# *Классификация биологических образцов (клеток) на раковые и нераковые по их спектрам рамановского рассеяния*

Студент: Мердалимова Анастасия

Skillfactory

Data Science Track

# Содержание

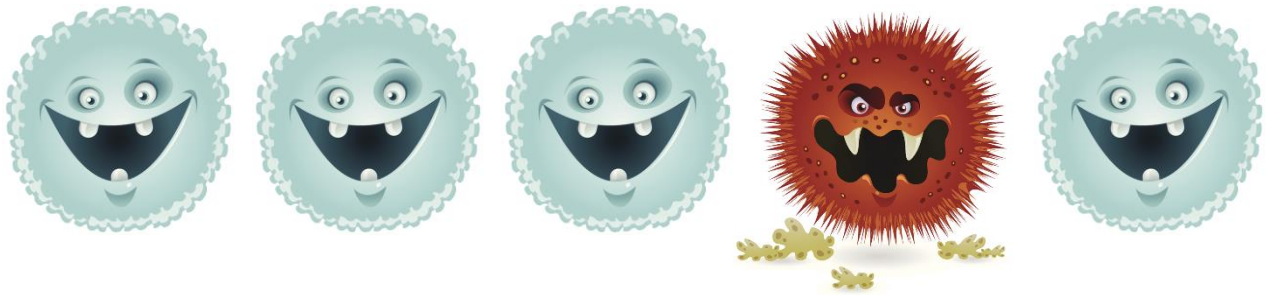
- Введение в тему
- Постановка задачи
- Исследование данных
- Применение методов машинного обучения
  - Снижение размерности
  - Классификация
- Модификация задачи

# Содержание

- **Введение в тему**
- Постановка задачи
- Исследование данных
- Применение методов машинного обучения
  - Снижение размерности
  - Классификация
- Модификация задачи

# Тема

- **Что?** **Диагностика опухолевых клеток** на ранней стадии



# Тема

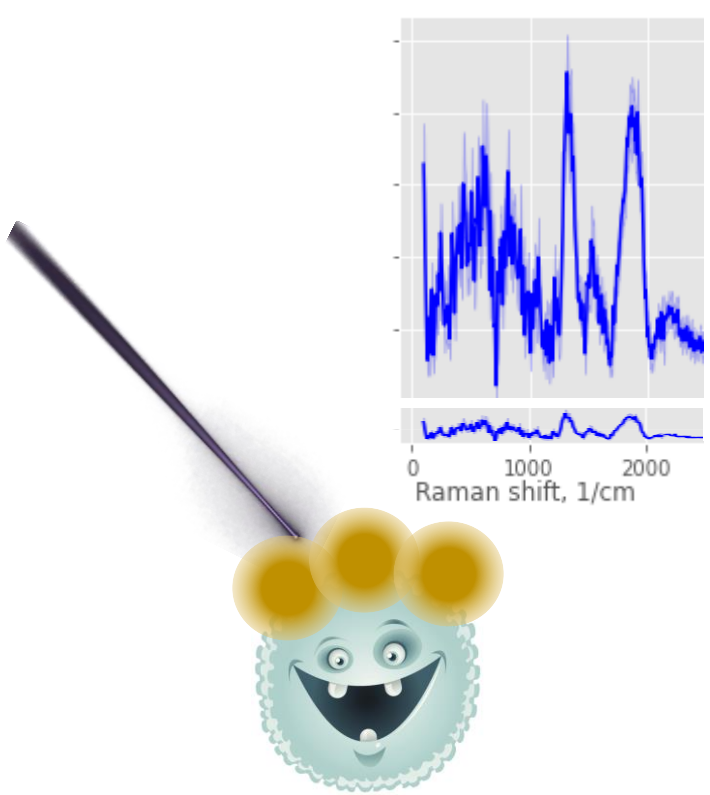
- **Как?**

**Рамановская спектроскопия**

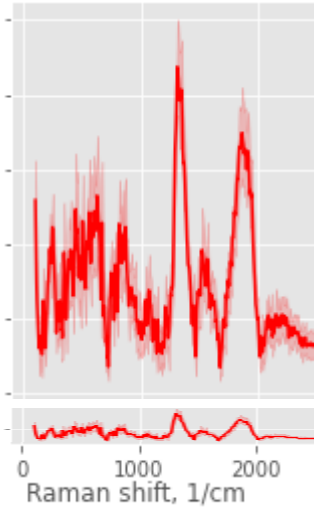
**+ усиливающие золотые частицы (AuNPs)**

Уровень сигнала слабый :(

Усиливают только вблизи себя :(





# Тема

- **Как?**

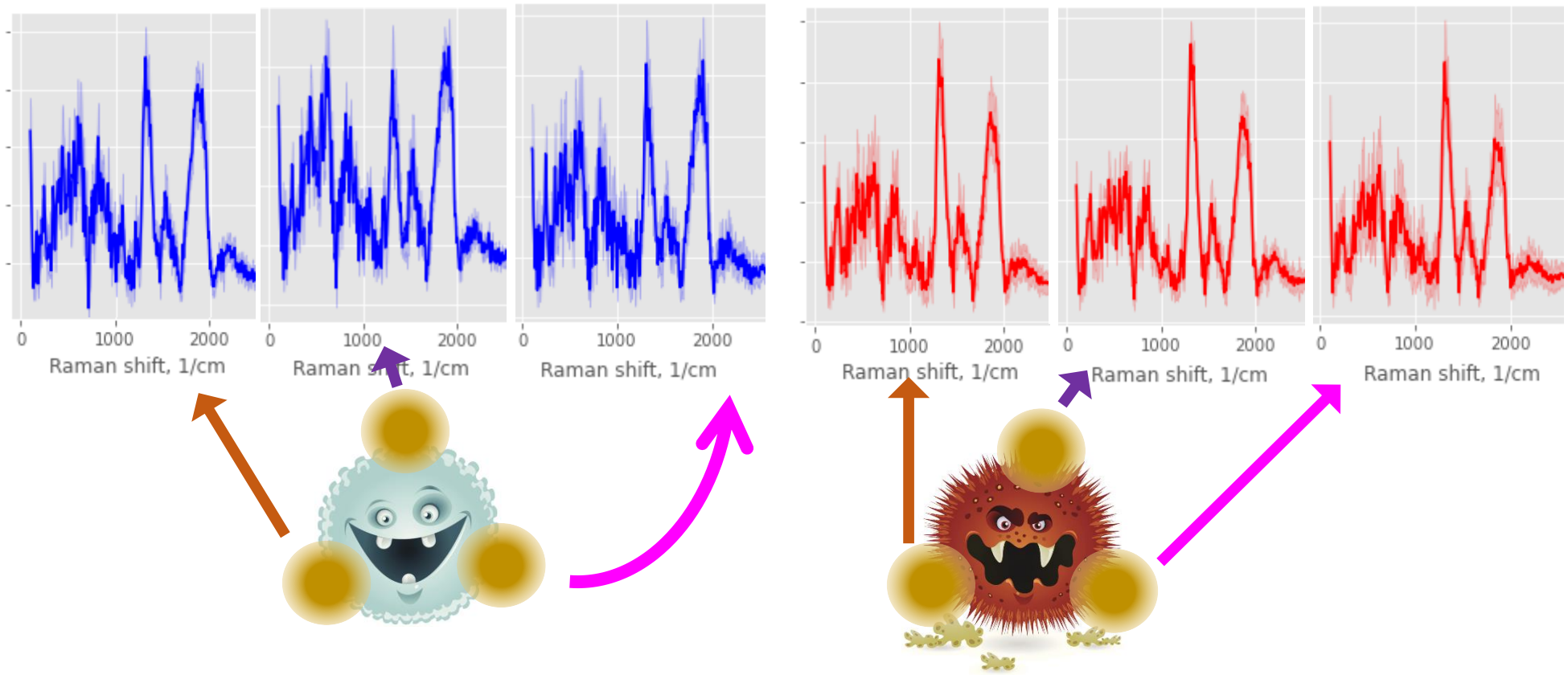
**Рамановская спектроскопия**

**+ усиливающие золотые частицы (AuNPs)**

**трех разных типов!**

Уровень сигнала слабый :(

Усиливают только вблизи себя :(



# Тема

- **Как?**

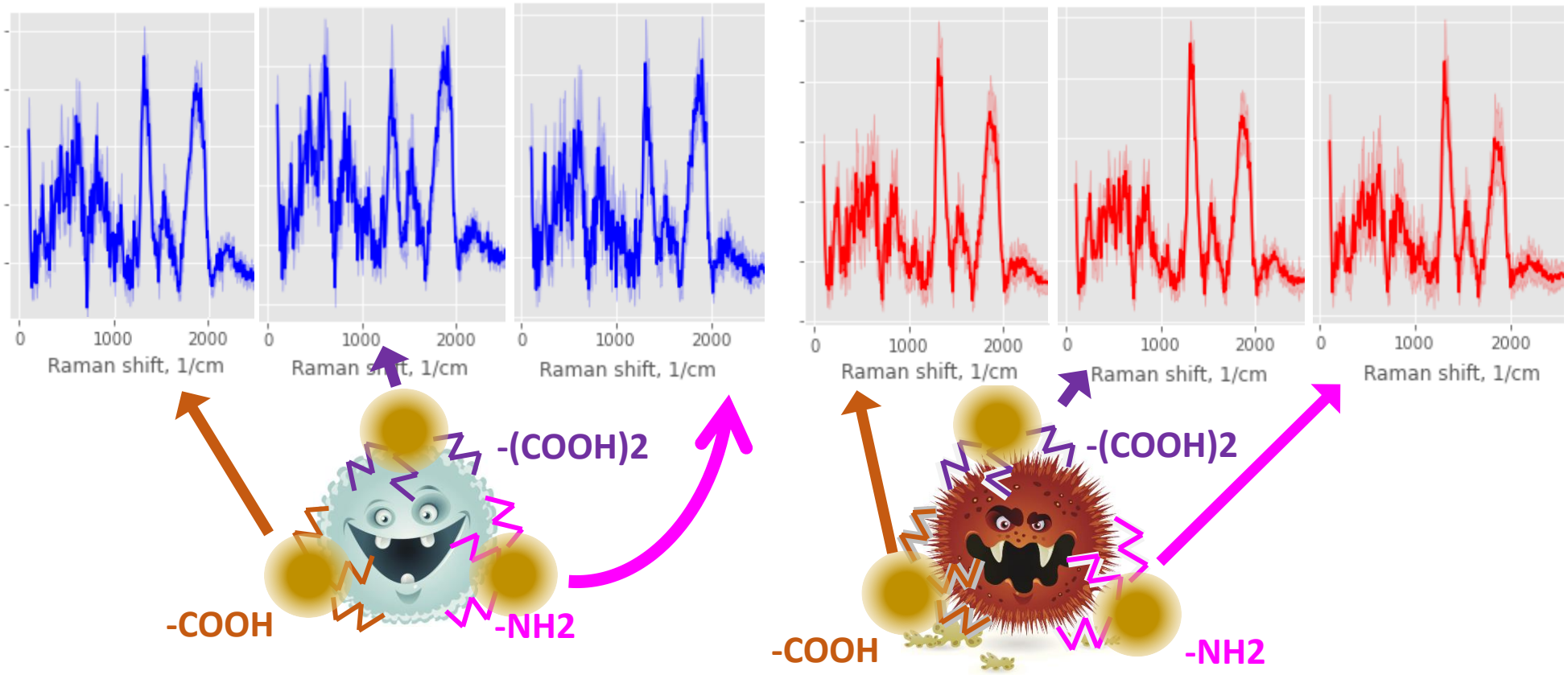
**Рамановская спектроскопия**

**+ усиливающие золотые частицы (AuNPs)**

**ТРЕХ разных типов!**

Уровень сигнала слабый :(

Усиливают только вблизи себя :(



# Содержание

- Введение в тему
- **Постановка задачи**
- Исследование данных
- Применение методов машинного обучения
  - Снижение размерности
  - Классификация
- Модификация задачи

# Задача

- Различать здоровые и раковые клетки по спектрам усиленного рамановского рассеяния

- Сейчас используется 3 типа частиц.
  Было бы здорово, если бы требовался только 1

## Формализованная задача

- **Задача классификации**

- **Исследование качества модели в зависимости от кол-ва входных признаков**

# Данные

**Покрытие частиц**

| Типы клеток | (COOH)2 | COOH | NH2 | Total |
|---|---|---|---|---|
| A | 53 | 53 | 59 | 165 |
| A-S | 51 | 56 | 50 | 157 |
| DMEM | 64 | 64 | 65 | 193 |
| DMEM-S | 53 | 52 | 53 | 158 |
| G | 52 | 54 | 51 | 157 |
| G-S | 50 | 51 | 50 | 151 |
| HF | 56 | 50 | 51 | 157 |
| HF-S | 50 | 51 | 50 | 151 |
| MEL | 49 | 50 | 50 | 149 |
| MEL-S | 50 | 52 | 51 | 153 |
| ZAM | 50 | 50 | 50 | 150 |
| ZAM-S | 49 | 50 | 52 | 151 |

**Healthy cells**

**Cancer cells**

*https://www.kaggle.com/datasets/andriitrelin/cells-raman-spectra*

*Erzina et al. Sensors & Actuators: B. Chemical 308 (2020) 127660*

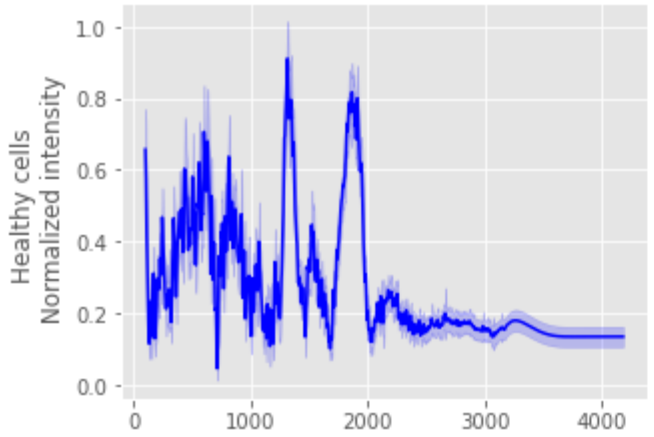
# Знакомство с данными

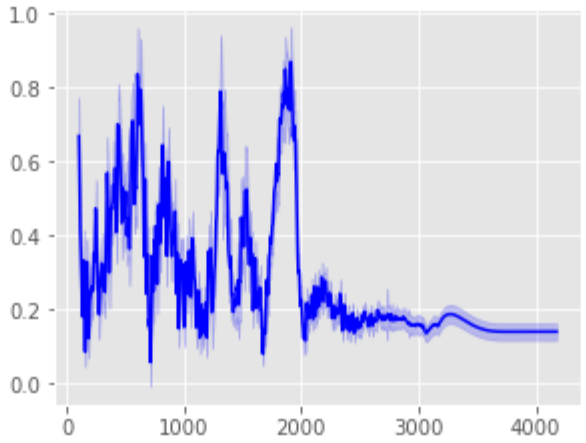
Усиление частицами с **-(COOH)2**

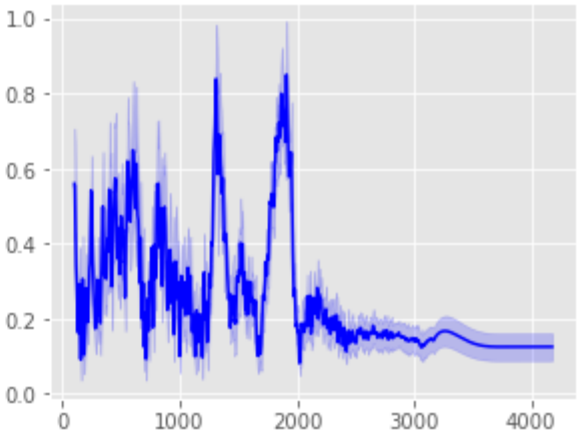
Усиление частицами с **-COOH**

Усиление частицами с **-NH2**

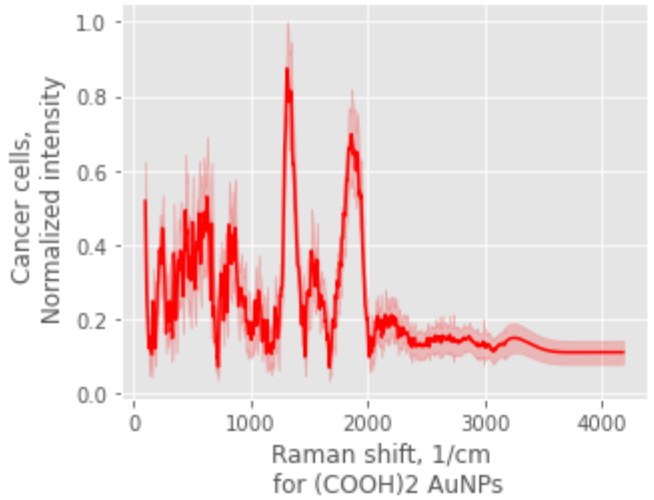
**Healthy cells**

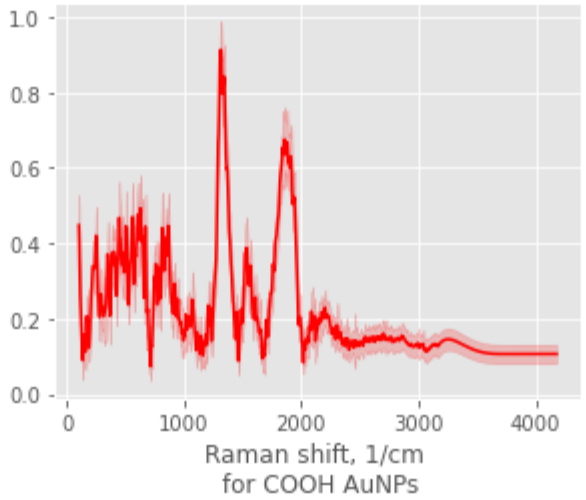


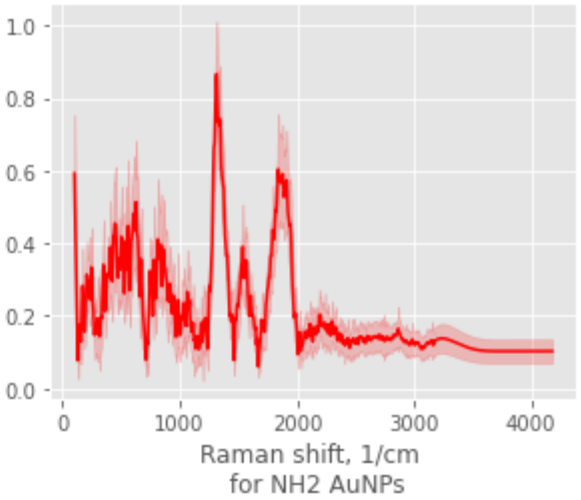




**Cancer cells**




В каждом спектре:

**2000**

**признаков**

Итого на один образец:

**6000**

**признаков**

# Содержание

- Введение в тему
- Постановка задачи
- **Исследование данных**
- Применение методов машинного обучения
  - Снижение размерности
  - Классификация
- Модификация задачи

# EDA и очистка

- Пропуски в данных: нет
- Выбросы: нет

# Статистический анализ

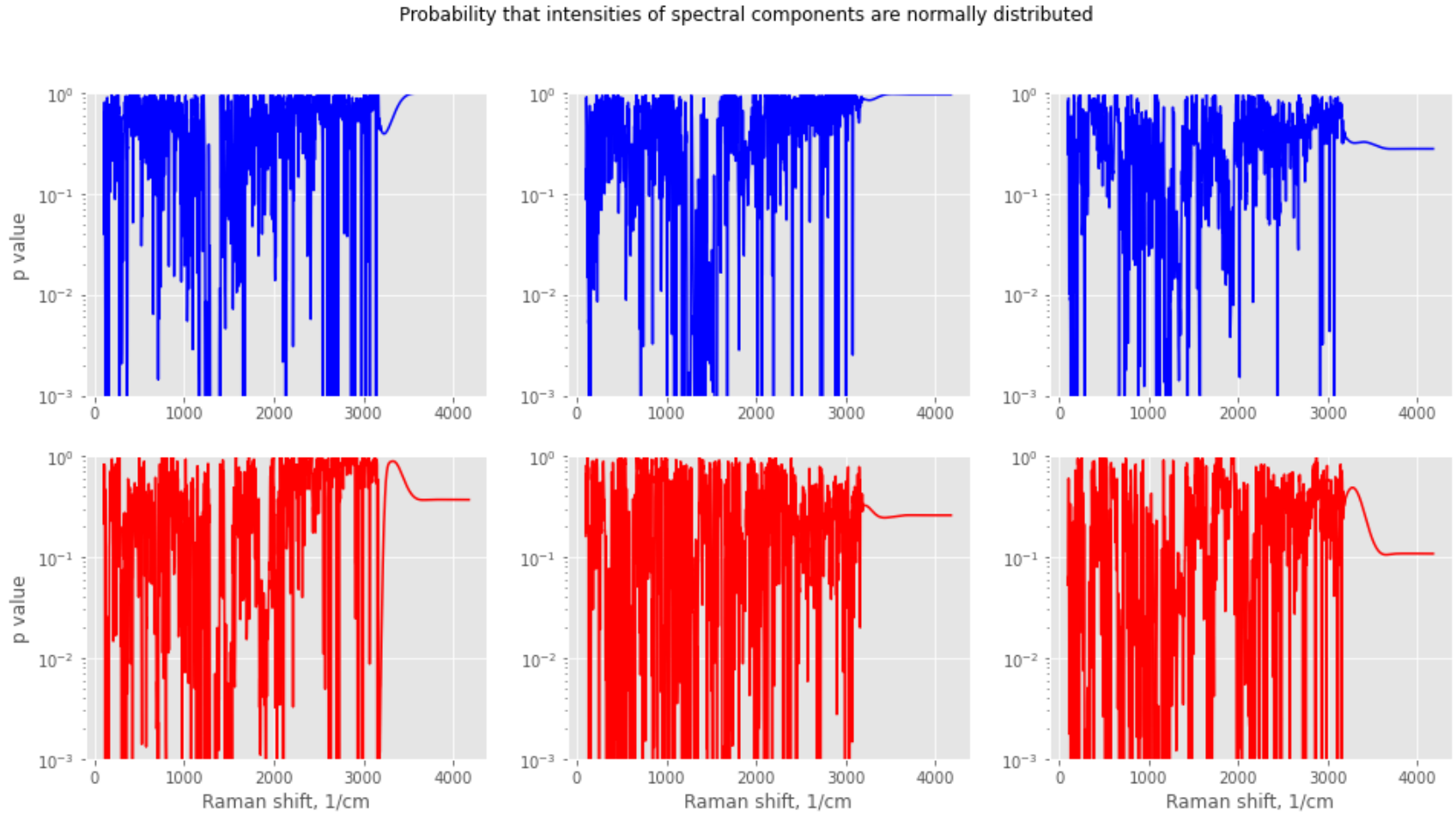
- Поищем спектральные компоненты, отличные у здоровых и больных нормализованных спектров. Чтобы можно было построить модель/бейзлайн без использования ML.

Ожидание:

1) **Проверим распределение в каждом признаке на нормальность**
2) **Выберем тип теста**
3) **Протестируем**

# Статистический анализ

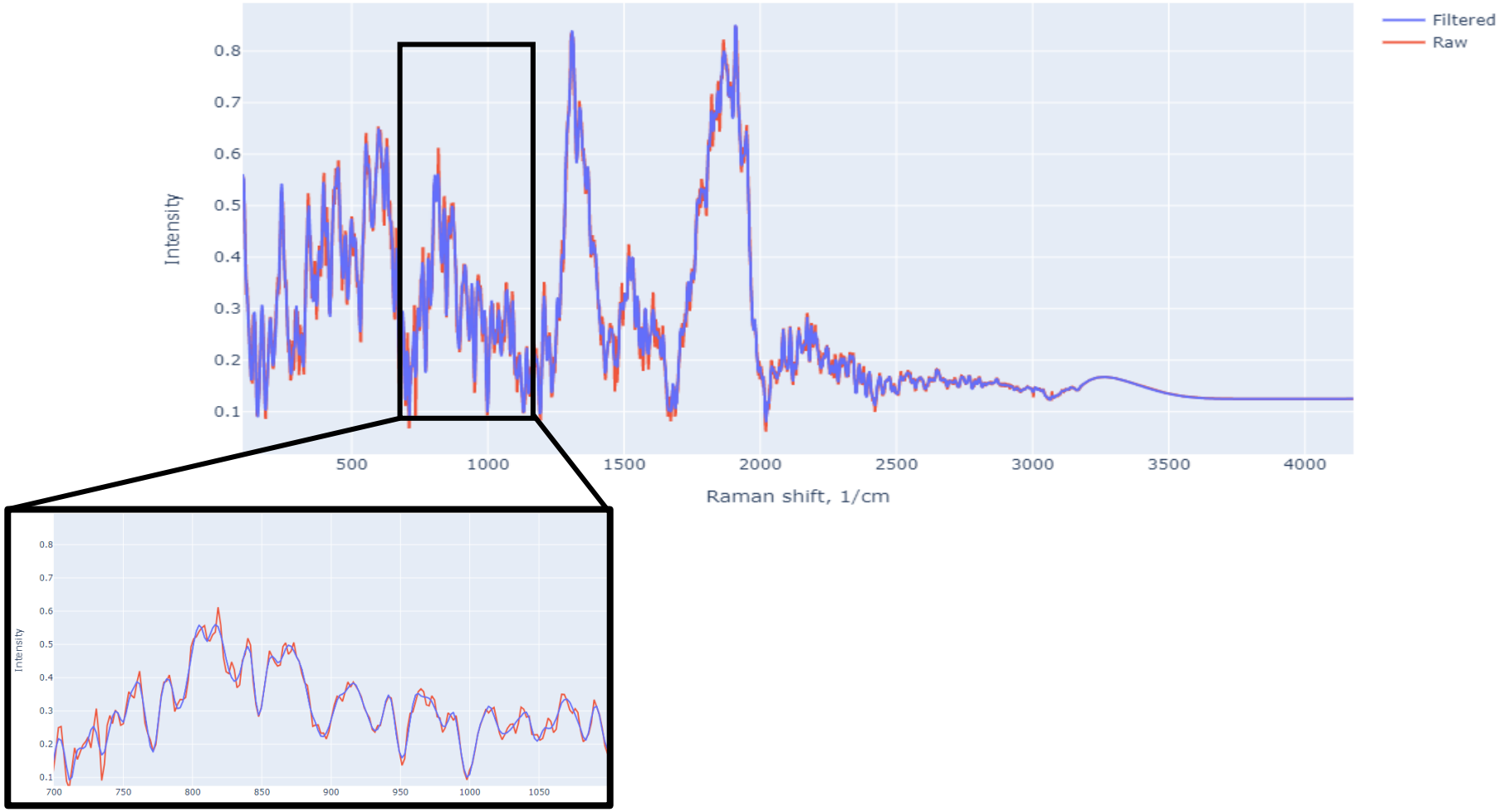
- Поищем спектральные компоненты, отличные у здоровых и больных нормализованных спектров.
  1) Проверим распределение в каждом признаке на нормальность




# Результат фильтра Савицкого-Голея

Trade-off between smoothing and informativity, as some peaks also may be sharp

Mean spectrum of filered HF NH2 spectra

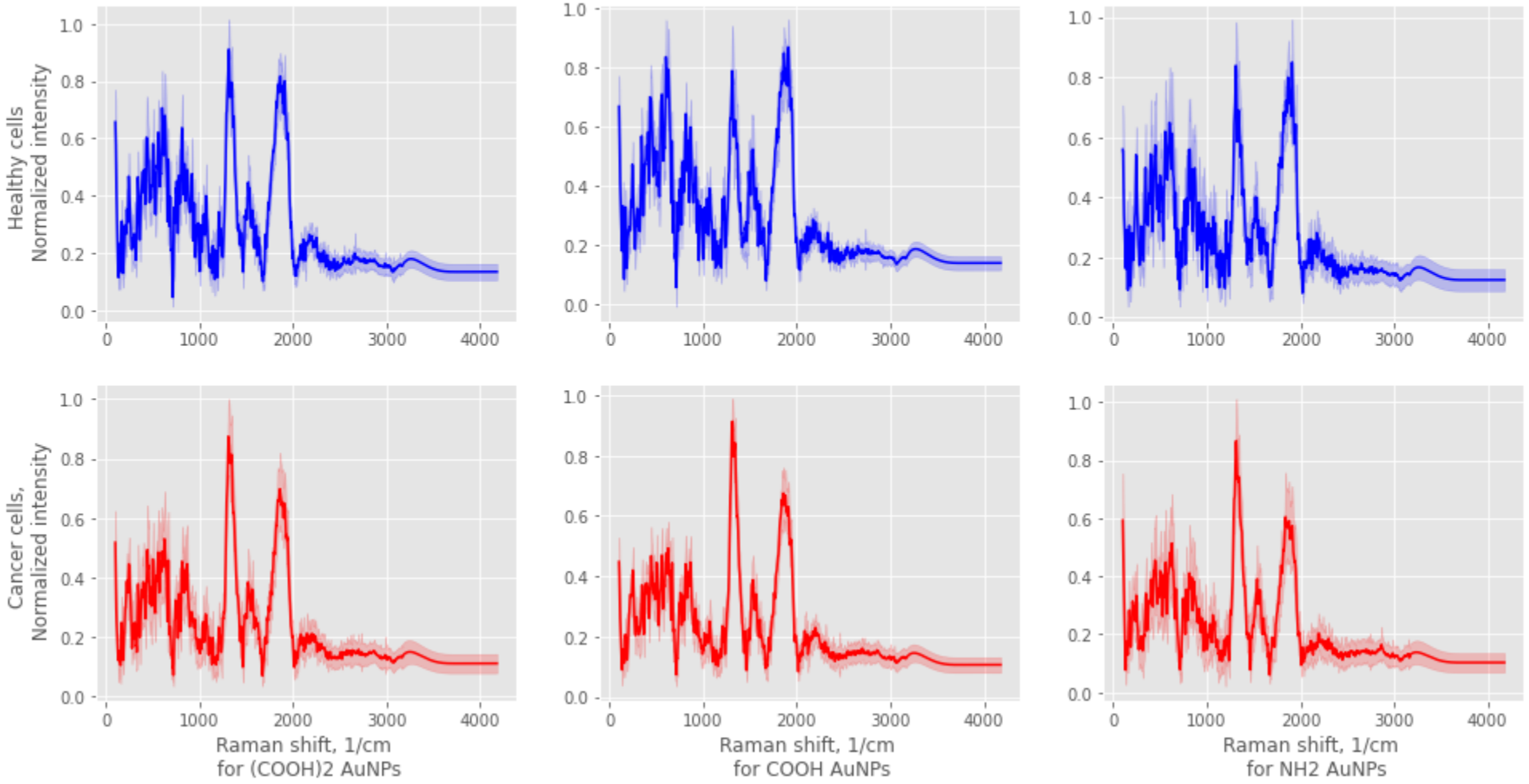


# Результат фильтра Савицкого-Голея

All spectra studied, mean+-std:

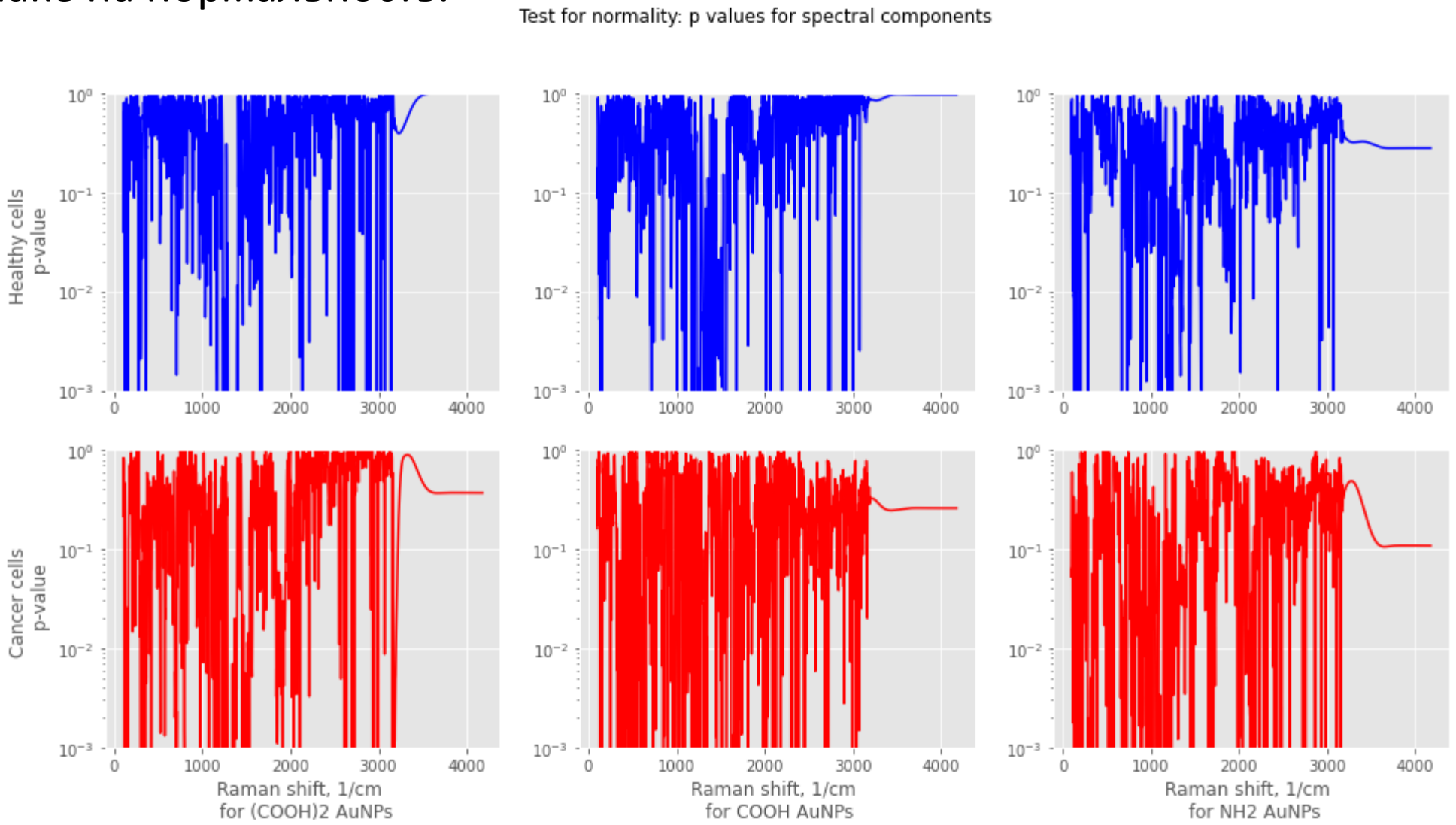
**Healthy cells**

**Cancer cells**



# Статистический анализ

После фильтра Савицкого-Голая, снова проверим распределение в каждом признаке на нормальность.

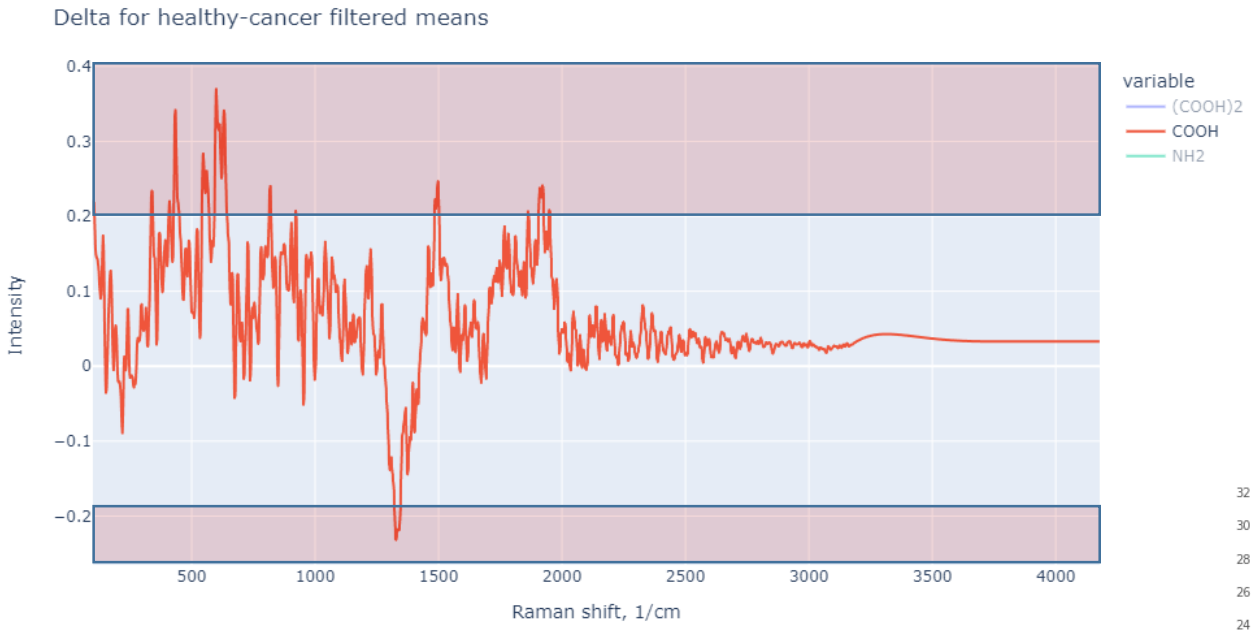



Снова очень разные p-value.

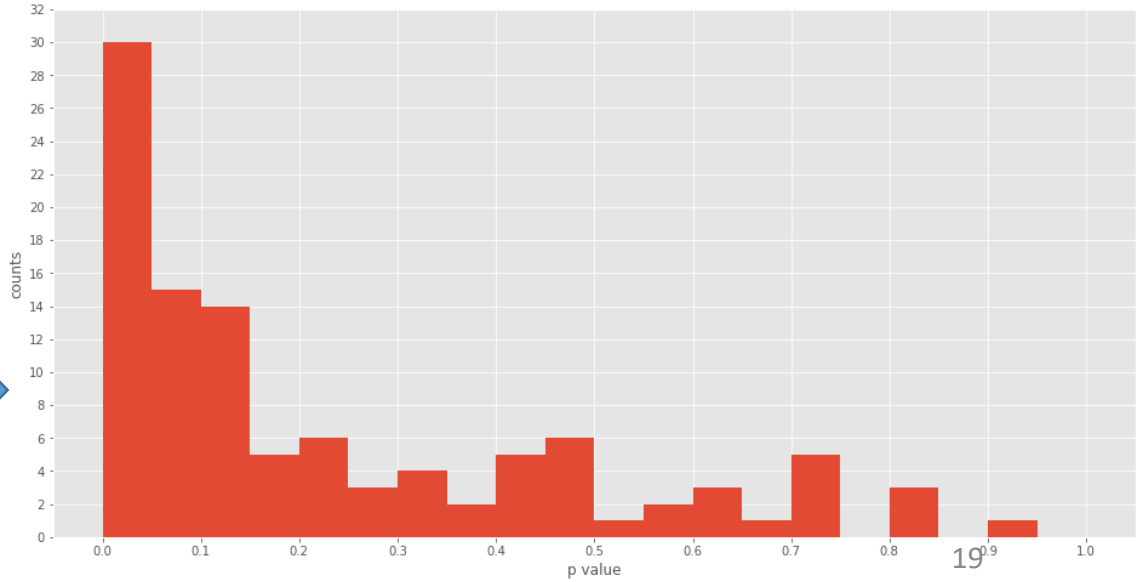
Но нам и не нужны все, нам нужны только в пиках

# Интересующие пики (фичи)

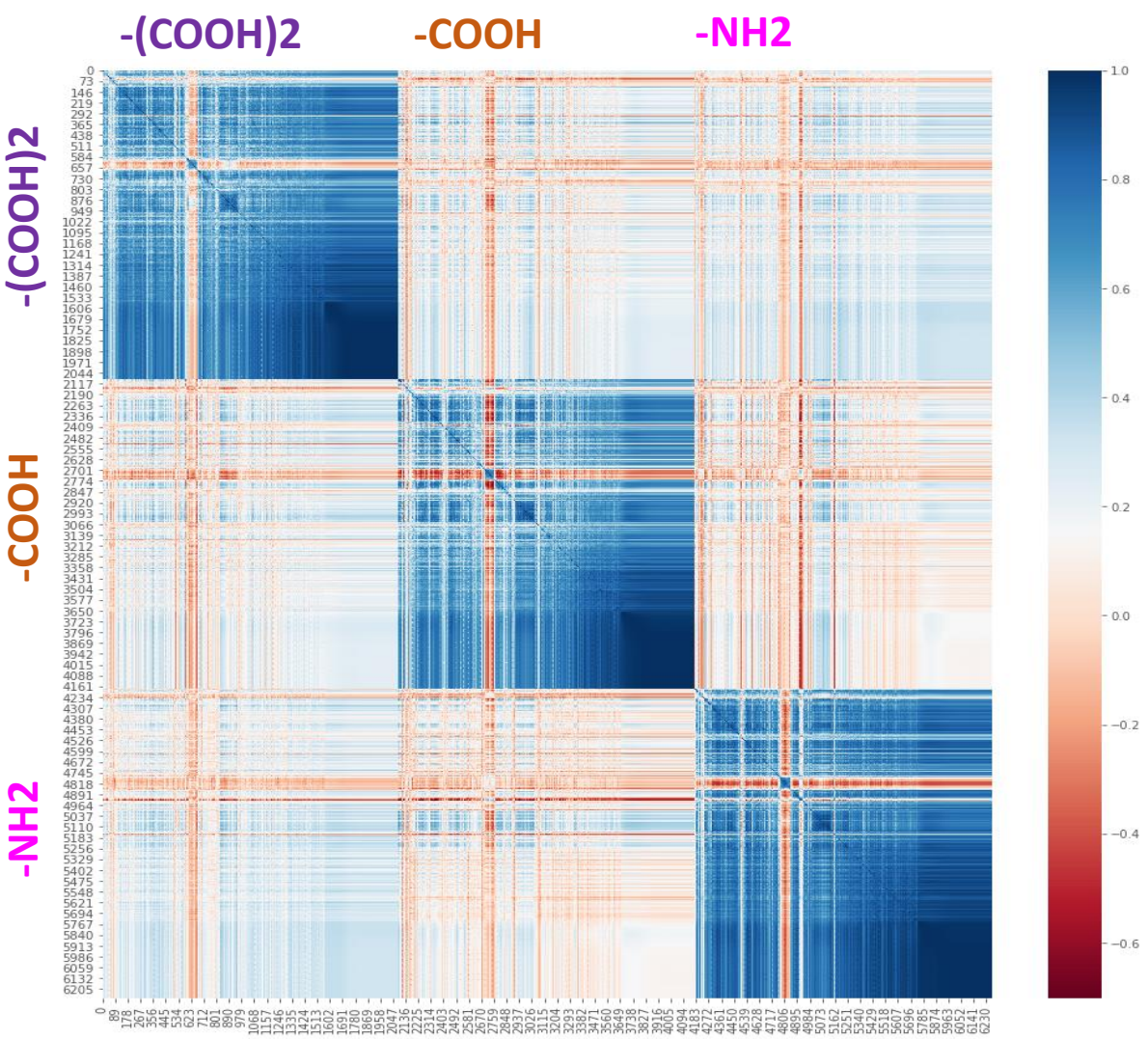
- Delta between mean spectra:




- Выделим те компоненты, где abs(delta) > 0.2
  Для COOH таких 106 штук.
- Проверим на нормальность их – получим такую гистограмму:
- Видим, что p-value<0.05 у 30 точек из 106.



# Correlation matrix




- Используем **корреляцию Спирмена**, т.к. уже знаем, что не все признаки имеют нормальное распределение
- Видим: данные в одном спектре скоррелированы между собой
- Вывод: стоит попробовать **снижение размерности**

# Содержание

- Введение в тему
- Постановка задачи
- Исследование данных
- Применение методов машинного обучения
  - **Снижение размерности**
  - Классификация
- Модификация задачи

# Снижение размерности: PCA

Тысячи признаков

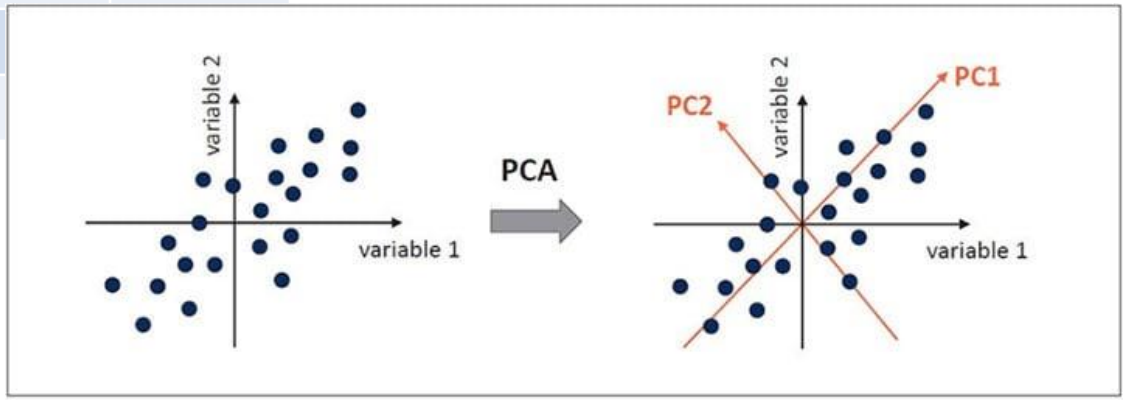
Данные очень большой размерности!

| Wave-number | Sample 1 Intensity | Sample 2 Intensity |
|---|---|---|
| 600 | | |
| 601 | | |
| 602 | | |
| 603 | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |

| Principal component | Sample 1 Intensity | Sample 2 Intensity |
|---|---|---|
| 1 | | |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | |

Всего несколько компонентов имеют значение!

=> Размерность сильно меньше

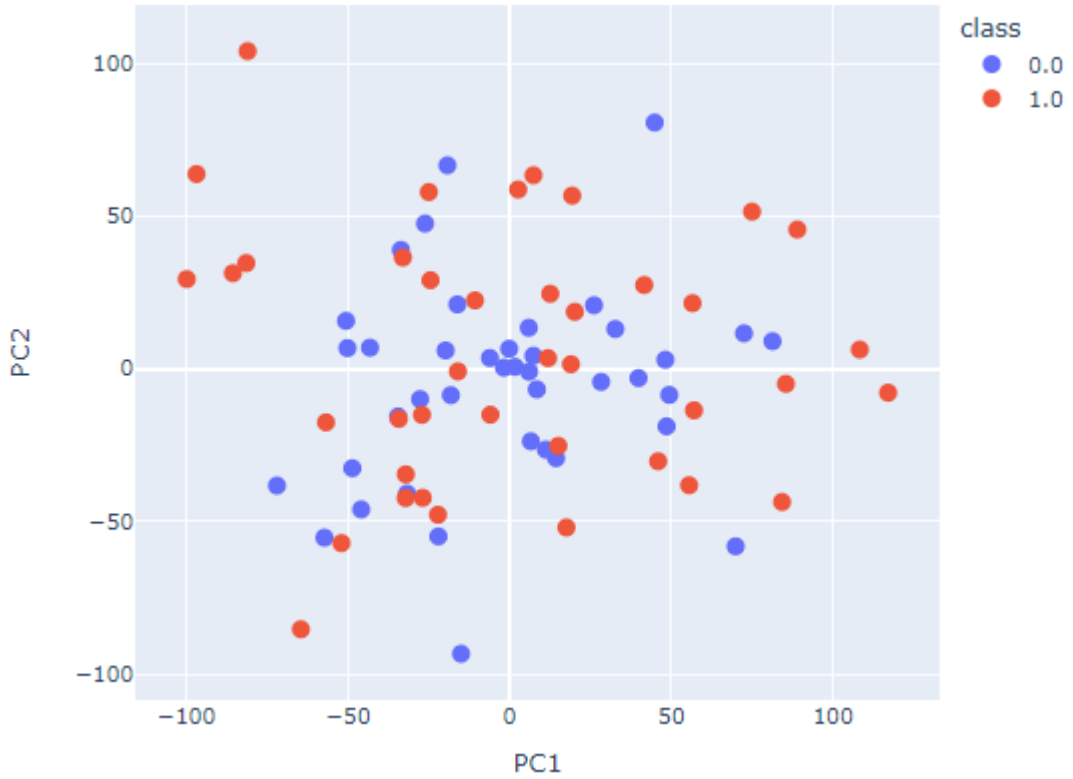


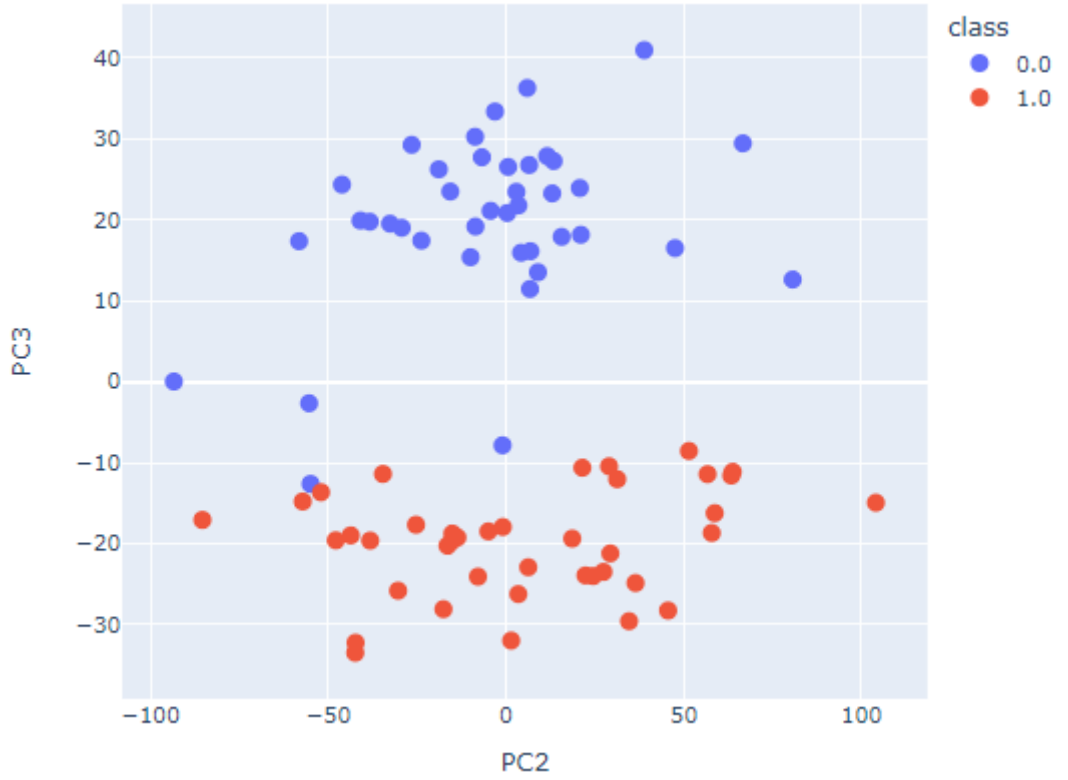

Source: ourcodingclub.github.io

# PCA – Principal Component Analysis

Предобработка:

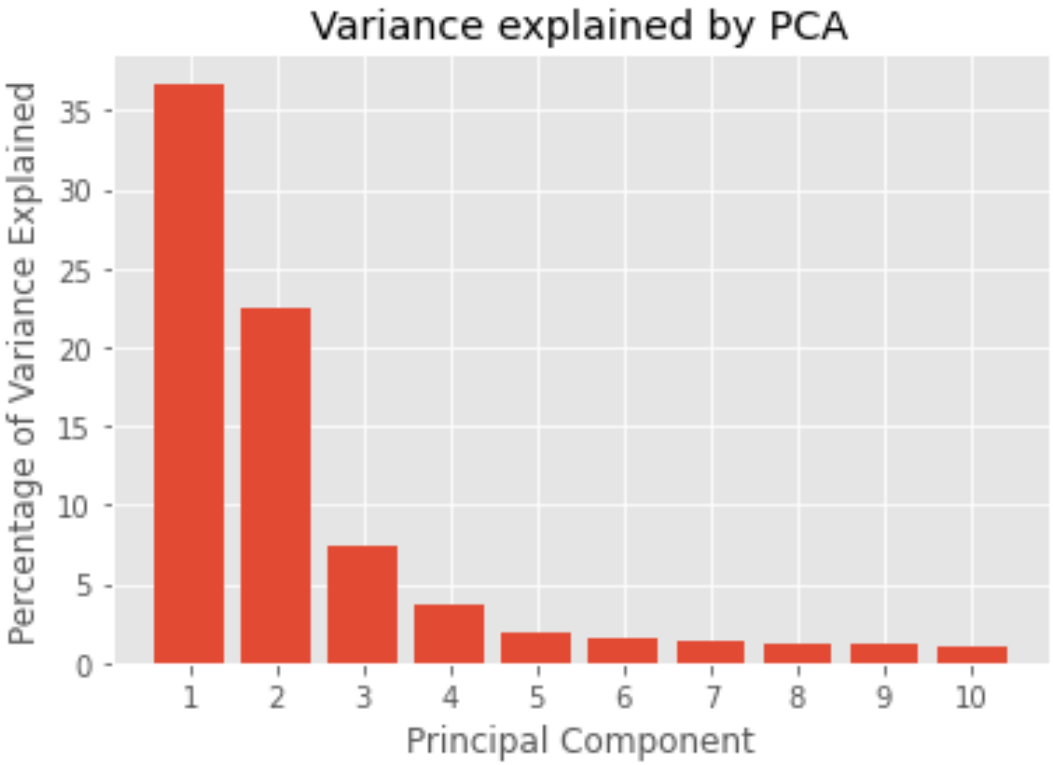
- Каждый спектр нормализовывался по площади под своей кривой
- Затем, каждый признак стандартизировался по всем спектрам

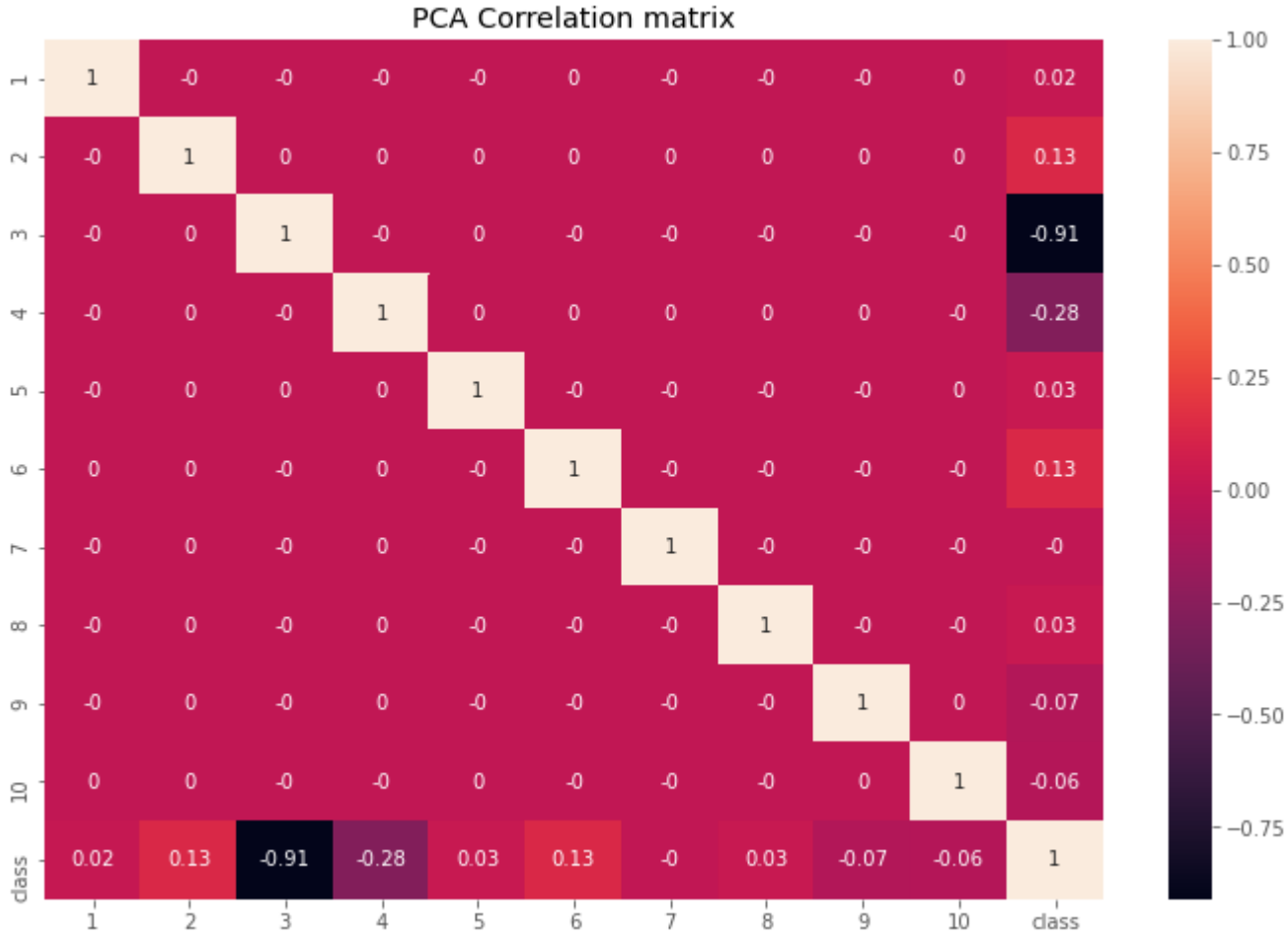



По PC3 можно наблюдать практически полное разделение классов

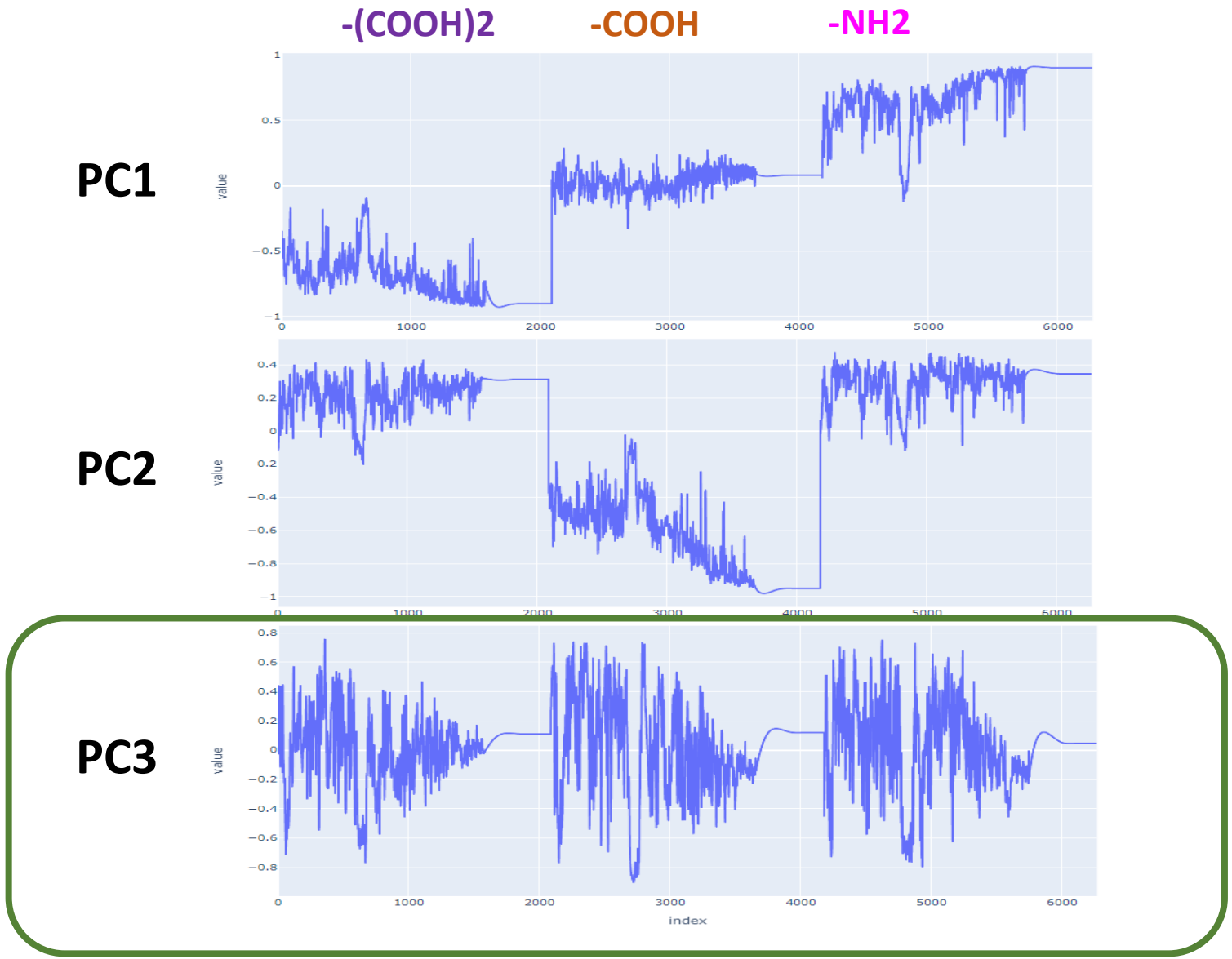
# PCA analytics




- все компоненты PCA ортогональны друг другу, их взаимные корреляции = 0
- PC3 сильно скоррелирован с меткой класса

# PCA loadings



# Содержание

- Введение в тему
- Постановка задачи
- Исследование данных
- Применение методов машинного обучения
  - Снижение размерности
  - **Классификация**
- Модификация задачи

# Logistic Regression

Train set

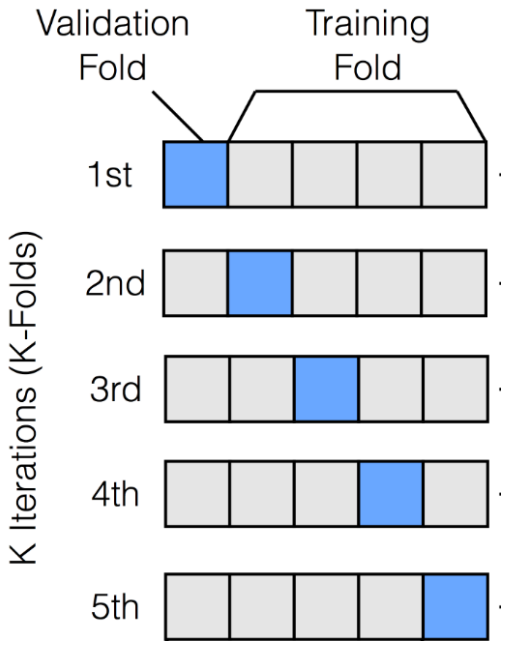
| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0.0 | 1.00 | 1.00 | 1.00 | 40 |
| 1.0 | 1.00 | 1.00 | 1.00 | 40 |
| accuracy | | | 1.00 | 80 |
| macro avg | 1.00 | 1.00 | 1.00 | 80 |
| weighted avg | 1.00 | 1.00 | 1.00 | 80 |

Test set

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0.0 | 1.00 | 1.00 | 1.00 | 10 |
| 1.0 | 1.00 | 1.00 | 1.00 | 10 |
| accuracy | | | 1.00 | 20 |
| macro avg | 1.00 | 1.00 | 1.00 | 20 |
| weighted avg | 1.00 | 1.00 | 1.00 | 20 |

100% Качество даже на тестовом сете

# Cross Validation

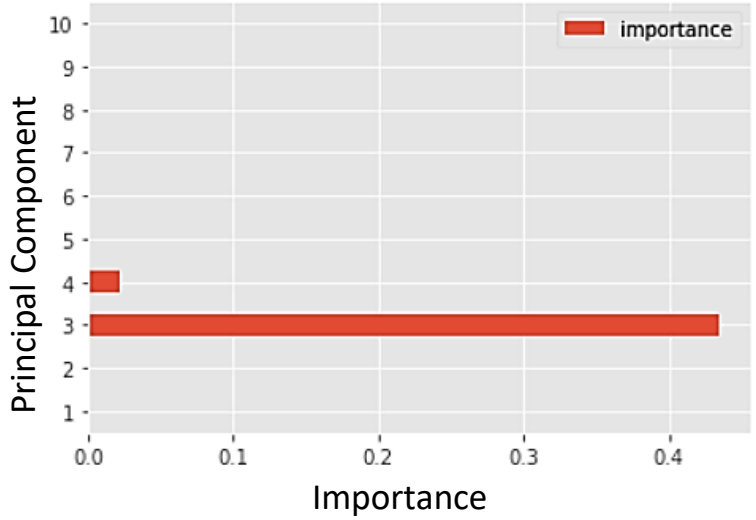


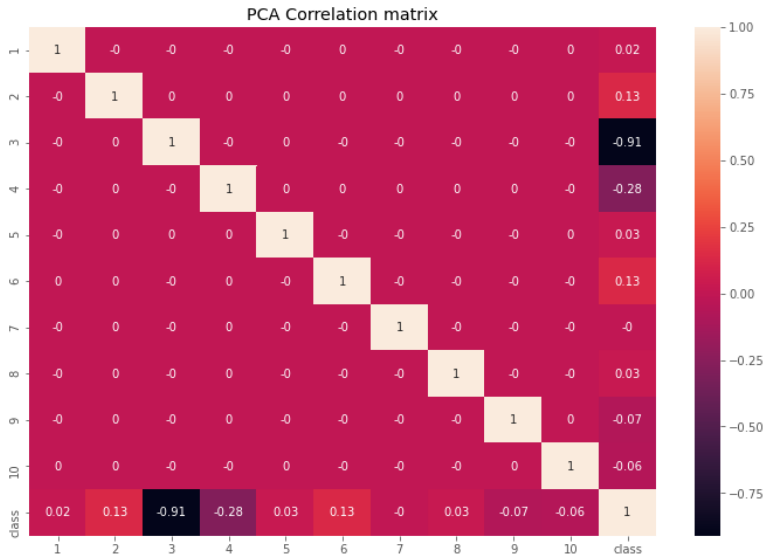

```
Train k-fold mean recall: 1.00
Valid k-fold mean recall: 1.00
```

```
Train k-fold mean rocauc: 1.00
Valid k-fold mean rocauc: 1.00
```

Все еще 100% качество

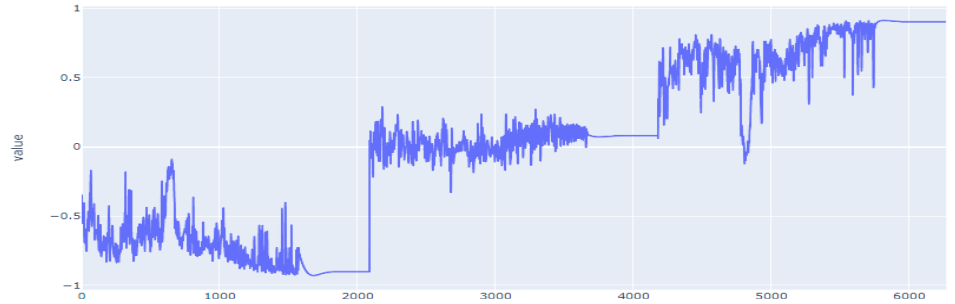
# Feature importances



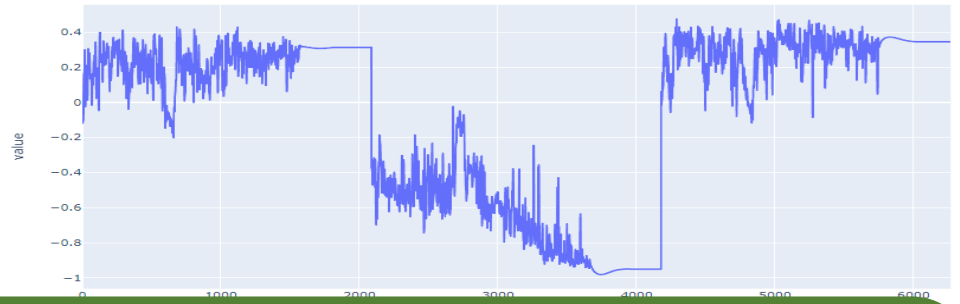




-(COOH)2 -COOH -NH2

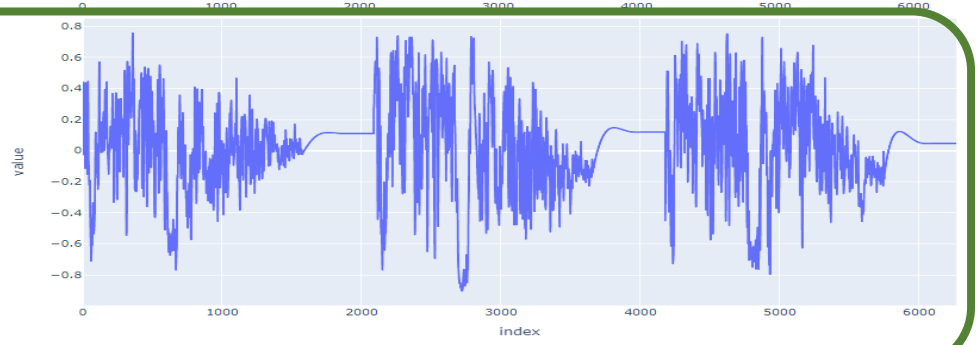
**PC1**




**PC2**




**PC3**

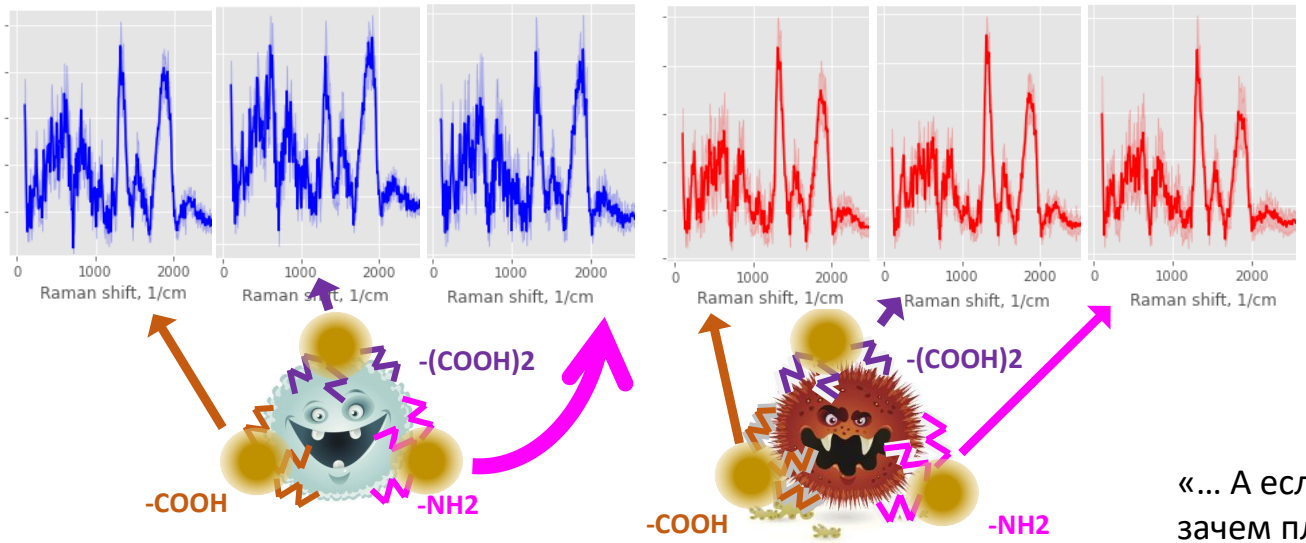


# Содержание

- Введение в тему
- Постановка задачи
- Исследование данных
- Применение методов машинного обучения
  - Снижение размерности
  - Классификация
- **Модификация задачи**

# Используем только 1 тип частиц

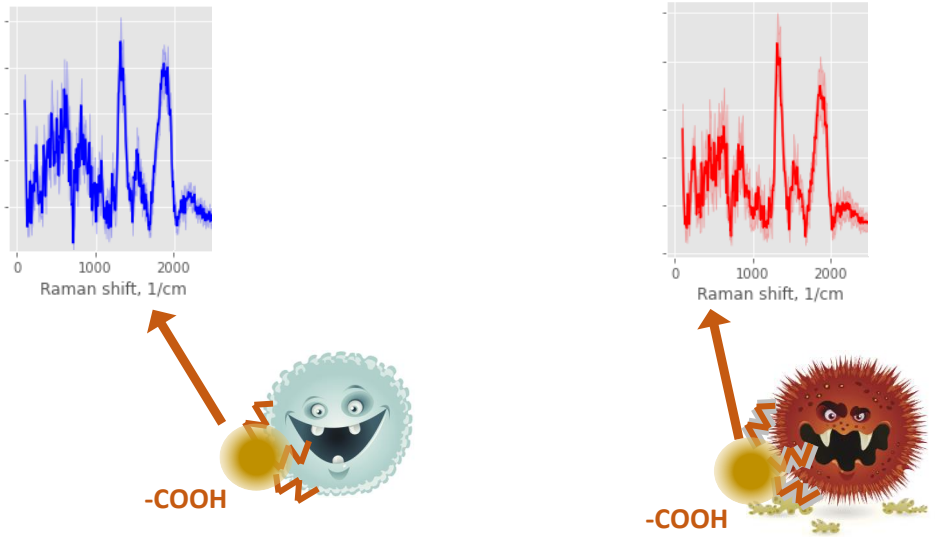
**Было:**




**6000**
признаков

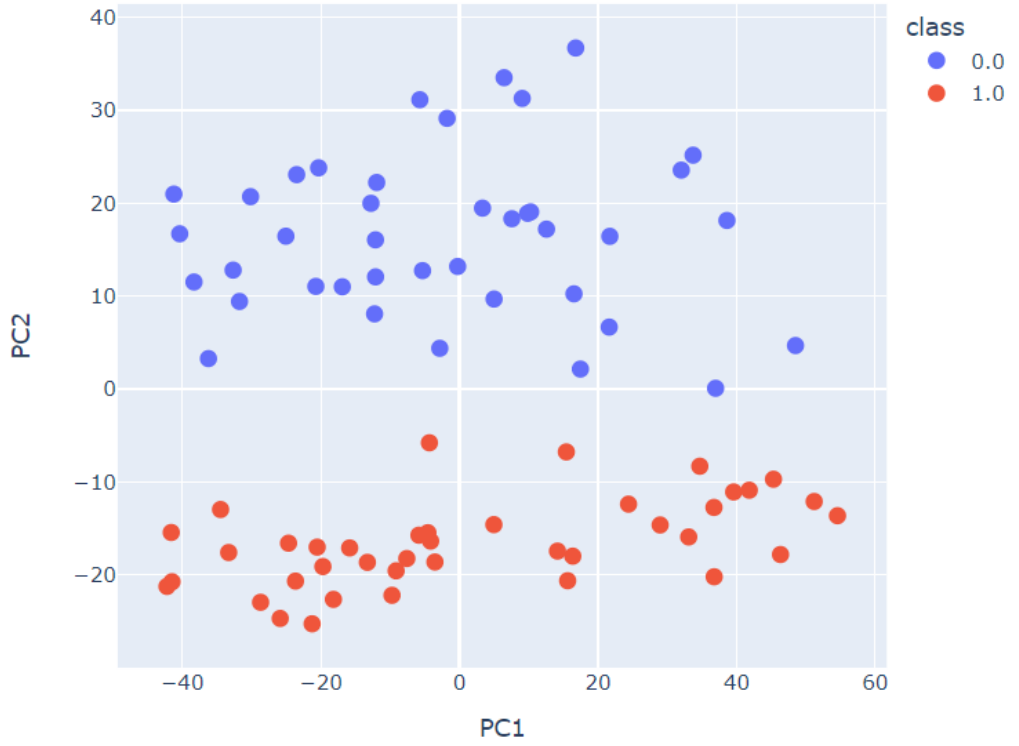
«… А если нет разницы, зачем платить больше?»

**Стало:**

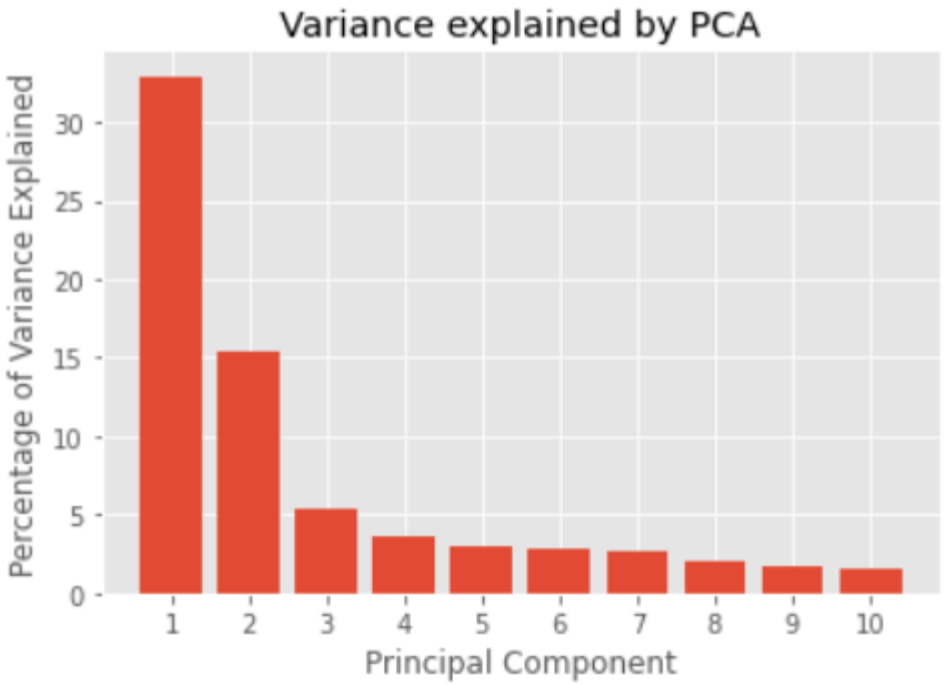



**2000**
признаков

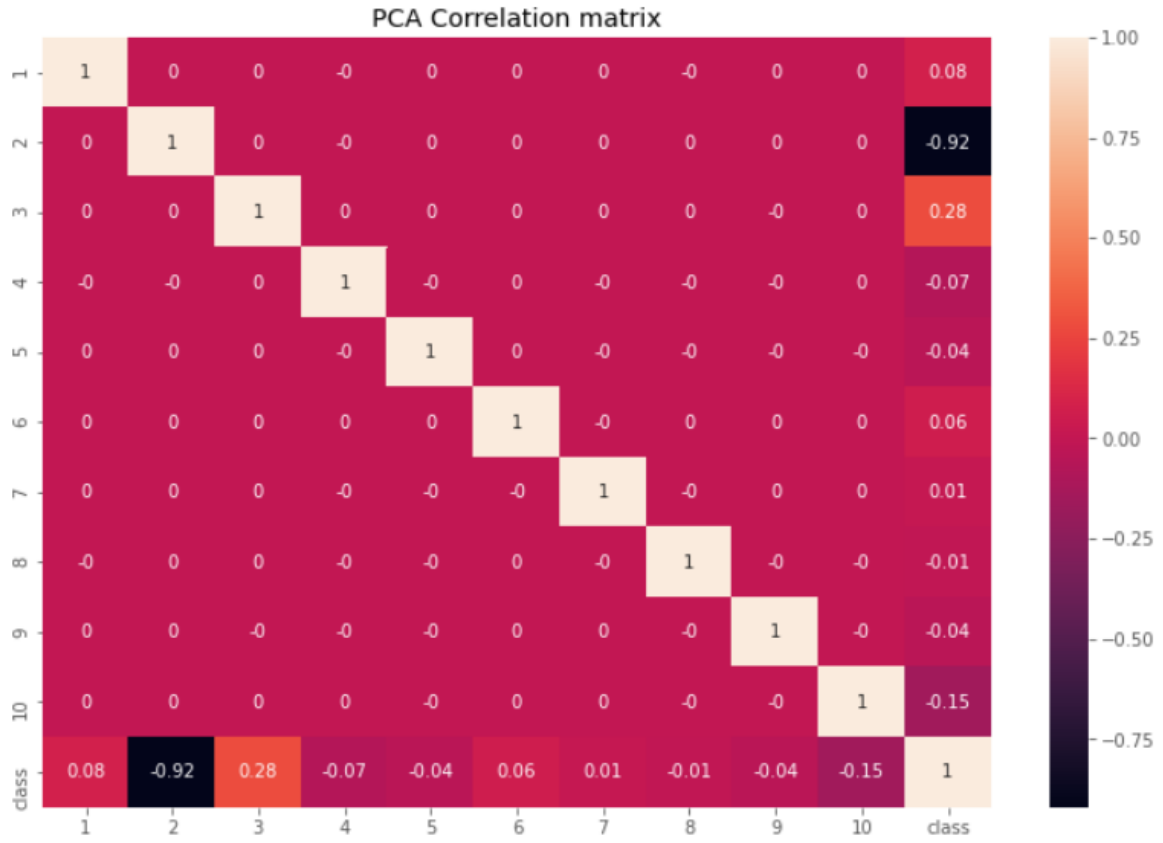
# PCA – Principal Component Analysis




# PCA analytics



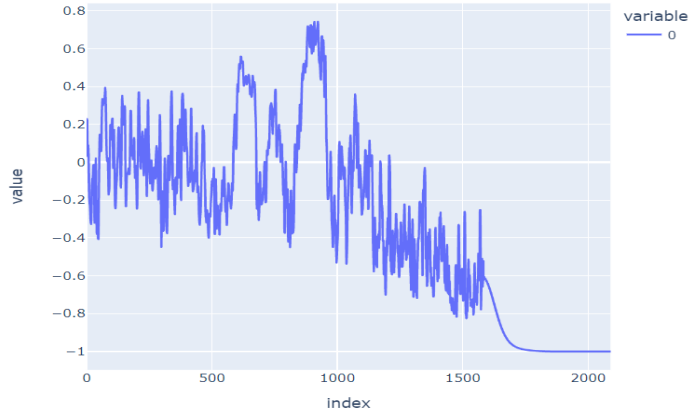

Матрица взаимных корреляций



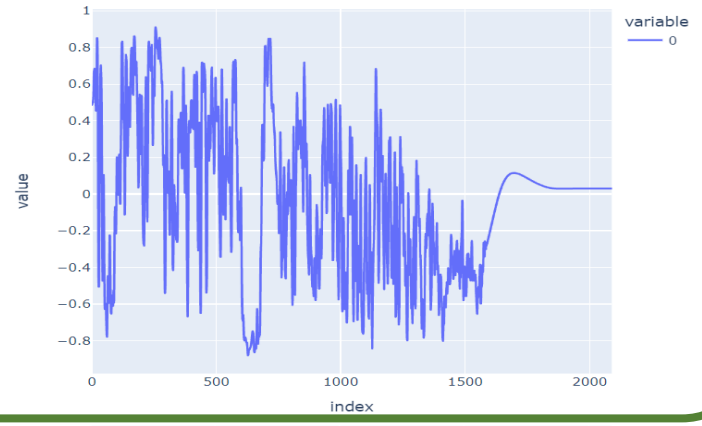

- все компоненты PCA ортогональны друг другу, их взаимные корреляции = 0
- PC2 сильно скоррелирован с меткой класса

# PCA loadings

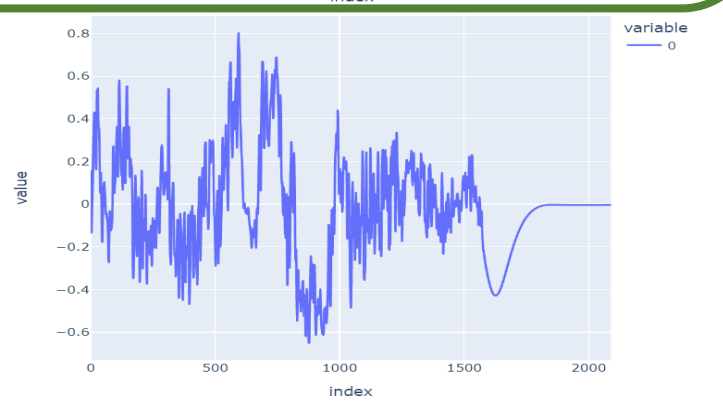
**PC1**




**PC2**




**PC3**



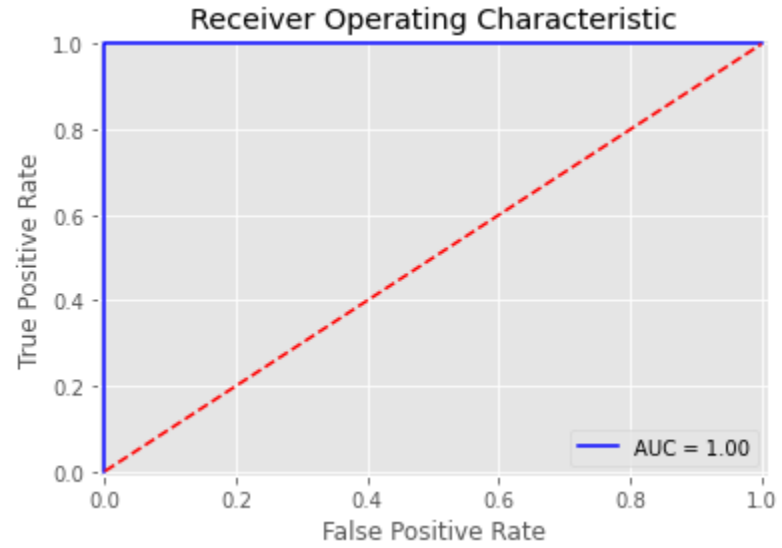
# Logistic Regression

Train set

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0.0 | 1.00 | 1.00 | 1.00 | 40 |
| 1.0 | 1.00 | 1.00 | 1.00 | 40 |
| accuracy | | | 1.00 | 80 |
| macro avg | 1.00 | 1.00 | 1.00 | 80 |
| weighted avg | 1.00 | 1.00 | 1.00 | 80 |

Test set

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0.0 | 1.00 | 1.00 | 1.00 | 10 |
| 1.0 | 1.00 | 1.00 | 1.00 | 10 |
| accuracy | | | 1.00 | 20 |
| macro avg | 1.00 | 1.00 | 1.00 | 20 |
| weighted avg | 1.00 | 1.00 | 1.00 | 20 |




100% Качество даже на тестовом сете 🤔🤔🤔

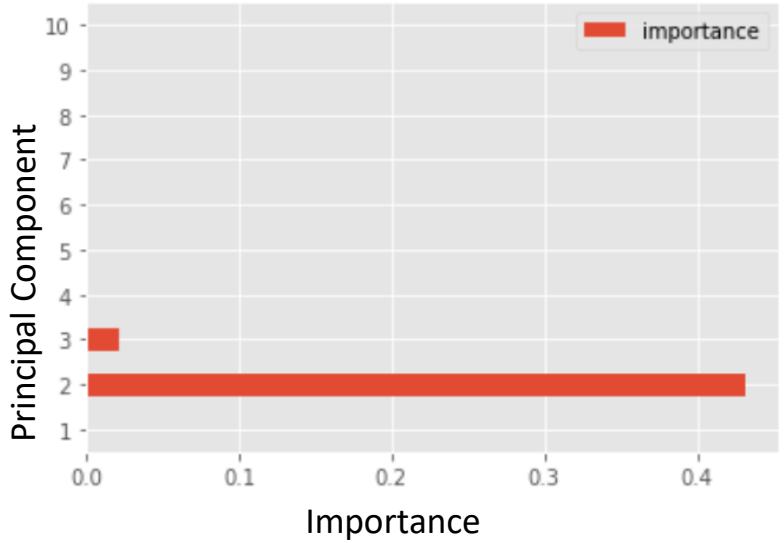
# Cross Validation

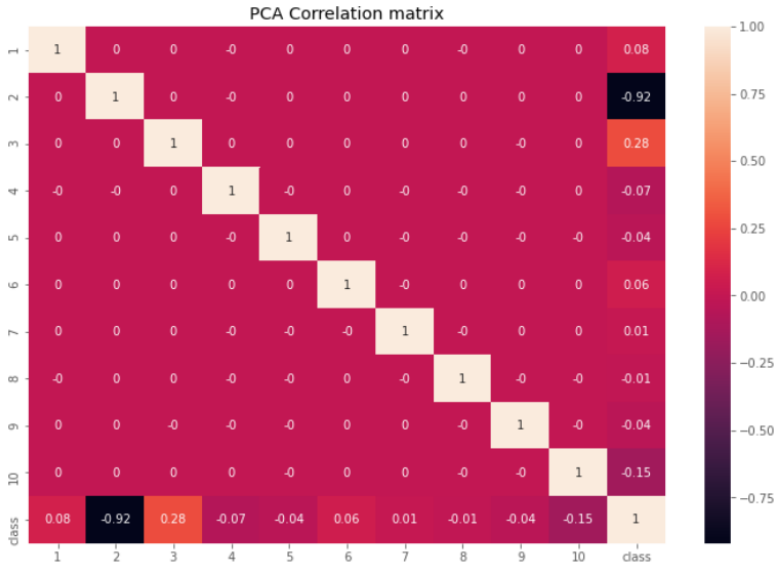
```
Train k-fold mean recall: 1.00
Valid k-fold mean recall: 1.00
```

```
Train k-fold mean rocauc: 1.00
Valid k-fold mean rocauc: 1.00
```

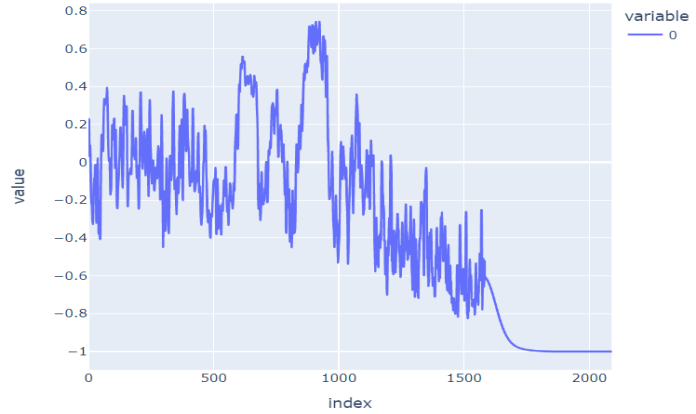
Все еще 100% качество

# Feature importances

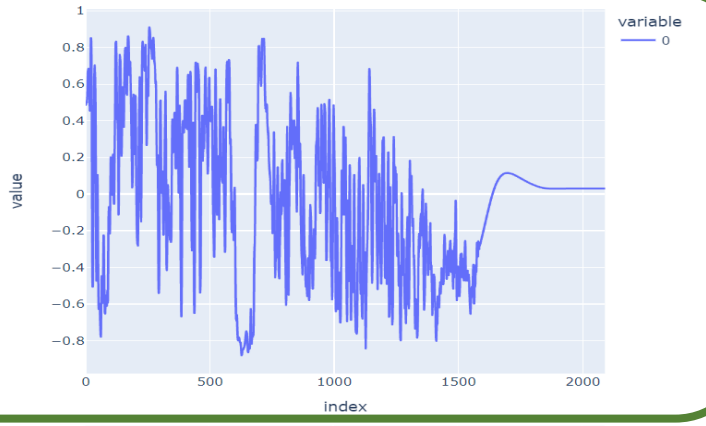


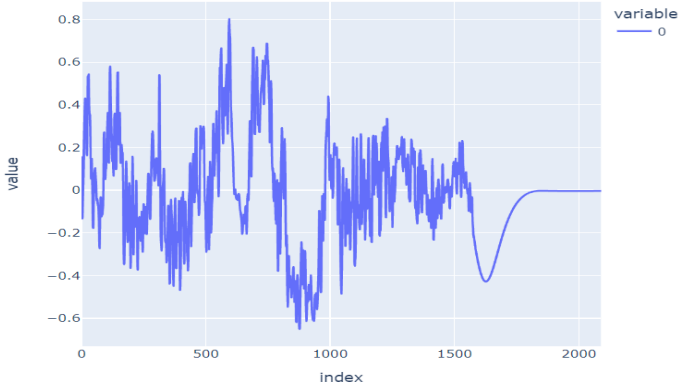

**PC1**




**PC2**




**PC3**



# Заключение

- Даже с использованием одного типа частиц можно достаточно просто реализовать классификацию клеток.
- Чтобы проверить «подозрительную идеальность» результатов:
  - Данные были изучены на предмет наличия утечек – не выявлено
  - Проведена кросс-валидация – качество все так же 100%

- ToDo:
  - Больше измерений

# Вопросы?