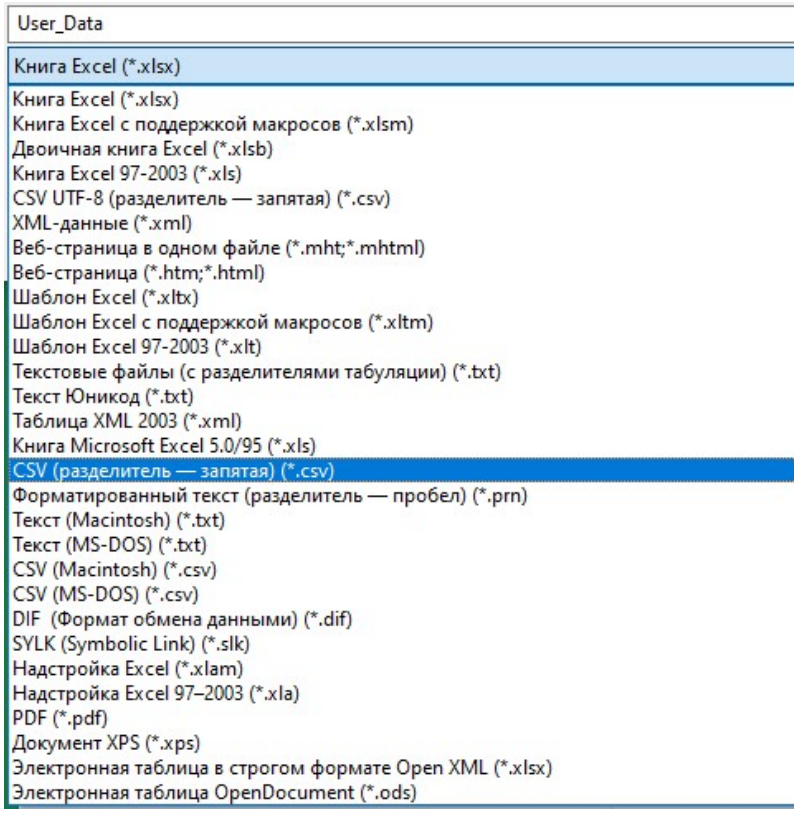
1.Данные в папке PredictModul – все необходимые данные для работы ML, должны находится в корневой папке бэкэнда

2. Пример того, в каком виде данные должны прийти от пользователя находится в файле User_Data_Example.csv

Данные в ML должны поступить именно в .csv с разделителем ‘,’



Данные по сути должны выглядеть так:

	date	time	station_name	-T-	| V |	_V_	pressure	humidity	precipitation
0	2020.12.31	12	koptevskii	-2	0.7	0	747.8	93	0

Пользователь вводит:

-дату, на которую делается предсказание (лучше выпадающим списком)

-время на который час предсказание (тоже лучше выпадающий, цифирный формат от 0 до 23)

- название геостанции (выпадающий список подробнее далее)

-T- температура воздуха (данные от метеостанций на день предсказания)

| V | - скорость ветра (данные от метеостанций на день предсказания)

_V_ направление ветра (данные от метеостанций на день предсказания)

pressure – давление (данные от метеостанций на день предсказания)

humidity – влажность (данные от метеостанций на день предсказания)

precipitation – осадки (данные от метеостанций на день предсказания)

3.Station Name (название гео станции) – было бы хорошо сделать выпадающим списком, чтобы пользователь не ломал голову как пишется та или иная станция

station_name
shabalovka
turistskaya
spiridonovka
proletarski
marino
koptevskii
glebovskaya
butlerova
anohina
ostankino

'shabalovka', 'turistskaya', 'spiridonovka', 'proletarski', 'marino', 'koptevskii', 'glebovskaya', 'butlerova', 'anohina', 'ostankino'

4.Чтобы вызвать ML необходимо набрать:

```
from Polution_modul import *

modelCO = polution_CO_model('model_CO','scaler')
modelCO.load_and_clean_data('User_Data.csv')
#modelCO.predicted_outputs()
modelCO.predicted_outputs().to_csv('modelCO_predictions.csv', index = False)

modelNO2 = polution_NO2_model('model_NO2','scaler')
modelNO2.load_and_clean_data('User_Data.csv')
#modelNO2.predicted_outputs()
modelNO2.predicted_outputs().to_csv('modelNO2_predictions.csv', index = False)

modelNO = polution_NO_model('model_NO','scaler')
modelNO.load_and_clean_data('User_Data.csv')
#modelNO.predicted_outputs()
modelNO.predicted_outputs().to_csv('modelNO_predictions.csv', index = False)

modelPM10 = polution_PM10_model('model_PM10','scaler')
modelPM10.load_and_clean_data('User_Data.csv')
#modelPM10.predicted_outputs()
modelPM10.predicted_outputs().to_csv('modelPM10_predictions.csv', index = False)

modelPM25 = polution_PM25_model('model_PM25','scaler')
modelPM25.load_and_clean_data('raw_check_ML_data.csv')
#modelPM25.predicted_outputs()
modelPM25.predicted_outputs().to_csv('modelPM25_predictions.csv', index = False)
```

5.Соответственно результаты ML запишутся в файлы

'modelCO_predictions.csv'

'modelNO2_predictions.csv'

'modelNO_predictions.csv'

'modelPM10_predictions.csv'

'modelPM25_predictions.csv'

В таком виде:

```
modelCO.predicted_outputs()
```

	Unnamed: 0	date	time	station_name	-T-	| V |	_V_	pressure	humidity	precipitation	season	week_day	Probability	CO
0	0	2020-12-31	12	koptevskii	-2	0.7	0	747.8	93	0	12	4	0.864403	0.183633

```
modelNO2.predicted_outputs()
```

	Unnamed: 0	date	time	station_name	-T-	| V |	_V_	pressure	humidity	precipitation	season	week_day	Probability	NO2
0	0	2020-12-31	12	koptevskii	-2	0.7	0	747.8	93	0	12	4	0.700586	0.015609

```
modelNO.predicted_outputs()
```

	Unnamed: 0	date	time	station_name	-T-	| V |	_V_	pressure	humidity	precipitation	season	week_day	Probability	NO
0	0	2020-12-31	12	koptevskii	-2	0.7	0	747.8	93	0	12	4	0.766441	0.009064

```
modelPM10.predicted_outputs()
```

	Unnamed: 0	date	time	station_name	-T-	| V |	_V_	pressure	humidity	precipitation	season	week_day	Probability	PM10
0	0	2020-12-31	12	koptevskii	-2	0.7	0	747.8	93	0	12	4	0.472715	0.008508

```
modelPM25.predicted_outputs()
```

	Unnamed: 0	date	time	station_name	-T-	| V |	_V_	pressure	humidity	precipitation	season	week_day	Probability	PM2.5
0	0	2020-12-31	12	koptevskii	-2	0.7	0	747.8	93	0	12	4	0.214412	0.004293

Где

Первые столбцы — это данные, введенные пользователем

Последний столбец – предсказание количества вещества в воздухе на выбранную дату, время

Столбец Probability показывает вероятность превышения среднего показателя по этому веществу

6. В папке так же присутствуют следующие файлы:

- building_density.csv

Плотность застройки по районам Москвы

Файл используется MLблоком для препроцессинга сырых данных от пользователя

При желании, данный файл можно менять, с условием, что название и формат остаются прежними

Внутри файл структурирован следующим образом:

Unnamed: 0	Moscow_region	station_name	dencity_coef
0	ЮАО	shabalovka	2188
1	СЗАО	turistskaya	1886
2	ЦАО	spiridonovka	4255
3	ЮАО	proletarski	2188
4	ЮВАО	marino	1878
5	САО	koptevskii	2051
6	ВАО	glebovskaya	1644
7	ЮЗАО	butlerova	2600
8	ЗАО	anohina	1838
9	СВАО	ostankino	2394

- factory_dencity.csv

Индексы промышленного производства предприятий Москвы

Файл используется MLблоком для препроцессинга сырых данных от пользователя

При желании, данный файл можно менять, с условием, что название и формат остаются прежними

Внутри файл структурирован следующим образом:

Unnamed: 0	season	industrial	electricity	processing	water_supply
0	1	80.3	100.0	78.0	63.0
1	2	110.0	95.0	114.0	95.0
2	3	100.3	93.8	101.8	102.5
3	4	92.0	85.0	95.0	95.0
4	5	90.0	60.0	102.0	100.0
5	6	114.2	84.9	117.8	110.1
6	7	107.0	120.0	105.0	105.0
7	8	104.0	100.0	102.8	110.3
8	9	102.2	102.8	102.8	95.5
9	10	104.0	153.0	100.0	110.0
10	11	107.0	125.0	105.0	96.0
11	12	110.5	129.1	106.8	147.9

где:

season – нумерация месяца с января по декабрь (от 1 до 12)

industrial-промышленное производство

electricity- обеспечение электрической энергией, газом и паром; кондиционирование воздуха

processing -обрабатывающие производства

water_supply - водоснабжение; водоотведение, организация сбора и утилизации отходов, деятельность по ликвидации загрязнений

- traffic_day_dencity.csv

Плотность трафика на дорогах Москвы в зависимости от дня недели и от времени

Файл используется MLблоком для препроцессинга сырых данных от пользователя

При желании, данный файл можно менять, с условием, что название и формат остаются прежними

Внутри файл структурирован следующим образом:

Unnamed: 0	time	week_day	traffic
0	1	1	0.0
1	2	1	0.0
2	3	1	0.0
3	4	1	0.0
4	5	1	0.0
...	...	...	...
163	20	7	2.0
164	21	7	1.5
165	22	7	1.3
166	23	7	1.0
167	0	7	0.8

где:

time - время в часах, когда оценивался уровень пробок(от 0 до 23)

week_day – день недели с Понедельника по Воскресенье (от 1 до 7)

- traffic_season_dencity.csv

Плотность трафика на дорогах Москвы в зависимости от месяца

Файл используется MLблоком для препроцессинга сырых данных от пользователя

При желании, данный файл можно менять, с условием, что название и формат остаются прежними

Внутри файл структурирован следующим образом:

Unnamed: 0	season	season_traffic
0	1	4.9
1	2	4.4
2	3	4.6
3	4	4.9
4	5	4.6
5	6	4.6
6	7	3.9
7	8	4.3
8	9	4.7
9	10	4.9
10	11	5.2
11	12	5.6

где:

season – нумерация месяца с января по декабрь (от 1 до 12)

- df_inversion.csv

Наличие температурных инверсий на высотах от 0 до 200, от 200 до 400, от 400 до 600

Внутри файл структурирован следующим образом:

Unnamed: 0	time	season	week_day	inversion_high200	inversion_high400	inversion_high600
0	0	1	3	1.0	0.0	0.0
1	0	1	3	1.0	0.0	1.0
2	0	1	3	1.0	0.0	0.0
3	0	1	3	0.0	1.0	0.0
4	0	1	3	1.0	0.0	0.0
...	...	...	...	...	...	...
85615	23	12	7	0.0	0.0	1.0
85616	23	12	7	1.0	0.0	0.0
85617	23	12	7	0.0	1.0	1.0
85618	23	12	7	1.0	1.0	1.0
85619	23	12	7	0.0	0.0	1.0

где:

time - время в часах, когда оценивался уровень пробок (от 0 до 23)

season – нумерация месяца с января по декабрь (от 1 до 12)

week_day – день недели с Понедельника по Воскресенье (от 1 до 7)

- wind253.csv

Данные с Останкино с высоты 253 метра о силе ветра и направлении

Файл используется MLблоком для препроцессинга сырых данных от пользователя

При желании, данный файл можно менять, с условием, что название и формат остаются прежними

Внутри файл структурирован следующим образом:

Unnamed: 0	time	_V0_	| V0 |	season	week_day
0	0	300.0	9.0	1	3
3	1	300.0	9.1	1	3
6	2	320.0	4.8	1	3
9	3	320.0	9.1	1	3
12	4	330.0	7.9	1	3
...	...	...	...	...	...
26337	19	180.0	3.5	12	4
26340	20	180.0	8.5	12	4
26343	21	180.0	7.1	12	4
26346	22	180.0	7.4	12	4
26349	23	170.0	7.4	12	4

где:

time - время в часах, когда оценивался уровень пробок (от 0 до 23)

season – нумерация месяца с января по декабрь (от 1 до 12)

week_day – день недели с Понедельника по Воскресенье (от 1 до 7)

_V0_ направление ветра

| V0 | скорость ветра

- prepared_Final_data.csv

Технический файл, используется MLблоком для нахождения средних значений и стандартных девиаций