candle representation in each bucket. This gave three conditions: Low (ATR ≤ 0.6391, approximately ≤ 0.11% of SPY price per bar), Medium (0.6391 < ATR ≤ 0.9177, approximately 0.11–0.16% per bar), and High (ATR > 0.9177, approximately > 0.16% per bar). The full ATR range was 0.22 to 2.31 with a mean of 0.81. Table 3 shows the Kelly parameters computed separately for each condition.

Table 3. Kelly parameters by ATR volatility condition.

ATR Condition	Trades	Wins	Losses	p̂	R	f*	G(f*)
Low	30	16	14	0.5333	1.5000	0.2222	0.036150
Medium	64	29	35	0.4531	1.5000	0.0885	0.005806
High	83	34	49	0.4096	1.5000	0.0161	0.000193
Overall	178	80	98	0.4494	1.5000	0.0824	0.005000

Trade 1 has no ATR (fewer than 14 prior candles), so the buckets total 177 trades.

Low-ATR condition has the strongest Kelly fraction ($f^* = 0.2222$) and highest expected log-growth ($G(f^*) = 0.03615$), driven by a win rate of 0.5333 that comfortably exceeds the 0.400 breakeven. The Medium condition is close to the overall result ($f^* = 0.0885$, $G = 0.00581$). The High condition barely registers a positive Kelly fraction ($f^* = 0.0161$, $G = 0.000193$): its win rate of 0.4096 is only marginally above breakeven, and the expected growth per trade is negligible. This ordering is consistent with the hypothesis that high volatility degrades entry precision: when bars are large relative to the fixed 0.25% stop-loss and 0.375% take-profit distances, random intrabar moves are more likely to trigger stops prematurely. A volatility-regime-conditional Kelly strategy — sizing down in High-ATR sessions — would be one practical implication of this finding.

The volatility split suggests that the assumption of constant p may be weak. The strategy performed differently across ATR conditions, meaning the empirical win probability was not necessarily stable across market environments. This supports the later reflection that a fixed Kelly fraction may be too aggressive when p is estimated from a limited sample. Furthermore, the data follows an apparent negative correlation (though section 8.2 tests whether this survives a more principled volatility measure)—as the volatility condition increases the win rate p decreases, highlighting an interesting effect that volatility has on win rate.

8.1 Volatility-Adjusted Kelly Fraction

The previous calculations have treated all completed trades as if they came from the same probability distribution, using one overall estimate of win probability:

$$\hat{p} = \frac{80}{178} = 0.4494$$

However, this assumes that the probability of a winning trade is constant across all market conditions, but the findings from section 8 suggest that the constant-probability assumption is too simple. The model becomes more realistic if the optimal fraction is allowed to vary with market volatility. Since market volatility changes over time, I separated the completed trades into the three ATR-based volatility regimes highlighted above. This allowed me to test whether the Kelly fraction changes when the win probability is estimated separately with each volatility condition. My hypothesis was unequivocally that it did.

Let A represent the ATR volatility condition of a trade. I define the estimated win probability as a function of volatility (from Table 3 above):

$$\hat{p}(A) = \begin{cases} 0.5333, & A = Low\ ATR \\ 0.4531, & A = Medium\ ATR \\ 0.4096, & A = High\ ATR \end{cases}$$

Since the reward-to-risk ratio remained fixed at:

$$R = 1.5$$

The Kelly fraction (5) can also be written as a function of volatility:

$$f^*(A) = \frac{2.5\hat{p}(A) - 1}{1.5}$$

Therefore:

$$f^*(A) = \begin{cases} \dfrac{2.5(0.5333) - 1}{1.5}, \\ \dfrac{2.5(0.4531) - 1}{1.5}, \\ \dfrac{2.5(0.4096) - 1}{1.5}, \end{cases} \begin{matrix} A = Low\ ATR \\ A = Medium\ ATR \\ A = High\ ATR \end{matrix}$$

This gives the volatility-adjusted Kelly fractions:

$$f^*(A) = \begin{cases} 0.2222, & A = Low\ ATR \\ 0.0885, & A = Medium\ ATR \\ 0.0161, & A = High\ ATR \end{cases}$$

Following this, a volatility-adjusted wealth model can be written by allowing each trade to use a different Kelly fraction depending on its volatility condition. Let A be the ATR condition of trade i, and define:

$$f_i = f^*(A_i)$$

Let:

$$X_i = \begin{cases} 1, if\ trade\ i\ is\ a\ win \\ 0, if\ trade\ i\ is\ a\ loss \end{cases}$$

This is important, as it forces the wealth to be multiplied by a value above 1, $(1 + f^*(A_i)R)^{X_i}$, if the trade is a win, as the exponent above the loss section becomes 0. The opposite happens when a loss occurs, as it is multiplied by a decimal value below 1, $(1 - f^*(A_i))^{(1-X_i)}$.

Then the wealth after n completed trades can be modelled as:

$$W_n = W_0 \prod_{i=1}^{n} (1 + f^*(A_i)R)^{X_i}(1 - f^*(A_i))^{(1-X_i)}$$

Extending the original fixed-fraction model (1):

$$W_n = W_0(1 + fR)^k(1 - f)^{(n-k)}$$

by replacing the constant fraction f with a volatility-dependent fraction $f^*(A_i)$.

The above multiplicative model means that starting from W_0, multiply the account by the correct win/loss multiplier for trade 1, then trade 2, then trade 3, all the way to trade n. This, then, allows for non-fixed values of f which is helpful, as volatility conditions are constantly changing. This is mathematically important because it shows that the Kelly Criterion does not have to produce one fixed position size. If the estimated win probability changes with volatility, then the optimal fraction should also change. So, the Kelly fraction can be a useful tool that changes on each trade.

Before applying this to the full sequence, a three-trade example clarifies how the product notation works. Suppose

$$W_0 = 100{,}000$$
$$R = 1.5$$

the first three trades are given in Table 4:

Table 4. Three-trade example of the volatility-adjusted wealth model. Source: author's calculations.

Trade	ATR regime	Result	$f^*(A_i)$
1	Low	Win	0.2222
2	High	Loss	0.0161
3	Medium	Win	0.0885

Then:

$$X_1 = 1,\ X_2 = 0,\ X_3 = 1$$

Using the model:

$$W_3 = 100{,}000\big(1 + 0.2222(1.5)\big)(1 - 0.0161)\big(1 + 0.0885(1.5)\big)$$
$$W_3 = 100{,}000(1.3333)(0.9839)(1.1328)$$
$$W_3 \approx 148{,}583$$

This example also shows why volatility-adjusted full Kelly would be aggressive in the low ATR condition.

As shown in Figure 4 below, applying the volatility-adjusted fractions across all 178 completed trades (trade 1, lacking an ATR regime, uses $f^* = 0.0824$) produces a terminal account of $489,710 (~+390%), against $244,891 for fixed full Kelly on the identical sequence: the regime-adjusted strategy roughly doubles terminal wealth by effectively taking on more risk in calm markets and lowering it in volatile ones. However, its maximum drawdown (61.1%) slightly exceeds full Kelly's (57.9%): the low-ATR fraction of 0.222 that drives most of the outperformance also deepens the worst peak-to-trough loss. Since both the regime thresholds and the per-regime win rates were only estimated from this sample, and the low-ATR fraction rests on only 30 trades, this figure ($489,710) is an in-sample upper bound rather than an achievable live result when extrapolated—the ATR regime of a bar is only clear after entry. Therefore, the advantage depends on the regime being known in advance, and in live trading volatility fluctuates in seconds.

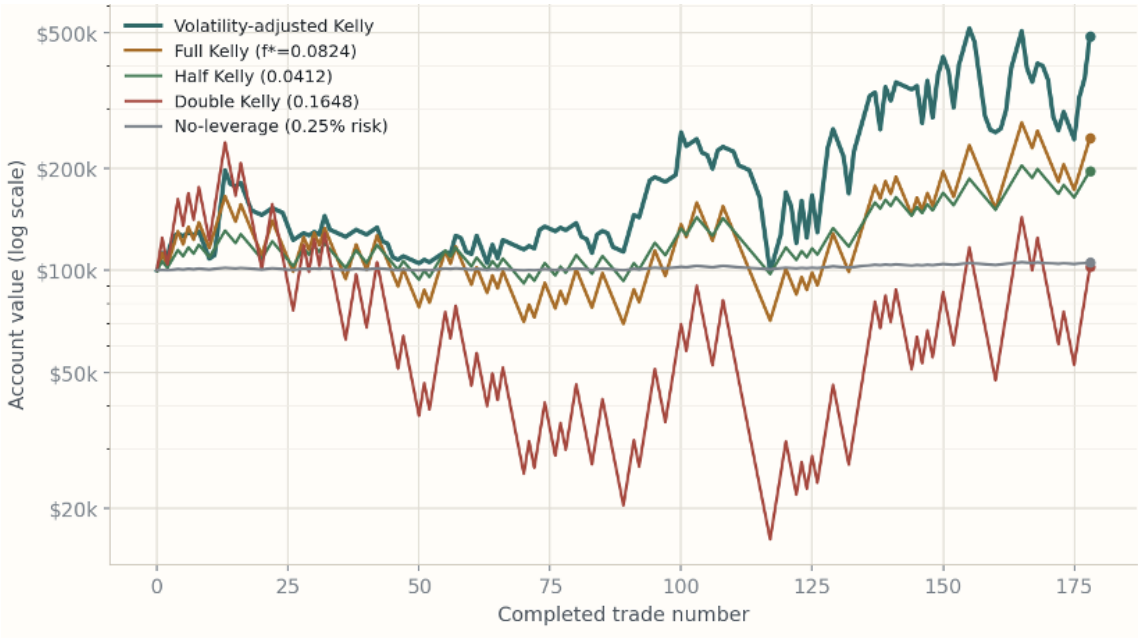

Figure 4. Wealth paths with volatility-adjusted Kelly versus fixed-fraction sizing

8.2 Brownian-motion model of the regime win rates

Section 8 showed the win rate falling as volatility rises. To judge whether this is a real effect or purely the result of random variation, I need a model of what the win rate should be, which first requires measuring volatility at entry accurately rather than through the ATR indicator.

I model the price after entry as geometric Brownian motion, the standard model of asset prices, where one part, $\mu S_t dt$, is predictable movement and one part represents random jitter, $\sigma S_t dW_t$ (Hull, 2018):

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

Or:

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t$$

Where S_t is the asset price at time t; dS_t its change over a small time step dt; μ is the drift, the average rate of return; σ is the volatility, measuring how much price fluctuates; and dW_t is the increment of a standard Brownian motion (called a Wiener process), which represents the random, unpredictable part of the price movement. Thus, an asset's price change is made up of a small expected return, μdt, plus a random shock, σdW_t. The size of that random component grows with volatility, and because the variance of that random component increases linearly with time (and volatility is the standard deviation of returns per unit of time), log-return variance over time t is $\sigma^2 t$. So, σ, the only unknown controlling how widely prices can move, is the crucial quantity I need to estimate at each entry.

I estimate σ from the high-low range of recent price bars using the Parkinson estimator derived from the same model (Parkinson, 1980):

$$\hat{\sigma}_P = \sqrt{\frac{1}{4\ln 2} \cdot \frac{1}{N} \sum_{i=1}^{N} (\ln \frac{H_i}{L_i})^2}$$

Where $\hat{\sigma}_P$ is the estimated volatility; N is the number of bars (price ranges within a specific time) in the look-back window; H_i and L_i are the high and low prices of bar i. The log ranges are squared because volatility is related to variance, and variance is squared deviations; and the constant $\frac{1}{4\ln 2}$ arises from the expected squared range of Brownian motion. Using the log ratio $\ln \frac{H_i}{L_i}$ rather than the raw difference, measures the percentage-like range rather than the raw dollar range, so a $5 move relative to a $500 asset is smaller than within a $50 one. Unlike ATR, which is a descriptive measure of price range, the Parkinson estimator is derived from a probabilistic price model and estimates volatility directly at entry.

To apply this at a specific entry, take trade 4, entered at 2026-02-20 10:05. Using the 14 five-minute candlestick bars immediately before entry, the Parkinson estimator gives

$$\hat{\sigma}_P = \sqrt{\frac{1}{4\ln 2} \cdot \frac{1}{14} \sum_{i=1}^{14} (\ln \frac{H_i}{L_i})^2} = 0.00142$$

0.142% estimated volatility per 5-minute bar. Annualising this volatility with 78 five-minute bars per session and 252 sessions,

$$\hat{\sigma}_P = 0.00142\sqrt{78 \cdot 252} \approx 0.199 \ (\approx 20\%)$$

For comparison, the cruder ATR-based approximation, ATR value as a fraction of price, yields 0.192% per bar, translating to roughly 27% annualised (this does not imply that volatility persists for a year, it simply shows the current volatility on a conventional annual scale). This runs higher than the Parkinson estimate because ATR is the average true range of each bar, and a bar's full range is always wider than a typical, standard-deviation sized move. Parkinson, by contrast, uses only the within-bar high-low range and multiplies by the factor $\frac{1}{4\ln 2}$, which converts that range into a standard deviation: so it is a properly scaled estimator rather than a range. Repeating the calculation at a low-volatility entry (trade 12) yields $\hat{\sigma}_P \approx 5.7\%$ annualised, nearly four times smaller, showing that volatility at entry genuinely varies.

Now the key prediction. Let TP = 0.375% be the take-profit distance, SL = 0.25% the stop-loss distance, and P the probability the price reaches the take-profit before the stop-loss (win rate). If the walk is driftless (μ is approximated as 0 because trade direction is randomly assigned long or short, so market drift should largely cancel across the strategy), the expected exit level must equal the entry level, so the trade exits at +TP with probability P and at -SL with probability 1-P:

$$P \cdot (TP) + (1 - P)(-SL) = 0$$

Solving for P gives:

$$P = \frac{SL}{TP + SL} = \frac{0.25}{0.625} = 0.400$$

So, independent of volatility σ, the driftless model predicts the same 0.400, 40% win rate in every volatility regime. My overall 0.4494 empirical win rate is only ~1.3 standard errors above this, so on aggregate the data is consistent with no effective edge.

Section 8's ATR buckets showed an apparent monotonic decline (as volatility increased win rate decreased). To test whether this persists with a principled volatility measure, I re-bucket by Parkinson $\hat{\sigma}$ (Table 5).

Table 5. Kelly parameters by Parkinson volatility condition. Source: author's calculations

Parkinson Band(Annualised)	**Trades n**	**Win Rate p**	$\boldsymbol{f^* = \frac{2.5p - 1}{1.5}}$
Low (3-9%)	59	0.441	0.068
Medium (9-13.5%)	59	0.525	0.209
High (13.5-30%)	59	0.373	-0.045

Grouping all 177 trades (the first trade lacks a full 14-bar Parkinson look back, so it is excluded) into three equal bands by Parkinson volatility rather than ATR, the win rates no longer fall monotonically. Through the optimal risk fraction these give the values denoted f^* above—the highest band falls below 0.400 breakeven, so its fraction is negative. Partly because ATR's true range includes overnight gaps that Parkinson's within-bar range excludes, the two produce conflicting patterns, so the regime effect is not robust to how volatility is measured. That fragility is crucial: the pattern is too estimator-dependent to support a genuine edge, making the 8.1 fractions, again, an upper bound rather than a convincing rule.

9. Reflection and Uncertainty

9.1 Uncertainty in the estimated win rate

The simulation produced an empirical estimate $\hat{p}$ = 0.4494. However, this is only a point estimate from 178 completed trades, not the true future probability of winning. The standard error of a sample proportion is (Grinstead & Snell, 1997):

$$SE(\hat{p}) = \sqrt{\frac{\hat{p}\,(1 - \hat{p})}{n}} \quad (13)$$

$$\text{SE} = \sqrt{\frac{0.449\,(1 - 0.449)}{178}} = 0.0373$$

95% interval :

$$0.4494 \pm 1.96(0.0373)$$

$$[0.376, 0.523]$$

This indicates that the true win probability could range anywhere from 0.376 to 0.523. Converting these endpoints into Kelly fractions yields:

Lower:

$$f^*_{low} = \frac{0.376(2.5) - 1}{1.5} = -0.04$$

Upper:

$$f^*_{high} = \frac{0.523(2.5) - 1}{1.5} = 0.205$$

An approximate 95% confidence interval (standard confidence level in statistics) for the true win probability p is therefore some value between 0.376 to 0.523. This means that if I repeated the same trading experiment many times, about 95% of the intervals constructed this way would contain the true probability p. The lower bound is below both the breakeven probability 0.400 and the profitability threshold for full Kelly, about 0.425. This means the simulation gives evidence of a positive edge in this sample, but not enough to guarantee that the edge is stable. A small error in p can shift the Kelly fraction dramatically.

In other words, using the standard error formula we can find that even though the estimated Kelly fraction from the 178 trades is 8.24%. The 95% confidence interval suggests the true Kelly fraction could plausibly range from about -4% (this would mean a negative fraction and tells you to risk 0%) to 20.5% (very aggressive). This is a very large range. Thus, although the sample estimate gives a positive Kelly fraction of 8.24%, sampling uncertainty means the true optimal fraction could be much lower, possibly even negative, supporting the use of fractional Kelly.

9.1.1 Bayesian interpretation of win-rate uncertainty

A second way to view the same limitation is through Bayesian updating. The empirical value $\hat{p} = 0.4494$ is a point estimate, but the true win probability is uncertain. The true win probability could be a bit lower or higher, as we only observed one sample and as highlighted in section 9.1. Rather than treating p as fixed immediately, I can begin with a neutral prior belief about p and then update this belief after observing the simulated trade outcomes (Gelman et al., 2013).

$$p \sim \text{Beta}(1,1)$$

The prior Beta(1,1) is uniform, meaning that before observing the data I do not favour any particular value of p between 0 and 1. In other words, before looking at the trades, I assume every win probability between 0 and 1 is equally possible. With a Beta prior, the update rule is as follows:

$$Beta(\alpha, \beta) \rightarrow Beta(\alpha + k, \beta + (n - k))$$

Where α and β indicate the prior Beta. After observing 80 wins and 98 losses, the posterior distribution becomes:

$$p \mid \text{"}data\text{"} \sim Beta(1 + 80, 1 + 98) = Beta(81\,, 99)$$

The posterior mean is therefore:

$$E(p \mid \text{data}) = \frac{81}{81 + 99} = 0.4500$$

Frequentist estimate from equation (7) yields $p = 0.4494$. Then to calculate Kelly using the posterior mean:

$$f^*_{Bayes} = \frac{0.45(1.5+1)-1}{1.5}$$

$$f^*_{Bayes} = 0.0833$$

Compared to original:

$$f^*_{Empirical} = 0.0824$$

Difference:

$$0.0833 - 0.0824 = 0.0009$$

This is very close to the observed win rate of 0.4494 because the sample size is large enough that the neutral prior has little influence, but the Bayesian interpretation is more cautious because it treats p as a distribution rather than as a known constant. This supports the same conclusion as the confidence interval: the Kelly fraction is highly sensitive to uncertainty in p and the issue is that 178 trades is not enough to know the true win probability perfectly. Therefore, the Bayesian framing is still valuable, as it shows that p should be treated as uncertain rather than fixed.

9.2 Sensitivity to p and R

From equation (5), the Kelly fraction can be written as:

$$\frac{p(R+1)-1}{R} \qquad (14)$$

This form makes the sensitivity clearer. Differentiating partially with respect to p and R gives:

$$\frac{\partial f^*}{\partial p} = \frac{(R+1)}{R} \qquad (15)$$

$$\frac{\partial f^*}{\partial R} = \frac{(1-p)}{R^2} \qquad (16)$$

For empirical value p:

$$\frac{\partial f^*}{\partial p} = \frac{2.5}{1.5} = 1.6667$$

This means that a 0.01 increase in estimated win probability raises the Kelly fraction by roughly 0.0167. This relationship is plotted in Figure 5. Appendix D tabulates f^* across win rates.

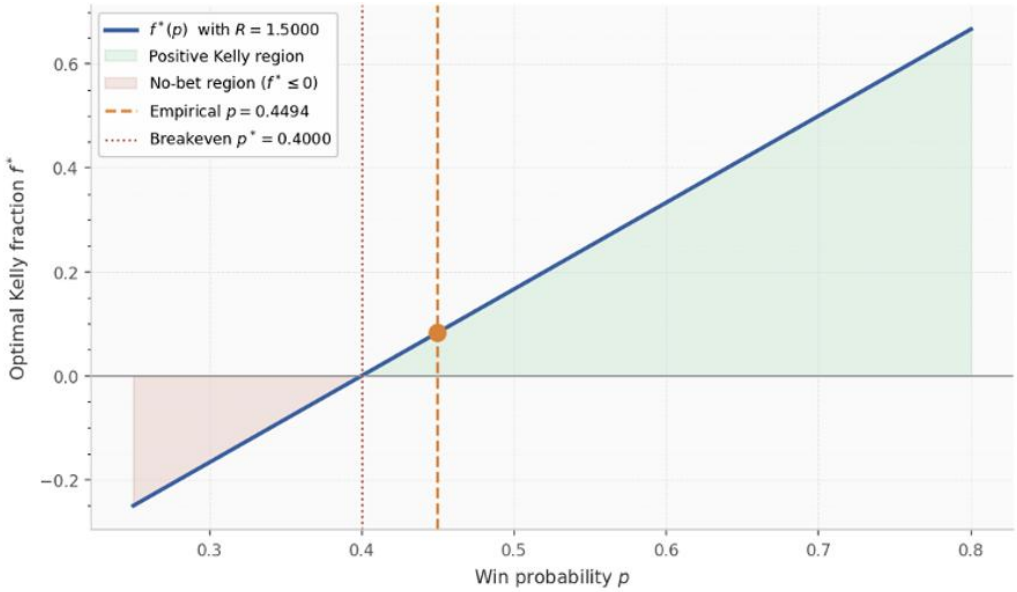

Figure 5. Sensitivity of f to win probability p, with R fixed at 1.5. Source: author's calculations*

For R:

$$\frac{\partial f^*}{\partial R} = \frac{1 - 0.4494}{1.5^2} = 0.2447$$

This means that an 0.1 increase in R raises the fraction by about 0.0245. Figure 6 shows the corresponding relationship for R:

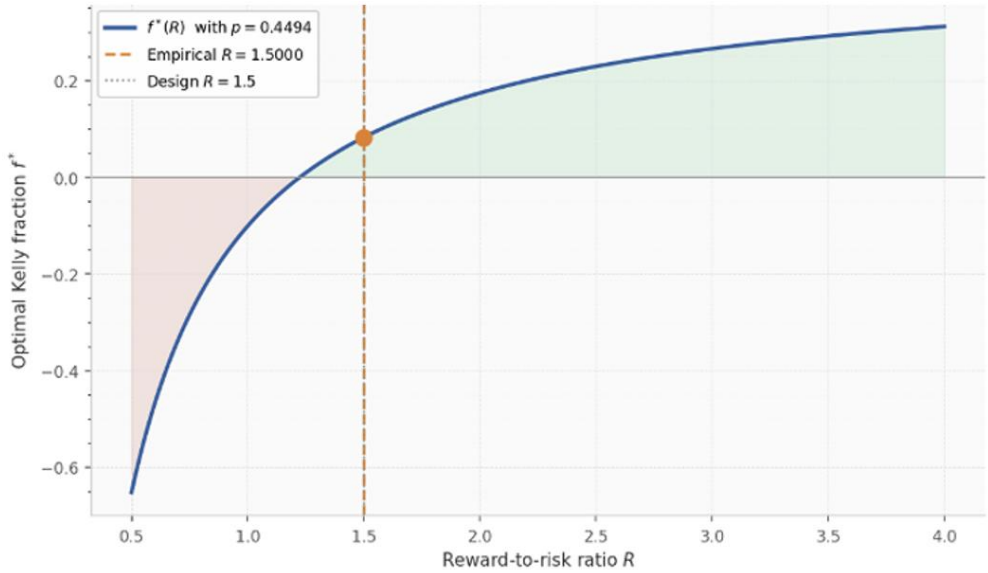

Figure 6. Sensitivity of f to reward-to-risk ratio R, with p fixed at 0.4494. Source: author's calculations*

Both derivatives are positive for $0 < p < 1$ and $R > 0$, so f^* increases when either the win probability or reward-to-risk ratio increases. For fixed R the sensitivity is constant in p; what changes near breakeven is the consequence of an error; a small misestimate of p can flip f^* from positive to negative. This is important because p is estimated from historical data and may not remain stable.

By rewriting f^* as shown below, and then taking the limit (17), an interesting property arises: even extremely large reward-to-risk ratios do not justify risking more than the win probability itself, as even if wins paid off infinitely well, the Kelly fraction caps risk at your probability of winning.

$$f^* = p + \frac{p-1}{R}$$

$$\lim_{R\to\infty} f^* = p \qquad (17)$$

9.3 Limitations of the trade simulation

The simulation is theoretical. It assumes exact entry at the candle close, exact exit at the take-profit or stop-loss level, no spread, no slippage, and no transaction costs (Pardo, 2008; Bailey et al., 2014). A realistic round-trip cost would raise the breakeven win rate above 0.400 and lower f^* below 0.0824; because my point estimate $\hat{p} = 0.4494$ is already close to breakeven, even a small transaction cost erodes a meaningful share of the modelled edge. It also excludes unresolved and ambiguous trades. These choices are mathematically defensible because they keep the outcomes comparable, but they simplify live trading conditions. The random long/short rule also means the model is not a trading strategy in the normal sense. It is a controlled experiment for estimating how a fixed payoff structure behaves on historical SPY candles. The observed p should, thus, not be treated as a permanent edge. This distinction matters because Kelly sizing can become dangerous when p is overestimated (MacLean et al., 2010). Part II tests these limitations directly on a real strategy, with transaction costs, an edge estimated from historical data and then tested on live data; the combined conclusion is given in Section 18.

10. Part I Conclusion

This exploration set out to model, locate, and evaluate. It modelled the expected long-run logarithmic growth of the fixed-rule SPY strategy as G(f); from 178 completed SPY trades, it estimated $\hat{p} = 0.4494$ and applied the fixed reward-to-risk ratio $R = 1.5$, as well as located the maximum by differentiation at $f^* = 0.0824$ via equation (5). The concavity argument from equation (6) confirmed this is a genuine maximum: $G''(f) < 0$ for all $f \in (0,1)$, meaning no other fraction produces higher expected log-growth under these parameters. What remained was to evaluate whether that optimum is usable in practice; on that question the answer is negative for two reasons—one statistical, one practical.

The investigation's more important findings, however, came from testing what happens when the model's assumptions are loosened.

The Bayesian update barely moved the number: $f^*_{Bayes} = 0.0833$ versus the frequentist 0.0824, a difference of 0.0009. With 178 trades, the sample is large enough that the prior has little influence over the posterior. The revealing figure forced into view is the interval: the 95% confidence interval on $\hat{p}$ runs from 0.376 to 0.523, which converts to a Kelly fraction range of roughly -4% to 20.5%. The point estimate looks stable; the interval is enormous. A trader using 178 trades to justify an 8.24% risk fraction is working from a figure that, statistically, could justify risking nothing at all.

The volatility analysis sharpened this rather than resolving it. Sorting trades by ATR made calm markets look advantageous—Low-ATR $f^* = 0.2222$ against High-ATR $f^* = 0.0161$—but re-measuring volatility with the Parkinson estimator (8.2) flattened and even reversed that pattern, the highest-volatility band falling below the 40%-win rate breakeven into a negative (no trade) fraction. The apparent volatility edge is therefore not robust to how volatility is measured, so it cannot be relied on as a decisive sizing rule. This connects directly to the leverage finding: full Kelly already implies 32.96:1

effective exposure at that stop-loss distance. In high-volatility conditions, where a 0.25% stop is narrow relative to typical bar sizes, clean execution at those levels becomes increasingly implausible.

A single tension runs through these findings and is, I think, the real conclusion of the investigation. Kelly's formula assumes p is known, but the confidence interval shows it is uncertain after 178 trades, and the Bayesian posterior confirms that width even at that sample size. The volatility analysis shows that even the apparent structure in p is not robust to how it is measured, let alone stable over time. All of this pushes against full Kelly in practice, without any flaw in the mathematics behind it.

The limit from equation (17), $\lim_{R\to\infty} f^* = p$, captures something true about this: regardless of how attractive the payoff becomes, the fraction you should risk never exceeds your probability of winning. No R is large enough to overcome an uncertain p. More data would not fix this either—because the win rate shifts with conditions, a larger sample only sharpens an estimate of a moving target, which is why even an algorithm with years of history and thousands of trades can be sized incorrectly. My parents watched Lehman's leverage assumptions collapse when the true probability of default turned out to be nothing like the model's estimate. The mathematics here says something structurally similar: uncertainty in p, not the formula itself, is where position sizing goes wrong.

The Kelly Criterion is not a sizing rule to follow. It is a theoretical upper bound. Used as a direct instruction with an estimated p, it recommends 33:1 leverage from a sample where the true fraction could plausibly be negative. Used as a benchmark, it answers a more useful question: the most you could rationally risk if everything the model assumes were exactly right. Any reasonable implementation belongs well below that bound, probably around half Kelly, and should shrink further wherever the edge is weakest and least certain.

Part II | Blackbox: Testing an Estimated Edge in Forward Trading

11. From a known edge to an estimated one

Part I showed that the Kelly fraction is fragile once p must be estimated: the 95% interval for p gave a Kelly fraction anywhere from -4% to 20.5% (Section 9.1). Its random-direction strategy, however, had no real edge to estimate. Part II applies the same question to Blackbox, an automated trading system, I built and ran on OANDA demo accounts from June to August 2026. Its foreign-exchange (FX) strategy looked profitable in backtests, then lost money in forward testing until its 25% drawdown limit stopped it. Consequently, a secondary research question arose:

To what extent does an estimated trading edge survive forward testing, and can pre-specified volatility-conditioned trading or sizing improve growth-risk outcomes without look-ahead bias?

- **H1:** backtest and forward results differ, and an exact-period replay can show how much is due to rule or implementation mismatch.
- **H2:** volatility known at entry predicts a trade's net expectancy.
- **H3:** volatility-conditioned sizing beats fixed sizing on untouched forward data.

All rules and thresholds were fixed on 2014-2021 data, check on 2022 – 7 June 2026, and then applied to the forward test (8 June – 4 August 2026). The design was pre-registered (Appendix G; Nosek et al.,

2018). The forward test used a demo account: prices and spreads are real broker quotes, but no real money was at risk.

12. The Blackbox system and data

12.1 The strategy

The strategy is a daily trend-following rule with a pullback entry (cf. Moskowitz, Ooi & Pedersen, 2012), checked at the close of every 4-hour (H4) candle for 14 currency pairs. A long trade (buy) requires the last completed daily close to be above a rising 20-day EMA, an H4 RSI(14) between 38 and 55, and ATR(14), 14 means it is calculated using 14 candles, above its recent average (Wilder, 1978); shorts are symmetric. The stop is $1.5 \times ATR$ $(20 - 90\ pips)$ and the target is 2.5 times the stop loss, so $R = 2.5$ as in Part I. Open trades move to breakeven at +0.9R, trail 1R behind the best price from +1R, and add a half size *pyramid* position at +1R. Entries are only made between 06:00 and 18:00 UTC. The live system added layers than no backtest modelled, including a news blackout, currency-exposure caps, a Friday *weekend guard* that halves every stop distance, and the 25% drawdown halt. This entire algorithmic trading system was built with the help of Claude Code.

12.2 Data and outcome measure

Backtests use OANDA mid-price candles for fixed dates (22,527 trades, 2014 – 7 June 2026). The 202 forward trades (167 entries and 35 pyramids) were rebuilt from the FX account's complete broker ledger of 2,581 transactions, which records every fill, stop, spread cost and financing charge and reconciles it to the final balance to the cent. Each trade is measured by its net R-multiple, the profit per unit of initial risk.

$$x_i = R_{net,i} = \frac{PL_i + financing_i}{|e_i - s_i| * u_i * c_i}$$

Where PL_i is the profit or loss on trade i, $financing_i$ the overnight financing charge, e_i the entry price, s_i the initial stop price, u_i the number of units traded and c_i the conversion into Swiss francs. Spreads are already inside the fill prices. Backtest trades use the same ratio on mid prices minus a fixed cost of 0.03R (spread costs), the assumption of all earlier Blackbox backtests.

12.3 Integrity and AI assistance

The corrected backtest engine produced identical buy, sell and no-trade signals to the live strategy across all 3528 four-hour candles tested, including 1,048 buy or sell signals. This verifies the signal-generation, although minor differences in trade execution and position management may have arisen. Two weaknesses remain: code after 19 June 2026 was not version-controlled, and the rules changed slightly during the forward test (ATR-scaled stops from 10 June, small changed to the traded pairs in June, automatically tuned stops from 29 June, and halved risk with machine learning sizing from 20 July). The software and analysis scripts were written largely with an AI coding assistant (Claude Code) under my direction; the research decisions and interpretation are mine (Appendix I).

13. Experiment A | Finding and validating an edge

A fixed-date backtest finds no statistically significant positive expectancy (Table 6). The (t)-statistic measures how many standard errors the estimated mean return lies above or below zero, allowing us to assess whether the observed profitability is statistically distinguishable from break-even. Before costs the expectancy is $+0.002R$ per trade ($t = 0.23$); after the assumed $0.03R$ cost it is negative in both periods, and only two of thirteen years were positive and no pair was significantly positive.

Table 6. Corrected backtest, 14 FX pairs, all trades including pyramids. Source: author's calculations

Segment and cost	n (number of trades)	Mean R	95% CI	t
All 2014 – Jun 2026, no cost	22,257	+0.002	[-0.013, 0.017]	0.23
Calibration 2014–21, 0.03R	14,361	-0.026	[-0.045, -0.007]	-2.72
Historical test 2022–26, 0.03R	8,166	-0.032	[-0.057, -0.008]	-2.57
All, 0.076R (measured cost)	22,527	-0.074	[-0.089, -0.059]	-9.77

This contradicts both backtests on which the live system was built, so I re-ran the *original* scripts unchanged on the same data and removed one possible bias at a time to figure out why the original backtests looked profitable.

Table 7. Replication of the original backtests (logged: 655 trades at +0.337R; 12,328 at +0.053R). Source: author's calculations.

Test	n	Mean R	t
Design-period backtest, original code	654	+0.337	5.70
… using only daily candles closed at the decision time	681	+0.034	0.64
… and entry evaluated at the H4 close	721	+0.029	0.55
12.5-year backtest, original code (FX pairs)	12,328	+0.053	4.32
… exit path starts after the entry candle	11,270	+0.111	8.29
… and only closed daily candles	11,558	+0.011	0.87

Both original results are reproduced almost exactly, and both vanish when one bias is removed. OANDA labels each daily candle with its *opening* time, 21:00 or 22:00 UTC on the previous calendar day (OANDA, 2026). Both scripts selected daily candles by calendar date, so at 13:00 UTC on 10 June the trend filter could already see the daily close of 11 June (Figure 7). Without this look-ahead the design-period result falls 10x to +0.034R (t = 0.64) and the 12.5-year result to +0.011R (t = 0.87), neither significant (Figure 8). The 12.5-year script also checked exits against the entry candle's own range but that error lowered its result. So, what happened was, my original scripts inadvertently used future market information.

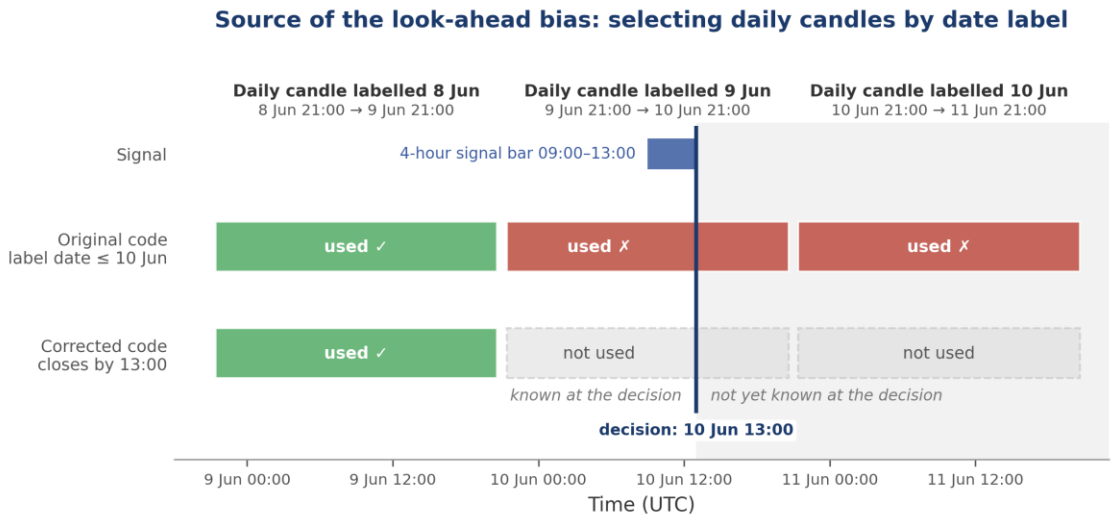

Figure 7. Visualiser: date-based selection of OANDA daily candles leaks a future close into a 4-hour decision. Source: author's calculations

The key distinction is between a candle's timestamp and the time its closing price becomes available. AT 13:00 UTC on 10 June, only the candle labelled 8 June had closed. By selecting candles according to their date labels rather than their completion times, the original backtest incorporated future price information into its trading decisions, artificially inflating its profitability.

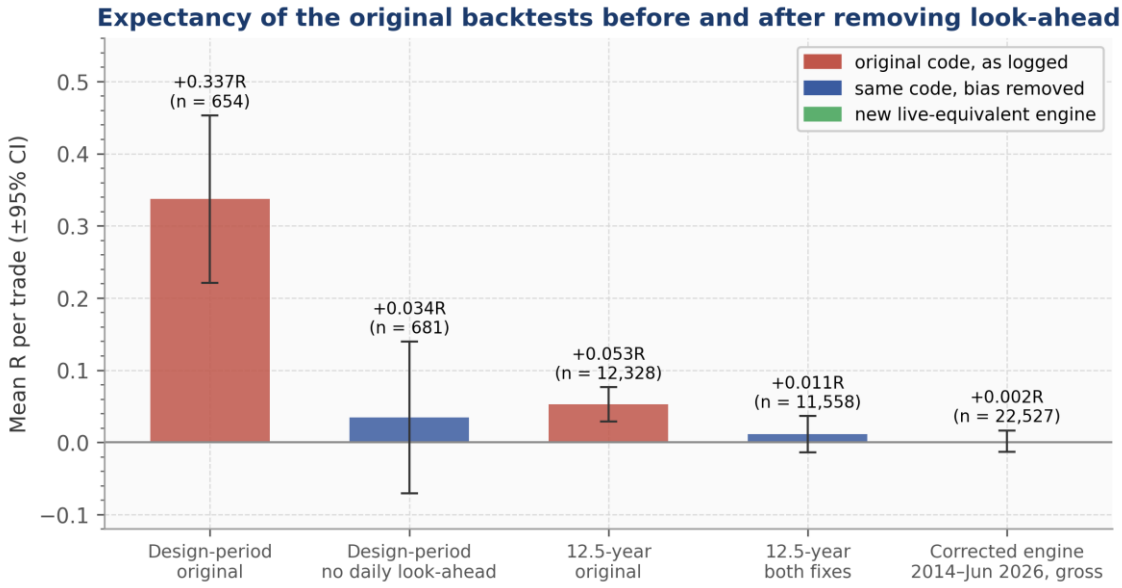

Figure 8. Expectancy (±95% CI) of the original backtests before and after removing the look-ahead. Source: author's calculations

A simulation that uses information unavailable at the time of the decision is a standard backtest failure (Bailey et al., 2014). **The premise of H1, that a real edge decayed in forward testing, is therefore not supported: once the look-ahead is removed, no backtest even finds a statistically significant positive expectancy under the tested data, costs, and time periods.** This does not prove that the strategy has no edge at all; it shows that the evidence used to deploy it did not support one.

14. Experiment B Reconciling the backtest with the forward test

The replay runs the corrected engine on exactly the forward-test dates and pairs, then moves one step at a time from backtest to reality (Table 8, Figure 9). Step B2 takes the actual live entries from the ledger and simulates their exits with the backtest rules on 5-minute candles. Therefore, B0-B3 are simulations, while B4 is what actually happened in the OANDA account, effectively answering: **If we simulate the same trading period but gradually make the simulation more realistic, can we reproduce Blackbox's actual losses?**

Table 8. From replay to reality, 8 June – 4 August 2026. Source: author's calculations.

Step	n	Mean R	$\pm 95\%$ CI
B0 replay with backtest assumptions (0.03R)	269	-0.080	0.136
B2 live entries, backtest exits, no cost	202	-0.199	0.144
B2w + live weekend guard	202	-0.157	0.136
B3 + actual spread and financing	202	-0.236	0.137
B4 actual observed result (R_{net})	202	-0.229	0.130

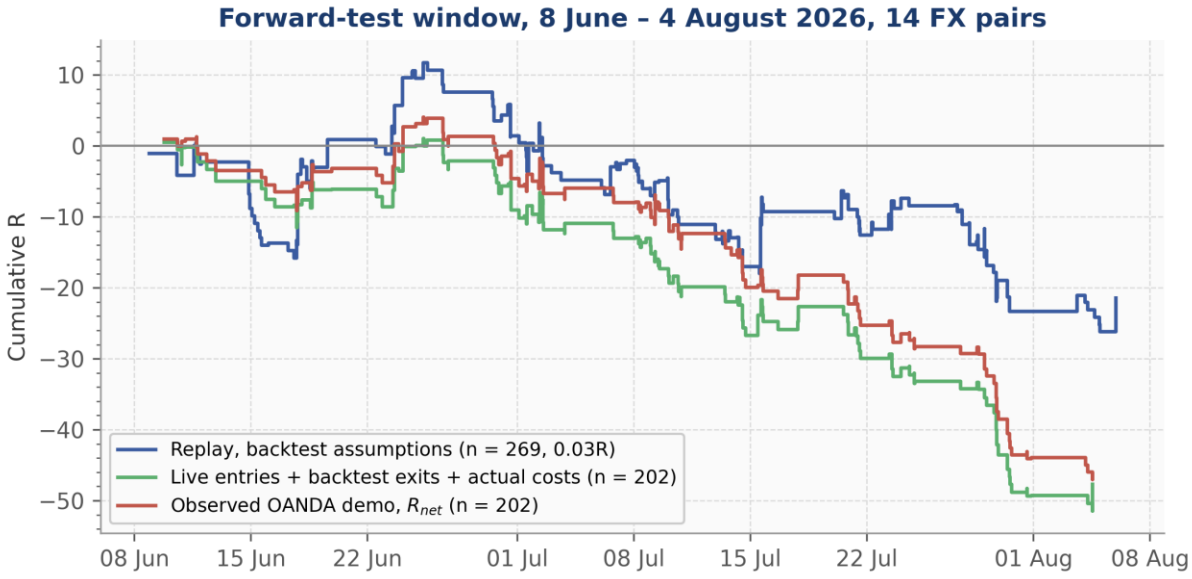

Figure 9. Cumulative R on identical dates: replay, live entries with backtest exits and actual costs, and the observed result. Source: author's calculations

Three findings follow. First, the strategy lost money even under backtest assumptions (B0), which Blackbox followed closely. Second, the trade management worked as designed: with actual costs, the backtest exit rules reproduce the observed result (-0.236R against -0.229R), and simulated and actual exits agree for 177 of 178 stop-outs and 15 of 16 take-profits; the untested weekend guard even helped by about 0.04R. Third, costs were 2.5 times the assumption: spreads averaged 0.076R per trade (from 0.046R on EUR/USD to 0.224R on EUR/GBP), about 60% of the account's loss.

Every one of the 167 live entries matches a same-direction replay signal (the backtest would have bought as well) but 65 of them (39%) came from the candle closing at 05:00 UTC. The live bot checks its rules every five minutes, so it entered these at 06:00 when the window opened; the backtest never takes them. They did worse (-0.359R against -0.113R), but the difference is not significant ($t \approx -1.6$) and was found after looking at the data, so it is exploratory. Of the 136 replay trades the bot did not take, all but 16 are explained by pairs it was not trading or positions it already held.

H1 is supported in a narrower form: implementation differences and underestimated costs explain part of the gap, but most of the difference between the logged +0.34R and the observed -0.23R is the look-ahead bias.

15. Experiment C | Volatility-adaptive rules

Part I found an apparent link between volatility and win rate that disappeared under the Parkinson estimator (Section 8.2); volatility-managed sizing is also studied in finance (Moreira & Muir, 2017). Here each trade's volatility is the ATR(14) percentile against the pair's previous 1,560 H4 candles (about one year), with the Parkinson estimator as a robustness check. This is not an independent test of whether volatility predicts trading profits in general: Blackbox already uses ATR to filter entries (no trade below half the average ATR), to set stops ($1.5 \times ATR\ (20 - 90\ pips)$) and therefore to define the R unit itself, so outcomes are already volatility-scaled. The regimes only ask whether the *remaining* variation in volatility predicts Blackbox's expectancy; Parkinson uses the same prices, so it checks the measurement, not the strategy. Calibration trades fixed the tercile cut-points (0.299, 0.676), which are not the same as Part I, and two rules: (ii) skip the regime with the lowest calibration expectancy, and (iii) scale size with each regime's calibration expectancy, bounded between 0.5x and 1.5x.

Table 9. Mean R by entry-volatility regime (backtest at 0.03R cost; forward R_{net}). Source: author's calculations.

ATR Regime	Calibration n | mean R	Historical test n | mean R	Forward n | mean R [95% CI]
Low	4,793 | -0.002	2,538 | +0.000	108 | -0.288 [-0.47, -0.11]
Mid	4,783 | -0.038	2,427 | -0.055	75 | -0.203 [-0.40, -0.01]
High	4775 | -0.036	3,201 | -0.041	19 | +0.003 [-0.52, 0.52]

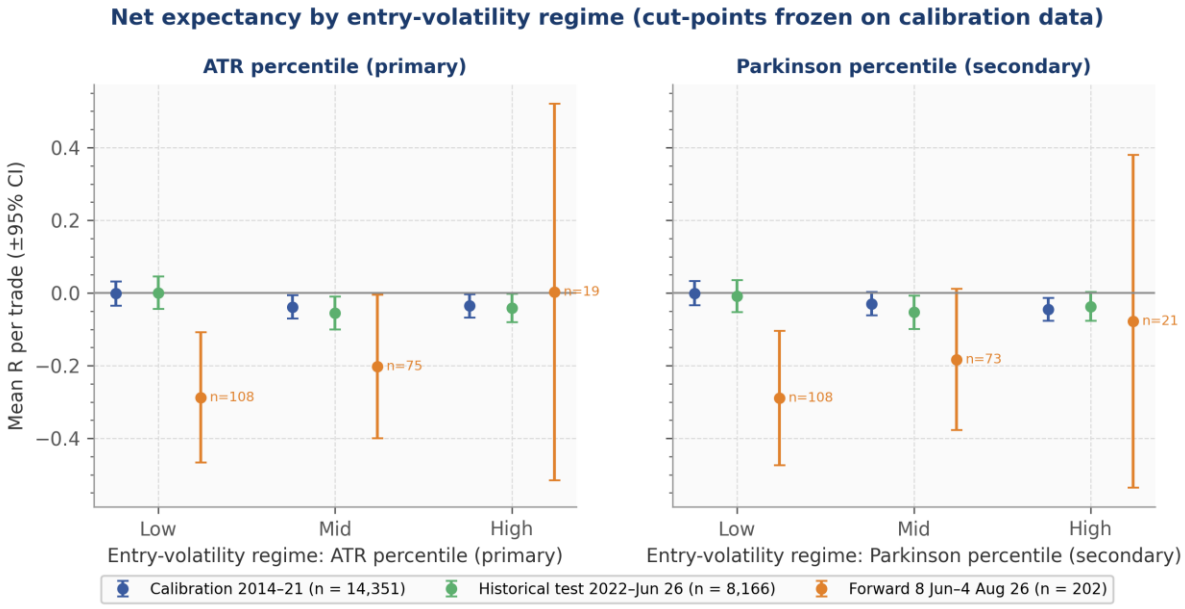

Figure 10. Mean R (±95% CI) by volatility regime for the ATR (primary) and Parkinson (secondary) measures. Source: author's calculations

No regime was profitable after costs, and the ordering reversed in the forward test, where low-volatility entries lost most. The filter moved the historical-test expectancy only from -0.032R to -0.023R and made the forward result worse (-0.229R to -0.244R). The sizing rule degenerated: every calibration regime had a non-positive mean, so every weight fell to the 0.5x floor and the rule simply halved all positions, a change of scale rather than an improvement. The Parkinson measure does not change the picture. **H2 is not supported for this ATR-based strategy,** matching Part I, and **H3 is not supported for volatility rules:** no regime showed a positive expectancy for a rule to select.

16. Experiment D | Kelly optimisation for realised R-multiples

16.1 Extending the growth function

Equation (3) assumes every trade ends at exactly $+R$ or -1. Blackbox trades do not: stops are moved, trailed and tightened, and every trade pays a spread. If $x_1, \ldots, x_n$ are realised R-multiples, with trade i, risking f multiplies wealth by $(1 + fx_i)$, and by the same argument as equations (1) – (3) the average log-growth per trade, for n number of trades, is

$$G(f) = \frac{1}{n}\sum_{i=1}^{n} \ln(1 + fx_i), 0 \leq f < \frac{1}{|min_i x_i|}$$

Differentiating twice (with respect to f, and treating each x_i constant) as in equations (4) and (6):

$$G'(f) = \frac{1}{n}\sum_{i=1}^{n}\frac{x_i}{1+fx_i}$$

$$G''(f) = -\frac{1}{n}\sum_{i=1}^{n}\frac{{x_i}^2}{(1+fx_i)^2} < 0$$

So G is strictly concave down and has at most one maximiser f^* (**Proposition 1**). At $f = 0$:

$$G'(0) = \frac{1}{n}\sum_{i=1}^{n} x_i = \bar{x}$$

Proposition 2. The slope (which tells us whether increasing risk increases or decreases growth) at zero, not trading at all, is the expectancy $\bar{x}$, which also represents the sample mean of all the x_i R-multiples. Since G' is decreasing, if $\bar{x} < 0$ then $G'(f) < 0$, which means that increasing f decreases growth, for all $f > 0$, so $f^* = 0$: the growth-optimal decision is not trade. A cost c per trade replaces $\bar{x}$ by $\bar{x} - c$, so any strategy whose gross edge, or expectancy, is below its cost has $f^* = 0$.

Since $G(0) = 0$, the Fundamental Theorem of Calculus allows the growth function be recovered by integrating its derivative:

$$G(f) = \int_0^f G'(u)\,du = \int_0^f \frac{1}{n}\sum_{i=1}^{n}\frac{x_i}{1+ux_i}\,du$$

This expresses average log-growth as the accumulated marginal effect of increasing the risk fraction from zero to f. When estimated expectancy in non-positive, the integrand is negative for all $(u > 0)$ within the valid domain, confirming that any positive f reduces expected log-growth under the model.

Proposition 3. By the Taylor approximation, for a small edge (when u is close to 0), $\ln(1 + u) \approx u - \frac{1}{2}u^2$, where $u = fx_i$, gives

$$\ln(1 + fx_i) \approx fx_i - \frac{1}{2}f^2{x_i}^2$$

$$G(f) \approx \frac{1}{n}\sum_{i=1}^{n}(fx_i - \frac{1}{2}f^2{x_i}^2)$$

$$G(f) \approx f\left(\frac{1}{n}\sum_{i=1}^{n}x_i\right) - \frac{1}{2}f^2\left(\frac{1}{n}\sum_{i=1}^{n}{x_i}^2\right)$$

So, by $G'(0)$:

$$G(f) \approx f\bar{x} - \frac{1}{2}f^2\bar{x}^2$$

Differentiate:

$$G'(f) \approx \bar{x} - f\overline{x^2}$$

Set equal to zero, and:

$$f^* \approx \frac{\bar{x}}{\overline{x^2}} = \frac{\mu}{(\sigma^2 + \mu^2)}$$

μ is the true mean R-multiple and σ^2 is its variance, so f^* **is roughly proportional to the estimated edge**: an error that multiplies $\bar{x}$ by 10 multiples the recommended risk by about ten. If every outcome is $+R$ or -1, the average log growth per trade equation reduces exactly to equation (3), so Part I itself is a special case. The uncertainty in f^* follows from $SE(\bar{x}) = \frac{s}{\sqrt{n}}$, the counterpart of equation (13); see also Baker and McHale (2013).

16.2 Kelly estimated only from past data

The log-growth equation was maximised on a grid from 0% to 5% at the start of each month, using only trades already closed, and compared with fixed 1% and 2% risk on the same trade sequence, with the 25% halt (Table 10).

Table 10. Sizing rules on identical trade sequences. Source: author's calculations.

Sizing Rule	Historical: wealth | max DD	Forward: wealth | max DD
Fixed 1%	0.87 | 25%	0.79 | 25% (halted)
Fixed 2%	1.00 | 25%	0.82 | 26% (halted)
Kelly, past data	1.00 | 0%	1.00 | 0%
Half Kelly	1.00 | 0%	1.00 | 0%

From 2022 onwards every estimate gave $f^* = 0$ because the corrected past expectancy was never positive (Proposition 2). Kelly therefore never traded and kept wealth at 1.00, while fixed 1% and 2% sizing both hit the 25% halt in the forward test. A volatility-adjusted half Kelly, estimated separately in each regime, also stayed out in the forward test but ended the historical test at 0.87.

16.3 How a biased estimate is amplified

Table 11. Kelly fractions from biased and corrected estimates, applied to the 202 forward trades. Source: author's calculations.

Kelly estimated from	Mean R	Unconstrained f^*	f applied (5% cap)	Forward: halt after trade	Wealth without halt
Original backtest (look-ahead)	+0.337	19.0%	5.0%	20	0.08
Same backtest, no look-ahead	+0.034	1.8%	1.8%	112	0.42
Corrected calibration 2014-21	-0.026	0%	0%	never	1.00

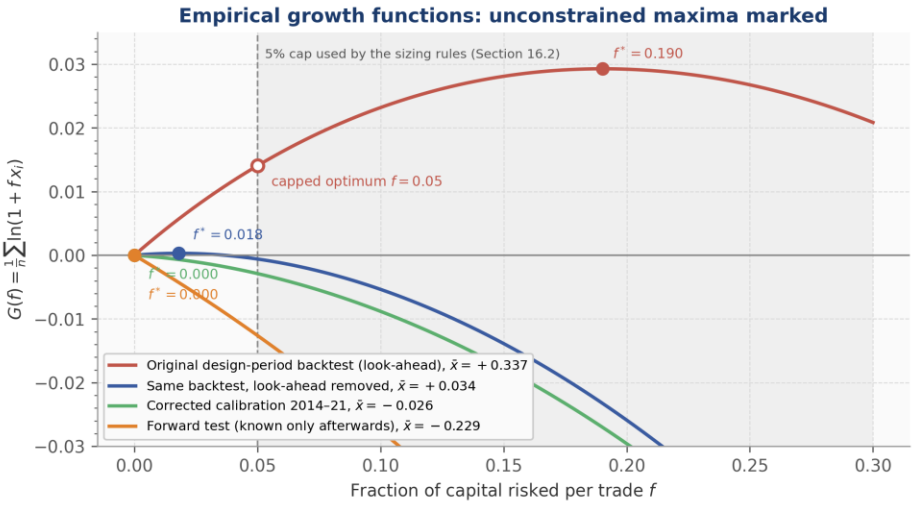

Figure 11. Empirical growth function (19) for each set of realised R-multiples. Filled markers: unconstrained maxima; open marker: optimum under the 5% cap. Source: author's calculations

Each maximum was found by solving $G'(f) = 0$ directly on the realised R-multiples and agrees with a 0.05% grid search. The curves are in-sample estimates from historical outcomes, not known future distributions; the forward curve could only be drawn afterwards. The biased backtest implies an *unconstrained* optimum of **19.0%** of capital per trade. Because G is concave down, the optimum under the 5% cap of Section 16.2 is the cap itself, and even that would have hit the halt after 20 forward trades (10 at the uncapped fraction). Removing the look-ahead gives 1.8%, and the full corrected data give 0% (Table 11, Figure 11). A bias that multiplied the edge by about ten multiplied the recommended risk by about ten, which is Proposition 3 in practice: Kelly sizing does not protect against a wrong estimate, it amplifies it.

17. Discussion and Limitations

- Simulation assumptions (strong evidence, main cause). One look-ahead produced both apparent backtest edges, and costs were underestimated 2.5-fold.
- Implementation (moderate). 65 untested session-open entries and untested auto-tuned stops, but the backtest exits with real costs reproduce the live result.
- Regime change (weak). The forward period was worse than average (−0.080R against about −0.03R in the replay), but within sampling noise, and the corrected history shows no significant edge that could have been lost.
- Risk amplification (strong, for the size of the damage). Kelly on the biased estimate would have risked 19% per trade; the bot's 0.7% risk and 25% halt limited the loss.

Testing many strategy variants on the same data inflates the best result even without look-ahead (Bailey & López de Prado, 2014), a further reason not to trust the original figures. The main limitations are the short forward test (eight weeks, 202 trades); correlated trades (35 pyramids, many simultaneous entries), handled with a day-clustered $t\ of\ -2.92$ and a check of only original trades ($-0.209R$, $t\ =\ -2.83$); demo rather than real-money fills; a single strategy system; hourly candles in backtest exits (stop assumed hit first); Kelly paths that ignore concurrent positions; and unversioned code after 19 June, which leaves 16 replay trades unexplained. Analyses added after pre-registration are labelled in Appendix G.

18. Combined Conclusion

Part I concluded that the Kelly fraction should be treated as an upper bound because uncertainty in p, not the formula, is where position sizing goes wrong. Part II tested this on a real automated strategy and found a sharper version of it.

H1 is supported in a narrower form: the replay identifies real implementation differences and costs 2.5 times higher than first assumed, but the main reason the backtests and the forward test disagree is an endogenous look-ahead bias caused by broker candle timestamps (confidence: high). **H2** is not supported for Blackbox; because its stops and filters already use ATR, this says nothing about volatility-conditioned trading in general (confidence: moderate, limited by the forward sample). **H3** is not supported: the best sizing decision was zero exposure, chosen by Kelly from past, corrected data (confidence: high for this strategy).

Extending Kelly to realised outcomes (equations in Part II) explains why: the optimal fraction is zero when the estimated expectancy is zero or negative, and otherwise roughly proportional to it. **Before asking how much to risk, check that the edge survives a bias-free test.** In part I a random sample made the edge uncertain; in Part II a timestamp made an edge appear where the corrected data wshow none, and the formula turned it into risk.

19. Bibliography

Anthropic. (2026). *Claude* (AI assistant) [Computer software]. https://www.anthropic.com/claude

Bailey, D. H., & López de Prado, M. (2014). The deflated Sharpe ratio: Correcting for selection bias, backtest overfitting, and non-normality. The Journal of Portfolio Management, 40(5), 94–107. https://doi.org/10.3905/jpm.2014.40.5.094

Bailey, D. H., Borwein, J. M., López de Prado, M., & Zhu, Q. J. (2014). Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-sample performance. *Notices of the American Mathematical Society, 61*(5), 458–471.

Baker, R. D., & McHale, I. G. (2013). Optimal betting under parameter uncertainty: Improving the Kelly criterion. Decision Analysis, 10(3), 189–199. https://doi.org/10.1287/deca.2013.0271

Breiman, L. (1961). Optimal gambling systems for favorable games. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability* (Vol. 1, pp. 65–78). University of California Press.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian data analysis* (3rd ed.). Chapman & Hall/CRC.

Grinstead, C. M., & Snell, J. L. (1997). *Introduction to probability* (2nd rev. ed.). American Mathematical Society.

Hull, J. C. (2018). *Options, futures, and other derivatives* (10th ed.). Pearson.

Kelly, J. L. (1956). A new interpretation of information rate. *Bell System Technical Journal, 35*(4), 917–926.

Latané, H. A. (1959). Criteria for choice among risky ventures. *Journal of Political Economy, 67*(2), 144–155.

MacLean, L. C., Thorp, E. O., & Ziemba, W. T. (2010). Good and bad properties of the Kelly criterion. *Risk and Decision Analysis, 2*(2), 1–10.

MacLean, L. C., Thorp, E. O., & Ziemba, W. T. (Eds.). (2011). *The Kelly capital growth investment criterion: Theory and practice*. World Scientific.

Moreira, A., & Muir, T. (2017). Volatility-managed portfolios. The Journal of Finance, 72(4), 1611–1644. https://doi.org/10.1111/jofi.12513

Moskowitz, T. J., Ooi, Y. H., & Pedersen, L. H. (2012). Time series momentum. Journal of Financial Economics, 104(2), 228–250. https://doi.org/10.1016/j.jfineco.2011.11.003

Nosek, B. A., Ebersole, C. R., DeHaven, A. C., & Mellor, D. T. (2018). The preregistration revolution. Proceedings of the National Academy of Sciences, 115(11), 2600–2606. https://doi.org/10.1073/pnas.1708274114

OANDA Corporation. (2026). v20 REST API: Instrument endpoints — candles (dailyAlignment, alignmentTimezone) [Documentation]. Retrieved October 31, 2026, from https://developer.oanda.com/restlive-v20/instrument-ep

Pardo, R. (2008). *The evaluation and optimization of trading strategies* (2nd ed.). John Wiley & Sons.

Parkinson, M. (1980). The extreme value method for estimating the variance of the rate of return. *The Journal of Business, 53*(1), 61–65.

Rotando, L. M., & Thorp, E. O. (1992). The Kelly criterion and the stock market. *The American Mathematical Monthly, 99*(10), 922–931.

State Street Global Advisors. (2026). *SPDR S&P 500 ETF Trust (SPY): Fund details*. https://www.ssga.com/us/en/individual/etfs/spy

Stewart, J. (2015). *Calculus: Early transcendentals* (8th ed.). Cengage Learning.

Thorp, E. O. (2006). The Kelly criterion in blackjack, sports betting, and the stock market. In S. A. Zenios & W. T. Ziemba (Eds.), *Handbook of asset and liability management* (Vol. 1, pp. 385–428). Elsevier.

Wiggins, R. Z., Piontek, T., & Metrick, A. (2014). *The Lehman Brothers bankruptcy A: Overview* (Yale Program on Financial Stability Case Study No. 2014-3A-V1). Yale School of Management.

Wilder, J. W. (1978). *New concepts in technical trading systems*. Trend Research.

Yahoo Finance. (2026). *SPDR S&P 500 ETF Trust (SPY): Historical data* [Dataset]. Retrieved May 2026 from https://finance.yahoo.com/quote/SPY/history

20. Appendix

(A)

Simulation Results

Metric	Value	Note
Trades simulated	183	200 buckets drawn; 17 blocked by active unresolved trades
Completed trades (W + L)	178	Only these enter p and R
Wins	80	Take-profit reached first
Losses	98	Stop-loss reached first
Unresolved	5	Neither barrier hit within 78-candle window — excluded
Ambiguous	0	Both barriers touched in same OHLC candle — excluded
Win rate p	0.449438	p = 80 / 178 (completed trades only)
Reward-to-risk ratio R	1.500000	R = TP% / SL% = 0.375 / 0.25 = 1.5 exactly by construction
Breakeven win rate p*	0.400000	p* = 1 / (1 + R) = 1 / 2.5 = 0.4000
p > p*?	0.4494 > 0.4000 ✓ Yes	Positive edge confirmed — Kelly recommends betting

(B)

Kelly Analysis

Quantity	Value	Formula / Interpretation
Optimal Kelly fraction f*	0.082397	f* = [p(R+1) − 1] / R = [0.4494 × 2.5 − 1] / 1.5
G(f*) — optimal log-growth	0.005032	G(f) = p·ln(1+fR) + (1−p)·ln(1−f) evaluated at f*
Actual Kelly fraction f_actual	0.002500	f_actual = SL_PCT = 0.0025 (fraction of account risked per trade, no leverage)
G(f_actual)	0.000304	G at f_actual = 0.0025 → 6.0% of optimal G(f*)
Half Kelly f*/2	0.041199	Conservative theoretical sizing
Double Kelly 2f*	0.164794	Aggressive theoretical sizing (above f* → increasing variance)

(C)

No-Leverage Account Path

Quantity	Value	Note
Starting account	$100'000.00	Initial capital (USD, own funds — no borrowed capital)
f_actual = SL_PCT	0.0025	Each losing trade loses 0.25 % of current account
Win return per trade	+0.375 % of account	W(n+1) = W(n) × 1.00375
Loss return per trade	−0.25 % of account	W(n+1) = W(n) × 0.9975
Final account (after 178 resolved trades)	$105,562.41	Net change: +5.56 % over 178 completed trades

(D)

Sensitivity: f*(p) with R = 1.5 fixed

Win rate p	f* (Kelly fraction)	G(f*)	Interpretation
0.30	-0.166667	0.00000000	No bet — below breakeven
0.31	-0.150000	0.00000000	No bet — below breakeven
0.32	-0.133333	0.00000000	No bet — below breakeven
0.33	-0.116667	0.00000000	No bet — below breakeven
0.34	-0.100000	0.00000000	No bet — below breakeven
0.35	-0.083333	0.00000000	No bet — below breakeven
0.36	-0.066667	0.00000000	No bet — below breakeven
0.37	-0.050000	0.00000000	No bet — below breakeven
0.38	-0.033333	0.00000000	No bet — below breakeven
0.39	-0.016667	0.00000000	No bet — below breakeven
0.40	0.000000	0.00000000	No bet — below breakeven
0.41	0.016667	0.00020777	Marginal positive edge
0.42	0.033333	0.00082897	Marginal positive edge
0.43	0.050000	0.00186071	Marginal positive edge
0.44	0.066667	0.00330047	Conservative
0.45	0.083333	0.00514611	Conservative
0.46	0.100000	0.00739582	Conservative
0.47	0.116667	0.01004813	Conservative
0.48	0.133333	0.01310191	Moderate
0.49	0.150000	0.01655636	Moderate
0.50	0.166667	0.02041100	Moderate
0.51	0.183333	0.02466566	Moderate
0.52	0.200000	0.02932051	Moderate
0.53	0.216667	0.03437603	Moderate
0.54	0.233333	0.03983302	Aggressive
0.55	0.250000	0.04569262	Aggressive
0.56	0.266667	0.05195628	Aggressive
0.57	0.283333	0.05862582	Aggressive
0.58	0.300000	0.06570339	Aggressive
0.59	0.316667	0.07319149	Aggressive
0.60	0.333333	0.08109302	Aggressive
0.61	0.350000	0.08941125	Aggressive
0.62	0.366667	0.09814986	Aggressive
0.63	0.383333	0.10731296	Aggressive
0.64	0.400000	0.11690510	Aggressive
0.65	0.416667	0.12693131	Aggressive

(E) Full Trade Log (183 trades)

n = 183 trades | Wins = 80 Losses = 98 Unresolved = 5 Ambiguous = 0 | Completed win-rate p = 0.4494 | Seed = 42

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
1	2026-02-19	09:40	SHORT	0.908581	684.6100	686.3215	682.0427	682.0427	Win	0.3750	$100'000.00	$100'375.00	0.000000	N/A
2	2026-02-19	13:20	SHORT	0.699707	682.3750	684.0809	679.8161	684.0809	Loss	-0.2500	$100'375.00	$100'124.06	0.940700	High
3	2026-02-20	09:30	LONG	0.265870	683.3200	681.6117	685.8824	685.8824	Win	0.3750	$100'124.06	$100'499.53	0.796882	Med
4	2026-02-20	10:05	SHORT	0.969176	687.0900	688.8077	684.5134	684.5134	Win	0.3750	$100'499.53	$100'876.40	1.322045	High
5	2026-02-20	11:25	SHORT	0.778751	686.1496	687.8650	683.5765	687.8650	Loss	-0.2500	$100'876.40	$100'624.21	1.327916	High
6	2026-02-20	14:10	SHORT	0.716890	688.5200	690.2413	685.9380	685.9380	Win	0.3750	$100'624.21	$101'001.55	1.132866	High
7	2026-02-23	10:05	LONG	0.449362	684.9300	683.2177	687.4985	683.2177	Loss	-0.2500	$101'001.55	$100'749.05	1.023989	High
8	2026-02-23	11:10	LONG	0.272242	680.7000	678.9983	683.2526	683.2526	Win	0.3750	$100'749.05	$101'126.86	0.955204	High
9	2026-02-23	13:20	LONG	0.096391	683.7400	682.0307	686.3040	682.0307	Loss	-0.2500	$101'126.86	$100'874.04	0.689060	Med
10	2026-02-23	14:00	SHORT	0.902602	681.8100	683.5145	679.2532	683.5145	Loss	-0.2500	$100'874.04	$100'621.85	0.646308	Med
11	2026-02-24	10:05	LONG	0.455776	683.9000	682.1902	686.4646	686.4646	Win	0.3750	$100'621.85	$100'999.19	1.086540	High
12	2026-02-24	12:55	LONG	0.202363	687.3700	685.6516	689.9476	689.9476	Win	0.3750	$100'999.19	$101'377.93	0.537237	Low
13	2026-02-25	09:35	LONG	0.305957	691.1500	689.4221	693.7418		Unresolved		$101'377.93	$101'377.93	0.710867	Med
14	2026-02-26	09:40	SHORT	0.579220	691.4500	693.1786	688.8571	688.8571	Win	0.3750	$101'377.93	$101'758.10	0.464126	Low
15	2026-02-26	10:15	LONG	0.176773	686.4281	684.7120	689.0022	684.7120	Loss	-0.2500	$101'758.10	$101'503.70	0.839182	Med
16	2026-02-26	10:35	SHORT	0.856614	686.5800	688.2965	684.0053	688.2965	Loss	-0.2500	$101'503.70	$101'249.95	1.082135	High
17	2026-02-26	11:15	SHORT	0.758520	688.2800	690.0007	685.6989	685.6989	Win	0.3750	$101'249.95	$101'629.63	0.974323	High
18	2026-02-26	12:45	SHORT	0.719463	685.2300	686.9431	682.6604	686.9431	Loss	-0.2500	$101'629.63	$101'375.56	0.766024	Med
19	2026-02-26	13:35	LONG	0.432093	687.4700	685.7513	690.0480	685.7513	Loss	-0.2500	$101'375.56	$101'122.12	0.786371	Med
20	2026-02-27	09:35	SHORT	0.627309	684.0650	685.7752	681.4998	685.7752	Loss	-0.2500	$101'122.12	$100'869.31	1.028756	High
21	2026-02-27	10:50	SHORT	0.584098	684.5400	686.2513	681.9730	686.2513	Loss	-0.2500	$100'869.31	$100'617.14	1.093090	High
22	2026-02-27	11:35	SHORT	0.649847	685.5100	687.2238	682.9393	682.9393	Win	0.3750	$100'617.14	$100'994.46	0.981419	High
23	2026-03-02	09:35	LONG	0.084444	681.0088	679.3063	683.5626	683.5626	Win	0.3750	$100'994.46	$101'373.18	1.328473	High
24	2026-03-02	10:05	LONG	0.415807	684.2100	682.4995	686.7758	682.4995	Loss	-0.2500	$101'373.18	$101'119.75	1.367736	High
25	2026-03-02	11:20	LONG	0.041614	684.6500	682.9384	687.2174	682.9384	Loss	-0.2500	$101'119.75	$100'866.95	1.122169	High
26	2026-03-02	13:00	LONG	0.493991	687.2200	685.5020	689.7971	685.5020	Loss	-0.2500	$100'866.95	$100'614.78	0.779525	Med
27	2026-03-02	14:25	LONG	0.329861	688.2000	686.4795	690.7808	686.4795	Loss	-0.2500	$100'614.78	$100'363.25	0.669876	Med
28	2026-03-03	10:35	LONG	0.144524	671.7000	670.0208	674.2189	674.2189	Win	0.3750	$100'363.25	$100'739.61	1.722254	High
29	2026-03-03	12:00	LONG	0.103403	677.4400	675.7464	679.9804	679.9804	Win	0.3750	$100'739.61	$101'117.38	1.390372	High
30	2026-03-03	13:40	SHORT	0.587645	681.0850	682.7877	678.5309	682.7877	Loss	-0.2500	$101'117.38	$100'864.59	1.172057	High
31	2026-03-04	09:35	LONG	0.170593	680.9900	679.2875	683.5437	683.5437	Win	0.3750	$100'864.59	$101'242.83	1.235327	High
32	2026-03-04	10:35	SHORT	0.925120	684.6200	686.3315	682.0527	686.3315	Loss	-0.2500	$101'242.83	$100'989.73	1.280167	High

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
33	2026-03-04	12:15	SHORT	0.581061	685.9400	687.6549	683.3677	683.3677	Win	0.3750	$100'989.73	$101'368.44	0.857051	Med
34	2026-03-05	09:35	LONG	0.346870	683.6800	681.9708	686.2438	681.9708	Loss	-0.2500	$101'368.44	$101'115.02	0.774495	Med
35	2026-03-05	11:10	SHORT	0.590915	679.8100	681.5095	677.2607	681.5095	Loss	-0.2500	$101'115.02	$100'862.23	1.166697	High
36	2026-03-05	12:05	LONG	0.022804	681.0100	679.3075	683.5638	679.3075	Loss	-0.2500	$100'862.23	$100'610.07	1.128271	High
37	2026-03-05	14:55	SHORT	0.958559	677.0900	678.7827	674.5509	678.7827	Loss	-0.2500	$100'610.07	$100'358.55	0.975452	High
38	2026-03-06	10:15	LONG	0.482303	673.4100	671.7265	675.9353	675.9353	Win	0.3750	$100'358.55	$100'734.89	1.704734	High
39	2026-03-06	11:05	SHORT	0.782735	674.8900	676.5772	672.3592	672.3592	Win	0.3750	$100'734.89	$101'112.65	1.493162	High
40	2026-03-06	13:45	LONG	0.082730	674.7400	673.0532	677.2703	673.0532	Loss	-0.2500	$101'112.65	$100'859.87	1.100947	High
41	2026-03-06	15:25	LONG	0.486658	672.2300	670.5494	674.7509	670.5494	Loss	-0.2500	$100'859.87	$100'607.72	0.983368	High
42	2026-03-09	11:15	LONG	0.490707	668.8700	667.1978	671.3783	671.3783	Win	0.3750	$100'607.72	$100'985.00	1.393942	High
43	2026-03-09	13:00	SHORT	0.937826	670.7900	672.4670	668.2745	668.2745	Win	0.3750	$100'985.00	$101'363.69	0.949073	High
44	2026-03-09	14:15	SHORT	0.571728	670.2100	671.8855	667.6967	671.8855	Loss	-0.2500	$101'363.69	$101'110.28	0.802922	Med
45	2026-03-09	15:35	LONG	0.473489	679.6600	677.9609	682.2087	677.9609	Loss	-0.2500	$101'110.28	$100'857.50	1.353257	High
46	2026-03-10	12:50	LONG	0.266976	681.6400	679.9359	684.1961	679.9359	Loss	-0.2500	$100'857.50	$100'605.36	0.866111	Med
47	2026-03-10	14:15	LONG	0.331569	678.0900	676.3948	680.6328	676.3948	Loss	-0.2500	$100'605.36	$100'353.85	1.093612	High
48	2026-03-11	10:00	SHORT	0.520672	678.3491	680.0450	675.8053	675.8053	Win	0.3750	$100'353.85	$100'730.17	1.290615	High
49	2026-03-11	11:30	LONG	0.438911	674.8900	673.2028	677.4208	673.2028	Loss	-0.2500	$100'730.17	$100'478.35	1.307403	High
50	2026-03-12	09:35	LONG	0.021612	671.4000	669.7215	673.9177	669.7215	Loss	-0.2500	$100'478.35	$100'227.15	1.145812	High
51	2026-03-12	09:50	SHORT	0.826292	669.0400	670.7126	666.5311	670.7126	Loss	-0.2500	$100'227.15	$99'976.58	1.234745	High
52	2026-03-12	10:05	SHORT	0.896161	670.3200	671.9958	667.8063	667.8063	Win	0.3750	$99'976.58	$100'351.50	1.261407	High
53	2026-03-12	12:00	LONG	0.140249	670.0200	668.3450	672.5326	668.3450	Loss	-0.2500	$100'351.50	$100'100.62	1.192481	High
54	2026-03-12	14:05	SHORT	0.554036	668.6700	670.3417	666.1625	666.1625	Win	0.3750	$100'100.62	$100'476.00	1.055072	High
55	2026-03-13	09:30	LONG	0.108576	669.7850	668.1105	672.2967	672.2967	Win	0.3750	$100'476.00	$100'852.78	1.074105	High
56	2026-03-13	10:35	SHORT	0.672240	668.8800	670.5522	666.3717	666.3717	Win	0.3750	$100'852.78	$101'230.98	1.197882	High
57	2026-03-13	13:30	LONG	0.281234	664.0199	662.3599	666.5100	662.3599	Loss	-0.2500	$101'230.98	$100'977.90	1.057515	High
58	2026-03-13	14:35	SHORT	0.659423	664.1500	665.8104	661.6594	661.6594	Win	0.3750	$100'977.90	$101'356.57	0.891826	Med
59	2026-03-16	11:10	SHORT	0.726995	669.3901	671.0636	666.8799	671.0636	Loss	-0.2500	$101'356.57	$101'103.18	1.019745	High
60	2026-03-16	15:10	SHORT	0.768647	669.8500	671.5246	667.3381	671.5246	Loss	-0.2500	$101'103.18	$100'850.42	0.795272	Med
61	2026-03-17	09:35	LONG	0.107741	673.9050	672.2202	676.4321	672.2202	Loss	-0.2500	$100'850.42	$100'598.29	0.996311	High
62	2026-03-17	10:25	SHORT	0.916012	672.5600	674.2414	670.0379	670.0379	Win	0.3750	$100'598.29	$100'975.54	0.869736	Med
63	2026-03-17	12:10	LONG	0.230214	670.7100	669.0332	673.2252	669.0332	Loss	-0.2500	$100'975.54	$100'723.10	0.844068	Med
64	2026-03-18	09:35	LONG	0.037413	669.0900	667.4173	671.5991	667.4173	Loss	-0.2500	$100'723.10	$100'471.29	0.650186	Med
65	2026-03-18	10:15	SHORT	0.554852	667.5400	669.2088	665.0367	665.0367	Win	0.3750	$100'471.29	$100'848.06	0.880439	Med
66	2026-03-18	14:35	LONG	0.370922	664.7600	663.0981	667.2528	663.0981	Loss	-0.2500	$100'848.06	$100'595.94	0.815796	Med

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
67	2026-03-18	15:25	SHORT	0.829790	662.9100	664.5673	660.4241	660.4241	Win	0.3750	$100'595.94	$100'973.17	0.808296	Med
68	2026-03-19	09:35	SHORT	0.808251	656.2400	657.8806	653.7791	657.8806	Loss	-0.2500	$100'973.17	$100'720.74	1.214142	High
69	2026-03-19	09:50	LONG	0.317139	658.8000	657.1530	661.2705	657.1530	Loss	-0.2500	$100'720.74	$100'468.94	1.256687	High
70	2026-03-19	12:10	SHORT	0.952899	657.8150	659.4595	655.3482	659.4595	Loss	-0.2500	$100'468.94	$100'217.76	1.099261	High
71	2026-03-19	15:05	LONG	0.290918	661.7100	660.0557	664.1914	660.0557	Loss	-0.2500	$100'217.76	$99'967.22	1.077608	High
72	2026-03-19	15:30	SHORT	0.515057	659.9400	661.5899	657.4652	657.4652	Win	0.3750	$99'967.22	$100'342.10	1.143192	High
73	2026-03-20	11:25	LONG	0.255965	651.9200	650.2902	654.3647	650.2902	Loss	-0.2500	$100'342.10	$100'091.24	1.280849	High
74	2026-03-20	14:20	SHORT	0.936044	648.9750	650.5974	646.5413	646.5413	Win	0.3750	$100'091.24	$100'466.58	0.873989	Med
75	2026-03-20	15:40	LONG	0.164608	646.9200	645.3027	649.3459	649.3459	Win	0.3750	$100'466.58	$100'843.33	0.918170	High
76	2026-03-23	10:40	LONG	0.044911	660.3900	658.7390	662.8665	658.7390	Loss	-0.2500	$100'843.33	$100'591.23	1.446922	High
77	2026-03-23	12:15	LONG	0.435097	656.0500	654.4099	658.5102	654.4099	Loss	-0.2500	$100'591.23	$100'339.75	1.297461	High
78	2026-03-23	14:00	SHORT	0.992376	658.1500	659.7954	655.6819	655.6819	Win	0.3750	$100'339.75	$100'716.02	1.105591	High
79	2026-03-24	10:05	SHORT	0.891677	653.4100	655.0435	650.9597	655.0435	Loss	-0.2500	$100'716.02	$100'464.23	1.273300	High
80	2026-03-24	12:15	SHORT	0.748608	655.9100	657.5498	653.4503	653.4503	Win	0.3750	$100'464.23	$100'840.97	1.164930	High
81	2026-03-24	14:35	SHORT	0.890792	656.5100	658.1513	654.0481	654.0481	Win	0.3750	$100'840.97	$101'219.13	1.026384	High
82	2026-03-24	15:35	SHORT	0.893447	653.9800	655.6150	651.5276	655.6150	Loss	-0.2500	$101'219.13	$100'966.08	0.864876	Med
83	2026-03-25	10:20	SHORT	0.518858	657.4000	659.0435	654.9348	659.0435	Loss	-0.2500	$100'966.08	$100'713.66	1.306360	High
84	2026-03-25	13:35	LONG	0.315929	658.6200	656.9735	661.0898	656.9735	Loss	-0.2500	$100'713.66	$100'461.88	0.866496	Med
85	2026-03-25	15:10	SHORT	0.772012	656.4300	658.0711	653.9684	653.9684	Win	0.3750	$100'461.88	$100'838.61	0.717818	Med
86	2026-03-26	10:40	SHORT	0.661661	652.6600	654.2916	650.2125	650.2125	Win	0.3750	$100'838.61	$101'216.76	1.184632	High
87	2026-03-26	12:20	LONG	0.373658	650.3600	648.7341	652.7989	648.7341	Loss	-0.2500	$101'216.76	$100'963.71	0.935823	High
88	2026-03-26	13:45	LONG	0.094467	647.5800	645.9611	650.0084	645.9611	Loss	-0.2500	$100'963.71	$100'711.30	0.720295	Med
89	2026-03-27	10:05	SHORT	0.746790	638.8500	640.4471	636.4543	640.4471	Loss	-0.2500	$100'711.30	$100'459.53	1.101295	High
90	2026-03-27	12:05	LONG	0.262461	639.1800	637.5820	641.5769	637.5820	Loss	-0.2500	$100'459.53	$100'208.38	1.027212	High
91	2026-03-27	13:25	SHORT	0.936813	636.6200	638.2115	634.2327	634.2327	Win	0.3750	$100'208.38	$100'584.16	0.858847	Med
92	2026-03-27	15:55	LONG	0.240971	634.0800	632.4948	636.4578	636.4578	Win	0.3750	$100'584.16	$100'961.35	0.734280	Med
93	2026-03-30	11:25	LONG	0.122758	636.4600	634.8689	638.8467	634.8689	Loss	-0.2500	$100'961.35	$100'708.95	1.133059	High
94	2026-03-30	13:05	SHORT	0.831113	635.4000	636.9885	633.0172	633.0172	Win	0.3750	$100'708.95	$101'086.60	0.807984	Med
95	2026-03-30	15:15	LONG	0.153284	630.0300	628.4549	632.3926	632.3926	Win	0.3750	$101'086.60	$101'465.68	0.691829	Med
96	2026-03-31	09:30	LONG	0.179268	639.7300	638.1307	642.1290	642.1290	Win	0.3750	$101'465.68	$101'846.18	1.300691	High
97	2026-03-31	11:15	SHORT	0.599383	641.7500	643.3544	639.3434	643.3544	Loss	-0.2500	$101'846.18	$101'591.56	1.261506	High
98	2026-03-31	14:05	SHORT	0.874562	647.6600	649.2791	645.2313	649.2791	Loss	-0.2500	$101'591.56	$101'337.58	1.374998	High
99	2026-03-31	15:55	LONG	0.196435	650.2400	648.6144	652.6784	652.6784	Win	0.3750	$101'337.58	$101'717.60	1.022709	High
100	2026-04-01	10:20	LONG	0.310324	654.6200	652.9835	657.0748	657.0748	Win	0.3750	$101'717.60	$102'099.04	1.315219	High

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
101	2026-04-01	13:35	SHORT	0.777405	658.0800	659.7252	655.6122	655.6122	Win	0.3750	$102'099.04	$102'481.91	0.500013	Low
102	2026-04-01	14:20	SHORT	0.971826	654.1300	655.7653	651.6770	655.7653	Loss	-0.2500	$102'481.91	$102'225.70	0.674298	Med
103	2026-04-02	10:55	SHORT	0.500741	655.3900	657.0285	652.9323	652.9323	Win	0.3750	$102'225.70	$102'609.05	1.760355	High
104	2026-04-02	12:10	LONG	0.143898	653.3882	651.7547	655.8384	655.8384	Win	0.3750	$102'609.05	$102'993.84	1.374770	High
105	2026-04-06	09:30	LONG	0.013936	657.3700	655.7266	659.8351	655.7266	Loss	-0.2500	$102'993.84	$102'736.35	0.724549	Med
106	2026-04-06	10:05	LONG	0.229656	657.1651	655.5222	659.6295	655.5222	Loss	-0.2500	$102'736.35	$102'479.51	1.120577	High
107	2026-04-06	13:15	LONG	0.131822	656.4800	654.8388	658.9418	654.8388	Loss	-0.2500	$102'479.51	$102'223.31	0.798855	Med
108	2026-04-07	09:50	SHORT	0.677659	655.7200	657.3593	653.2610	653.2610	Win	0.3750	$102'223.31	$102'606.65	0.826768	Med
109	2026-04-07	10:10	LONG	0.121833	652.7551	651.1232	655.2029	655.2029	Win	0.3750	$102'606.65	$102'991.42	0.929718	High
110	2026-04-07	11:00	SHORT	0.506330	651.8669	653.4966	649.4224	653.4966	Loss	-0.2500	$102'991.42	$102'733.94	1.005142	High
111	2026-04-07	12:05	SHORT	0.694262	654.7800	656.4169	652.3246	656.4169	Loss	-0.2500	$102'733.94	$102'477.11	1.113886	High
112	2026-04-07	15:00	SHORT	0.581117	654.0800	655.7152	651.6272	655.7152	Loss	-0.2500	$102'477.11	$102'220.92	0.730567	Med
113	2026-04-08	10:00	LONG	0.199776	675.0550	673.3674	677.5865	673.3674	Loss	-0.2500	$102'220.92	$101'965.36	1.901794	High
114	2026-04-08	11:45	SHORT	0.804125	674.7850	676.4720	672.2546	676.4720	Loss	-0.2500	$101'965.36	$101'710.45	1.027125	High
115	2026-04-08	14:20	SHORT	0.715407	675.0000	676.6875	672.4688	676.6875	Loss	-0.2500	$101'710.45	$101'456.18	0.710226	Med
116	2026-04-09	11:35	SHORT	0.738984	677.2400	678.9331	674.7003	678.9331	Loss	-0.2500	$101'456.18	$101'202.53	0.801321	Med
117	2026-04-09	12:05	LONG	0.131058	679.8700	678.1703	682.4195		Unresolved		$101'202.53	$101'202.53	0.851788	Med
118	2026-04-10	12:10	LONG	0.123754	679.9400	678.2402	682.4898	678.2402	Loss	-0.2500	$101'202.53	$100'949.53	0.601701	Low
119	2026-04-13	09:35	SHORT	0.927563	677.2500	678.9431	674.7103	678.9431	Loss	-0.2500	$100'949.53	$100'697.15	0.591113	Low
120	2026-04-13	09:50	LONG	0.397578	678.6600	676.9633	681.2050	681.2050	Win	0.3750	$100'697.15	$101'074.77	0.763607	Med
121	2026-04-13	12:30	LONG	0.300949	681.3500	679.6466	683.9051	683.9051	Win	0.3750	$101'074.77	$101'453.80	0.659515	Med
122	2026-04-13	15:35	LONG	0.488584	684.1800	682.4696	686.7457	686.7457	Win	0.3750	$101'453.80	$101'834.25	0.362037	Low
123	2026-04-14	09:35	SHORT	0.662864	689.0300	690.7526	686.4461	690.7526	Loss	-0.2500	$101'834.25	$101'579.67	0.655168	Med
124	2026-04-14	10:15	SHORT	0.955623	690.6600	692.3866	688.0700	692.3866	Loss	-0.2500	$101'579.67	$101'325.72	0.618746	Low
125	2026-04-14	11:30	LONG	0.286446	692.4700	690.7388	695.0668	695.0668	Win	0.3750	$101'325.72	$101'705.69	0.506468	Low
126	2026-04-15	09:35	SHORT	0.924808	696.2800	698.0207	693.6689	698.0207	Loss	-0.2500	$101'705.69	$101'451.42	0.586998	Low
127	2026-04-15	14:05	LONG	0.024859	698.1000	696.3548	700.7179	700.7179	Win	0.3750	$101'451.42	$101'831.87	0.487449	Low
128	2026-04-16	09:35	SHORT	0.555198	700.6200	702.3715	697.9927	702.3715	Loss	-0.2500	$101'831.87	$101'577.29	0.524766	Low
129	2026-04-16	12:05	SHORT	0.633975	702.3199	704.0757	699.6862	699.6862	Win	0.3750	$101'577.29	$101'958.20	0.626905	Low
130	2026-04-16	14:30	LONG	0.105897	700.4800	698.7288	703.1068	703.1068	Win	0.3750	$101'958.20	$102'340.54	0.636121	Low
131	2026-04-17	09:35	LONG	0.140340	707.0900	705.3223	709.7416	709.7416	Win	0.3750	$102'340.54	$102'724.32	0.826129	Med
132	2026-04-17	10:35	LONG	0.419114	710.3300	708.5542	712.9937	708.5542	Loss	-0.2500	$102'724.32	$102'467.51	0.793115	Med
133	2026-04-20	09:35	SHORT	0.966232	708.6400	710.4116	705.9826	710.4116	Loss	-0.2500	$102'467.51	$102'211.34	0.648746	Med
134	2026-04-21	09:35	SHORT	0.596043	709.0399	710.8125	706.3810	710.8125	Loss	-0.2500	$102'211.34	$101'955.81	0.518687	Low

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
135	2026-04-21	09:55	SHORT	0.933023	710.7500	712.5269	708.0847	708.0847	Win	0.3750	$101'955.81	$102'338.15	0.561402	Low
136	2026-04-21	11:00	SHORT	0.804361	706.7350	708.5018	704.0847	704.0847	Win	0.3750	$102'338.15	$102'721.92	0.748037	Med
137	2026-04-21	13:20	LONG	0.467382	704.4500	702.6889	707.0917	707.0917	Win	0.3750	$102'721.92	$103'107.12	0.801681	Med
138	2026-04-21	13:45	SHORT	0.784763	706.1800	707.9454	703.5318	703.5318	Win	0.3750	$103'107.12	$103'493.78	0.872669	Med
139	2026-04-21	15:45	LONG	0.017837	703.9999	702.2399	706.6399	706.6399	Win	0.3750	$103'493.78	$103'881.88	1.053377	High
140	2026-04-22	09:35	LONG	0.109144	708.9000	707.1277	711.5584		Unresolved		$103'881.88	$103'881.88	1.335290	High
141	2026-04-23	09:40	SHORT	0.829429	710.5700	712.3464	707.9054	712.3464	Loss	-0.2500	$103'881.88	$103'622.17	0.576655	Low
142	2026-04-23	11:35	SHORT	0.796817	711.8050	713.5845	709.1357	709.1357	Win	0.3750	$103'622.17	$104'010.76	0.526978	Low
143	2026-04-23	13:05	LONG	0.232641	707.9300	706.1602	710.5847	706.1602	Loss	-0.2500	$104'010.76	$103'750.73	0.667288	Med
144	2026-04-23	13:25	SHORT	0.530770	706.0500	707.8151	703.4023	703.4023	Win	0.3750	$103'750.73	$104'139.79	0.818849	Med
145	2026-04-23	13:45	SHORT	0.606016	703.5396	705.2984	700.9013	705.2984	Loss	-0.2500	$104'139.79	$103'879.44	1.145184	High
146	2026-04-23	13:55	SHORT	0.867739	706.6350	708.4016	703.9851	708.4016	Loss	-0.2500	$103'879.44	$103'619.75	1.387480	High
147	2026-04-23	14:25	SHORT	0.603107	708.5101	710.2814	705.8532	710.2814	Loss	-0.2500	$103'619.75	$103'360.70	1.245153	High
148	2026-04-24	09:35	LONG	0.412572	710.3100	708.5342	712.9737	712.9737	Win	0.3750	$103'360.70	$103'748.30	1.065771	High
149	2026-04-24	11:45	LONG	0.374184	713.2800	711.4968	715.9548		Unresolved		$103'748.30	$103'748.30	0.825236	Med
150	2026-04-27	11:50	LONG	0.425882	713.2091	711.4261	715.8836	711.4261	Loss	-0.2500	$103'748.30	$103'488.93	0.498223	Low
151	2026-04-28	09:35	SHORT	0.651931	712.0700	713.8502	709.3997	709.3997	Win	0.3750	$103'488.93	$103'877.01	0.586931	Low
152	2026-04-28	10:55	SHORT	0.867491	710.5200	712.2963	707.8555		Unresolved		$103'877.01	$103'877.01	0.790239	Med
153	2026-04-29	11:00	LONG	0.453897	711.1100	709.3322	713.7767	709.3322	Loss	-0.2500	$103'877.01	$103'617.32	0.446731	Low
154	2026-04-29	11:55	LONG	0.247840	710.0400	708.2649	712.7026	712.7026	Win	0.3750	$103'617.32	$104'005.88	0.588380	Low
155	2026-04-30	09:35	LONG	0.236662	711.8400	710.0604	714.5094	714.5094	Win	0.3750	$104'005.88	$104'395.91	0.884313	Med
156	2026-04-30	11:15	SHORT	0.746014	714.2100	715.9955	711.5317	715.9955	Loss	-0.2500	$104'395.91	$104'134.92	0.889017	Med
157	2026-04-30	13:35	SHORT	0.816569	716.4900	718.2812	713.8032	718.2812	Loss	-0.2500	$104'134.92	$103'874.58	0.549302	Low
158	2026-04-30	14:35	LONG	0.105278	718.5100	716.7137	721.2044	721.2044	Win	0.3750	$103'874.58	$104'264.11	0.463840	Low
159	2026-05-01	09:35	LONG	0.066559	722.1100	720.3047	724.8179	724.8179	Win	0.3750	$104'264.11	$104'655.10	0.752488	Med
160	2026-05-01	10:25	SHORT	0.594434	724.2588	726.0694	721.5428	721.5428	Win	0.3750	$104'655.10	$105'047.56	0.721001	Med
161	2026-05-01	12:50	LONG	0.146173	722.1500	720.3446	724.8581	720.3446	Loss	-0.2500	$105'047.56	$104'784.94	0.640268	Med
162	2026-05-04	09:35	SHORT	0.824664	719.8400	721.6396	717.1406	721.6396	Loss	-0.2500	$104'784.94	$104'522.97	0.555428	Low
163	2026-05-04	10:25	LONG	0.310335	721.2400	719.4369	723.9446	719.4369	Loss	-0.2500	$104'522.97	$104'261.67	0.614385	Low
164	2026-05-04	11:20	LONG	0.143872	717.6800	715.8858	720.3713	715.8858	Loss	-0.2500	$104'261.67	$104'001.01	0.817674	Med
165	2026-05-04	12:10	SHORT	0.920970	715.9400	717.7299	713.2552	717.7299	Loss	-0.2500	$104'001.01	$103'741.01	1.032018	High
166	2026-05-04	12:25	LONG	0.165532	717.0000	715.2075	719.6887	719.6887	Win	0.3750	$103'741.01	$104'130.04	1.090497	High
167	2026-05-05	09:35	LONG	0.284720	722.0600	720.2549	724.7677	724.7677	Win	0.3750	$104'130.04	$104'520.53	0.793597	Med
168	2026-05-05	15:15	LONG	0.153613	724.8800	723.0678	727.5983	727.5983	Win	0.3750	$104'520.53	$104'912.48	0.262800	Low

Trade #	Date	Time	Dir	Random Value	Entry Price	Stop Loss	Take Profit	Exit Price	Result	% Return	Account Before	Account After	ATR at Entry	Vol Cond
169	2026-05-06	09:35	LONG	0.115490	729.0800	727.2573	731.8140	731.8140	Win	0.3750	$104'912.48	$105'305.90	0.695235	Med
170	2026-05-06	10:45	LONG	0.021148	731.7900	729.9605	734.5342	734.5342	Win	0.3750	$105'305.90	$105'700.80	0.690644	Med
171	2026-05-06	15:55	LONG	0.055395	733.7700	731.9356	736.5216	731.9356	Loss	-0.2500	$105'700.80	$105'436.55	0.414904	Low
172	2026-05-07	12:20	LONG	0.174641	732.7000	730.8683	735.4476	730.8683	Loss	-0.2500	$105'436.55	$105'172.95	0.645912	Med
173	2026-05-07	13:30	LONG	0.053382	731.0000	729.1725	733.7412	733.7412	Win	0.3750	$105'172.95	$105'567.35	0.870343	Med
174	2026-05-08	09:35	SHORT	0.591144	734.9300	736.7673	732.1740	736.7673	Loss	-0.2500	$105'567.35	$105'303.43	0.969117	High
175	2026-05-08	10:05	SHORT	0.680715	736.5400	738.3813	733.7780	738.3813	Loss	-0.2500	$105'303.43	$105'040.18	0.871474	Med
176	2026-05-11	09:45	LONG	0.393630	738.7800	736.9330	741.5504	736.9330	Loss	-0.2500	$105'040.18	$104'777.58	0.537354	Low
177	2026-05-12	09:35	LONG	0.317991	736.0900	734.2498	738.8503	734.2498	Loss	-0.2500	$104'777.58	$104'515.63	0.658549	Med
178	2026-05-12	10:35	SHORT	0.504526	734.6400	736.4766	731.8851	731.8851	Win	0.3750	$104'515.63	$104'907.56	0.869946	Med
179	2026-05-12	11:30	SHORT	0.875005	732.8700	734.7022	730.1217	734.7022	Loss	-0.2500	$104'907.56	$104'645.30	0.814402	Med
180	2026-05-12	13:25	SHORT	0.851132	734.3500	736.1859	731.5962	736.1859	Loss	-0.2500	$104'645.30	$104'383.68	0.675633	Med
181	2026-05-12	14:30	LONG	0.043475	736.1200	734.2797	738.8804	738.8804	Win	0.3750	$104'383.68	$104'775.12	0.589056	Low
182	2026-05-13	10:55	LONG	0.181498	738.8300	736.9829	741.6006	741.6006	Win	0.3750	$104'775.12	$105'168.03	0.684330	Med
183	2026-05-13	12:25	LONG	0.236745	741.7250	739.8707	744.5065	744.5065	Win	0.3750	$105'168.03	$105'562.41	0.572573	Low

(F) Python code for SPY data retrieval

"""

Kelly Criterion IB Maths IA — No-Leverage Simulation (ONL)
Primary model: each trade risks f_actual = 0.25 % of current account (no borrowed capital).
Win → account × (1 + TP_PCT) i.e. +0.375 % of account

Loss → account × (1 − SL_PCT)   i.e. −0.25 % of account

Same entry selection as LLL (identical rng draw sequence, seed 42) → same 183 trades,
same Win/Loss/Unresolved outcomes. Only the P&L mechanic differs.
"""

```
import math, os, warnings
import numpy as np
import pandas as pd
import matplotlib
matplotlib.use("Agg")
import matplotlib.pyplot as plt
import matplotlib.ticker as mticker
import openpyxl
from openpyxl.styles import (PatternFill, Font, Alignment, Border, Side)
from openpyxl.utils import get_column_letter
from openpyxl.drawing.image import Image as XLImage

warnings.filterwarnings("ignore")

# ─────────────────────────────────────────────────────────────────────────────
# PATHS
# ─────────────────────────────────────────────────────────────────────────────
BASE   = "/Users/nicholasportelli/Documents/Claude/Projects/Kelly Criterion Maths IA"
CSV    = os.path.join(BASE, "spy_candles_raw.csv")
XLFILE = os.path.join(BASE, "2026-05-14-kelly-criterion-dataset-ONL.xlsx")
IMGDIR = os.path.join(BASE, "charts-ONL")
os.makedirs(IMGDIR, exist_ok=True)

# ─────────────────────────────────────────────────────────────────────────────
# CONSTANTS
# ─────────────────────────────────────────────────────────────────────────────
SEED          = 42
N_TRADES_TARGET = 200
ACCOUNT       = 100_000.0
SL_PCT        = 0.0025      # 0.25 % stop-loss  (= f_actual in Kelly terms)
TP_PCT        = 0.00375     # 0.375 % take-profit → R = 1.5 by design
MAX_CANDLES   = 78
RR_DESIGN     = 1.5
F_ACTUAL      = SL_PCT      # fraction of account lost per losing trade = 0.0025

rng = np.random.default_rng(SEED)

# ─────────────────────────────────────────────────────────────────────────────
# LOAD DATA
# ─────────────────────────────────────────────────────────────────────────────
df = pd.read_csv(CSV)
df["datetime"] = pd.to_datetime(df["date"] + " " + df["time"])
df = df.sort_values("datetime").reset_index(drop=True)
N  = len(df)
```

```
high       = df["high"].values
low        = df["low"].values
close      = df["close"].values
dates_arr  = df["date"].values

# ─────────────────────────────────────────────────────────────────────
# ATR-14 (Wilder smoothing)
# ─────────────────────────────────────────────────────────────────────
tr = np.zeros(N)
tr[0] = high[0] - low[0]
for i in range(1, N):
    tr[i] = max(high[i] - low[i],
                abs(high[i] - close[i-1]),
                abs(low[i]  - close[i-1]))

atr14 = np.zeros(N)
atr14[13] = np.mean(tr[:14])
for i in range(14, N):
    atr14[i] = (atr14[i-1] * 13 + tr[i]) / 14

valid_atr = atr14[atr14 > 0]
p33 = np.percentile(valid_atr, 33.33)
p67 = np.percentile(valid_atr, 66.67)

def vol_condition(a):
    if a <= 0:   return "N/A"
    if a <= p33: return "Low"
    if a <= p67: return "Med"
    return "High"

# ─────────────────────────────────────────────────────────────────────
# SIMULATE TRADES  (bucket method, identical to LLL)
# ─────────────────────────────────────────────────────────────────────
usable      = N - MAX_CANDLES
bucket_size = usable / N_TRADES_TARGET

entry_indices = []
for i in range(N_TRADES_TARGET):
    lo  = int(i * bucket_size)
    hi  = min(int((i + 1) * bucket_size), usable - 1)
    if lo > hi: lo = hi
    entry_indices.append(int(rng.integers(lo, hi + 1)))

trades       = []
trade_num    = 0
last_end_idx = -1
account      = ACCOUNT          # running account value (compounds trade-by-trade)

for entry_idx in entry_indices:
```

```
if entry_idx <= last_end_idx:
    entry_idx = last_end_idx + 1
if entry_idx >= usable:
    continue

rand_val    = float(rng.random())
direction   = "LONG" if rand_val < 0.5 else "SHORT"
entry_price = float(close[entry_idx])

if direction == "LONG":
    sl = entry_price * (1 - SL_PCT)
    tp = entry_price * (1 + TP_PCT)
else:
    sl = entry_price * (1 + SL_PCT)
    tp = entry_price * (1 - TP_PCT)

# Scan subsequent candles
result        = "Unresolved"
exit_price    = None
trade_end_idx = entry_idx + MAX_CANDLES

for j in range(entry_idx + 1, min(entry_idx + MAX_CANDLES + 1, N)):
    h, l = high[j], low[j]
    if direction == "LONG":
        tp_hit = h >= tp
        sl_hit = l <= sl
    else:
        tp_hit = l <= tp
        sl_hit = h >= sl

    if tp_hit and sl_hit:
        result        = "Ambiguous"
        trade_end_idx = j
        break
    elif tp_hit:
        result        = "Win"
        exit_price    = tp
        trade_end_idx = j
        break
    elif sl_hit:
        result        = "Loss"
        exit_price    = sl
        trade_end_idx = j
        break

# No-leverage P&L: account updates as a percentage of current equity
acct_before = account
if result == "Win":
    pct_return  = TP_PCT
    account     = acct_before * (1 + TP_PCT)
```

```
    elif result == "Loss":
        pct_return  = -SL_PCT
        account     = acct_before * (1 - SL_PCT)
    else:
        pct_return  = None
        # Unresolved / Ambiguous: account unchanged

    acct_after = account

    trade_num += 1
    trades.append({
        "Trade #":         trade_num,
        "Date":            dates_arr[entry_idx],
        "Time":            df["time"].iloc[entry_idx],
        "Direction":       direction,
        "Random Value":    round(rand_val, 6),
        "Entry Price":     round(entry_price, 4),
        "Stop Loss":       round(sl, 4),
        "Take Profit":     round(tp, 4),
        "Exit Price":      round(exit_price, 4) if exit_price else None,
        "Result":          result,
        "% Return":        round(pct_return * 100, 4) if pct_return is not None else None,
        "Account Before":  round(acct_before, 2),
        "Account After":   round(acct_after, 2),
        "ATR at Entry":    round(float(atr14[entry_idx]), 6),
        "Volatility Cond": vol_condition(atr14[entry_idx]),
    })

    last_end_idx = trade_end_idx

N_TRADES = len(trades)
tdf = pd.DataFrame(trades)

# ──────────────────────────────────────────────────────────────────
# KELLY METRICS  (completed = Win or Loss only)
# ──────────────────────────────────────────────────────────────────
completed = tdf[tdf["Result"].isin(["Win", "Loss"])].copy()
wins      = completed[completed["Result"] == "Win"]
losses    = completed[completed["Result"] == "Loss"]

n_comp   = len(completed)
n_wins   = len(wins)
n_losses = len(losses)
n_unres  = (tdf["Result"] == "Unresolved").sum()
n_ambig  = (tdf["Result"] == "Ambiguous").sum()

p        = n_wins / n_comp  if n_comp > 0 else 0
avg_gain = wins["% Return"].mean()   / 100 if n_wins   > 0 else TP_PCT
avg_loss = abs(losses["% Return"].mean()) / 100 if n_losses > 0 else SL_PCT
R        = avg_gain / avg_loss if avg_loss > 0 else RR_DESIGN
```

```
f_star      = (p * (R + 1) - 1) / R
p_star      = 1 / (1 + R)
f_practical = max(f_star, 0.0)

def G(f, p_val, R_val):
    """Expected log-growth per trade at Kelly fraction f."""
    if f <= 0:
        return 0.0
    w = 1 + f * R_val
    l = 1 - f
    if w <= 0 or l <= 0:
        return -math.inf
    return p_val * math.log(w) + (1 - p_val) * math.log(l)

G_opt    = G(f_practical, p, R)
G_actual = G(F_ACTUAL, p, R)     # G at the ONL betting fraction

print("=" * 52)
print(f"Total trades simulated : {N_TRADES}")
print(f"  Wins                 : {n_wins}")
print(f"  Losses               : {n_losses}")
print(f"  Unresolved           : {n_unres}")
print(f"  Ambiguous            : {n_ambig}")
print(f"  Completed (W+L)      : {n_comp}")
print(f"p (win rate)           : {p:.6f}")
print(f"R (avg gain / avg loss): {R:.6f}")
print(f"f* (Kelly formula)     : {f_star:.6f}")
print(f"f_actual (ONL)         : {F_ACTUAL:.6f}")
print(f"Breakeven p*           : {p_star:.6f}")
print(f"G(f*)                  : {G_opt:.8f}")
print(f"G(f_actual)            : {G_actual:.8f}")
print(f"Final account          : ${account:,.2f}")
print("=" * 52)

# ─────────────────────────────────────────────────────────────
# CHART STYLE HELPERS
# ─────────────────────────────────────────────────────────────
NAVY   = "#1B3A6B"
BLUE   = "#2E5FA3"
GREEN  = "#27AE60"
RED    = "#C0392B"
ORANGE = "#E67E22"
GREY   = "#7F8C8D"
PURPLE = "#8E44AD"

def save_fig(fig, name):
    path = os.path.join(IMGDIR, name)
    fig.savefig(path, dpi=150, bbox_inches="tight")
    plt.close(fig)
```

```
    return path

def style_ax(ax, title="", xlabel="", ylabel=""):
    ax.set_title(title, fontsize=13, fontweight="bold", color=NAVY, pad=10)
    ax.set_xlabel(xlabel, fontsize=10, color="#333333")
    ax.set_ylabel(ylabel, fontsize=10, color="#333333")
    ax.spines["top"].set_visible(False)
    ax.spines["right"].set_visible(False)
    ax.spines["left"].set_color("#CCCCCC")
    ax.spines["bottom"].set_color("#CCCCCC")
    ax.tick_params(colors="#555555", labelsize=9)
    ax.yaxis.set_minor_locator(mticker.AutoMinorLocator())
    ax.grid(which="major", linestyle="--", linewidth=0.5, color="#E0E0E0", alpha=0.8)
    ax.grid(which="minor", linestyle=":", linewidth=0.3, color="#EEEEEE", alpha=0.6)
    ax.set_facecolor("#FAFAFA")

IMG_PATHS = {}

# ─────────────────────────────────────────────────────────────────────
# CHART 1 — G(f) curve
# Shows BOTH f_actual (ONL) AND f* (Kelly optimal)
# ─────────────────────────────────────────────────────────────────────
f_arr = np.linspace(0, 0.99, 2000)
G_arr = np.array([G(f, p, R) if (1 - f) > 0 else np.nan for f in f_arr])
G_arr = np.where(np.isfinite(G_arr), G_arr, np.nan)

valid_mask = np.isfinite(G_arr)
G_max_vis  = float(np.nanmax(G_arr[valid_mask]))

fig, ax = plt.subplots(figsize=(8.5, 5))
ax.plot(f_arr, G_arr, color=BLUE, linewidth=2.2, zorder=3, label="$G(f)$")

# y = 0 baseline
ax.axhline(0, color="#999999", linewidth=1.1, linestyle="-", zorder=2,
           label="$G = 0$ (no-bet baseline)")

# f* — Kelly-optimal maximum
G_at_fstar = G(f_star, p, R)
ax.axvline(f_star, color=ORANGE, linewidth=1.6, linestyle="--", zorder=4,
           label=f"$f^* = {f_star:.4f}$ (Kelly optimal)")
ax.scatter([f_star], [G_at_fstar], color=ORANGE, s=90, zorder=6,
           label=f"$G(f^*) = {G_at_fstar:.5f}$")
ax.annotate(
    f"Kelly optimal\n$f^* = {f_star:.4f}$\n$G(f^*) = {G_at_fstar:.5f}$",
    xy=(f_star, G_at_fstar),
    xytext=(min(f_star + 0.09, 0.80), G_at_fstar - G_max_vis * 0.18),
    fontsize=8.5, color=ORANGE,
    arrowprops=dict(arrowstyle="->", color=ORANGE, lw=1.1),
    bbox=dict(boxstyle="round,pad=0.35", facecolor="white",
              edgecolor=ORANGE, alpha=0.9))
```

```
# f_actual — ONL actual betting fraction
ax.axvline(F_ACTUAL, color=PURPLE, linewidth=1.6, linestyle=":", zorder=5,
        label=f"$f_{{actual}} = {F_ACTUAL:.4f}$ (ONL — no leverage)")
ax.scatter([F_ACTUAL], [G_actual], color=PURPLE, s=80, zorder=7,
        label=f"$G(f_{{actual}}) = {G_actual:.6f}$")
ax.annotate(
   f"ONL actual\n$f_{{actual}} = {F_ACTUAL:.4f}$\n$G = {G_actual:.6f}$",
   xy=(F_ACTUAL, G_actual),
   xytext=(F_ACTUAL + 0.07, G_actual + G_max_vis * 0.20),
   fontsize=8.5, color=PURPLE,
   arrowprops=dict(arrowstyle="->", color=PURPLE, lw=1.1),
   bbox=dict(boxstyle="round,pad=0.35", facecolor="white",
           edgecolor=PURPLE, alpha=0.9))

y_lo = -0.005
y_hi = G_max_vis * 1.20
ax.set_xlim(0, min(f_star * 3.5, 0.40))
ax.set_ylim(y_lo, y_hi)
style_ax(ax,
       f"Kelly Growth Function  $G(f) = p \\cdot \\ln(1+fR) + (1-p) \\cdot \\ln(1-f)$\n"
       f"$p = {p:.4f}$,  $R = {R:.4f}$  — ONL uses $f_{{actual}} = {F_ACTUAL:.4f}$ vs $f^* =
{f_star:.4f}$",
       "Kelly fraction $f$",
       "Expected log-growth $G(f)$ per trade")
ax.legend(fontsize=8, framealpha=0.95, loc="upper left")
fig.tight_layout()
IMG_PATHS["gf_curve"] = save_fig(fig, "chart1_gf_curve-ONL.png")

# ──────────────────────────────────────────────────────────────
# CHART 2 — f*(p), R = 1.5 fixed
# ──────────────────────────────────────────────────────────────
p_range = np.linspace(0.25, 0.80, 600)
fstar_p = (p_range * (R + 1) - 1) / R

fig, ax = plt.subplots(figsize=(8, 5))
ax.plot(p_range, fstar_p, color=BLUE, linewidth=2.2, label=f"$f^*(p)$  with $R = {R:.4f}$")
ax.axhline(0, color="#999999", linewidth=1.0, linestyle="-")
ax.fill_between(p_range, fstar_p, 0, where=(fstar_p > 0),
            alpha=0.12, color=GREEN, label="Positive Kelly region")
ax.fill_between(p_range, fstar_p, 0, where=(fstar_p < 0),
            alpha=0.12, color=RED, label="No-bet region")
ax.axvline(p, color=ORANGE, linewidth=1.6, linestyle="--",
        label=f"Empirical $p = {p:.4f}$")
ax.axvline(p_star, color=RED, linewidth=1.2, linestyle=":",
        label=f"Breakeven $p^* = {p_star:.3f}$")
emp_fstar = (p * (R + 1) - 1) / R
ax.scatter([p], [emp_fstar], color=ORANGE, s=80, zorder=5)
style_ax(ax, f"Optimal Kelly Fraction vs Win Probability  ($R = {R:.4f}$) [ONL]",
       "Win probability $p$", "Optimal Kelly fraction $f^*$")
```

```
ax.legend(fontsize=8.5, framealpha=0.95)
fig.tight_layout()
IMG_PATHS["fp_curve"] = save_fig(fig, "chart2_fp_curve-ONL.png")

# ──────────────────────────────────────────────────────────────────
# CHART 3 — f*(R), empirical p fixed
# ──────────────────────────────────────────────────────────────────
R_range = np.linspace(0.5, 4.0, 600)
fstar_R = (p * (R_range + 1) - 1) / R_range

fig, ax = plt.subplots(figsize=(8, 5))
ax.plot(R_range, fstar_R, color=BLUE, linewidth=2.2, label=f"$f^*(R)$  with $p = {p:.4f}$")
ax.axhline(0, color="#999999", linewidth=1.0, linestyle="-")
ax.fill_between(R_range, fstar_R, 0, where=(fstar_R > 0), alpha=0.12, color=GREEN)
ax.fill_between(R_range, fstar_R, 0, where=(fstar_R < 0), alpha=0.12, color=RED)
ax.axvline(R, color=ORANGE, linewidth=1.6, linestyle="--",
           label=f"Empirical $R = {R:.4f}$")
ax.axvline(RR_DESIGN, color=GREY, linewidth=1.2, linestyle=":",
           label=f"Design $R = {RR_DESIGN}$")
ax.scatter([R], [f_star], color=ORANGE, s=80, zorder=5)
style_ax(ax, f"Optimal Kelly Fraction vs Reward-to-Risk Ratio  ($p = {p:.4f}$) [ONL]",
         "Reward-to-risk ratio $R$", "Optimal Kelly fraction $f^*$")
ax.legend(fontsize=8.5, framealpha=0.95)
fig.tight_layout()
IMG_PATHS["fr_curve"] = save_fig(fig, "chart3_fr_curve-ONL.png")

# ──────────────────────────────────────────────────────────────────
# CHART 4 — Wealth curves
# Shows ONL actual (f_actual), f*/2, f*, 2f*
# Illustrates that ONL significantly under-bets Kelly optimal
# ──────────────────────────────────────────────────────────────────
completed_sorted = completed.sort_values("Trade #").reset_index(drop=True)
returns_dec      = completed_sorted["% Return"].values / 100

def wealth_series(fraction, start=ACCOUNT):
    vals = [start]
    w    = start
    for r in returns_dec:
        w = w * (1 + fraction * R) if r > 0 else w * (1 - fraction)
        vals.append(max(w, 0.0))
    return np.array(vals)

wActual = wealth_series(F_ACTUAL)         # ONL actual (0.0025)
wHalf   = wealth_series(f_star / 2)       # Half Kelly
wFull   = wealth_series(f_star)           # Full Kelly
wDouble = wealth_series(f_star * 2)       # Double Kelly
xs = np.arange(len(wActual))

lbl_actual = f"ONL actual  ($f_{{actual}} = {F_ACTUAL:.4f}$)"
lbl_half   = f"Half Kelly  ($f = {f_star/2:.4f}$)"
```

```
lbl_full   = f"Full Kelly  ($f^* = {f_star:.4f}$)"
lbl_double = f"Double Kelly  ($f = {f_star*2:.4f}$)"

fig, ax = plt.subplots(figsize=(9, 5))
ax.plot(xs, wActual, color=PURPLE, linewidth=2.0, label=lbl_actual)
ax.plot(xs, wHalf,   color=GREEN,  linewidth=2.0, label=lbl_half)
ax.plot(xs, wFull,   color=BLUE,   linewidth=2.2, label=lbl_full)
ax.plot(xs, wDouble, color=RED,    linewidth=1.8, linestyle="--", label=lbl_double)
ax.axhline(ACCOUNT, color="#AAAAAA", linewidth=0.9, linestyle=":",
           label=f"Initial \\${ACCOUNT:,.0f}")
style_ax(ax,
         f"Account Growth: ONL Actual vs Kelly Fractions [ONL]",
         "Resolved trade number", "Account value (\\$)")
ax.yaxis.set_major_formatter(mticker.FuncFormatter(lambda x, _: f"${x:,.0f}"))
ax.legend(fontsize=8.5, framealpha=0.95)
ax.set_xlim(0, len(xs) - 1)
fig.tight_layout()
IMG_PATHS["wealth_curves"] = save_fig(fig, "chart4_wealth_curves-ONL.png")

# ──────────────────────────────────────────────────────────────
# CHART 5 — Volatility split win rate
# ──────────────────────────────────────────────────────────────
vol_cats = ["Low", "Med", "High"]
vol_stats = {}
for vc in vol_cats:
    sub = completed[completed["Volatility Cond"] == vc]
    n   = len(sub)
    w   = (sub["Result"] == "Win").sum() if n > 0 else 0
    vol_stats[vc] = (w, n, w / n if n > 0 else 0)

x_pos      = np.arange(len(vol_cats))
wr_bar     = [vol_stats[v][2] for v in vol_cats]
n_bar      = [vol_stats[v][1] for v in vol_cats]
bar_colors = [GREEN if wr >= p else RED for wr in wr_bar]

fig, ax = plt.subplots(figsize=(7, 5))
bars = ax.bar(x_pos, wr_bar, color=bar_colors, alpha=0.85, width=0.5, edgecolor="white")
ax.axhline(p,      color=BLUE, linewidth=1.6, linestyle="--",
           label=f"Overall $p = {p:.4f}$")
ax.axhline(p_star, color=GREY, linewidth=1.2, linestyle=":",
           label=f"Breakeven $p^* = {p_star:.3f}$")
for bar, n_v, wr in zip(bars, n_bar, wr_bar):
    ax.text(bar.get_x() + bar.get_width() / 2,
            bar.get_height() + 0.005,
            f"{wr:.3f}\n$(n={n_v})$",
            ha="center", va="bottom", fontsize=9)
ax.set_xticks(x_pos)
ax.set_xticklabels(vol_cats, fontsize=10)
ax.set_ylim(0, max(wr_bar + [p]) + 0.10)
ax.yaxis.set_major_formatter(mticker.FuncFormatter(lambda x, _: f"{x:.2f}"))
```

```
style_ax(ax, "Win Rate by ATR Volatility Condition [ONL]",
        "Volatility condition (ATR percentile tercile)",
        "Win rate $p$")
ax.legend(fontsize=8.5, framealpha=0.95)
fig.tight_layout()
IMG_PATHS["vol_split"] = save_fig(fig, "chart5_vol_split-ONL.png")

print("Charts saved:", list(IMG_PATHS.values()))

# ═══════════════════════════════════════════════════════════════
# EXCEL STYLE CONSTANTS
# ═══════════════════════════════════════════════════════════════
FILL_HDR   = PatternFill("solid", fgColor="1B3A6B")
FILL_SUBHDR= PatternFill("solid", fgColor="2E5FA3")
FILL_WIN   = PatternFill("solid", fgColor="D5F5E3")
FILL_LOSS  = PatternFill("solid", fgColor="FADBD8")
FILL_UNRES = PatternFill("solid", fgColor="FEF9E7")
FILL_AMBIG = PatternFill("solid", fgColor="EBF5FB")
FILL_LIGHT = PatternFill("solid", fgColor="EAF2FB")
FILL_WHITE = PatternFill("solid", fgColor="FFFFFF")
FILL_YELLOW= PatternFill("solid", fgColor="FFF9C4")
FILL_PURPLE= PatternFill("solid", fgColor="F0E6FF")

FONT_HDR  = Font(name="Arial", bold=True, color="FFFFFF", size=10)
FONT_BODY = Font(name="Arial", size=9)
FONT_BOLD = Font(name="Arial", bold=True, size=9)
FONT_TITLE= Font(name="Arial", bold=True, size=13, color="1B3A6B")
FONT_SUB  = Font(name="Arial", bold=True, size=11, color="2E5FA3")
FONT_NOTE = Font(name="Arial", italic=True, size=9, color="555555")

ALIGN_C    = Alignment(horizontal="center", vertical="center")
ALIGN_L    = Alignment(horizontal="left",   vertical="center")
ALIGN_WRAP = Alignment(horizontal="left",   vertical="top", wrap_text=True)

def thin():
    s = Side(style="thin", color="CCCCCC")
    return Border(left=s, right=s, top=s, bottom=s)

def set_widths(ws, d):
    for col, w in d.items():
        ws.column_dimensions[get_column_letter(col)].width = w

def write_header(ws, row, texts, fill=FILL_HDR, font=FONT_HDR):
    for ci, txt in enumerate(texts, 1):
        c = ws.cell(row=row, column=ci, value=txt)
        c.fill = fill; c.font = font
        c.alignment = ALIGN_C; c.border = thin()

def cell_set(ws, row, col, val=None, fill=FILL_WHITE, font=FONT_BODY,
             align=ALIGN_C, fmt=None):
```

```
    c = ws.cell(row=row, column=col)
    if val is not None:
        c.value = val
    c.fill = fill; c.font = font; c.alignment = align; c.border = thin()
    if fmt:
        c.number_format = fmt
    return c

# ═══════════════════════════════════════════════════════════════
# TRADE ROW WRITER  (ONL columns)
# ═══════════════════════════════════════════════════════════════
TRADE_COLS = list(tdf.columns)
TRADE_WIDTHS = {
    1:9, 2:12, 3:7, 4:10, 5:14, 6:13, 7:13, 8:13,
    9:13, 10:12, 11:11, 12:15, 13:15, 14:15, 15:16
}

RESULT_FILL = {
    "Win":        FILL_WIN,
    "Loss":       FILL_LOSS,
    "Unresolved": FILL_UNRES,
    "Ambiguous":  FILL_AMBIG,
}

def write_trade_row(ws, row_num, t):
    result = t["Result"]
    fill   = RESULT_FILL.get(result, FILL_WHITE)
    col_fmts = {
        "Entry Price":    "#,##0.0000",
        "Stop Loss":      "#,##0.0000",
        "Take Profit":    "#,##0.0000",
        "Exit Price":     "#,##0.0000",
        "% Return":       "0.0000",
        "Account Before": '"$"#,##0.00',
        "Account After":  '"$"#,##0.00',
        "ATR at Entry":   "0.000000",
        "Random Value":   "0.000000",
    }
    for ci, col in enumerate(TRADE_COLS, 1):
        val = t[col]
        if not isinstance(val, str) and pd.isna(val):
            val = None
        c = ws.cell(row=row_num, column=ci, value=val)
        c.fill = fill; c.font = FONT_BODY
        c.alignment = ALIGN_C; c.border = thin()
        if col in col_fmts and val is not None:
            c.number_format = col_fmts[col]

def build_trade_sheet(ws, df_sub, title):
    ws.sheet_view.showGridLines = False
```

```
    ws.merge_cells(f"A1:{get_column_letter(len(TRADE_COLS))}1")
    c = ws["A1"]
    c.value = title; c.font = FONT_TITLE; c.alignment = ALIGN_C
    ws.row_dimensions[1].height = 28
    ws.merge_cells(f"A2:{get_column_letter(len(TRADE_COLS))}2")
    sub   = df_sub[df_sub["Result"].isin(["Win","Loss"])]
    n_w   = (sub["Result"] == "Win").sum()
    n_l   = (sub["Result"] == "Loss").sum()
    p_sub = n_w / (n_w + n_l) if (n_w + n_l) > 0 else 0
    ws["A2"].value = (f"n={len(df_sub)} trades  |  Wins={n_w}  Losses={n_l}  "
                      f"Unresolved={(df_sub['Result']=='Unresolved').sum()}  "
                      f"Ambiguous={(df_sub['Result']=='Ambiguous').sum()}  |  "
                      f"Completed p = {p_sub:.4f}  |  Seed=42  |  No Leverage [ONL]")
    ws["A2"].font = FONT_NOTE; ws["A2"].alignment = ALIGN_C
    write_header(ws, 3, TRADE_COLS)
    set_widths(ws, TRADE_WIDTHS)
    ws.freeze_panes = "A4"
    for i, (_, row) in enumerate(df_sub.iterrows(), 4):
        write_trade_row(ws, i, row)

# ─────────────────────────────────────────────────────────────────────
# BUILD WORKBOOK
# ─────────────────────────────────────────────────────────────────────
wb = openpyxl.Workbook()
wb.remove(wb.active)

# ═══════════════════════════════════════════════════════════════════
# SHEET 1 — No-Leverage Summary [ONL]
# ═══════════════════════════════════════════════════════════════════
ws1 = wb.create_sheet("No-Leverage Summary ONL")
ws1.sheet_view.showGridLines = False

ws1.merge_cells("A1:H1")
c = ws1["A1"]
c.value = "Kelly Criterion — No-Leverage Summary Statistics [ONL]"
c.font = FONT_TITLE; c.alignment = ALIGN_C
ws1.row_dimensions[1].height = 30

# ── Core Kelly Metrics ──────────────────────────────
metrics = [
    ("Total Trades Simulated",     N_TRADES,
     "200-bucket entry selection, seed 42"),
    ("Completed Trades (W + L)",   n_comp,
     "Used for p and R calculation"),
    ("Wins",                       n_wins,
     "Take-profit reached first"),
    ("Losses",                     n_losses,
     "Stop-loss reached first"),
    ("Unresolved",                 n_unres,
     "Neither barrier hit within 78-candle window — EXCLUDED"),
```

```
    ("Ambiguous",                n_ambig,
     "Both TP and SL touched in same OHLC candle — EXCLUDED"),
    ("Win Rate  p",              round(p, 6),
     "p = Wins / Completed trades"),
    ("Avg % Gain per Win",       round(avg_gain * 100, 4),
     "= 0.375 % by design (TP = entry × 1.00375)"),
    ("Avg % Loss per Loss",      round(avg_loss * 100, 4),
     "= 0.25 % by design (SL = entry × 0.9975)"),
    ("Reward-to-Risk Ratio  R",  round(R, 6),
     "R = avg gain / avg loss = 1.5 by construction"),
    ("Optimal Kelly Fraction  f*", round(f_star, 6),
     "f* = [p(R+1) − 1] / R"),
    ("Breakeven Win Rate  p*",   round(p_star, 6),
     "p* = 1/(1+R); f* > 0 requires p > p*"),
    ("G(f*)  Optimal Log-Growth", round(G_opt, 8),
     "Expected log-growth per trade at f* (Kelly maximum)"),
    ("Half-Kelly  f*/2",         round(f_star / 2, 6),
     "Conservative sizing (half of Kelly-optimal)"),
    ("Double-Kelly  2f*",        round(f_star * 2, 6),
     "Aggressive sizing (double Kelly-optimal)"),
    ("— ONL Actual Betting Fraction —", "──────────", "──────────"),
    ("f_actual  [ONL]",          round(F_ACTUAL, 6),
     "Fraction of account risked per trade = SL_PCT = 0.0025 (no leverage)"),
    ("G(f_actual)  [ONL]",       round(G_actual, 8),
     "Expected log-growth at ONL betting fraction — well below G(f*)"),
    ("Ratio f_actual / f*",      round(F_ACTUAL / f_star, 6),
     "ONL bets ≈ 3 % of Kelly-optimal → significant under-betting"),
    ("— Final Account Values —",  "──────────", "──────────"),
    ("Initial Account",          ACCOUNT,
     "Starting capital (USD, no leverage)"),
    ("Final Account  [ONL]",     round(account, 2),
     f"After {n_comp} resolved trades at f_actual = {F_ACTUAL}"),
    ("Final Account  Half-Kelly", round(float(wealth_series(f_star / 2)[-1]), 2),
     f"Simulated at f = {f_star/2:.4f}"),
    ("Final Account  Full-Kelly", round(float(wealth_series(f_star)[-1]), 2),
     f"Simulated at f* = {f_star:.4f} (theoretical optimal)"),
    ("Final Account  Double-Kelly",round(float(wealth_series(f_star * 2)[-1]), 2),
     f"Simulated at f = {f_star*2:.4f}"),
]

write_header(ws1, 3, ["Metric", "Value", "Description"])
set_widths(ws1, {1: 34, 2: 18, 3: 55})

for i, (met, val, desc) in enumerate(metrics, 4):
    is_divider = isinstance(val, str) and val == "──────────"
    bg = PatternFill("solid", fgColor="2E5FA3") if is_divider else (
        FILL_PURPLE if ("f_actual" in met or "ONL" in met.upper() or "Final Account" in met) else (
        FILL_LIGHT if i % 2 == 0 else FILL_WHITE))
    fnt = Font(name="Arial", bold=True, color="FFFFFF", size=9) if is_divider else (
         FONT_BOLD if ("──" not in str(val)) else FONT_BODY)
```

```
    for ci, (v, al, fmt) in enumerate([
        (met,  ALIGN_L, None),
        (val if not is_divider else "",  ALIGN_C,
         "0.000000" if isinstance(val, float) and not is_divider else (
         '"$"#,##0.00' if "Account" in met and isinstance(val, (int, float)) and not is_divider else None)),
        (desc if not is_divider else "",  ALIGN_L, None),
    ], 1):
        c = ws1.cell(row=i, column=ci, value=v)
        c.fill = bg; c.alignment = al; c.border = thin()
        c.font = fnt if ci == 1 else (Font(name="Arial", bold=True, color="FFFFFF", size=9) if is_divider
else FONT_BODY)
        if fmt and not is_divider: c.number_format = fmt

last_metrics_row = 3 + len(metrics)

# ── Sensitivity Table: f*(p) ──────────────────────────
r0 = last_metrics_row + 3
ws1.merge_cells(f"A{r0}:D{r0}")
c = ws1.cell(r0, 1, f"Sensitivity: f*(p) with R = {RR_DESIGN} (design value)")
c.font = FONT_SUB; c.alignment = ALIGN_L

write_header(ws1, r0 + 1, ["Win Rate p", "f* (Kelly Fraction)", "G(f*)", "Interpretation"])

p_sens = np.round(np.arange(0.30, 0.66, 0.01), 2)
for i, pv in enumerate(p_sens, r0 + 2):
    fv = (pv * (RR_DESIGN + 1) - 1) / RR_DESIGN
    gv = G(max(fv, 0.0), pv, RR_DESIGN)
    bg = FILL_YELLOW if abs(pv - round(p, 2)) < 0.006 else (
        FILL_LIGHT if i % 2 == 0 else FILL_WHITE)
    interp = ("No bet — no positive edge" if fv <= 0
              else "Minimal — bet cautiously" if fv < 0.08
              else "Conservative" if fv < 0.15
              else "Moderate" if fv < 0.25
              else "Aggressive")
    for ci, (v, fmt) in enumerate([(round(pv, 2), "0.00"),
                                   (round(fv, 6), "0.000000"),
                                   (round(gv, 8), "0.00000000"),
                                   (interp, None)], 1):
        c = ws1.cell(row=i, column=ci, value=v)
        c.fill = bg; c.font = FONT_BODY; c.alignment = ALIGN_C; c.border = thin()
        if fmt: c.number_format = fmt

last_sens = r0 + 1 + len(p_sens)

# ── G(f) Table ────────────────────────────────────────
r1 = last_sens + 3
ws1.merge_cells(f"A{r1}:D{r1}")
ws1.cell(r1, 1, f"G(f) Table  — p = {p:.4f},  R = {R:.4f}").font = FONT_SUB
ws1.cell(r1, 1).alignment = ALIGN_L
```

```
write_header(ws1, r1 + 1, ["Kelly Fraction f", "G(f)", "Relative to G(f*)", "Status"])

f_tbl = np.concatenate([np.arange(0.000, 0.010, 0.001),
                        np.arange(0.010, 0.10, 0.01),
                        np.arange(0.10, 0.32, 0.02)])
for i, fv in enumerate(f_tbl, r1 + 2):
    gv  = G(fv, p, R)
    rel = gv - G_opt
    is_actual = abs(fv - F_ACTUAL) < 0.0005
    is_opt    = abs(fv - f_practical) < 0.005 and f_star > 0
    bg = FILL_PURPLE if is_actual else (
         FILL_WIN    if is_opt else (
         FILL_LIGHT  if i % 2 == 0 else FILL_WHITE))
    status = ("ONL actual ← f_actual" if is_actual
              else "Optimal ✓" if is_opt
              else "Over-bet"  if fv > max(f_star, 0)
              else "Under-bet")
    for ci, (v, fmt) in enumerate([
        (round(fv, 4), "0.0000"),
        (round(gv, 8) if math.isfinite(gv) else "—", "0.00000000"),
        (round(rel, 8) if math.isfinite(rel) else "—", "0.00000000"),
        (status, None)
    ], 1):
        c = ws1.cell(row=i, column=ci, value=v)
        c.fill = bg; c.font = FONT_BODY; c.alignment = ALIGN_C; c.border = thin()
        if fmt: c.number_format = fmt

last_gf = r1 + 1 + len(f_tbl)

# ── Wealth comparison table ──────────────────────────
r2 = last_gf + 3
ws1.merge_cells(f"A{r2}:F{r2}")
ws1.cell(r2, 1, "Half / Full / Double Kelly  vs  ONL Actual Comparison").font = FONT_SUB
ws1.cell(r2, 1).alignment = ALIGN_L

write_header(ws1, r2 + 1, ["Strategy", "Fraction", "G(f)",
                           "Final Wealth ($)", "Max Drawdown", "Ruin?"])
set_widths(ws1, {5: 18, 6: 14})

for i, (name_s, frac, w_series) in enumerate([
    ("ONL Actual  [ONL]", F_ACTUAL,      wActual),
    ("Half Kelly",        f_star / 2,    wHalf),
    ("Full Kelly",        f_star,        wFull),
    ("Double Kelly",      f_star * 2,    wDouble),
], r2 + 2):
    gv    = G(frac, p, R)
    final = w_series[-1]
    peak  = np.maximum.accumulate(w_series)
    dd    = (w_series - peak) / np.where(peak > 0, peak, 1)
    max_dd = dd.min()
```

```
    ruin  = (w_series < 1.0).any()
    bg    = FILL_PURPLE if "ONL" in name_s else (FILL_LIGHT if i % 2 == 0 else FILL_WHITE)
    for ci, (v, fmt) in enumerate([
        (name_s,              None),
        (round(frac, 6),      "0.000000"),
        (round(gv, 8),        "0.00000000"),
        (round(final, 2),     '"$"#,##0.00'),
        (f"{max_dd*100:.2f}%", None),
        ("YES" if ruin else "No", None),
    ], 1):
        c = ws1.cell(row=i, column=ci, value=v)
        c.fill = bg; c.font = FONT_BODY; c.alignment = ALIGN_C; c.border = thin()
        if fmt: c.number_format = fmt

# ═══════════════════════════════════════════════════════════════════
# SHEET 2 — Trade Log IA Sample (30) [ONL]
# ═══════════════════════════════════════════════════════════════════
ws2 = wb.create_sheet("Trade Log – IA Sample (30) ONL")
build_trade_sheet(ws2, tdf.head(30),
                  "Kelly Criterion Trade Log — IA Sample (First 30 Trades) [No Leverage — ONL]")

# ═══════════════════════════════════════════════════════════════════
# SHEET 3 — Full Dataset [ONL]
# ═══════════════════════════════════════════════════════════════════
ws3 = wb.create_sheet(f"Full Dataset ({N_TRADES}) ONL")
build_trade_sheet(ws3, tdf,
                  f"Kelly Criterion Trade Log — Full Simulation ({N_TRADES} Trades) [No Leverage —
ONL]")

# ═══════════════════════════════════════════════════════════════════
# SHEET 4 — Charts [ONL]
# ═══════════════════════════════════════════════════════════════════
ws4 = wb.create_sheet("Charts ONL")
ws4.sheet_view.showGridLines = False
ws4.column_dimensions["A"].width = 2

chart_meta = [
    ("gf_curve",
     "Chart 1 — G(f): Kelly Growth Function [ONL]",
     f"Shows expected log-growth per trade vs Kelly fraction f (p={p:.4f}, R={R:.4f}). "
     f"Two markers: orange dashed line = f* = {f_star:.4f} (Kelly optimal maximum); "
     f"purple dotted line = f_actual = {F_ACTUAL:.4f} (ONL no-leverage actual). "
     f"ONL bets far below Kelly-optimal, giving G(f_actual) = {G_actual:.6f} vs G(f*) = {G_opt:.5f}."),
    ("fp_curve",
     f"Chart 2 — f*(p): Optimal Fraction vs Win Probability  (R = {R:.4f} fixed) [ONL]",
     "With R held at the design value 1.5, f* rises linearly with p. "
     "Empirical p is marked; any p below the breakeven p* gives a negative f*."),
    ("fr_curve",
     f"Chart 3 — f*(R): Optimal Fraction vs Reward-to-Risk  (p = {p:.4f} fixed) [ONL]",
     "With p held at the empirical value, f* rises with R. "
```

```
     "Shows the minimum R required for a positive Kelly fraction."),
    ("wealth_curves",
     "Chart 4 — Wealth Curves: ONL Actual vs Kelly Fractions [ONL]",
     f"Simulated account value over all {n_comp} completed trades in sequence. "
     f"Purple = ONL actual (f_actual = {F_ACTUAL:.4f}); "
     f"green = Half Kelly (f = {f_star/2:.4f}); "
     f"blue = Full Kelly (f* = {f_star:.4f}); red dashed = Double Kelly (f = {f_star*2:.4f}). "
     "Illustrates the large gap between ONL actual and Kelly-optimal growth."),
    ("vol_split",
     "Chart 5 — Win Rate by ATR Volatility Condition [ONL]",
     "Bars show empirical win rate within each ATR percentile bucket "
     "(Low ≤ 33rd, Med ≤ 67th, High > 67th). "
     "Overall p and breakeven p* lines added for reference."),
]

cur_row = 2
for key, c_title, c_desc in chart_meta:
    ws4.merge_cells(f"B{cur_row}:L{cur_row}")
    ct = ws4.cell(cur_row, 2, c_title)
    ct.fill = FILL_SUBHDR; ct.font = FONT_HDR; ct.alignment = ALIGN_L
    ws4.row_dimensions[cur_row].height = 22
    cur_row += 1

    ws4.merge_cells(f"B{cur_row}:L{cur_row}")
    cd = ws4.cell(cur_row, 2, c_desc)
    cd.font = FONT_NOTE; cd.alignment = ALIGN_WRAP
    ws4.row_dimensions[cur_row].height = 40
    cur_row += 1

    img = XLImage(IMG_PATHS[key])
    img.width = 680; img.height = 425
    ws4.add_image(img, f"B{cur_row}")
    for r in range(cur_row, cur_row + 33):
        ws4.row_dimensions[r].height = 14
    cur_row += 34

# ═══════════════════════════════════════════════════════════════
# SHEET 5 — Leverage Extension Summary [LLL]
# ═══════════════════════════════════════════════════════════════
ws5 = wb.create_sheet("Leverage Extension Summary LLL")
ws5.sheet_view.showGridLines = False
set_widths(ws5, {1: 3, 2: 110})

ws5.merge_cells("B1:B1")
c = ws5.cell(1, 2, "Leverage Extension — Key Results [LLL]  (see separate file: 2026-05-14-kelly-
criterion-dataset-LLL.xlsx)")
c.font = FONT_TITLE; c.alignment = ALIGN_L
ws5.row_dimensions[1].height = 30

# Key LLL results table
```

```
r = 3
ws5.merge_cells(f'B{r}:E{r}')
ws5.cell(r, 2, "LLL Simulation — Key Results (retained as leverage reference only)").font =
FONT_SUB
ws5.cell(r, 2).alignment = ALIGN_L
r += 1

write_header(ws5, r, ["", "Metric", "LLL Value", "Notes"], fill=FILL_SUBHDR)
set_widths(ws5, {2: 3, 3: 36, 4: 22, 5: 60})
r += 1

lll_data = [
    ("Total trades",              "183",
     "17 of 200 buckets blocked by 5 unresolved trades"),
    ("Wins / Losses / Unresolved","80 / 98 / 5",
     "Ambiguous = 0"),
    ("Win rate  p",               "0.449438",
     "Same as ONL — identical entry selection and outcomes"),
    ("Reward-to-risk  R",         "1.500000",
     "R = TP% / SL% = 0.375 / 0.25 = 1.5 by construction"),
    ("Optimal Kelly f*",          "0.082397",
     "f* = [p(R+1)−1]/R = [0.449×2.5−1]/1.5"),
    ("G(f*)",                     "0.00503169",
     "Expected log-growth at Kelly-optimal fraction"),
    ("Account size",              "$100,000",
     "Fixed starting capital"),
    ("Risk per trade (LLL)",      "$1,000  (1 % of account)",
     "Dollar risk = ACCOUNT × RISK_PCT = 100,000 × 0.01"),
    ("Implied position size (LLL)","≈ $400,000  (4× notional)",
     "shares = $1,000 / (entry × 0.0025) → e.g. 588 shares at $680 = $400k"),
    ("Implied leverage (LLL)",    "4:1",
     "0.25 % SL on a $100k account requires $400k position to lose $1k"),
    ("f_actual in Kelly terms (LLL)", "0.01",
     "Kelly fraction = risk per trade / account = 0.01 (1 %)"),
    ("Kelly multiple (LLL)",      f"≈ {round(0.01 / f_star, 2)}×  (over-betting)",
     f"f_actual_LLL / f* = 0.01 / {f_star:.4f}  — betting more than Kelly optimal"),
]

for i, (metric, val, note) in enumerate(lll_data, r):
    bg = FILL_LIGHT if i % 2 == 0 else FILL_WHITE
    for ci, (v, al) in enumerate([("", ALIGN_C), (metric, ALIGN_L),
                                  (val, ALIGN_C), (note, ALIGN_L)], 2):
        c = ws5.cell(row=i, column=ci, value=v)
        c.fill = bg; c.alignment = al; c.border = thin()
        c.font = FONT_BOLD if ci == 3 else FONT_BODY

r += len(lll_data) + 2

# Methodology explanation — leverage problem
sections_lll = [
```

```
    ("Why LLL Uses Leverage",
     "In the LLL simulation each trade risks a fixed $1,000 (1 % of the $100,000 account). "
     "With a stop-loss of only 0.25 % of the entry price, the required position size is:\n\n"
     "    position = $1,000 / (entry_price × 0.0025)\n\n"
     "At a typical SPY price of ~$680 this produces ≈ 588 shares × $680 = $400,000 notional "
     "— four times the account value. This is 4:1 leverage: $100,000 of equity controls "
     "$400,000 of assets. The simulation is valid for analysing the Kelly Criterion mathematically "
     "(win/loss statistics, G(f), sensitivity tables) but would require margin lending in practice. "
     "It is retained as a 'Leverage Extension' — the figures are internally consistent but reflect "
     "a leveraged implementation, not a no-leverage deployment."),

    ("Why ONL Is the Primary Model",
     "The ONL (no-leverage) simulation removes all borrowed capital. Each trade risks exactly "
     "f_actual = 0.0025 (0.25 %) of the current account equity:\n\n"
     "    Win  →  account × (1 + 0.00375)   [+0.375 % of equity]\n"
     "    Loss →  account × (1 − 0.0025)    [−0.25 % of equity]\n\n"
     "At an account of $100,000 this is a $250 maximum loss per trade — no margin, no borrowed "
     "capital, no notional exposure beyond what the account actually holds. This is the correct "
     "framework for evaluating the Kelly Criterion as a purely mathematical money-management rule: "
     "f_actual represents the fraction of the bankroll wagered, which is the quantity Kelly's "
     "formula directly optimises."),

    ("LLL vs ONL — Key Difference",
     "The trade entry candles, directions, and outcomes (Win/Loss/Unresolved) are IDENTICAL in "
     "both simulations. The two models share the same random seed (42) and the same 183 trades. "
     "The only difference is the P&L mechanism:\n\n"
     "  LLL: position_size = $1,000 / (entry × SL_PCT) → fixed dollar risk, variable leverage\n"
     "  ONL: account_after = account_before × (1 ± SL/TP pct) → fixed fraction risk, no leverage\n\n"
     "As a result, ONL produces a slow compound growth curve (f_actual << f*) while LLL "
     "produces much higher nominal returns but only because it is implicitly leveraged at 4:1. "
     "For the IB Maths IA, ONL is the primary data source. LLL is referenced only in the "
     "Leverage Extension to show what happens when the same mathematical strategy is implemented "
     "with borrowed capital."),
]

for heading, body in sections_lll:
    ws5.row_dimensions[r].height = 22
    c = ws5.cell(r, 2, heading)
    c.font = Font(name="Arial", bold=True, size=11, color="1B3A6B")
    c.alignment = ALIGN_L
    r += 1

    n_lines = body.count("\n") + 1
    row_h   = max(15 * n_lines, 50)
    ws5.row_dimensions[r].height = row_h
    c = ws5.cell(r, 2, body)
    c.font = Font(name="Arial", size=10)
    c.alignment = ALIGN_WRAP
    c.fill = FILL_LIGHT if (sections_lll.index((heading, body)) % 2 == 0) else FILL_WHITE
    r += 2
```

```
# ═══════════════════════════════════════════════════════════════
# METHODOLOGY (implicit in sheet 5 above for ONL)
# Adding a dedicated Methodology sheet
# ═══════════════════════════════════════════════════════════════
ws6 = wb.create_sheet("Methodology ONL")
ws6.sheet_view.showGridLines = False
set_widths(ws6, {1: 3, 2: 115})

ws6.merge_cells("B1:B1")
c = ws6.cell(1, 2, "Methodology Note — Kelly Criterion No-Leverage Simulation [ONL]")
c.font = FONT_TITLE; c.alignment = ALIGN_L
ws6.row_dimensions[1].height = 30

sections = [
   ("1.  Data Source",
    "All trades were simulated on historical SPY (SPDR S&P 500 ETF Trust) 5-minute OHLC "
    "candle data spanning 2026-02-19 to 2026-05-14 — 60 US trading days, 4,663 candles, "
    "covering regular market hours (09:30–16:00 ET) only. No real money was used at any stage."),

   ("2.  Entry Selection (200-bucket method, non-overlapping)",
    "The dataset (4,663 candles) was divided into 200 equal-width buckets, each covering "
    "approximately 23 consecutive candles. One entry candle was drawn uniformly at random "
    "(NumPy default_rng, seed 42) from within each bucket. This produces ≈ 3–4 trades per "
    "trading day automatically.\n"
    "Non-overlap was enforced globally: if a bucket's drawn candle index fell at or before "
    "the resolution index of the previous trade, the entry was advanced to the first candle "
    "after that resolution index. This prevents two positions being open simultaneously."),

   ("3.  Trade Parameters",
    "Entry price = closing price of the selected candle.\n"
    "Direction = LONG if a second uniform random draw < 0.5, else SHORT.\n"
    "Stop-loss (SL): 0.25 % from entry (below for LONG, above for SHORT).\n"
    "Take-profit (TP): 0.375 % from entry (above for LONG, below for SHORT).\n"
    "Reward-to-risk ratio R = TP % ÷ SL % = 0.375 % ÷ 0.25 % = 1.5 exactly by construction.\n\n"
    "No-Leverage P&L [ONL]:\n"
    "  Win  →  account_after = account_before × (1 + TP_PCT) = account_before × 1.00375\n"
    "  Loss →  account_after = account_before × (1 − SL_PCT) = account_before × 0.9975\n"
    "  In Kelly terms this is f_actual = SL_PCT = 0.0025 (the fraction of the account lost "
    "  on each losing trade). No borrowed capital or margin is involved."),

   ("4.  Trade Resolution Logic",
    "Starting from the candle immediately after entry, subsequent OHLC bars were scanned "
    "forward in time. For each bar, the following priority rules applied:\n"
    "  • Ambiguous: TP and SL both breached within the same OHLC candle → EXCLUDED\n"
    "  • Win: TP reached first → exit at exact TP level\n"
    "  • Loss: SL reached first → exit at exact SL level\n"
    "  • Unresolved: neither barrier breached within 78 consecutive 5-minute bars → EXCLUDED\n"
    "Only Win and Loss trades enter the p and R calculations."),
```

```
    ("5.  Kelly Criterion Framework",
     "  p  = (number of Wins) ÷ (number of completed Win+Loss trades)\n"
     "  R  = (mean % return on Wins) ÷ (mean % loss on Losses) = 1.5 by construction\n"
     "  f* = [p(R + 1) − 1] / R          (Kelly criterion formula)\n"
     "  p* = 1 / (1 + R)                 (breakeven win rate)\n"
     "  G(f) = p · ln(1 + fR) + (1−p) · ln(1 − f)   (expected log-growth per trade)\n\n"
     f"  Empirical results: p = {p:.4f} > p* = {p_star:.4f} → f* = {f_star:.4f} > 0 (positive edge)\n"
     f"  ONL bets at f_actual = {F_ACTUAL:.4f} (≈ {F_ACTUAL/f_star*100:.1f} % of f*) →
G(f_actual) = {G_actual:.6f}\n"
     f"  Kelly-optimal G(f*) = {G_opt:.5f} → ONL under-bets significantly relative to the theoretical
optimum.\n"
     "  This is intentional: f_actual = 0.0025 corresponds to risking only one's own capital "
     "  at a conservative fraction, demonstrating where ONL sits on the G(f) curve."),

    ("6.  Leverage Extension (LLL) Reference",
     "The LLL simulation (file: 2026-05-14-kelly-criterion-dataset-LLL.xlsx) uses a fixed "
     "$1,000 dollar risk per trade. With SL = 0.25 % of price, the required position size is "
     "$1,000 / (entry × 0.0025) ≈ $400,000 at typical SPY prices — 4× the $100,000 account. "
     "This is 4:1 leverage. LLL is retained only as a leverage extension to illustrate the "
     "impact of borrowed capital on strategy returns. The IB Maths IA primary analysis uses "
     "ONL, which is leverage-free by construction."),

    ("7.  Reproducibility",
     "The entire simulation was executed with NumPy's default_rng(seed=42), ensuring "
     "identical results on every run. Python 3, NumPy, pandas, matplotlib, and openpyxl "
     "were used. No real brokerage, API, or live data feed was involved."),
]

cur = 3
for heading, body in sections:
    ws6.row_dimensions[cur].height = 22
    c = ws6.cell(cur, 2, heading)
    c.font = Font(name="Arial", bold=True, size=11, color="1B3A6B")
    c.alignment = ALIGN_L
    cur += 1

    n_lines = body.count("\n") + 1
    row_h   = max(15 * n_lines, 40)
    ws6.row_dimensions[cur].height = row_h
    c = ws6.cell(cur, 2, body)
    c.font = Font(name="Arial", size=10)
    c.alignment = ALIGN_WRAP
    c.fill = FILL_LIGHT if (sections.index((heading, body)) % 2 == 0) else FILL_WHITE
    cur += 2

# ──────────────────────────────────────────────────────────────────────
# SAVE
# ──────────────────────────────────────────────────────────────────────
wb.save(XLFILE)
print(f"\n✓ Workbook saved : {XLFILE}")
```

```
print(f"✓ Charts in      : {IMGDIR}")
print(f"✓ Total trades   : {N_TRADES}")
print(f"  Wins={n_wins}  Losses={n_losses}  "
      f"Unresolved={n_unres}  Ambiguous={n_ambig}")
print(f"  f_actual={F_ACTUAL}  f*={f_star:.6f}")
print(f"  G(f_actual)={G_actual:.8f}  G(f*)={G_opt:.8f}")
print(f"  Final account: ${account:,.2f}")
```

(7) Raw SPY candles dataset(first 35 candle sample, as full data is 170 pages)

date	time	open	high	low	close	volume	resolution
2026-02-19	09:30	683.84	684.52	683.56	683.95	2975274	5
2026-02-19	09:35	683.95	684.02	682.47	683.62	1566088	5
2026-02-19	09:40	683.64	684.75	682.96	684.61	1535366	5
2026-02-19	09:45	684.62	684.72	683.25	684.09	1252700	5
2026-02-19	09:50	684.06	685.29	684.04	684.88	1105015	5
2026-02-19	09:55	684.86	685.82	684.7501	685.01	1375513	5
2026-02-19	10:00	685.0	686.18	684.36	685.51	1286639	5
2026-02-19	10:05	685.51	685.618	684.28	684.395	1045326	5
2026-02-19	10:10	684.44	684.87	683.98	684.15	631468	5
2026-02-19	10:15	684.17	685.3	684.12	684.8	895733	5
2026-02-19	10:20	684.8	685.32	684.18	685.27	783478	5
2026-02-19	10:25	685.275	685.275	683.81	683.9	841743	5
2026-02-19	10:30	683.89	685.73	683.71	685.7007	750377	5
2026-02-19	10:35	685.71	685.86	685.22	685.48	557121	5
2026-02-19	10:40	685.46	685.86	685.295	685.52	371162	5
2026-02-19	10:45	685.53	685.8099	685.16	685.495	426586	5
2026-02-19	10:50	685.51	685.51	684.23	684.35	686453	5
2026-02-19	10:55	684.34	685.48	684.33	685.32	467769	5
2026-02-19	11:00	685.32	685.6	684.5501	684.64	332192	5
2026-02-19	11:05	684.66	684.78	683.9	684.66	580601	5

2026-02-19	11:10	684.72	684.98	684.29	684.905	308767	5
2026-02-19	11:15	684.91	685.76	684.84	685.61	385278	5
2026-02-19	11:20	685.61	685.83	685.33	685.53	443423	5
2026-02-19	11:25	685.54	685.65	684.11	684.145	604880	5
2026-02-19	11:30	684.15	685.27	684.15	685.2	552817	5
2026-02-19	11:35	685.2	685.22	684.42	684.63	382748	5
2026-02-19	11:40	684.64	684.74	683.56	683.63	457752	5
2026-02-19	11:45	683.63	684.03	683.01	683.24	910835	5
2026-02-19	11:50	683.25	684.12	682.72	684.04	780002	5
2026-02-19	11:55	684.06	684.09	683.42	683.52	556175	5
2026-02-19	12:00	683.525	683.705	682.83	683.01	533260	5
2026-02-19	12:05	683.01	683.55	682.7	683.04	626896	5
2026-02-19	12:10	683.03	683.65	682.94	683.55	329572	5
2026-02-19	12:15	683.55	684.23	683.25	684.135	481123	5

(G)

Deviations: (1) a coding error first restricted replay matching to in-session signals and was corrected; (2) the replication (Table 7) and Table 11 were added afterwards to explain results, and no rule was chosen from them; (3) the session-open subgroup (Section 14) is exploratory; (4) the degenerate sizing rule is reported as registered.

(H)

Raw OANDA ledgers and logs were dumped with `a0_dump_raw.py` (account numbers removed; SHA-256 hashes in `MANIFEST.json`), trades rebuilt with `a1_build_trades.py` and candles fetched with `a2_fetch_candles.py`. Experiments: `a_backtest.py`, `a3_original_replication.py`, `a3b_bt10y_daily.py` (A); `b0_verify_signals.py`, `b_replay.py` (B); `c_volatility.py` (C); `d_kelly.py`, `d2_kelly_biased.py`, `p2_leverage.py` (D). Figures: `p2_figures.py`; this document: `p2_build_part2.py`.

(I) Use of Generative AI

Generative AI tools, primarily **Claude and Claude Code (Anthropic), including the models Opus 5.5 and 4.8,** and **ChatGPT (OpenAI)**, including the model GPT-6, were used during the development of Blackbox and the preparation of this investigation. Their use extended beyond proofreading and is described below in accordance with the Swiss Youth in Science Contribution Declaration (10 June 2026).

Trading system development. Blackbox's software was largely written by Claude Code under my direction., implementing, debugging and modifying Blackbox's trading software, including its trading signals, execution logic, risk management and backtesting infrastructure. I directed the project's development, made decisions concerning the strategy and its deployment, and monitored its forward-testing performance.

Data processing and experimental analysis. Claude Code assisted with the extraction and reconstruction of OANDA transaction records, the implementation of corrected backtests, the comparison of simulated and observed trades, the volatility-regime experiments, and the Kelly-optimisation calculations. AI-generated scripts were also used to produce tables and figures. The results reported in Part II were obtained from these computational analyses, rather than generated independently by an AI language model.

Research design and mathematical assistance. I used Claude and ChatGPT to discuss research questions, evaluate methodological choices, clarify statistical and mathematical concepts, and develop explanations connecting the Blackbox experiments to the Kelly Criterion investigated in Part I. AI assistance also supported the identification and investigation of potential backtesting errors, including look-ahead bias.

Writing and presentation. ChatGPT was used to review explanations suggest improvements to the report's structure. I subsequently reviewed and edited the material for the final submission.

Responsibility and limitations. I take responsibility for the research question, the writing, the decisions attributed to me, the interpretation of the findings and the final submitted report. AI assistance does not constitute independent validation of the analysis. The limitations of the available data, the backtesting implementations and the experimental design are discussed in the report.

AI-generated or AI-assisted material has therefore been acknowledged rather than presented as entirely unaided work. The computational methods and relevant scripts are identified in Appendix H.