

The Nirvana Club

BASH 2016

ARM4 Advances:

Genetic Algorithm Improvements

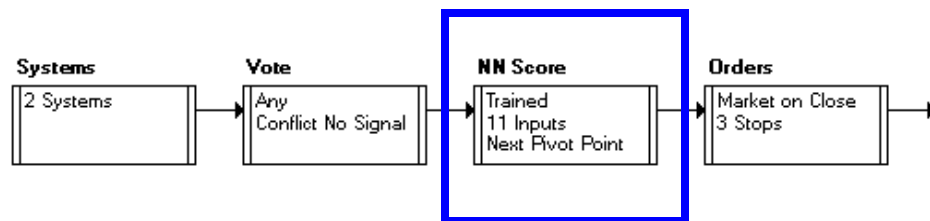
Ed Downs & Gianluca Paganoni

- **We Know Some Good Things to Look at in Charts.**
We often know that **Market Factors** help predict price action, such as Price Behavior, Volume, Patterns, etc.
- **But We Don't Know How to Put them Together.**
However, we often don't know HOW they interact, e.g., “How much Volume indicates a move?” or “Which Patterns with Which Indicators are most predictive?”
- **A.I. Tools can Figure it Out.**
ARM4 can automatically find **combinations of market events and measurements** that have resulted in profitable opportunities in the past (“Back Test”) – and validate those combinations in a “Forward Test”.

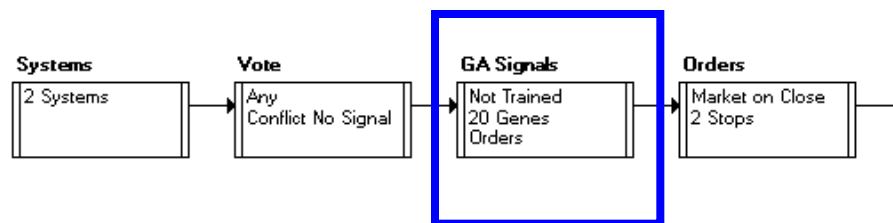
Two Powerful Components

Two powerful A.I. techniques are **Neural Networks** and **Genetic Algorithms**.

Neural Networks are implemented in ARM4 as the **NN Score** Component



Genetic Algorithms are implemented in ARM4 as the **GA Signals** Component



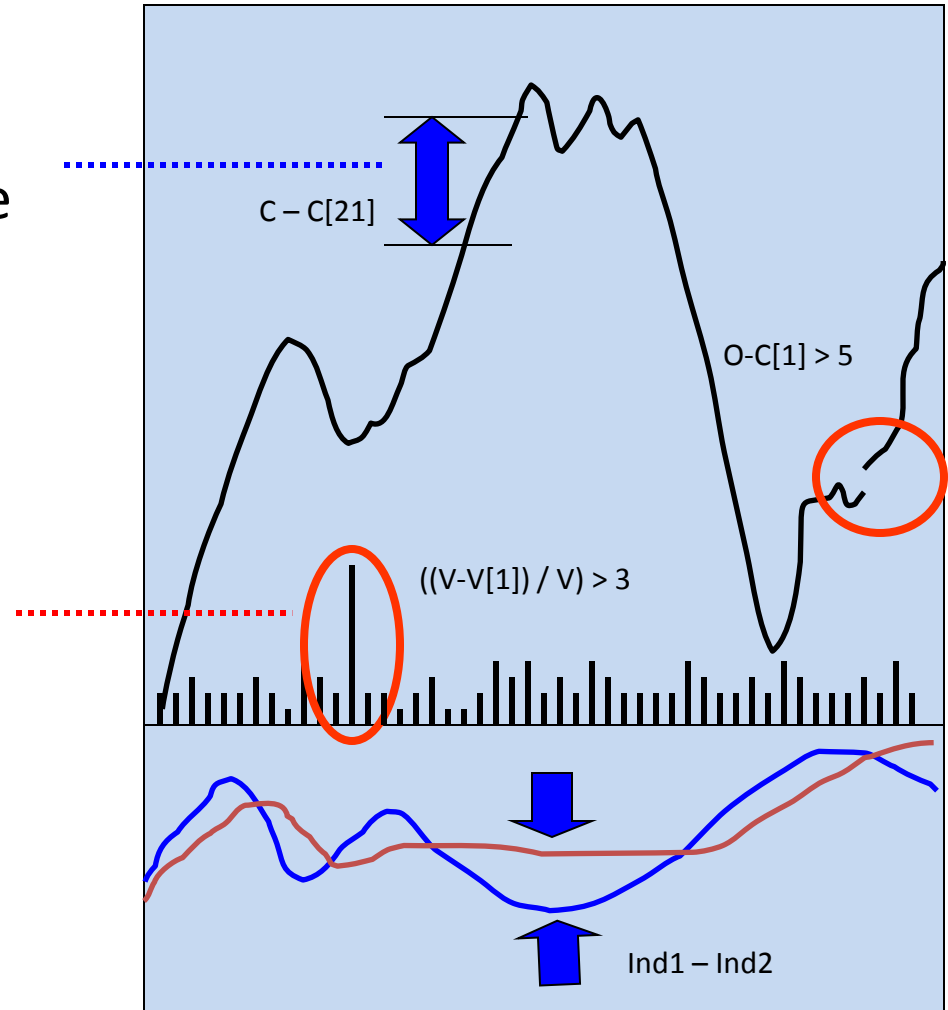
The ARM3 NN Score and GA Signal components are interchangeable. Which tool to use depends on the problem we are trying to solve.

Neural Networks:

Work best on **Measurements** (Ranges) such as Volatility, Price Differences, Indicators, etc.

Genetic Algorithms:

Work best on **Discrete Events**, such as the existence of Gaps, Trading Signals, and other “Boolean” expressions.



Focus on the Genetic Algorithm

**BASH
2016**

Neural Networks work well on problems where we have no idea how the inputs are related to profitability.

Genetic Algorithms work well when each input is generally understood and we see clear evidence of a relation to profit.

Fuzzy Inputs:

- Difference between Close and a Moving Average
- Value of the Stochastics Indicator (0 to 100)

Explicit Inputs:

- 21p Moving Average Above 55p Moving Average.
- Stochastics Indicator Is Below 20

Since 2008, the Genetic Algorithm has not been used as much as the Neural Network. In recent experiments, we are seeing evidence that the GA can provide Amazing Results if used in new ways. Before we study these new uses, let's take some time to understand how a GA works...

UNDERSTANDING GENETIC ALGORITHMS



From BASH 2007

What is a Genetic Algorithm?

BASH
2016

Find **RULES** based on specific combinations of Inputs, such as:

IF:

MACD is rising AND

We have a Triangle Consolidation AND

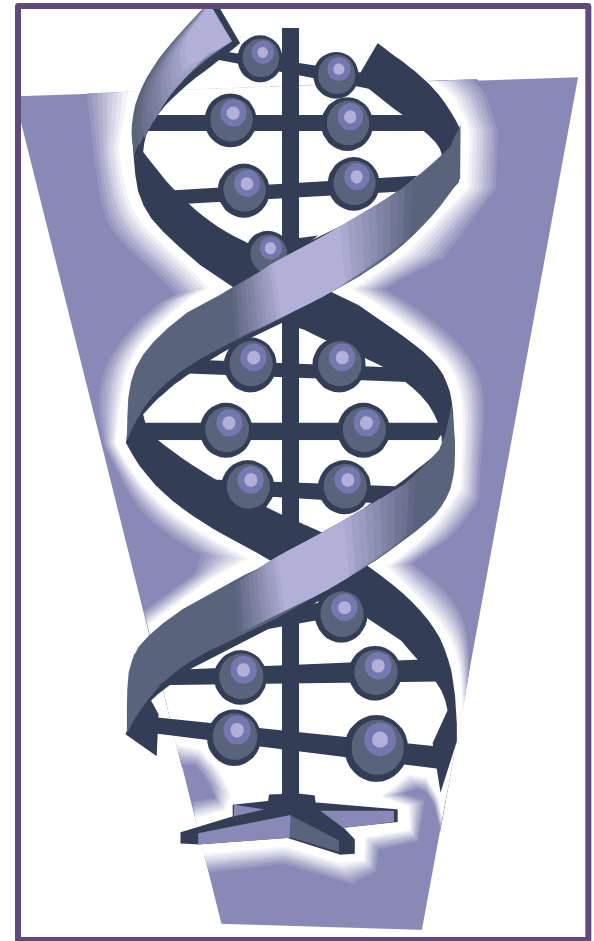
Volume is increasing AND

Today's Close is > Yesterday's High,

THEN:

Average profit is observed to be 5%.

GAs are good for finding specific relationships between a set of inputs.



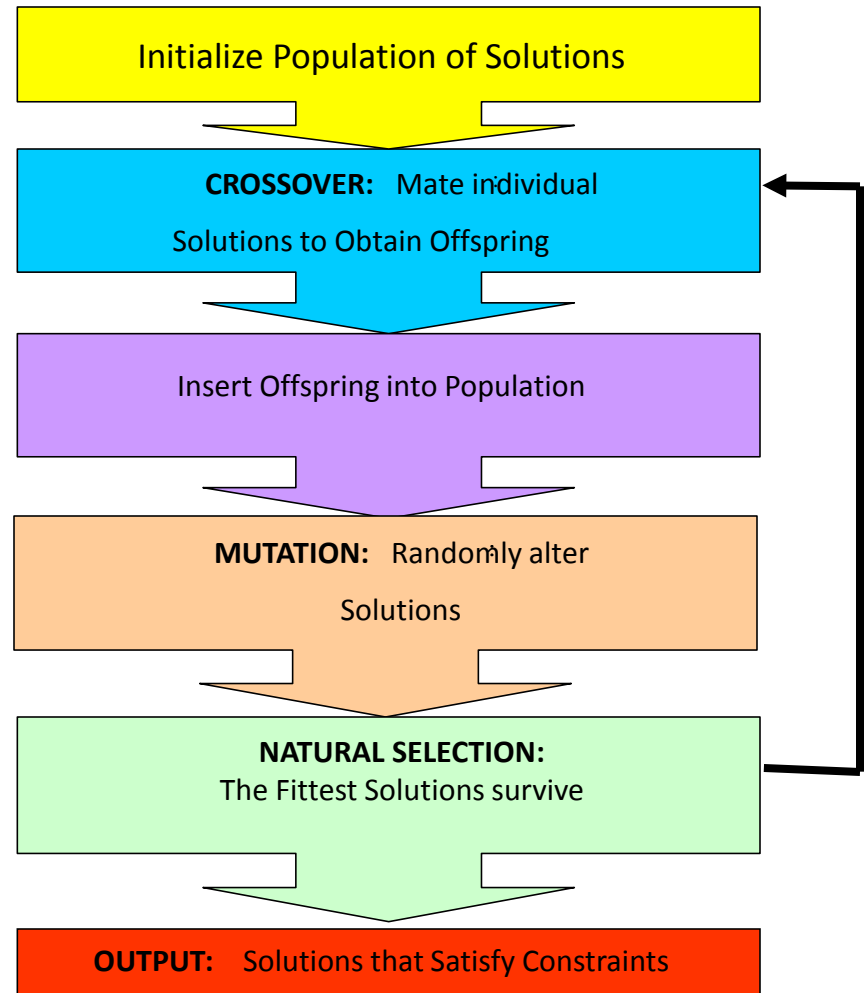
BASH 2007

A Genetic Algorithm works by executing a sequence of steps that mimic the way genes combine to create new offspring.

After each generation, the **Output** is measured to see if it satisfies the constraints.

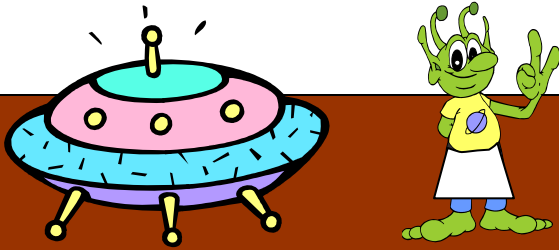
Only the “fittest” solutions survive the process. These solutions are used to create more solutions.

This continues until the **Output** is not improving much with successive generations.



Martian Cake Problem

BASH
2016



- Marvin the Martian visits Earth and gets a job as a cook. Unfortunately, on his first day, the owner asks him to bake a cake!
- Problem: Marvin has never seen a cake. So, he asks a waiter how to do it.
- The waiter doesn't know either, but tells him some ingredients that he thinks will work.
- He also offers to taste a limited number of cakes and tell Marvin if they taste good.

Potential Ingredients

Flour

Sugar

Yeast

Baking Powder

Egg Whites

Butter

Milk

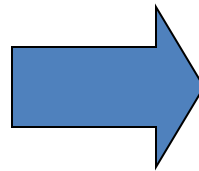
Vanilla

Oil

Salt

BASH 2007

- The waiter tells him the correct amount of each ingredient is probably one of the standard baking measures, or a combination.



Possible Amounts

(none)

1 teaspoon

2 teaspoons

3 teaspoons

1 tablespoon

2 tablespoons

3 tablespoons

4 tablespoons

1/4 cup

1/3 cup

1/2 cup

1 cups

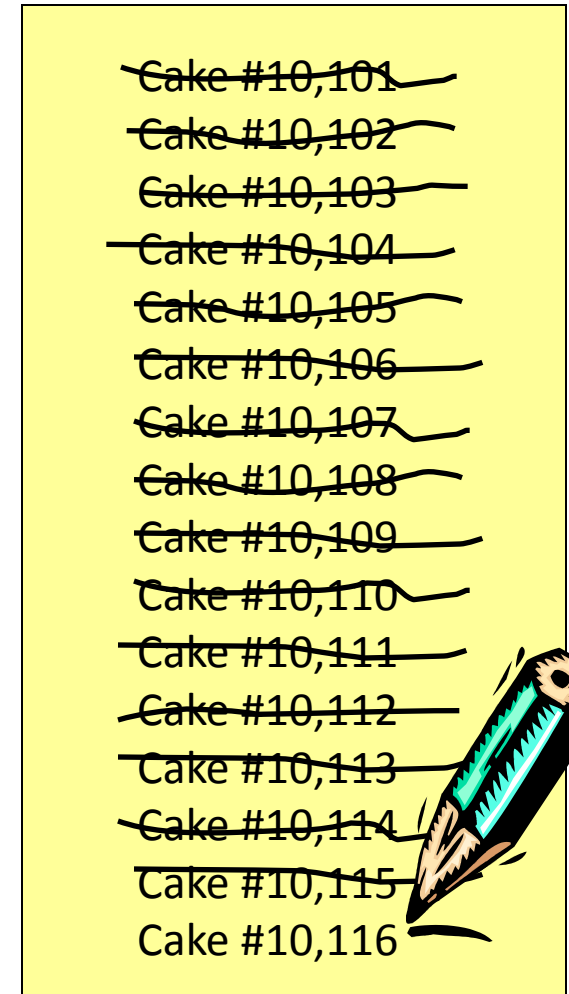
2 cups

3 cups

OR Combinations!

A Huge Problem

- There are 10 ingredients and about 20 different measurement combinations (like 1 ½ Cup, 1 ¼ Cup, etc.)
- This means there are 10 to the 20th or **100,000,000,000,000,000,000 combinations.**
-
- He wisely decides to use a Genetic Algorithm to solve the problem!



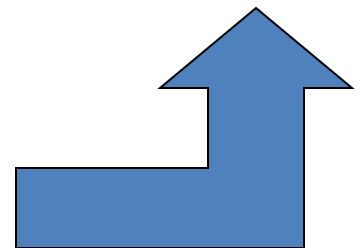
GA Step 1: Initial Population

Chrom	Flour	Vanilla	Sugar	Salt	Bake S.	Eggs	Milk	Oil	T.O.
1	½ cup	¼ cup	¼ cup	1 cup	½ cup	1 tbsp	1 cup	1 tbsp	?
2	1 cup	¼ cup	1 cup	½ cup	1 cup	3 cup	½ cup	3 cup	?
3	½ cup	2 cup	¼ cup	½ cup	1 cup	1 cup	3 cup	1 cup	?
4	3 cup	1 cup	¼ cup	½ cup	1 tbsp	1 tbsp	1 cup	1 tbsp	?
5	1 cup	1 tbsp	2 cup	2 cup	1 cup	3 cup	½ cup	1 cup	?
6	¼ cup	1 tbsp	1 tsp	1 tsp	½ cup	1 cup	3 cup	½ cup	?
7	2 cups	1 cup	1 tbsp	1 tbsp	¼ cup	¼ cup	3 cup	½ tbsp	?
8	2 cup	2 cup	1 tbsp	2 cup	1 cup	¼ cup	1 cup	3 cup	?
9	1 tsp	1 tsp	¼ cup	1 tsp	1 tsp	2 cup	¼ cup	2 cup	?
10	1 tbsp	1 tsp	2 cup	1 tbsp	1 tbsp	1 tbsp	¼ cup	1 tsp	?

He makes an initial set of guesses, totally randomly.

Typically, it will be 100 or so. Here are the first 10.

The Target Output (T.O.) is how good the cake tastes.



GA Step 2: Measure the Outputs

Chrom	Flour	Vanilla	Sugar	Salt	Bake S.	Eggs	Milk	Oil	T.O.
1	½ cup	¼ cup	¼ cup	1 cup	½ cup	1 tbsp	1 cup	1 tbsp	YUK
2	1 cup	¼ cup	1 cup	½ cup	1 cup	3 cup	½ cup	3 cup	YUK
3	½ cup	2 cup	¼ cup	½ cup	1 cup	1 cup	3 cup	1 cup	Nope
4	3 cup	1 cup	¼ cup	½ cup	1 tbsp	1 tbsp	1 cup	1 tbsp	OK
5	1 cup	1 tbsp	2 cup	2 cup	1 cup	3 cup	½ cup	1 cup	YUK
6	¼ cup	1 tbsp	1 tsp	1 tsp	½ cup	1 cup	3 cup	½ cup	YUK
7	2 cups	1 cup	1 tbsp	1 tbsp	¼ cup	¼ cup	3 cup	½ tbsp	OK
8	2 cup	2 cup	1 tbsp	2 cup	1 cup	¼ cup	1 cup	3 cup	YUK
9	1 tsp	1 tsp	¼ cup	1 tsp	1 tsp	2 cup	¼ cup	2 cup	YUK
10	1 tbsp	1 tsp	2 cup	1 tbsp	1 tbsp	1 tbsp	¼ cup	1 tsp	Ugh.

The waiter reluctantly tastes all 10 and Marvin fills in the Target Output column with his comments. **“Chromosomes” number 4 and 7 are the best.**

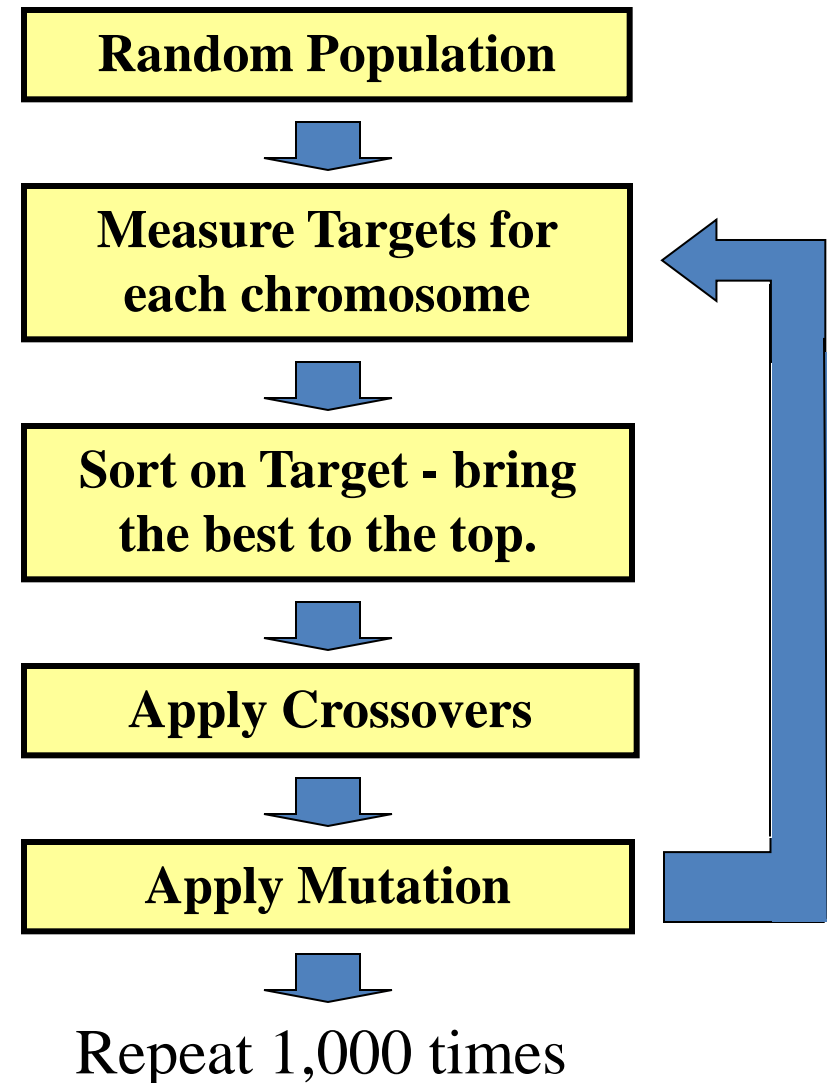
GA Step 3: Crossover, Mutation

	Chrom	Flour	Vanilla	Sugar	Salt	Bake S.	Eggs	Milk	Oil	T.O.
Best	4	3 cup	1 cup	¼ cup	½ cup	1 tbsp	1 tbsp	1 cup	1 tbsp	OK
	7	2 cups	1 cup	1 tbsp	1 tbsp	¼ cup	¼ cup	3 cup	½ tbsp	OK
New	NEW	3 cup	1 cup	¼ cup	½ cup	1 cup	¼ cup	3 cup	½ tbsp	?
	NEW	2 cups	½ cup	1 tbsp	1 tbsp	1 tbsp	1 tbsp	1 cup	1 tbsp	?
	NEW	3 cup	1 cup	1 tbsp	1 tbsp	1 tbsp	1 tbsp	2 cup	½ tbsp	?
	NEW	2 cups	1 cup	¼ cup	½ cup	¼ cup	¼ cup	1 cup	1 tbsp	?
	NEW	3 cup	1 cup	1 tbsp	1 tbsp	1 tbsp	¼ cup	1 cup	½ tbsp	?
	NEW	2 cups	1 cup	¼ cup	½ cup	¼ cup	2 tbsp	1 cup	1 tbsp	?
	NEW	2 cups	1 cup	¼ cup	1 tbsp	1 tbsp	1 tbsp	1 cup	1 tbsp	?
	NEW	3 cup	2 cup	¼ cup	½ cup	¼ cup	¼ cup	3 cup	1 tbsp	?

He moves the best to the top of the list, then use **crossovers** on the best chromosomes to make new ones. He also randomly changes a few values to create **mutations**.

GA Step 4: Repeat to Improve

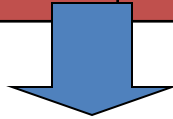
- The same process repeats over and over, until the **Target Output** (taste) is not changing very much.
- The list of **chromosomes** is sorted on the value of the **Target Output** (which is “taste” in this case) so that the best results “float to the top.”



Success!

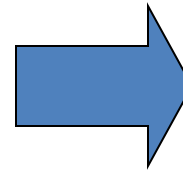
BASH 2016

Chrom	Flour	Vanilla	Sugar	Salt	Bake S.	Eggs	Milk	Oil	T.O.
175	2.5 cup	1 tsp	1.5 cup	1.0 tsp	1 tbsp	4	1.2 cup	0.6 cup	Great
202	2.6 cups	1 tsp	1.5 cup	1.5 tsp	1 tbsp	4	1.3 cup	0.6 cup	Perfect!



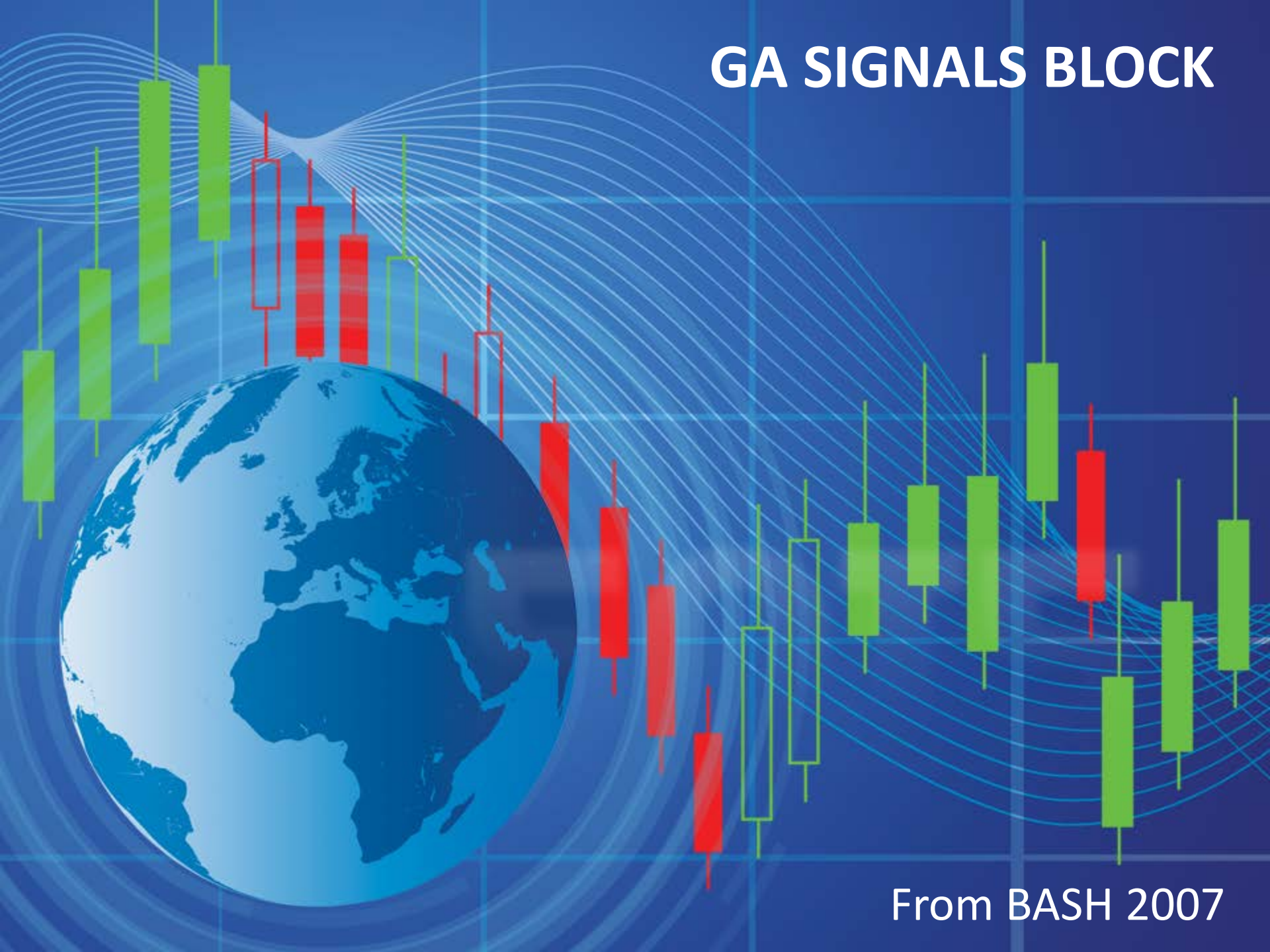
Cake Recipe

- 2.6 Cups flour
- 1.5 Cups Sugar
- 4.0 Tablespoons Baking Powder
- 1.0 Teaspoon Salt
- 0.6 Cups Oil
- 1.3 Cups Milk
- 1.0 Teaspoon Vanilla
- 4.0 Egg Whites



BASH 2007

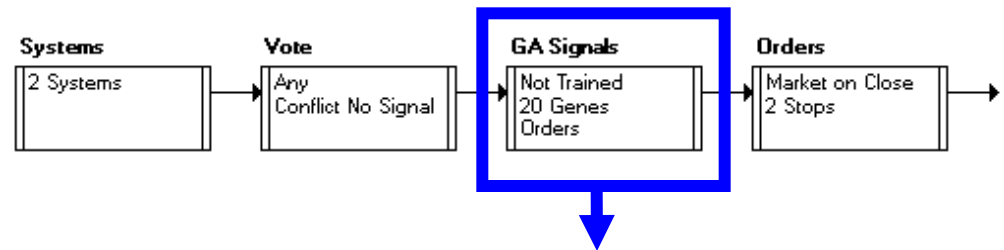
GA SIGNALS BLOCK



From BASH 2007

GA Signals uses a Genetic Algorithm to generate “smart signals” based on the current values of several Ingredients or **Inputs** at each bar.

Profitable combinations form the **Rules** of a **Knowledge Base**.



Genetic Algorithm

Generate Signals for:
 All Symbols
 Commodity Group:
 Specific Symbols:

Canned Settings
Target Accuracy:
 High
 Medium
 Low
 Custom

Knowledge Base Lookup
Minimum Requirements:
Profit per Trade:
Hit Rate: 65
APR: 100
Hits: 3
Std. Deviation:

Signals / Bar / 1000 Symbols
Long Signals
Number of rules: 12
Number of signals: 13
Short Signals
Number of rules: 26
Number of signals: 30

GA Information
A Genetic Algorithm for longs was run on 5/31/2006 10:45:05 AM.
Number of Iterations: 198.
Final fitness: 128.23%.
Total number of rules found: 15.
Number of collected samples: 3000.
Disabled genes: 8.

A Genetic Algorithm for shorts was run on 5/31/2006 10:45:45 AM.
Number of Iterations: 143.
Final fitness: 149.44%.
Total number of rules found: 37.
Number of collected samples: 3000.

Data Collection Date Range: 1/6/2005 8:30:00 AM to 6/1/2005 8:30:00 AM.

#	Gene	Symbol	Signal
1	ADX(14)		Both
2	ASWING(14,3)		Both
3	ATR(14) - EMA(ATR(14),9)		Both
4	BOL_UPPER(13,2) - BOL_LOWER(13,		Both
5	BOP(14) - EMA(BOP(14), 9)		Both
6	CHMF(21)		Both
7	CCI(14)		Both
8	EMA(9) - EMA(21)		Both
9	INERTIA(14,10,14) - INERTIA(14,10,1		Both
10	LNREG_HIST(10,14,14)		Both
11	LNREG_SLOPE(9)		Both
12	MACD_HIST(12,26,9)		Both
13	RSI(14)		Both

Fitness Results
APR: 152%
Rules Found: 40

Ingredients or Inputs

Knowledge Base

A **Knowledge Base** is a collection of **Trading Rules**, that are used to Forecast future Trends.

Example Rule:

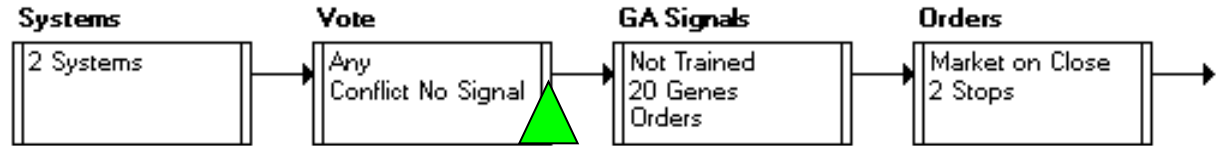
If Close(Today) < Open(Today)
And Volume(Today) – Volume(Yesterday) > 3K
And VTL-B(Today) is a LONG SIGNAL
And Trendline Break on \$SPX within the past week

Then ENTER LONG

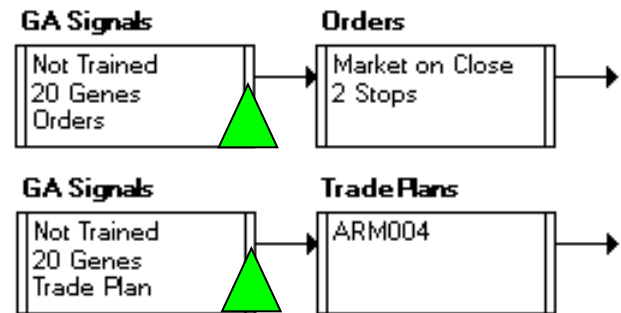
Rules are found by searching combinations of the inputs using the GA Process of **Crossover** and **Mutation**.

Ways to Use GA Signals

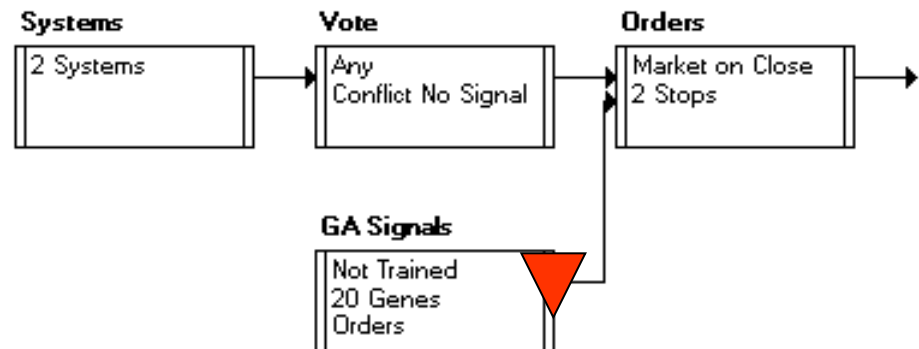
1. Use GA Signal to **FILTER** a **Signal**



2. Let GA Signal **GENERATE** a **Signal**



3. Use GA Signal to **Generate** an **Exit**



Configuring GA Signals

Settings Dialog

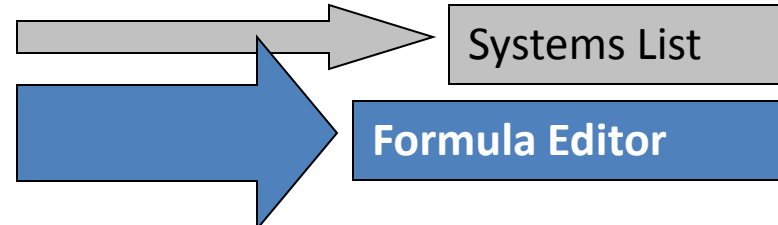
Define inputs for GA Signal to create a Knowledge Base.

Inputs can be:

- Signals from a Trading System
- A Fluid Measurement
- A Boolean Expression

#	Gene	Symbol	Signal
1	ADX(14)		Both
2	ASWING(14,3)		Both
3	ATR(14) - EMA(ATR(14),9)		Both
4	BOL_UPPER(13,2) - BOL_LOWER(13,2)		Both
5	BOP(14) - EMA(BOP(14), 9)		Both
6	CHMF(21)		Both
7	CCI(14)		Both
8	EMA(9) - EMA(21)		Both
9	INERTIA(14,10,14) - INERTIA(14,10,14)		Both
10	LNREG_HIST(10,14,14)		Both
11	LNREG_SLOPE(9)		Both
12	MACD_HIST(12,26,9)		Both
13	RSI(14)		Both
14	RVI(9,2)		Both
15	TRIX(9)		Both
16	V - EMA(V,20)		Both
17	WMA(9) - WMA(21)		Both
18	VTY_PRICE(14,5) - EMA(VTY_PRICE, 14)		Both
19	ULT(7,14,28)		Both
20	(H-L) - ATR(14)		Both

System
Measurement
Boolean



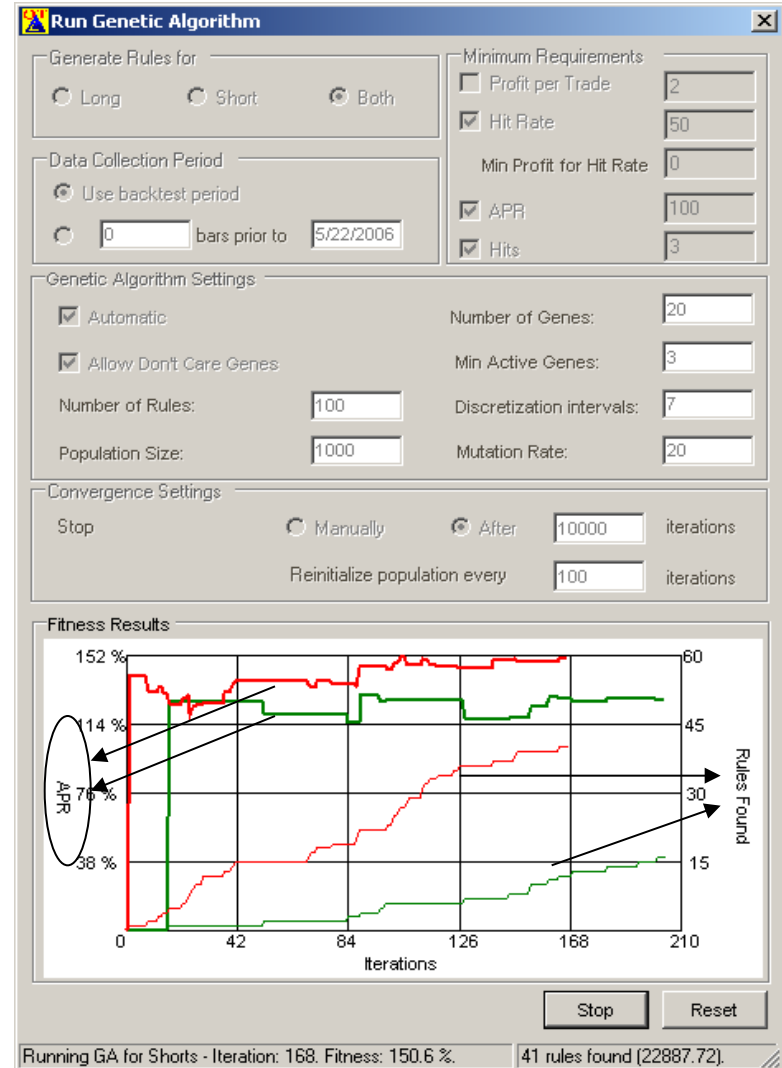
Training Dialog

Shows the settings for training and also the Certainty Plot.

Certainties between 45 and 70 are considered very good.

Certainties above 70 indicate over-training.

You can watch the progress of training using the graph, which shows quantity of rules found and average APR per Rule.



Main Dialog

Minimum Requirements can be set for Rules in the Knowledge Base.

- **Profit per Trade**
- **Hit Rate**
- **APR** (per Trade)
- **# of Hits**
- **Standard Deviation**

A Rule can only be used if its performance is above the **Hit Rate**, **APR**, and **# of Hits** specified.

Genetic Algorithm

Generate Signals for:
 All Symbols
 Commodity Group: []
 Specific Symbols: []

Canned Settings
Target Accuracy
 High
 Medium
 Low
 Custom

Knowledge Base Lookup
Minimum Requirements:
Profit per Trade: []
Hit Rate: [65]
APR: [100]
Hits: [3]
Std. Deviation: []

Signals / Bar / 1000 Symbols
Long Signals
Number of rules: 12
Number of signals: 13
Short Signals
Number of rules: 26
Number of signals: 30
[Update]

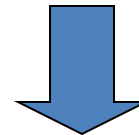
GA Information
A Genetic Algorithm for longs was run on 5/31/2006 10:45:05 AM.
Number of Iterations: 198.
Final fitness: 128.29 %.
Total number of rules found: 15.
Number of collected samples: 3000.
Disabled genes: 8.
A Genetic Algorithm for shorts was run on 5/31/2006 10:45:45 AM.
Number of Iterations: 143.
Final fitness: 149.44 %.
Total number of rules found: 37.
Number of collected samples: 3000.

Data Collection Date Range: 1/6/2005 8:30:00 AM to 6/1/2005 8:30:00 AM.

#	Gene	Symbol	Signal
1	ADX(14)		Both
2	ASWING(14,3)		Both
3	ATR(14) - EMA(ATR(14),9)		Both
4	BOL_UPPER(13,2) - BOL_LOWER(13,		Both
5	BOP(14) - EMA(BOP(14), 9)		Both
6	CHMF(21)		Both
7	CCI(14)		Both
8	EMA(9) - EMA(21)		Both
9	INERTIA(14,10,14) - INERTIA(14,10,1		Both
10	LNREG_HIST(10,14,14)		Both
11	LNREG_SLOPE(9)		Both
12	MACD_HIST(12,26,9)		Both
13	RSI(14)		Both

Fitness Results
APR: 0% to 152%
Rules Found: 0 to 40
Iterations: 0 to 200

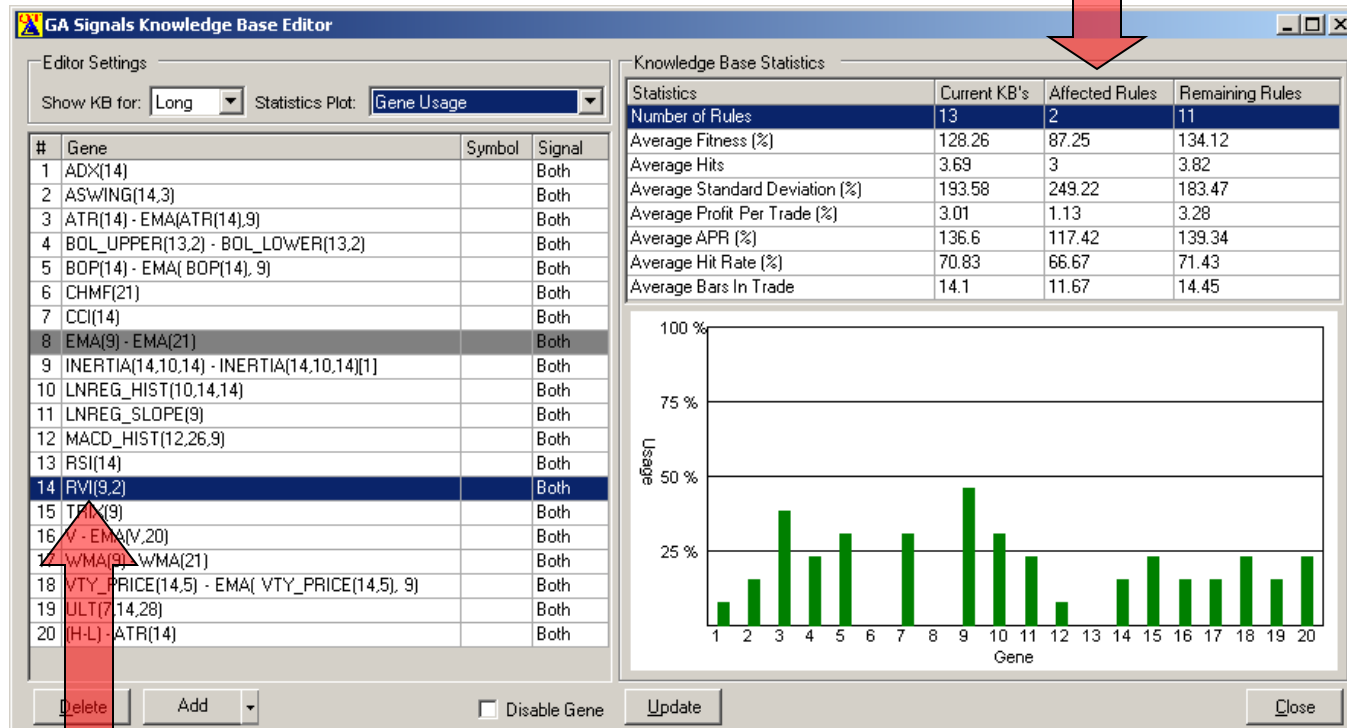
[Edit KBs] [Settings] [Cancel] [OK]



KB Editor

The Editor makes it easy to isolate inputs that do not help the Knowledge Base.

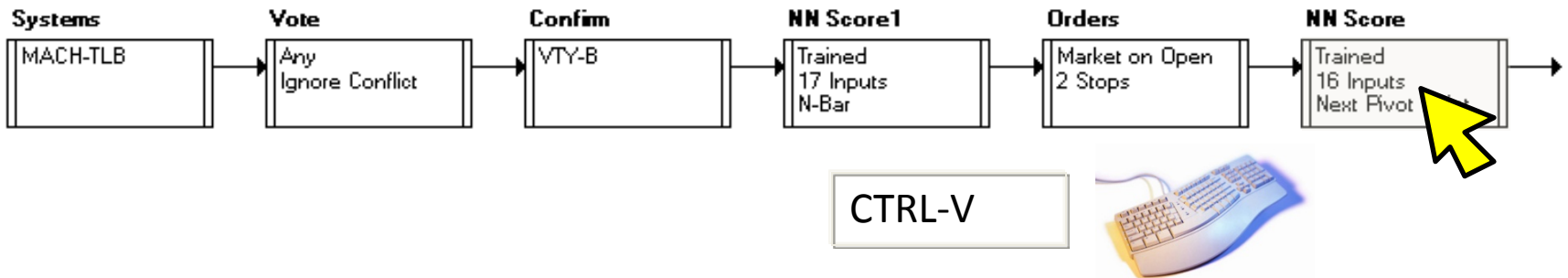
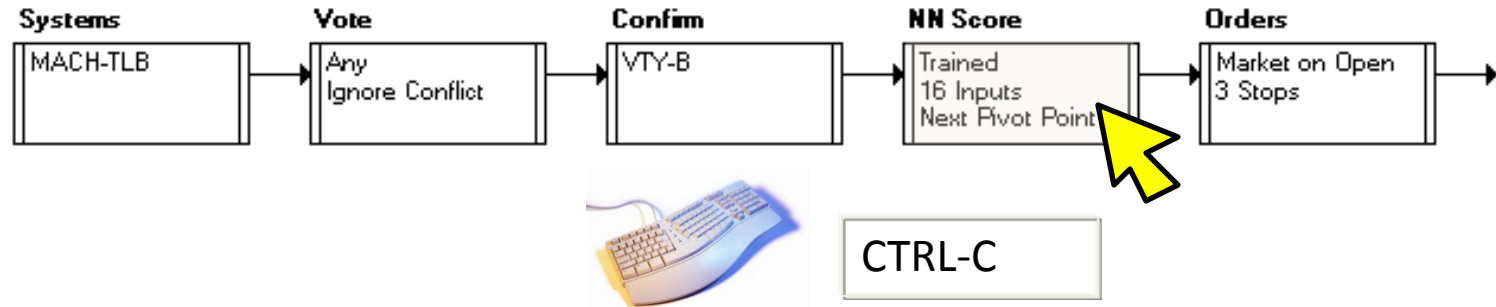
By removing these inputs, we can reduce size and make it possible to add more inputs to further improve the Knowledge Base.



Input #14 is used in 2 of the 13 rules in this Knowledge Base. The rules that use Input #14 have lower performance, which means we won't lose much by deleting this input "Gene."

Moving a Knowledge Base

BASH
2016



You can Cut and Paste Components from one Strategy to another. Click a component, press CTRL-C, open another Strategy and press CTRL-V. All training information is kept with the copy. This is the easiest way to get started on a new idea.

BASH 2007

THE POWER OF BOOLEAN INPUTS



Three Kinds of Inputs (“Genes”)

There are 3 Kinds of Inputs we can feed to the ARM4 Components: **Measurements, Booleans, and Systems.**

Measurement:
STO(14)

Bins (5):	1	2	3	4	5
Range:	0-20	21-40	41-60	61-80	81-100

System:
RSI-P (from System list)

Value	1	2	3
System	Long	Short	No Signal

Boolean:
C > EMA(55)

Bin:	1	2
Boolean	True	False

Advantages of Systems and Booleans:

1. Dramatically Simplify the Problem
2. Fewer Possible Combinations = FAST Training Time

Boolean & System “Genes”



A Boolean is either True or False

Rule Description	Formula
Today's Close Greater than Today's Open	$C > O$
Today's Close Greater than 55p EMA	$C > \text{EMA}(55)$
Today's 21p EMA Greater than Today's 55p EMA	$\text{EMA}(21) > \text{EMA}(55)$
Today's Volume Greater than Twice the Average Volume over 14 bars	$V > 2 * \text{Avg}(V, 14)$
Today's Range Greater than Twice the Average Range over 14 bars	$\text{Abs}(H-L) > 2 * \text{Avg}(\text{Abs}(H-L), 14)$
RSI-P System Long	RSI-P

A New Focus for the G.A.

BASH
2016

Genetic Algorithms based on **Booleans** are surprisingly powerful in predicting chart movement.

The screenshot shows the 'Genetic Algorithm' window with the following sections:

- Generate Signals for:** All Symbols, Commodity Group, Specific Symbols.
- Canned Settings:** Target Accuracy: High, Medium, Low, Custom.
- Knowledge Base Lookup:** Minimum Requirements: Basic | Advanced. Profit per Trade: [], Hit Rate: 70, APR: [], Hits: 100, Max Std. Deviation: [].
- Signals / Bar / 1000 Symbols:** Long Signals: Number of rules: 33, Number of signals: 135. No rules for Short Signals. [Update]
- GA Information:** A Genetic Algorithm for longs was run on 5/10/2016 3:35:17 PM. Number of Iterations: 9. Final fitness: 0.41517 %. Total number of rules found: 304. Number of collected samples: 99553. Data Collection Date Range: 12/1/2011 8:30:00 AM to 5/1/2014 8:30:00 AM.
- Table of Genes:**

#	Gene	Symbol	Timeframe	Dynamic	Signal
1	TRII(60,30) > 80	SPY	Current	<input checked="" type="checkbox"/>	Both
2	RelMom(31,"SPY") > 0		Current	<input checked="" type="checkbox"/>	Both
3	RelMom(21,"SPY") > 0		Current	<input checked="" type="checkbox"/>	Both
4	RelMom(14,"SPY") > 0		Current	<input checked="" type="checkbox"/>	Both
5	TRII(60,15) > 80		Current	<input checked="" type="checkbox"/>	Both
6	TRII(60,7) > 80		Current	<input checked="" type="checkbox"/>	Both

Fitness Results Graph: Shows Profit per Trade (left axis, 0.11% to 0.44%) and Rules Found (right axis, 0 to 320) over Iterations (0 to 20). The graph shows a sharp increase in both metrics within the first few iterations, reaching a plateau.

Bottom Buttons: Edit KBs, Settings, Cancel, OK.

Callout Box: Simple GA with 6 Boolean Inputs

The Power of Simple

BASH
2016

Results from our very simple, 6-
Input Boolean Genetic Algorithm

Back Test (training) Period:
May 2012 – May 2014

Forward Test (out of sample):
May 2014 – May 2016

GA Cutoff Parameters:
Hit Rate: 70%
Min Hits: 100



All Trades: GA FUNDS

Abbreviation	Statistic	Back Test	Forward Test
NT	Number of Trades	1,535	2,202
PT	Profitable Trades	1,020	1,228
HR%	Average Hit Rate (%)	66.45	55.77
ANP%	Average Net Profit Per Symbol (%)	6.08	2.31
PPT%	Average Net Profit Per Trade (%)	0.84	0.23
ABT	Average Bars Per Trade	6	6

Not bad for just 6 Inputs!

Can we do Better?

Here's a GA with 48 Genes

#	Gene	Symbol
1	TRII(60,30) > 80	SPY
2	RelMom(31,"SPY") > 0	
3	RelMom(21,"SPY") > 0	
4	RelMom(14,"SPY") > 0	
5	TRII(60,15) > 80	
6	TRII(60,7) > 80	
7	EMA(21)>EMA(55)	
8	EMA(14)>EMA(21)	
9	EMA(21)>EMA(31)	
10	EMA(31)>EMA(55)	
11	C > 15	\$VIX
12	C > 20	\$VIX
13	C > 25	\$VIX
14	TRII(60,15) > 80	SPY
15	TRII(60,7) > 80	SPY
16	C < 0	\$VIX
17	C[1] < O[1]	\$VIX
18	C[2] < O[2]	\$VIX
19	C[3] < O[3]	
20	V < V[1]	
21	V[1] < V[2]	
22	V[2] < V[3]	
23	RSI(6) < 30	
24	RSI(14) < 30	

#	Gene	Symbol
25	RSI(21) < 30	
26	RSI(6) > RSI(6)[3]	
27	RSI(14) > RSI(14)[3]	
28	RSI(21) > RSI(21)[3]	
29	Inreg_slope(20)[40] < 0	
30	Inreg_slope(20)[20] < 0	
31	Inreg_slope(20) < 0	
32	Inreg_slope(10)[20] < 0	
33	Inreg_slope(10)[10] < 0	
34	Inreg_slope(10) < 0	
35	Inreg_slope(50)[100] < 0	
36	Inreg_slope(50)[50] < 0	
37	Inreg_slope(50) < 0	
38	EMA(31)>EMA(55)	SPY
39	EMA(21)>EMA(31)	SPY
40	ABS(H-L) > AVG(H-L,10)	
41	ABS(H-L) > 1.2 * AVG(H-L,10)	
42	ABS(H-L) > 1.4 * AVG(H-L,10)	
43	V > Avg(V,10)	
44	V > 1.5*avg(v,10)	
45	V - 2*Avg(V,10)	
46	V - 2.5*Avg(V,10)	
47	V > 3*avg(v,10)	
48	V > 4 * avg(v,10)	

This is a broad collection of Boolean Inputs, some of which are calculated against other symbols (like SPY and \$VIX)

Improvement

Here's the Result.

In this case we flipped Back Test and Forward Test (training was on more recent data).



All Trades: GA FUNDS 20 5-14 to 5-16 (t3)			
Abbreviation	Statistic	Back Test	Forward Test
NT	Number of Trades	530	1,054
PT	Profitable Trades	440	747
HR%	Average Hit Rate (%)	83.02	70.87
ANP%	Average Net Profit Per Symbol (%)	8.88	14.81
PPT%	Average Net Profit Per Trade (%)	3.37	3.05
ABT	Average Bars Per Trade	21	21

You don't see 83% accuracy and 3% PPT every day!

Understanding it is Easy!

BASH
2016

If we probe any GA Signal line, we can see which Genes are firing at a Bar.

We can also use the Knowledge Base Editor (see Review section) to understand which Genes generally work well and which ones are not used much.



Rule Report

The Rule Report shows specific Signals identified in the Back Test.

This is useful in validating operation of the GA, and observing how many times the Rule fired on the Same Date.

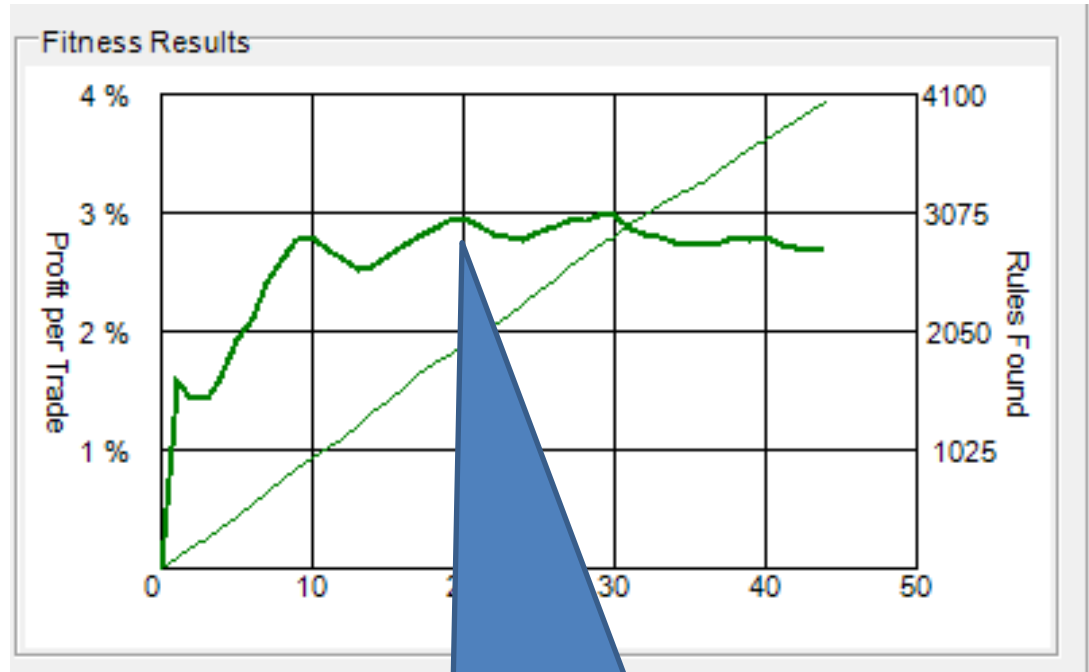
GA Rule Summary for the 'GA Signals' block

Rule Number	334
Total Hits	2,006
Average Bars In Trade	21.00
Average Profit (%)	2.00
Average APR (%)	24.40

<u>Symbol</u>		<u>Date</u>	<u>Bars In Trade</u>	<u>Profit</u>	<u>APR</u>
\$VIX	5/9/2012	8:30:00AM	21	5.73	71.89
\$VIX	5/9/2012	8:30:00AM	21	5.73	71.89
\$VIX	5/10/2012	8:30:00AM	21	25.12	323.83
\$VIX	5/10/2012	8:30:00AM	21	25.12	323.83
DOG	5/14/2012	8:30:00AM	21	0.93	10.66
DOG	5/14/2012	8:30:00AM	21	0.93	10.66
UUP	5/14/2012	8:30:00AM	21	1.59	18.33
DOG	5/15/2012	8:30:00AM	21	-0.87	-12.74
DOG	5/15/2012	8:30:00AM	21	-0.87	-12.74
DOG	5/16/2012	8:30:00AM	21	-1.79	-24.57
DOG	5/16/2012	8:30:00AM	21	-1.79	-24.57
UNG	5/16/2012	8:30:00AM	21	-8.82	-117.39
DOG	5/17/2012	8:30:00AM	21	-2.83	-38.22
DOG	5/17/2012	8:30:00AM	21	-2.83	-38.22
UNG	5/17/2012	8:30:00AM	21	-3.23	-44.72
UNG	5/17/2012	8:30:00AM	21	-3.23	-44.72
UNG	5/18/2012	8:30:00AM	21	-9.06	-120.38
UNG	5/18/2012	8:30:00AM	21	-9.06	-120.38
UNG	5/21/2012	8:30:00AM	21	-6.87	-92.17
UNG	5/21/2012	8:30:00AM	21	-6.87	-92.17
UNG	5/22/2012	8:30:00AM	21	-6.88	-92.04
UNG	5/22/2012	8:30:00AM	21	-6.88	-92.04
UNG	5/22/2012	8:30:00AM	21	-6.88	-92.04
UNG	5/22/2012	8:30:00AM	21	-6.88	-92.04

Training is Super Fast!

Even using 50 Genes over many years of data, a Knowledge Base can reach most of its profitable combinations in about One Minute!



2,000 Rules were generated with average Profit per Trade of 3% in about ONE MINUTES!

Now, Finding Great Rules is Easy!



- Identify Technical Indicators that imply upward movement and add them to the GA.
- Enable Training
- Start the ToDo List
- Examine Statistics for Validation
- Trade the Signals!



THE **NEW** GENETIC INDICATORS



The Concept

BASH
2016

With Boolean Inputs, a large number of rules can be generated that are predictive because they can have many “hits”.

This means we can now COMBINE Rules to generate multi-Rule Statistics!

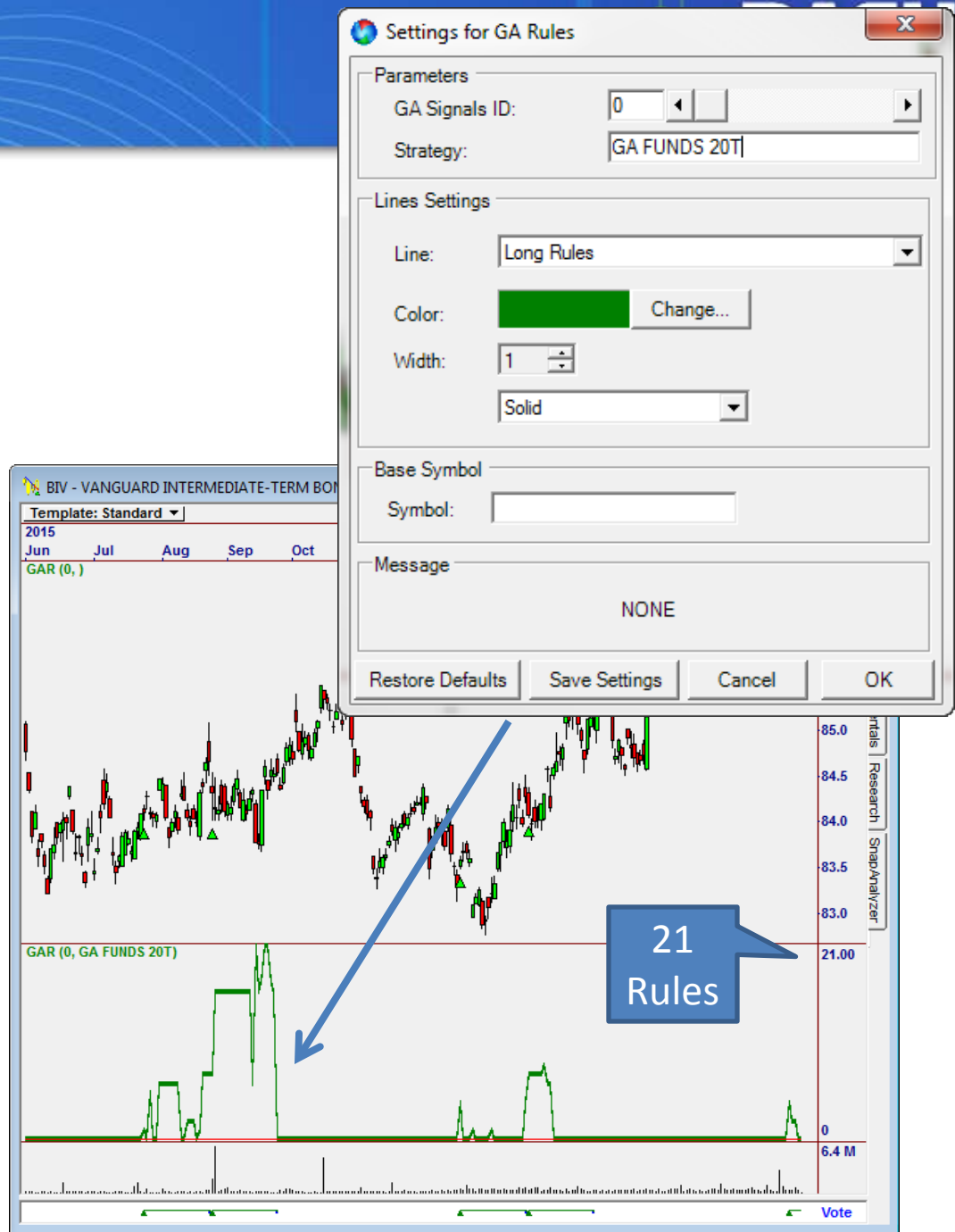


Signals from a Boolean Knowledge Base
650 Hits and 83% Accurate

Rule Count

GARules() is a NEW indicator that tells us how many rules are firing on a bar.

Clusters of Rules Tend to occur at viable trading points.

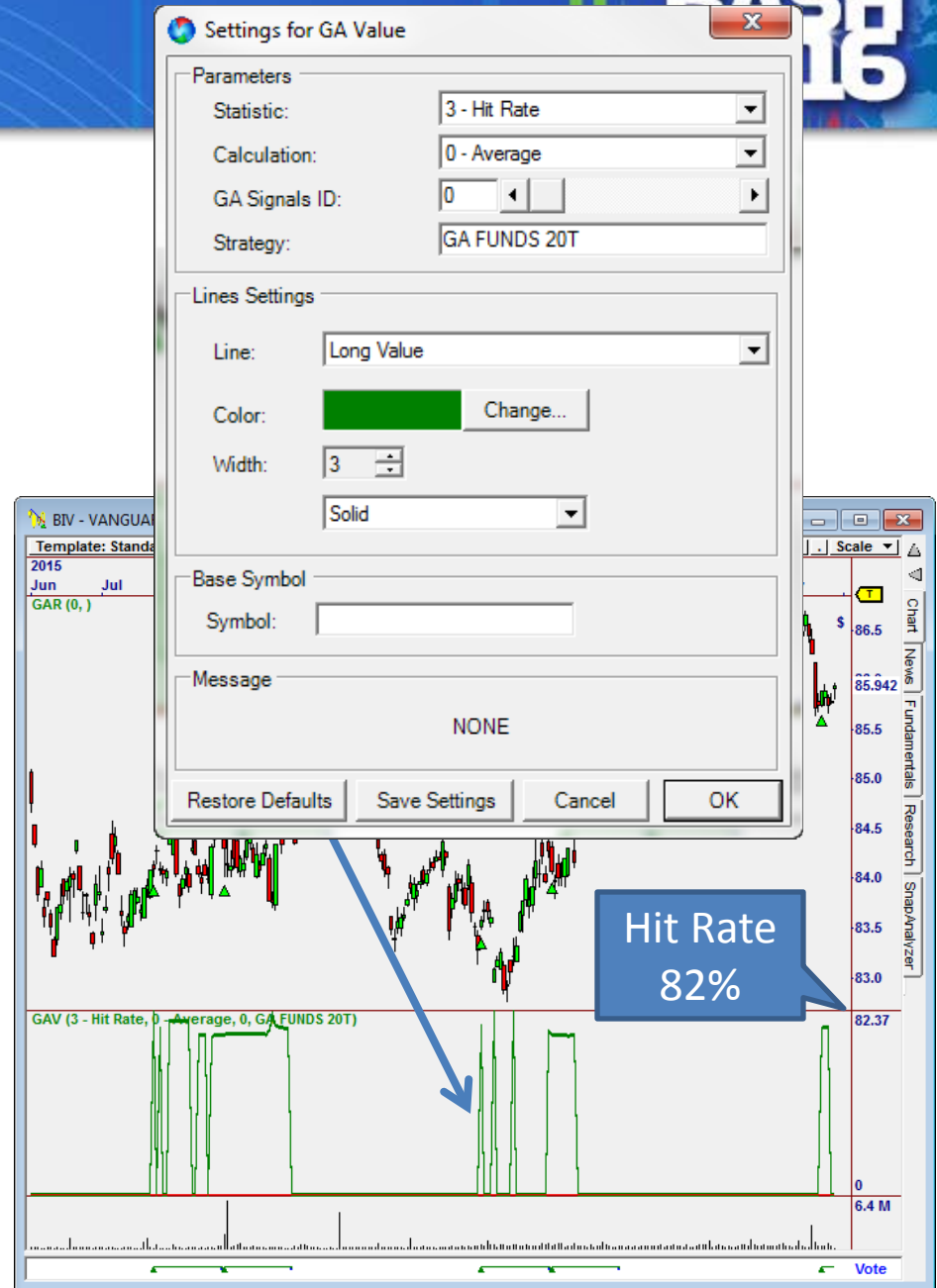


Rule Statistics

GValue() is an Indicator that gives us Summary Information on all the Rules that fire on a bar.

<u>Calculation:</u>	<u>Statistic:</u>
Average	Hit Rate
Highest	APR
Median	Profit/Trade

This indicator has many Powerful Uses – including RANKING in OmniTrader, VisualTrader, & OmniFunds!



GA Indicators are very new.

The **increased Signals** and information provided by **GA Indicators** yields the potential for many advancements in our platforms, including:

1. Filters in Strategies
2. Sorting Signals derived from Strategies
3. Measurements in VisualTrader
4. Ranking Opportunities in ALL our platforms
5. Voting Multiple GA's that are built with different Genes or Targets

A Recipe for Success

BASH
2016

New discoveries are enabling us to apply **Genetic Algorithms** in ways we never before considered.

Boolean Genes and **GA Indicators** are an exciting **NEW** field of study that are already generating exciting results!



The Nirvana Club

BASH 2016

