Markov Model Skill Distribution PP Rework

Peter Gregory Husisian

June 2019

1 Notation

Let:

1. $\forall i \leq n, c_i = (x_i, t_i)$ be represent the ith hitcircle with location x_i and time t_i.
2. s be a d dimensional skill vector, meant to represent the skill of a player
3. $\forall i \leq n, h_i$ be the random indicator variable that is one if a player hits the ith hitcircle, and is zero otherwise.
4. $R(c)$, where c is a circle, is the region of cursor positions that land within the hit-circle for c.

For any variable that may be indexed into, such as c_i, for sake of notational compactness, if an index is omitted, it is the equivalent of writing all elements of the sequence (e.g. $c = c_1, \dots, c_n$). We also define a subset of indices by subscripting $i : j$, where i represents the beginning index to be included in the subset, and j represents the ending index to be included. For example: $c_{i:j} = c_i, \dots, c_j$.

2 Introduction

We seek the probability distribution over a sequence of hits/misses (the h_i's) given the map (the c_i's):

$$p(h|c) = \int_{\forall s} p(h, s|c)ds = \int_{\forall s} p(h|s, c)p(s)ds$$

Where $s \in \{(y_1, \dots, y_d)|\forall i, y_i \geq 0\}$. Intuitively, the magnitude of s ought to represent a measure of overall skill, and one could think of the overall skill of a map being the sum of skills required to hit each circle of the map. This seemingly odd requirement that s have elements greater than or equal to zero is to guarantee that if any two skill vectors are added, the result will be of greater or equal magnitude than either were on their own, as no components can be negative.

The justification for including a skill vector at all is that it allows for additional interpretability – that is, we may solve for the skill vector that maximizes the probability of FC to get an idea for the diversity of the skill set required to FC the map. Additionally, it forces the model to "pick" which skills are most important to represent when determining the probability of FC. While s is latently defined by the model and may not easily converted into weightings of traditional skills such as reading and aim, it's finite dimension prevents niche and not necessarily skillful, but statistically unlikely, skills from skewing the distribution toward overweighting the PP for a map requiring those skills. Further, it better models the real-life scenario of some player, with skill distribution s, attempting to FC a map given his skill.

3 Defining $p(s)$

As s is latent, we may choose any probability distribution for s, and once the model is fit, the resulting skill should fit that distribution as optimally as possible. However, one strong candidate is the unit Normal Distribution, renormalized so that it is supported only for vectors with positive components:

$$p(s) = \frac{2^d}{\sqrt{(2\pi)^d}} e^{-\|s\|_2^2}$$

4 Defining $p(h|s,c)$

The joint probability is given by:

$$
\begin{aligned}
p(h|s,c) &= p(h_1|s,c_1)p(h_2|s,h_1,c_1,c_2)\dots p(h_n|s,h_{1:n-1},c_{1:n}) \\
&= \prod_{i=1}^{n} p(h_i|s,h_{1:i-1},c_{1:i})
\end{aligned}
$$

The factors of this distribution become increasingly complex as the number of notes in the map increases. However, it is reasonable to assume that the notes and hits nearest to h_i play a significantly larger role than notes nowhere near c_i and hits nowhere near h_i. If we assume that the nearest k hits and notes to i are all that is relevant when calculating $p(h_i|s,h_1,\dots,h_{i-1},c_1,\dots,c_i)$, we yield the following "kth order Markov Chain" approximation to $p(h|s,c)$:

$$p(h|s,c) \approx p(h_1|s,c_1)\dots p(h_k|s,h_{1:k-1},c_{1:k}) \prod_{i=k+1}^{n} p(h_i|s,h_{i-k:i-1},c_{i-k:i})$$

The factors of this approximation are significantly simpler as they only require conditioning on $2k$ parameters, rather than the skill and all hits and notes prior to h_i. Further, when performing Maximum Likelihood optimization, we need only solve for the parameters for the far simpler conditional distribution $p(h_i|s,h_{i-k:i-1},c_{i-k:i})$ rather than the joint distribution as a whole. As there are n observations per replay that can be used to train the conditional distribution, a replay can only be used as one observation when being used to train the joint distribution as a whole as would be necessary if the Markov Assumption approximation were not leveraged. osu! seems a perfect game to leverage the Markov Assumption, as its core gameplay is simple enough (in that a player is attempting only to click a small number of circles that appear on screen at a given point) that one could say with extreme confidence there exists some small k for which the kth order Markov Model approximation to the joint distribution is almost exactly equal to the joint distribution itself.

5 Choosing an Ideal Distribution for $p(h_i|s,h_{i-k:i-1},c_{i-k,i})$

There are multiple ways of going about this, and any function that maps the parameters conditioned upon into a probability between 0 and 1 could be leveraged. One possible method is to define a distribution over f_i, the cursor position of the player at the time t_i given the skill and prior notes and hits:

$$p(f_i|s,h_{i-k:i-1},c_{i-k:i}) = ...$$

Intuitively, this distribution is likely radial, and likely centered at x_i. One potential candidate is a Normal Distribution whose mean and covariance matrix are a function of $s, h_{i-k:i-1}, c_{i-k,i}$. Another might be a Von Mises distribution modified to add a radial component (this could account for the fact that it is difficult to aim at extreme corners on a tablet, etc. by the egg-shaped form of a Von Mises distribution).

We then have:

$$p(h_i|s,h_{i-k:i-1},c_{i-k,i}) = \int_{f_i \in R(c_i)} p(f_i|s,h_{i-k:i-1},c_{i-k:i})df_i$$

But it is possible that this definition of an additional distribution over cursor positions is unnecessary, and that there exists a means of determining $p(h_i|s,h_{i-k:i-1},c_{i-k:i})$ through some function of its conditioned parameters.

6 Determining the Optimal Values for the Parameters of $p(h_i|s,h_{i-k:i-1},c_{i-k:i})$

The functions that assign the values of the parameters of this distribution will be determined through Maximum Likelihood Estimation using replay data on observed sequences of hits/misses and cursor positions (if used by the conditional distribution).