

## 6.034 Recitation 7: Evolutionary Algorithms (Oct. 24, 25)

Kimberle Koile and Howard Shrobe

In this recitation, we'll look at the use of evolutionary algorithms to discover effective programs. Suppose that you wanted to make some extra income by betting on the horse races; how would you go about doing this?

Most people who "play the ponies" study the results of prior races and try to come up with some rules of thumb such as: Bet on a horse with a winning streak; Bet on the horse with the best pole position; Bet on a horse that's good at the distance of the race. This sounds like a natural for a rule based system, and it is; but in this case we're going to try to use evolutionary algorithms to discover the rules rather than relying on a purported expert. (There are two seminal papers on this approach: "SEAGUL Visits the Race Track", by Michael de la Maza, Proceedings of the 3<sup>rd</sup> International Conference on Genetic Algorithms, 1989, pp 208 – 212 which seems difficult to find online and "Eddie Beats the Bookies", by Tsang, Butler and Li", Journal of Software – Practice and Experience" Vol. 28(10), August 1998, 1033 – 1043 which is easily retrieved).

The idea we'll investigate is how to use genetic programming techniques to develop our rule base. First of all, we're going to need a data base of facts about racing; this will include the raw data about a corpus of races, including:

- Which horses ran
- Which jockeys rode which horses
- Which horse won
- Length of the race
- ...

A sample rule for picking a winner might say (in English): If the horse is good at the track length and the jockey is familiar with the horse and the horse is on a winning streak, then the horse is a good bet to win.

I. Write a rule that encodes this advice

II. Now suppose that this rule is representative of our "genome". What do you suppose might play the role of the individual "genes" that our evolutionary algorithm would act on.

III. What would represent the basic operations of evolutionary algorithms: Cross-over, Mutation (and Gene duplication)?

IV. What would you use for a measure of fitness?

V. What would you use for your initial population?

VI. Our original "rule of thumb" included notions like "winning streak", "horse is good at a track length", jockey is familiar with the horse. None of these notions are actually present in the data base of facts about past races. How might an evolutionary algorithm invent these notions? Be creative.

VII. Now suppose that instead of a rule, you are trying to form a mathematical expression that decides how much to bet on each horse. We'll let this expression include a non-mathematical expression like "(If <predicate> then <expression> else <expression>)" (i.e. we're using Scheme expressions). How might a genetic algorithm work in this case.

By the way, here is a chart from the paper "EDDIE beats the Bookies" that shows the results of using genetic programming to develop a program to place bets on horses. The authors have gone on to trying to develop software to place bets (or investments as they are called technically) on the stock market.

System	Number of Bets	Number of Winning Bets	Strike Rate	ROI
EDDIE-1	17	4	23.5%	88.2%
Favourites	21	6	28.6%	44.9%
Handicapper	28	3	10.7%	-63.4%
Chance	30	0	00.0%	-100.0%

*Figure 5. Results of testing EDDIE-1 on 30 handicap races*