```python
from pydrive.auth import GoogleAuth
```

```python
from pydrive.drive import GoogleDrive
```

```python
from google.colab import auth
```

```python
from oauth2client.client import GoogleCredentials
```

```python
auth.authenticate_user()
gauth = GoogleAuth()
gauth.credentials = GoogleCredentials.get_application_default()
drive = GoogleDrive(gauth)


def get_file_from_drive(folder_id, file_name):
  file_list = drive.ListFile({'q': "'" + folder_id + "' in parents and trashed=false"}).GetLi
  for file in file_list:
    if file['title'] == file_name:
      return file['id']


def upload_file_to_drive(file_name, file_data):
  uploaded = drive.CreateFile({'title': file_name})
  uploaded.SetContentString(file_data)
  uploaded.Upload()
  print('Uploaded file with ID {}'.format(uploaded.get('id')))

drive_folder_id = '<Folder ID>'
```

```python
from google.colab import drive
drive.mount('/content/drive/')
```

```
    Mounted at /content/drive/
```

```python
from numpy import genfromtxt
import numpy as np
```

```python
X = genfromtxt("/content/drive/My Drive/Mirai_dataset.csv", delimiter=",")
```

```python
Y = genfromtxt("/content/drive/My Drive/Mirai_labels.csv", delimiter=",")
```

```
nn_save('Mirai_data_npy'   X)
```

```python
np.save('Mirai_data.npy', X)
np.save('Mirai_labels.npy', Y)


X = np.load('Mirai_data.npy')
Y = np.load('Mirai_labels.npy')


X = np.delete(X, obj=np.s_[:24], axis=1)
# X= X[:,24:116]

print('The covariates have size {}.'.format(X.shape))
print('The labels have size {}.'.format(Y.shape))

# Print the number of examples of each class
print('Class 0 has {} examples'.format(sum([1 for i in Y if i==0])))
print('Class 1 has {} examples'.format(sum([1 for i in Y if i==1])))
```

```
The covariates have size (764137, 92).
The labels have size (764137,).
Class 0 has 121621 examples
Class 1 has 642516 examples
```

```python
print(np.isnan(X))
print(np.isnan(Y))
print('Labels have {} NaNs'.format(np.sum(np.isnan(Y))))

# Print the number of NaNs in the covariates
print('Covariates have {} NaNs'.format(np.sum(np.isnan(X))))
```

```
[[False False False ... False False False]
 [False False False ... False False False]
 [False False False ... False False False]
 ...
 [False False False ... False False False]
 [False False False ... False False False]
 [False False False ... False False False]]
[False False False ... False False False]
Labels have 0 NaNs
Covariates have 0 NaNs
```

+ Code   + Text

✓ 0s   completed at 10:28 PM