Moving Average as an Unsupervised Learning Algorithm

Horace W. Tso

{ *htso.github.io* }

Dec. 18, 2017

Disclaimer

The information in this presentation is provided for information purposes only. It is not intended to be and does not constitute financial advice or any other advice, is general in nature and not specific to you. Before using this information to make an investment decision, you should seek the advice of a qualified and registered securities professional and undertake your own due diligence. None of the information here is intended as investment advice, as an offer or solicitation of an offer to buy or sell, or as a recommendation, endorsement, or sponsorship of any security, Company, or fund. I am not responsible for any investment decision made by you. You are responsible for your own investment research and investment decisions.

The information in this presentation is based on financial models, and trading signals are generated mathematically.

All of the signals, timing systems, and forecasts are the result of backtesting, and are therefore merely hypothetical.

Trading signals or forecasts used to produce our results were derived from equations which were developed through hypothetical reasoning based on a variety of factors. Theoretical buy and sell methods were tested against the past to prove the profitability of those methods in the past. Performance generated through back testing has many and possibly serious limitations. I do not claim that the historical performance of the timing systems, signals or forecasts will be indicative of future results. There will be substantial and possibly extreme differences between historical performance and future performance. Past performance is no guarantee of future performance. There is no guarantee that out-of-sample performance will match that of prior in-sample performance. I provide absolutely no guarantee that the trading signals, forecasts, opinions or analyses presented here are consistent, logical or free from hindsight or other bias or that the data used to generate signals in the backtests was available to investors on the dates for which theoretical signals were generated.

Moving Average

- Another moving average system ?

Moving Average

- Another moving average system ?

- False breaks, leading to frequent whipsaw.

Moving Average

- Another moving average system ?

- False breaks, leading to frequent whipsaw.
- Buy high and sell low.

Problems with Moving Average

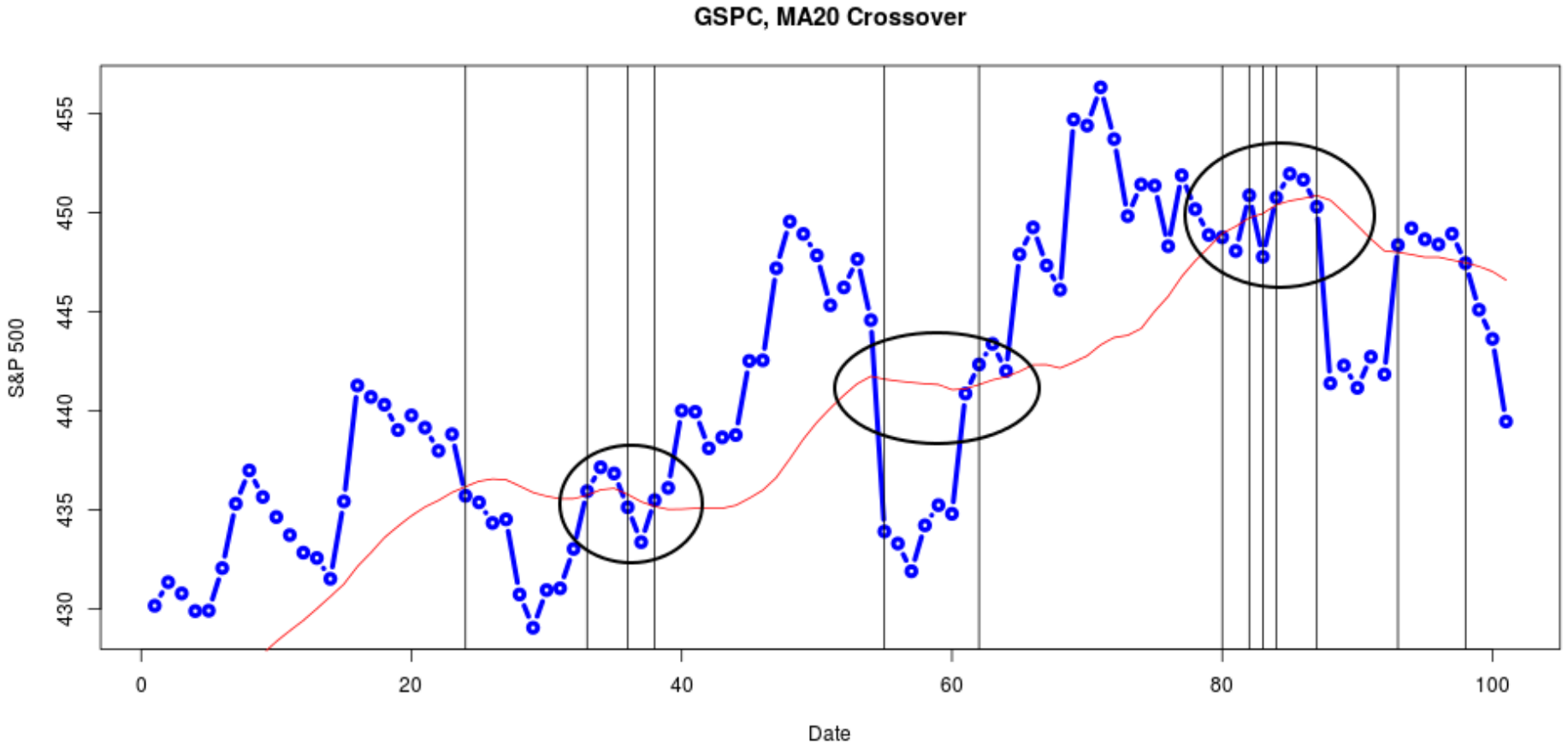

Figure 1 : So many false breaks, so much whipsaw

Problem with Moving Averages

- Underlying problem with traditional MA strategies is their rigidity, unresponsive to market dynamics.

Problem with Moving Averages

- Underlying problem with traditional MA strategies is their rigidity, unresponsive to market dynamics.
- Simple moving average is too average to be useful,

$$ma = \frac{1}{m} \sum_{i=1}^{m} x_i \tag{1}$$

Problem with Moving Averages

- Underlying problem with traditional MA strategies is their rigidity, unresponsive to market dynamics.
- Simple moving average is too average to be useful,

$$ma = \frac{1}{m} \sum_{i=1}^{m} x_i \tag{1}$$

- Gives equal and/or fixed weight to every data point. Indiscriminate treatment of all history. Predictive price events ignored.

Problem with Moving Averages

- Underlying problem with traditional MA strategies is their rigidity, unresponsive to market dynamics.
- Simple moving average is too average to be useful,

$$ma = \frac{1}{m} \sum_{i=1}^{m} x_i \tag{1}$$

- Gives equal and/or fixed weight to every data point. Indiscriminate treatment of all history. Predictive price events ignored.
- Other types of moving average, e.g. EMA, WMA, etc. suffer from the same problem.

Problem with Moving Averages

- Underlying problem with traditional MA strategies is their rigidity, unresponsive to market dynamics.
- Simple moving average is too average to be useful,

$$ma = \frac{1}{m} \sum_{i=1}^{m} x_i \tag{1}$$

- Gives equal and/or fixed weight to every data point. Indiscriminate treatment of all history. Predictive price events ignored.
- Other types of moving average, e.g. EMA, WMA, etc. suffer from the same problem.
- Can we do better?

How to make learning possible

- Main idea here is to encode a signal with historic patterns.

How to make learning possible

- Main idea here is to encode a signal with historic patterns.
- Detect and retain the interdependence in past prices – an alternative to autoregression in statistics.

How to make learning possible

- Main idea here is to encode a signal with historic patterns.
- Detect and retain the interdependence in past prices – an alternative to autoregression in statistics.
- Contrast with standard *ARIMA* models.
  - Does not require stationarity. Works directly with prices, not their first difference, as typically required to fit $ARMA(p, q)$.
  - Temporal dependence is revealed in trading profit, not statistical significance.

How to make learning possible

- Main idea here is to encode a signal with historic patterns.
- Detect and retain the interdependence in past prices – an alternative to autoregression in statistics.
- Contrast with standard *ARIMA* models.
  - Does not require stationarity. Works directly with prices, not their first difference, as typically required to fit $ARMA(p, q)$.
  - Temporal dependence is revealed in trading profit, not statistical significance.
- Absence of *linear* autocorrelation in the first difference does not preclude long range dependence.

Learnable Moving Average

Introduce a moving average that learns.

Definition (Adma)

For $m \in \mathbb{Z}^+$, $\vec{w} \in \mathbb{R}^m$, the Adaptive Moving Average is

$$\sum_{i=1}^{m} w_i \cdot x_{t-i+1} \tag{2}$$

where

$$0 \leq w_i \leq 1, \qquad i = 1, 2, \ldots, m \tag{3}$$

$$\sum_{i=1}^{m} w_i = 1. \tag{4}$$

Weights are the model parameters; one weight per time step that measures the relative importance of history. Note Eq. (3) and (4) ensure $\vec{w}$ spreads a unit weight among m points.

Unsupervised Learning

- *Adma* allows parametric modeling of trading strategy.
- Choice of $\vec{w}$ determines the timing of trade signals.
- For example, a buy when crossover from below, a sell when cross from above.
- Strategy P/L can be calculated from such decision rule.
- Different weight leads to a different strategy, results in different P/L.

Unsupervised Learning

- *Adma* allows parametric modeling of trading strategy.
- Choice of $\vec{w}$ determines the timing of trade signals.
- For example, a buy when crossover from below, a sell when cross from above.
- Strategy P/L can be calculated from such decision rule.
- Different weight leads to a different strategy, results in different P/L.
- Thus, the outcome of a strategy can be learned without a "true label" $\Rightarrow$ unsupervised learning.

Properties

Any simple moving average can be represented as an *Adma* of the same or higher order. The class of all simple MAs up to order m is a subset of a single *Adma* with parameter vector $\vec{w}$.

Lemma 1

For a given $n \in \mathbb{Z}^+$, let

$$\mathcal{M}_n := \{ma(1), ma(2), \ldots, ma(n)\},$$

for each $ma \in \mathcal{M}_n$, $\exists \vec{w} \in \mathbb{R}^n$ such that

$$Adma(\vec{w}) = ma$$

Proof : Set $w_i = 1/n$ for $i = 1, 2, \ldots, n$, and $w_i = 0, i > n$. ■

Properties

Adma preserves information in the window of history. It doesn't introduce new information into the model.

Lemma 2

Given $\{X_t\}_{t\in\mathcal{T}}$, $m \in \mathbb{Z}^+$,

$$\min(X_t^m) \leq Adma_t(m) \leq \max(X_t^m)$$

where $X_t^m \equiv X_{(t-m+1):t}$.

Proof : Since $w_i \geq 0$, $\sum_{i=1}^m w_i x_{t-m+i} \leq \sum_{i=1}^m w_i \max(x_i) = \max(X_t^m)$. Same argument for min, where we've used $\sum w_i = 1$. ■

Properties

- The weight of some data point may be zero, which means that these points are discarded.

Properties

- The weight of some data point may be zero, which means that these points are discarded.
- Example 1: if $w_m = 1$ and $w_i = 0$ for all $i < m$, then it picks up the last data point only and throws away everything else. Equivalent to $MA(1)$.

Properties

- The weight of some data point may be zero, which means that these points are discarded.
- Example 1: if $w_m = 1$ and $w_i = 0$ for all $i < m$, then it picks up the last data point only and throws away everything else. Equivalent to $MA(1)$.
- Example 2 : For $n < m$, $w_i = 1/n$ for $i > n$, its memory is shortened to order n, i.e., MA(n).
  $Admа_m = \sum_{i=1}^{m} w_i x_i = \sum_{i=1}^{m-n} 0 \cdot x_i + \sum_{j=m-n+1}^{m} w_j x_j = \sum_{i=1}^{n} w_j x_j = ma_n$.

Properties

- The weight of some data point may be zero, which means that these points are discarded.
- Example 1: if $w_m = 1$ and $w_i = 0$ for all $i < m$, then it picks up the last data point only and throws away everything else. Equivalent to $MA(1)$.
- Example 2 : For $n < m$, $w_i = 1/n$ for $i > n$, its memory is shortened to order n, i.e., MA(n).
  $Admа_m = \sum_{i=1}^{m} w_i x_i = \sum_{i=1}^{m-n} 0 \cdot x_i + \sum_{j=m-n+1}^{m} w_j x_j = \sum_{i=1}^{n} w_j x_j = ma_n$.
- Unlike simple moving average, any segment of history could be selected for a predictive strategy.

Decision Rule

- Like typical usage, an event is triggered when price crosses the *Adma*.
- Two varieties :
  a) *Trend following*. Long one share if price moves above the signal line, cover/short one share if goes below,

$$v_t = sign(x_t - adma_{t-1})$$

  b) *Reversion strategy*. Short one share if price moves above the signal, cover/long the same amount if it drops below,

$$v_t = -sign(x_t - adma_{t-1})$$

  where $sign(x)$ returns the sign of its argument.
- Alternative decision rule : trade occurs at every time step, but direction of trade flips at the crossing.

Trade Accounting

- Keep track of trades over time. Let v_t be the new position entered at time t, and V_t the total net position at time t before new trade is executed. Thus, $V_t = \sum_{i=1}^{t-1} v_i$.
- Initial condition $V_1 = 0$, i.e. no position at the start. Progressing in time, $V_2 = v_1, V_3 = v_2, \ldots$ etc. Notice a lag of one period is built into V_t to prevent look-ahead bias. No trade at the last time step.
- Valuing the positions over time. Let $C(t)$ be historic cost of all closed position up to time t. Per trade cost is number of shares times price,

$$\begin{aligned} C(t) &= \sum_{i=1}^{t-1} v_i \cdot x_i \\ &= C(t-1) + v_{t-1} \cdot x_{t-1} \end{aligned}$$

- Similarly, we set $C(1) = 0$, ie. no money is invested at the beginning, and $C(2) = v_1 x_1, C(3) = v_1 x_1 + v_2 x_2, \ldots$ etc.

Trade Accounting

- Let $R(t)$ be MTM of open positions at t. So, $R(t) = -V_t \cdot x_t$.
- $R(t)$ is the residual, or liquidation value of the entire portfolio given the latest price of the stock. Thus, $V_t = \sum_{i=1}^{t-1} v_i = 0$ implies $R(t) = 0$.
- If all the trades have been covered, there'd be no residual value.
- Strategy's cumulative loss $L(t)$ up to time t is therefore,

$$L(t) = C(t) + R(t)$$

- It's sum of net cost of trades minus the MTM value of open positions. Reflects all positions ever traded up to $t-1$, and net position being liquidated at the current price x_t.

Strategy Comparison

In strategy view, *Adma* allows us to construct expressive models for a given dataset. Everything achievable with a traditional MA is within reach with an *Adma* signal.

Theorem

Let $D : E \to \mathbb{R}$ be a given decision rule on the event set E generated by a signal line, let $\mathbb{G}(m, D)$ be the set of all simple moving average strategies of order m using decision rule D, $\mathbb{F}(m, D)$ the set of all *Adma* strategies of same order and same decision rule. For any $v \in \mathbb{G}(m, D)$, there exists a $u \in \mathbb{F}(m, D)$ such that

$$\pi_D(u) \geq \pi_D(v)$$

where $\pi(\cdot)$ is the strategy's total profit.

Problem Formulation

Question : How to learn this model from data?

Solve a constraint optimization problem with the strategy's P/L as the objective function.

Given $m \in \mathbb{Z}^+, C_0 \in \mathbb{R}^+, \{X_t\}_{t \in \mathcal{T}}, g : \mathbb{R} \to \mathbb{R}^+,$

$$\vec{w}^* = \arg\min_{\vec{w}} \; L(\vec{w}, X) \tag{5}$$

subject to

$$0 \leq w_i \leq 1, \qquad i = 1, 2, \ldots, m \tag{6}$$

$$\sum_{i=1}^{m} w_i = 1 \tag{7}$$

$$g(R(t)) \leq C_0, \qquad \forall t \tag{8}$$

Note Eq (8) enforces capital limit, leverage, and/or risk policy.

Optimization

Very challenging problem because the loss function $L(\vec{w})$ is,

- Non-convex $\Rightarrow$ no guarantee of global solution, unknown (and large) number of local optima, and saddle points,

Optimization

Very challenging problem because the loss function $L(\vec{w})$ is,

- Non-convex $\Rightarrow$ no guarantee of global solution, unknown (and large) number of local optima, and saddle points,
- Non-differentiable $\Rightarrow$ all gradient methods would fail, e.g. SGD, L-BFGS-B, 2nd order Newton's methods,

Optimization

Very challenging problem because the loss function $L(\vec{w})$ is,

- Non-convex $\Rightarrow$ no guarantee of global solution, unknown (and large) number of local optima, and saddle points,
- Non-differentiable $\Rightarrow$ all gradient methods would fail, e.g. SGD, L-BFGS-B, 2nd order Newton's methods,
- ... but continuous.

Optimization

Very challenging problem because the loss function $L(\vec{w})$ is,

- Non-convex $\Rightarrow$ no guarantee of global solution, unknown (and large) number of local optima, and saddle points,
- Non-differentiable $\Rightarrow$ all gradient methods would fail, e.g. SGD, L-BFGS-B, 2nd order Newton's methods,
- ... but continuous.
- Jagged terrain induced by crossover decision rule (see Fig).

Optimization

Very challenging problem because the loss function $L(\vec{w})$ is,

- Non-convex $\Rightarrow$ no guarantee of global solution, unknown (and large) number of local optima, and saddle points,
- Non-differentiable $\Rightarrow$ all gradient methods would fail, e.g. SGD, L-BFGS-B, 2nd order Newton's methods,
- ... but continuous.
- Jagged terrain induced by crossover decision rule (see Fig).
- Combinatoric search runs up against the curse of dimensionality as $\vec{w}$ gets bigger.

Optimization

Very challenging problem because the loss function $L(\vec{w})$ is,

- Non-convex $\Rightarrow$ no guarantee of global solution, unknown (and large) number of local optima, and saddle points,
- Non-differentiable $\Rightarrow$ all gradient methods would fail, e.g. SGD, L-BFGS-B, 2nd order Newton's methods,
- ... but continuous.
- Jagged terrain induced by crossover decision rule (see Fig).
- Combinatoric search runs up against the curse of dimensionality as $\vec{w}$ gets bigger.
- Current research in non-convex optimization suggests that finding general solution is NP-hard.

Objective Function

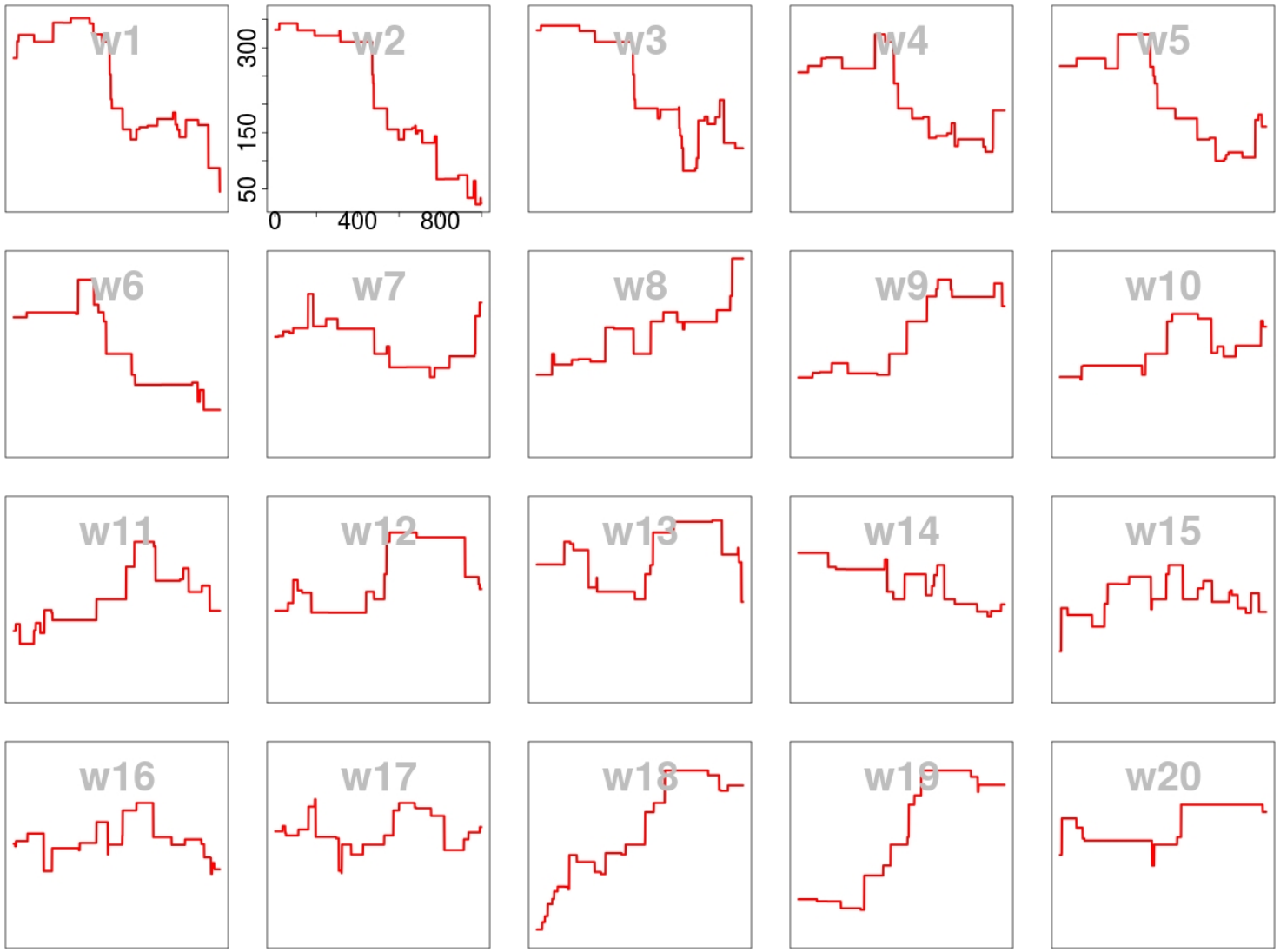

Figure 2 : Rugged terrain : Loss function over parameter space

Non-Convex Optimization Techniques

- Some optimization techniques do not require convexity and differentiability.

Non-Convex Optimization Techniques

- Some optimization techniques do not require convexity and differentiability.
- *Particle Swarm Optimization* uses analogous social behavior of fish school and bees to search for optimum [3].

Non-Convex Optimization Techniques

- Some optimization techniques do not require convexity and differentiability.
- *Particle Swarm Optimization* uses analogous social behavior of fish school and bees to search for optimum [3].
- *Simulated Annealing* from statistical physics. Heat up a system then gradually cool it down to settle in an optimal state [4].

Non-Convex Optimization Techniques

- Some optimization techniques do not require convexity and differentiability.
- *Particle Swarm Optimization* uses analogous social behavior of fish school and bees to search for optimum [3].
- *Simulated Annealing* from statistical physics. Heat up a system then gradually cool it down to settle in an optimal state [4].
- *Differential Evolution*. One type of genetic algorithm that searches with ever evolving candidates [2].

Non-Convex Optimization Techniques

- Some optimization techniques do not require convexity and differentiability.
- *Particle Swarm Optimization* uses analogous social behavior of fish school and bees to search for optimum [3].
- *Simulated Annealing* from statistical physics. Heat up a system then gradually cool it down to settle in an optimal state [4].
- *Differential Evolution*. One type of genetic algorithm that searches with ever evolving candidates [2].
- Open-source solvers available in C++, python and R.

Training Data

- Dataset : 26 years of S&P 500 daily close from 01-01-1985 to 10-11-2011.
  Source : Yahoo

Training Data

- Dataset : 26 years of S&P 500 daily close from 01-01-1985 to 10-11-2011. *Source* : Yahoo
- Raw index without dividend adjustment. Total 6713 observations.

Results

Procedure : Weights initialized with uniform random values whenever possible. Run multiple instances of all three solvers in parallel on a multi-core machine (max out computing capacity). Choose # iterations according to optimizer's capability and training budget. $m = 20$ for *Adma* and baselines.

	Lifetime Loss	Ret p.t.	Volatility (Ann)	Max Drawdown	# trades
	pts	%	%	pts	
Adma	-1716.65	0.34	8.3	353.23	613
Baselines* [5]:					
SMA	600.95	0.005	10.9	1253.65	751
EMA	333.38	0.042	9.94	1094.90	837
HMA	1994.75	-0.101	10.3	2623.31	1363
WMA	795.84	-0.035	10.3	1318.12	973
ZLEMA	2367.04	-0.132	11.1	2797.53	1358

Table 1 : Strategy Comparsion

* Simple MA, Exponential MA, Hull MA, Weighted MA, Zero-lag exponential MA[5].

Results

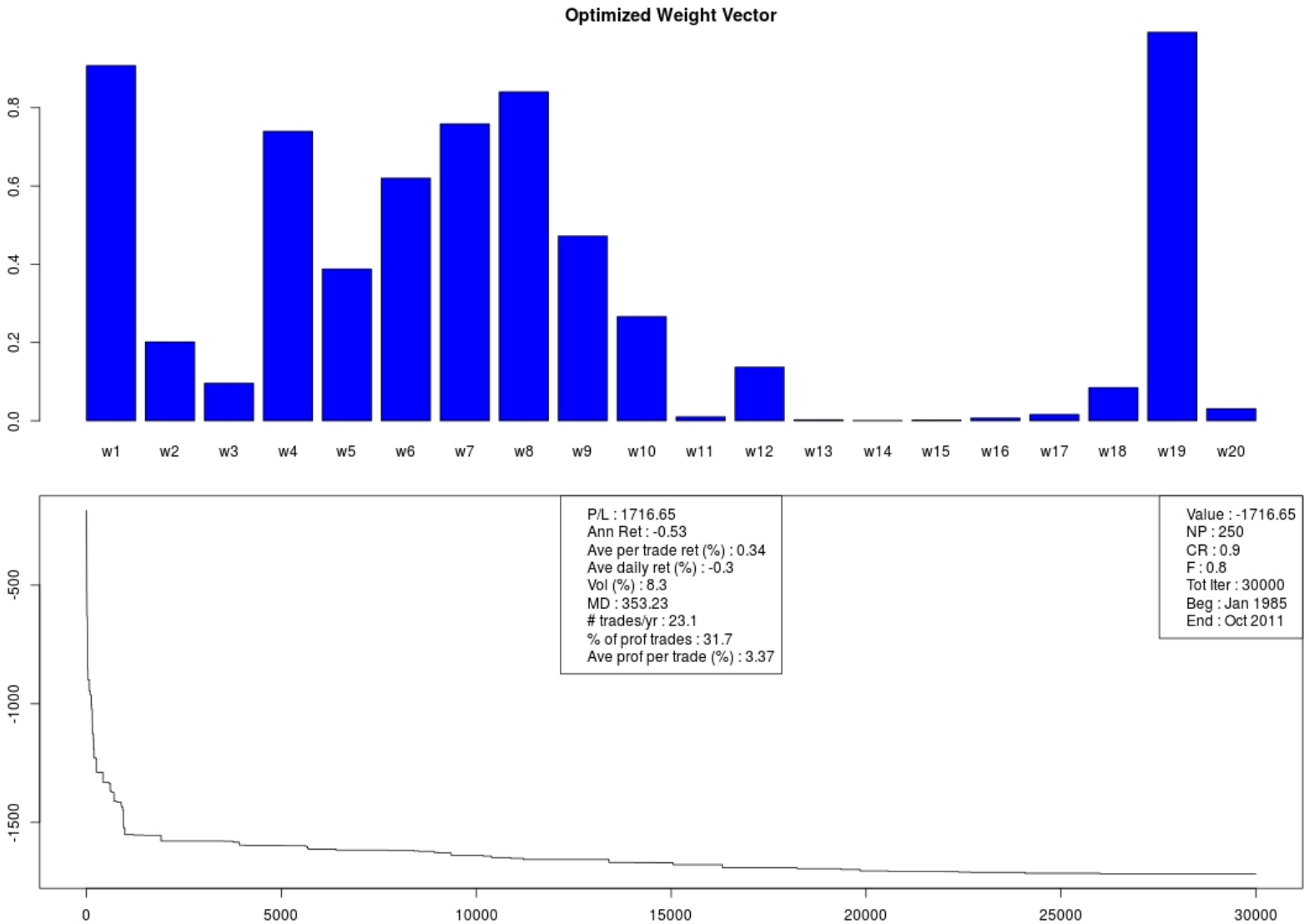

Figure 3 : Weight shape shows discriminative structure in price history.

Results

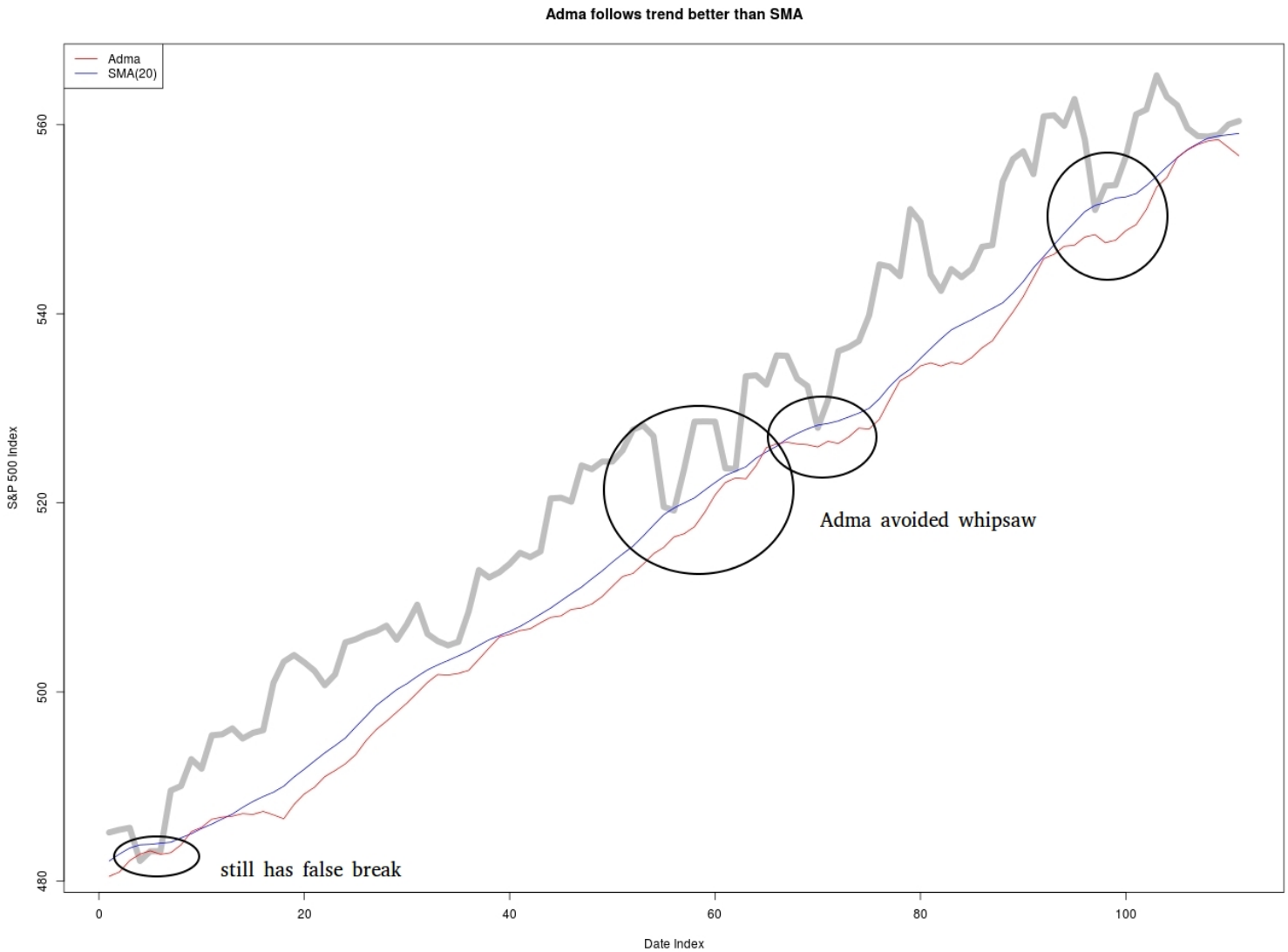

Figure 4 : Adma follows trend better than SMA

Model Validation

- How well does it do in period outside of the training set? Out-of-sample validation.

Model Validation

- How well does it do in period outside of the training set? Out-of-sample validation.
- Use the optimized weight vector to generate signals in six progressively longer periods following training set.

	1 Year	2 Years	3 Years	4 Years	5 Years	6 Years
Loss	226.40	203.51	203.79	307.76	853.02	936.52
Volatility (%)	12.24	11.44	10.57	26.73	9.33	9.58
Max Drawdown	319.84	319.84	335.80	413.37	1083.54	1140.81
No of trades	28.00	48.00	76.00	107.00	157.00	188.00

Table 2 : Extended Period Performance

Model Validation

- How well does it do in period outside of the training set? Out-of-sample validation.
- Use the optimized weight vector to generate signals in six progressively longer periods following training set.

	1 Year	2 Years	3 Years	4 Years	5 Years	6 Years
Loss	226.40	203.51	203.79	307.76	853.02	936.52
Volatility (%)	12.24	11.44	10.57	26.73	9.33	9.58
Max Drawdown	319.84	319.84	335.80	413.37	1083.54	1140.81
No of trades	28.00	48.00	76.00	107.00	157.00	188.00

Table 2 : Extended Period Performance

- Symptoms of overfitting.

Regularization

- Regularization reduces overfitting by penalizing the excess capacity of a model. Bias variance trade-off[6].

Regularization

- Regularization reduces overfitting by penalizing the excess capacity of a model. Bias variance trade-off[6].
- Only need to consider L2 and L0 regularization.

Regularization

- Regularization reduces overfitting by penalizing the excess capacity of a model. Bias variance trade-off[6].
- Only need to consider L2 and L0 regularization.
- L0 induces sparsity in weight parameter space, while L2 penalizes concentrated weights.

	1 Year	2 Years	3 Years	4 Years	5 Years	6 Years
Loss	(111.44)	(221.51)	(222.05)	(67.64)	268.08	517.94
Volatility (%)	11.76	9.85	9.40	8.92	8.22	8.89
Max Drawdown	190.85	190.85	190.85	319.81	689.81	887.55
No of trades	24	50	80	111	145	170

Table 3 : L0-Regularization

Discussion

- Optimized *Adma* strategy performs better than any fixed weight moving average on key metrics.

Discussion

- Optimized *Adma* strategy performs better than any fixed weight moving average on key metrics.
- Learns memory structure that is otherwise hard to see from price chart.

Discussion

- Optimized *Adma* strategy performs better than any fixed weight moving average on key metrics.
- Learns memory structure that is otherwise hard to see from price chart.
- Drawbacks : difficult to assess the quality of the local minima; require large investment in computational power and training time.

Discussion

- Optimized *Adma* strategy performs better than any fixed weight moving average on key metrics.
- Learns memory structure that is otherwise hard to see from price chart.
- Drawbacks : difficult to assess the quality of the local minima; require large investment in computational power and training time.
- Requires careful planning and interpretation of the resulting optimized parameters.

Discussion

- Optimized *Adma* strategy performs better than any fixed weight moving average on key metrics.
- Learns memory structure that is otherwise hard to see from price chart.
- Drawbacks : difficult to assess the quality of the local minima; require large investment in computational power and training time.
- Requires careful planning and interpretation of the resulting optimized parameters.
- Promising approach that could make many technical indicators learnable.

- Strongin, R. G., Sergeyev, Y. D. (2013). Global optimization with non-convex constraints: Sequential and parallel algorithms (Vol. 45). Springer Science & Business Media.
- Storn, R.; Price, K. (1997), Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization. 11: 341-359.
- Shi, Y. (2001). Particle swarm optimization: developments, applications and resources. In evolutionary computation, 2001. Proceedings of the 2001 Congress on (Vol. 1, pp. 81-86). IEEE.
- Khachaturyan, A.; Semenovskaya, S.; Vainshtein, B. (1979), Statistical-Thermodynamic Approach to Determination of Structure Amplitude Phases. Sov.Phys. Crystallography. 24 (5): 519-524.
- www.fmlabs.com/reference
- Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, pp. 241-249). New York: Springer series in statistics.