山东大学 计算机科学与技术 学院

大数据分析实践 课程实验报告

学号：202300130028	姓名： 苗雨健	班级： 数据 23
实验题目：数据采样方法实践		
实验学时：2	实验日期： 2025/9/19	
实验目的：比较不同抽样方法在网络节点数据中的适用性和效果，培养根据数据结构特征选择合适抽样策略的能力		
硬件环境： 计算机一台		
软件环境： python3.9，jupyter notebook		

实验步骤与内容：

加载数据部分：

在加载数据时，由于未知原因，下载的数据集并非 utf-8 格式，而 pandas 默认读取 utf-8，导致加载失败，将其改为 gbk 模式读取解决问题

[4]:

	from_dev	from_port	from_city	from_level	to_dev	to_port	to_city	to_level	traffic	bandwidth
0	47	71	通辽	一般节点	1756	585	北京	网络核心	49636052613	1.000000e+11
1	47	74	通辽	一般节点	1756	776	北京	网络核心	50056871412	1.000000e+11
2	47	240	通辽	一般节点	1756	802	北京	网络核心	49453581081	1.000000e+11
3	47	241	通辽	一般节点	1997	464	天津	网络核心	49733361585	1.000000e+11
4	47	242	通辽	一般节点	474	672	哈尔滨	一般节点	50492573662	1.000000e+11
...	...	...	...	...	...	...	...	...	...	...
1113	1129	546	上海	网络核心	2050	502	石家庄	网络核心	48731433404	1.000000e+11
1114	1129	514	上海	网络核心	2473	946	吉林	一般节点	50060666120	1.000000e+11
1115	36036	499	长春	一般节点	1257	178	上海	网络核心	50545082113	1.000000e+11
1116	36422	346	天津	网络核心	1997	41	天津	网络核心	50628787089	1.000000e+11
1117	2701	619	大连	网络核心	2549	1070	沈阳	网络核心	48753971761	1.000000e+11

1118 rows × 10 columns

删除空行部分：

依然未知原因，数据集并没有出现空行。

[3]:

	from_dev	from_port	from_city	from_level	to_dev	to_port	to_city	to_level	traffic	bandwidth
0	47	71	通辽	一般节点	1756	585	北京	网络核心	49636052613	1.000000e+11
1	47	74	通辽	一般节点	1756	776	北京	网络核心	50056871412	1.000000e+11
2	47	240	通辽	一般节点	1756	802	北京	网络核心	49453581081	1.000000e+11
3	47	241	通辽	一般节点	1997	464	天津	网络核心	49733361585	1.000000e+11
4	47	242	通辽	一般节点	474	672	哈尔滨	一般节点	50492573662	1.000000e+11
...	...	...	...	...	...	...	...	...	...	...
1113	1129	546	上海	网络核心	2050	502	石家庄	网络核心	48731433404	1.000000e+11
1114	1129	514	上海	网络核心	2473	946	吉林	一般节点	50060666120	1.000000e+11
1115	36036	499	长春	一般节点	1257	178	上海	网络核心	50545082113	1.000000e+11
1116	36422	346	天津	网络核心	1997	41	天津	网络核心	50628787089	1.000000e+11
1117	2701	619	大连	网络核心	2549	1070	沈阳	网络核心	48753971761	1.000000e+11

1118 rows × 10 columns

过滤部分：

依照示例代码成功进行过滤

加权采样：

首先对原始数据进行清洗和过滤，然后根据 to_level 字段为每个样本分配权重（'一般节点'权重为 1，其他节点权重为 5），最后使用 sample()方法按权重抽取 50 个样本。这种抽样方法确保了非一般节点有更高的被抽中概率，适用于需要重点考察特定类别数据的场景

	from_dev	from_port	from_city	from_level	to_dev	to_port	to_city	to_level	traffic	bandwidth	weight
578	63	70	通辽	一般节点	235	1749	北京	网络核心	50871621460	1.000000e+11	5
95	474	359	哈尔滨	一般节点	2050	502	石家庄	网络核心	51299508559	1.000000e+11	5
118	474	1238	哈尔滨	一般节点	1756	1008	北京	网络核心	51270474683	1.000000e+11	5
2	47	240	通辽	一般节点	1756	802	北京	网络核心	49453581081	1.000000e+11	5
330	96	336	呼和浩特	一般节点	1756	1106	北京	网络核心	51277669375	1.000000e+11	5
304	63	230	通辽	一般节点	3227	77	济南	网络核心	50504074996	1.000000e+11	5
544	63	54	通辽	一般节点	2050	336	石家庄	网络核心	51911829933	1.000000e+11	5
416	591	60	绥化	一般节点	180	52	呼和浩特	一般节点	50126205393	1.000000e+11	1
393	474	1238	哈尔滨	一般节点	1997	122	天津	网络核心	49693039378	1.000000e+11	5
353	180	188	呼和浩特	一般节点	1536	2274	广州	网络核心	50649912010	1.000000e+11	5
119	474	1246	哈尔滨	一般节点	3227	705	济南	网络核心	50954049859	1.000000e+11	5
133	591	13	绥化	一般节点	36422	324	天津	网络核心	50085514419	1.000000e+11	5
449	787	316	玉溪	一般节点	36422	394	天津	网络核心	50880826227	1.000000e+11	5
942	36036	52	长春	一般节点	2050	272	石家庄	网络核心	49916177327	1.000000e+11	5
339	180	18	呼和浩特	一般节点	47	241	通辽	一般节点	51793025548	1.000000e+11	1
540	47	427	通辽	一般节点	2549	1430	沈阳	网络核心	50264080533	1.000000e+11	5
1059	47	252	通辽	一般节点	1997	250	天津	网络核心	50358481161	1.000000e+11	5

随机抽样：

在数据清洗和过滤的基础上，直接使用 sample()方法无放回地随机抽取 50 个样本，每个样本被抽中的概率均等。这种方法简单易行，适用于对总体没有先验知识、希望获得代表性样本的场景。

	from_dev	from_port	from_city	from_level	to_dev	to_port	to_city	to_level	traffic	bandwidth
705	47	242	通辽	一般节点	63	286	通辽	一般节点	49144860439	1.000000e+11
441	591	1300	绥化	一般节点	47	252	通辽	一般节点	50817586398	1.000000e+11
129	474	1410	哈尔滨	一般节点	4069	1205	宁波	一般节点	46523775334	1.000000e+11
134	591	15	绥化	一般节点	1385	1490	广州	网络核心	49228307349	1.000000e+11
455	787	360	玉溪	一般节点	2701	71	大连	网络核心	50222195150	1.000000e+11
352	180	98	呼和浩特	一般节点	1997	190	天津	网络核心	48859748771	1.000000e+11
381	474	475	哈尔滨	一般节点	2473	941	吉林	一般节点	49402590822	1.000000e+11
495	47	258	通辽	一般节点	235	1958	北京	网络核心	48574009525	1.000000e+11
793	180	20	呼和浩特	一般节点	474	359	哈尔滨	一般节点	50601340670	1.000000e+11
318	96	124	呼和浩特	一般节点	1536	1891	广州	网络核心	49479386359	1.000000e+11
339	180	18	呼和浩特	一般节点	47	241	通辽	一般节点	51793025548	1.000000e+11
146	591	502	绥化	一般节点	1129	546	上海	网络核心	49465128399	1.000000e+11
138	591	27	绥化	一般节点	3443	117	青岛	网络核心	49213859972	1.000000e+11
353	180	188	呼和浩特	一般节点	1536	2274	广州	网络核心	50649912010	1.000000e+11
173	787	307	玉溪	一般节点	4953	686	贵阳	一般节点	49399787960	1.000000e+11

分层抽样：

首先将数据按 to_level 分为"一般节点"和"网络核心"两个层级，然后从"一般节点"层抽取

17 个样本，从"网络核心"层抽取 33 个样本，最后合并形成 50 个样本的总样本。

14]:

	from_dev	from_port	from_city	from_level	to_dev	to_port	to_city	to_level	traffic	bandwidth
13	47	314	通辽	一般节点	96	152	呼和浩特	一般节点	50161220081	1.000000e+11
151	591	586	绥化	一般节点	180	192	呼和浩特	一般节点	49061517661	1.000000e+11
110	474	672	哈尔滨	一般节点	47	242	通辽	一般节点	51555817613	1.000000e+11
334	96	391	呼和浩特	一般节点	96	120	呼和浩特	一般节点	51609945530	1.000000e+11
180	787	360	玉溪	一般节点	3615	191	长沙	一般节点	49629725686	1.000000e+11
542	63	10	通辽	一般节点	4360	472	南京	一般节点	49716409605	1.000000e+11
61	96	407	呼和浩特	一般节点	4069	1196	宁波	一般节点	49745162804	1.000000e+11
445	787	60	玉溪	一般节点	47	314	通辽	一般节点	49484495071	1.000000e+11
19	63	12	通辽	一般节点	180	252	呼和浩特	一般节点	49290094443	1.000000e+11
780	96	391	呼和浩特	一般节点	180	205	呼和浩特	一般节点	50103206178	1.000000e+11
441	591	1300	绥化	一般节点	47	252	通辽	一般节点	50817586398	1.000000e+11
541	63	6	通辽	一般节点	2473	1043	吉林	一般节点	48954016072	1.000000e+11
770	474	672	哈尔滨	一般节点	180	42	呼和浩特	一般节点	51263599555	1.000000e+11
743	4069	1195	宁波	一般节点	96	134	呼和浩特	一般节点	50099141709	1.000000e+11
59	96	391	呼和浩特	一般节点	47	417	通辽	一般节点	51570663870	1.000000e+11

系统抽样：

在数据清洗和过滤后，通过计算抽样间隔（k = 总样本量/目标样本量），随机选择起始点，然后每隔 k 个样本抽取一个，最终获得 50 个系统分布的样本。这种方法保证了样本在总体中的均匀分布，避免了简单随机抽样可能出现的聚类现象。

```
data_before_sample = data_after_filter_2

def systematic_sampling(data, sample_size, random_start=True):
    n = len(data)
    if n == 0:
        return pd.DataFrame()
    k = n // sample_size
    if k == 0:
        k = 1
    if random_start:
        start = np.random.randint(0, k)
    else:
        start = 0
    indices = []
    for i in range(sample_size):
        index = start + i * k
        if index < n:
            indices.append(index)
        else:
            break
    return data.iloc[indices].copy().reset_index(drop=True)

sample_size = 50
systematic_sample = systematic_sampling(data_before_sample, sample_size)
systematic_sample
```

	from_dev	from_port	from_city	from_level	to_dev	to_port	to_city	to_level	traffic	bandwidth
0	47	242	通辽	一般节点	474	672	哈尔滨	一般节点	50492573662	1.000000e+11
1	47	425	通辽	一般节点	1756	1018	北京	网络核心	50796899329	1.000000e+11
2	63	74	通辽	一般节点	2701	181	大连	网络核心	50364636480	1.000000e+11
3	96	108	呼和浩特	一般节点	2360	236	太原	网络核心	48210462086	1.000000e+11
4	96	141	呼和浩特	一般节点	474	422	哈尔滨	一般节点	49429192047	1.000000e+11
5	96	391	呼和浩特	一般节点	47	417	通辽	一般节点	51570663870	1.000000e+11
6	180	36	呼和浩特	一般节点	2194	406	唐山	网络核心	50973267302	1.000000e+11
7	180	202	呼和浩特	一般节点	36272	247	太原	网络核心	49867223584	1.000000e+11
8	180	272	呼和浩特	一般节点	3443	316	青岛	网络核心	52854391127	1.000000e+11
9	474	614	哈尔滨	一般节点	3227	724	济南	网络核心	51504522549	1.000000e+11
10	474	1238	哈尔滨	一般节点	1756	1008	北京	网络核心	51270474683	1.000000e+11
11	474	1410	哈尔滨	一般节点	4069	1205	宁波	一般节点	46523775334	1.000000e+11
12	591	56	绥化	一般节点	36036	52	长春	一般节点	48627355195	1.000000e+11

整群抽样：

首先识别数据中的自然群组，然后随机选择 5 个群组，将被选中群组的所有个体纳入样本。

```
def cluster_sampling(data, cluster_column, n_clusters=None, cluster_ratio=None, random_state=None):
    if random_state is not None:
        np.random.seed(random_state)

    unique_clusters = data[cluster_column].unique()
    total_clusters = len(unique_clusters)
    if n_clusters is not None:
        sample_n = min(n_clusters, total_clusters)
    elif cluster_ratio is not None:
        sample_n = max(1, int(total_clusters * cluster_ratio))
    else:
        sample_n = max(1, int(total_clusters * 0.3))
    selected_clusters = np.random.choice(unique_clusters, size=sample_n, replace=False)
    sampled_data = data[data[cluster_column].isin(selected_clusters)].copy()
    return sampled_data, selected_clusters

possible_cluster_columns = ['to_level', 'from_id', 'to_id']

cluster_column = None
for col in possible_cluster_columns:
    if col in data_before_sample.columns:
        cluster_column = col
        break
```

4	47	242	通辽	一般节点	474	672	哈尔滨	一般节点	50492573662	1.000000e+11
5	47	243	通辽	一般节点	96	124	呼和浩特	一般节点	49942713747	1.000000e+11
7	47	250	通辽	一般节点	2473	762	吉林	一般节点	49108721007	1.000000e+11
9	47	252	通辽	一般节点	96	134	呼和浩特	一般节点	50256475808	1.000000e+11
13	47	314	通辽	一般节点	96	152	呼和浩特	一般节点	50161220081	1.000000e+11
...	...	...	...	...	...	...	...	...	...	...
1057	47	243	通辽	一般节点	2473	769	吉林	一般节点	49117847542	1.000000e+11
1063	47	314	通辽	一般节点	47	252	通辽	一般节点	49900452417	1.000000e+11
1079	63	224	通辽	一般节点	4069	1196	宁波	一般节点	50209459772	1.000000e+11
1097	2473	1460	吉林	一般节点	591	586	绥化	一般节点	48409925693	1.000000e+11
1104	63	6	通辽	一般节点	36036	20	长春	一般节点	50355678076	1.000000e+11

结论分析与体会：

在大数据实践过程中，通过对同一数据集应用不同的抽样方法。简单随机抽样操作简便且每个样本被抽中的概率均等，但可能忽略重要子群体的代表性；分层抽样通过按节点等级划分层级，确保了各类节点的均衡覆盖，显著降低了抽样误差；系统抽样以固定间隔抽取样本，实现了样本在总体中的均匀分布，避免了随机抽样可能出现的聚类现象；整群抽样基于网络拓扑特征分组抽取，大大提高了现场实施的效率。