# 技术路线

在本比赛中，我们解决**T2 + PG2 + PP1**场景下的新冠病毒进化树构建任务。具体技术场景如下：

* **T2**：甲乙双方采用安全多方计算隐私计算技术；
* **PP1**：甲乙双方基因序列对输入前已经完成对齐(alignment)，大赛主办方预先采用成熟的软件mafft进行比对，确保序列对齐的准确性，然后进行数据截取；
* **PG2**：任一参与方的基因序列集数据明文不能被其他参与方以截获的方式获知。结果NJ树仅需保留节点和边的树形结构。

# 方案概述

在本方案中，我们使用了安全多方计算完成隐私计算过程。使用已经完成对齐的基因序列数据集，使用安全多方计算技术，在互不泄漏双方明文数据集的前提下，计算双方持有基因数据的距离（pdistance），并使用基因序列距离结果和ID名称，构建基因数据所对应的相似性矩阵，最后以相似性矩阵作为输入，构建出新冠病毒的进化树。

在下文中，为了能让大赛评委清晰了解本文所述技术方案，我们称两个计算方为甲方与乙方，其中：

甲、乙两方作为安全多方计算中的两方，双方无法通过传输内容，推知或反解出明文的基因序列内容，具体描述将在后续章节中进行分析。

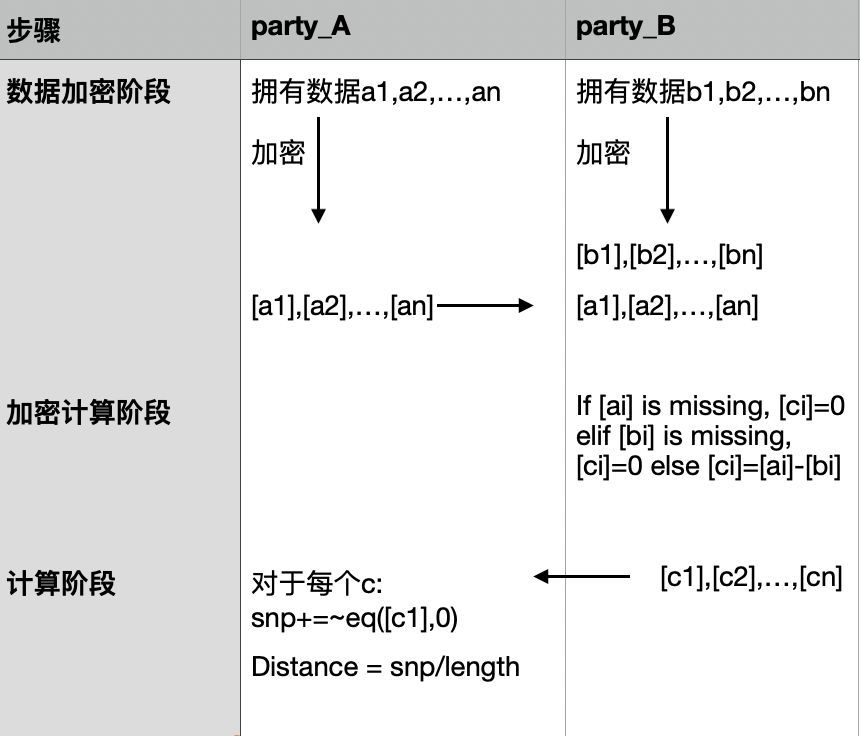
# 基因距离度量

## 安全多方计算的基本思路

首先，选择MPyC（<https://www.win.tue.nl/~berry/mpyc/>）开源的安全多方计算的Python package作为算法的核心。其主要特征有：安全类型、安全的input、output, eq方法，安全的加、减法（其实还是加法）、乘法计算。其安全类型对于每一方来说只能得到自己方贡献的份额的信息，其余的信息不能够被该方获取。

## 从基因序列比较入手

我们的总体方法如下图所示，在展示总体方法的步骤之后，我们介绍设计此方法的思路：



我们首先需要了解基因比较（计算距离）函数的逻辑，这是比较函数的源码，如下图所示：

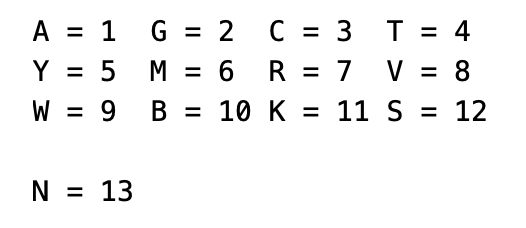
一張含有 文字 的圖片

自動產生的描述

梳理之后发现，基因序列的比较需要满足以下**三个原则**：

1. 两个序列都是有效碱基，且碱基相同，则 snp 不变，或者 snp += 0；
2. 两个序列都是有效碱基，且碱基不同，则 snp += 1；
3. 如果两个序列有任何一个碱基位置是缺失，则 snp 不累加；

而作为基因序列，碱基都是以字符形式出现的，但是作为同态加密的输入数据，肯定需要编码为数字，我们初步对基因进行如下编码（以下编码针对L1/目录中的fasta文件，L3中还存在新的碱基，并未在下图中显示），编码方式如下图所示：



对于原则3，按照该编码原则，首先可以进行逻辑判断该碱基是否缺失，缺失则snp += 0;

经过判断为有效碱基后，对于原则1和原则2，进行碱基比较，相同结果为1，不同结果为0，可知snp只需累加比较结果的相反值即可。

# 伪代码

|  |
| --- |
| 计算enc\_seq\_A和enc\_seq\_B距离 |
| 输入：enc\_seq\_A,enc\_seq\_B  输出：距离  For i in range(length)  if enc\_seq\_A[i]. is missing  res = 0  else if enc\_seq\_B[i] is missing  res = 0  else  res = enc\_seq\_A[i]-enc\_seq\_B[i]  diff\_cnt = secint(0)  for i in res:  diff\_cnt += ~mpc.eq(res[i],0)  distance = diff\_cnt / length |