4.2 下载原始数据（可选）

(English Version)

如果用户的数据集已经在本地磁盘中，请确保它被存放在目录 `raw_dir` 中。如果用户想在任何地方运行代码而又不想自己下载数据并将其移动到正确的目录中，则可以通过实现函数 `download()` 来自动完成。

如果数据集是一个zip文件，可以直接继承 `dgl.data.DGLBuiltinDataset` 类。后者支持解压缩zip文件。否则用户需要自己实现 `download()`，具体可以参考 `QM7bDataset` 类：

```
import os
from dgl.data.utils import download

def download(self):
    # 存储文件的路径
    file_path = os.path.join(self.raw_dir, self.name + '.mat')
    # 下载文件
    download(self.url, path=file_path)
```

上面的代码将一个.mat文件下载到目录 `self.raw_dir`。如果文件是.gz、.tar、.tar.gz或.tgz文件，请使用 `extract_archive()` 函数进行解压缩。以下代码展示了如何在 `BitcoinOTCDataset` 类中下载一个.gz文件：

```
from dgl.data.utils import download, check_sha1

def download(self):
    # 存储文件的路径，请确保使用与原始文件名相同的后缀
    gz_file_path = os.path.join(self.raw_dir, self.name + '.csv.gz')
    # 下载文件
    download(self.url, path=gz_file_path)
    # 检查 SHA-1
    if not check_sha1(gz_file_path, self._sha1_str):
        raise UserWarning('File {} is downloaded but the content hash does not match.'
                          'The repo may be outdated or download may be incomplete. '
                          'Otherwise you can create an issue for it.'.format(self.name +
'.csv.gz'))
    # 将文件解压缩到目录self.raw_dir下的self.name目录中
    self._extract_gz(gz_file_path, self.raw_path)
```

上面的代码会将文件解压缩到 `self.raw_dir` 下的目录 `self.name` 中。如果该类继承自 `dgl.data.DGLBuiltinDataset` 来处理zip文件，则它也会将文件解压缩到目录 `self.name` 中。

一个可选项是用户可以按照上面的示例检查下载后文件的SHA-1字符串，以防作者在远程服务器上更改了文件。