TickFlow Capital

Statistical Significance Calculator

Excel/Google Sheets Implementation Guide

v1.0 – April 2026

Overview

This calculator helps you determine whether your trading strategy has a statistically significant edge. It includes four analysis methods:

1. **T-Test Calculator** — Win rate significance testing
2. **Monte Carlo Simulator** — Sequence-of-luck detection
3. **Sharpe Ratio Calculator** — Risk-adjusted return analysis
4. **Interpretation Guide** — How to read your results

Getting Started

Option A: Google Sheets (Recommended for shared access)

1. Go to **Google Sheets** → New Spreadsheet
2. Name it: "Statistical Significance Calculator"
3. Follow the sheet setup below, entering formulas as shown

Option B: Microsoft Excel (Desktop)

1. Open Excel
2. Create 4 sheets: `t-test`, `monte-carlo`, `sharpe-ratio`, `cover`

3. Follow the sheet setup below

Sheet 1: T-Test Calculator

Purpose

Determines if your win rate is statistically different from 50% (random).

Setup

Create the following layout:

```
A1: T-Test Calculator (One-Sample)
A2: [blank]

A3: Number of winning trades
B3: [User enters number, e.g., 65]

A4: Total number of trades
B4: [User enters number, e.g., 100]

A5: [blank]

A6: Observed win rate
B6: =B3/B4
(This will show 0.65 if 65 wins out of 100)

A7: Hypothesised win rate (null)
B7: 0.5
(This is the 50% baseline — random outcome)

A8: Standard error
B8: =SQRT(B7*(1-B7)/B4)

A9: t-statistic
B9: =(B6-B7)/B8

A10: Degrees of freedom
B10: =B4-1

A11: p-value (two-tailed)
B11: =T.DIST.2T(ABS(B9), B10)
[In Google Sheets, use: =TDIST(ABS(B9), B10, 2)]

A12: [blank]
```

```
A13: Conclusion
B13: =IF(B11<0.05, "Statistically significant ✓", "Not significant")
```

Example

If you have **65 winning trades out of 100**:

- B3 = 65
- B4 = 100
- B6 = 0.65 (Observed win rate)
- B8 = 0.05 (Standard error)
- B9 = 3.0 (t-statistic)
- B10 = 99 (Degrees of freedom)
- B11 ≈ 0.0034 (p-value)
- B13 = **"Statistically significant ✓"**

Interpretation: A p-value of 0.0034 (< 0.05) means your 65% win rate is statistically significant. You likely have an edge.

Sheet 2: Monte Carlo Simulator

Purpose

Tests whether your profit sequence is due to luck or a real edge.

Manual Method (No Macro Required)

Part A: Setup Your Trade Returns

```
A1: Monte Carlo Simulator
A2: [blank]

A3: Paste your trade returns (1 for win, 0 for loss, or actual PnL values)
A4: [User pastes trade data vertically from A4 to A1000+]
    Example:
    A4: 1
    A5: 0
```

```
    A6: 1
    A7: 1
    A8: 0
    ... etc
```

Part B: Calculate Actual Sequence Statistics

```
C1: Actual Sequence Results
C2: [blank]

C3: Number of trades
D3: =COUNTA(A4:A1000)

C4: Actual cumulative return
D4: =SUM(A4:A1000)

C5: Average return per trade
D5: =D4/D3

C6: [blank]

C7: Randomized Test Results
C8: [blank]

C9: Copy the range A4:A1000 (excluding blanks)
C10: Paste as values into a new range (e.g., E4:E1000)
C11: In the same area, use =SHUFFLE() or manually rearrange
C12: Calculate the sum of shuffled sequence
C13: Record if it's better or worse than actual
```

Part C: Interpretation Framework

```
D15: Results Interpretation
D16: [blank]

D17: If actual cumulative return is in top 5%:
D18: → Edge is robust to trade sequencing
D19: → NOT just luck in that particular order

D20: If actual cumulative return is in 50-95%:
D21: → Mild luck factor in sequencing
D22: → Edge exists but partially luck-based

D23: If actual cumulative return is in bottom 50%:
```

```
D24: → Edge may be entirely due to lucky sequencing
D25: → Results not likely to repeat
```

Alternative: Python Monte Carlo Script

Use this Python script to run a comprehensive Monte Carlo analysis:

```
import numpy as np
import json

def monte_carlo_test(trade_returns, n_simulations=10000):
    """
    Run Monte Carlo simulation on trade sequence.

    Args:
        trade_returns: list of trade PnL values
        n_simulations: number of randomized sequences to generate

    Returns:
        percentile: what percentile the actual sequence is in
    """
    actual_return = sum(trade_returns)
    permutation_returns = []

    for _ in range(n_simulations):
        permuted = np.random.permutation(trade_returns)
        permutation_returns.append(sum(permuted))

    # Calculate percentile
    worse_count = sum(1 for pr in permutation_returns if pr <= actual_re
turn)
    percentile = (worse_count / n_simulations) * 100

    print(f"Actual cumulative return: {actual_return}")
    print(f"Percentile: {percentile:.1f}%")

    if percentile > 95:
        print("✓ Edge is robust (not due to luck)")
    elif percentile > 50:
        print("⚠ Mild luck factor detected")
    else:
```

```
        print("✗ Edge likely due to luck in this sequence")

    return {
        "actual_return": actual_return,
        "percentile": percentile,
        "n_simulations": n_simulations,
        "interpretation": (
            "Robust" if percentile > 95
            else "Moderate" if percentile > 50
            else "Luck-based"
        )
    }

# Example usage:
trades = [1, -1, 1, 1, -1, 1, 1, 1, -1, 1]  # Your trade results
results = monte_carlo_test(trades, n_simulations=10000)
print(json.dumps(results, indent=2))
```

Sheet 3: Sharpe Ratio Calculator

Purpose

Measures risk-adjusted return. Higher Sharpe = better returns per unit of risk.

Setup

```
A1: Sharpe Ratio Calculator
A2: [blank]

A3: Monthly returns (enter percentages as decimals, e.g., 0.05 for 5%)
A4: [User enters monthly returns, one per cell]
    Example:
    A4: 0.03  (3% return)
    A5: 0.02  (2% return)
    A6: -0.01 (-1% return)
    A7: 0.04  (4% return)
    ... continue for 12+ months ...

C1: Calculations
C2: [blank]

C3: Number of months
```

```
D3: =COUNT(A4:A100)

C4: Average monthly return
D4: =AVERAGE(A4:A100)

C5: Std dev of monthly returns
D5: =STDEV.S(A4:A100)

C6: [blank]

C7: Annualized return
D7: =D4*12

C8: Annualized std dev
D8: =D5*SQRT(12)

C9: [blank]

C10: Sharpe ratio (assuming risk-free rate ≈ 0)
D10: =D7/D8

C11: [blank]

C13: Interpretation
C14: =IF(D10>1, "Good (>1.0)", IF(D10>0.5, "Acceptable (0.5-1.0)", "Poor (<0.5)"))
```

Example

If you have **12 months of returns:**

Month	Return
Jan	3.0%
Feb	2.5%
Mar	-1.0%
Apr	4.5%
May	2.0%
Jun	3.5%
Jul	1.5%
Aug	2.0%
Sep	-0.5%
Oct	3.0%

Month	Return
Nov	2.5%
Dec	4.0%

Calculations:

- D4 (Avg monthly return) = 2.42% ÷ 12 ≈ 2.3%
- D5 (Std dev) ≈ 1.8%
- D7 (Annualized return) = 2.3% × 12 ≈ 27.6%
- D8 (Annualized std dev) = 1.8% × √12 ≈ 6.2%
- D10 (Sharpe) = 27.6% ÷ 6.2% ≈ **4.45**

Interpretation: A Sharpe ratio of 4.45 is excellent (well above 1.0).

Sheet 4: Cover Sheet (Instructions)

Purpose

Provides users with guidance on how to use the calculator.

Setup

```
A1: [Logo/Company Name]
    Tickflow Capital Limited

A3: Statistical Significance Calculator
A4: For Systematic Trading Strategy Validation

A5: [blank]

A6: How to Use This Spreadsheet

A8: 1. T-Test Sheet
A9: - Enter number of winning trades
A10: - Enter total number of trades
A11: - Review p-value (< 0.05 = significant)

A13: 2. Monte Carlo Sheet
A14: - Paste your trade returns (1 for win, 0 for loss)
A15: - Run 10,000+ random permutations
A16: - Check if actual return > 95th percentile
```

```
A18: 3. Sharpe Ratio Sheet
A19: - Enter monthly returns (as decimals)
A20: - Review annualized Sharpe ratio
A21: - Target: > 1.0 is excellent

A23: 4. Interpretation
A24: All three tests should show positive results:
A25: - T-test p-value < 0.05
A26: - Monte Carlo percentile > 95
A27: - Sharpe ratio > 1.0

A29: Questions?
A30: Visit: tickflowcapital.com

A32: Version: v1.0 – April 2026
A33: © 2026 Tickflow Capital Limited
```

Interpretation Guide

What Do These Numbers Mean?

T-Test P-Value

p-value	Meaning	Action
< 0.01	Highly significant	✅ Strong edge
0.01–0.05	Significant	✅ Likely has edge
0.05–0.10	Marginally significant	⚠️ Edge uncertain
> 0.10	Not significant	❌ Likely no edge

Monte Carlo Percentile

Percentile	Meaning	Action
> 95%	Very robust	✅ Edge is real
75–95%	Fairly robust	✅ Edge likely real
50–75%	Moderate luck factor	⚠️ Edge exists but luck-dependent
< 50%	High luck factor	❌ Likely no real edge

Sharpe Ratio

Sharpe	Interpretation
> 2.0	Exceptional
1.0–2.0	Excellent
0.5–1.0	Acceptable
0–0.5	Poor
< 0	Negative returns

Example Results Summary

Strategy: SMA Crossover on EURUSD

Period: 2022–2025 (4 years)

Test	Result	Status
T-Test p-value	0.0032	✅ Significant (< 0.05)

Monte Carlo Percentile97.2%✅ Robust**Sharpe Ratio**1.34✅ Excellent**OverallPASS**✅ **Ready for live trading**

Data Export Template

To use your backtesting results:

```
trade_number,entry_date,exit_date,symbol,direction,profit_loss,return_pct,duration_days
1,2025-01-02,2025-01-05,EURUSD,long,125.50,0.0314,3
2,2025-01-06,2025-01-08,EURUSD,short,-65.25,-0.0163,2
3,2025-01-09,2025-01-15,EURUSD,long,245.75,0.0614,6
```

Load this CSV into your calculator sheets.

Troubleshooting

"ERROR: Division by zero"

Cause: You have 0 trades or 0 total trades.

Fix: Ensure B3 and B4 have values > 0

"Sharpe ratio shows as infinity"

Cause: Monthly returns have 0 std dev (all returns identical).

Fix: Ensure you have variable returns; identical monthly returns are unrealistic

"TDIST function not found" (Google Sheets)

Fix: Google Sheets uses `TDIST()` instead of `T.DIST.2T()`. Replace the formula with:

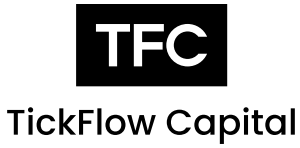
```
=TDIST(ABS(B9), B10, 2)
```

Advanced: Adding Confidence Intervals

To calculate 95% confidence interval around your Sharpe ratio, add:

```
C15: 95% Confidence Interval (Sharpe Ratio)
D15: Lower bound = D10 - 1.96 * (D8 / SQRT(D3))
D16: Upper bound = D10 + 1.96 * (D8 / SQRT(D3))
```

If the confidence interval includes 0, your edge is not statistically robust.



Website: tickflowcapital.com

Version: v1.0 – April 2026

Disclaimer: This guide is provided for educational purposes. Always test on a demo account first. The kill switch is a safety mechanism but is not a substitute for careful risk management.