构造后续算法所需的数据结构directed skyline graph

### 基本思路

主要分为以下几个步骤：

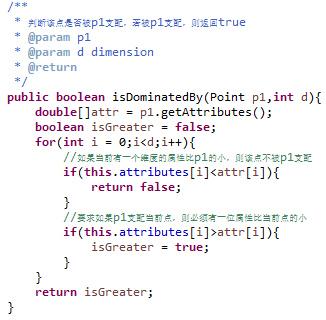
1. 获得给定的数据集后，按照第一维度属性值升序排序
2. 判断属性维度，如果等于2则直接使用二维下的方法进行skyline生成，否则进入高维属性下的skyline生成方法
3. 获得相应的k条skyline后，生成DSG

### 代码实现

1. Point类

Point类用于存放从文件中读入的每条数据，成员变量包括用于标记当前数据的标号以及存放各个维度数据的double类型数组。Point类实现了Comparable的compareTo方法，使得Point可以按照其第一维度属性值进行升序排序。

此外，Point类包含一个isDominatedBy方法利用定义判断当前点是否被给定的点所支配：如果当前点某个维度上的属性值比给定的点小，则不被其支配；如果所有维度上的属性值都不比给定的点相应的值小，且至少存在一个维度上的属性值比给定点的大，则该点被给定的点所支配：



1. DSGNode类

DSGNode类用于表示DSG中的每一个节点，成员变量包括用于标记当前数据标号的pointIndex、记录数据所在的skyline层号的layerIndex、存放各个维度数据的double类型数组attributes、记录其所有父节点的列表parents、记录其所有子节点的列表children以及记录属性维度个数的整形变量d。

1. DSG类

DSG类记录构造好的directed skyline graph，成员变量包括一个存放所有DSGNode的列表。

1. DSGGenerator类

DSGGenerator类为生成directed skyline graph的主要处理类，成员变量包括所生成的group skyline中的成员个数k、传入的待分组数据集、记录生成的k条skyline的列表以及每条数据的维度个数d。

在调用generateDSG方法生成DSG时，该方法首先调用成员方法generateSkylines生成k条skyline。在generateSkylines方法中，首先对数据集按照第一维度属性升序排序，然后判断属性的总维度，若为2，则调用generate2D方法利用论文中的algorithm1生成k条skyline，否则调用generateHighDimension方法使用简单的遍历生成k条skyline。

在调用generateSkylines方法生成k条skyline后，对每条skyline中的每一个点遍历在其所在skyline之前的所有skyline上的点，判断这些点是否支配该点，若支配，则将该点加入对应点的children列表中，相应的将这些点加入该点的parents列表中，最终生成DSG。

generate2D方法即对论文中algorithm1的实现，在其中调用了binarySearchLayer方法加快对给定点所属skyline层号的判断。此外，在生成新的skyline时判断当前skyline的层数，若大于k，则不再继续生成。

generateHighDimension方法即将论文中的algorithm1算法应用到高维属性中，此时无法利用二维情况下的单调性，即需要遍历每个已加入skyline的节点进行判断新加入的节点属于哪个skyline。