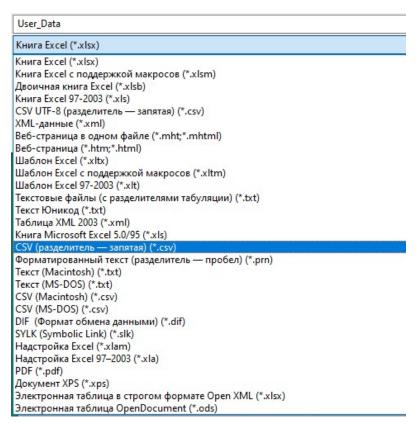
- 1.Данные в папке PredictModul все необходимые данные для работы ML, должны находится в корневой папке бэкэнда
- 2. Пример того, в каком виде данные должны прийти от пользователя находится в файле User\_Data\_Example.csv

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)
- -Т- 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

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

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. Cooтветственно результаты ML запишутся в файлы

'modelCO\_predictions.csv'

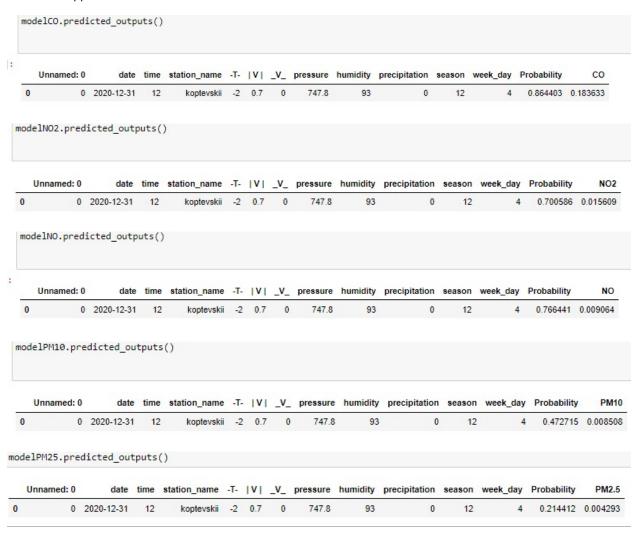
'modelNO2\_predictions.csv'

'modelNO\_predictions.csv'

'modelPM10\_predictions.csv'

'modelPM25\_predictions.csv'

# В таком виде:



### 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 МЬблоком 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:

dencity_coef	station_name	Moscow_region	Unnamed: 0
2188	shabalovka	0 HOAO shaba	
1886	turistskaya	C3AO	1
4255	spiridonovka	ЦАО	2
2188	proletarski	ЮАО	3
1878	marino	ЮВАО	4
2051	koptevskii	CAO	5
1644	glebovskaya	BAO	6
2600	butlerova	ЮЗАО	7
1838	anohina	3A0	8
2394	ostankino	CBAO	9

## - factory\_dencity.csv

Индексы промышленного производства предприятий Москвы

Файл используется МЬблоком 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:

деятельность по ликвидации загрязнений

# 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,

- traffic\_season\_dencity.csv

Плотность трафика на дорогах Москвы по сезонно (по месяцам)

Плотность застройки по районам Москвы

Файл используется МЬблоком 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:

			traffic	
Unnamed: 0	time	week_day		
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)

- 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
					(4)	(1999)
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 МЬблоком 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