

# Contents

<b>Contents</b>	1
<b>1 Base</b>	3
1.1 Base.h	3
1.2 random.h	3
1.3 stl_util.h	3
1.4 probability.h	3
1.5 io_util.h	3
1.6 stat.h	4
<b>2 ML</b>	5
2.1 document.h	5
2.2 rbm	5
2.2.1 ais.h	5
<b>References</b>	7



# Chapter 1

## Base

### 1.1 Base.h

1. void Sum(const VVReal &, VReal\*);

### 1.2 random.h

1. void UniformSample(int t, VInt\* v);  
sample between 0 and size(v) uniformly t times.

### 1.3 stl\_util.h

1. void Multiply(const VReal &src, double m, VReal\* des);
2. void Multiply(const VVReal &src, double m, VVReal\* des);
3. int DiffNum(const VInt &lhs, const VInt &rhs);  
the number of different value between lhs and rhs
4. double Max(const T &data);
5. void Push(int num, const E &e, C\* des);

### 1.4 probability.h

1. int SumTopN(const VInt &src, int len);
2. bool NextMultiSeq(int num, VInt\* des);  
产生下一个多项式分布的序列
3. bool NextBinarySeq(VInt\* des);  
产生下一个二进制的序列

### 1.5 io\_util.h

1. void ReadFileToStr(const Str &file, Str\* str);
2. void ReadFileToStr(const Str &file, const Str &del, VStr\* data);
3. Str ReadFileToStr(const Str &file);

4. void WriteStrToFile(const Str &str, const Str &file);
5. void ReadFile(const Str &file, VInt\* des);
6. void WriteFile(const Str &file, const VInt &data);

## 1.6 stat.h

1. double LogSum(double log\_a, double log\_b);
2. double LogPartition(const VReal &data);  
输入data表示负能量值，返回对数配分函数。很多时候势函数由于很大造成溢出，采用取对数的方法可以防止溢出。

## Chapter 2

### ML

#### 2.1 document.h

#### 2.2 rbm

##### 2.2.1 ais.h

1. `double LogPartition(int doc_len, int word_num, const RepSoftMax &rep);`



## References