Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Курс «Методы машинного обучения»

Отчет по лабораторной работе №1:
«Создание "истории о данных"»

Выполнила:
студентка группы ИУ5-24М
Мащенко Е. И.

Проверил:
Балашов А.М.

2023

Цель работы

Изучение различных методов визуализация данных и создание истории на основе данных.

Задание

Создать "историю о данных" в виде юпитер-ноутбука, с учетом следующих требований:

1. История должна содержать не менее 5 шагов (где 5 - рекомендуемое количество шагов). Каждый шаг содержит график и его текстовую интерпретацию.
2. На каждом шаге наряду с удачным итоговым графиком рекомендуется в юпитер-ноутбуке оставлять результаты предварительных "неудачных" графиков.
3. Не рекомендуется повторять виды графиков, желательно создать 5 графиков различных видов.
4. Выбор графиков должен быть обоснован использованием методологии data-to-viz. Рекомендуется учитывать типичные ошибки построения выбранного вида графика по методологии data-to-viz. Если методология Вами отвергается, то просьба обосновать Ваше решение по выбору графика.
5. История должна содержать итоговые выводы. В реальных "историях о данных" именно эти выводы представляют собой основную ценность для предприятия.

Выполнение работы

Лабораторная работа №1

In [1]:
```
from datetime import datetime
import pandas as pd
import seaborn as sns
```

In [2]:
```
# Enable inline plots
%matplotlib inline

# Set plot style
sns.set(style="ticks")

# Set plots formats to save high resolution PNG
from IPython.display import set_matplotlib_formats
set_matplotlib_formats("retina")
```

```
<ipython-input-2-52d19245dee9>:9: DeprecationWarning: `set_matplotlib_formats` is deprecated since IPython 7.23, directly us
e `matplotlib_inline.backend_inline.set_matplotlib_formats()`
  set_matplotlib_formats("retina")
```

In [3]:
```
pd.set_option("display.width", 70)
```

In [4]:
```
data = pd.read_csv("insurance.csv")
```

In [5]:
```
data.dtypes
```

Out[5]:
```
age           int64
sex          object
bmi         float64
children      int64
smoker       object
region       object
charges     float64
dtype: object
```

In [6]:
```
data.head()
```

Out[6]:

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

In [7]:
```
data.shape
```

Out[7]:
```
(1338, 7)
```

In [8]:
```
data.describe()
```

Out[8]:

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

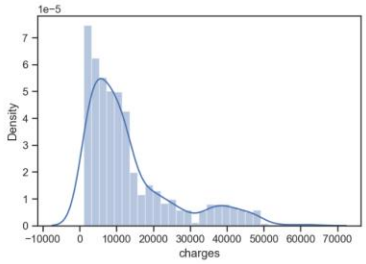
Визуальное исследование датасета

Оценим распределение целевого признака — расходы на страхование:

In [9]:
```
sns.distplot(data["charges"])
```

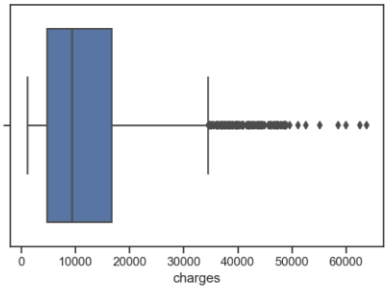
```
c:\Users\User\AppData\Local\Programs\Python\Python38\lib\site-packages\seaborn\distributions.py:2557: FutureWarning: `distpl
ot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figu
re-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[9]:
```
<AxesSubplot:xlabel='charges', ylabel='Density'>
```



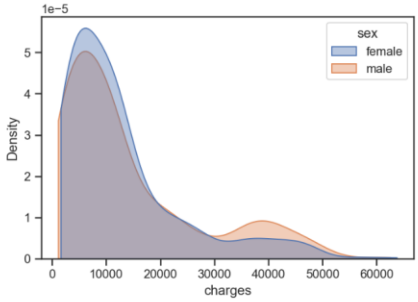
B [10]:
```
sns.boxplot(x=data['charges'])
```

Out[10]: <AxesSubplot:xlabel='charges'>



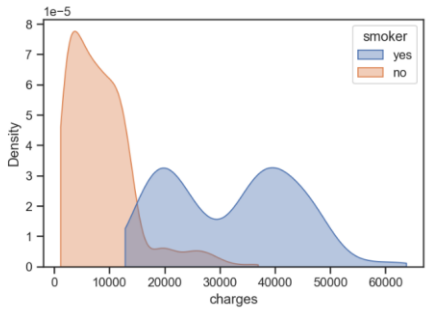
B [13]:
```
sns.kdeplot(data=data, x="charges", hue="sex", cut=0, fill=True, common_norm=False, alpha=0.4)
```

Out[13]: <AxesSubplot:xlabel='charges', ylabel='Density'>



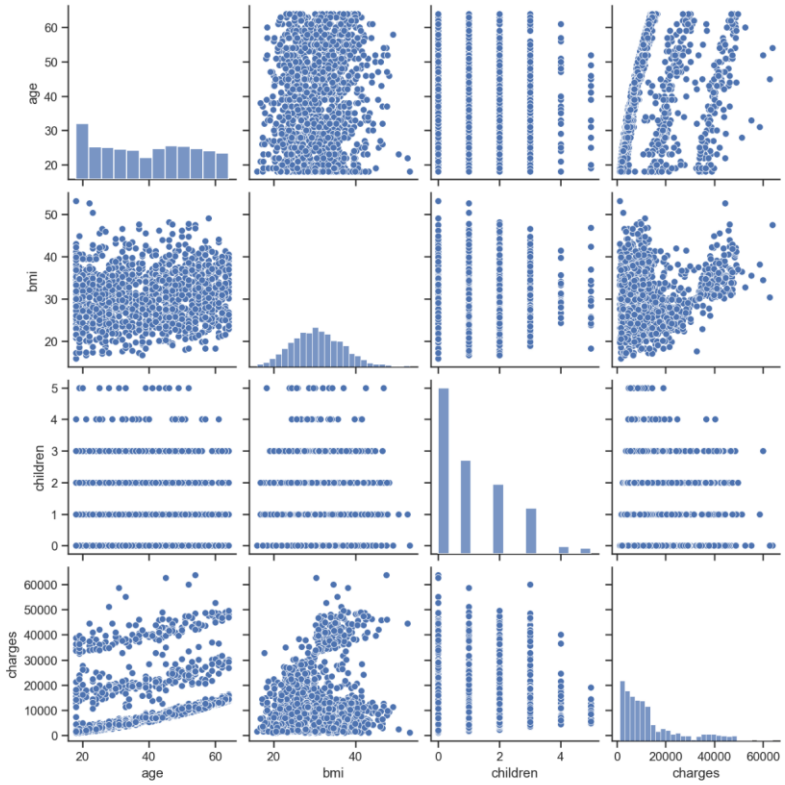
B [14]:
```
sns.kdeplot(data=data, x="charges", hue="smoker", cut=0, fill=True, common_norm=False, alpha=0.4)
```

Out[14]: <AxesSubplot:xlabel='charges', ylabel='Density'>



B [15]:
```
sns.pairplot(data)
```

Out[15]: <seaborn.axisgrid.PairGrid at 0x1151e4f8490>



B [16]:
```
corr_matrix = data.corr()
```

B [17]:
```
sns.heatmap(data.corr(), annot=True, fmt='.3f')
```

Out[17]: <AxesSubplot:>