Reversion-Based Arrival Price Trading: The Impact of Non-Linear Volume Participation Scaling

Spencer Butler
April 2025

Abstract

This research investigates the differences between linear and non-linear volume participation scaling when executing US equity orders benchmarked to arrival price. The study focuses on a reversion-based execution algorithm where the participation rate dynamically adjusts based on price movements, using standard deviations from the arrival price as a signal.

To isolate the effects of participation scaling, it is assumed that the trader can accurately adjust participation rates to the thousandth on a minute-by-minute rolling basis. It is also assumed that the trader can execute at the volume-weighted average price in rolling one-minute buckets. This removes uncertainties related to volume forecasting and allows direct comparison of execution outcomes based on different participation rate functions.

The study compares fixed participation, clamped linear scaling, and non-linear scaling – specifically a four-parameter logistic scaling function. Performance is measured by a paired-samples t-test. The backtesting framework uses historical volatility, price, and volume data to simulate order execution under the different strategies.

The findings aim to provide insight into optimal execution strategies for traders on algorithmic execution desks, particularly those running mean-reverting strategies. The accompanying Jupyter Notebook file is available on GitHub[1]. This paper does not constitute investment advice.

[1] https://github.com/spencerbutler888/dynamic-scaling

1 Introduction

In US equities, arrival price trading is a benchmarking concept where trade performance is measured by the difference between final average execution price and the last price at the time the order entered the market. Buy orders completed at or below the arrival price, and sell orders completed at or above the arrival price are considered successful.

1.1 Clamped Linear Scaling

Scaling algorithms are a common method for executing orders benchmarked to arrival. Reversion-based scaling is based on the assumption that stock prices follow a mean-reverting random walk. Scaling algorithms dynamically alter the rate at which child orders are executed as measured by the participation rate, or percentage of volume at a given point in time.

Reversion-based scaling algorithms enter the market and snap benchmark to the arrival price, executing shares at the base participation rate, $\bar{V}$. $P_b = P_{t_0}$ where P_b is the benchmark arrival price and P_{t_0} is the prevailing market price at time t_0. Traders parameterize scaling algorithms with upper and lower bounded participation rates, and minimum and maximum price levels. Bounded price levels are generally a function of historical volatility. Traders can adjust all four parameters to suit different execution objectives.

Let:

- P_t prevailing market price at time *t*
- $V_{\min}$ minimum volume participation rate
- $V_{\max}$ maximum volume participation rate
- $\bar{V}$ benchmark participation rate
- $P_{\min}$ minimum price
- $P_{\max}$ maximum price

Figure 1 shows a clamped linear scaling function and a fixed percentage of volume (POV) function. As the market price drops below arrival price, the scaling buy order becomes more aggressive and increases participation rate. As the market price rises above arrival price, the scaling buy order becomes more passive by scaling down participation rate. By contrast, the fixed POV function executes at a static participation rate, irrespective of moves in P_t from P_b.

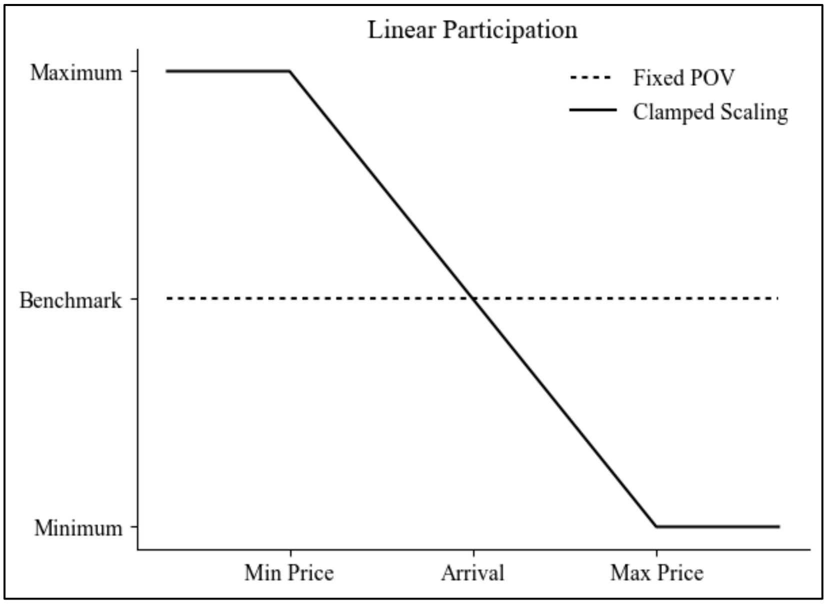


Figure 1. Participation rate as a function of price for a buy order

1.2 Four-Parameter Logistic Scaling

A logistic function is an S-shaped (sigmoid) curve commonly used to model growth that saturates over time.[2] The standard logistic function, shown in Figure 2, is defined as: $f(x) = \frac{1}{1+e^{-x}}$

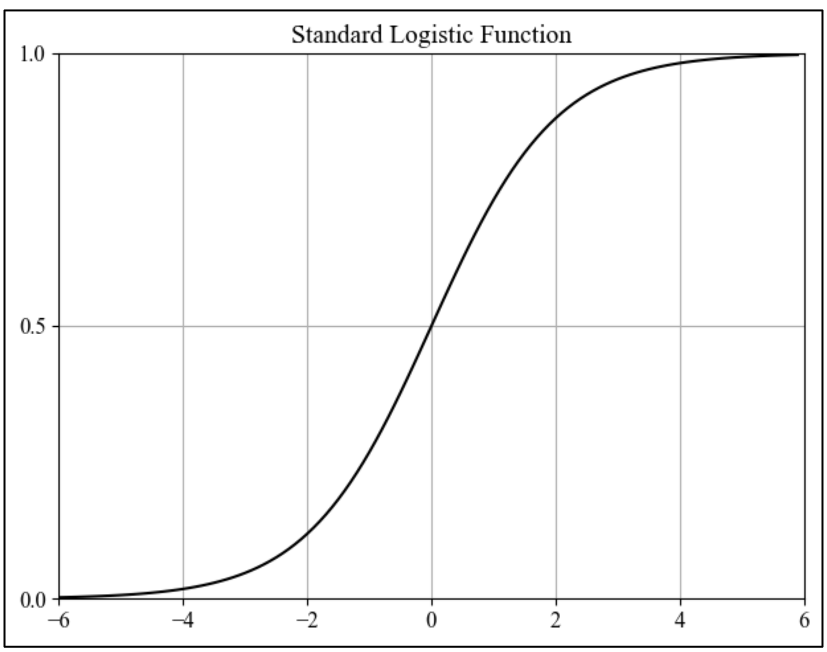


Figure 2.

[2] https://en.wikipedia.org/wiki/Logistic_function

The four-parameter logistic function (4PL) extends this idea by introducing parameters to control the shape and range. The 4PL is defined as:

$$f(x) = d + \frac{a - d}{1 + \left(\frac{x}{c}\right)^b}$$

Where:

- a is the lower asymptote
- d is the upper asymptote
- c is the inflection point
- b is the steepness of the curve

This research seeks to determine whether a customized algorithm that follows a 4PL scaling function leads to better outcomes than a clamped linear scaling function.

In this case, the 4PL function's asymptotes represent the minimum and maximum participation rates, the inflection point is the arrival price, and the slope parameter can be tuned to control how aggressively the scaling algorithm responds to price deviations. As shown in Figure 3, the 4PL results in smooth, bounded participation scaling compared to the clamped linear function. Given the assumption that stock prices follow a lognormal distribution, using the 4PL makes more sense than a clamped linear function, as it better aligns with the statistical properties of price paths.

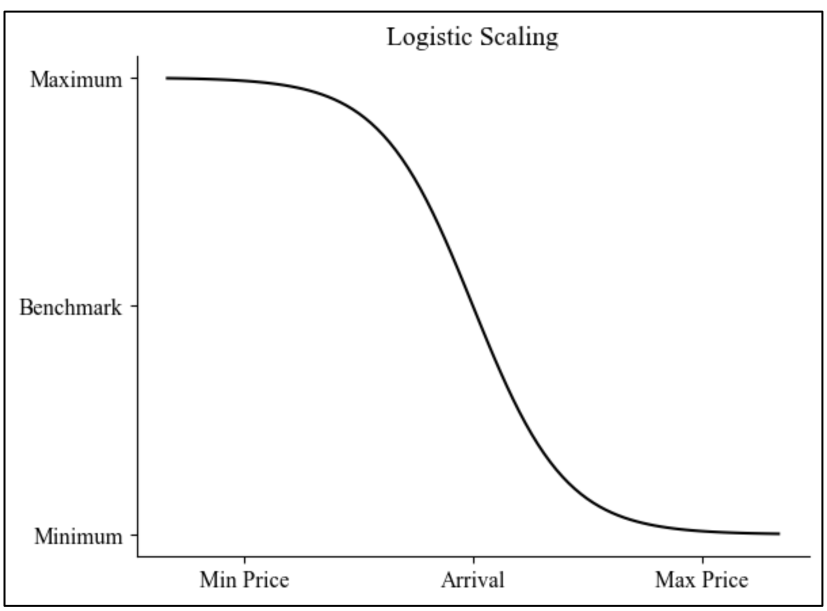


Figure 3. Participation rate as a function of price for a buy order

2 Data

2.1 Individual Stock Selection

Data was collected using the Bloomberg Python API[3]. One stock was selected at random from four different US equity indexes. The indexes chosen were the Standard & Poor's 100 Index, Standard & Poor's 500 Index, Standard & Poor's 400 Index, and Standard & Poor's 600 Index. These indexes were chosen to reflect two large-cap equities, one mid-cap equity, and one small-cap equity.

```
# select one stock at random from each of four indexes
index = ["OEX Index","SPX Index","MID Index","SML Index"]

test_sym = blp.bds(tickers = index,
                   flds = "INDX_MEMBERS").reset_index().rename(columns =
                                                                {'index':'index_id'})
test_sym = test_sym.groupby('index_id')[['index_id',
'member_ticker_and_exchange_code']].apply(lambda x:
x.sample(1)).reset_index(drop=True)['member_ticker_and_exchange_code'].tolist()

#modify symbology to match Bloomberg convention
test_sym = [sym[:-3] + " US Equity" for sym in test_sym]
```

[3] https://pypi.org/project/xbbg/

2.2 Testing Date Selection

Thirty unique, full trading days were selected at random from the first quarter of 2025 for testing. Holidays and half-days were accounted for and excluded from selection.

```
# select 30 full trading days in Q1 2025
def get_days(start_date, end_date):
    dates = []
    q1_holidays_us = [date(2025,1,1),
                      date(2025,1,20),
                      date(2025,2,17)] # holidays and half-days

    current_date = start_date
    while current_date <= end_date:
        if current_date not in q1_holidays_us and current_date.weekday() < 5:
            dates.append(current_date)
        current_date += timedelta(days=1)
    return dates

start_date = date(2025, 1, 1)
end_date = date(2025, 3, 31)
day_list = get_days(start_date, end_date)
test_days: list = random.sample(day_list, 30)
```

2.3 One-minute OHLC, Volume, and VWAP

For each of the four test symbols, one-minute open, high, low, close, volume, trade count, and trade value data were collected from Bloomberg. Volume and value data were used to calculate the one-minute volume-weighted average price (VWAP). Only data from the continuous trading session was used (09:30 to 15:59 ET).

	open	high	low	close	volume	num_trds	value	VWAP_1m
0	577.98	581.99	577.98	580.00	301207	926	174528544	579.4306
1	580.00	581.65	579.67	581.54	160759	849	93373152	580.8269
2	581.62	583.04	580.81	581.25	166773	1016	97087856	582.1557
3	581.47	582.43	580.71	580.82	131465	811	76453160	581.5476
4	580.93	581.09	579.71	579.91	96775	591	56169780	580.4162

Figure 4. Example data

2.4 Volatility and Average Daily Trading Volume

The 20-day volatility was collected for each test symbol on each test day. It reflects the standard deviation of day-to-day logarithmic relative historical price changes for the 20 most recent trading days' closing prices, annualized and expressed as a percentage.[4] Daily volatility was calculated as the annualized 20-day volatility divided by $\sqrt{252}$ and expressed as a decimal.

The 20-day average daily volume was collected for each test symbol on each test day. It is calculated as the total volume over the past 20 trading days divided by 20.[5]

```
for sym in test_sym:
    for dt in test_days:
        # trade, volume, and volatility data from bbg
        ohlc = blp.bdib(ticker = sym, dt = dt).reset_index(drop=True)
        ohlc = ohlc[ohlc.index < 390]
        twentyd_vol = blp.bdh(tickers = sym,
                              flds = ["VOLATILITY_20D"],
                              start_date = dt,
                              end_date = dt).reset_index(drop=True)
        twentyd_volm = blp.bds(tickers = sym,
                               flds = ["VOLUME_AVG_20D"]).reset_index(drop=True)
        twentyd_vol.columns = twentyd_vol.columns.droplevel()
        ohlc.columns = ohlc.columns.droplevel()

        # Establish daily volatility and 1-minute VWAP
        stddev_1d = twentyd_vol.loc[0, 'VOLATILITY_20D']/math.sqrt(252)/100
        ohlc['VWAP_1m'] = (ohlc['value'] / ohlc['volume'])
```

2.5 Limitations

Research was limited by constraints on the availability of one-minute OHLC data. Bloomberg supplies this data as far back as 140 days. A more comprehensive study could use a more expansive dataset to support analysis across multiple volume and volatility regimes.

[4] Bloomberg
[5] Bloomberg

3 Assumptions

To isolate the effects of participation scaling, it is assumed that the trader can accurately adjust participation rates to the thousandth on a minute-by-minute rolling basis. It is also assumed that the trader can execute at the volume-weighted average price in rolling one-minute buckets. This removes uncertainties related to volume forecasting and allows direct comparison of execution outcomes based on different participation rate functions.

3.1 Definitions

Benchmark: Performance testing begins at the open of the continuous trading session each test day. Benchmark price is taken as index position zero in the 'open' column of the daily OHLC dataset.

Minimum Participation Rate: Participation lower bound; set to 5 percent for testing.

Maximum Participation Rate: Participation upper bound; set to 15 percent for testing.

Mean (Base) Participation Rate: Participation rate at the arrival price; set to 10 percent for testing.

Minimum Price: The stock price corresponding to the maximum participation rate for buy orders, determined as a multiple of daily volatility; set to -1.5σ for testing.

Maximum Price: The stock price corresponding to the minimum participation rate for buy orders, determined as a multiple of daily volatility; set to $+1.5\sigma$ for testing.

```
# Set volume, volatility, and benchmark parameters
benchmark = ohlc.loc[0, 'open']
vol_min = 0.05
vol_max = 0.15
vol_mean = (vol_min + vol_max) / 2
px_max = benchmark * (1 + 1.5 * stddev_1d) # tuned to +1.5 std dev
px_min = benchmark * (1 - 1.5 * stddev_1d) # tuned to -1.5 std dev
```

4 Methodology

4.1 Clamped Linear Function

A clamped linear function can be established based on the definitions shown in Section 3.1.

Let:

- P_b benchmark price (e.g., arrival price)
- P prevailing market price
- $V_{\min}$ minimum volume participation rate
- $V_{\max}$ maximum volume participation rate
- $P_{\min}$ minimum price
- $P_{\max}$ maximum price
- $\bar{V}$ average (benchmark) participation rate, or $\bar{V} = \frac{V_{\min} + V_{\max}}{2}$
- m slope of the linear function
- b y-intercept of the linear function

The linear component is defined as:

$$m = \frac{V_{\max} - V_{\min}}{P_{\min} - P_{\max}}$$

$$b = \bar{V} - m \times P_b$$

$$f(P) = m \times P + b$$

The clamped linear participation function is defined as:

$$V(P) = \min(V_{\max}, \max\left(V_{\min}, f(P)\right))$$

Expanded inline:

$$V(P) = \min(V_{\max}, \max(V_{\min}, m \times P + b))$$

```
# Establish clamped linear function
def clamped_linear(P, benchmark, vol_min, vol_max, vol_mean, px_min, px_max):
    m = (vol_max - vol_min) / (px_min - px_max)
    b = vol_mean - m * benchmark

    return(np.minimum(vol_max, np.maximum(vol_min, m * P + b)))
```

4.2 Four-Parameter Logistic Function

A four-parameter logistic function can be established based on the definitions shown in Section 3.1.

Let:

- P_b benchmark price (e.g., arrival price)
- P prevailing market price
- $V_{\min}$ minimum volume participation rate
- $V_{\max}$ maximum volume participation rate
- $\bar{V}$ average (benchmark) participation rate, typically $\bar{V} = \frac{V_{\min} + V_{\max}}{2}$
- m slope of the linear function

The four-parameter logistic scaling function is defined as:

$$V(P) = V_{\min} + \frac{\bar{V}}{1 + e^{-\frac{4m}{\bar{V}}(P - P_b)}}$$

Where $V(P)$ is the participation rate at price P and where:

- $V_{\min}$ lower asymptote
- $V_{\max}$ upper asymptote
- P_b inflection point
- $-\frac{4m}{\bar{V}}$ slope parameter

```
# Establish four parameter logistic function
def logistic(P, benchmark, vol_min, vol_max, vol_mean, px_min, px_max):
    m = (vol_max - vol_min) / (px_min - px_max)
    b = vol_mean - m * benchmark

    return(vol_min + (vol_mean / (1 + np.exp(-4 * m / vol_mean) * (P - benchmark))))
```

The slope parameter controls how sharply the function transitions from the lower asymptote to the upper asymptote around the inflection point. In this case, how aggressively participation rate increases or decreases immediately around the benchmark price. The slope parameter m, defined as $-\frac{4m}{\bar{V}}$, was reverse engineered so that the 4PL and linear scaling functions had the same instantaneous slope at the inflection point, establishing consistency between the two models.

5 Results

The four stocks selected for testing were RBC Bearings Inc (RBC), The Bank of New York Mellon Corp (BK), Adeia Inc (ADEA), and Incyte Corp (INCY). Two paired-samples t-tests were run for each symbol to determine if clamped linear scaling leads to superior outcomes versus fixed participation and to determine if 4PL scaling leads to superior outcomes compared to clamped linear scaling.

Each test was a one-tailed relative t-test with $\alpha = 0.05$.

$$H_0: \mu_1 = \mu_2$$
$$H_1: \mu_1 \geq \mu_2$$

In all four cases, clamped linear scaling led to statistically significant superior outcomes versus fixed POV. However, in all four cases, 4PL scaling failed to demonstrate statistically significant superiority compared to clamped linear scaling and therefore the null hypothesis could not be rejected. See appendices A through D for specific price data.

- **RBC Bearings Inc (RBC)**

Paired One-Tailed t-Test: RBC

		N	Mean	Variance
Fixed POV		30	340.8172	502.0098
Clamped Linear		30	340.6890	501.1283
t-statistic	5.82723488			
p-value	1.2806E-06			

Paired One-Tailed t-Test: RBC

		N	Mean	Variance
Clamped Linear		30	340.6890	501.1283
4PL		30	340.7055	501.1607
t-statistic	-3.32966887			
p-value	9.9881E-01			

- **The Bank of New York Mellon Corp (BK)**

Paired One-Tailed t-Test: BK

	N	Mean	Variance
Fixed POV	30	84.4915	10.7603
Clamped Linear	30	84.4434	10.7080
t-statistic	6.37337404		
p -value	2.8747E-07		

Paired One-Tailed t-Test: BK

	N	Mean	Variance
Clamped Linear	30	84.4434	10.7080
4PL	30	84.4494	10.7122
t-statistic	-1.88923589		
p -value	9.6555E-01		

- **Adeia Inc (ADEA)**

Paired One-Tailed t-Test: ADEA

	N	Mean	Variance
Fixed POV	30	14.0701	1.5693
Clamped Linear	30	14.0592	1.5398
t-statistic	1.90654173		
p -value	3.3266E-02		

Paired One-Tailed t-Test: ADEA

	N	Mean	Variance
Clamped Linear	30	14.0592	1.5398
4PL	30	14.0595	1.5391
t-statistic	-2.07611524		
p -value	9.7657E-01		

- **Incyte Corp (INCY)**

Paired One-Tailed t-Test: INCY				
		N	Mean	Variance
Fixed POV		30	69.5437	19.6928
Clamped Linear		30	69.5126	19.6509
t-statistic	6.31928609			
p -value	3.3296E-07			

Paired One-Tailed t-Test: INCY				
		N	Mean	Variance
Clamped Linear		30	69.5126	19.6509
4PL		30	69.5147	19.6521
t-statistic	-0.95784285			
p -value	8.2697E-01			

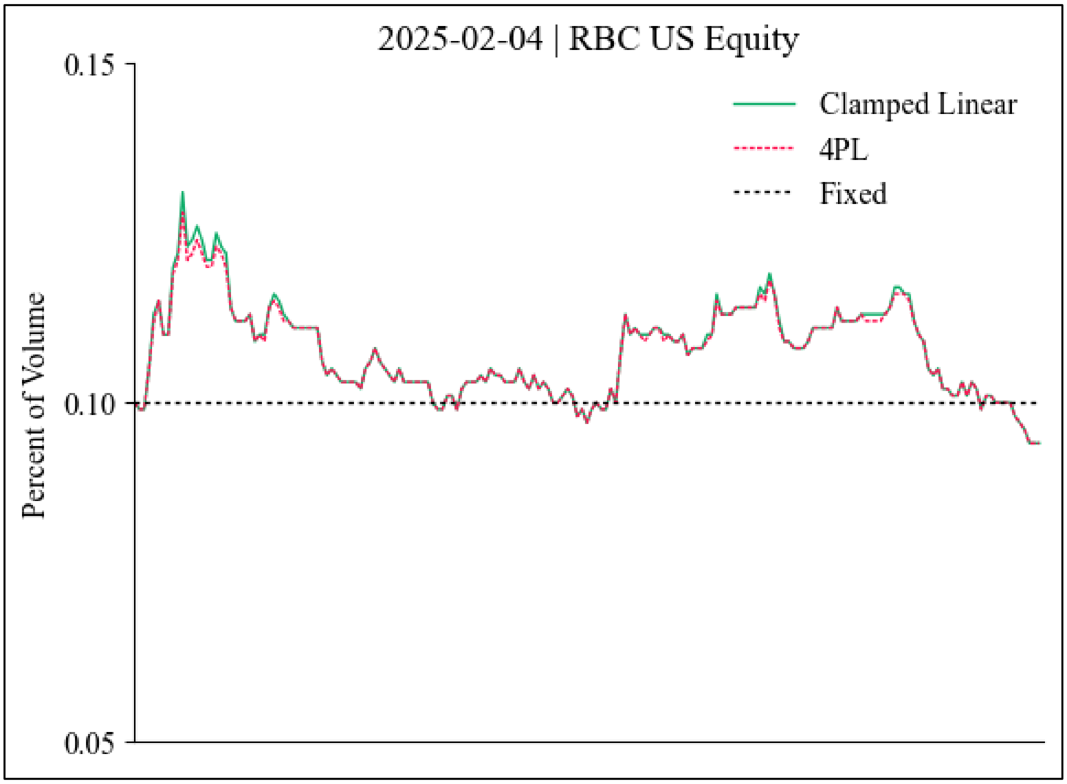
6 Conclusion

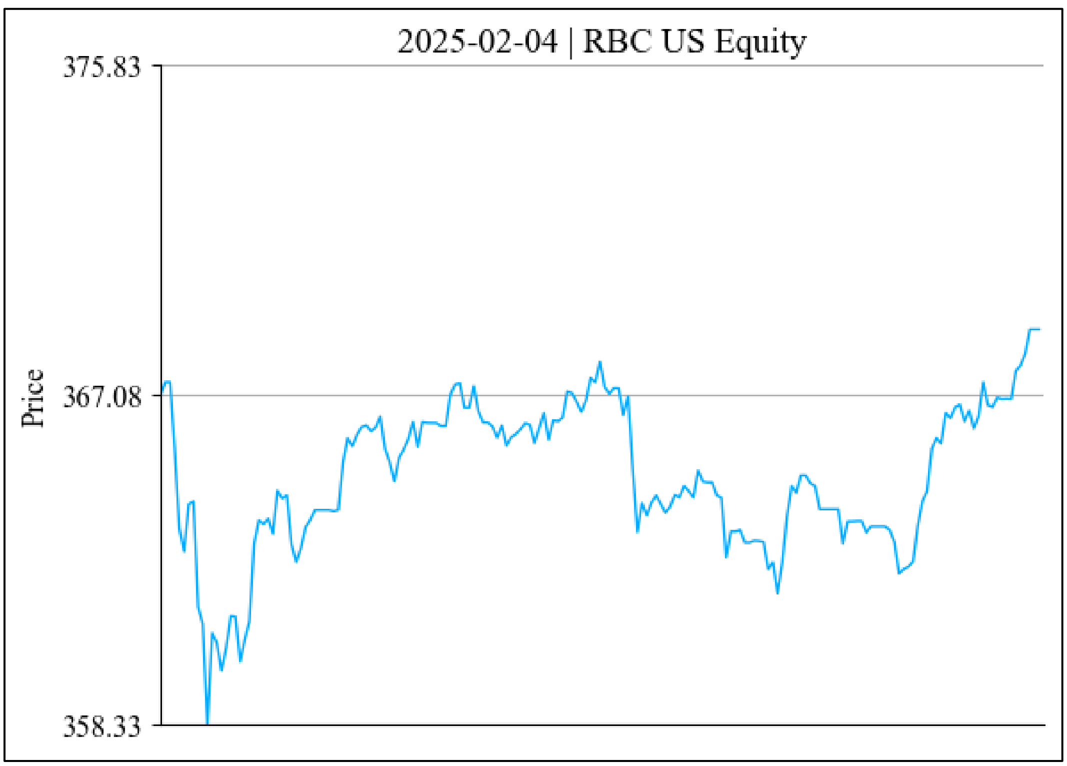
As stated, the clamped linear scaling function led to superior outcomes versus the fixed POV function in all four test cases, while the 4PL scaling function failed to demonstrate statistically significant superiority compared to clamped linear scaling in any case.

This could be due to the initial parameterization of the functions. The clamped linear and 4PL algorithms were set to scale between $\pm 1.5\sigma$ of the arrival price. The 4PL slope parameter $\left(-\frac{4m}{\bar{V}}\right)$ was reverse engineered to match the slope of the linear equation at the arrival price. While this seemed intuitive, in low-volatility environments the clamped linear and 4PL algorithms are practically indistinguishable. This is highlighted in the examples below. On 4-Feb-2025, the clamped linear algorithm and the 4PL algorithm are nearly identical for symbols RBC, BK and ADEA. Only INCY demonstrated a noticeable difference between clamped linear and 4PL scaling.

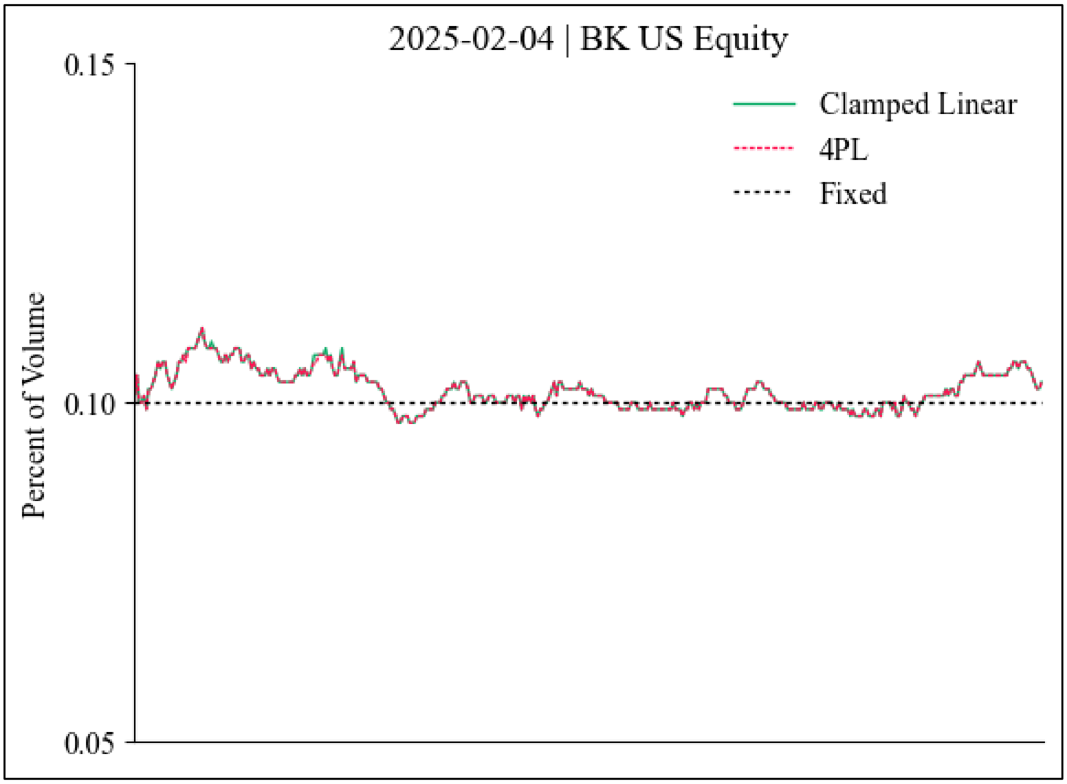
4PL functions are possibly still useful under the right circumstances. Future work could be to conduct sensitivity analysis by varying the standard deviation thresholds that drive participation adjustments, or by varying the four-parameter logistic function slope parameter.

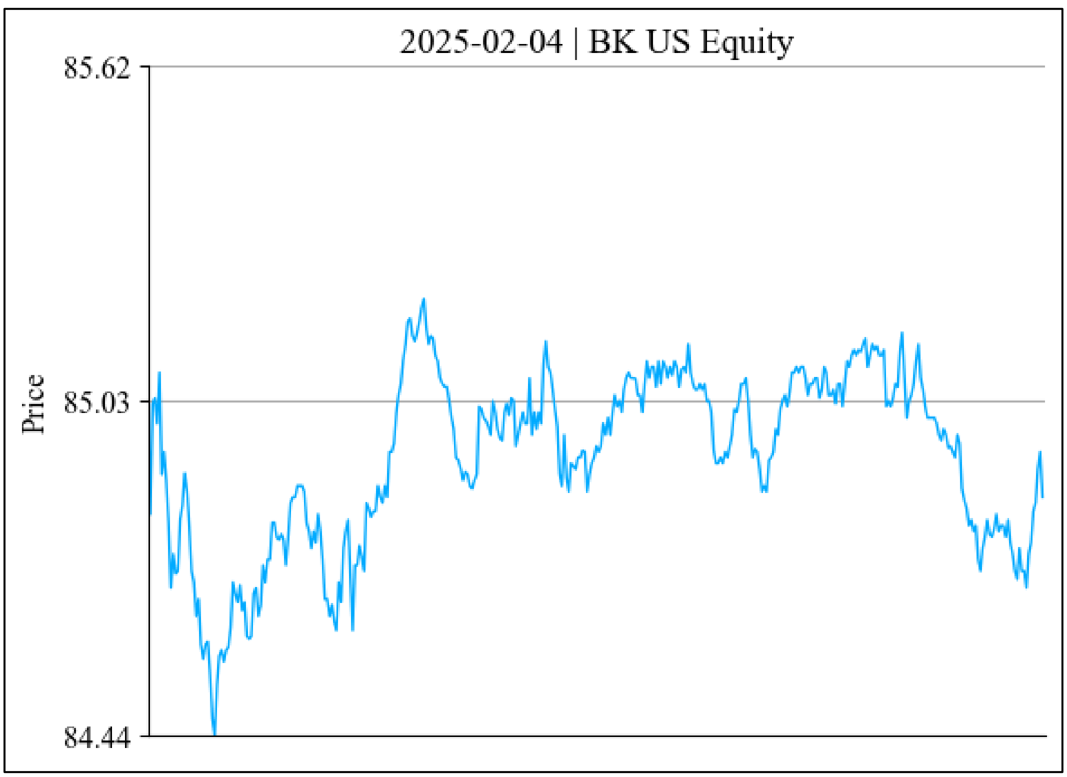
- **RBC Bearings Inc (RBC)**



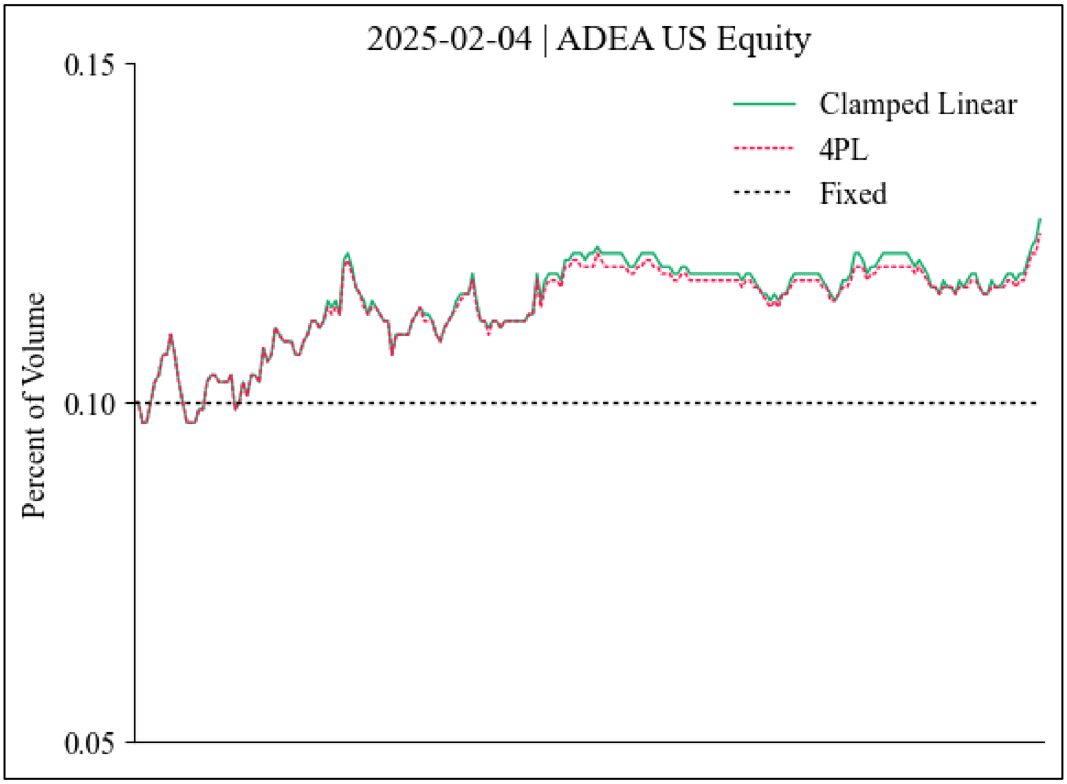


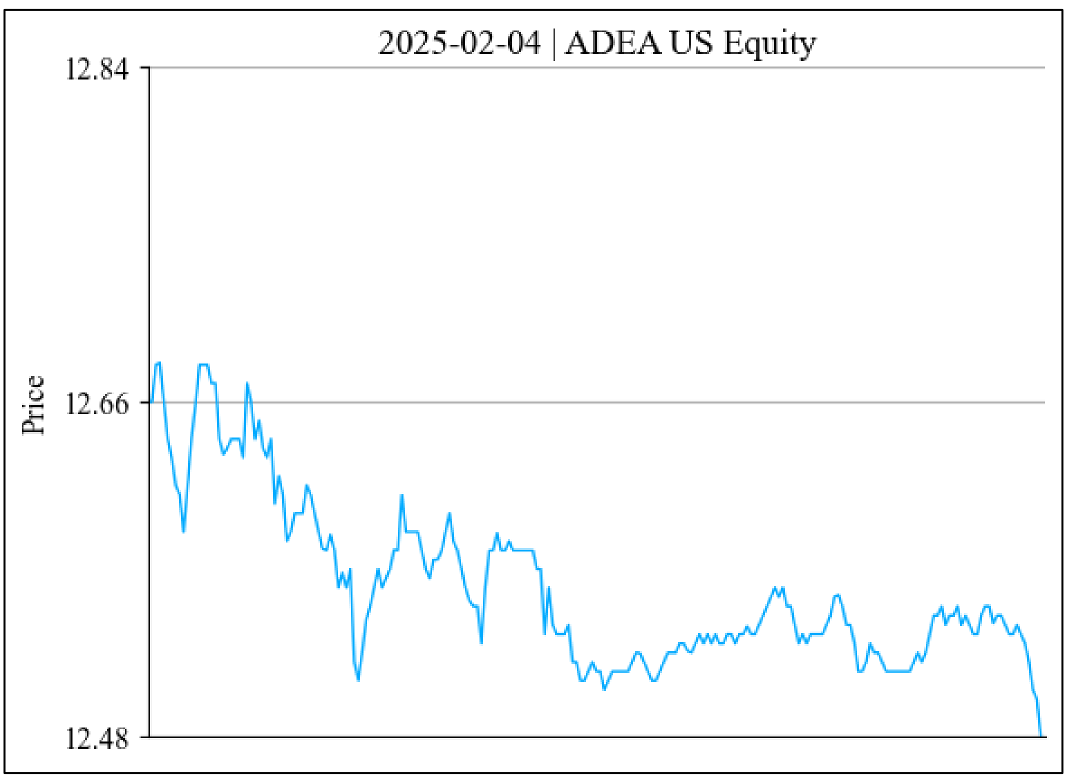
- **The Bank of New York Mellon Corp (BK)**



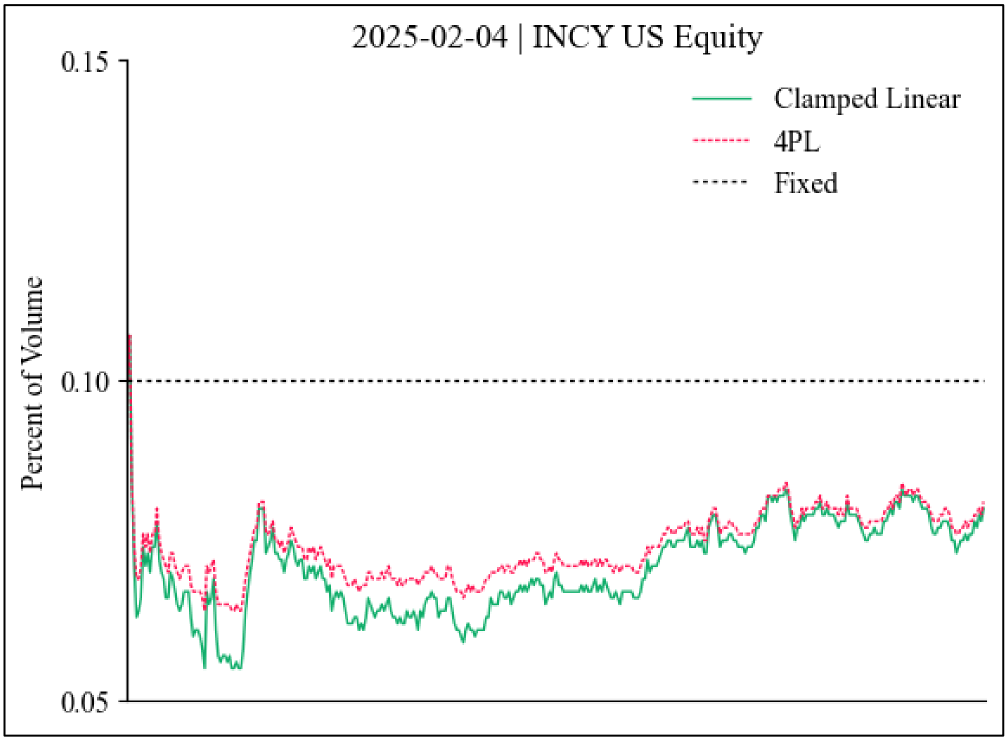


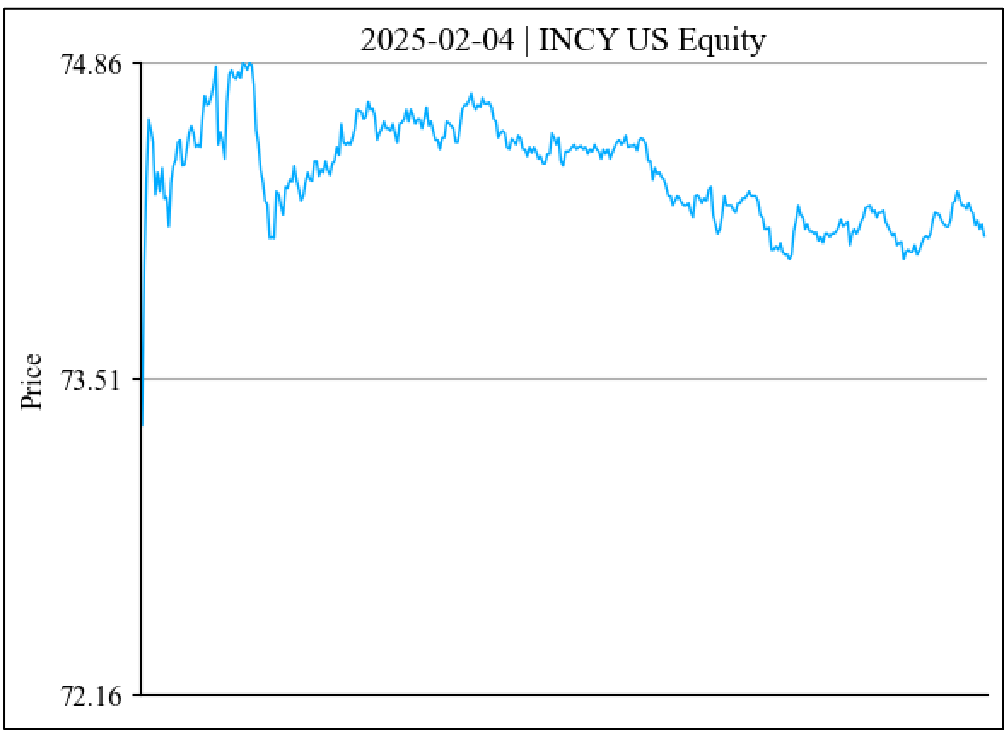
- **Adeia Inc (ADEA)**





- **Incyte Corp (INCY)**





Appendix A

A.1 RBC US Price Data

date	symbol	fixed	linear	log
14-Jan-25	RBC US	$ 303.9594	$ 303.8108	$ 303.8425
25-Mar-25	RBC US	$ 333.7259	$ 333.6056	$ 333.6170
11-Mar-25	RBC US	$ 341.7032	$ 341.5950	$ 341.6149
2-Jan-25	RBC US	$ 298.1543	$ 298.0506	$ 298.0659
14-Feb-25	RBC US	$ 363.1709	$ 363.1380	$ 363.1380
12-Feb-25	RBC US	$ 362.5497	$ 362.5207	$ 362.5206
26-Feb-25	RBC US	$ 359.0451	$ 358.9277	$ 358.9284
13-Mar-25	RBC US	$ 331.5950	$ 331.4099	$ 331.4479
24-Mar-25	RBC US	$ 334.6113	$ 334.5401	$ 334.5437
25-Feb-25	RBC US	$ 357.8552	$ 357.8271	$ 357.8271
19-Feb-25	RBC US	$ 370.6542	$ 370.6145	$ 370.6153
19-Mar-25	RBC US	$ 336.9603	$ 336.7101	$ 336.7330
16-Jan-25	RBC US	$ 308.8939	$ 308.7578	$ 308.7744
24-Jan-25	RBC US	$ 314.8596	$ 314.7875	$ 314.7879
5-Mar-25	RBC US	$ 354.5422	$ 354.0795	$ 354.2155
27-Jan-25	RBC US	$ 313.0284	$ 312.9906	$ 312.9926
12-Mar-25	RBC US	$ 336.6488	$ 336.5516	$ 336.5638
15-Jan-25	RBC US	$ 306.7884	$ 306.7095	$ 306.7257
28-Jan-25	RBC US	$ 314.4686	$ 314.3683	$ 314.3814
11-Feb-25	RBC US	$ 363.9833	$ 363.9303	$ 363.9346
31-Mar-25	RBC US	$ 320.2750	$ 320.2483	$ 320.2501
28-Feb-25	RBC US	$ 358.2867	$ 358.1316	$ 358.1415
14-Mar-25	RBC US	$ 336.1287	$ 336.0786	$ 336.0839
3-Mar-25	RBC US	$ 357.9745	$ 357.4478	$ 357.5200
27-Mar-25	RBC US	$ 333.3065	$ 333.2696	$ 333.2709
18-Feb-25	RBC US	$ 369.5484	$ 369.4827	$ 369.4878
4-Mar-25	RBC US	$ 350.4789	$ 350.2708	$ 350.3005
24-Feb-25	RBC US	$ 358.4052	$ 358.3114	$ 358.3129
5-Feb-25	RBC US	$ 367.6129	$ 367.3055	$ 367.3197
4-Feb-25	RBC US	$ 365.3024	$ 365.1982	$ 365.2086

Appendix B

B.1 BK US Price Data

date	symbol	fixed	linear	log
14-Jan-25	BK US	$ 76.1100	$ 76.0928	$ 76.0936
25-Mar-25	BK US	$ 85.1911	$ 85.1872	$ 85.1872
11-Mar-25	BK US	$ 81.8173	$ 81.7254	$ 81.7087
2-Jan-25	BK US	$ 77.4369	$ 77.4117	$ 77.4127
14-Feb-25	BK US	$ 87.9436	$ 87.9298	$ 87.9323
12-Feb-25	BK US	$ 85.1231	$ 85.0641	$ 85.0725
26-Feb-25	BK US	$ 87.0241	$ 86.9048	$ 86.9110
13-Mar-25	BK US	$ 79.9420	$ 79.9078	$ 79.9037
24-Mar-25	BK US	$ 84.8106	$ 84.8048	$ 84.8049
25-Feb-25	BK US	$ 87.0091	$ 86.9465	$ 86.9220
19-Feb-25	BK US	$ 88.2527	$ 88.2127	$ 88.2163
19-Mar-25	BK US	$ 83.7496	$ 83.7094	$ 83.7150
16-Jan-25	BK US	$ 81.5789	$ 81.5696	$ 81.5697
24-Jan-25	BK US	$ 85.8291	$ 85.8083	$ 85.8120
5-Mar-25	BK US	$ 86.3404	$ 86.2606	$ 86.2737
27-Jan-25	BK US	$ 85.2556	$ 85.2407	$ 85.2408
12-Mar-25	BK US	$ 80.3448	$ 80.2942	$ 80.3348
15-Jan-25	BK US	$ 80.7966	$ 80.7491	$ 80.7277
28-Jan-25	BK US	$ 86.2139	$ 86.2019	$ 86.2027
11-Feb-25	BK US	$ 85.0174	$ 84.9949	$ 84.9954
31-Mar-25	BK US	$ 83.4383	$ 83.2863	$ 83.3426
28-Feb-25	BK US	$ 87.9325	$ 87.8108	$ 87.8426
14-Mar-25	BK US	$ 81.3278	$ 81.2415	$ 81.2556
3-Mar-25	BK US	$ 89.0626	$ 88.9436	$ 88.9477
27-Mar-25	BK US	$ 84.7017	$ 84.6990	$ 84.6991
18-Feb-25	BK US	$ 88.7548	$ 88.7173	$ 88.7202
4-Mar-25	BK US	$ 85.4859	$ 85.4305	$ 85.4757
24-Feb-25	BK US	$ 87.8348	$ 87.7545	$ 87.7590
5-Feb-25	BK US	$ 85.5175	$ 85.5035	$ 85.5036
4-Feb-25	BK US	$ 84.9011	$ 84.8981	$ 84.8981

Appendix C

C.1 ADEA US Price Data

date	symbol	fixed	linear	log
14-Jan-25	ADEA US	$ 12.8539	$ 12.8488	$ 12.8496
25-Mar-25	ADEA US	$ 14.1942	$ 14.1915	$ 14.1922
11-Mar-25	ADEA US	$ 14.1300	$ 14.1275	$ 14.1275
2-Jan-25	ADEA US	$ 13.7958	$ 13.7856	$ 13.7875
14-Feb-25	ADEA US	$ 13.3190	$ 13.3155	$ 13.3158
12-Feb-25	ADEA US	$ 12.9769	$ 12.9750	$ 12.9751
26-Feb-25	ADEA US	$ 16.2319	$ 16.2267	$ 16.2267
13-Mar-25	ADEA US	$ 13.7697	$ 13.7669	$ 13.7669
24-Mar-25	ADEA US	$ 13.9262	$ 13.9243	$ 13.9243
25-Feb-25	ADEA US	$ 16.3713	$ 16.3627	$ 16.3616
19-Feb-25	ADEA US	$ 16.4600	$ 16.2856	$ 16.2838
19-Mar-25	ADEA US	$ 13.8407	$ 13.8365	$ 13.8368
16-Jan-25	ADEA US	$ 12.8114	$ 12.8073	$ 12.8075
24-Jan-25	ADEA US	$ 13.3317	$ 13.3264	$ 13.3268
5-Mar-25	ADEA US	$ 15.1477	$ 15.1438	$ 15.1438
27-Jan-25	ADEA US	$ 13.0749	$ 13.0639	$ 13.0645
12-Mar-25	ADEA US	$ 14.0844	$ 14.0822	$ 14.0822
15-Jan-25	ADEA US	$ 12.9778	$ 12.9711	$ 12.9729
28-Jan-25	ADEA US	$ 13.1197	$ 13.1164	$ 13.1165
11-Feb-25	ADEA US	$ 13.2257	$ 13.2123	$ 13.2147
31-Mar-25	ADEA US	$ 13.1858	$ 13.1851	$ 13.1852
28-Feb-25	ADEA US	$ 15.6051	$ 15.6034	$ 15.6034
14-Mar-25	ADEA US	$ 13.7510	$ 13.7479	$ 13.7479
3-Mar-25	ADEA US	$ 15.6497	$ 15.6474	$ 15.6472
27-Mar-25	ADEA US	$ 13.9583	$ 13.9547	$ 13.9549
18-Feb-25	ADEA US	$ 13.4122	$ 13.4068	$ 13.4071
4-Mar-25	ADEA US	$ 14.8921	$ 14.8717	$ 14.8720
24-Feb-25	ADEA US	$ 16.8734	$ 16.8610	$ 16.8623
5-Feb-25	ADEA US	$ 12.6149	$ 12.6139	$ 12.6140
4-Feb-25	ADEA US	$ 12.5174	$ 12.5150	$ 12.5152

Appendix D

D.1 INCY US Price Data

date	symbol	fixed	linear	log
14-Jan-25	INCY US	$ 72.0652	$ 72.0363	$ 72.0375
25-Mar-25	INCY US	$ 61.9688	$ 61.9354	$ 61.9416
11-Mar-25	INCY US	$ 68.3573	$ 68.2952	$ 68.2885
2-Jan-25	INCY US	$ 69.4628	$ 69.4537	$ 69.4539
14-Feb-25	INCY US	$ 70.5110	$ 70.4881	$ 70.4885
12-Feb-25	INCY US	$ 68.3443	$ 68.2158	$ 68.2583
26-Feb-25	INCY US	$ 74.3206	$ 74.3038	$ 74.3044
13-Mar-25	INCY US	$ 68.0254	$ 67.9937	$ 67.9947
24-Mar-25	INCY US	$ 62.7847	$ 62.7727	$ 62.7734
25-Feb-25	INCY US	$ 75.0012	$ 74.9242	$ 74.9357
19-Feb-25	INCY US	$ 70.3088	$ 70.2885	$ 70.2889
19-Mar-25	INCY US	$ 60.4031	$ 60.3969	$ 60.3970
16-Jan-25	INCY US	$ 72.4995	$ 72.4860	$ 72.4862
24-Jan-25	INCY US	$ 72.7449	$ 72.7130	$ 72.7177
5-Mar-25	INCY US	$ 69.6219	$ 69.6089	$ 69.6090
27-Jan-25	INCY US	$ 72.6309	$ 72.5968	$ 72.5992
12-Mar-25	INCY US	$ 67.5612	$ 67.5354	$ 67.5375
15-Jan-25	INCY US	$ 72.6801	$ 72.6540	$ 72.6547
28-Jan-25	INCY US	$ 73.1553	$ 73.0842	$ 73.0910
11-Feb-25	INCY US	$ 67.2014	$ 67.1558	$ 67.1571
31-Mar-25	INCY US	$ 60.3065	$ 60.2810	$ 60.2814
28-Feb-25	INCY US	$ 73.3473	$ 73.3361	$ 73.3363
14-Mar-25	INCY US	$ 67.7538	$ 67.7472	$ 67.7472
3-Mar-25	INCY US	$ 71.1741	$ 71.1470	$ 71.1053
27-Mar-25	INCY US	$ 60.6806	$ 60.6729	$ 60.6735
18-Feb-25	INCY US	$ 71.1189	$ 71.1105	$ 71.1107
4-Mar-25	INCY US	$ 69.1559	$ 69.1268	$ 69.1327
24-Feb-25	INCY US	$ 73.6854	$ 73.6143	$ 73.6272
5-Feb-25	INCY US	$ 75.1890	$ 75.1743	$ 75.1754
4-Feb-25	INCY US	$ 74.2506	$ 74.2301	$ 74.2356