```
import copy, json

df_len = len(df)
train_ds = {}
test_ds = {}
ds_item = { 'start': None, 'target': [] }

# 学習用とテスト用のデータを 8:2 で分割する
train_df = df[:int(df_len*0.8)]
test_df = df[int(df_len*0.8):]


# データを以下の形式に変換する。
# {
#   "start": "2016-01-01 00:00:00",
#   "target": [4.9, 5.3, 3.4, 5.9, ...]
# }
for column in columns:
    train_ds[column] = copy.deepcopy(ds_item)
    train_ds[column]['start'] = str(train_df.index.values[0])
    test_ds[column] = copy.deepcopy(ds_item)
    test_ds[column]['start'] = str(test_df.index.values[0])

for index, row in train_df.iterrows():
    for column in columns:
        train_ds[column]['target'].append(str(row[column]))

for index, row in test_df.iterrows():
    for column in columns:
        test_ds[column]['target'].append(str(row[column]))


# jsonlineの形式で 一時ファイルに書き出し
train_path = 'train.jsonl'
test_path = 'test.jsonl'

def save_to_file(path, dataset):
    with open(path, 'wb') as fp:
        for column in columns:
            fp.write(json.dumps(dataset[column]).encode("utf-8"))
            fp.write("¥n".encode('utf-8'))

# ファイルに保存
save_to_file(train_path, train_ds)
```

```
save_to_file(test_path, test_ds)

# s3にアップロード
deepar_prefix = 'deepar'
s3 = boto3.resource('s3')
bucket = s3.Bucket(s3_bucket)

with open(train_path, 'rb') as data:
    bucket.put_object(Key=f'{deepar_prefix}/{train_path}', Body=data)

with open(test_path, 'rb') as data:
    bucket.put_object(Key=f'{deepar_prefix}/{test_path}', Body=data)
```