Лабораторна робота №3.
Візуальний аналіз даних про публікації на сайті Хабрахабр

Заповніть код в клітинках

In [ ]:
```
import pandas as pd
%matplotlib inline
import seaborn as sns
import matplotlib.pyplot as plt
```

Завантаження і знайомство з даними

Для роботи вам знадобляться поепердньо оброблені дані з платформи kaggle (https://www.kaggle.com/c/howpop-habrahabr-favs) (https://www.kaggle.com/c/howpop-habrahabr-favs)).

Доступ до даних на google drive, якщо ви відкриваєте блокнот в google colab, а не на PC, можна отримати шляхом монтування google drive

In [ ]:
```
from google.colab import drive
drive.mount('/content/gdrive')
```

In [ ]:
```
!ls gdrive/'My Drive'/TEACHING/IntroDataScience/intro_to_data_science/Lab_3_4/da
```

In [ ]:
```
# шлях до папки з даними на моєму google drive, відредагуйте згідно вашого випад
data_folder = "gdrive/My Drive/TEACHING/IntroDataScience/intro_to_data_science/L
```

In [ ]:
```
#df = pd.read_csv('data/howpop_train.csv')
df = pd.read_csv(data_folder+'/howpop_train.csv')
```

In [ ]:
```
df.shape
```

Out[3]:
```
(134137, 17)
```

In [ ]:
```
df.head(3).T
```

Out[4]:

	0	1	
url	https://habrahabr.ru/post/18284/	https://habrahabr.ru/post/18285/	https://habrahabr.ru/po
domain	habrahabr.ru	habrahabr.ru	hab
post_id	18284	18285	
published	2008-01-01 18:19:00	2008-01-01 18:30:00	2008-01-01
author	@Tapac	@DezmASter	@De
flow	develop	design	
polling	False	False	
content_len	4305	7344	
title	Новогодний подарок блоггерам — WordPress 2.3.2	Сумасшедшие яйца, или сервис для отслеживания ...	Сумасшедшие я сервис для отслежи
comments	0	1	
favs	0	1	
views	236	353	
votes_plus	0	1	
votes_minus	0	0	
views_lognorm	-0.792687	-0.485214	
favs_lognorm	-1.34407	-0.831946	
comments_lognorm	-2.43687	-1.87319	

Позбавимося відразу від змінних, назви яких закінчуються на `_lognorm` (потрібні для змагань на Kaggle). Виберемо їх за допомогою `filter()` і видаимо `drop` -ом:

In [ ]:
```
df.drop(filter(lambda c: c.endswith('_lognorm'), df.columns),
        axis = 1,       # axis = 1: столвбці
        inplace = True) # позбавляє від необхідності зберігати датасет
```

In [ ]:
```
df.describe().T
```

Out[6]:

	count	mean	std	min	25%	50%	75%	m
post_id	134137.0	181307.054265	81766.350702	18284.0	115565.0	182666.0	257401.0	3140
content_len	134137.0	7516.929699	8724.771640	1.0	2179.0	4949.0	9858.0	3524
comments	134137.0	39.625994	62.398958	0.0	7.0	19.0	48.0	22
favs	134137.0	71.999866	145.854135	0.0	8.0	27.0	78.0	87
views	134137.0	16631.013084	31479.819691	3.0	2100.0	7600.0	18700.0	17300
votes_plus	133566.0	35.536888	42.461073	0.0	11.0	22.0	45.0	14
votes_minus	133566.0	8.050035	11.398282	0.0	2.0	5.0	10.0	4

In [ ]:
```
df.describe(include = ['object', 'bool'] # бінарні і категоріальні змінні
            ).T
```

Out[7]:

	count	unique	top	freq
url	134137	134137	https://habrahabr.ru/post/87933/	1
domain	134137	2	habrahabr.ru	97048
published	134137	130291	2011-06-14 15:52:00	39
author	97657	22077	@alizar	5292
flow	97048	6	develop	53318
polling	134137	2	False	129412
title	134137	133628	Интересные события, произошедшие в выходные	56

In [ ]:
```
# налаштування зовнішнього вигляду графіків в seaborn
sns.set_style("dark")
sns.set_palette("RdBu")
sns.set_context("notebook", font_scale = 1.5,
                rc = { "figure.figsize" : (15, 5), "axes.titlesize" : 18 })
```

Стовбець **published** (час публікації) містить рядки. Щоб ми мали змогу працювати з цими даними як з датою/часом публікації, приведемо їх до типу `datetime` :

In [ ]:
```
print(df.published.dtype)
df['published'] = pd.to_datetime(df.published, yearfirst = True)
print(df.published.dtype)
```

```
object
datetime64[ns]
```

Створимо декілька стовбців на основі даних про час публікації:

In [ ]:
```
df['year'] = [d.year for d in df.published]
df['month'] = [d.month for d in df.published]

df['dayofweek'] = [d.isoweekday() for d in df.published]
df['hour'] = [d.hour for d in df.published]
```

Відтепер Ваша черга. В кожному пункті пропонується побудувати картинку і з її допомогою відповісти на питання. Звичано, можна спробувати відповісти на всі питання лише використовуючи Pandas, без картинок, але ми радимо Вам потренуватися будувати (красиві) візуалізації.

1. В якому місяці (і якого року) було більше всього публікацій?

- березень 2016
- березень 2015
- квітень 2015
- квітень 2016

In [ ]:
```

```

2. Проаналізуйте публікації в місяці з попереднього питання

Виберіть один чи декілька варіантів:

- Один чи декілька днів сильно виділяються із загальної картини
- На хабрі *завжди* більше статей, ніж на гіктаймсі
- По суботам на гіктаймс і на хабрахабр публікують приблизно одинакову кількість статей

Підказки: побудуйте графік залежності числа публікацій від дня; використовуйте параметр `hue` ; не переймайтесь сильно з відповідями і не шукайте прихованого змісту :)

In [ ]:
```

```

3. Коли найкраще всього опублікувати статтю?

- Більш всього переглядів набирають статті, опубліковані в 12 годин дня
- В опублікованих о 10-й ранку постів більше всього коментарів
- Більше всього переглядів набирають статтті, опубліковані в 6 годин ранку
- Максимальне число коментарів на гіктаймсі набрала стаття, опублікована в 9 годин вечора
- На хабрі денні статті комментують частіше, ніж вечірні

In [ ]:

4. Кого з топ-20 авторів частіше всього мінусують?

- @Mordatyj
- @Mithgol
- @alizar
- @ilya42

In [ ]:

5. Порівняйте суботи і понеділки

Чи правда, що по суботам автори пишуть в основному вдень, а по понедількам — в основному вечером?

In [ ]: