**АРИФМЕТИЧЕСКОЕ КОДИРОВАНИЕ**

## Использование чисел в качестве кодовых слов

Техника, известная как арифметическое кодирование, основана на соответствии между бинарными словами (элементами множества ) и дробями (элементами множества  рациональных чисел).

Для любого слова  из  существует рациональное число .

Это число лежит между 0 и 1, и обозначается  в *двоичной системе счисления*. Таким образом, для каждого целого числа  мы имеем функцию , обозначенную с помощью .

***Пример 1*** Запишем рациональные числа, соответствующие словам 0110101 и 1010000.

*Решение* Здесь . Слово 0110101 соответствует рациональному числу , или 0,4140625 в обычном (десятичном) обозначении.

Аналогично, слово 1010000 соответствует рациональному числу ,

или 0,625 в десятичной системе счисления. Обратите внимание, что нулевое окончание, как правило, опускается, так что бинарная форма будет записана в виде 0,101. Но когда важно записать значение , нулевые окончания должны быть показаны.

В арифметическом кодировании мы используем рациональные числа, чтобы определить кодовые слова, представленные строками символов. Цель состоит в том, чтобы получить коды, которые близки к оптимальным. Как обычно мы делаем это, пытаясь убедиться, что строка  с высокой вероятностью представлена кодовым словом  малой длины . Кодовые слова следовательно будут определены с точки зрения вероятностей.

Мы начинаем с множества действительных чисел , потому что вероятность события, как правило, представлена действительным числом, а не рациональным. Выберем кодовое сово  как число из соответствующего интервала вида ,

где  – действительное число и  – вероятность . Следующая теорема объясняет как мы можем выбрать рациональное число вида , небольшой длины  этого интервала.

***Теорема 1*** Предположим, что  и  такие, что . Пусть  любое целое число такое, что . Тогда существует слово  такое, что .

## Арифметическое кодирование

Мы будем использовать Теорему 1 для создания беспрефиксных бинарных кодов для строк символов, испускаемых стационарным источником. В этом случае существует распределение вероятностей в множестве  строк  длины  множества . (Для того чтобы избежать громоздких обозначений мы будем обозначать распределение на  через , при любом значении ). В общем случае мы не предполагаем свойство независимости

 *не обязательно равно* .

Будем считать, что символы в алфавите  расположены в определенном порядке

.

Используя эту последовательность, мы можем определить как упорядочены символы в строках, аналогично буквам в алфавите.

***Определение 1*** (Лексикографический порядок) Лексикографический порядок в  определяется следующим. Если  две различные строки в , пусть  наименьшее целое число такое, что. Если , то поместить , в противном случае поместить .

Удобно занимать  первым символом, а  последним символом, независимо от размера множества . Например, если , мы возьмем , с порядком . В этом случае количество строк длиной 4 символа 34 = 81. Первые семь строк



и последние семь строк .

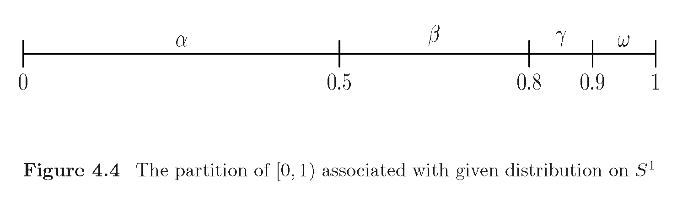
***Определение 2*** (Функция распределения) Учитывая распределение  на  мы определяем соответствующую функцию распределения  на  следующим образом. Если  – это первая строка , присваиваем . В противном случае, полагаем .

Например, предположим, что  и возьмем . Если значения  на 

,

тогда значения : .

Отметим, что для каждого  существует соответствующий интервал , и эти интервалы разбивают на части интервал  (Рис. 1).

Рис. 1 Разбиение интервала , связанное с данным распределением на 

В основном, учитывая строку , мы отмечаем с помощью  соответствующий интервал: .

При каждом фиксированном  эти мнтервалы разбиваю на части интервал .

***Определение 3*** (Арифметическое кодирование) Предположим, что стационарный источник испускает символы из множества , а распределение вероятности  известно заранее. Для  пусть , обозначим  как наименьшее целое такое, что , и пусть . Арифметическим кодом , обозначенным с помощью  является слово , которое представляет единственное целое число , для которого .

Согласно Теореме 1, условие  гарантирует, что число  находится в интервале .

***Пример 2*** Пусть . В нижеследующей таблице представлено распределение вероятности на : например, число в строке  и столбце  – это вероятность .

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  | 0,31 | 0,13 | 0,04 | 0,02 |
|  | 0,13 | 0,15 | 0,01 | 0,01 |
|  | 0,04 | 0,01 | 0,03 | 0,02 |
| ω | 0,02 | 0,01 | 0,02 | 0,05 |

Определить арифметический код для .

*Решение* Для каждой строки  возьмем  и вычислим , , и . Тогда определим  такое, что  и возьмем  как бинарное слово длиной  символов, соответствующее . Расчеты для первых нескольких кодовых слов приведены в таблице ниже. Отметим, что каждая дробь  лежит в соответствующем интервале 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  | 0,31 | 0,00 | 3,20 | 2 | 3 | [0\*23]=1 | 001 |
|  | 0,13 | 0,31 | 7,70 | 3 | 4 | [0.31\*24]=5 | 0101 |
|  | 0,04 | 0,44 | 25,00 | 5 | 6 | [0.44\*26]=29 | 011101 |
|  | 0,02 | 0,48 | 50,00 | 6 | 7 | [0.48\*27]=62 | 0111110 |
|  | 0,13 | 0,50 | 7,70 | 3 | 4 | [0.5\*24]=9 | 1001 |

Следует подчеркнуть, что мы не должны вычислять кодовые слова в последовательности. Например, если нам требуется кодовое слово для, мы можем найти его непосредственно, зная только, что .

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  | 0,01 | 0,84 | 100 | 7 | 8 | [0.84\*28]=216 | 11011000 |

Отметим, что это не источник без памяти, так как (например)

а .

***Задание.***

Пусть задан алфавит  и распределение распределение вероятностей на  (задать самостоятельно)

Для каждой строки  возьмем  и вычислим , , и . Определим  такое, что  и возьмем  как бинарное слово длиной  символов, соответствующее .

Независимость источника (наличие памяти) 