Andrew Pyzdek
4/30/09
Math466

Linear Regression of Progressive Percentage in World of Warcraft

The Gambler's Folly is a common statistical misconception. In this, the gambler is playing a game and losing unusually badly. At that point, his average return is lower than the expectation over time. However, this leads him to believe that he is "due" for a win. Being educated in statistics, however, we can tell him that his future games are independent of his past results. World of Warcraft, though, has a system that makes the Gambler's Folly not be a folly.

In World of Warcraft, quests are used to break up the game. A common type of quest falls into the category of "Get X number of items from Y creature." These quests can be equated to a set of Bernoulli trials. Under this analogy, a quest might consist of "perform trials until you reach 10 successes," where a trial is killing the monster and a success is getting the desired item. Originally, as with most Bernoulli trials, the probability for success was a constant, independent of time or previous results. According to Nick Breckon, from his article on the phenomena, "Originally, Blizzard settled on a standard WoW quest item drop rate of around 35%, with no system in place that would tweak the percentage as players killed more enemies." In the most recent expansion, though, they implemented a new system that changes the chance of success based on the number of trials since the last success. In this paper, I will detail the method in which I gathered, analyzed, and verified data, and ultimately give a linear regression that expresses the probability of a success as a function of the number of unsuccessful trials since the last success.

We began by working under the assumption that the trials are not independent. As such, the order in which the results arose is important. Additionally, the number of successes since a "reset" was also of note. In the context of World of Warcraft, a "reset" consisted of abandoning the current quest and accepting it again, which is a rather arbitrary action. A reset, in the context of Bernoulli trials, would simply be to restart the numbering for the experiment, which is similarly arbitrary. For example, if one numbers coin flips 1, 2, 3, and 4, then began again with 1, 2, 3, 4, that would constitute a reset. The goal with this reset was to track if the probability was related to the number of total trials. Ultimately, a table was set up where 50 samples would be taken, with each result noted, then another set of 50 trials would be produced. The Appendix contains the result of this data collection with 0 representing a failure and 1 representing a success, under the Bernoulli terminology. In 200 total trials, there were 93 successes. Overall, this gave a 46.5% chance of success. This is above the probability from before the expansion, but World of Warcraft designer Jeffrey Kaplan stated in the article referenced earlier that "overall we needed to raise the base drop percentage to around 45%." With no exact percentage to go on, a hypothesis test at this point would be unproductive. However, we have evidence that the sample meets the criterion set in the article. Next, the analysis of this data will be given.

In the analysis of the data, we first consider the possibility that the number of trials since a reset is the driving force of the probability. As such, we use linear regression to find an equation relating $P(X)$ to k.

$P(X = 1|k) = \alpha + \beta k$
Using the equation for ß
$$\beta = \frac{\overline{xy} - \overline{x} \cdot \overline{y}}{\overline{x^2} - \overline{x}^2} = -0.0662$$
We note that this is a very small number, which suggests little correlation. We then move on to the possibility that the number of trials since the last success is the explanatory variable.

The trials are binned into categories based on *h*, where *h* is the number of trials since the last success. Then, the number of successes is determined. Table 1 shows the data arranged in this manner.

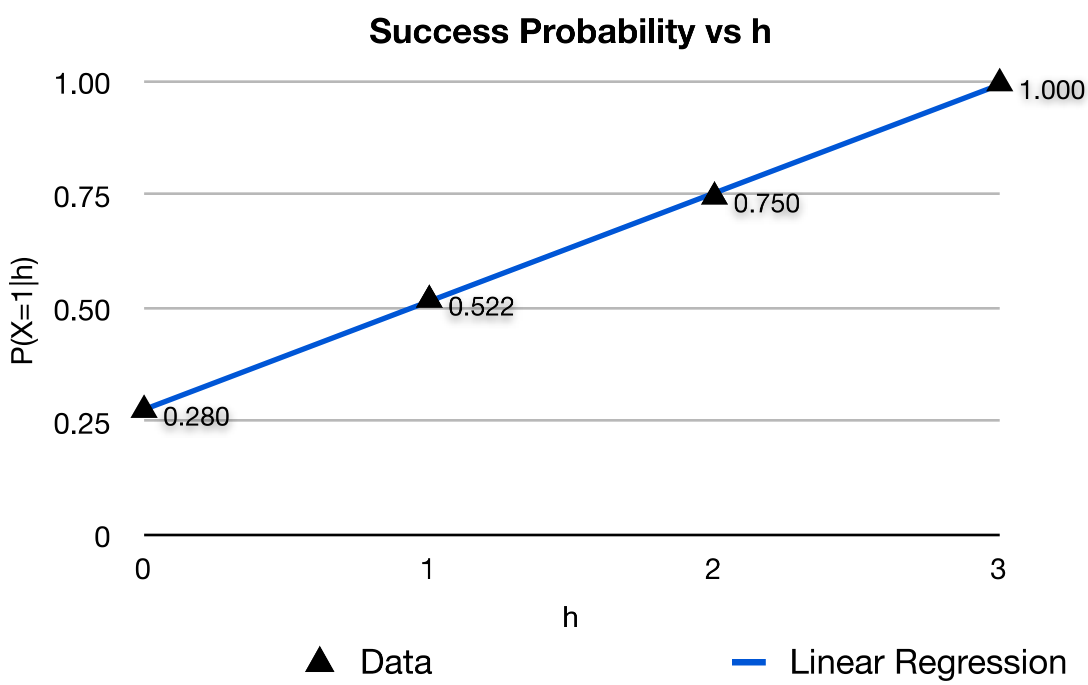

Figure 1: Linear regression of success probability with number of trials since last success as the explanatory variable.

h	0	1	2	3
Trials	93	67	32	8
Successes	26	35	24	8
P(X=1|h)	0.28	0.52	0.75	1

Table 1: Trials categorized by h, the number of trials since the last success.

At first glance, we can see some relationship between the probability of success and *h*. We now use linear regression as before, yielding the equation:
$P(X = 1|h) = 0.28 + 0.239h$

We consider the fact that our response variable is a probability, and enforce the limit that *h* is less than or equal to 3. The fit is qualitatively very good, as seen in Figure 1. However, we want a quantitative assessment of how good the linear regression fits. Thus, we consider the R-squared test for the linear regression.

$$R^2 = \frac{SS_{reg}}{SS_{tot}} = .9997$$

This value indicates a very strong relationship between the data and the linear regression. The analysis of this data, however, does not imply causation, but does strongly suggest correlation.

The results of the linear regression analysis are consistent with our prior knowledge of the game. It is stated in the article that the probability of a success can reach 100% under the right circumstances. That was, indeed, observed in the results. However, it is worth noting that the sample size for h=3 was only 8. As such, this value has a higher likelihood of error. Also, the analysis gave equal weighting to each of the sets of samples. This may be inappropriate because there are more samples for h=0, so the value given is more precise. Additionally, the information is only gathered from a single quest, “Where Dragons Fell” in Icecrown, so it may not be representative.

There are some valuable uses for this information. First, from a game design perspective, this way of handling the probabilities gives a very consistent result with no long streaks of failures. This can be good for customer satisfaction, as the players never have to deal with bad streaks, but the designer can still maintain an overall success rate that is to their liking. Second, from the perspective of a botter (Someone who has their character controlled automatically in direct violation of the game’s rules), this allows for easier automation of characters. For example, you can be guaranteed that if you need to get 10 successes, 40 attempts will be sufficient. Finally, from the perspective of the player, this gives a clear test to see if they are killing the correct enemy for a quest. Simply put, if more than four trials pass without a success, then there is something wrong that needs to be solved. The value of the linear regression itself is mostly a novelty, but it may prove to be useful in the future. Regardless of the uses, however, there are ways to expand upon this concept.

There are several extensions to this project that are readily apparent. First, as with any statistical endeavor, more data can be collected to improve estimates. Additionally, since the experiment was only done with a single quest being repeated, determining if this applies to other quests added in the expansion. A more ambitious project would be to determine if the older quests, those added before the latest expansion, were changed to accommodate the new system. Finally, other determining factors could be examined, such as the actual effect of a “reset” on the probability, if there is one. From a different angle, that of the game producer, one could extend this project by applying the concept of a progressive probability to less structured systems. Ultimately, a quest is a structure where you want one specific item from one specific enemy, and it is easy to manipulate this. However, some items are not so easy to identify, or may be accessible to multiple people. Finding a way to extend this concept to situations with more than two results and more than one parameter could be an exciting endeavor.

Appendix

This appendix gives the raw data as collected in the project. It is broken into four sets of 50 trials, with the bold values representing when quest items were destroyed. The number of quest items in inventory was not used as an explanatory variable, but could be examined using the data as collected here.

Trial\K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	0	1	1	0	0	1	1	1	0	0	**1**	0	0	1	0	1	0	1	0	1	0	0	1
2	0	0	1	1	1	0	1	0	0	0	1	0	0	0	1	0	**1**	0	1	0	0	1	0
3	1	0	1	0	0	1	0	1	0	1	0	**1**	1	0	0	1	1	1	0	0	1	0	1
4	0	1	0	0	0	1	1	0	1	1	0	**1**	0	1	1	0	0	1	0	1	1	0	**1**

24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
0	**1**	0	0	0	1	1	0	0	1	0	1	0	0	0	1	0	**1**	0	0	1	1	0	1	0	0	1
1	0	1	1	0	0	**1**	0	1	1	0	0	1	0	1	0	1	0	0	**1**	0	0	0	1	0	0	0
0	0	1	0	0	1	0	1	1	0	1	0	0	**1**	1	1	1	0	0	1	0	0	1	0	**1**	0	1
0	0	0	1	1	0	1	1	1	**1**	1	0	1	0	1	0	0	1	0	1	0	0	**1**	0	0	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	0	1	1	1	1	0	1	0	0	1	1	0	0	1

Breckon, Nick. “Blizzard Details Secret World of Warcraft 'Progressive Percentage' Item Drop Mechanic”. Shack News. <http://www.shacknews.com/onearticle.x/57886> 26 Mar. 2009.

“Coefficient of Determination”. *Wikipedia*. <http://en.wikipedia.org/wiki/Coefficient_of_determination> 29 April, 2009.