Random Processes

By:
Nick Kingsbury

Random Processes

By:
Nick Kingsbury

Online:
< http://cnx.org/content/col10204/1.3/ >

C O N N E X I O N S

Rice University, Houston, Texas

Collection structure revised: January 22, 2004
PDF generated: October 25, 2012
For copyright and attribution information for the modules contained in this collection, see p. 50.

Table of Contents

Chapter 1

Probability Distributions

1.1 Aims and Motivation for the Course[1]

We aim to:

- Develop a theory which can characterize the behavior of **real-world Random Signals and Processes**;
- Use standard **Probability Theory** for this.

Random signal theory is important for

- Analysis of signals;
- Inference of underlying system parameters from noisy observed data;
- Design of optimal systems (digital and analogue signal recovery, signal classification, estimation ...);
- Predicting system performance (error-rates, signal-to-noise ratios, ...).

Example 1.1: Speech signals
Use probability theory to characterize that some sequences of vowels and consonants are more likely than others, some waveforms more likely than others for a given vowel or consonant. Please see Figure 1.1.

Use this to achieve: speech recognition, speech coding, speech enhancement, ...

[1] This content is available online at <http://cnx.org/content/m10983/2.4/>.

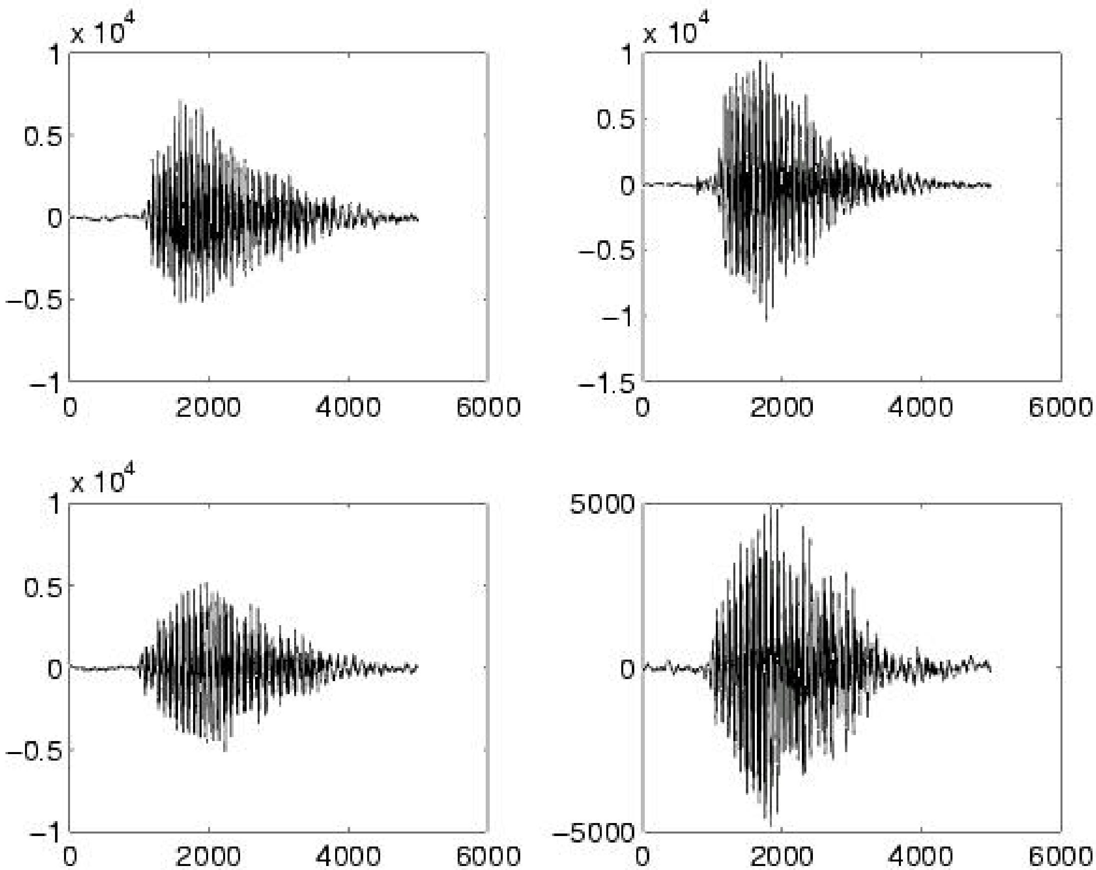

Figure 1.1: Four utterances of the vowel sound 'Aah'.

Example 1.2: Digital communications
Characterize the properties of the digital data source (mobile phone, digital television transmitter, ...), characterize the noise/distortions present in the transmission channel. Please see Figure 1.2.

Use this to achieve: accurate regeneration of the digital signal at the receiver, analysis of the channel characteristics ...

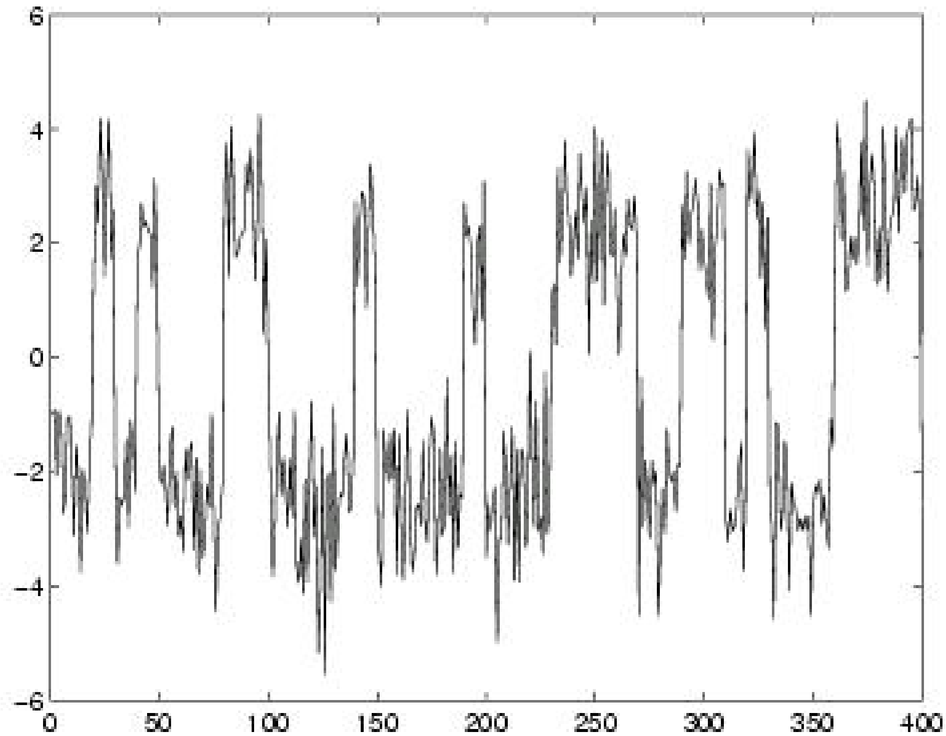

Figure 1.2: Digital data stream from a noisy communications Channel.

Probability theory is used to give a mathematical description of the behavior of real-world systems which involve elements of **randomness**. Such a system might be as simple as a coin-flipping experiment, in which we are interested in whether 'Heads' or 'Tails' is the outcome, or it might be more complex, as in the study of random errors in a coded digital data stream (e.g. a CD recording or a digital mobile phone).

The basics of probability theory should be familiar from the IB Probability and Statistics course. Here we summarize the main results from that course and develop them into a framework that can encompass random signals and processes.

1.2 Probability Distributions[2]

The distribution P_X of a random variable X is simply a probability measure which assigns probabilities to events on the real line. The distribution P_X answers questions of the form:

What is the probability that X lies in some subset F of the real line?

In practice we summarize P_X by its **Probability Mass Function - pmf** (for discrete variables only), **Probability Density Function - pdf** (mainly for continuous variables), or **Cumulative Distribution Function - cdf** (for either discrete or continuous variables).

[2]This content is available online at <http://cnx.org/content/m10984/2.8/>.

1.2.1 Probability Mass Function (pmf)

Suppose the discrete random variable X can take a set of M real values $\{x_1, \ldots, x_M\}$, then the **pmf** is defined as:

$$\begin{aligned} p_X(x_i) &= Pr[X = x_i] \\ &= P_X(\{x_i\}) \end{aligned} \tag{1.1}$$

where $\sum_{i=1}^{M} p_X(x_i) = 1$. e.g. For a normal 6-sided die, $M = 6$ and $p_X(x_i) = \frac{1}{6}$. For a pair of dice being thrown, $M = 11$ and the pmf is as shown in (a) of Figure 1.3.

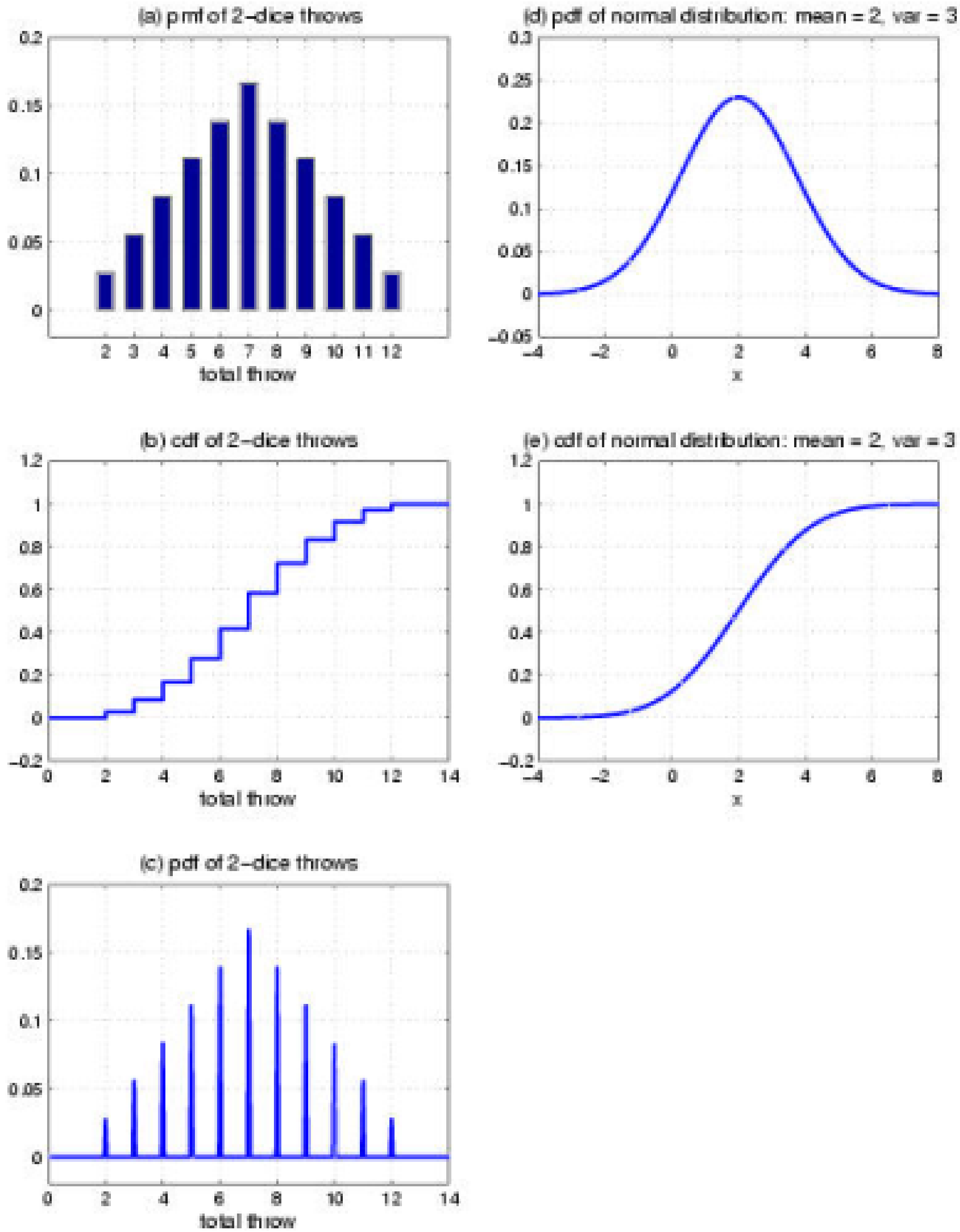

Figure 1.3: Examples of pmfs, cdfs and pdfs: (a) to (c) for a discrete process, the sum of two dice; (d) and (e) for a continuous process with a normal or Gaussian distribution, whose mean = 2 and variance = 3.

1.2.2 Cumulative Distribution Function (cdf)

The **cdf** can describe discrete, continuous or mixed distributions of X and is defined as:

$$\begin{aligned} F_X(x) &= Pr[X \leq x] \\ &= P_X((-\infty, x]) \end{aligned} \quad (1.2)$$

For discrete X:

$$F_X(x) = \sum_i \{p_X(x_i) \mid x_i \leq x\} \quad (1.3)$$

giving step-like cdfs as in the example of (b) of Figure 1.3.

Properties follow directly from the Axioms of Probability:

1. $0 \leq F_X(x) \leq 1$
2. $F_X(-\infty) = 0$, $F_X(\infty) = 1$
3. $F_X(x)$ is non-decreasing as x increases
4. $Pr[x_1 < X \leq x_2] = F_X(x_2) - F_X(x_1)$
5. $Pr[X > x] = 1 - F_X(x)$

where there is no ambiguity we will often drop the subscript X and refer to the cdf as $F(x)$.

1.2.3 Probability Density Function (pdf)

The **pdf** of X is defined as the derivative of the cdf:

$$f_X(x) = \frac{d}{dx} F_X(x) \quad (1.4)$$

The pdf can also be interpreted in derivative form as $\delta(x) \to 0$:

$$\begin{aligned} f_X(x)\,\delta(x) &= Pr[x < X \leq x + \delta(x)] \\ &= F_X(x + \delta(x)) - F_X(x) \end{aligned} \quad (1.5)$$

For a discrete random variable with pmf given by $p_X(x_i)$:

$$f_X(x) = \sum_{i=1}^{M} p_X(x_i)\,\delta(x - x_i) \quad (1.6)$$

An example of the pdf of the 2-dice discrete random process is shown in (c) of Figure 1.3. (Strictly the delta functions should extend vertically to infinity, but we show them only reaching the values of their areas, $p_X(x_i)$.)

The pdf and cdf of a continuous distribution (in this case the **normal** or **Gaussian** distribution) are shown in (d) and (e) of Figure 1.3.

NOTE: The cdf is the integral of the pdf and should always go from zero to unity for a valid probability distribution.

Properties of pdfs:

1. $f_X(x) \geq 0$
2. $\int_{-\infty}^{\infty} f_X(x)\,dx = 1$
3. $F_X(x) = \int_{-\infty}^{x} f_X(\alpha)\,d\alpha$
4. $Pr[x_1 < X \leq x_2] = \int_{x_1}^{x_2} f_X(\alpha)\,d\alpha$

As for the cdf, we will often drop the subscript X and refer simply to $f(x)$ when no confusion can arise.

1.3 Conditional Probabilities and Bayes' Rule[3]

If A and B are two separate but possibly dependent random events, then:

1. Probability of A and B occurring together $= Pr[A, B]$
2. The conditional probability of A, given that B occurs $= Pr[A \mid B]$
3. The conditional probability of B, given that A occurs $= Pr[B \mid A]$

From elementary rules of probability (Venn diagrams):

$$\begin{aligned} Pr[A, B] &= Pr[A \mid B]\, Pr[B] \\ &= Pr[B \mid A]\, Pr[A] \end{aligned} \quad (1.7)$$

Dividing the right-hand pair of expressions by $Pr[B]$ gives Bayes' rule:

$$Pr[A \mid B] = \frac{Pr[B \mid A]\, Pr[A]}{Pr[B]} \quad (1.8)$$

In problems of probabilistic inference, we are often trying to estimate the most probable underlying model for a random process, based on some observed data or evidence. If A represents a given set of model parameters, and B represents the set of observed data values, then the terms in (1.8) are given the following terminology:

- $Pr[A]$ is the **prior** probability of the model A (in the absence of any evidence);
- $Pr[B]$ is the probability of the **evidence** B;
- $Pr[B \mid A]$ is the **likelihood** that the evidence B was produced, given that the model was A;
- $Pr[A \mid B]$ is the **posterior** probability of the model being A, given that the evidence is B.

Quite often, we try to find the model A which maximizes the posterior $Pr[A \mid B]$. This is known as **maximum a posteriori** or **MAP** model selection.

The following example illustrates the concepts of Bayesian model selection.

Example 1.3: Loaded Dice

Problem:

Given a tub containing 100 six-sided dice, in which one die is known to be loaded towards the six to a specified extent, derive an expression for the probability that, after a given set of throws, an arbitrarily chosen die is the loaded one? Assume the other 99 dice are all fair (not loaded in any way). The loaded die is known to have the following pmf:

$$p_L(1) = 0.05$$

$$\{p_L(2), \ldots, p_L(5)\} = 0.15$$

$$p_L(6) = 0.35$$

Here derive a good strategy for finding the loaded die from the tub.

Solution:

The pmfs of the fair dice may be assumed to be:

$$p_F(i) = \frac{1}{6}\ , \quad i = \{1, \ldots, 6\}$$

[3]This content is available online at <http://cnx.org/content/m10985/2.8/>.

Let each die have one of two states, $S = L$ if it is loaded and $S = F$ if it is fair. These are our two possible **models** for the random process and they have underlying pmfs given by $\{p_L(1), \ldots, p_L(6)\}$ and $\{p_F(1), \ldots, p_F(6)\}$ respectively.

After N throws of the chosen die, let the sequence of throws be $\Theta_N = \{\theta_1, \ldots, \theta_N\}$, where each $\theta_i \in \{1, \ldots, 6\}$. This is our **evidence**.

We shall now calculate the probability that this die is the loaded one. We therefore wish to find the **posterior** $Pr[S = L \mid \Theta_N]$.

We cannot evaluate this directly, but we can evaluate the **likelihoods**, $Pr[\Theta_N \mid S = L]$ and $Pr[\Theta_N \mid S = F]$, since we know the expected pmfs in each case. We also know the **prior** probabilities $Pr[S = L]$ and $Pr[S = F]$ before we have carried out any throws, and these are $\{0.01, 0.99\}$ since only one die in the tub of 100 is loaded. Hence we can use Bayes' rule:

$$Pr[S = L \mid \Theta_N] = \frac{Pr[\Theta_N \mid S = L]\, Pr[S = L]}{Pr[\Theta_N]} \tag{1.9}$$

The denominator term $Pr[\Theta_N]$ is there to ensure that $Pr[S = L \mid \Theta_N]$ and $Pr[S = F \mid \Theta_N]$ sum to unity (as they must). It can most easily be calculated from:

$$\begin{aligned} Pr[\Theta_N] &= Pr[\Theta_N, S = L] + Pr[\Theta_N, S = F] \\ &= Pr[\Theta_N \mid S = L]\, Pr[S = L] + Pr[\Theta_N \mid S = F]\, Pr[S = F] \end{aligned} \tag{1.10}$$

so that

$$\begin{aligned} Pr[S = L \mid \Theta_N] &= \frac{Pr[\Theta_N \mid S=L] Pr[S=L]}{Pr[\Theta_N \mid S=L] Pr[S=L] + Pr[\Theta_N \mid S=F] Pr[S=F]} \\ &= \frac{1}{1+R_N} \end{aligned} \tag{1.11}$$

where

$$R_N = \frac{Pr[\Theta_N \mid S = F]\, Pr[S = F]}{Pr[\Theta_N \mid S = L]\, Pr[S = L]} \tag{1.12}$$

To calculate the likelihoods, $Pr[\Theta_N \mid S = L]$ and $Pr[\Theta_N \mid S = F]$, we simply take the product of the probabilities of each throw occurring in the sequence of throws Θ_N, given each of the two modules respectively (since each new throw is independent of all previous throws, given the model). So, after N throws, these likelihoods will be given by:

$$Pr[\Theta_N \mid S = L] = \prod_{i=1}^{N} p_L(\theta_i) \tag{1.13}$$

and

$$Pr[\Theta_N \mid S = F] = \prod_{i=1}^{N} p_F(\theta_i) \tag{1.14}$$

We can now substitute these probabilities into the above expression for R_N and include $Pr[S = L] = 0.01$ and $Pr[S = F] = 0.99$ to get the desired a **posteriori** probability $Pr[S = L \mid \Theta_N]$ after N throws using (1.11).

We may calculate this iteratively by noting that

$$Pr[\Theta_N \mid S = L] = Pr[\Theta_{N-1} \mid S = L]\, p_L(\theta_n) \tag{1.15}$$

and

$$Pr[\Theta_N \mid S = F] = Pr[\Theta_{N-1} \mid S = F]\, p_F(\theta_n) \tag{1.16}$$

so that

$$R_N = R_{N-1} \frac{p_F(\theta_n)}{p_L(\theta_n)} \tag{1.17}$$

where $R_0 = \frac{Pr[S=F]}{Pr[S=L]} = 99$. If we calculate this after every throw of the current die being tested (i.e. as N increases), then we can either move on to test the next die from the tub if $Pr\left[S = L \mid \Theta_N\right]$ becomes sufficiently small (say $< \left(10^{-4}\right)$) or accept the current die as the loaded one when $Pr\left[S = L \mid \Theta_N\right]$ becomes large enough (say $> (0.995)$). (These thresholds correspond approximately to $R_N > 10^4$ and $R_N < 5 \times 10^{-3}$ respectively.)

The choice of these thresholds for $Pr\left[S = L \mid \Theta_N\right]$ is a function of the desired tradeoff between speed of searching versus the probability of failure to find the loaded die, either by moving on to the next die even when the current one is loaded, or by selecting a fair die as the loaded one.

The lower threshold, $p_1 = 10^{-4}$, is the more critical, because it affects how long we spend before discarding each fair die. The probability of correctly detecting all the fair dice before the loaded die is reached is $(1 - p_1)^n \simeq 1 - np_1$, where $n \simeq 50$ is the expected number of fair dice tested before the loaded one is found. So the failure probability due to incorrectly assuming the loaded die to be fair is approximately $np_1 \simeq 0.005$.

The upper threshold, $p_2 = 0.995$, is much less critical on search speed, since the **loaded** result only occurs once, so it is a good idea to set it very close to unity. The failure probability caused by selecting a fair die to be the loaded one is just $1 - p_2 = 0.005$. Hence the overall failure probability $= 0.005 + 0.005 = 0.01$

NOTE: In problems with significant amounts of evidence (e.g. large N), the evidence probability and the likelihoods can both get very very small, sufficient to cause floating-point underflow on many computers if equations such as (1.13) and (1.14) are computed directly. However the ratio of likelihood to evidence probability still remains a reasonable size and is an important quantity which must be calculated correctly.

One solution to this problem is to compute only the ratio of likelihoods, as in (1.17). A more generally useful solution is to compute log(likelihoods) instead. The product operations in the expressions for the likelihoods then become sums of logarithms. Even the calculation of likelihood ratios such as R_N and comparison with appropriate thresholds can be done in the log domain. After this, it is OK to return to the linear domain if necessary since R_N should be a reasonable value as it is the **ratio** of very small quantities.

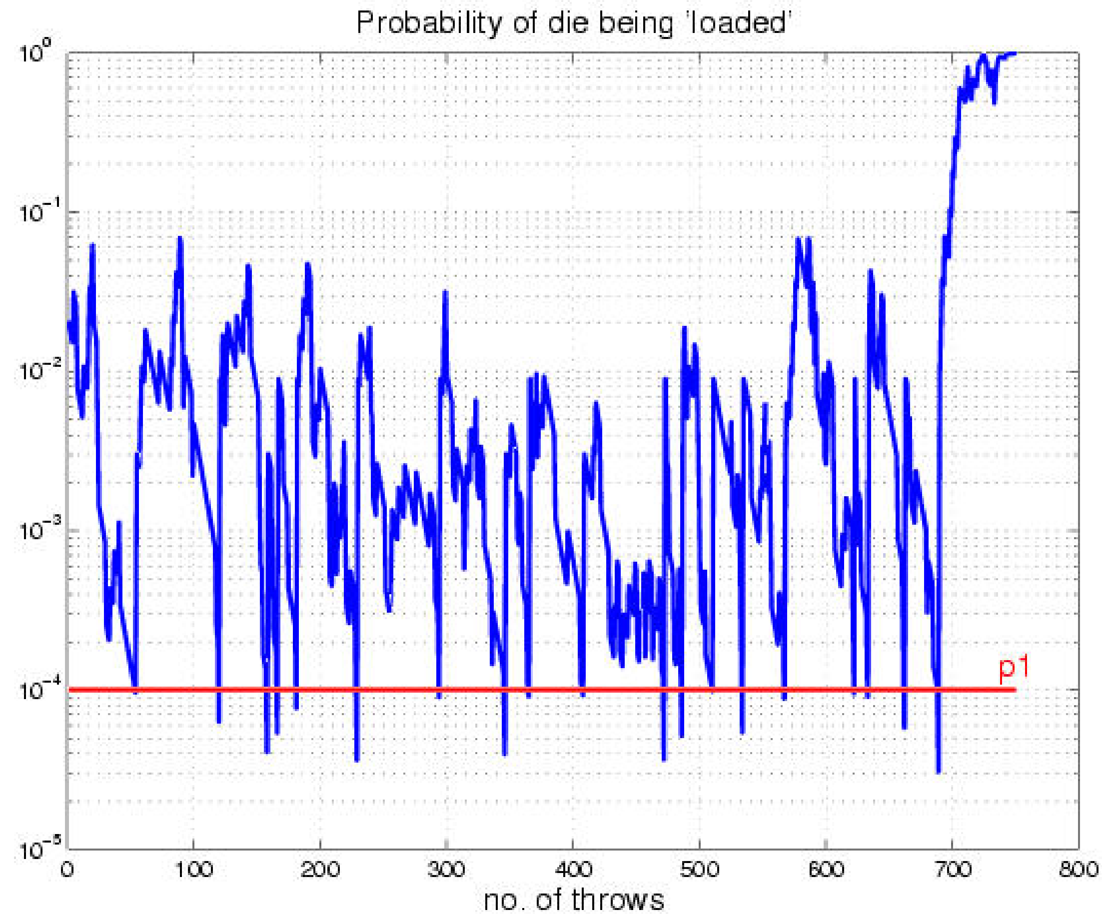

Figure 1.4: Probabilities of the current die being the loaded one as the throws progress (20th die is the loaded one). A new die is selected whenever the probability falls below p_1.

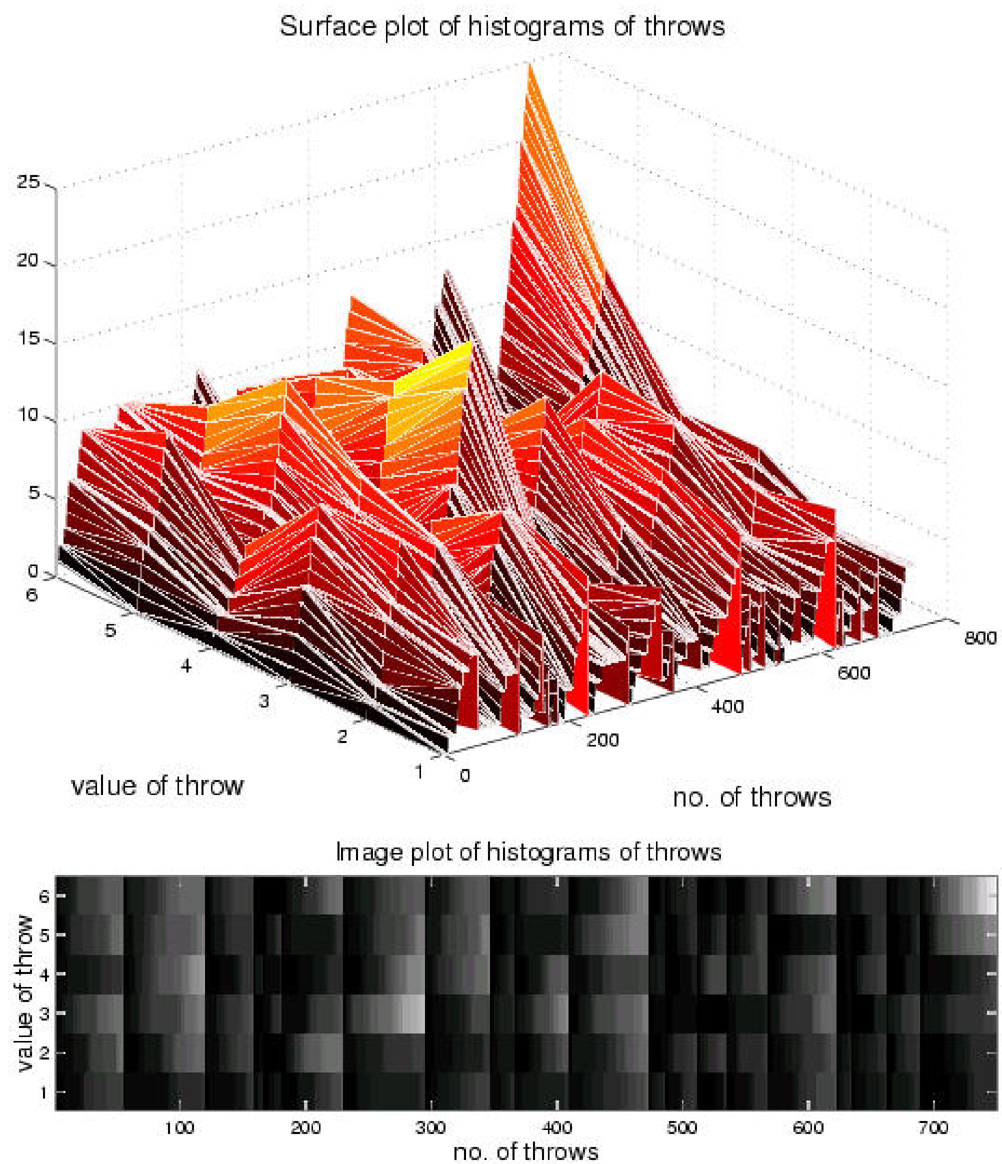

Figure 1.5: Histograms of the dice throws as the throws progress. Histograms are reset when each new die is selected.