In [25]:

```
import scipy as sps
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
from tqdm.notebook import tqdm
import numba
import gc

sns.set(font_scale=1.5)
```

Данные хранятся в файле `stop_points_data.txt` . Попробуем прочитать первую строку в даных.

Выведем начало строки.

In [2]:

```
with open("stop_points_data.txt") as file_handler:
    line = file_handler.readline()
    print(line[:1000])
```

```
session=[{"status": 0, "y": 0.0, "ts": 0, "x": 0.0}, {"status": 0,
"y": 0.2911288692583093, "ts": 9, "x": -0.6448019382406134}, {"statu
s": 0, "y": 1.229173178592953, "ts": 17, "x": -0.24366284626788443},
{"status": 0, "y": 13.056778378200342, "ts": 25, "x": 2.44714445650604
96}, {"status": 0, "y": 21.47505743411422, "ts": 33, "x": -3.184498571
7071896}, {"status": 0, "y": 21.482221727177166, "ts": 41, "x": -3.891
5564540242107}, {"status": 0, "y": 24.859388237736596, "ts": 49, "x":
-7.891632600692803}, {"status": 0, "y": 40.66878618117626, "ts": 57,
"x": -8.823989467671135}, {"status": 0, "y": 63.53642212023506, "ts":
65, "x": 21.015910259296565}, {"status": 0, "y": 81.4936540237547, "t
s": 73, "x": 47.33571398853448}, {"status": 0, "y": 107.8015420044901
2, "ts": 81, "x": 76.63476932223945}, {"status": 0, "y": 115.940234559
99715, "ts": 86, "x": 89.87931785951574}, {"status": 0, "y": 115.94023
455999715, "ts": 90, "x": 89.87931785951574}, {"status": 0, "y": 115.9
4023455999715, "ts": 98, "x":
```

В каждой строке хранится одна сессия. Из вида строки ясно, что её можно исполнить, используя функцию `exec` и получить список из словарей для каждой сессии. Далее, все словари всех полученных спсиков можно объеденить в один датафрейм, добавив колонку `session` , отвечающую за номер сессии.

Добавим, также, колонку `begin` , значение в которой будет 1, если данная точка - первая в сессии, иначе - 0.

Посчитаем количество строк в файле:

In [3]:

```
with open("stop_points_data.txt") as file_handler:
        i = 0
        for line in tqdm(file_handler):
            i+=1

print(i)
```

A Jupyter widget could not be displayed because the widget state could not be found. This could happen if the kernel storing the widget is no longer available, or if the widget state was not saved in the notebook. You may be able to create the widget by running the appropriate cells.

```
115204
```

Функция обработки данных.

In [45]:

```
def process_data(input_filename, max_iter_num=np.inf):
    df = pd.DataFrame()
    with open(input_filename) as file_handler:
        i = 0
        for line in tqdm(file_handler):
            if (i == max_iter_num):
                break

            ldic=locals()
            exec(line,globals(),ldic)
            session = ldic['session']

            if (i == 0):
                df = pd.DataFrame(session)
                df['session'] = i
                df['begin'] = 0
                df.loc[0, 'begin'] = 1
            else:
                curr_df = pd.DataFrame(session)
                curr_df['session'] = i
                df = df.append(curr_df)
                df['begin'] = 0
                df.loc[0, 'begin'] = 1

            i += 1

    return(df)
```

Для сокращения времени, преобразуем на данном этапе только часть данных.

In [47]:

```
input_name = "stop_points_data.txt"
data = process_data(input_name, 3000)
```

A Jupyter widget could not be displayed because the widget state could not be found. This could happen if the kernel storing the widget is no longer available, or if the widget state was not saved in the notebook. You may be able to create the widget by running the appropriate cells.

In [6]:

```
data.head()
```

Out[6]:

	status	y	ts	x	session	begin
0	0	0.000000	0	0.000000	0	1
1	0	0.291129	9	-0.644802	0	0
2	0	1.229173	17	-0.243663	0	0
3	0	13.056778	25	2.447144	0	0
4	0	21.475057	33	-3.184499	0	0

In [7]:

```
data.shape
```

Out[7]:

```
(312288, 6)
```

Сохраним получившийся датафрем в файл для дальнейшего использоваиния.

In [8]:

```
data.to_csv('./data/3k_processed_data.csv', index=False)
```

Обработаем все данные для финального решения.

Для более быстрой обработки будем просто парсить строки файла.

In [117]:

```
def process_data_fast(input_filename, max_iter_num=np.inf):
    with open(input_filename) as file_handler:
        i = 0
        all_data = []
        for line in tqdm(file_handler):
            if (i == max_iter_num):
                break

            points = line[9:-2].split('}, ')

            for j, point in enumerate(points):
                feature_list = []

                if j == len(points)-1:
                    point = point[:-1]

                features = point[1:].split(', ')

                for feature in features:
                    feature_list.append(float(feature.split(': ')[1]))
                feature_list.append(int(j == 0))
                feature_list.append(i)
                all_data.append(feature_list)

            i += 1

    return all_data
```

In [119]:

```
res = process_data_fast("stop_points_data.txt")
```

A Jupyter widget could not be displayed because the widget state could not be found. This could happen if the kernel storing the widget is no longer available, or if the widget state was not saved in the notebook. You may be able to create the widget by running the appropriate cells.

In [121]:

```
all_data = pd.DataFrame(res, columns = ['status', 'y', 'ts',
                                        'x', 'begin', 'session'])
```

In [124]:

```
1 all_data.head()
```

Out[124]:

	status	y	ts	x	begin	session
0	0.0	0.000000	0.0	0.000000	1	0
1	0.0	0.291129	9.0	-0.644802	0	0
2	0.0	1.229173	17.0	-0.243663	0	0
3	0.0	13.056778	25.0	2.447144	0	0
4	0.0	21.475057	33.0	-3.184499	0	0

Сохраним все данные в отдельный файл.

In [125]:

```
1 all_data.to_csv('./data/processed_data.csv', index=False)
```