



TrendCalculus, defined.

How to execute a data science of trends.

2017-05-01

Andrew Morgan.

Copyright © 2015 ByteSumo Limited
All rights reserved.

Set your PDF viewer to display this
doc as a slide show for best viewing

Part 1. Calculating Trends
Part 2. Predicting Trend Change
Part 3. Co-Trending

Calculuating Trends

How I do the math.

Studying Trends

My focus is broad patterns; big flock behaviours, and my objective is long range predictions.

Trends are a natural way to think, explain, and forecast.

Yet we lack tools to study Trends effectively.

This project seeks to rectify that.



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

In these slides I will

- Set out a revised definition of a trend
- Set out an equation for finding trends, reversals.
- Explain some of my data formats, charting etc.
- Explain how to handle special cases
- Show how to identify trend reversals on streams



Studying Trends

before we begin

A comparative study of billions of time series demands methods that are quick, simple, elegant.

Don't shrug off the difficulty of achieving simplicity.
“Obvious” is the toughest goal of invention.



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

Let's start from scratch.



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

I define a trend as:

Rising = Higher Highs, Higher Lows

Falling = Lower Highs, Lower Lows



Studying Trends

It's an established market definition.

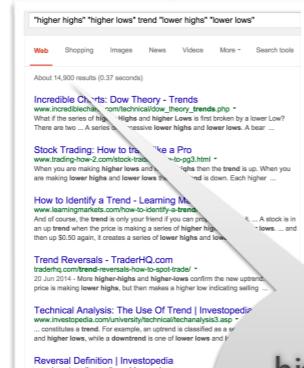
Rising = Higher Highs, Higher Lows
Falling = Lower Highs, Lower Lows

This definition comes from market traders
right back to W. D. Gann (1923)



Studying Trends

Market practitioners really use it.



Copyright © 2015 ByteSumo Limited
All rights reserved.

~15k google hits reference this definition, so it's quite well known.

- = Higher Highs, Higher Lows
- = Lower Highs, Lower Lows

This comes from market traders back to W. D. Gann (1923)



Studying Trends

I wish to *calculate* this definition as:

Rising = +1 Higher Highs, Higher Lows
Falling = -1 Lower Highs, Lower Lows



Studying Trends

So I must create a way to do so.

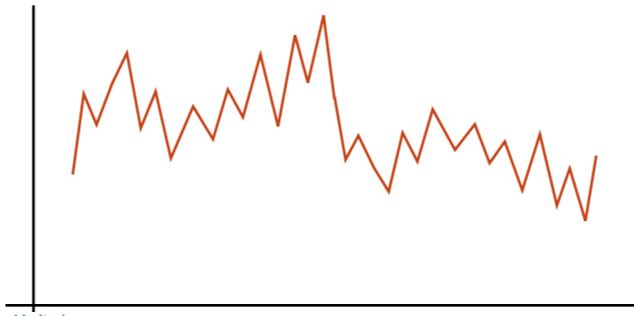


Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

So I must create a way to do so.
But how?

I'll illustrate.

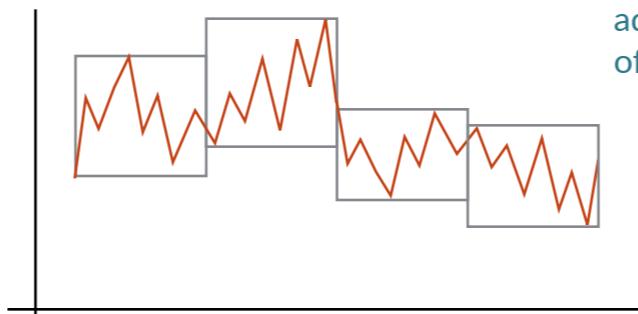


Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

Stream data across fixed window size of N.



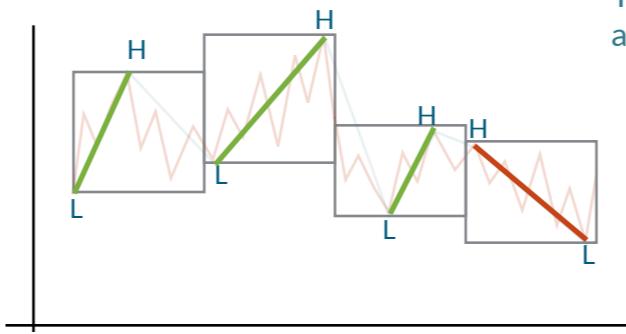
Selecting N will
adjusts our Scale
of investigation



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

Calculate a summary for each window
Find dated Highs and Lows for them.

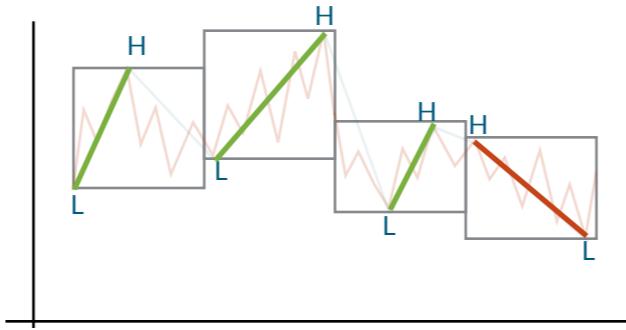


Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

Examine summaries across windows.

Compare the leading window against the previous

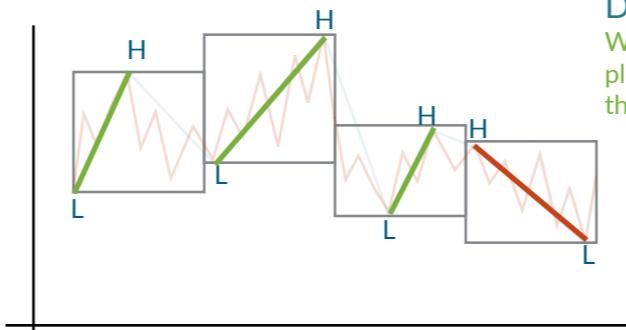


Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

We now need some math, to generate +1 or -1 values for each window.



Copyright © 2015 ByteSumo Limited
All rights reserved.

Develop a Calculation
What equation can emit plus and minuses from this input data?



Studying Trends

the TrendCalculus equation is:

where

$H, L = High, Low$

$p_i = CurrentWindow$

$p_{i-1} = PreviousWindow$



Studying Trends

the TrendCalculus equation is:

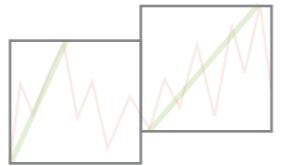
$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$

where

$H, L = \text{High, Low}$

$p_i = \text{CurrentWindow}$

$p_{i-1} = \text{PreviousWindow}$

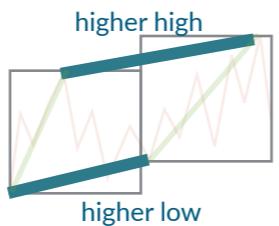


Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

the TrendCalculus equation is:

$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$



where

$H, L = \text{High, Low}$

$p_i = \text{CurrentWindow}$

$p_{i-1} = \text{PreviousWindow}$



Copyright © 2015 ByteSumo Limited
All rights reserved.

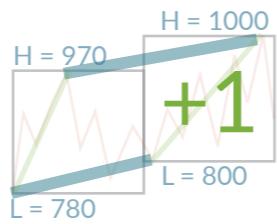
Studying Trends

Calculate trends as (+1, -1) for a window=

$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$

for example:

Simple Trend = Sign(sign(HighDiff) + sign(LowDiff))
Simple Trend = Sign(sign(1000-970) + sign(800-780))
Simple Trend = Sign(sign(30) + sign(20))
Simple Trend = Sign(1 + 1)
Simple Trend = Sign(2)
Simple Trend = +1



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

Calculate trends as (+1, -1) for a window=

$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$

for example:



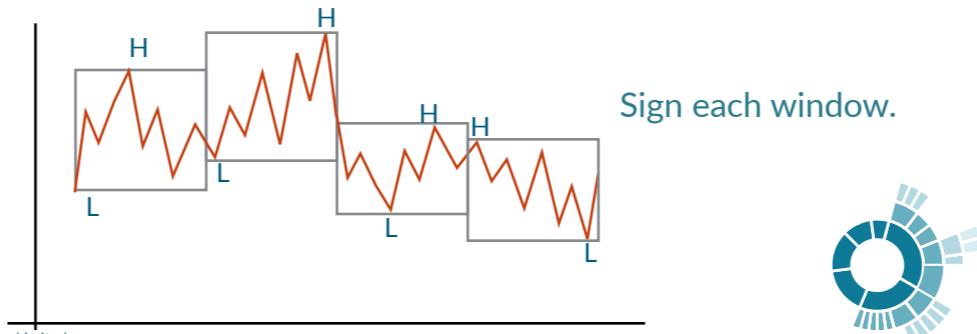
Simple Trend = Sign(sign(HighDiff) + sign(LowDiff))
Simple Trend = Sign(sign(1000-970) + sign(800-780))
Simple Trend = Sign(sign(30) + sign(20))
Simple Trend = Sign(1 + 1)
Simple Trend = Sign(2)
Simple Trend = +1



Studying Trends

More detailed example

$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$



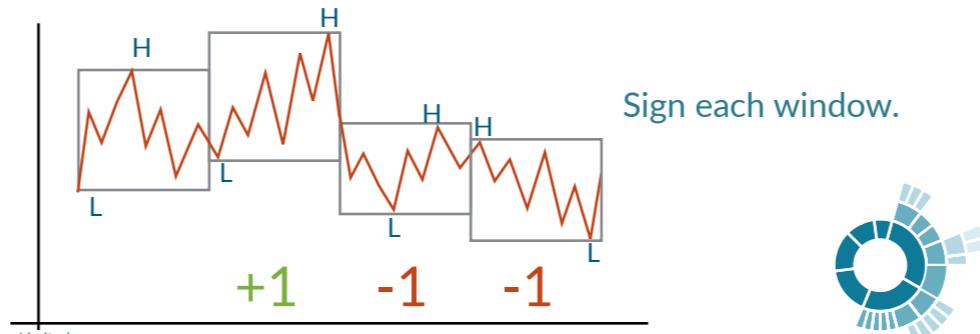
Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

...where we identify a reversal.

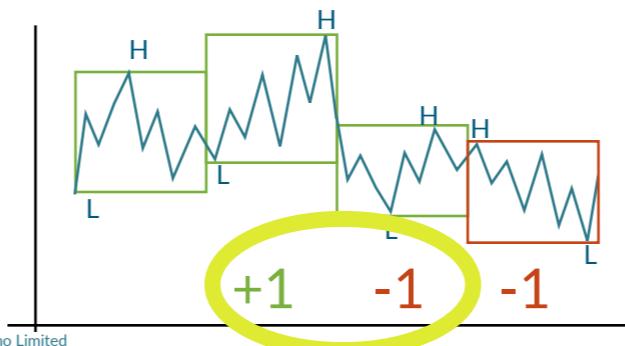
$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

When the trend values flip, we find
a change in trend on the previous window.



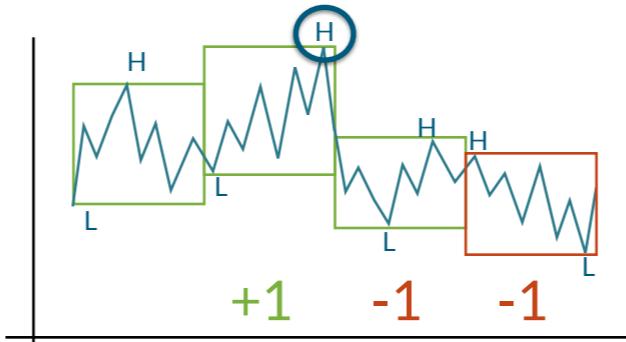
Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

Reversal finding rule:

If trend moves from **+1 to -1**, previous high is reversal.
If trend moves from **-1 to +1**, previous low is reversal.



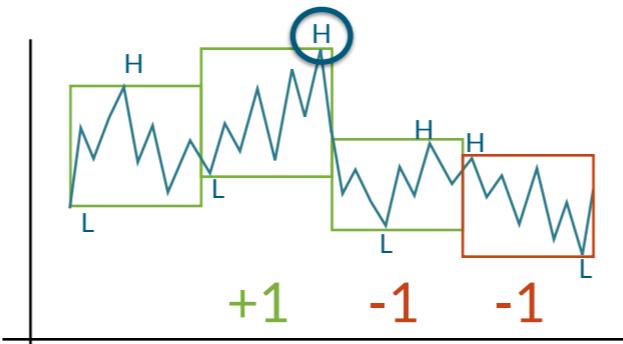
Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

Reversal outputs

Using these simple rules we output a new time series that includes just the reversal points found on this scale, N. Output is the dated value found: Date, Value

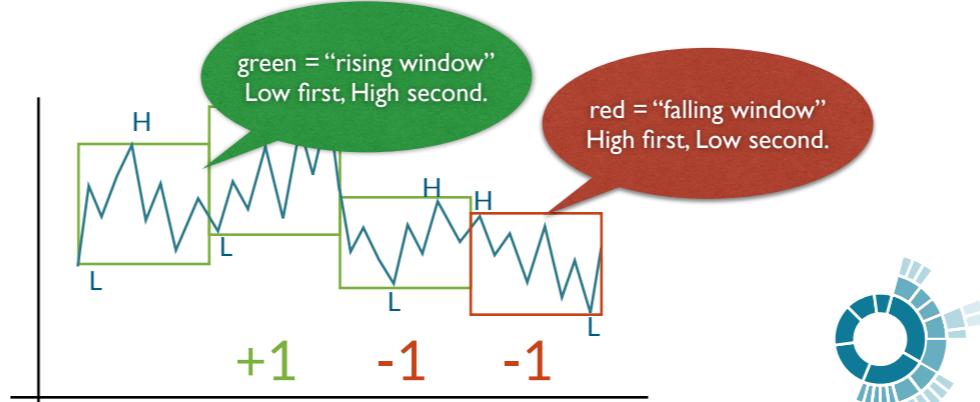


Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

just a note, on charting the windows:

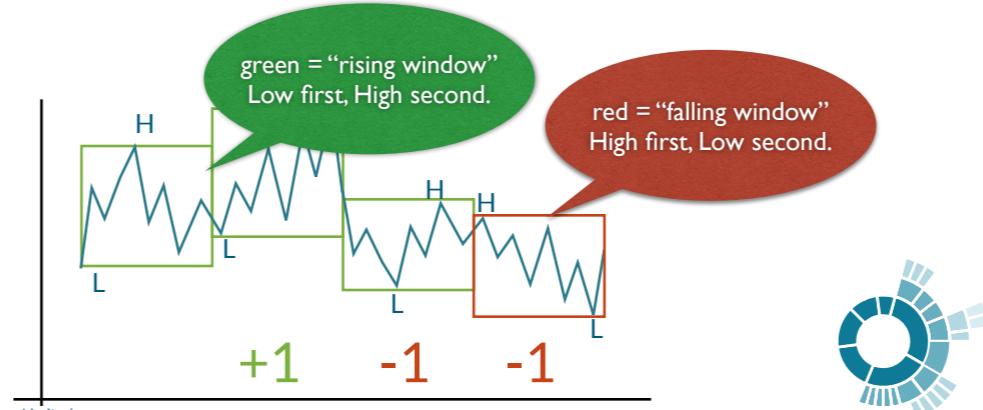


Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

How should we summarise our windows?



Copyright © 2015 ByteSumo Limited
All rights reserved.



Studying Trends

I create a “FHLS bar” data structure

The FHLS data format is:

1. Time Series Id
2. OpenDate (of bar)
3. First of H/L ← I replace the Open with the value of which ever came FIRST, the high or the low.
4. High
5. Low
6. Second of H/L ← I replace the Close with the value of which ever came SECOND, the high or the low.
7. High Date
8. Low Date
9. First Date
10. Second Date
11. CloseDate



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

I create a “FHLS bar” data structure

The FHLS data format is:

1. Time Series Id
2. Open Date
3. First of H/L
4. High
5. Low
6. Second of H/L
7. High Date
8. Low Date
9. First Date
10. Second Date
11. Close Date

This format may look very simple, but was constructed carefully.

The dated first and second values are very important to our calculations downstream.



Studying Trends

We can draw charts this way, a FHLS chart:



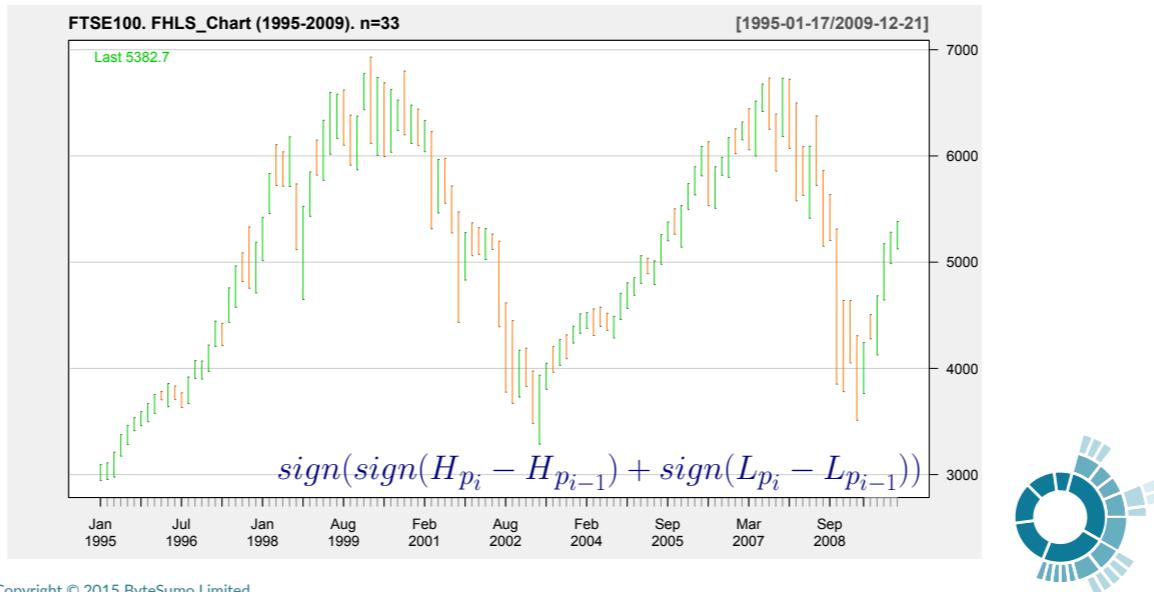
Studying Trends

We can draw charts this way, a FHLS chart:



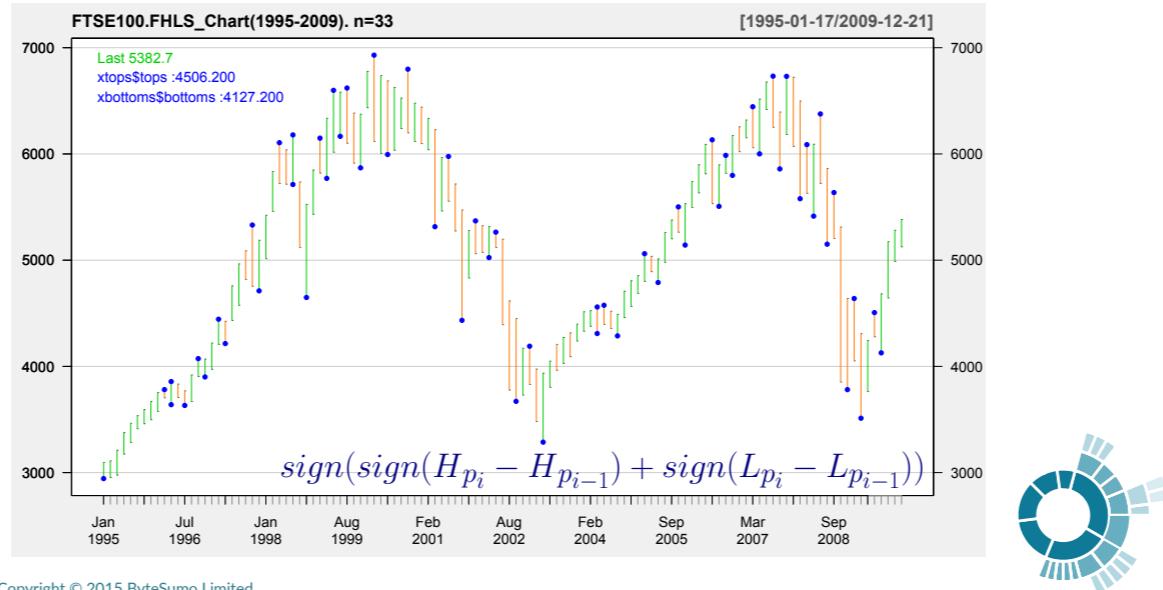
Studying Trends

We draw reversals as dots, on the FHLS chart.



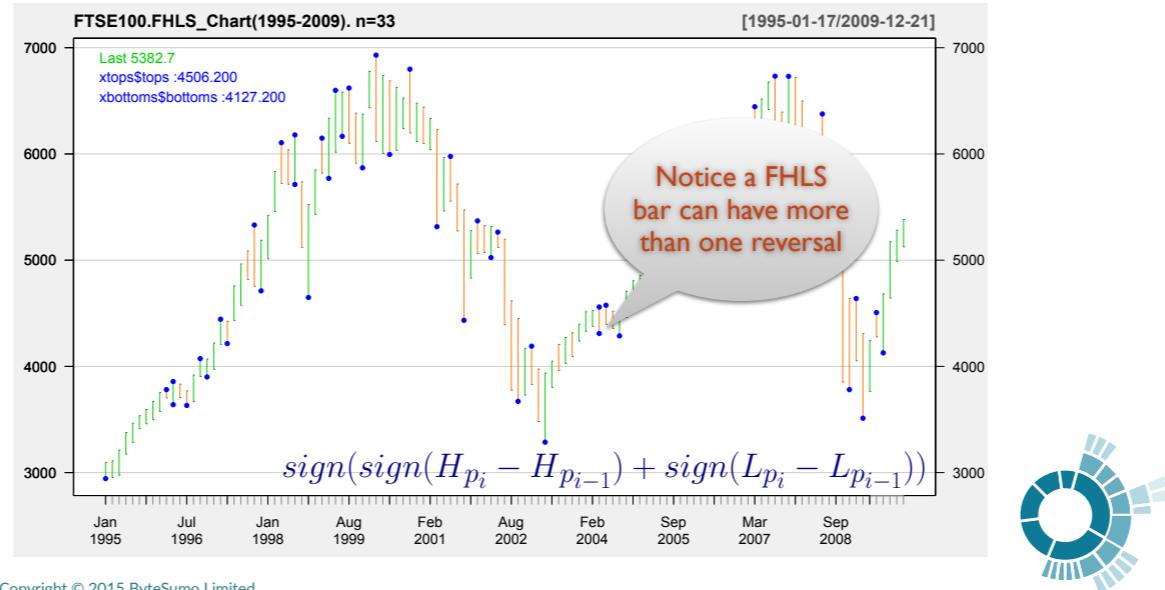
Studying Trends

We draw reversals as dots, on the FHLS chart.



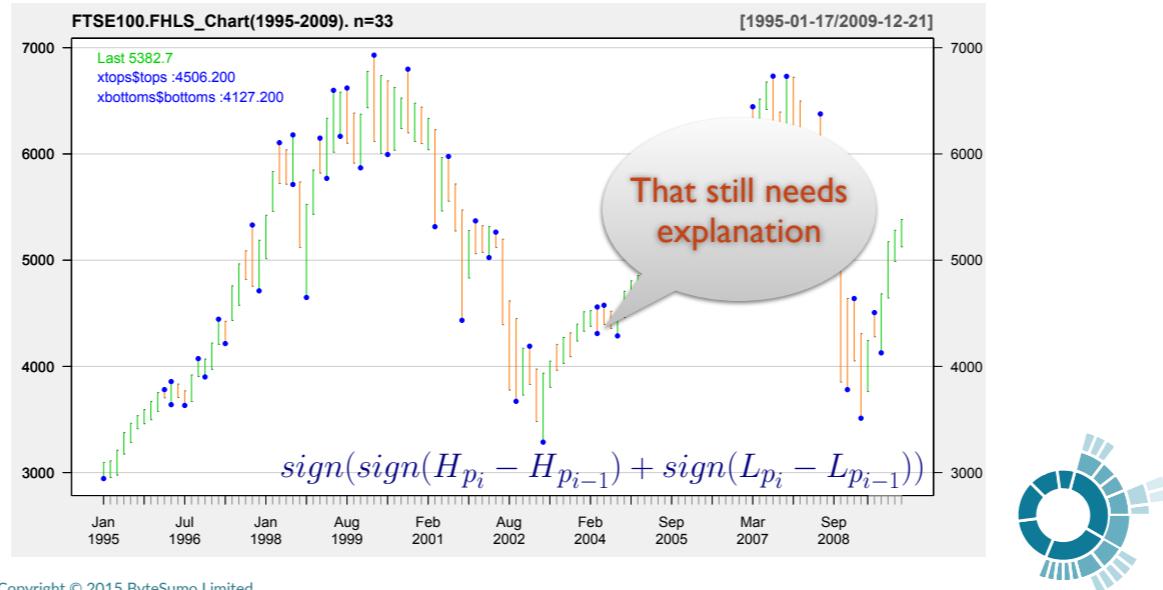
Studying Trends

We draw reversals as dots, on the FHLS chart.



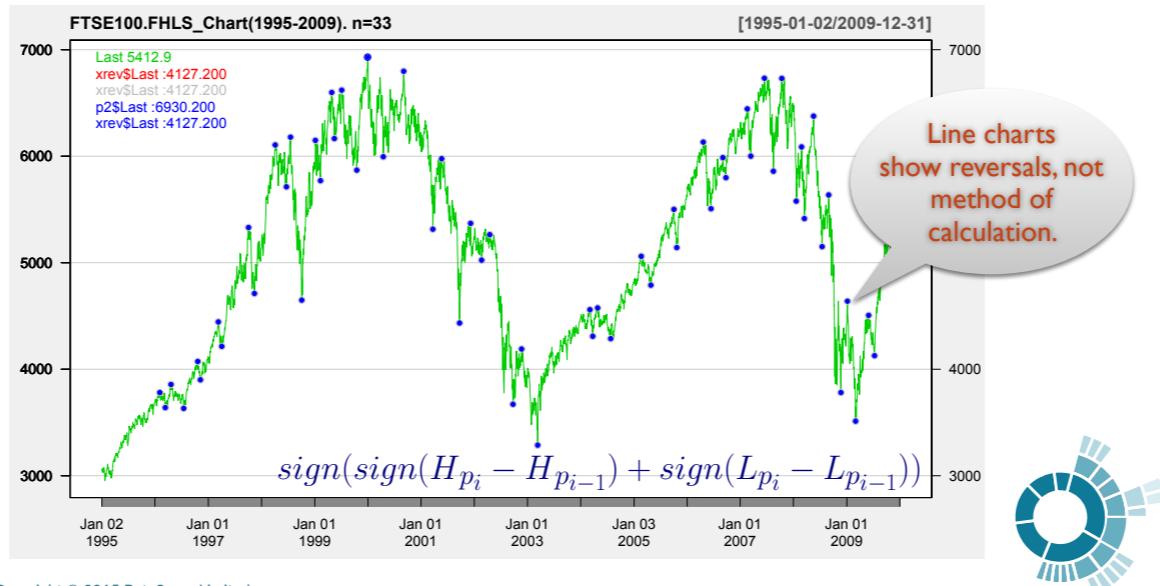
Studying Trends

We draw reversals as dots, on the FHLS chart.



Studying Trends

We draw reversals as dots, on the FHLS chart.



Special Cases

I skirted over some special cases to handle

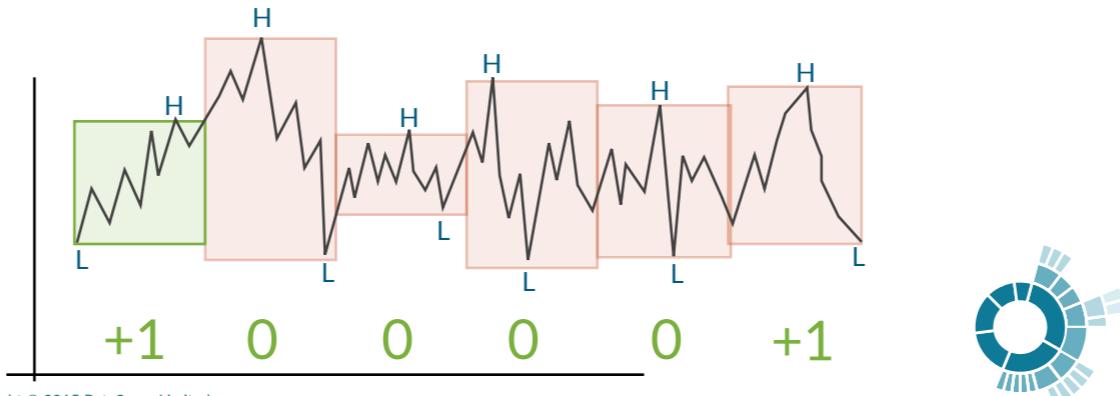


Copyright © 2015 ByteSumo Limited
All rights reserved.

Special Cases

The equation also finds **0** values.

$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$

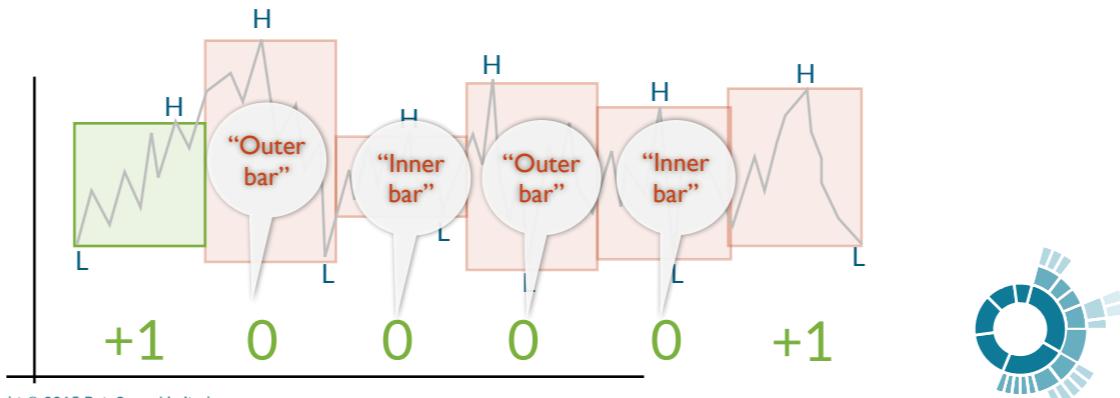


Special Cases

The equation also finds 0 values.

$$\text{sign}(\text{sign}(H_{p_i} - H_{p_{i-1}}) + \text{sign}(L_{p_i} - L_{p_{i-1}}))$$

"Inner" "Outer" is trader language for these cases.

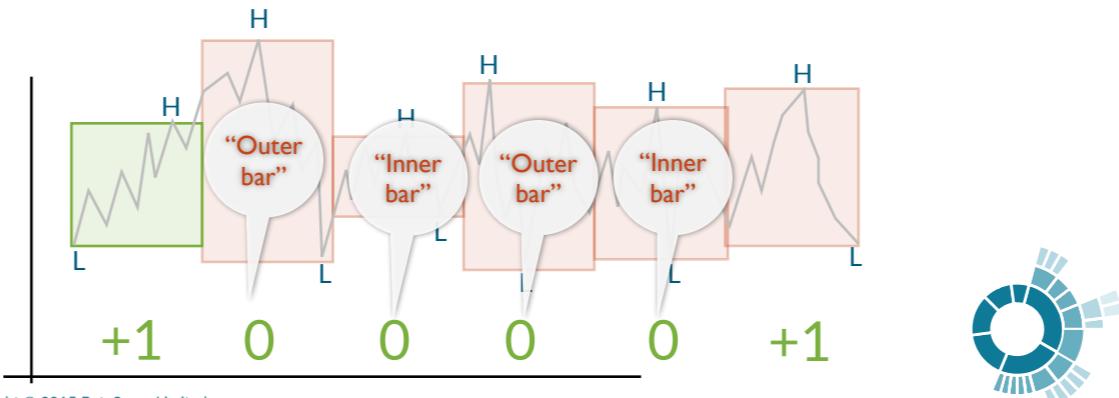


Copyright © 2015 ByteSumo Limited
All rights reserved.

Special Cases

The equation also finds 0 values.

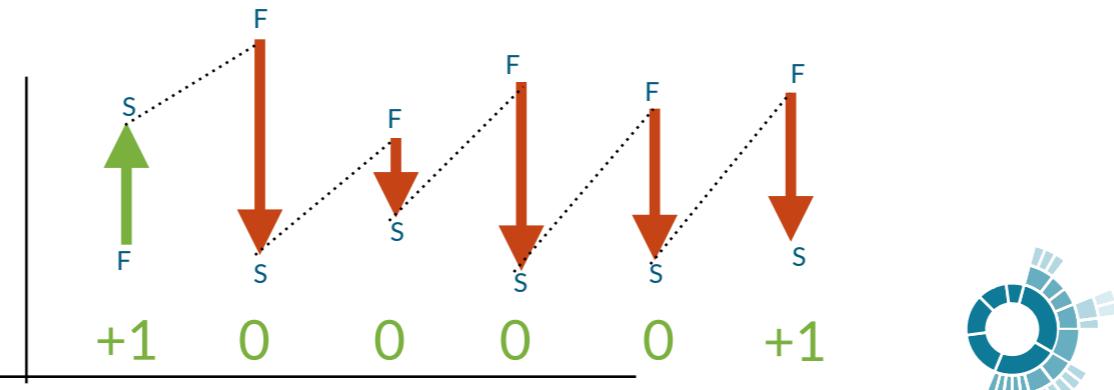
This sort of breaks our algo, no?



Special Cases

The equation also finds **0** values.

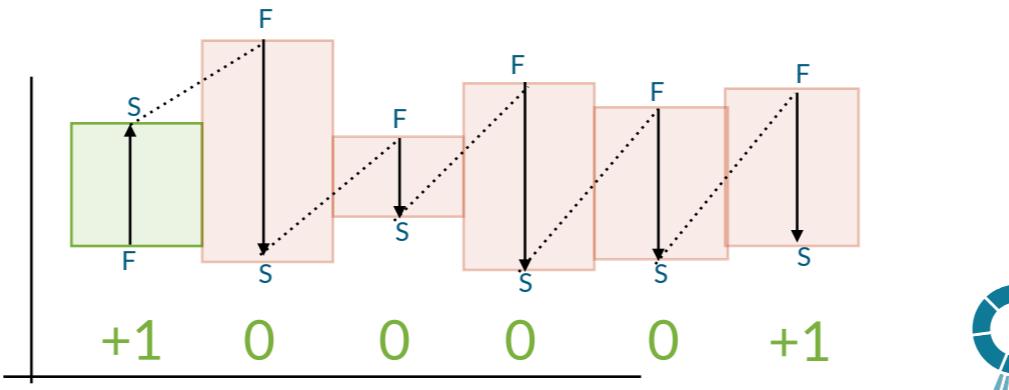
Nope, there is an efficient solution...



Special Cases

The equation also finds **0** values.

Nope, there is an efficient solution...



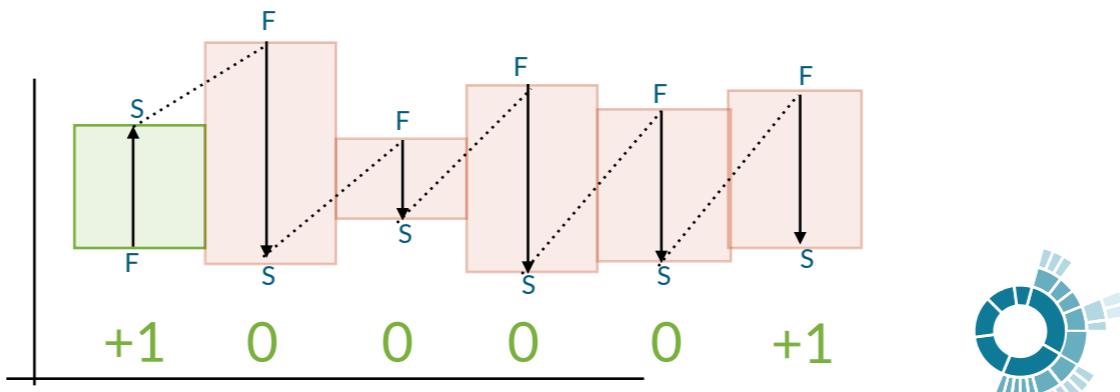
Copyright © 2015 ByteSumo Limited
All rights reserved.



Special Cases

The equation also finds **0** values.

Our eyes imagine the path taken across FHLS bars.
But these points are all dated in time... hence we have First
and Second values in our summary to indicate the "flow"



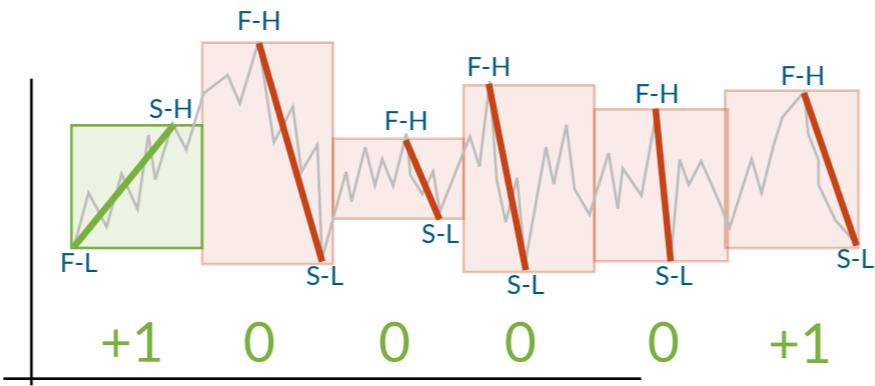
Copyright © 2015 ByteSumo Limited
All rights reserved.



Special Cases

The equation also finds 0 values.

... and that dotted line is in fact really there
not just implied.



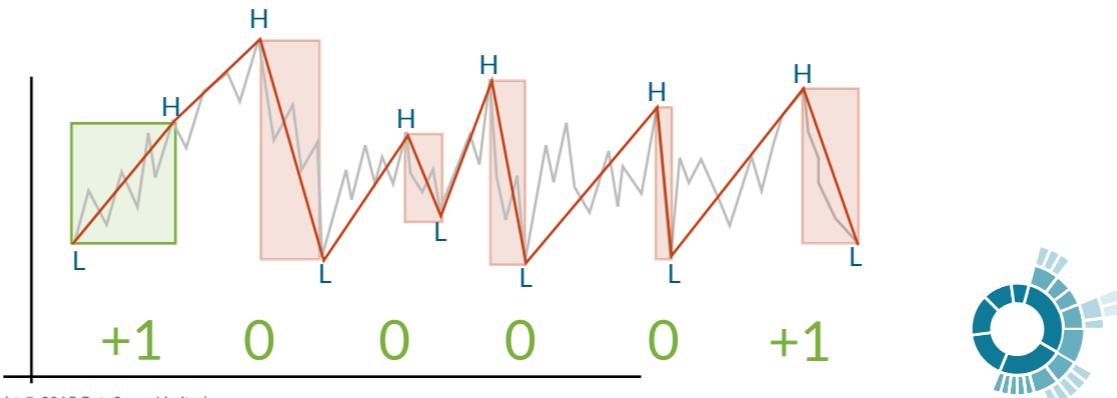
Copyright © 2015 ByteSumo Limited
All rights reserved.



Special Cases

The equation also finds **0** values.

the dated highs / lows have gaps between them!

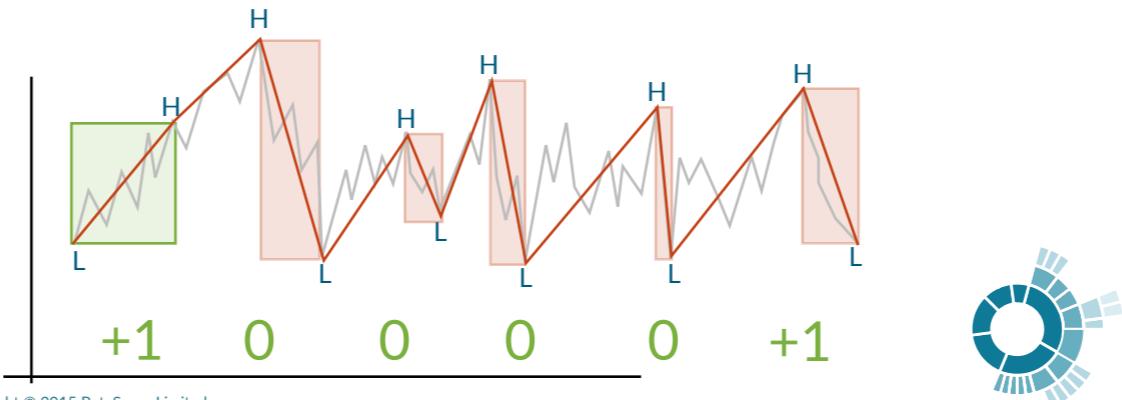


Copyright © 2015 ByteSumo Limited
All rights reserved.

Special Cases

The equation also finds **0** values.

Let's make use of this property for special cases,
for handling those **0** values.



Copyright © 2015 ByteSumo Limited
All rights reserved.



Special Cases

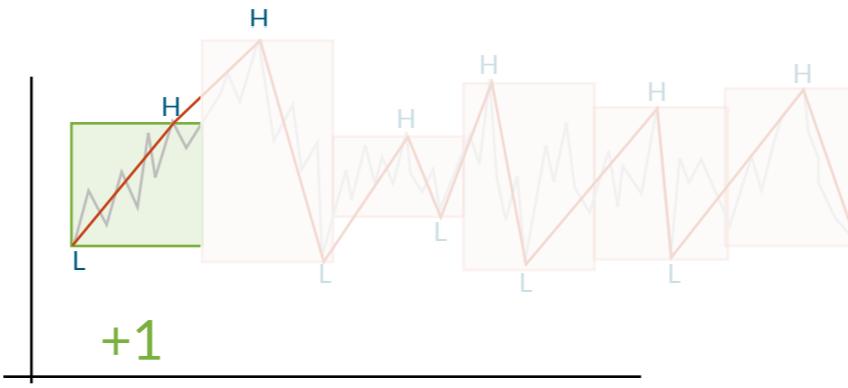
I'll illustrate, from the beginning:



Copyright © 2015 ByteSumo Limited
All rights reserved.

Special Cases

I'll illustrate, from the beginning:

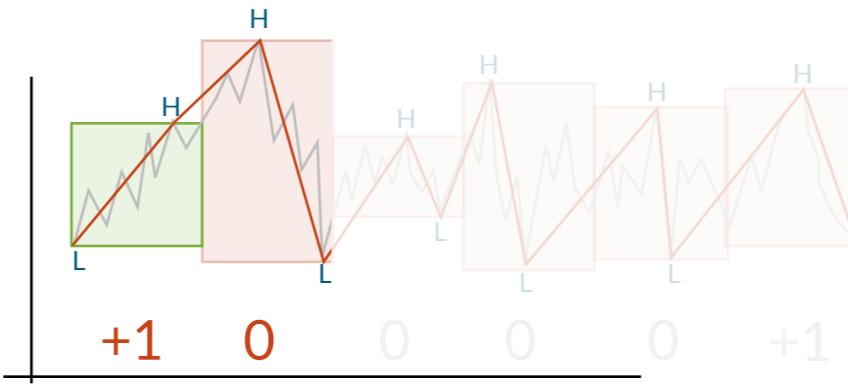


Copyright © 2015 ByteSumo Limited
All rights reserved.



Special Cases

Find a 0 Trend (inner / outer bar)

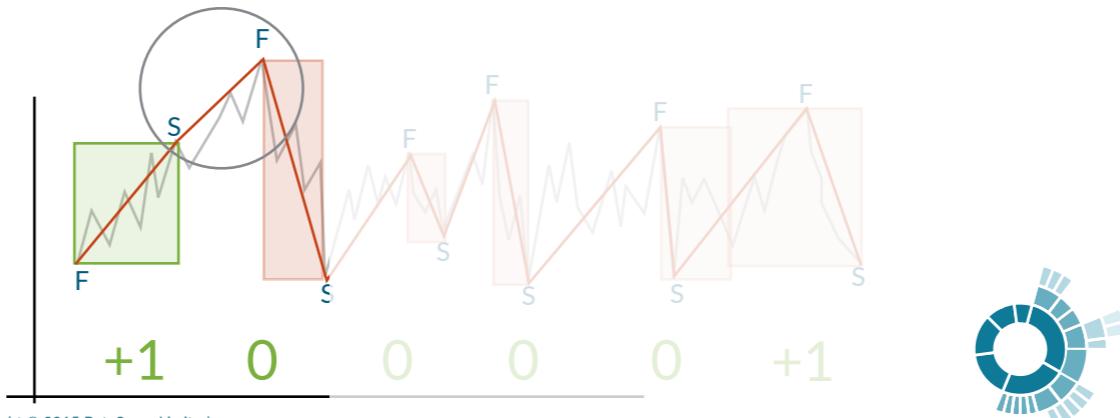


Copyright © 2015 ByteSumo Limited
All rights reserved.



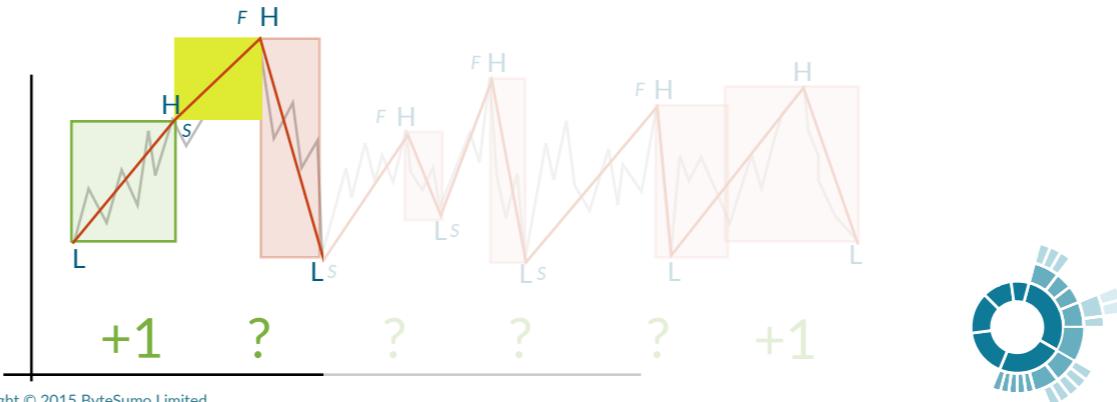
Special Cases

insert a new FHLS summary for the gap implied, by “borrowing” adjacent Second/First values.



Special Cases

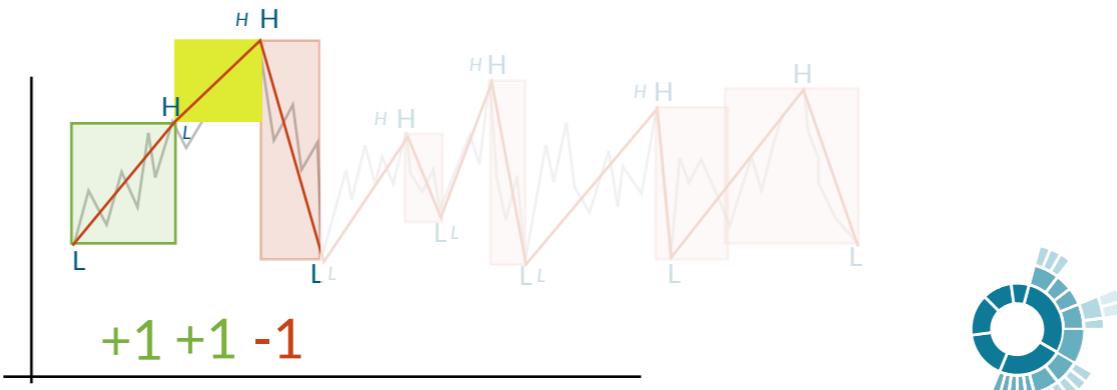
insert a new FHLS summary for the gap implied,
by “borrowing” adjacent Second/First values.
(ie, of “Second” of previous window and “First” of current window)



Copyright © 2015 ByteSumo Limited
All rights reserved.

Special Cases

Emit these “special bars” during FHLS construction,
format them like a regular window of highs/lows. Use
them to find the trend changes



Copyright © 2015 ByteSumo Limited
All rights reserved.

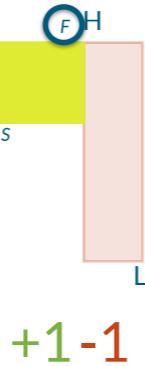
Special Cases

Then ... compare the current bar against the “split bar”:

One of the signs (hdif or ldif) will have a value of zero
but the other will be a +1 or -1, so we never get a zero.

Then find reversals as usual...

i.e. we see here we find a lower low, so the last high is reversal.

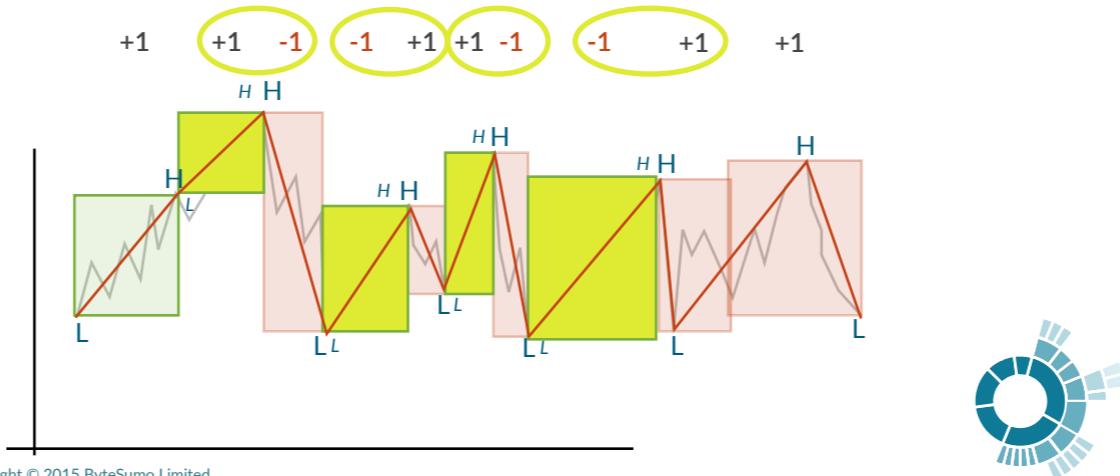


Copyright © 2015 ByteSumo Limited
All rights reserved.



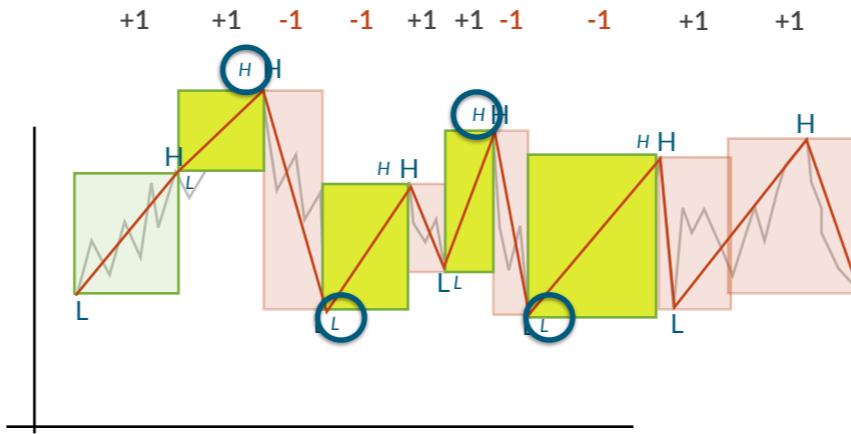
Special Cases

continue to find trends over this modified stream.



Special Cases

we're not seeing zeros now...

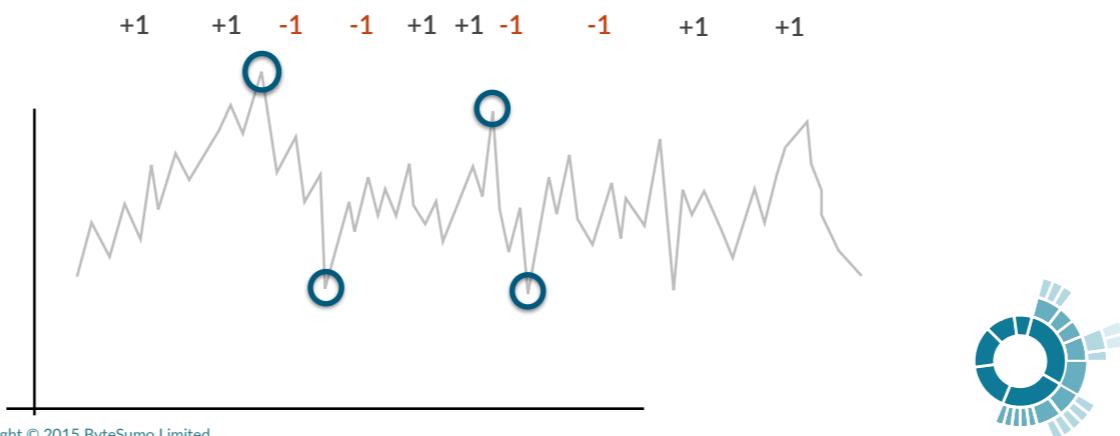


Copyright © 2015 ByteSumo Limited
All rights reserved.



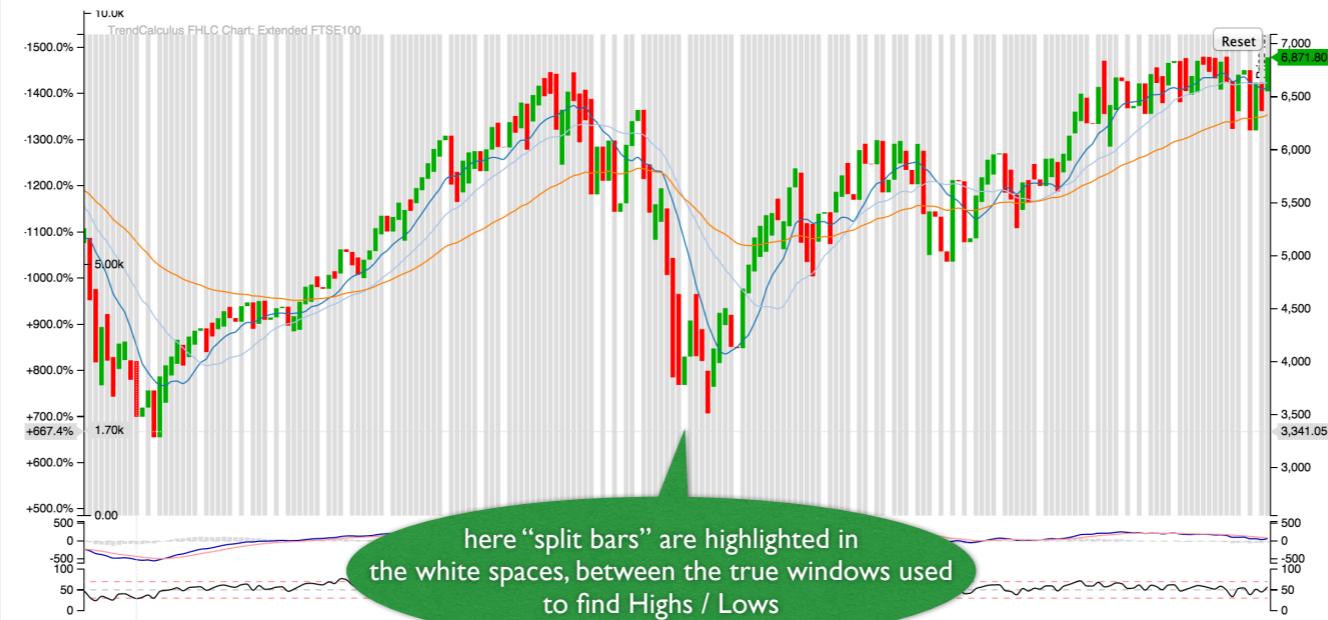
Special Cases

and when we find reversals, the results are good, and **Very** fast to compute.



FHLS data charted

FHLS chart, with split bars in white spaces to solve Inner/Outer cases



FHLS data charted

The results are so intuitive you can “eyeball” the results.



FHLS data charted

The results are so intuitive you can “eyeball” the reversals.



Stackable Processing

The final output is a stream of trend reversals

The output is a time series of trend *change points*.

You can run this output through the algo again, to “stack” TrendCalculus. This creates a multi-scale reversal finder. Processing data in several passes is a highly efficient process due to the data reduction on later passes.

In this case, N is not fixed windows of time, but a number of change points to summarise over.



Copyright © 2015 ByteSumo Limited
All rights reserved.

Stackable Processing

This is what “stacking” means:

```
time cat db_output.csv | luajit trendcalculus.lua -F "," -OFS "," -p 1 -h -H -n $z -r \
| tee out/rev01.csv | luajit trendcalculus.lua -F "," -OFS "," -p 2 -h -H -n $p -r \
| tee out/rev02.csv | luajit trendcalculus.lua -F "," -OFS "," -p 3 -h -H -n $p -r \
| tee out/rev03.csv | luajit trendcalculus.lua -F "," -OFS "," -p 4 -h -H -n $p -r \
| tee out/rev04.csv | luajit trendcalculus.lua -F "," -OFS "," -p 5 -h -H -n $p -r \
| tee out/rev05.csv | luajit trendcalculus.lua -F "," -OFS "," -p 6 -h -H -n $p -r \
| tee out/rev06.csv | luajit trendcalculus.lua -F "," -OFS "," -p 7 -h -H -n $p -r \
| tee out/rev07.csv | luajit trendcalculus.lua -F "," -OFS "," -p 8 -h -H -n $p -r \
| tee out/rev08.csv | luajit trendcalculus.lua -F "," -OFS "," -p 9 -h -H -n $p -r \
| tee out/rev09.csv | luajit trendcalculus.lua -F "," -OFS "," -p 10 -h -H -n $p -r \
| tee out/rev10.csv | luajit trendcalculus.lua -F "," -OFS "," -p 11 -h -H -n $p -r \
| tee out/rev11.csv | luajit trendcalculus.lua -F "," -OFS "," -p 12 -h -H -n $p -r \
| tee out/rev12.csv | luajit trendcalculus.lua -F "," -OFS "," -p 13 -h -H -n $p -r \
| tee out/rev13.csv | luajit trendcalculus.lua -F "," -OFS "," -p 14 -h -H -n $p -r > out/rev14.csv
```

```
Andrews-MacBook-Pro-2:trendcalculus-public andrewmorgan$ wc -l out/*.csv
33936 out/db_output.csv
3625 out/rev01.csv
1247 out/rev02.csv
475 out/rev03.csv
166 out/rev04.csv
59 out/rev05.csv
26 out/rev06.csv
10 out/rev07.csv
4 out/rev08.csv
1 out/rev09.csv
1 out/rev10.csv
1 out/rev11.csv
1 out/rev12.csv
1 out/rev13.csv
1 out/rev14.csv
```

Copyright © 2015 ByteSumo Limited
All rights reserved.

14 stacks

We feed the output of each pass back into the algorithm. We set the -p option to indicate the “pass number” in output. We fix n to a fixed increment (n=3) in the stack. Each further run, p, additionally reduces noise and filters down the number of change points further. Below is a count of change points created.

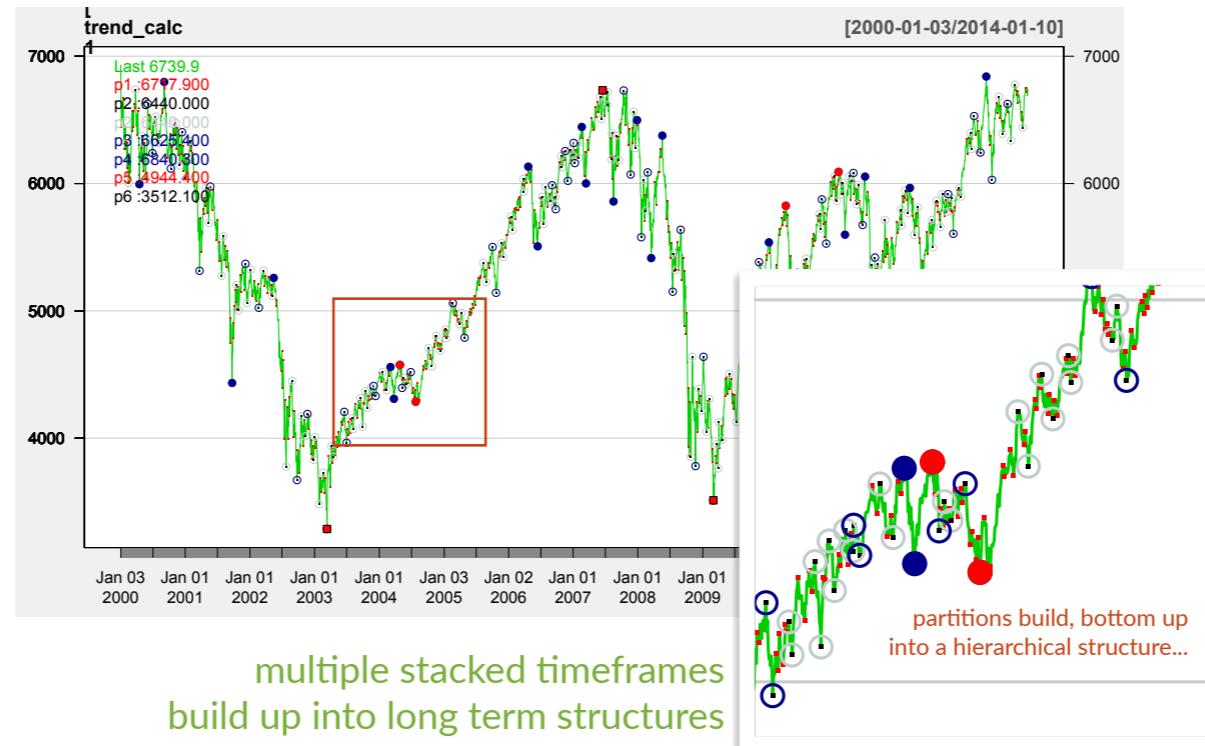
I fed in **33936** raw values, representing the daily close of the Dow Jones Industrial Average

The trend turning points on the first pass reduce the time series to **3625** rows. Subsequent passes summarise further.

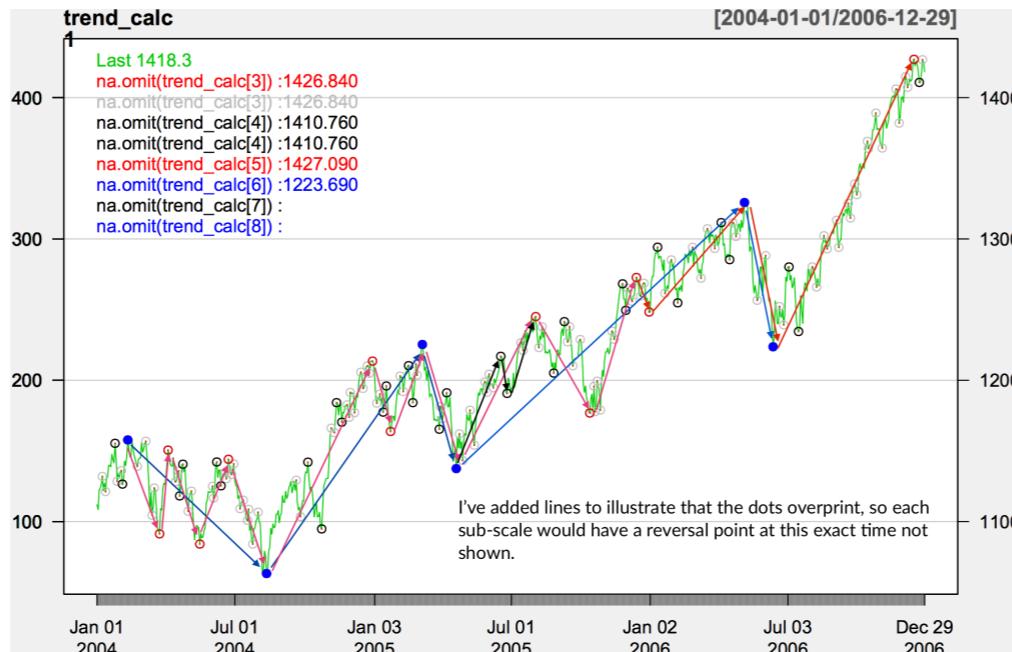
The summarisation rate looks geometric to me. What does this mean? ... apart from it being a fantastic way to compress the information and to index the partitions of time?



MultiScale Trends, Visualised



MultiScale Trends, Visualised



Importance of Multi-Scale

All branching and circulatory systems in nature are said to observe "Horton's Law of Stream Order"

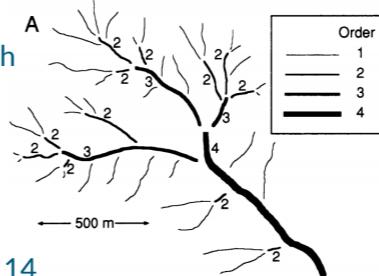
What about TrendCalculus induced multi-scale trend partitions?

Each pass of my algorithm, I simplify trends further. This "Pass number" is a "Stream Order" of the broader trend which branches into sequences of smaller trends on lower scales.

"Stream Order" frequency diminishes geometrically (says Horton). So, how about my stacked trend outputs?

My test script, (included in repo for trendcalculus) calculates 14 passes of the algorithm (aka Stream Orders) for the DJI index.

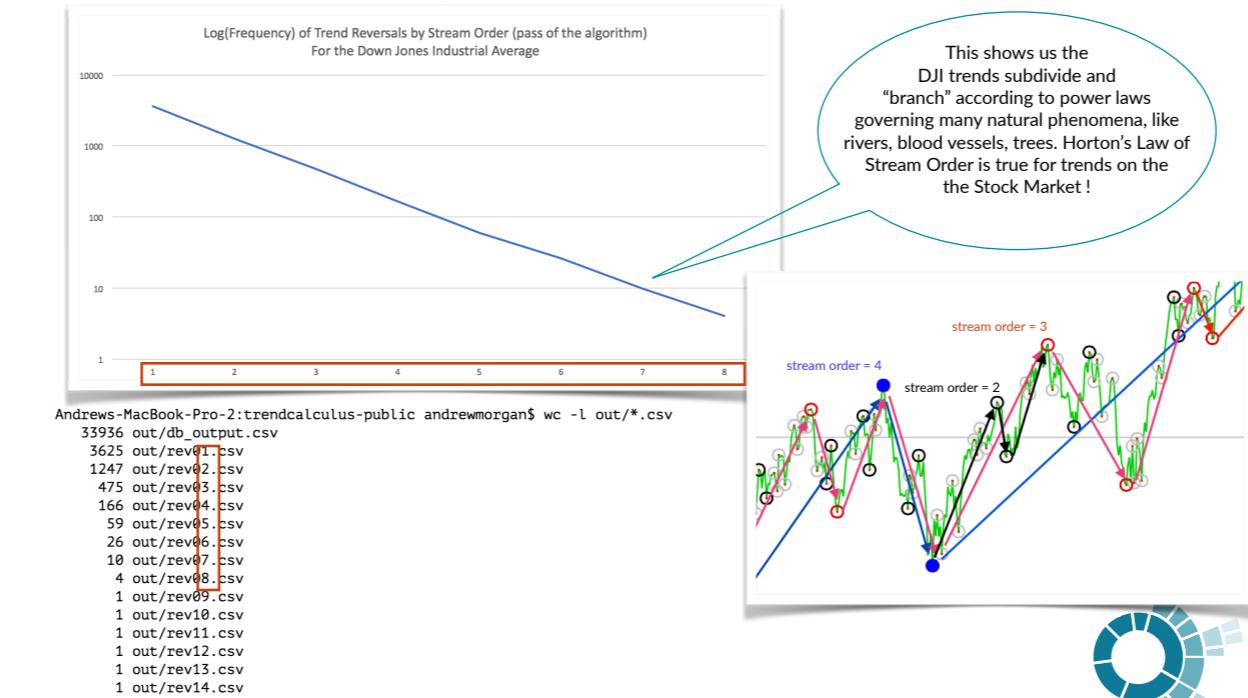
Does my output follow Horton's Law?
(Yes it does!)



Stream Order



Trend Branching: Horton's Law



Studying Trends

In summary, TrendCalculus:

Pros:

- stream based
- simple, fast
- visual, interpretable
- stackable, multi-scale (fractal?)

Cons:

- The trend reversal finding method is a lagging indicator.
- the lags accumulate for higher scales.



Copyright © 2015 ByteSumo Limited
All rights reserved.

Studying Trends

Now I have explained how it works:
to do:

Get implementation right.
Describe why we need it.
Do great things.



Copyright © 2015 ByteSumo Limited
All rights reserved.

5 minute break

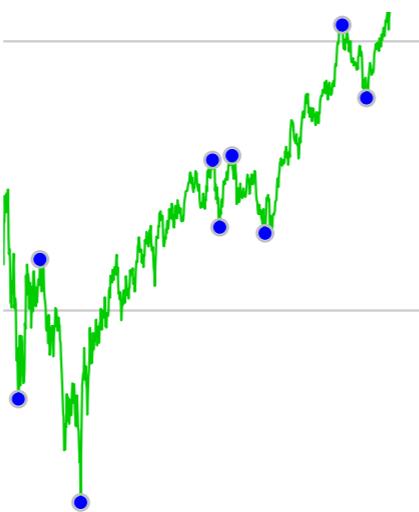
Predicting Trend Change

Experimental Approach

A Holy Grail of Predicting

Can we predict tops and bottoms in the stock market?

For example, predict turning points AS THEY HAPPEN?



This would seem to be
impossible right?

But putting that aside :-)
how might we design
experiments to at least try?

I will explain my approach to
an experimental design...
... show early results.



How would we start?

With a Hypothesis !

Hypothesis:

*With TrendCalculus generated trend reversals
data as training labels,
merged with realtime rolling trend “features”,
I can machine-learn realtime “predicted trend” (+1, -1)
over stock market indices data
that helps beats the “lag” inherent in straight TrendCalculus.*



Experimental Design

The goal demands designing great **supporting data**.

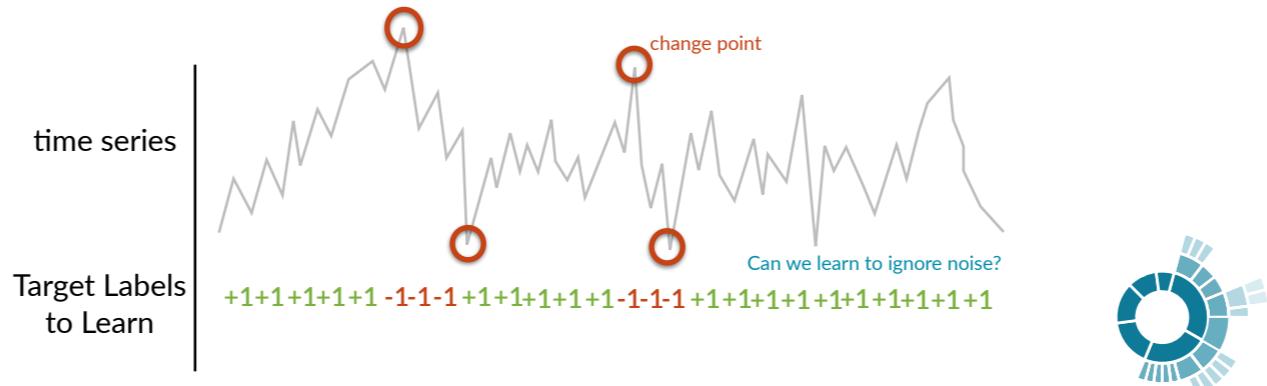
We need Labels! *We need great Features!*

Use trend calculus change points to build LABELS.
Use internal rolling trend calculus arrays to build FEATURES.



Experiment Approach

- a) find Trend change points for timeframe, n, as targets
 - b) “tag” my time-series with rolling trend signals, per “tick”
 - c) merge the rolling features against actual trend change points.
 - d) try and machine learn the trend labels, at tick level.



Introducing “TCFG”

a “TrendCalculus Feature Generator”

(1) “Rewind” and “Hindsight” produces training targets:



Introducing “TCFG”

a “TrendCalculus Feature Generator”

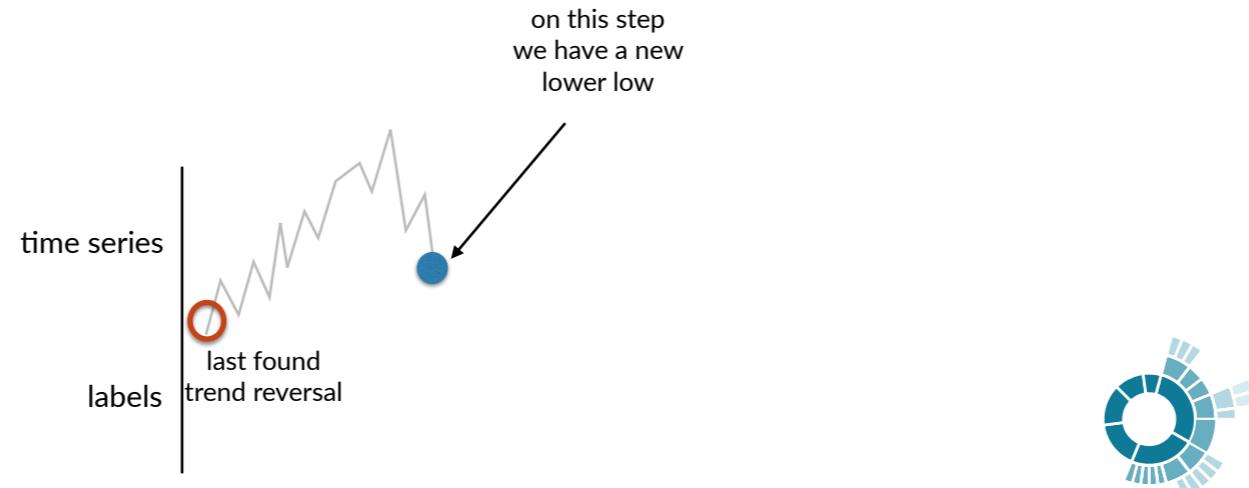
(1) “Rewind” and “Hindsight” produces training targets:



Introducing “TCFG”

“TrendCalculus Feature Generator”

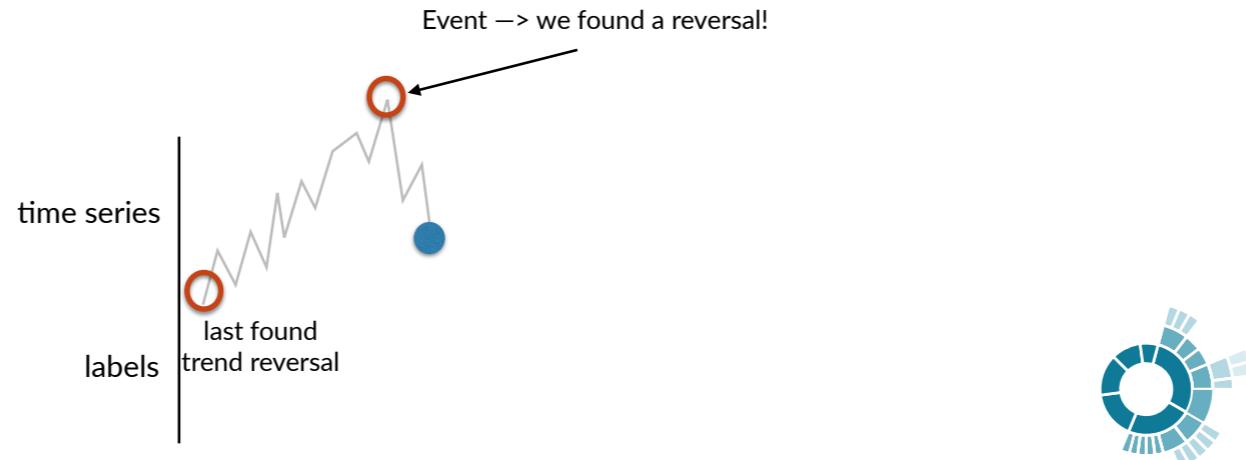
(1) “Rewind” and “Hindsight” produces training targets:



Introducing “TCFG”

“TrendCalculus Feature Generator”

(1) “Rewind” and “Hindsight” produces training targets:



Introducing “TCFG”

“TrendCalculus Feature Generator”

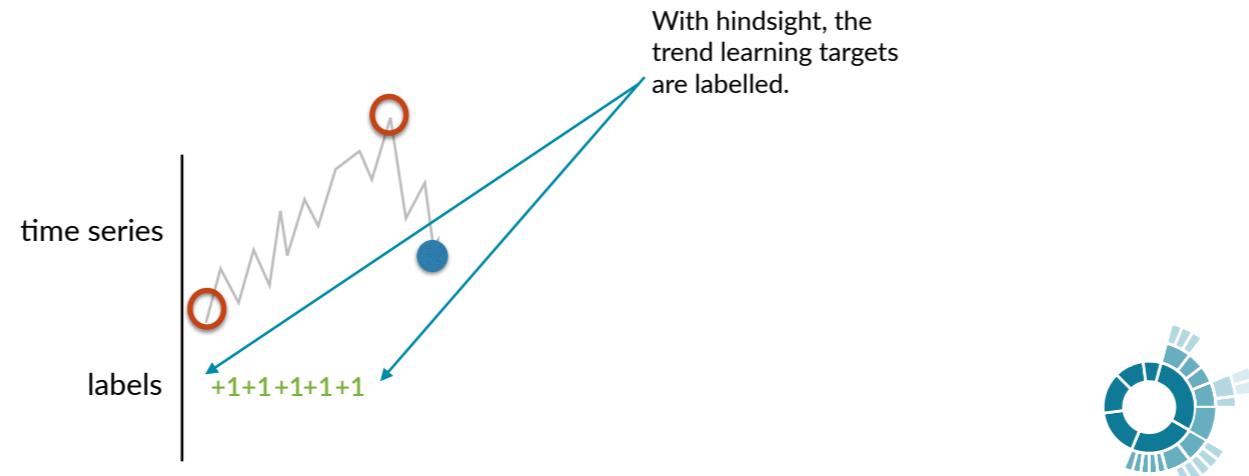
(1) “Rewind” and “Hindsight” produces training targets:



Introducing “TCFG”

“TrendCalculus Feature Generator”

(1) “Rewind” and “Hindsight” produces training targets:

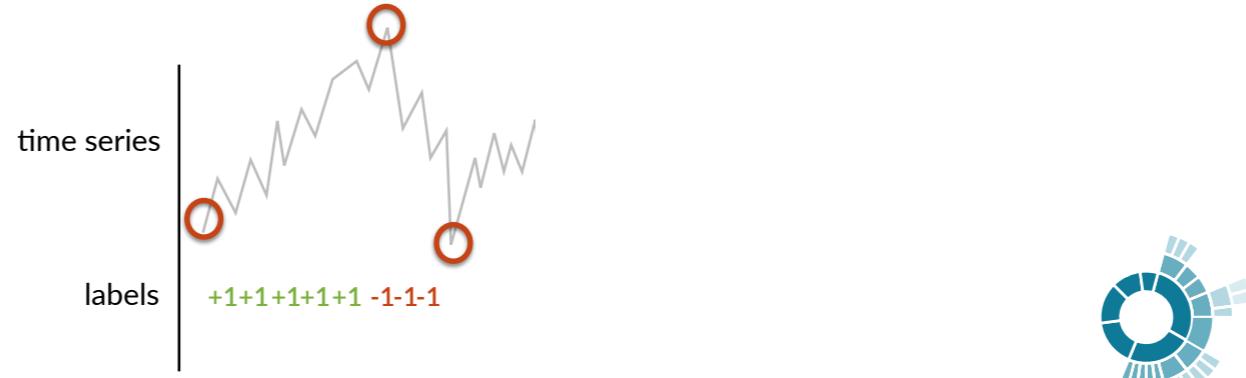


Introducing “TCFG”

“TrendCalculus Feature Generator”

(1) “Rewind” and “Hindsight” produces training targets:

and we continue to label through time... as a lagging labeller

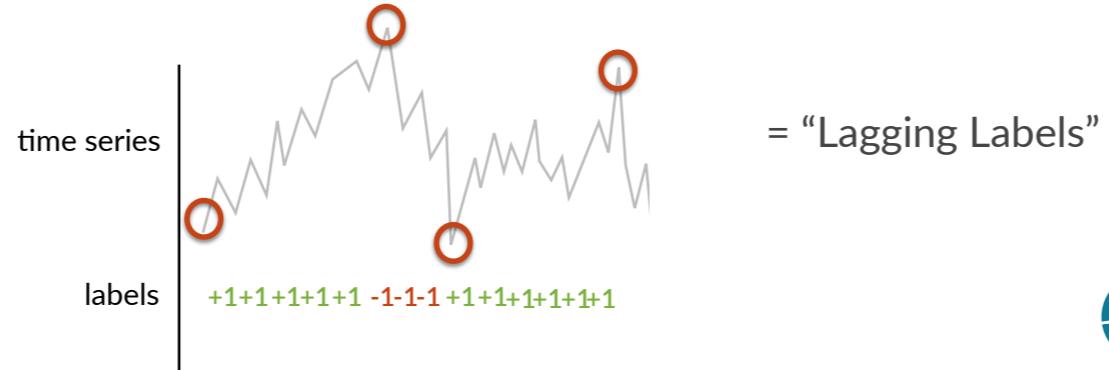


Introducing “TCFG”

“TrendCalculus Feature Generator”

(1) “Rewind” and “Hindsight” produces training targets:

And we populate our training target labels in this way
in an incremental batch process as reversals are found.
So – how about features?

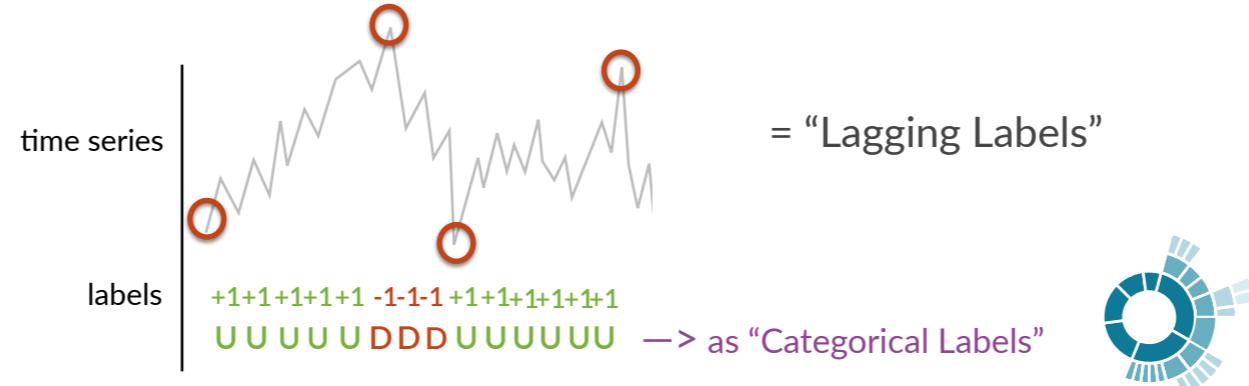


Introducing “TCFG”

“TrendCalculus Feature Generator”

(1) “Rewind” and “Hindsight” produces training targets:

And we populate our training target labels in this way
in a batch process as reversals are found.
So – how about features?

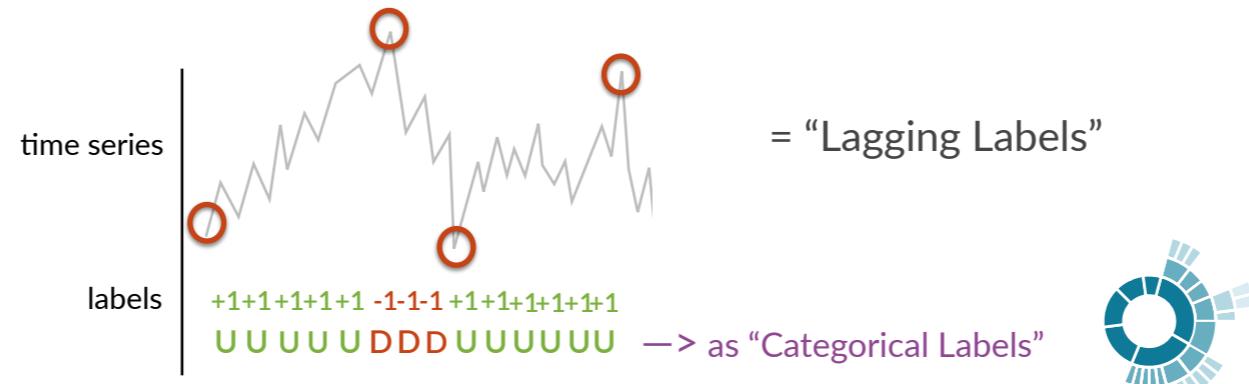


Introducing “TCFG”

“TrendCalculus Feature Generator” - feature builder code

(1) “Rewind” and “Hindsight” produces training targets:

Why a lagging labeller?
Incremental training can be organised over time.
The model can then be updated as new trends end



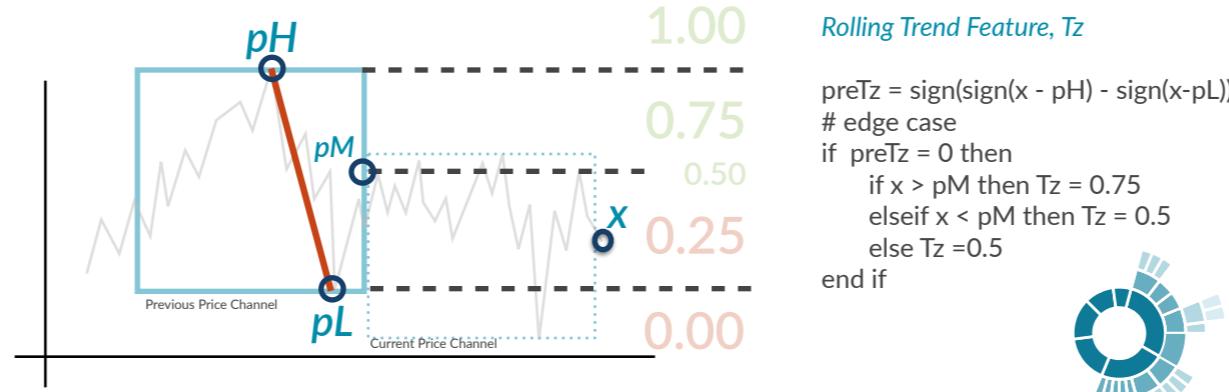
Introducing “TCFG”

a “TrendCalculus Feature Generator”

(2) Let's merge Labels with normalised Features.

What might they be?

I'm trialling a simple **rolling trend indicator** over all timeframes:

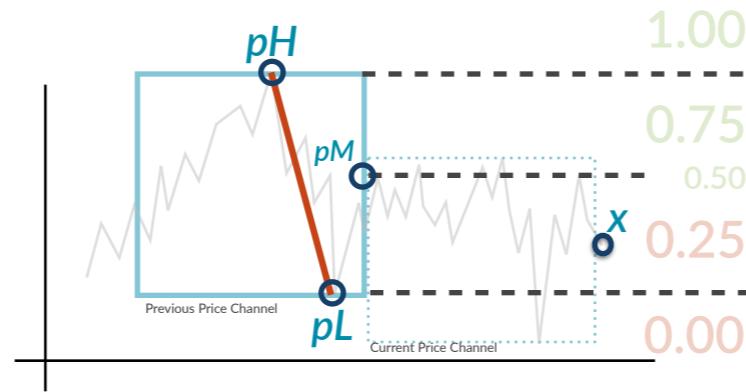


Introducing “TCFG”

Using this feature to build “Trendmaps”, I’m inspecting trends on all timeframes visually. They show Tz on ALL timeframes from 1 to N. They are generated in real time and do not have look ahead bias.

They reveal rich multi-scale trend structures.

Lets see them...

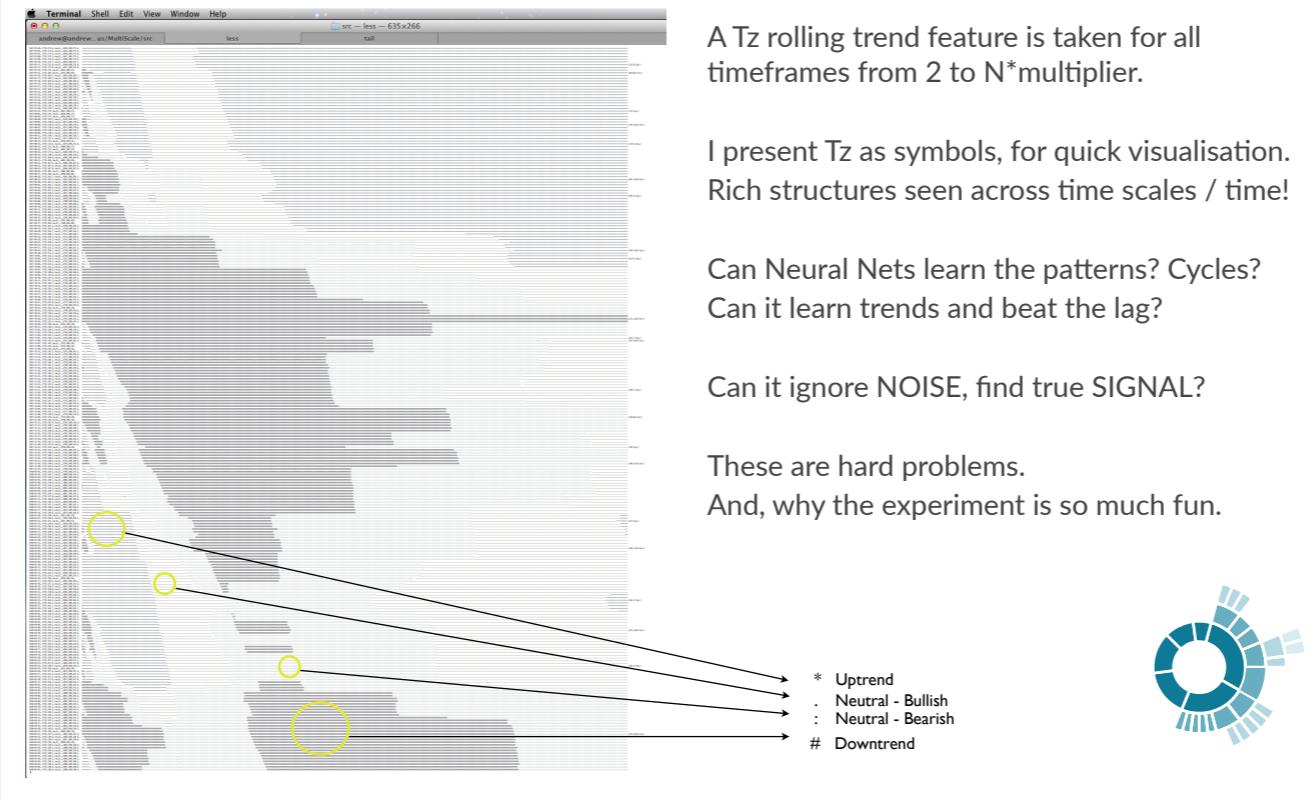


Rolling Trend Feature, Tz

```
preTz = sign(sign(x - pH) - sign(x-pL))
# edge case
if preTz = 0 then
    if x > pM then Tz = 0.75
    elseif x < pM then Tz = 0.5
    else Tz = 0.5
end if
```



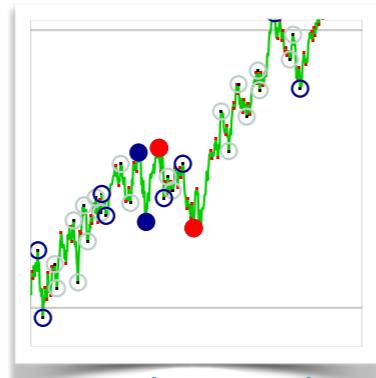
“TrendMap” features



Experiment Approach

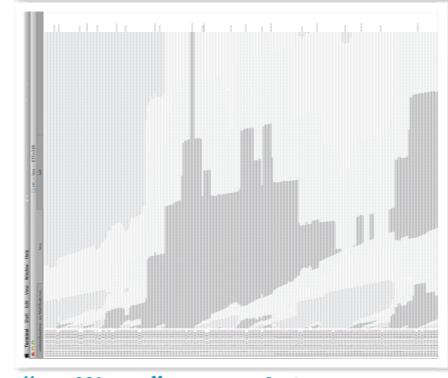
(2) Let's merge trend Labels with these trend Features.

Source of
Trend Labels



trend reversals

Source of
Trend Features



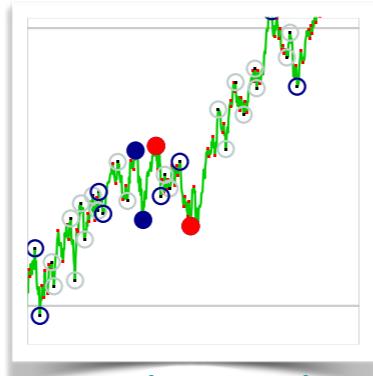
"rolling" trend features



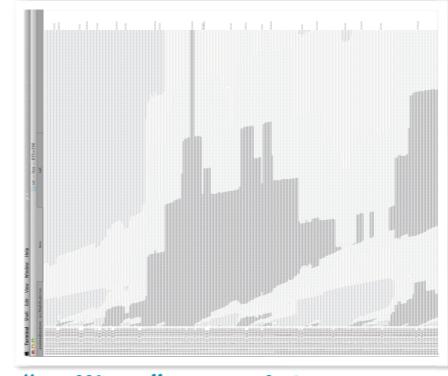
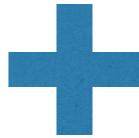
Experiment Approach

(2) Let's merge trend Labels with these trend Features.

Create a machine learning dataset this way.
Objective is to predict Trend Reversals.



trend reversals



"rolling" trend features



Experiment Approach

(2) Let's merge trend Labels with simple trend Features.

Here's an example of labelled training data:

Row	TCFG	UserDate	Value	trend	change	dummy	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	
1	tcfg	BARC.L	1078841600000	496.0	D	-0.4613	.	0	0	0	0	0	0.2500	0.2500	0.7500	0.7500	0.7500	0.7500	0.7500	0.7500	
2	tcfg	BARC.L	1078928000000	495.0	D	-0.1065	.	0	0	0	0	0	0	0.2500	0.2500	0.2500	0.7500	0.7500	0.7500	0.7500	
3	tcfg	BARC.L	1071014400000	497.0	D	-0.6685	.	0	0	0	0	0	0	0.2500	0.2500	0.2500	0.2500	0.2500	0.2500	0.2500	
4	tcfg	BARC.L	1071108000000	479.2500	D	-0.5616	.	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	tcfg	BARC.L	1071107000000	476.2500	D	-0.3932	.	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	tcfg	BARC.L	1071146400000	480.2500	D	0.3969	.	1.0	0	0	0	0	0	0	0	0	0	0	0	0	
7	tcfg	BARC.L	1071532000000	476.0	D	-0.4157	.	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	tcfg	BARC.L	1071619200000	487.7500	U	0.8428	.	1.0	1.0	0.7500	0.7500	0	0	0	0	0.2500	0.2500	0.2500	0.2500	0.2500	
9	tcfg	BARC.L	1071709600000	491.2500	U	0.3441	.	1.0	1.0	0.7500	0.7500	0.2500	0	0	0	0	0.2500	0.2500	0.2500	0.2500	
10	tcfg	BARC.L	1071792000000	494.0	U	0.2728	.	1.0	1.0	0.8	0.7500	0.7500	0.2500	0	0	0	0.2500	0.2500	0.2500	0.2500	
11	tcfg	BARC.L	1072051200000	492.7500	U	-0.1258	.	1.0	1.0	1.0	0.7500	0.7500	0.2500	0	0	0	0.2500	0.2500	0.2500	0.2500	
12	tcfg	BARC.L	1072137600000	490.0	U	-0.3728	.	0	0	0.7500	1.0	0	0	0	0	0	0	0	0	0	
13	tcfg	BARC.L	1072224000000	491.2500	U	0.1269	.	0	0	1.0	1.0	1.0	0.7500	0.7500	0.2500	0	0	0	0.2500	0.2500	0.2500
14	tcfg	BARC.L	1072310400000	491.2500	U	0	.	1.0	0	0.7500	0	0	0.7500	1.0	1.0	0.7500	0.7500	0.2500	0.2500	0.2500	
15	tcfg	BARC.L	1072309600000	491.2500	U	0	.	0	0.5000	0.2500	0	0.7500	0.7500	0.2500	0.2500	0.2500	0.2500	0.2500	0.2500		
16	tcfg	BARC.L	1072356000000	492.0	U	0.1763	.	1.0	1.0	0.7500	0.7500	1.0	0.7500	0.7500	0.2500	0.2500	0.2500	0.2500	0.2500		
17	tcfg	BARC.L	1072712400000	496.2500	U	0.3182	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500	0.7500	0.5000	0.2500	0.2500	0.2500	
18	tcfg	BARC.L	1072828000000	498.2500	U	0.1988	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500	0.7500	0.7500	0.7500	0.2500		
19	tcfg	BARC.L	1072915200000	498.2500	U	0	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500	0.7500	0.7500	0.7500	0.7500		
20	tcfg	BARC.L	1073001600000	504.0	U	0.5285	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500		
21	tcfg	BARC.L	1073208000000	506.0	U	0.1958	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500		
22	tcfg	BARC.L	1073347200000	513.0	U	0.5991	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500		
23	tcfg	BARC.L	1073433800000	518.7500	U	-0.2158	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500		
24	tcfg	BARC.L	1073520000000	518.5000	U	-0.0245	.	0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7500		
25	tcfg	BARC.L	1073606400000	518.5000	U	0.3390	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
26	tcfg	BARC.L	1073655600000	508.5000	U	-0.4892	.	0	0	0.5000	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
27	tcfg	BARC.L	1073952000000	511.2500	U	0.2646	.	0	1.0	0.7500	0.7500	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
28	tcfg	BARC.L	1074038400000	521.5000	U	0.7626	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
29	tcfg	BARC.L	1074124000000	527.7500	U	0.5365	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
30	tcfg	BARC.L	1074211200000	536.0	U	0.6536	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
31	tcfg	BARC.L	1074478400000	538.7500	D	-0.4548	.	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
32	tcfg	BARC.L	1074556800000	528.7500	D	-0.7362	.	0	0	0.7500	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
33	tcfg	BARC.L	1074643200000	517.5000	D	-0.3623	.	0	0	0.2500	0.7500	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
34	tcfg	BARC.L	1074729600000	517.0	D	-0.8483	.	0	0	0	0.2500	0.7500	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
35	tcfg	BARC.L	1074816000000	567.5000	D	-0.7253	.	0	0	0	0	0	0	0.2500	0.7500	0.7500	0.7500	0.7500	1.0	1.0	
36	tcfg	BARC.L	1075075200000	568.0	D	0.8492	.	0	0	0	0	0	0	0	0.2500	0.7500	0.7500	0.7500	0.7500	1.0	
37	tcfg	BARC.L	1075161600000	566.0	D	-0.6569	.	0	0	0	0	0	0	0	0	0.2500	0.2500	0.2500	0.2500	0.2500	
38	tcfg	BARC.L	1075248000000	511.0	D	0.8005	.	1.0	0.2500	0	0	0	0.2500	0.2500	0.2500	0.7500	0.7500	0.7500	0.7500		



Experiment Approach

(2) Let's merge trend Labels with trend Features.

Row	TCFG	UserDate	Value	trend	change	dummy	T1	T2	T3
1	tcfg	BARC.L	1078841600000	496.0	D	-0.4613	.	0	0
2	tcfg	BARC.L	1078928000000	495.0	D	-0.1005	.	0	0
3	tcfg	BARC.L	1071014400000	487.0	D	-0.6685	.	0	0
4	tcfg	BARC.L	1071108000000	478.2500	D	-0.6516	.	0	0
5	tcfg	BARC.L	1071147900000	476.2500	D	-0.3952	.	0	0
6	tcfg	BARC.L	1071446400000	480.2500	D	0.3969	.	1.0	0
7	tcfg	BARC.L	1071532800000	476.0	D	-0.4157	.	0	0
8	tcfg	BARC.L	1071619200000	487.7500	U	0.8428	.	1.0	0.7500
9	tcfg	BARC.L	1071705600000	491.2500	U	0.3441	.	1.0	1.0
10	tcfg	BARC.L	1071792000000	484.0	U	0.2728	.	1.0	0.8
11	tcfg	BARC.L	1072051200000	492.7500	U	-0.1258	.	1.0	1.0
12	tcfg	BARC.L	1072137600000	480.0	U	-0.3728	.	0	0.7500
13	tcfg	BARC.L	1072224600000	491.2500	U	0.1269	.	0	1.0
14	tcfg	BARC.L	1072310400000	491.2500	U	0	.	1.0	0.7500
15	tcfg	BARC.L	1072396600000	491.2500	U	0	.	0.5000	0.2500
16	tcfg	BARC.L	1072656000000	492.0	U	0.1763	.	1.0	0.7500
17	tcfg	BARC.L	1072712400000	496.2500	U	0.3182	.	1.0	1.0
18	tcfg	BARC.L	1072828000000	498.2500	U	0.1988	.	1.0	1.0
19	tcfg	BARC.L	1072915200000	498.2500	U	0	.	1.0	1.0
20	tcfg	BARC.L	1073001600000	504.0	U	0.5285	.	1.0	1.0
21	tcfg	BARC.L	1073268800000	506.0	U	0.1958	.	1.0	1.0
22	tcfg	BARC.L	1073347200000	513.0	U	0.5991	.	1.0	1.0
23	tcfg	BARC.L	1073433800000	518.7500	U	-0.2158	.	1.0	1.0
24	tcfg	BARC.L	1073520000000	518.5000	U	-0.6245	.	0	1.0
25	tcfg	BARC.L	1073606400000	518.5000	U	0.3390	.	1.0	1.0
26	tcfg	BARC.L	1073656500000	508.5000	U	-0.4892	.	0	0.5000
27	tcfg	BARC.L	1073952000000	511.2500	U	0.2648	.	0	1.0
28	tcfg	BARC.L	1074038400000	521.5000	U	0.7626	.	1.0	1.0
29	tcfg	BARC.L	1074124000000	527.7500	U	0.5365	.	1.0	1.0
30	tcfg	BARC.L	1074211200000	536.0	U	0.6536	.	1.0	1.0
31	tcfg	BARC.L	1074478400000	538.7500	D	-0.4548	.	1.0	1.0
32	tcfg	BARC.L	1074556800000	528.7500	D	-0.7362	.	0	0.7500
33	tcfg	BARC.L	1074643200000	517.5000	D	-0.3623	.	0	0.2500
34	tcfg	BARC.L	1074729600000	517.0	D	-0.8483	.	0	0
35	tcfg	BARC.L	1074816000000	567.5000	D	-0.7253	.	0	0
36	tcfg	BARC.L	1075075200000	508.0	D	0.8492	.	0	0
37	tcfg	BARC.L	1075161600000	506.0	D	-0.6569	.	0	0
38	tcfg	BARC.L	1075248000000	511.0	D	0.8005	.	1.0	0.2500

Value	trend	change	dummy	T1	T2	T3
496.0	D	-0.4613	.	0	0	0
495.0	D	-0.1005	.	0	0	0
487.0	D	-0.6685	.	0	0	0
479.2500	D	-0.6616	.	0	0	0
476.2500	D	-0.3032	.	0	0	0
480.2500	D	0.3969	.	1.0	0	0
476.0	D	-0.4157	.	0	0	0
487.7500	U	0.8438	.	1.0	1.0	0.7500
491.2500	U	0.3441	.	1.0	1.0	1.0
494.0	U	0.2728	.	1.0	1.0	1.0
492.7500	U	-0.1258	.	1.0	1.0	1.0
490.0	U	-0.2720	.	0	0.7500	1.0



Experiment Approach

(2) Let's merge trend Labels with trend Features.

"30day Trend" Labels [Up, Down]

Row	TCFG	UserDate	Value	trend	change	dummy	T1	T2	T3
1	tcfg	BARC.L	1078841600000	496.0	D	-0.4613	.	0	0
2	tcfg	BARC.L	1078928000000	495.0	D	-0.1005	.	0	0
3	tcfg	BARC.L	1071014400000	497.0	D	-0.6685	.	0	0
4	tcfg	BARC.L	1071108000000	479.2500	D	-0.5616	.	0	0
5	tcfg	BARC.L	1071107800000	476.2500	D	-0.3932	.	0	0
6	tcfg	BARC.L	1071146400000	480.2500	D	0.3969	.	1.0	0
7	tcfg	BARC.L	1071532800000	476.0	D	-0.4157	.	0	0
8	tcfg	BARC.L	1071619200000	487.7500	U	0.8428	.	1.0	0.7500
9	tcfg	BARC.L	1071705600000	491.2500	U	0.3441	.	1.0	1.0
10	tcfg	BARC.L	1071792000000	494.0	U	0.2728	.	1.0	1.0
11	tcfg	BARC.L	1072051200000	492.7500	U	-0.1258	.	1.0	1.0
12	tcfg	BARC.L	1072137600000	490.0	U	-0.3728	.	0	0.7500
13	tcfg	BARC.L	1072224000000	491.2500	U	0.1269	.	0	1.0
14	tcfg	BARC.L	1072310400000	491.2500	U	0	.	1.0	0.7500
15	tcfg	BARC.L	1072306600000	491.2500	U	0	.	0.5000	0.2500
16	tcfg	BARC.L	1072356000000	492.0	U	0.1763	.	1.0	0.7500
17	tcfg	BARC.L	1072712400000	496.2500	U	0.3182	.	1.0	1.0
18	tcfg	BARC.L	1072828000000	498.2500	U	0.1988	.	1.0	1.0
19	tcfg	BARC.L	1072915200000	498.2500	U	0	.	1.0	1.0
20	tcfg	BARC.L	1073001600000	504.0	U	0.5285	.	1.0	1.0
21	tcfg	BARC.L	1073208000000	506.0	U	0.1958	.	1.0	1.0
22	tcfg	BARC.L	1073317200000	513.0	U	0.5991	.	1.0	1.0
23	tcfg	BARC.L	1073433800000	510.7500	U	-0.2158	.	1.0	1.0
24	tcfg	BARC.L	1073520000000	510.5000	U	-0.0245	.	0	1.0
25	tcfg	BARC.L	1073606400000	509.0	U	0.3390	.	1.0	1.0
26	tcfg	BARC.L	1073656500000	508.5000	U	-0.4892	.	0	0.5000
27	tcfg	BARC.L	1073952000000	511.2500	U	0.2648	.	0	1.0
28	tcfg	BARC.L	1074038400000	521.5000	U	0.7626	.	1.0	1.0
29	tcfg	BARC.L	1074124000000	527.7500	U	0.5365	.	1.0	1.0
30	tcfg	BARC.L	1074211200000	536.0	U	0.6536	.	1.0	1.0
31	tcfg	BARC.L	1074478400000	530.7500	D	-0.4548	.	1.0	1.0
32	tcfg	BARC.L	1074556800000	528.7500	D	-0.7362	.	0	0.7500
33	tcfg	BARC.L	1074643200000	517.5000	D	-0.3623	.	0	0.2500
34	tcfg	BARC.L	1074739600000	517.0	D	-0.8483	.	0	0.2500
35	tcfg	BARC.L	1074816000000	567.5000	D	-0.7253	.	0	0
36	tcfg	BARC.L	1075075200000	508.0	D	0.8492	.	0	0
37	tcfg	BARC.L	1075161600000	506.0	D	-0.6569	.	0	0
38	tcfg	BARC.L	1075248000000	511.0	D	0.8005	.	1.0	0.2500

Value	trend	change	dummy	T1	T2	T3
495.0	D	-0.4613	.	0	0	0
495.0	D	-0.1005	.	0	0	0
487.0	D	-0.6685	.	0	0	0
479.2500	D	-0.6616	.	0	0	0
476.2500	D	-0.3032	.	0	0	0
480.2500	D	0.3969	.	1.0	0	0
476.0	D	-0.4157	.	0	0	0
487.7500	U	0.8438	.	1.0	1.0	0.7500
491.2500	U	0.3441	.	1.0	1.0	1.0
494.0	U	0.2728	.	1.0	1.0	1.0
492.7500	U	-0.1258	.	1.0	1.0	1.0
490.0	U	-0.2720	.	0	0.7500	1.0



Experiment Approach

(2) Let's merge trend Labels with trend Features.

Tz Trend Features

Row	TCFG	UserDate	Value	trend	change	dummy	T1	T2	T3	T1	T2	T3
1	tcfg	BARC.L	10788416000000	496.0	D	-0.4613	.	0	0	0	0	0
2	tcfg	BARC.L	10789280000000	495.0	D	-0.1085	.	0	0	0	0	0
3	tcfg	BARC.L	10710144000000	497.0	D	-0.6685	.	0	0	0	0	0
4	tcfg	BARC.L	10711080000000	479.2500	D	-0.5616	.	0	0	0	0	0
5	tcfg	BARC.L	10711070000000	476.2500	D	-0.3932	.	0	0	0	0	0
6	tcfg	BARC.L	10711464000000	480.2500	D	-0.3969	.	1.0	0	0	0	0
7	tcfg	BARC.L	10715320000000	476.0	D	-0.4157	.	0	0	0	0	0
8	tcfg	BARC.L	10716192000000	487.7500	U	0.8428	.	1.0	1.0	0.7500	0.7500	0.7500
9	tcfg	BARC.L	10717056000000	491.2500	U	0.3441	.	1.0	1.0	0.7500	0.7500	0.7500
10	tcfg	BARC.L	10717200000000	494.0	U	0.2728	.	1.0	1.0	0.8	0.8	0.8
11	tcfg	BARC.L	10720512000000	492.7500	U	-0.1258	.	1.0	1.0	0.8	0.8	0.8
12	tcfg	BARC.L	10721376000000	492.7500	U	-0.2728	.	0	0	0.7500	0.7500	0.7500
13	tcfg	BARC.L	10722240000000	491.2500	U	0.1269	.	0	0	1.0	1.0	1.0
14	tcfg	BARC.L	10723104000000	491.2500	U	0	.	1.0	0	0.7500	0.7500	0.7500
15	tcfg	BARC.L	10723296000000	491.2500	U	0	.	0.5000	0.2500	0	0.7500	0.7500
16	tcfg	BARC.L	10726560000000	492.0	U	0.1763	.	1.0	0	0.7500	0.7500	0.7500
17	tcfg	BARC.L	10727124000000	496.2500	U	0.3182	.	1.0	1.0	1.0	1.0	1.0
18	tcfg	BARC.L	10728280000000	498.2500	U	0.1988	.	1.0	1.0	1.0	1.0	1.0
19	tcfg	BARC.L	10729152000000	498.2500	U	0	.	1.0	1.0	1.0	1.0	1.0
20	tcfg	BARC.L	10730016000000	504.0	U	0.5285	.	1.0	1.0	1.0	1.0	1.0
21	tcfg	BARC.L	10732080000000	506.0	U	0.1958	.	1.0	1.0	1.0	1.0	1.0
22	tcfg	BARC.L	10733172000000	513.0	U	0.5991	.	1.0	1.0	1.0	1.0	1.0
23	tcfg	BARC.L	10734238000000	510.7500	U	-0.2158	.	1.0	1.0	1.0	1.0	1.0
24	tcfg	BARC.L	10735200000000	510.5000	U	-0.0245	.	0	1.0	1.0	1.0	1.0
25	tcfg	BARC.L	10736064000000	509.0	U	0.2390	.	1.0	1.0	1.0	1.0	1.0
26	tcfg	BARC.L	10736565000000	508.5000	U	-0.4892	.	0	0	0.5000	0	1.0
27	tcfg	BARC.L	10739520000000	511.2500	U	0.2648	.	0	1.0	0.7500	0.7500	0.7500
28	tcfg	BARC.L	10740384000000	521.5000	U	0.7626	.	1.0	1.0	1.0	1.0	1.0
29	tcfg	BARC.L	10741240000000	527.7500	U	0.5365	.	1.0	1.0	1.0	1.0	1.0
30	tcfg	BARC.L	10742112000000	536.0	U	0.6536	.	1.0	1.0	1.0	1.0	1.0
31	tcfg	BARC.L	10744784000000	530.7500	D	-0.4548	.	1.0	1.0	1.0	1.0	1.0
32	tcfg	BARC.L	10745568000000	528.7500	D	-0.7362	.	0	0	0.7500	0.7500	0.7500
33	tcfg	BARC.L	10746432000000	517.5000	D	-0.3623	.	0	0	0.2500	0.7500	1.0
34	tcfg	BARC.L	10747296000000	517.0	D	-0.8483	.	0	0	0	0.2500	0.7500
35	tcfg	BARC.L	10748160000000	567.5000	D	-0.7253	.	0	0	0	0	0.2500
36	tcfg	BARC.L	10750752000000	506.0	D	0.8492	.	0	0	0	0	0.2500
37	tcfg	BARC.L	10751616000000	506.0	D	-0.6569	.	0	0	0	0	0.2500
38	tcfg	BARC.L	10752480000000	511.0	D	0.8005	.	1.0	0.2500	0	0	0.2500



ML Experiments tried:

Non-Temporal Models

1. Gradient Boosted Trees
2. Random Forest
3. Feed Forward Neural Nets

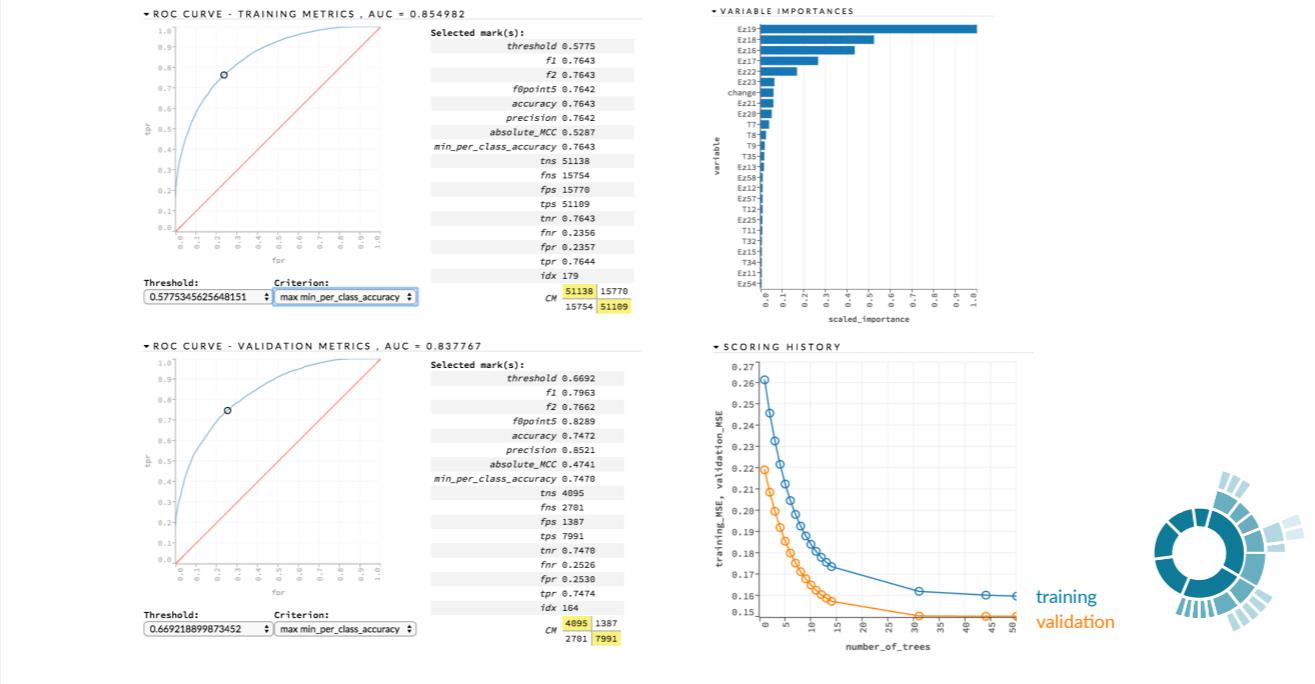
Temporal Models

- a. LSTM Recurrent Neural Nets
- b. Echo State Machines



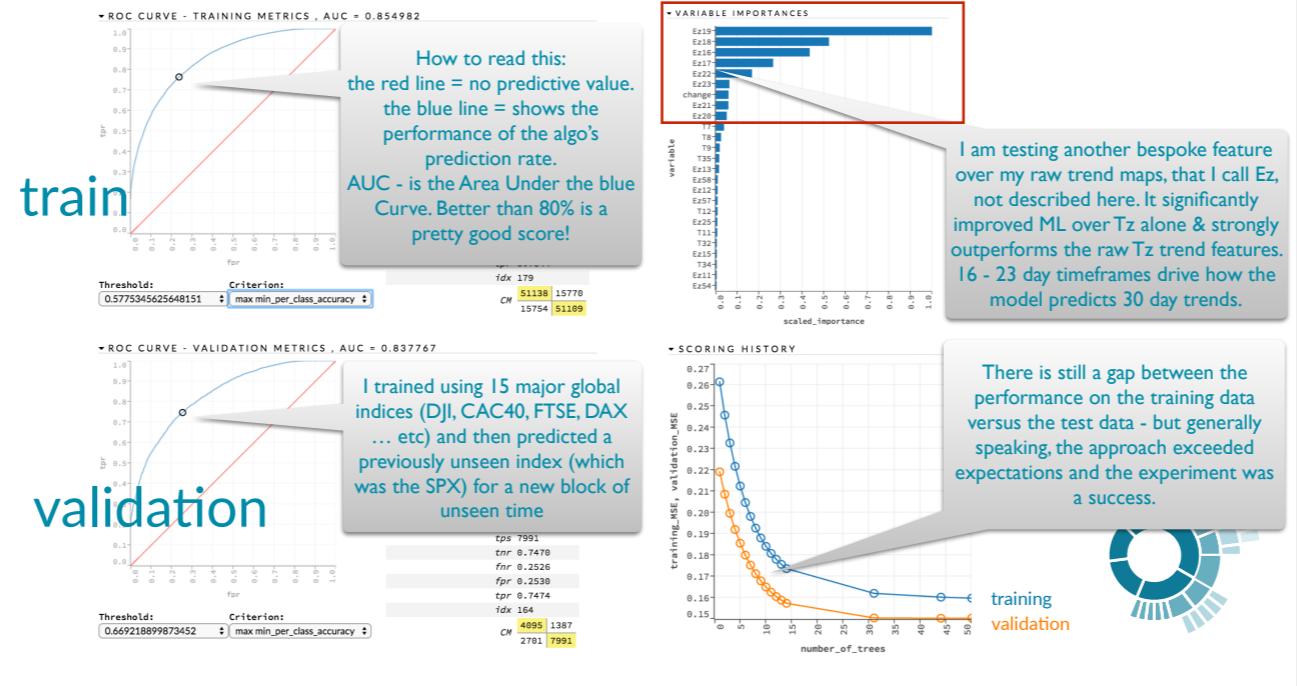
Experiment Results:

Gradient Boosted Trees: AUC = 0.837767



Experiment Results:

Gradient Boosted Trees: AUC = 0.837767



What are my other results?

The results so far are pretty good, but there's lots to do:

Major point of learning:

My “temporal features” trained non-temporal algorithms!

To Do: I should generate Sharpe Ratios to evaluate performance properly, which is still to do.

I tested out some other algorithms too, including:

Non-Temporal Models

- 1. Gradient Boosted Trees AUC 0.8377 - best score
- 2. Random Forest (slightly worse than GBT, but didn't grid search much)
- 3. Feed Forward Neural Nets AUC 0.7803 (perhaps my poor tuning? Needs rerunning)

Temporal Models - unfinished work in progress

- a. LSTM Recurrent Neural Nets (see my github for Torch7 code - basic coding done)
- b. Echo State Machines (i did this too, it's an amazing story, for later)

Next Steps. As these tests were all done 2 years ago, I should refactor to latest Keras + Tensorflow, formalise the methods, results evaluation, and documentation.



Next steps

The results so far are pretty good, but there's lots to do.

For example, I developed a trend feature that has *really* improved the quality of the predictions beyond what I've described here ... so lots of things to try out including more advanced feature building.

The future: as public interest in RNNs catches up with me, some great new RNN methods coming available outside of Torch7 which could be helpful.

Note: My torch7 code for my LSTM experiments is now online here:
<https://github.com/bytesumo/LSTM-Experiments>



Summary

My experiments have much to do with a new way of thinking about trends in time series and how we learn about them.

I'm proposing through my work that **trends in society and economics exists, irrespective of their measurement.**

How we sense these, and build methods to understand them, is a challenge we've not yet tackled well.

My experiments demonstrate the research is fruitful, and full of new ground to explore.



Co-Trending

Taking things even further.

New Directions

So far we have examined a single Time Series on its own.

But - there are lots of time series. Billions of them?

How do we look at **co-trending across these series?**

How might we better predict, through understanding co-trending, and potentially do causation discovery?

We are moving into a new area, the study of Causality Discovery through the lens of Co-trending.

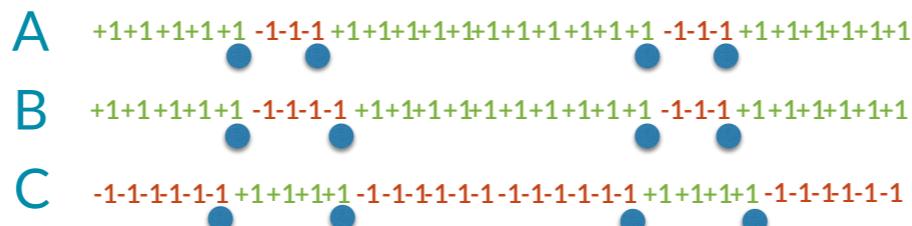


What is co-trending?

Trend reversals define *time periods*, on a time scale, having a consistent “trend” - a tendency up or down.

Trends can thus be viewed as a partitioning of time. Turning points, when trends change, are important.

Co-trending - is when times series “tend” to change their trend directions, in concert.



What is co-trending?

Distance metrics are an obvious approach to finding similarity - and several methods exist.

But what about finding leading indicators?

For prediction, we need methods for finding these.

A +1+1+1+1+1+1 -1-1-1 +1+1+1+1+1+1+1+1+1+1+1+1+1+1+1 -1-1-1 +1+1+1+1+1+1+1

B +1+1+1+1+1+1 -1-1-1-1 +1+1+1+1+1+1+1+1+1 -1-1-1 +1+1+1+1+1+1

C -1-1-1-1-1-1+1+1+1+1-1-1-1-1-1-1-1-1-1+1+1+1+1-1-1-1-1-1



Process discovery?

If we consider “leading indicator discovery” as a process discovery task, that could be useful too.

It would seek to discover “unseen process couplings” of trend effects seen across our time series, to suggest relations between trends (rather than shock propagation)

A +1+1+1+1+1 -1-1-1 +1+1+1+1+1+1+1+1+1+1 -1-1-1 +1+1+1+1+1+1
B +1+1+1+1+1 -1-1-1-1 +1+1+1+1+1+1+1+1+1+1 -1-1-1 +1+1+1+1+1+1
C -1-1-1-1-1 +1+1+1+1 -1-1-1-1-1-1-1-1-1-1 +1+1+1+1 -1-1-1-1-1



Shoulders to stand on.

Several approaches to measuring Co-Trending come to mind, and the best paper is here:

<http://wolfweb.unr.edu/~zal/pubs/MTA.pdf>

But before we proceed with methods that are Available, perhaps we need to ask, what is Needed?

By defining a target “causal Model”, we may better define the requirements for “Co-Trending” methods we’ll develop.



Next Steps

I have a ton more research and have enlisted some help from some amazing people ... but that work is not ready for public distribution just yet.

If you are interested in learning more, or helping, do reach out.

< More details coming soon! >



andrew@bytesumo.com



Data Science, Data Architecture, Big Data Engineering.

Andrew Morgan is the Head of Data Science for a top 4 audit firm client. He is also the CEO of ByteSumo Ltd., a data science consultancy. He's been excited about "big data" since 1993.

He is a specialist in data processing languages, data platform design, emerging data technologies, exotic data structures, data science methods, technical architecture, and data security systems.

He founded ByteSumo to build a data science led consultancy that has the experts and tools needed to transform and disrupt traditional enterprises.

(curr. client role)	2014 - present	Head of Data Science
ByteSumo	2013 - present	CEO
Capgemini	2010 - 2013	Senior Enterprise Architect, BIM
Thomson Reuters	2006 - 2010	Architect, Senior Technologist
Aprimo (now Teradata)	2005 - 2006	Senior Consultant
Axiom Corporation	2000 - 2005	Business Solutions Architect
dunnhumby	1999 - 2000	Database Consultant
Elf Gas & Power UK	1995 - 1999	Operational Dev. Executive
Gov't of Ontario	1994 - 1994	Jnr. Planner, GIS systems. Bachelor of Arts, Geography. University of Toronto. 1994 Specialism in Geostatistics, and Geographic Information Systems.



Copyright © 2015 ByteSumo Limited
All rights reserved.

