Форматы данных (2)

Материалы:

- Макрушин С.В. "Лекция 5: Форматы данных (часть 2)"
- https://docs.python.org/3/library/csv.html
- https://docs.h5py.org/en/stable/
- Уэс Маккини. Python и анализ данных

Задачи для совместного разбора

[ ] *4 cells hidden*

Лабораторная работа 5

```
import pandas as pd
from google.colab import drive
drive.mount('/content/gdrive')
import json
```

```
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/content/gdrive", force_remount=True).
```

csv

1.1 В файле `tags_sample.csv` находится информация о тэгах, приписываемых рецептам. Воспользовавшись `csv.reader`, считайте этот файл и создайте словарь вида `id_рецепта: [список тэгов]`. Сохраните этот словарь в файл `tags_sample.json`.

```
tags_sample0 = pd.read_csv('gdrive/My Drive/Colab Notebooks/files2/tags_sample.csv', header = 0)
tags_sample = pd.DataFrame(tags_sample0)
```

```
tags_sample
```

	id	tag
0	44123	weeknight
1	44123	time-to-make
2	44123	course
3	44123	main-ingredient
4	44123	cuisine
...	...	...
533485	298512	cookies-and-brownies
533486	298512	dietary
533487	298512	high-calcium
533488	298512	high-in-something
533489	298512	number-of-servings

533490 rows × 2 columns

```
tags_sample['tag'].isnull().sum()
```

```
16
```

```
print(tags_sample.isnull().sum())
```

```
id      0
tag    16
```

```
dtype: int64
```

```
null_id = list(tags_sample[tags_sample.isnull().any(axis=1)].id.to_dict().values())
null_id
# tags_sample[tags_sample.isnull().any(axis=1)].id
```

```
[505748,
 506224,
 290380,
 506142,
 506068,
 506032,
 505610,
 505750,
 506104,
 506007,
 505968,
 505729,
 505793,
 505647,
 361341,
 506238]
```

```
# tags_sample = tags_sample.dropna()
```

```
tags_ = tags_sample.to_dict(orient='records')
tags_
```

```
# for i in null_id:
#   print(tags_[i])

# for i in null_id:
#   tags_[i]['tag']

# print(tags_[3089])
# tags_sample[tags_sample['id']==505748]

for dic in tags_:
  if dic['id'] in null_id:
    print(dic['tag'])
```

```
tags_s_dict = {i:[] for i in set(list(tags_sample['id']))}
# tags_s_dict
for pair in tags_:
  rem = tags_s_dict[pair['id']]
  if pair['id'] not in null_id:
    rem.append(pair['tag'])
    tags_s_dict[pair['id']] = rem
tags_s_dict
```

```
for id in null_id:
  print(tags_s_dict[id])
```

```
with open('gdrive/My Drive/Colab Notebooks/files2/tags_sample.json', 'w') as file:
  json.dump(tags_s_dict, file, indent=2)
```

1.2 Считайте файл `recipes_sample_with_filled_nsteps.csv` (**ЛР4**) в виде `pd.DataFrame`. Добавьте к таблице 2 столбца: `n_tags`, содержащий количество тэгов у этого рецепта; и `tags`, содержащий набор тэгов в виде строки (тэги внутри строки разделяются символом `;`)

```
recipes_s0 = pd.read_csv('gdrive/My Drive/Colab Notebooks/files/recipes_sample_with_filled_nsteps.csv', header = 0)
recipes_s0 = recipes_s0.drop(columns=['Unnamed: 0'])
recipes_s = pd.DataFrame(recipes_s0)
recipes_s
```

	name	id	minutes	contributor_id	submitted	n_steps	description	n_ingred
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	11	an original recipe created by chef scott meska...	
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	3	my children and their friends ask for my homem...	
2	i can t believe it s spinach	38798	30	1533	2002-08-29	5	these were so go, it surprised	
3	italian gut busters	35173	45	22724	2002-07-	7	law made	

```
tags_c_dict = {i:len(tags_s_dict[i]) for i in tags_s_dict}
tags_c_dict
```

```
len(tags_c_dict)
```

```
30000
```

```
recipes_s_id = recipes_s.to_dict(orient = 'list')['id']

tags_c_dict_df = []

for id in recipes_s_id:
  if id in tags_c_dict.keys():
    tags_c_dict_df.append(tags_c_dict[id])
  else:
    tags_c_dict_df.append(0)
# tags_c_dict_df


recipes_s['n_tags'] = tags_c_dict_df
# recipes_s = recipes_s.drop(columns=['n_tags'])
recipes_s
```

	name	id	minutes	contributor_id	submitted	n_steps	description	n_ingred
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	11	an original recipe created by chef scott meska...	
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	3	my children and their friends ask for my homem...	
2	i can t believe it s spinach	38798	30	1533	2002-08-29	5	these were so go, it surprised even me.	
3	italian gut busters	35173	45	22724	2002-07-27	7	my sister-in-law made these for us at a family...	

```
recipes_s[recipes_s.id == 505748]
```

	name	id	minutes	contributor_id	submitted	n_steps	description	n_ingredients	n_t
	3						who says that recipes		

```
# tags_s_dict
# recipes_s_id = recipes_s.to_dict(orient = 'list')['id']

tags_s_dict_df = []

for id in recipes_s_id:
  tags_s_dict_df.append(';'.join(tags_s_dict[id]))
```

```
# tags_s_dict_df


recipes_s['tags'] = tags_s_dict_df
# recipes_s = recipes_s.drop(columns=['n_tags'])
recipes_s
```

	name	id	minutes	contributor_id	submitted	n_steps	descriptio
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	11	an origina recip created b chef sco meska.
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	3	my childre and the friends as for m homem.
2	i can t believe it s spinach	38798	30	1533	2002-08-29	5	these wer so go, surprise even me
3	italian gut busters	35173	45	22724	2002-07-27	7	my sister-in law mad these for u at a family.
4	love is in the air beef fondue sauces	84797	25	4470	2004-02-23	4	i think fondue is very romanti casual din.
...	...	...	...	...	...	...	.
29995	zurie s holey rustic olive and cheddar bread	267661	80	200862	2007-11-25	16	this is base on a frenc recipe but changed.

1.3 В файле `ingredients_sample.csv` находится информация о ингредиентах, необходимых для рецепта. Воспользовавшись `csv.DictReader`, считайте этот файл и создайте словарь вида `id_рецепта: [список ингредиентов]`.

```
from csv import DictReader


ingredients_records = {}

with open('gdrive/My Drive/Colab Notebooks/files2/ingredients_sample.csv') as csvfile:
  reader = DictReader( csvfile )

  for record in reader:
    # records[record['recipe_id']] = [[record['ingredient']]]
    if record['recipe_id'] in ingredients_records.keys():
      rem = ingredients_records[record['recipe_id']]
      rem.append(record['ingredient'])
      ingredients_records[record['recipe_id']] = rem
    else:
      ingredients_records[record['recipe_id']] = [record['ingredient']]

print(ingredients_records['44123'])
# print(len(records))
```

```
['unsalted butter', 'carrot', 'onion', 'celery', 'broccoli stem', 'dried thyme', 'dried oregano', 'dried sweet basil leaves', 'dry white
```

1.4 Добавьте к таблице из задания 1.2 столбец `ingredients`, содержащий набор ингредиентов в виде строки (ингредиенты внутри строки разделяются символом `*`)

Для строк, которые содержат пропуски в столбце `n_ingredients`, заполните их на основе файла `ingredients_sample.csv`

```
ingredients_records_df = []
```

```
for id in recipes_s_id:
  ingredients_records_df.append('*'.join(ingredients_records[str(id)]))

ingredients_count = {int(id):len(ingredients_records[id]) for id in ingredients_records.keys()}
ingredients_count


recipes_s['ingredients'] = ingredients_records_df
# recipes_s = recipes_s.drop(columns=['n_tags'])
recipes_s
```

	name	id	minutes	contributor_id	submitted	n_steps	descriptio
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	11	an origina recip created b chef sco meska.
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	3	my childre and the friends as for m homem.
2	i can t believe it s spinach	38798	30	1533	2002-08-29	5	these wer so go, surprise even me
3	italian gut busters	35173	45	22724	2002-07-27	7	my sister-in law mad these for u at a family.
4	love is in the air beef fondue sauces	84797	25	4470	2004-02-23	4	i think fondue is very romanti casual din.
...	...	...	...	...	...	...	.
29995	zurie s holey rustic olive and cheddar bread	267661	80	200862	2007-11-25	16	this is base on a frenc recipe but changed.
29996	zwetschgenkuchen bavarian plum cake	386977	240	177443	2009-08-24	22	this is traditiona fresh plur cake thought.
29997	zwiebelkuchen southwest german	103312	75	161745	2004-11-	10	this is traditiona

```
recipes_s.isnull().sum()


recipes_s_copy = recipes_s.copy()


recipes_s_copy.n_ingredients.isnull().sum()
```

```
8880
```

```
for i in range(len(recipes_s)):
    if str(recipes_s.loc[i]['n_ingredients']) == "nan":
      recipes_s.loc[[i], 'n_ingredients'] = ingredients_count[recipes_s.loc[i,'id']]


recipes_s
```

	name	id	minutes	contributor_id	submitted	n_steps	descriptio
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	11	an origina recip created b chef sco meska.
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	3	my childre and the friends as for m homem.
2	i can t believe it s spinach	38798	30	1533	2002-08-29	5	these wer so go, surprise even me
3	italian gut busters	35173	45	22724	2002-07-27	7	my sister-in law mad these for u at a family.
4	love is in the air beef fondue sauces	84797	25	4470	2004-02-23	4	i think fondue is very romanti casual din.
...	...	...	...	...	...	...	.
29995	zurie s holey rustic olive and cheddar bread	267661	80	200862	2007-11-25	16	this is base on a frenc recipe but changed. this is

```
recipes_s.n_ingredients.isnull().sum()
```

```
0
```

1.5 Проверьте, содержит ли столбец `n_ingredients` пропуски. Если нет, преобразуйте его к целочисленному типу и сохраните результаты в файл `recipes_sample_with_tags_ingredients.csv`

```
recipes_s['n_ingredients'] = recipes_s['n_ingredients'].astype(int)
```

```
recipes_s
```

```
recipes_s.to_csv('gdrive/My Drive/Colab Notebooks/files2/recipes_sample_with_tags_ingredients.csv')
```

npy

2.1 Разделите таблицу, полученную в результате 1.5, на две таблицы: одна содержит рецепты, загруженные до 2000 года; вторая - все остальные. В полученных таблицах оставьте только числовые столбцы и преобразуйте их к `numpy.array`

```
# recipes.loc[(recipes['minutes']<=20) & (recipes['n_ingredients']<=5)]
recipes_s.loc[recipes_s['submitted']<'2000-01-01']
```

```
recipes_after_2000_df, recipes_until_2000_df = [x for _, x in recipes_s.groupby(recipes_s['submitted']<'2000-01-01')]
```

```
recipes_after_2000_df
```

	name	id	minutes	contributor_id	submitted	n_steps	description	n_ingred
0	george s at the cove black bean soup	44123	90	35193	2002-10-25	11	an original recipe created by chef scott meska...	
1	healthy for them yogurt popsicles	67664	10	91970	2003-07-26	3	my children and their friends ask for my homem...	
2	i can t believe it s spinach	38798	30	1533	2002-08-29	5	these were so go, it surprised even me.	
3	italian gut busters	35173	45	22724	2002-07-27	7	my sister-in-law made these for us at a family...	
4	love is in the air beef fondue sauces	84797	25	4470	2004-02-23	4	i think a fondue is a very romantic casual din...	
...	...	...	...	...	...	...	...	
29995	zurie s holey rustic olive and cheddar bread	267661	80	200862	2007-11-25	16	this is based on a french recipe but i changed...	

```
recipes_after_2000 = recipes_after_2000_df.drop(columns=['name', 'description', 'tags', 'ingredients']).to_numpy()
recipes_after_2000
```

```
array([[44123, 90, 35193, ..., 11, 18, 25],
       [67664, 10, 91970, ..., 3, 3, 31],
       [38798, 30, 1533, ..., 5, 8, 17],
       ...,
       [103312, 75, 161745, ..., 10, 13, 20],
       [486161, 60, 227978, ..., 7, 22, 20],
       [298512, 29, 506822, ..., 9, 10, 12]], dtype=object)
```

```
recipes_until_2000 = recipes_until_2000_df.drop(columns=['name', 'description', 'tags', 'ingredients']).to_numpy()
recipes_until_2000
```

```
array([[3441, 30, 1562, ..., 8, 8, 10],
       [4205, 25, 1617, ..., 3, 5, 14],
       [3258, 0, 1534, ..., 8, 6, 20],
       ...,
       [3752, 0, 1535, ..., 13, 4, 9],
       [4801, 20, 1598, ..., 4, 7, 18],
       [2982, 0, 124030, ..., 6, 7, 13]], dtype=object)
```

2.2. Сохраните 2 полученных массива в архив `npz`. Дайте массивам читаемые имена.

```
import numpy as np
np.savez('gdrive/My Drive/Colab Notebooks/files2/recipes_archive.npz', recipes_after_2000=recipes_after_2000, recipes_until_2000=recipes_unti
```

2.3 Считайте созданный архив и продемонстрируйте, что данные считались корректно.

```
npzfile = np.load('gdrive/My Drive/Colab Notebooks/files2/recipes_archive.npz', allow_pickle=True)
list(npzfile)
```

```
['recipes_after_2000', 'recipes_until_2000']
```

```
npzfile['recipes_after_2000']
```

```
array([[44123, 90, 35193, ..., 11, 18, 25],
       [67664, 10, 91970, ..., 3, 3, 31],
       [38798, 30, 1533, ..., 5, 8, 17],
       ...,
       [103312, 75, 161745, ..., 10, 13, 20],
```

```
       [486161, 60, 227978, ..., 7, 22, 20],
       [298512, 29, 506822, ..., 9, 10, 12]], dtype=object)
```

hdf

3.1 Выведите названия всех датасетов, находящихся в файле `nutrition_sample.h5`, а также размерность матриц, содержащихся в данных датасетах и их метаданные.

Формат вывода:

```
Dataset name=dataset_0, dataset size=(30000,), metadata={'info': 'calories (#)'}
Dataset name=dataset_1, dataset size=(30000,), metadata={'info': 'total fat (PDV)'}
...
```

```
import h5py

import time
import os
```

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_sample.h5', 'r') as f:
    for key in f.keys():
      info = str(f[key]).split()
      # print(info)
      print('Dataset name=',info[2][:-1], ', dataset size=', info[4], info[5], ' metadata={', info[6], '=',info[7], '}')
```

```
Dataset name= "dataset_0" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
Dataset name= "dataset_1" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
Dataset name= "dataset_2" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
Dataset name= "dataset_3" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
Dataset name= "dataset_4" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
Dataset name= "dataset_5" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
Dataset name= "dataset_6" , dataset size= (30000, 2),  metadata={ type = "<f8"> }
```

3.2 Разбейте каждый из имеющихся датасетов на две части: 1 часть содержит только те строки, где PDV (Percent Daily Value) превышает 100%; 2 часть содержит те строки, где PDV не составляет не более 100%. Создайте 2 группы в файле и разместите в них соответствующие части датасета с сохранением метаданных исходных датасетов. Итого должно получиться 2 группы, содержащие несколько датасетов. Если датасет не содержитСохраните результаты в файл `nutrition_grouped.h5`

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_sample.h5', 'r') as f:
  dss = {}
  for ds in f.keys():
    dss[ds] = pd.DataFrame(f[ds])
    # dss[ds] = dss[ds][()]
    # ds_0 = f['dataset_0']
    # ds_0 = ds_0[()]
```

```
print(dss['dataset_2'])
```

```
                0      1
     0       44123.0   26.0
     1       67664.0    5.0
     2       38798.0    2.0
     3       35173.0   11.0
     4       84797.0  323.0
     ...         ...    ...
     29995  267661.0   16.0
     29996  386977.0  122.0
     29997  103312.0   30.0
     29998  486161.0   34.0
     29999  298512.0   57.0

     [30000 rows x 2 columns]
```

```
# recipes_after_2000_df, recipes_until_2000_df = [x for _, x in recipes_s.groupby(recipes_s['submitted']<'2000-01-01')]
under_100 = []
above_100 = []
for ds in dss.keys():
  ds1, ds2 = [x for _, x in dss[ds].groupby(dss[ds][1]<100)]
```

```
  under_100.append(ds2)
  above_100.append(ds1)
```

```
len(under_100)
```

```
7
```

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'w') as f:
  g1 = f.create_group('Under-100')
  g2 = f.create_group('Above-100')

  i = 0
  for ds in under_100:
    d = g1.create_dataset(name='dataset_u_'+str(i), data=ds)
    i += 1

  j = 0
  for ds in above_100:
    d = g2.create_dataset(name='dataset_a_'+str(j), data=ds)
    j += 1



# for ds in under_100:
#   ds.to_hdf('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'Under-100')

# for ds in above_100:
#   ds.to_hdf('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'Above-100')
```

3.3 Выведите названия всех групп и датасетов, находящихся в этих группах, из файла `nutrition_grouped.h5` а также размерность матриц, содержащихся в датасетах и их метаданные.

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'r') as f:
  for k in f.keys():
    print(k)
```

```
Above-100
Under-100
```

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'r') as f:
  def get_all(name):
    print(name)
  f.visit(get_all)
```

```
Above-100
Above-100/dataset_a_0
Above-100/dataset_a_1
Above-100/dataset_a_2
Above-100/dataset_a_3
Above-100/dataset_a_4
Above-100/dataset_a_5
Above-100/dataset_a_6
Under-100
Under-100/dataset_u_0
Under-100/dataset_u_1
Under-100/dataset_u_2
Under-100/dataset_u_3
Under-100/dataset_u_4
Under-100/dataset_u_5
Under-100/dataset_u_6
```

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'r') as f:
  base_items = list(f.items())
  # print(base_items)
  G1 = list(f.get('Above-100').items())
  for ds in G1:
    print(ds)
  print('')
  G2 = list(f.get('Under-100').items())
  for ds in G2:
    print(ds)
```

```
('dataset_a_0', <HDF5 dataset "dataset_a_0": shape (26740, 2), type "<f8">)
('dataset_a_1', <HDF5 dataset "dataset_a_1": shape (1764, 2), type "<f8">)
```

```
('dataset_a_2', <HDF5 dataset "dataset_a_2": shape (5391, 2), type "<f8">)
('dataset_a_3', <HDF5 dataset "dataset_a_3": shape (1274, 2), type "<f8">)
('dataset_a_4', <HDF5 dataset "dataset_a_4": shape (1830, 2), type "<f8">)
('dataset_a_5', <HDF5 dataset "dataset_a_5": shape (2914, 2), type "<f8">)
('dataset_a_6', <HDF5 dataset "dataset_a_6": shape (650, 2), type "<f8">)

('dataset_u_0', <HDF5 dataset "dataset_u_0": shape (3260, 2), type "<f8">)
('dataset_u_1', <HDF5 dataset "dataset_u_1": shape (28236, 2), type "<f8">)
('dataset_u_2', <HDF5 dataset "dataset_u_2": shape (24609, 2), type "<f8">)
('dataset_u_3', <HDF5 dataset "dataset_u_3": shape (28726, 2), type "<f8">)
('dataset_u_4', <HDF5 dataset "dataset_u_4": shape (28170, 2), type "<f8">)
('dataset_u_5', <HDF5 dataset "dataset_u_5": shape (27086, 2), type "<f8">)
('dataset_u_6', <HDF5 dataset "dataset_u_6": shape (29350, 2), type "<f8">)
```

```
# df_1 = pd.read_hdf('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'Above-100')
# df_1
```

3.4 Модифицируйте код из 3.3 таким образом, чтобы сохранить датасеты, используя сжатие. Сравните размер полученного файла с размерами файла из 3.3. Прокомментируйте результат.

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped_compr.h5', 'w') as f:
  i = 0
  for ds in under_100:
    d = f.create_dataset(name='dataset_u_'+str(i), data=ds, compression='gzip', compression_opts=9)
    i += 1
  # ds.to_hdf('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'Under-100')
  j = 0
  for ds in above_100:
    d = f.create_dataset(name='dataset_a_'+str(j), data=ds, compression='gzip', compression_opts=9)
    j += 1
  # ds.to_hdf('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5', 'Above-100')

print(os.path.getsize('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped.h5'))
print(os.path.getsize('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped_compr.h5'))
```

```
3369280
1137192
```

```
with h5py.File('gdrive/My Drive/Colab Notebooks/files2/nutrition_grouped_compr.h5', 'r') as f:
  for k in f.keys():
    print(k)
```

```
dataset_a_0
dataset_a_1
dataset_a_2
dataset_a_3
dataset_a_4
dataset_a_5
dataset_a_6
dataset_u_0
dataset_u_1
dataset_u_2
dataset_u_3
dataset_u_4
dataset_u_5
dataset_u_6
```

✓ 2s completed at 10:59 PM