

# How to Evaluate Horse Races Using ChaosHunter

*By DaGeniusNZ*

## Background

In the 1990's I made a living for a short time as a professional punter using crude computer ratings of my own design. I bet in \$1000 units and was successful until my life partner of the time threatened to leave if I didn't quit "gambling". Despite making money, I quit and went off a got a "real job". A short time later she left anyway and went off to "experience the world".

During that time I invented the "Value Rating" (which was plagiarised by a publisher I sent my manuscript to, called the Value Bet and given away to subscribers of their magazine) and also developed a trifecta method that captured 80% of all trifectas using economical outlays. I wrote about my methods in two self-published books, hopefully soon to be available on Amazon.

By the year 2000 there were a lot of computer rating methods, most of which have faded away because they couldn't predict very well on data they'd never seen before. I got stuck into developing new methods as my old methods no longer had an edge given the prevalence of other computer methods (although my value rating and trifecta methods still work well). In 2004 I remarried and shelved my interest in racing to focus on building a solid marriage.

Since the advent of ChaosHunter, **quite possibly the most brilliant mathematical software in the world today**, I've pulled out my old database and returned to serious study. With ChaosHunter, I've developed one formula that's quite successful in pinpointing value with show betting (called place betting in NZ – picking horses that finish, 1<sup>st</sup>, 2<sup>nd</sup> or 3<sup>rd</sup>) and I'm now busy perfecting methods for picking the winner in the least amount of selections.

I'm still working on improving my predictions. Not only are there are lot of different outputs to choose from, but there is an improved range of variables to analyse. These new variables allow a handicapper to go back into a horses racing history and assign performance ratings for each and every run it has had and then to average it's performances and compare them with every other runner in the same race. The only difficulty has been obtaining a good quality database to analyse.

## Comparing Patterns

I've been evaluating horse races using Neuroshell 2 and ChaosHunter. ChaosHunter is bar far the more powerful tool of the two and I've learned a lot about how to get the best performance out of it for making successful predictions.

I find that the old method of comparing patterns doesn't work very well for assessing horse races as there are too many variables to assess. The old method set out patterns thus:

### Where

Pattern One = data for Horse One; and

Pattern Two = data for Horse Two

Evaluate the following:

Pattern One + Pattern Two = output

Pattern Two + Pattern One = output

This method works well where Patterns are short and does not work well where Patterns are long and consist of many variables. In horse racing, the Patterns are normally long. In these situations the method I prefer is thus:

Evaluate the following:

Pattern One + comparison of Pattern One with other patterns = output

Pattern Two + comparison of Pattern Two with other patterns = output

What my method does is add comparisons to the patterns being analysed and thereby removes the need for doubling the pattern length.

Pattern One often consists of the following data:

- No of races run, plus no of 1sts, 2nds, 3rds, 4ths
- Percentages for above data - 1<sup>st</sup>%, 2<sup>nd</sup>%, 3<sup>rd</sup>%, 4<sup>th</sup>%
- Combination of 1+2%, 1+2+3%, 1+2+3+4%
- Weight
- Barrier
- Jockey success rate
- Last finish position
- Last field size
- Combination of last finish position with last field size to give a Form Rating.

## **Underlying Principles**

Most of my evaluation focuses on two principles. These two principles influence the way in which I use ChaosHunter, particularly as regards the range of data I choose to analyse.

### **Principle One:**

In every race some horses are overbet and some are underbet.

### **Principle Two:**

There are some races made for favourites to win and some made for outsiders.

Three variables that influence this situation are:

1. Field Size
2. Distance
3. Track Conditions

Ideally, these different kinds of races should be evaluated differently as there is likely to be different dynamics driving horse performance under each of these types of races.

*Please note that the figures given below are guidelines only and don't represent the actual results, rather they are rounded to the nearest 5%. Nevertheless, they reliably illustrate the likely outcomes.*

#### Field Size

6-14 – Top 2 favs win 45% or more

15+ - Top 2 favs win 30% or less

#### Distance

1200-1400 – Top 2 favs win 50% or more

1500-2000 – Top 2 favs win 45% or more

2100+ - Top 2 favs win 40% or less

#### Track Conds

Dry – Top 2 favs win 50% or more

Wet – Top 2 Fav's win 45% or less

### **The Database**

A database full of raw data is very difficult to analyse and make sense of. Steve Ward often emphasises the need to preselect variables to assist ChaosHunter in doing its work. I find that in order to make sense of raw data, it's necessary to add in useful statistics. For instance, barrier draw is just a number whereas statistics for barrier draws add meaning to that number.

There are 3 basic stages to engage in before the database is ready for analysis and, following this, 4 steps.

Stage 1 – select raw data to analyse.

Stage 2 – process the data for analysis.

Stage 3 – decide on an output variable .

Ideally, the ChaosHunter predictions out can be converted to a dividend by dividing output into 1 = 1/output = win dividend

Dividends can be compared for assessing value

I enclose several sample databases. Sample 1 shows the complete range of raw data available for assessment. Sample 2 shows a reduced database which uses less memory and is easier to analyse as it contains only a range of potentially relevant variables..

#### **The first step is to remove some races:**

Some types of races are simply not predictable because either there is too little information on each contender. This is especially true where there are horses starting for the first time at a race meeting or horses that have yet to win a race. No-one knows how well they will perform and even horses that have performed well at trials may run poorly due to fear of the barrier, nervousness in front of a large crowd, and other reasons.

I remove from my database all races where there are first starters.

I also remove all maiden races.

In New Zealand we also have lots of jump races, i.e. races over hurdles and steeples. These kinds of races have different drivers for performance and should be rated separately to regular flat races.

I remove all races for hurdles or steeples.

Some races are tailor-made for favourites and some for outsiders. In particular, some types of races are simply not predictable because there are either too many contenders to select from or the large number of contenders inhibits performance. In large fields a horse can be blocked for a run, trapped in the middle of a bunch, or be forced to run extra distance by running wide in order to get around slower horses.

My research shows that the top two favourites in a race win at a much lower rate in these larger fields (30% of the time or less) than they do in smaller fields (45% of the time or more. In very small fields as much as 53% of the time).

I remove all races where there are 15 or more horses. I find that where I am interested in betting on any race with a large field, say for a pick six event, I can apply the analysis carried out on smaller fields to these larger fields and get reasonable results.

I remove all races with field size  $\geq 15$

*(here the strike rate for top two favs = 30% whereas in smaller fields it's usually 45% or better).*

My statistics show that longer races are also races that are difficult for favourites to win as favourites win longer races at a much lower rate than they do for shorter races. I attribute this to a greater need for fitness and stamina than a good jockey or consistent performance, which tend to be the drivers behind favouritism.

Remove all races greater than 2100 metres.

**The second step is to add in relevant statistics as follows:**

Raw data can be meaningless and it can be difficult for ChaosHunter to find the relevant relationships in it. It makes the task of analysis so much easier where statistics are used to interpret variables.

For instance, how is ChaosHunter to properly understand the relationship between the number of starts and the number of wins and places? No doubt it can do the job but I find analysis proceeds more smoothly when the data is pre-interpreted (just a little) for ChaosHunter.

Probably the most useful variable of this nature is a simple form rating I developed to compare finish positions to field size and award a higher rating to placings in larger fields than placings in smaller fields. My theory is that to beat 17 horses in a field of 18 has more merit than beating only 9 horses in a field of ten. A second part of my theory is that to win carries more merit than finishing second. So I've constructed a

form table based on these theories and use it to represent Last finish position and last field size. In most of my analysis, Chaoshunter seizes on this as an important variable.

Percentages for wins and places in the track conditions, over the distance, and on the course.

For examples; raced 10 on the course for 4 wins and 2 places =  $(4+2)/10 = 60\%$

Percentages for 1sts, 2nds, 3rds, 4ths, “1st+2nds”, “”, and “1sts+ 2nds+3rds+4ths”.

For examples; raced a total of 20 times for 6 X 1st and 2 X 2<sup>nd</sup>, 1 X 3<sup>rd</sup>, 3 x 4<sup>th</sup> =

6/20 = 33% - 1sts

2/20 = 10% - 2nds

1/20 = 5% - 3rds

3/20 = 15% - 4ths

8/20 = 40% - 1st+2nds

9/20 = 45% - 1<sup>st</sup>+2nds+3rds

12/20 -60% - 1<sup>st</sup>+2nds+3rds+4ths (called places)

ChaosHunter finds it much easier to use these figures than to try and work out the relationships between times raced (starts) and performance results (no of 1sts, 2nds etc).

I assign percentages to weeks since last raced, no of starts, age, and barrier - based on tables constructed from an analysis of past results of 29,000 races. For example, barriers draws as follows:

Barrier Draw Win %

1	0.0865
2	0.0845
3	0.0826
4	0.0819
5	0.0815
6	0.0813
7	0.0811
8	0.0810
9	0.0794
10	0.0778
11	0.0750
12	0.0712
13	0.0683
14	0.0655
15	0.0629
16	0.0603
17	0.0568
18	0.0532

Other tables I construct are tables that marry last finish position to field size (based on the idea that number of horses beaten also counts).

An analysis of starts (times raced) shows that there is a clear bias towards starters with fewer races but with some experience under their belt. A horse that has raced 8 times previously is likely to win more often than any other horse.

I add in the jockey success rate for places.

**The third step is to average some variables:**

I average:

- Placings in the last five starts. I also assign a Last Five place percentage.
- No of horses beaten in the last five races.
- Beaten lengths in the last five races.
- Favouritism in the last five starts. I view favouritism as a fuzzy logic assessment of ability.
- Weights in last five.
- Barriers in last five.
- Distances raced in last five.

**The fourth step is to rank the relevant variables from best to worst:**

Which jockey is the best, 2<sup>nd</sup> best, 3<sup>rd</sup> best etc for this race?

Which horse in the race has the best number of starts ranked down to the worst number of starts?

I also rank age, barrier, weight carried, form position, track success rates, distance success rates, and all statistics.

Sometimes I add in a fourth step as per below. I am still evaluating the technique and feel it has a lot of merit as it both assigns a weight to a variable and compares that weight to the weight of the same variable for each other horse in the race.

**The fifth step is to establish the relevant importance of variables from best to worst in each race:**

As an alternative to Step 4, I sometimes use the following method of comparing variables in my database. I think this method has a lot of potential as it both assigns a weight to a variable and ranks that variable in comparison to other variables but using only the one column to do it (as compared to 2 columns if I assign a weight and then rank that weight separately).

First I sum the weights for each variable.

Then I divide each individual variable into the sum.

For Example: Jockey Strike Rates

Suppose that in Field One the Jockey Strike Rates are as follows:

Horse A – Jockey strike rate = .485 (achieves 48.5% placings)

Horse B – Jockey strike rate = .421

Horse C – Jockey strike rate = .334

Horse D – Jockey strike rate = .275

Horse E – Jockey strike rate = .194  
Sum of Percentages = 1.709

The rank for each jockey would be A=1, B=2, C=3, and D=4.

Suppose that in Field Two the Jockey Strike Rates are as follows:

Horse A – Jockey strike rate = .485  
Horse B – Jockey strike rate = .224  
Horse C – Jockey strike rate = .213  
Horse D – Jockey strike rate = .210  
Horse E – Jockey strike rate = .194  
Sum of Percentages = 1.326

The rank for each jockey would also be A=1, B=2, C=3, and D=4.

However, as you can see, in Field Two the jockey ranked as no1 is twice as good as the jockey ranked as no2 whereas in Field 1 there is not much difference between the two.

So, how do we express this difference in terms ChaosHunter can understand and put to good use? Simple, we express the data as follows:

Divide each individual Jockey percentage into the sum for that field and we get the following:

Field One

JOCKEY%	RELATIVE JOCKEY%
0.485	0.283792
0.421	0.246343
0.334	0.195436
0.275	0.160913
0.194	0.113517

Field Two

JOCKEY%	RELATIVE JOCKEY%
0.485	0.365762
0.224	0.168929
0.213	0.160633
0.210	0.158371
0.194	0.146305

Using this method of comparison, the top jockey in Field 2 now gets a rating more appropriate to his relative skill when compared to the other jockeys in the race.

All variables that have comparative merit can be treated this way, especially any variable with a percentage figure.

**One final point: keep your data set to a minimum!**

I've found that when using a large data file with many rows of patterns and many columns of variables, that the outputs can end up scrambled. It seems that when ChaosHunter loads or writes the file that some of the data goes missing and the formulas also seem to end up flawed.

Now, I don't know if the data goes missing on loading for the analysis and if that affects the formulas or whether there is some other trivial bug in the software but I do find that when I keep the data set to the bare minimum for analysis, especially in terms of number of columns of variables, then these flaws do not occur.

### **Things to predict**

1. Probability of finishing 1,2,3, or 4+
2. Win dividends
3. Place dividends
4. Finish Position – use favouritism and actual 123

### **Strategy Suggestions.**

- Develop one formula to predict the horse most likely to win. This will produce the top selection. In my case this yields a top selection with 24.6% winners.
- Remove that selection from the data and develop a second formula for predicting the second most likely horse to win. In my case this yields a 2<sup>nd</sup> top selection with 17.2% winners.
- Remove that selection also and develop a third formula for predicting the third most likely horse to win. In my case this yields a 3<sup>rd</sup> top selection with 17.7% winners.
- Group all 3 likely winners together, then develop a new formula using the classify function to identify which of these horses is most likely to be a winner. In my case this yields selections with a strike rate of 40% winners and 80% finishing in the first 3 (i.e. 1<sup>st</sup>, 2<sup>nd</sup> or 3<sup>rd</sup>). There are selections in roughly 5 out of 10 races and many of these bets come in at very good prices.

These methods will produce a very different range of predictions to a method that tries to predict winners using only the one formula. This is potentially very useful for exotic betting for bets such as the TREBLE, PICK SIX and QUADDIE.

The theory behind this idea is that some horses win on form and some on consistency. The PICK SIX often contains one field with a very large number of horses and in large fields, horses often win on form rather than consistency. As mentioned earlier, the strike rate for favourites is much lower in larger fields.

### **ChaosHunter Options**

After many hours of analysis using the same settings tested by Steve, I've developed a preference for using a slightly different range of options. The reason for using these different options is that ChaosHunter seems to develop better formulas when it has these options to use. Without them, it sometimes makes very little progress and the formulas don't work well at all.

Arithmetic: All

Algebra: -x and 1/x



Trig etc: Sin and Cos  
Neural: Neuron2 and Neuron3  
Boolean: None  
Relational: None  
Polynomials: All  
Statistical: All

Max equation, Max Constants etc as recommended by Steve.  
Population Size, Random Number Seeds as recommended by Steve. Occasionally I will resort to a population size of 1,000 when progress seems difficult.

I've hooked together several comps in a network. One workhorse uses an AMD Quad CPU and I find that the 4 processors on this do a mighty good job at a low cost.

Optimisation Strategy - I can't seem to get the Swarm Optimisation to work very well (except in the very early stages of development) and it usually gets stuck for long periods with no improvement whereas Evolution Strategy makes easy progress. Is that because the problem is hard? I don't know, but it works better on this kind of problem so I use it and seldom ever use Swarm.