

# Classifier

## Описание

Программа выполняет классификацию аэрозоля по трем типам:

- пылевой
- городской
- сажевый

разделение происходит на основании анализа двух параметров:

- $Gf$  - емкость флуоресценции
- $\delta_p$  - аэрозольная деполяризация

Основанием для классификации служит статья (<https://doi.org/10.5194/amt-17-3367-2024>)

для работы программы нужно указать реперные значения  $Gf$ ,  $\delta$  для каждого из рассматриваемых типов аэрозолей (urban, soot, dust)

## Алгоритм работы

Доля вклада каждого компонента в измеренные значения  $Gf_m$  и  $\delta_m$  решается следующая система линейных уравнений

$$\begin{cases} Gf_d \cdot n_d + Gf_u \cdot n_u + Gf_s \cdot n_s = Gf_m \\ \delta'_d \cdot n_d + \delta'_u \cdot n_u + \delta'_s \cdot n_s = \delta_m \\ n_d + n_u + n_s = 1 \end{cases}$$

При этом  $\delta' = \delta / (1 + \delta)$ .

Решение этой системы методом наименьших квадратов позволяет разложить влияние каждого из типов аэрозолей в смесь, а также уточнить значения параметров для каждого из классов.

Реперные значения классов аэрозолей мы задаем с некоторой погрешностью, при этом они напрямую отвечают за результирующее решение  $(n_d, n_u, n_s)$ . чтобы получить статистически обоснованное решение и получить уточненные значения параметров классов мы выполняем моделирование, для этого мы создаем  $N$  наборов реперных значений  $(Gf_d^i, \delta_d^i, Gf_u^i, \delta_u^i, Gf_s^i, \delta_s^i, i = 1..N)$ , которые располагаются вблизи изначально заданных классов и имеют нормальное распределение, параметры которого контролируются параметрами запуска программы. Для каждого набора мы решаем систему линейных уравнений. На выходе мы получаем вектор вида:

$$\alpha_i = (n_d \quad n_u \quad n_s \quad F \quad Gf_d \quad \delta_d \quad Gf_u \quad \delta_u \quad Gf_s \quad \delta_s)$$

Помимо уже известных нам величин,  $F$  - соответствует L2 норме нашей невязки в решении систему уравнений

Далее, мы ранжируем наши векторы решений, в порядке возрастания и берем  $p\%$  полученных решений с наименьшим значением  $F$  и усредняем. В результате мы получим

$$\langle \alpha \rangle = (n_d \quad n_u \quad n_s \quad F \quad Gf_d \quad \delta_d \quad Gf_u \quad \delta_u \quad Gf_s \quad \delta_s)$$

вектор, состоящий из средних значений каждого из компонентов

Этот этап повторяется для каждого из измерений  $(Gf, \delta)$

Полученные результаты оформляются в виде матрицы, которая сохраняется в csv и pdf

## Параметры командной строки

```
./classifier -h
```

Usage of temp:

```
-config string
```

path to config file (default "config.yml")

Usage of ./classifier:

```
-delta-dust float
```

реперное значение аэрозольной деполяризации, соответствующее пыли (центр кластера из статьи) (default 0.26)

```
-delta-soot float
```

реперное значение аэрозольной деполяризации, соответствующее саже (центр кластера из статьи) (default 0.06)

```
-delta-urban float
```

реперное значение аэрозольной деполяризации, соответствующее городскому аэрозолю (центр кластера из статьи) (default 0.05)

```
-gf-dust float
```

реперное значение флуоресцентной емкости пыли (центр кластера из статьи)

fluorescence capacity of dust aerosol (default 3e-05)

```
-gf-soot float
```

реперное значение флуоресцентной емкости сажи (центр кластера из

статьи)

fluorescence capacity of soot aerosol (default 0.0004)

-gf-urban float

реперное значение флуоресцентной емкости городского аэрозоля (центр кластера из статьи)

fluorescence capacity of urban aerosol (default 5.5e-05)

-input-dir string

путь к папке где лежат файлы (FL\_cap.txt, dep.txt) (default "../examples/test")

-num-points int

размер моделируемого набора данных параметров кластеров (default 100)

-avg-percent float

доля профилей для усреднения (default 0.1 от числа, заданного в num-points )

-sigma-h int

полуширина сглаживающего окна по высоте в отсчетах (default 5)

-sigma-t int

полуширина сглаживающего окна по времени в отсчетах (default 3)

-size int

размер сглаживающего окна в точках (default 7)

-smooth

флаг, указывающий необходимость сглаживания (default false)

-var-coef float

величина, указывающая на сколько сильно нужно делать разброс значений при моделировании набора данных параметров кластеров (default 0.1)

Все эти параметры можно задавать явно, либо указать в файле config.yml

gf\_urban: 0.55e-4

gf\_soot: 4e-4

gf\_dust: 0.3e-4

delta\_urban: 0.05

```
delta_soot: 0.06
delta_dust: 0.26
variation_coefficient: 0.1
input_dir: "../examples/test"
num_points: 100
do_smooth: true
sigma_h: 5
sigma_t: 3
size: 7
avg_percent: 0.1
```

При запуске программа сначала ищет файл `config.yml`, читает оттуда настройки, потом проверяет, есть ли какие другие параметры, указанные явно. Если есть - применяет их к уже считанным из файла. Если `config.yml` не найден, программа считывает настройки из флагов, если и их нет - использует значения по умолчанию

В результате мы получаем следующие файлы в той же директории, где лежат входные файлы `FL_cap.txt` и `dep.txt`

- `delta_d.csv`, `delta_u.csv`, `delta_s.csv` - файлы с уточненными значениями `delta`
- `Gf_d.csv`, `Gf_u.csv`, `Gf_s.csv` - файлы с уточненными значениями емкости флуоресценции
- `Eta_d.csv`, `Eta_u.csv`, `Eta_s.csv` - файлы с рассчитанными вкладами в смесь `dust`, `urban` и `soot`
- `Eta_d.pdf`, `Eta_u.pdf`, `Eta_s.pdf` - графическое представление данных `Eta`
- `dep.pdf` - графическое представление деполяризации
- `rand.csv` - смоделированные данные для реперных кластеров

## Сглаживание контролируется параметрами

- `sigma-h`
- `sigma-t`
- `size`
- `smooth`

## Статистика данных контролируется параметрами

- `num-points`
- `var-coef`
- `avg-percent`