

# 训练数据与测试数据

数据分为两种，一种是随机生成数据，另一种是真实地图数据。

## 数据中的变量表

变量	变量含义
$\mathcal{V} = \{v_0, v_1, \dots, v_N\}$	结点集合
$v_0$	仓库结点
$v_i, i \neq 0$	客户结点
$N$	客户数量
$(x_i, y_i)$	结点 $v_i$ 的坐标
$q_i$	结点 $v_i$ 的需求
$\mathcal{E} = \{e_{ij}   v_i, v_j \in \mathcal{V}\}$	边集
$d_{ij}$	边 $e_{ij}$ 的旅行距离
$\mathcal{Q}_o$	车辆初始容量

## 随机生成数据

随机生成数据包含两个子集，分别对应 $N = 20$ 和 $N = 50$ ，并分别记为R-20与R-50。每个子集又分为训练部分和测试部分。总结如下：

数据集	问题规模	数据集大小
R-20-training	20	50000
R-20-testing	20	10000
R-50-training	50	50000
R-50-testing	50	10000

每个子集的VRP算例变量服从同一分布，只是在问题规模或数据集大小上有所区别。

变量服从的分布：

结点 $v_i$ 的坐标 $(x_i, y_i)$ 服从实数集合 $[0, 1]$ 上的均匀分布：

$$x_i \sim U(0, 1), \quad y_i \sim U(0, 1), \quad \text{其中} U \text{是均匀分布.}$$

客户结点 $v_i (i \neq 0)$ 的需求 $q_i$ 服从整数集合 $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ 上的均匀分布，注意仓库结点的需求 $v_0$  设置为0：

$$q_i (i \neq 0) \sim U\{1, 2, 3, 4, 5, 6, 7, 8, 9\}, \quad q_0 = 0.$$

注意在输入神经网络时，结点需求要进行归一化操作，即 $q'_i = q_i / Q_0$ .

边 $e_{ij}$ 的旅行距离 $d_{ij}$ 计算公式如下：

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} + \delta |\epsilon|, \quad \epsilon \sim \mathcal{N}(0, 1).$$

其中 $\delta$ 是超参数，用于控制算例的非对称程度，这次作业中取值 $\delta = 0.1$ ，感兴趣的同学还可以尝试不同的 $\delta$ 值， $\epsilon$ 服从均值为0，方差为1的正态分布。

车辆初始容量 $Q_0$

对于 $N = 20$ 的算例， $Q_0 = 30$ 。对于 $N = 50$ 的算例， $Q_0 = 40$ 。

数据格式

随机生成数据四个numpy文件(R-xx)对应前面表格中总结的四个数据集。

文件中字段定义如下：

graph: 结点坐标。shape：[ 问题数量, 结点数量, 2 ] (注意结点数量为 $N$ 客户加上1个仓库，所以为 $N + 1$ )

demand: 结点需求，已经归一化。shape：[ 问题数量 x 结点数量 ]

dis: 结点间旅行距离矩阵。shape：[ 问题数量 x 结点数量 x 结点数量 ]

## 真实地图数据

真实地图数据同样分为数据规模为 $N = 20$ 和 $N = 50$ 两个子集，分别记为G-20，G-50。同样每个子集包含训练部分和测试部分，总结如下：

数据集	问题规模	数据集大小
G-20-training	20	50000
G-20-testing	20	10000
G-50-training	50	50000
G-50-testing	50	10000

原始数据

原始地图数据获取方法是在广州市海珠区选择150个社区作为潜在客户结点，将中山大学南校区作为仓库结点, 再使用百度地图API获取各个结点的坐标和真实交通距离。

经纬度坐标.xlsx：仓库和150个潜在客户结点的经纬度坐标

旅行距离.xlsx: 各个结点之间的交通距离

实验数据的结点坐标和旅行距离生成方法：

在生成实验使用的训练和测试算例时，随机从150个潜在客户结点中选取20(50)个结点构成问题规模为20(50)的算例。

结点需求 $q_i$ 的生成方法与随机生成数据相同

初始车辆容量 $Q_0$ 与随机生成数据相同

数据格式

真是地图数据四个numpy文件(G-xx)对应前面表格中总结的四个数据集。

graph: 结点坐标。已经归一化。shape: [ 问题数量 x 结点数量 x 2 ] (注意结点数量为 $N$ 客户加上一个仓库，所以为 $N + 1$ )

demand: 结点需求，已经归一化。shape: [ 问题数量 x 结点数量 ]

dis: 结点间旅行距离矩阵。已经归一化。shape: [ 问题数量 x 结点数量 x 结点数量 ]

真实地图数据在输入神经网络时都进行归一化操作. 即  $(x'_i, y'_i) = (x_i/100, y_i/100)$ ,  $d'_{ij} = d_{ij}/\text{MAX}_{d_{ij} \in \mathcal{E}}$ .

## 数据链接

---

链接: <https://pan.baidu.com/s/1hoQrsfAKbX9MBc4a5ruokg>

提取码: r47g