
Проект 1

НИКИТА КОТЕЛЕВСКИЙ

О чем поговорим сегодня

1. О чем был проект?

2. Для чего нам нужен pandas?

3. Общие ошибки.

4. Разбор кейсов.

5. Лучшие решения.

6. Немного рекомендаций.

7. Вопросы

О чем был проект

Проект был направлен на отработку навыков работы с библиотекой pandas.

Основная оценка складывалась за счет:

1. Количество правильно решенных задач
 2. Качество решения (насколько быстро, понятно и компактно получилось решить)
 3. PEP-8
 4. Оформление репозитория.
-

Для чего нам нужен pandas?

Этот проект был первым “боевым” — вы использовали реальные данные для анализа.

1. Pandas активно применяется специалистами по анализу данных особенно часто в тех областях, где ведется работам с табличными данными (банковская сфера, консалтинг, тд)
 2. Те инструменты Pandas, которые мы изучили, позволяют нам провести первичный EDA — понять, что из себя представляют данные, какие в них особенности и закономерности.
 3. Хороший инструмент для подготовки к обучению моделей — позволяет выбрать нужные признаки, создать новые, нормализовать данные, отсеять выбросы и тд.
-

Общие ошибки

Репозиторий создан только под один проект 0/проект 1:

Как **не надо**:

OverviewRepositoriesProjects

Find a repository...

Type: All

Language: All

module_0

Jupyter NotebookUpdated 19 days ago

Star

Как **надо**:

OverviewRepositories1Projects

Find a repository...

Type: All

Language: All

SkillFactory-projects

PythonUpdated 18 days ago

Star

ElenaSrbq / SkillFactory-projects

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

master

1 branch

0 tags

Go to file

Add file

Code

Name changes

0e7abf818 days ago6 commits

module_0

changes

18 days ago

.DS_Store

Fold added

18 days ago

Общие ошибки

Код должен быть рабочим! Проверяйте его перед отправкой, чтобы не допускать глупых ошибок (буквально, запускайте все ячейки. Все они должны удачно отработать):

11. Какого жанра фильмов больше всего?

```
[34]: answers['11'] = '3. Drama'
```

```
[35]: all_genres = [] # создаю список из всех жанров
c = collections.Counter()
for genres in data.genres.apply(split_func):
    for genre in genres:
        all_genres.append(genre)
for genre in all_genres:
    c[genre] += 1
print(c)
```

Здесь и далее я использую функцию print, потому что она автоматически сортирует Counter по значениям

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-35-ca882870ea6c> in <module>
      1 all_genres = [] # создаю список из всех жанров
----> 2 c = collections.Counter()
      3 for genres in data.genres.apply(split_func):
      4     for genre in genres:
      5         all_genres.append(genre)

NameError: name 'collections' is not defined
```

ВАРИАНТ 2

Общие ошибки

Код должен быть рабочим! Проверяйте его перед отправкой, чтобы не допускать глупых ошибок (буквально, запускайте все ячейки. Все они должны удачно отработать):

14. Какой режиссер снял больше всего фильмов в стиле Action?

```
[36]: answers['14'] = 'Robert Rodriguez'
```

```
[37]: func(data[data.genres.str.contains('Action')], 'director')
```

```
-----  
NameError                                Traceback (most recent call last)  
<ipython-input-37-7c8442ad9bdb> in <module>  
----> 1 func(data[data.genres.str.contains('Action')], 'director')  
  
NameError: name 'func' is not defined
```


Общие ошибки

Код должен быть рабочим! Проверяйте его перед отправкой, чтобы не допускать глупых ошибок (буквально, запускайте все ячейки. Все они должны удачно отработать):

15. Фильмы с каким актером принесли самые высокие кассовые сборы в 2012 году?

```
[53]: sorted_data=data[['release_year','revenue','castlist']]
```

```
[54]: sorted_data = sorted_data[sorted_data.release_year == 2012]
sorted_data = sorted_data.explode('castlist').groupby('castlist')
sorted_data = sorted_data.agg('sum').sort_values('revenue',ascending=False)
```

```
[55]: answers['15']='Chris Hemsworth'
```

```
[56]: sorted_data.drop('cast',axis=1)
```

```
-----
KeyError                                Traceback (most recent call last)
<ipython-input-56-c13f7c600684> in <module>
----> 1 sorted_data.drop('cast',axis=1)

~/anaconda3/lib/python3.7/site-packages/pandas/core/frame.py in drop(self, labels, axis, index, columns, level, inplace, errors)
   3995         level=level,
   3996         inplace=inplace,
-> 3997         errors=errors,
   3998     )
   3999

~/anaconda3/lib/python3.7/site-packages/pandas/core/generic.py in drop(self, labels, axis, index, columns, level, inplace, errors)
   3934     for axis, labels in axes.items():
   3935         if labels is not None:
-> 3936             obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   3937
   3938     if inplace:

~/anaconda3/lib/python3.7/site-packages/pandas/core/generic.py in _drop_axis(self, labels, axis, level, errors)
   3968         new_axis = axis.drop(labels, level=level, errors=errors)
   3969     else:
-> 3970         new_axis = axis.drop(labels, errors=errors)
   3971         result = self.reindex(**{axis_name: new_axis})
   3972

~/anaconda3/lib/python3.7/site-packages/pandas/core/indexes/base.py in drop(self, labels, errors)
   5016         if mask.any():
   5017             if errors != "ignore":
-> 5018                 raise KeyError(f"{labels[mask]} not found in axis")
   5019             indexer = indexer[~mask]
   5020         return self.delete(indexer)
```

```
KeyError: "[cast]" not found in axis"
```


Разбор кейсов.

Какие ошибки/слабости вы видите в этом коде?

22. Сколько суммарно вышло фильмов летом? (за июнь, июль, август)

```
[13]: answers['22'] = '2. 450'

[14]: # А вообще для удобства создадим-ка еще колонку с месяцем релиза – тогда задания станут понятными
def split_date(x):
    return str(x).split('/')[0]
data['release_month'] = data.release_date.apply(split_date)

[15]: # чтобы посчитать, превратим таблицу, полученную функцией value_counts, в Series
monthly_movies = pd.Series(data.release_month.value_counts())

summer_movies = monthly_movies[6] + monthly_movies[7] + monthly_movies[8]
# print(summer_movies)

[16]: monthly_movies

[16]: 9      227
      12     190
      10     186
       8     161
       3     156
       4     149
       6     147
      11     146
       7     142
       5     140
       2     135
       1     110
      Name: release_month, dtype: int64
```

23. Для какого режиссера зима – самое продуктивное время года?

Разбор кейсов.

Какие ошибки/слабости вы видите в этом коде?

22. Сколько суммарно вышло фильмов летом? (за июнь, июль, август)

```
[13]: answers['22'] = '2. 450'
```

```
[14]: # А вообще для удобства создадим-ка еще колонку с месяцем релиза – тогда задания станут понятными
def split_date(x):
    return str(x).split('/')[0]
data['release_month'] = data.release_date.apply(split_date)
```

```
[15]: # чтобы посчитать, превратим таблицу, полученную функцией value_counts, в Series
monthly_movies = pd.Series(data.release_month.value_counts())

summer_movies = monthly_movies[6] + monthly_movies[7] + monthly_movies[8]
# print(summer_movies)
```

```
[17]: print(monthly_movies[6], monthly_movies[7], monthly_movies[8], monthly_movies[6] + monthly_movies[7] + monthly_movies[8])
print(monthly_movies['6'], monthly_movies['7'], monthly_movies['8'], monthly_movies['6'] + monthly_movies['7'] + monthly_movies['8'])

147 146 142 435
147 142 161 450
```

```
[18]: monthly_movies
```

```
[18]: 9      227
12     190
10     186
8      161
3      156
4      149
6      147
11     146
7      142
5      140
2      135
1      110
Name: release_month, dtype: int64
```

Разбор кейсов.

Какие ошибки/слабости вы видите в этом коде?

24. Какая студия дает самые длинные названия своим фильмам по количеству символов?

```
[56]: # определяем максимальную длину имени фильма
max_len = data.original_title.str.len().max()

# отфильтровываем датафрейм – оставляем только с максимальной длиной названия
# обращаемся к колонке "кинокомпания", разделяем по сплитеру и получаем значение кинокомпании
data[data.original_title.str.len() == max_len].production_companies.iloc[0].split('|')[1]
```

```
[56]: 'Four By Two Productions'
```

```
[57]: # +
answers['24'] = 'Four By Two Productions'
```

Разбор кейсов.

Какие ошибки/слабости вы видите в этом коде?

25. Описание фильмов какой студии в среднем самые длинные по количеству слов?

```
[49]: def prepare2xplode (str2split): #мини-функция для метода xplode
      if str2split.find('|') > -1:
          res = str2split.split("|")
          return res
      else: return str2split

      maxlen = 0
      titlelen = 0
      production = 'lol'

      df2 = data[['production_companies', 'overview']]
      df2.production_companies = df2.production_companies.apply(prepare2xplode)
      df2.overview = df2.overview.apply(lambda x: x.count(" ")+1)
      df2 = df2.explode('production_companies')

      companiesarray = df2.production_companies.value_counts().keys()

      for company in companiesarray:
          titlelen = df2[df2.production_companies==company].overview.mean()
          if titlelen > maxlen:
              maxlen = titlelen
              production = company

      answers['25'] = production
      answers['25']
```

```
[49]: 'Midnight Picture Show'
```


Разбор кейсов.

Какие ошибки/слабости вы видите в этом коде?

25. Описание фильмов какой студии в среднем самые длинные по количеству слов?

```
[47]: %%timeit

def prepare2xplode (str2split): #мини-функция для метода xplode
    if str2split.find('|') > -1:
        res = str2split.split("|")
        return res
    else: return str2split

maxlen = 0
titlelen = 0
production = 'lol'

df2 = data[['production_companies', 'overview']]
df2.production_companies = df2.production_companies.apply(prepare2xplode)
df2.overview = df2.overview.apply(lambda x: x.count(" ")+1)
df2 = df2.explode('production_companies')

companiesarray = df2.production_companies.value_counts().keys()

for company in companiesarray:
    titlelen = df2[df2.production_companies==company].overview.mean()
    if titlelen > maxlen:
        maxlen = titlelen
        production = company

answers['25'] = production

1.31 s ± 56.9 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
```

```
[48]: answers['25']
```

```
[48]: 'Midnight Picture Show'
```

Разбор кейсов.

Какие ошибки/слабости вы видите в этом коде?

25. Описание фильмов какой студии в среднем самые длинные по количеству слов?

```
[46]: %%timeit
def prepare2xplode (str2split): #мини-функция для метода xplode
    if str2split.find('|') > -1:
        res = str2split.split("|")
        return res
    else: return str2split

maxlen = 0
titlelen = 0
production = 'lol'

df2 = data[['production_companies','overview']]
df2.production_companies = df2.production_companies.apply(prepare2xplode)
df2.overview = df2.overview.apply(lambda x: x.count(" ")+1)
df2 = df2.explode('production_companies')

# companiesarray = df2.production_companies.value_counts().keys()

# for company in companiesarray:
#     titlelen = df2[df2.production_companies==company].overview.mean()
#     if titlelen > maxlen:
#         maxlen = titlelen
#         production = company

# answers['25'] = production
# answers['25']

df2.groupby(by='production_companies').overview.mean().sort_values(ascending=False).index[0]
```

10.7 ms ± 366 µs per loop (mean ± std. dev. of 7 runs, 100 loops each)

Давайте разберем решения!

Немного рекомендаций

1. Git:

Нужно создать общий репозиторий под все проекты! Необходимо пользоваться командной строкой для работы с гитом (или приложением), но не загрузкой файлов через веб-интерфейс.

2. Python:

Проверка кода перед отправкой

PEP-8

Информативные комментарии

Вопросы!
