A Way To Analyze the Number Guessing Game

Gautam Ramasubramanian and Professor Anand Sarwate

Rutgers University, 604 Bartholomew Rd, Piscataway Township, NJ 08854,
`ramasubg2@gmail.com`,
WWW home page: `http://reu.dimacs.rutgers.edu/~gautamr/`

Abstract. In this paper, I introduce the number guessing game, as well as the Bayesian scheme used to analyze the game. The goal is to come up with a way to find the optimal guess at each stage of the game. This paper introduces the risk function, and claims that the guess that minimized the expected risk is the best one to make.

Keywords: Bayesian Statistics, Risk Function, Random Process

1 The Number Guessing Game

Two entities are playing a simple game. One player thinks of a number from 1 to n, and the other player tries to guess at the number repeatedly. The first player tells the second if the guess is too high or too low, until the second player guesses correctly.

A cost is associated with each verdict of "Too High" or "Too Low". Let us call them α and β respectively. The cost accumulates as the first player keeps guessing. The goal is to come up with the best guessing strategy that can minimize the cost.

2 Prior Knowledge

It is known that if $\alpha = \beta$, then the best strategy is Binary Search of the numbers between 1 and n, which minimizes the number of guesses the first player has to make overall. It is also known that if one of the costs, say α, is 0, then the best strategy is linear search. Since α is the cost of getting a "Too High", we should start the linear search from n, and keep getting "Too High" verdicts until reaching the correct number. Likewise, if β is 0, then the best strategy is linear search starting from 1.

It is helpful to think of these as special cases, when $\frac{\alpha}{\beta} = 1$ and when $\frac{\alpha}{\beta} = 0, \infty$.

3 Bayesian Scheme

Although the correct number, which we will call θ, is created and known by the first player, it is not known to the second player, so from the second player's perspective, θ is a discrete random variable with uniform distribution on the set $1, 2, \ldots n$. We can say that second player looks at this distribution and produces a guess θ_g, which is then communicated to the first player. The first player then formulates a verdict v, which is either "Too High" or "Too Low", and based on that the probability distribution for θ changes. More specifically, it changes using the Bayesian scheme, characterized by the following equation, known as *Bayes' Rule*.

$$P(\theta|v) = \frac{P(v|\theta)P(\theta)}{P(v)}$$

In this case, v, the verdict, and θ_g, the guess, are both known. For simplicity's sake, let us consider the case when v is "Too High". $P(\theta)$ is given by the uniform probability distribution from 1 to n as $\frac{1}{n}$. $P(v|\theta)$ is the probability of a zero verdict *when θ is given*. This is either 0 or 1, depending on whether $\theta \geq \theta_g$ or $\theta < \theta_g$ respectively. Finally, $P(v)$ can be represented as

$$\sum_{\theta_0=1}^{n} P(v|\theta = \theta_0)P(\theta = \theta_0) = \sum_{\theta_0=1}^{\theta_g-1} P(\theta = \theta_0) = \sum_{\theta_0=1}^{\theta_g-1} \frac{1}{n} = \frac{\theta_g - 1}{n}$$

Therefore, we can conclude that the posterior probability distribution of θ after the second player's guess is

$$P(\theta|v = \text{Too High}) = \frac{\frac{1}{n}}{\frac{\theta_g-1}{n}} = \frac{1}{\theta_g - 1} \ (1 \leq \theta \leq \theta_g)$$

Similarly, it can be shown that when the verdict, v is "Too Low"

$$P(\theta|v = \text{Too Low}) = \frac{\frac{1}{n}}{\frac{n-\theta_g}{n}} = \frac{1}{n - \theta_g} \ (\theta_g + 1 \leq \theta \leq n)$$

4 How To Choose θ_g

We know the initial probability distribution of the discrete random variable θ. We know what will happen to the the probability distribution after choosing a guess θ_g. Now, the fundamental question is how to choose θ_g, given the dirstribution of *theta*. The choice should take into account two things.

1. The cost of each verdict, α and β (cost)
2. The minimization of the number of remaining choices. This model only looks at the formation of one guess in one turn, so the fact that the cost accumulates as the game progresses must be taken into account. (minimization of version space)

To do that, we introduce an idea of *risk function*. This risk function is actually a random process, a collection of random variables, functions of θ, that are parametrized by θ_g. A better way to think about it is that

$$\begin{aligned} risk &= f(\theta, \theta_g) \\ E[risk] &= g(\theta_g) \end{aligned}$$

The goal is to define the *risk* function such that it includes the cost and minimization of version space and it agrees with prior knowledge. My hypothesized risk function is as follows.

$$risk = f(\theta, \theta_g) = \begin{cases} (\theta_g - 1)\alpha^\gamma & \theta < \theta_g \\ (n - \theta_g)\beta^\gamma & \theta > \theta_g \\ 0 & \theta = \theta_g \end{cases}$$

γ is a positive constant. It measures how important it is to minimize the hypothesis space vs minimizing the α-β cost. If γ is greater than 1, then it is more important to minimize the α-β cost. If less than 1, then vice-versa. The way to choose $theta_g$ is to find the number that minimizes the expected risk $E[risk]$. We can take the expectation of the risk, differentiate it with respect to θ_g, find the value of θ_g for which the derivitive is 0. We can take the second derivative if it is neccessary to prove θ_g is a minimium risk guess rather than a maximum risk guess.

5 Binary Search Example

Suppose $\alpha = \beta = k > 0$. The risk function will now be

$$risk = f(\theta, \theta_g) = \begin{cases} (\theta_g - 1)k^\gamma & \theta < \theta_g \\ (n - \theta_g)k^\gamma & \theta > \theta_g \\ 0 & \theta = \theta_g \end{cases}$$

To calculate expected risk, we use the definition of expected value.

$$\begin{aligned} E[risk] &= \sum_{\theta_0=1}^{n} f(\theta, \theta_g) P(\theta = \theta_0) \\ &= \sum_{\theta_0=1}^{\theta_g - 1} (\theta_g - 1)k^\gamma \cdot P(\theta = \theta_0) + \sum_{\theta_0=\theta_g}^{\theta_g} 0 \cdot P(\theta = \theta_0) + \sum_{\theta_0=\theta_g+1}^{n} (n - \theta_g)k^\gamma \cdot P(\theta = \theta_0) \\ &= \sum_{\theta_0=1}^{\theta_g - 1} (\theta_g - 1)k^\gamma \cdot \frac{1}{n} + \sum_{\theta_0=\theta_g}^{\theta_g} 0 \cdot \frac{1}{n} + \sum_{\theta_0=\theta_g+1}^{n} (n - \theta_g)k^\gamma \cdot \frac{1}{n} \\ &= (\theta_g - 1)k^\gamma \cdot \frac{\theta_g - 1}{n} + (n - \theta_g)k^\gamma \cdot \frac{n - \theta_g}{n} \\ &= \frac{(\theta_g - 1)^2\, k^\gamma}{n} + \frac{(n - \theta_g)^2\, k^\gamma}{n} \end{aligned}$$

If we take the derivative, we get the following equation.

$$\frac{d}{d\theta_g}E[risk] = 2 \cdot \frac{(\theta_g - 1)\, k^\gamma}{n} - 2 \cdot \frac{(n - \theta_g)\, k^\gamma}{n} = 0$$

This reduces to the following equation

$$\begin{aligned} \cdot \frac{(\theta_g - 1)\, k^\gamma}{n} &= 2 \cdot \frac{(n - \theta_g)\, k^\gamma}{n} \\ \theta_g - 1 &= n - \theta_g \\ \theta_g &= \frac{n+1}{2} \end{aligned}$$

This indicates if α and β are equal, then the best guess is the median, which corresponds to binary search.

6 Generalized Formulas

The general equation for the expected risk, derivative and θ_g are

$$\begin{aligned} E[risk] &= \frac{(\theta_g - 1)^2\, \alpha^\gamma}{n} + \frac{(n - \theta_g)^2\, \beta^\gamma}{n} \\ \frac{d}{d\theta_g}E[risk] &= 2 \cdot \frac{(\theta_g - 1)\, \alpha^\gamma}{n} - 2 \cdot \frac{(n - \theta_g)\, \beta^\gamma}{n} \\ \theta_g &= \frac{\beta^\gamma n + \alpha^\gamma}{\alpha^\gamma + \beta^\gamma} = \frac{\beta^\gamma}{\alpha^\gamma + \beta^\gamma} \cdot n + \frac{\alpha^\gamma}{\alpha^\gamma + \beta^\gamma} \end{aligned}$$

The optimal θ_g is the weighted average of 1 and n, weighted by α^γ and β^γ.

Notes and Comments.

The plan is to use empirical data to find out what γ is.

If it turns out that γ cannot be expressed as a constant, or as a simple function of n, α and β, then the model is wrong or needs to be modified. In any case, we are assuming that the *risk* has a particular form - the one that has been presented. That may not be the best form, so further work must be done.

References

1. Dasgupta S. *Two faces of active learning.* (2009)