

Обнаружение выбросов и новизны **Anomaly Detection**

Александр Дьяконов

06 апреля 2021 года

План

Обнаружение выбросов и новизны

Терминология: Outlier Detection, Novelty Detection

Приложения

Особенности

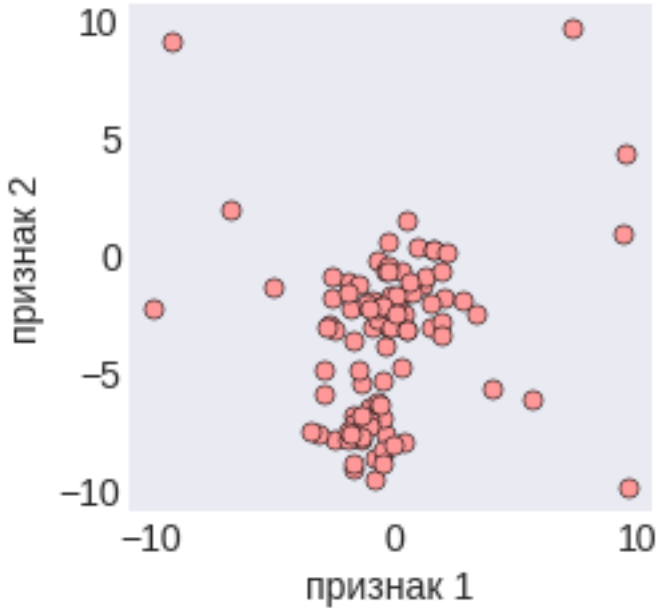
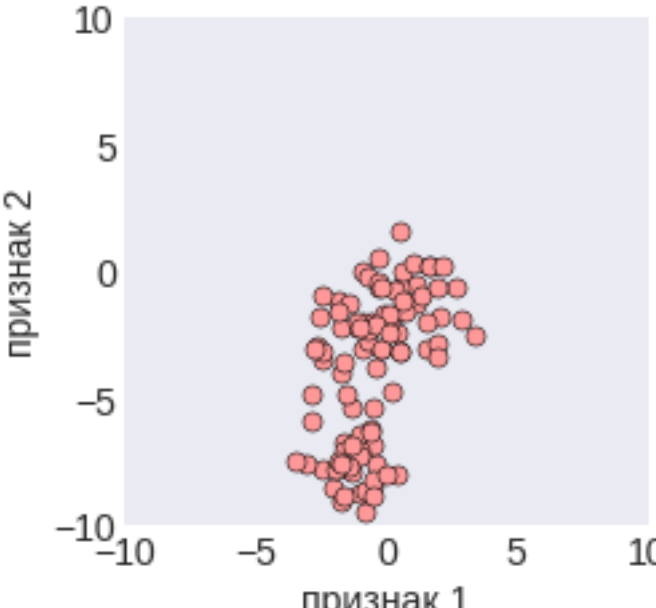
Примеры выбросов

Методы обнаружения

- Статистические тесты
 - Модельные тесты
- Итерационные методы
- Метрические методы (Proximity-Based Models), Local Outlier (LOF), Local cluster approach
 - Методы подмены задачи
- Методы машинного обучения (OneClassSVM, Изолирующий лес, EllipticEnvelope)

История из практики

Терминология

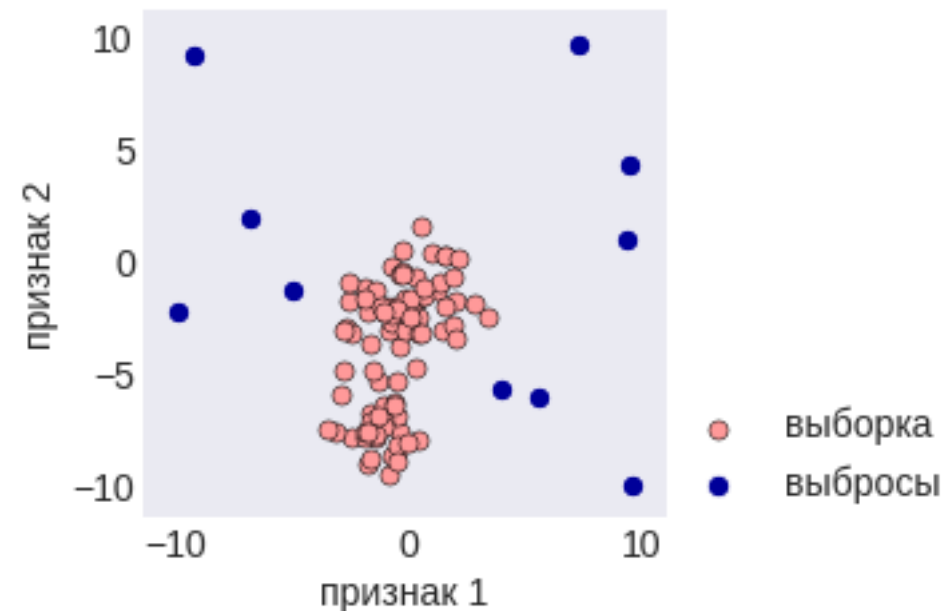
Outlier Detection – выбросы	Novelty Detection – новизна
объекты, которые по своим статистическим свойствам отличаются от объектов выборки	
Из другого распределения	этого же распределения может уже не быть... изменился параметр распределения
В обучении есть выбросы – их ищем	В обучении нет выбросов, но они скоро появятся
	

Терминология

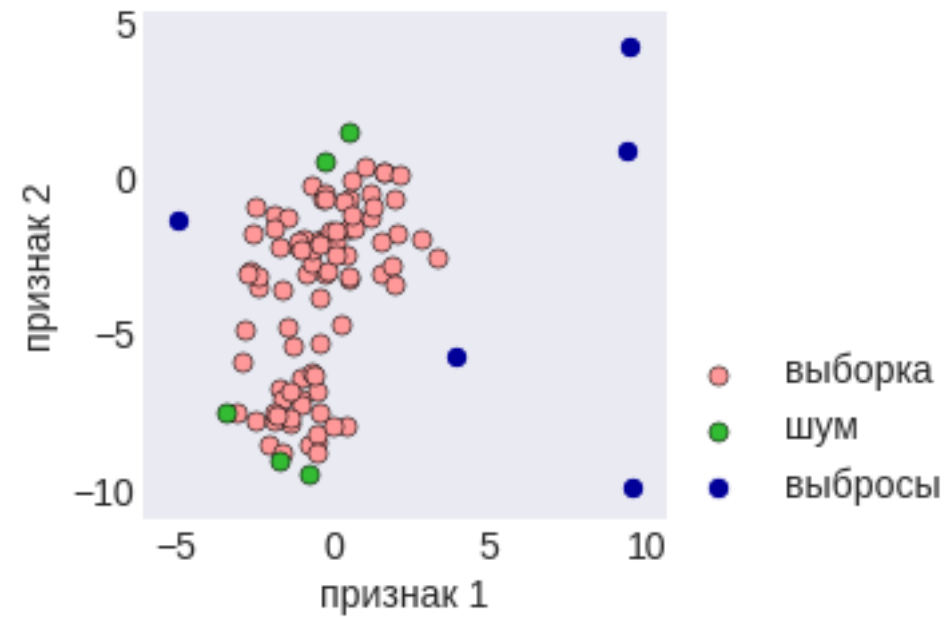
Резюме: выбросы в ней есть, новизна – скоро появится

Причины выбросов

- ошибки в данных
- шумовые объекты
- точки других выборок



Терминология



**шум – выбросы в слабом смысле,
аномалии – выбросы в сильном смысле**

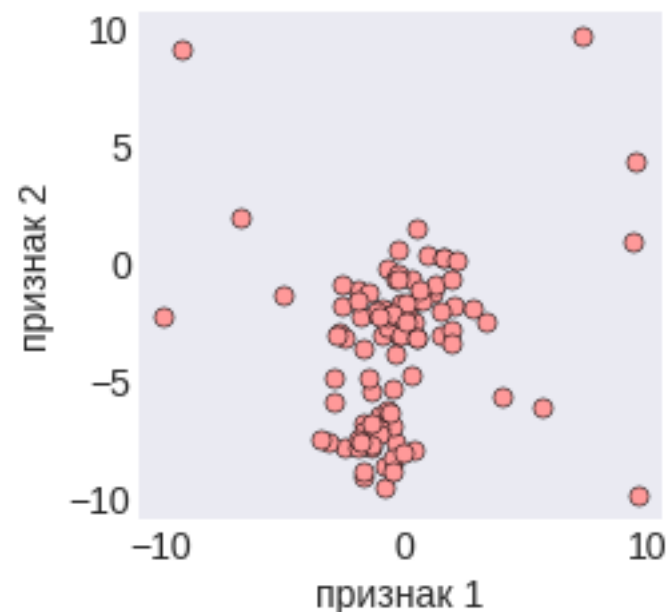
Приложения

**Это не только служебная задача для подготовки данных,
но и самостоятельная**

- Обнаружение подозрительных банковских операций (**credit-card fraud**)
- Обнаружение вторжений (**intrusion detection**)
- Обнаружение нестандартных игроков на бирже (инсайдеров)
- Обнаружение неполадок в механизмах по показаниям датчиков
- Медицинская диагностика (**Medical diagnosis**)
- Сейсмологий

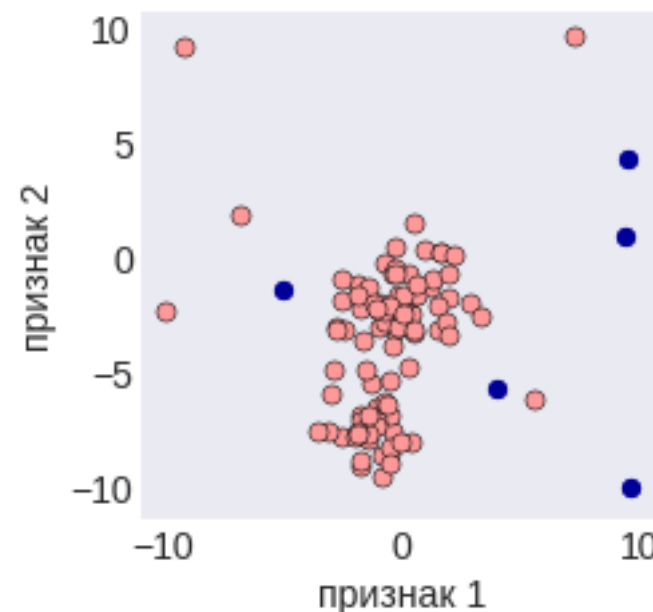
2.5 вида постановки задачи обнаружения аномалий

**есть выборка без
разметки**



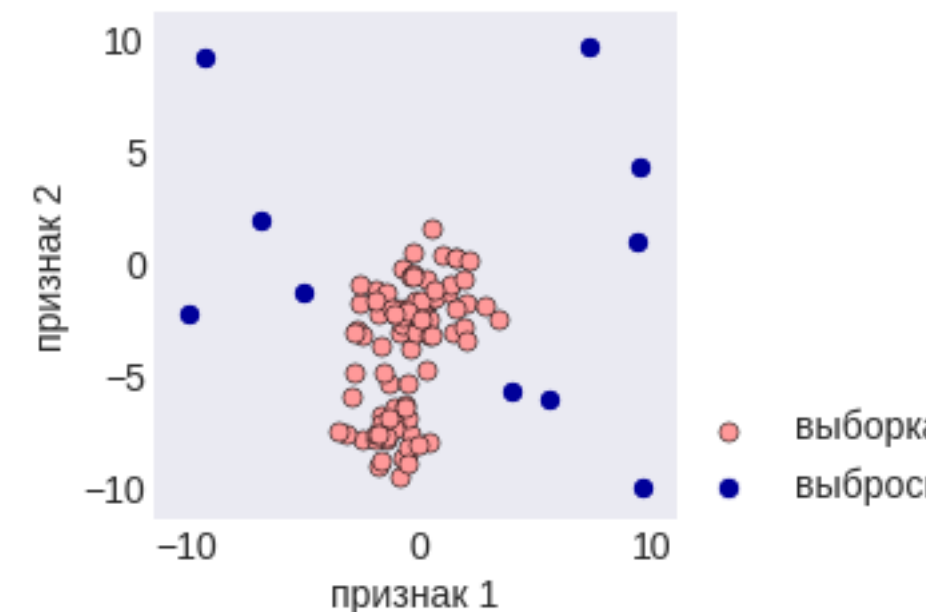
самая частая ситуация

есть частичная разметка класса 1



**при детектировании поломок
PUC – Positive-Unlabeled Class-n
часть выбросов обозначена 1,
остальные объекты обучения – 0 –
нормальные и м.б. немного
выбросов**

полная разметка



**такого практически не
бывает**

Детектирование выбросов

- **но всегда есть дисбаланс классов**
например, поломки оборудования относительно редки

Типы задач

Классификация: выброс / не выброс

Скоринг: степень (не) нормальности объекта

Semi-supervised

Разметка есть, но это далеко не все типы выбросов

Пример: работа вирусов, хакерские атаки

Active learning

есть возможность получать разметку конкретных объектов

Нужно ли удалять выбросы?

Практика: часто не надо об этом задумываться

- Многие статистики устойчивы к выбросам
какие?
- Многие методы достаточно робастны
какие?
- В тестовой выборке тоже будут выбросы...

Более того, выбросы могут быть частью данных

(каждая 370я точка из нормального распределения выходит из отрезка трёх сигм)

Примеры: выбросы в матрицах

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
x_0	0	5	6	4	6	1	10	6	5	1
x_1	5	7	10	6	7	5	9	9	2	5
x_2	2	4	5	5	4	4	8	5	9	4
x_3	3	5	5	4	8	3	7	5	9	1
x_4	0	6	2	2	1	3	3	7	0	2
x_5	1	8	14	12	7	6	12	6	11	7

что это за матрицы?

где выбросы в этой матрице?

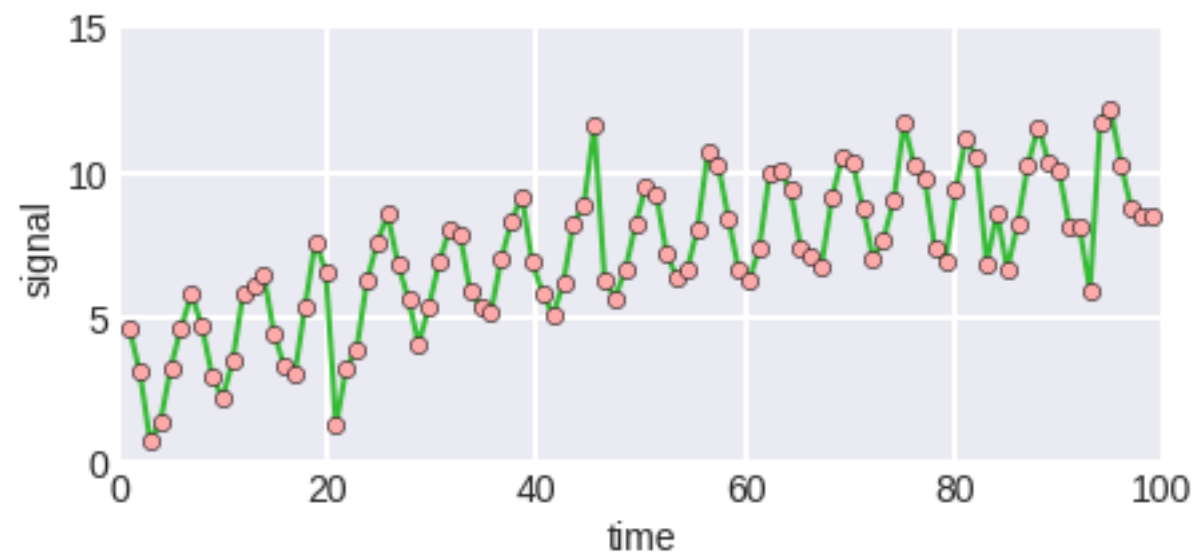
Примеры: выбросы в матрицах

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
x_0	0	5	6	4	6	1	10	6	5	1
x_1	5	7	10	6	7	5	9	9	2	5
x_2	2	4	5	5	4	4	8	5	9	4
x_3	3	5	5	4	8	3	7	5	9	1
x_4	0	6	2	2	1	3	3	7	0	2
x_5	1	8	14	12	7	6	12	6	11	7

что это за матрицы?

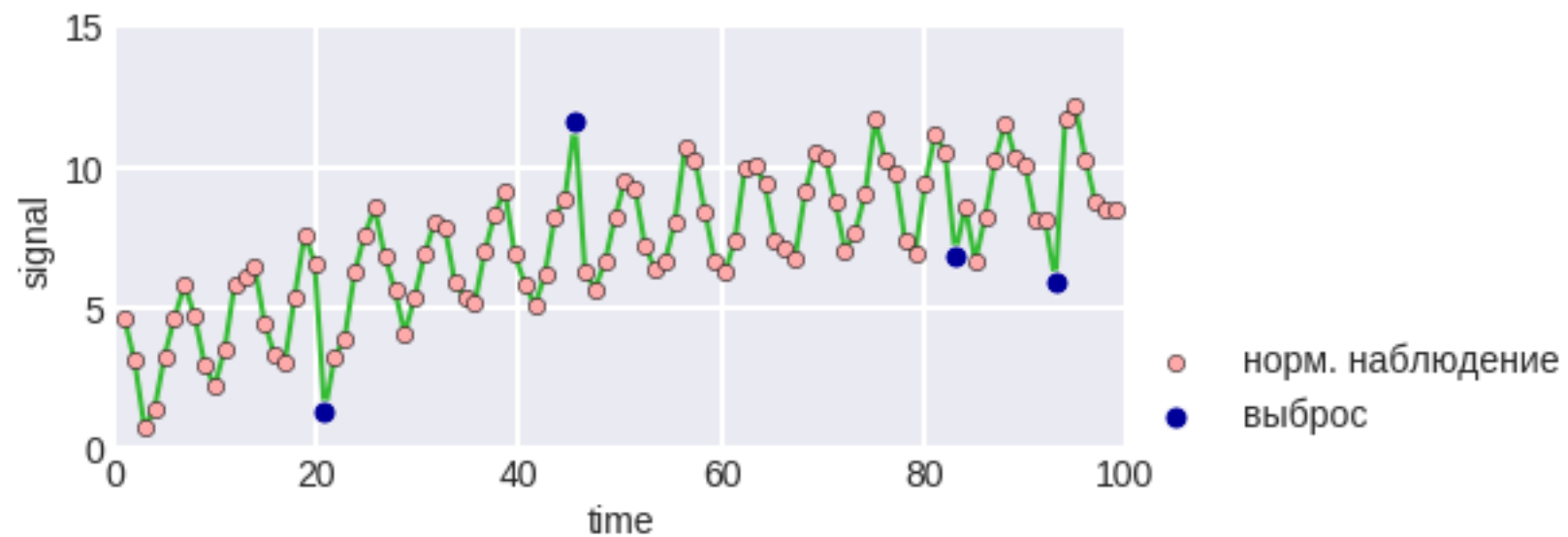
- матрица «user-item»
- матрица «документ-слово»
- матрица «время-показатель»
- матрица «объект-признак»

Примеры: выбросы в сигналах



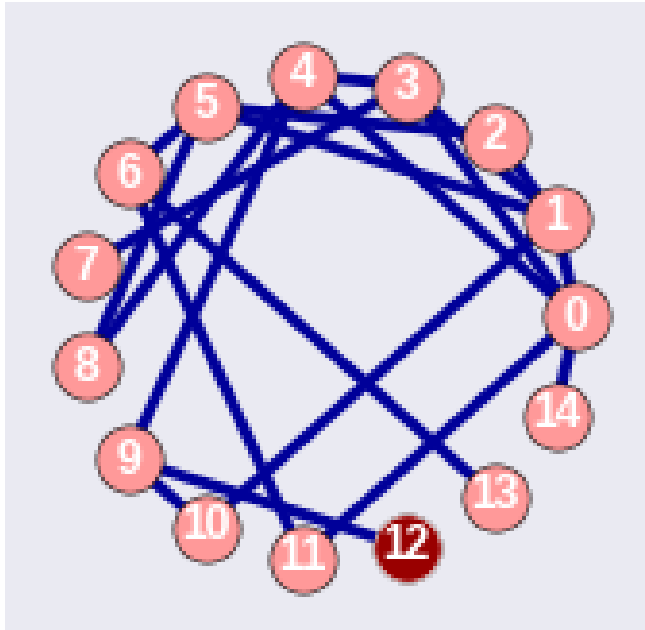
где здесь выбросы?

Примеры: выбросы в сигналах



выбросы не всегда «видно на глаз»!

Примеры: выбросы в графах



Могут быть выбросы-вершины,
выбросы-рёбра

Как определить?

Примеры: выбросы в последовательностях / в текстах

АААВВССААВВВСАААВВСАВВССАВААВВССАААВВВВСАВВВСС

Функционалы качества

Здесь совсем сильно зависят от заказчика

Часто стандартные:

PR AUC

AUROC

Методы обнаружения

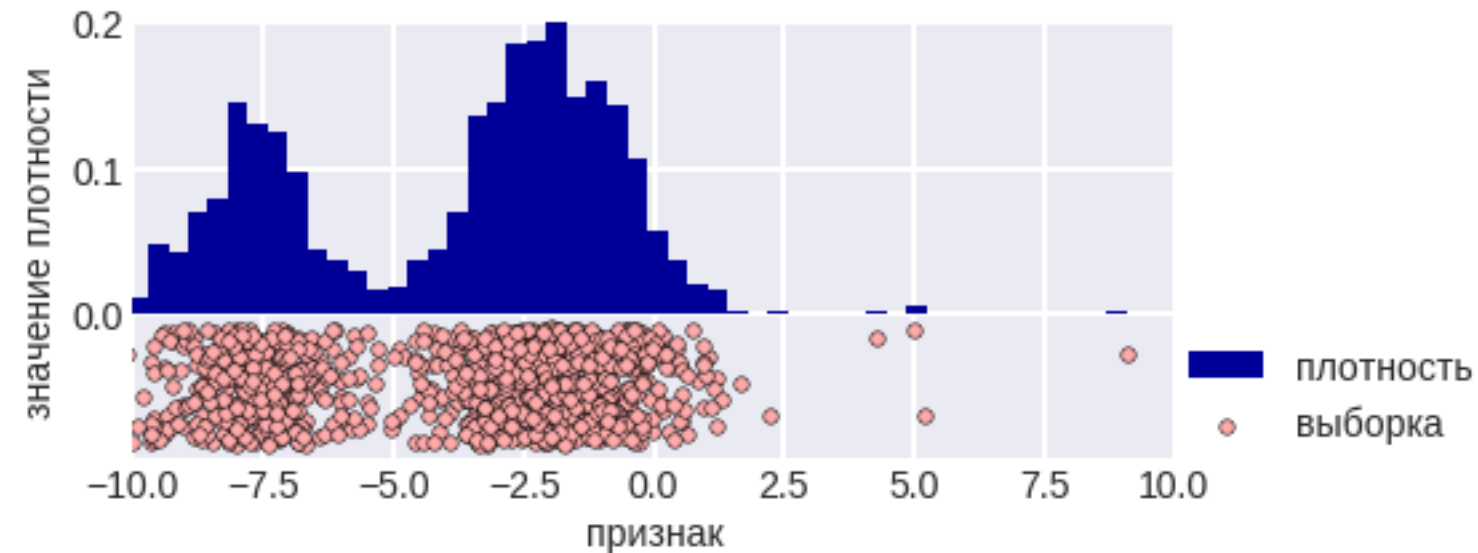
Статистические тесты

Пусть точки из некоторого распределения. Вычисляем вероятность, что точка соответствует распределению.

Модельные тесты

**Пусть точки описываются некоторой моделью (например линейной регрессией).
Вычисляем отклонение от модели (ошибку).**

Методы обнаружения: статистические тесты



Стандартные статистические тесты: отклонение от среднего

$$\text{Z-value } Z_i = \frac{|x_i - \mu|}{\sigma}$$

в предположении нормальности данных...

Методы обнаружения: статистические тесты

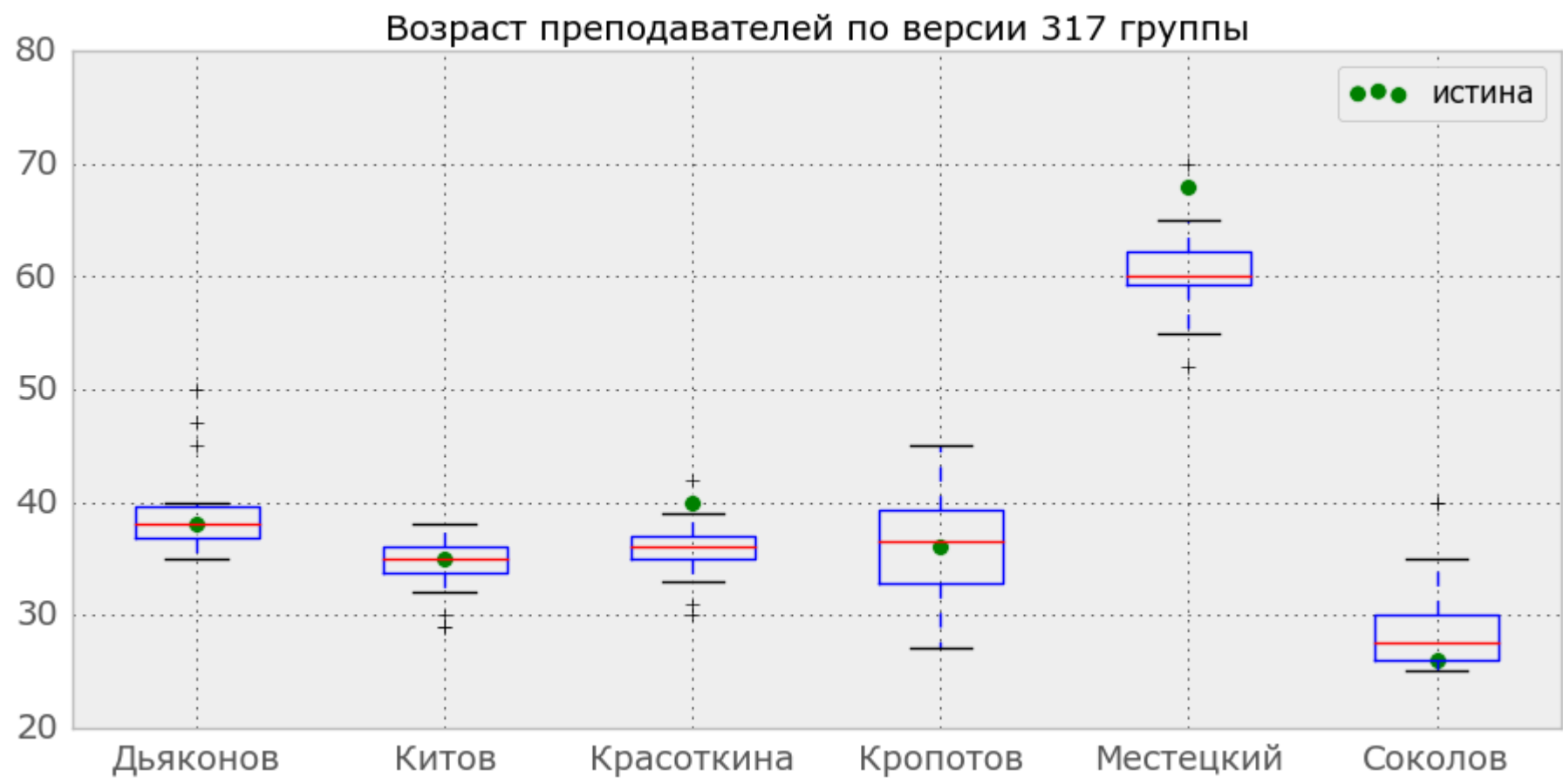
Kurtosis measure

$$\frac{1}{n} \sum_{i=1}^n Z_i^4$$

Если большие значения – есть выброс

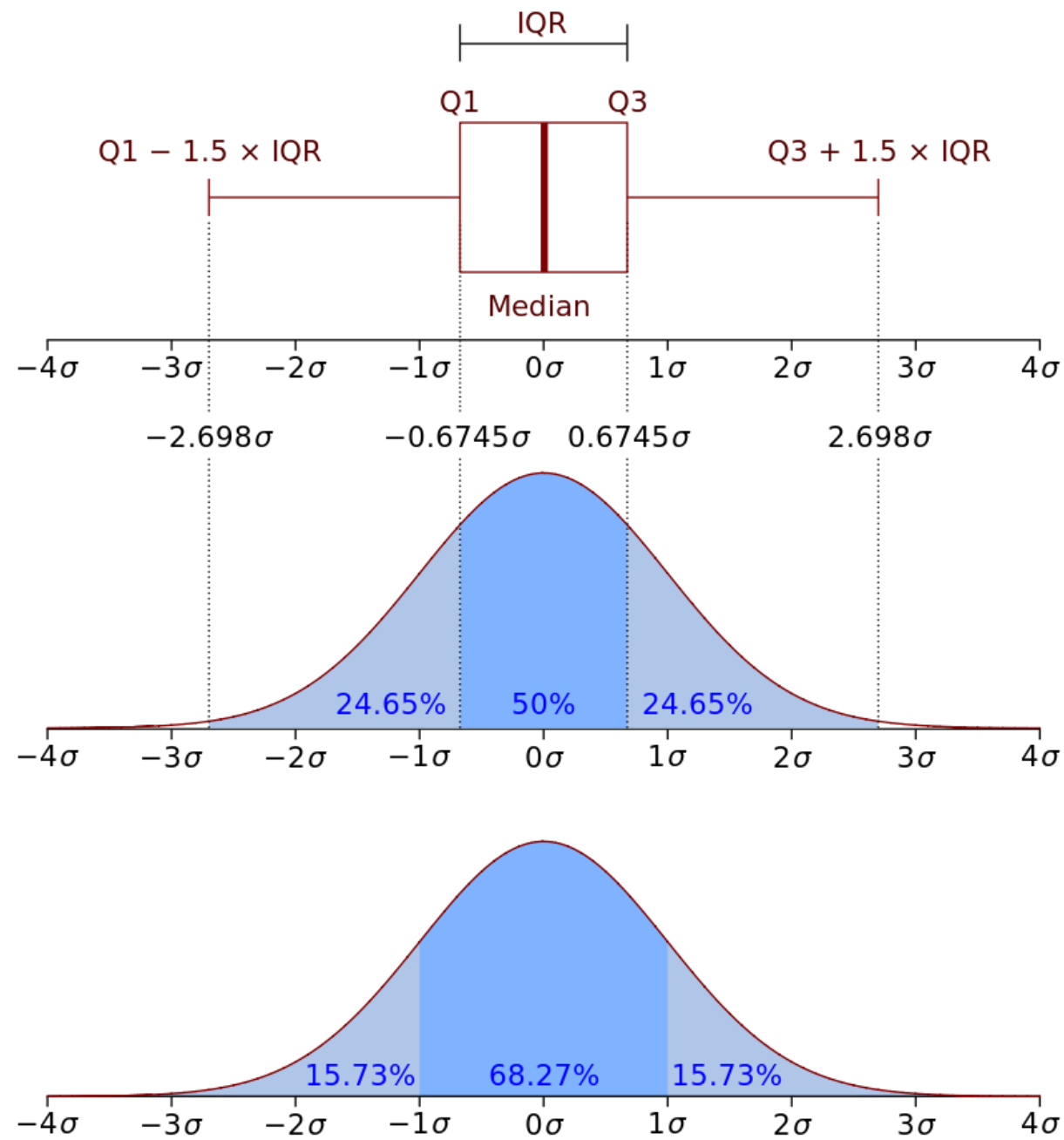
Кстати, $EZ_i^2 = 1$

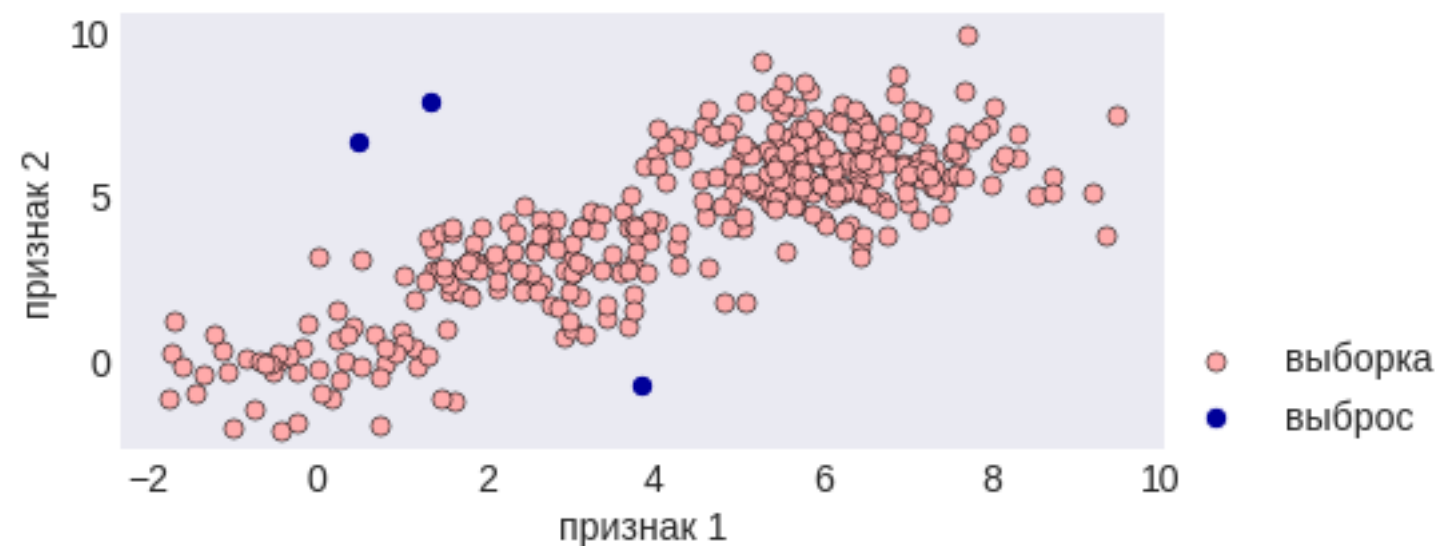
Визуализация: ящик с усами



Часто используют для визуализации в том числе выбросов

Визуализация: ящик с усами



Важно

**Почти всегда на практике
аномалия – не выброс по какому-то признаку!**

Важно

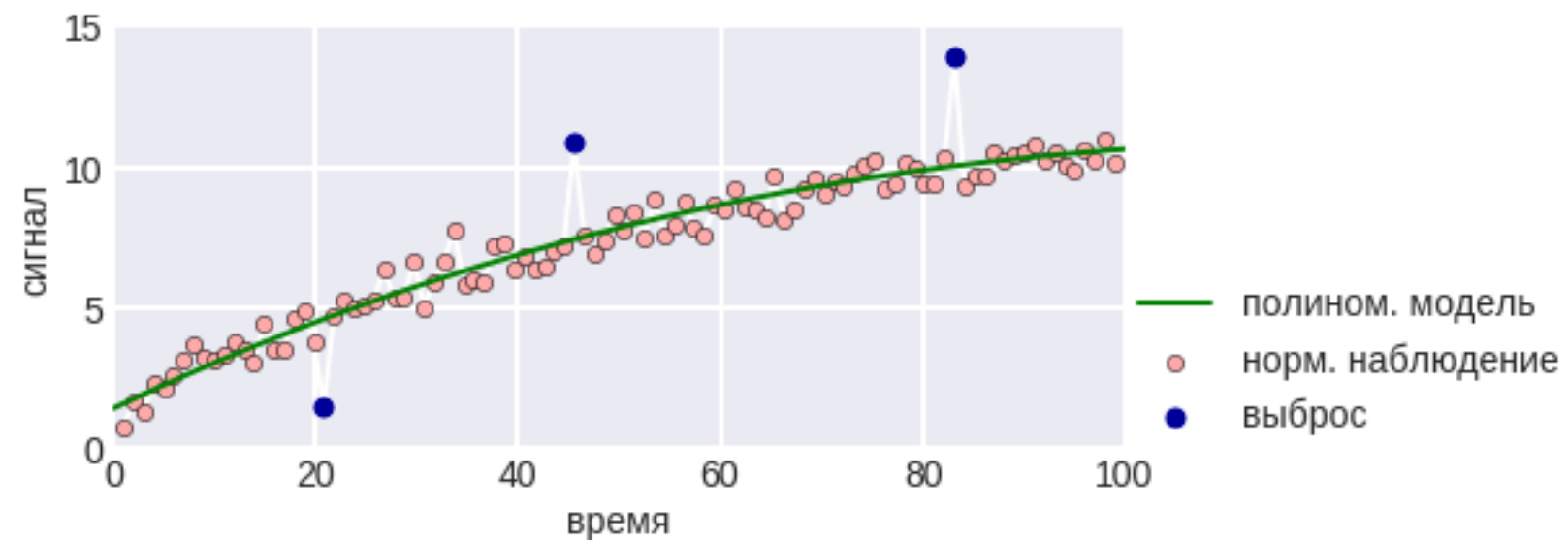
Экстремальные значения и выбросы – разные понятия!

1, 3, 100, 1, 101, 2, 102, 2, 2, 100, 1, 101, **50**, 1, 3, 2, 101, 1, 102, 101, 3, 102, ...

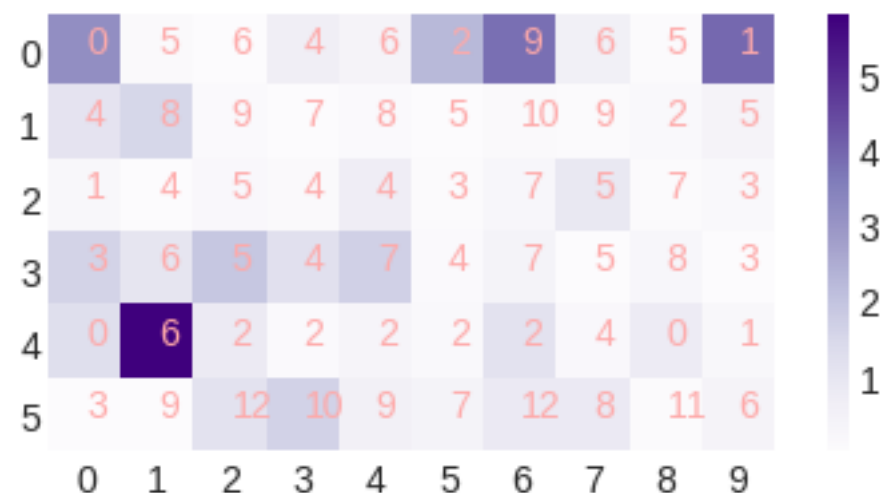
Extreme-Value Analysis

Хотя... последний этап большинства алгоритмов детектирования аномалий – нахождение экстремальных значений

Методы обнаружения: модельные тесты



Методы обнаружения: модельные тесты



```
from scipy.sparse.linalg import svds
```

```
U,L,V = svds(H.astype(float), k=2)
```

```
X = U @ np.diag(L) @ V
```

```
plt.imshow((H - X) ** 2, cmap='Purples')
```

$$X \approx U_{m \times k} L_{k \times k} V_{k \times n}$$

приближаем матрицей малого ранга

Методы обнаружения: модельные тесты

Проблема

**На построение модели/распределения влияют выбросы,
которые мы ищем**

Выход – новая группа методов:

Выбросы – точки при удалении которых модель можно лучше подстроить под данные.

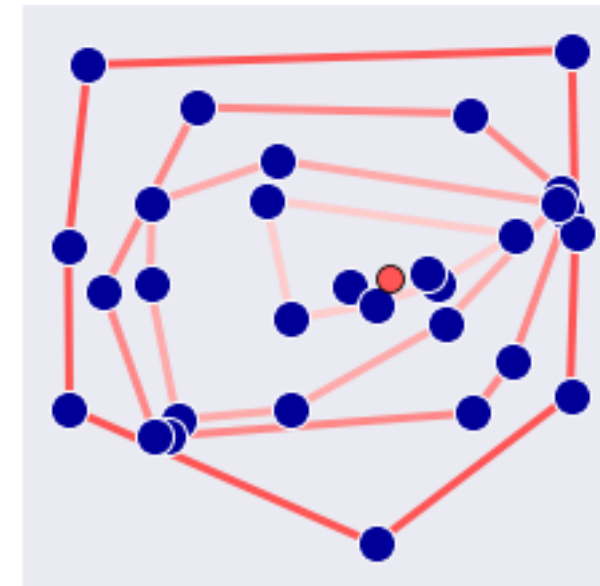
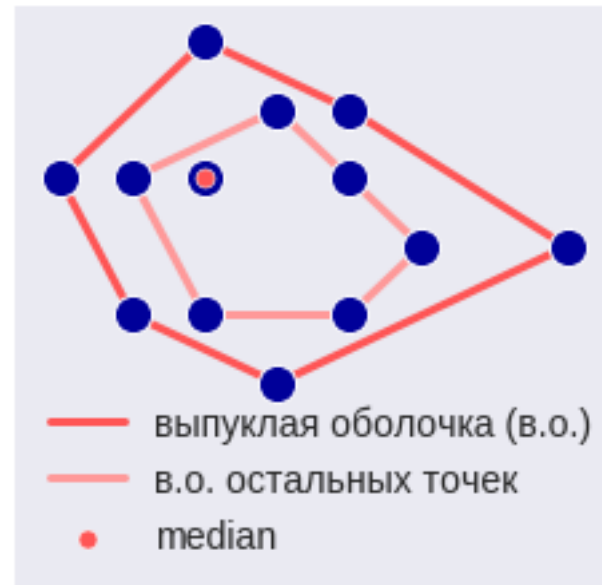
– самые естественные!

Что такое «аномалия»

определяется нашим знаниями о природе данных

Их можно заложить в модель!

Методы обнаружения: итерационные методы



Чтобы получить медиану надо последовательно отбрасывать крайние значения (наверняка выбросы)

Аналогичная процедура в многомерном пространстве...

Итерационные методы

Ещё идеи:

- **последовательно исключать объекты максимально понижая разброс**

+ обобщение одномерного случая

+ можно получать оценку новизны

+ нет предположений о распределении

- вычислительная трудоёмкость

Метрические методы (Proximity-Based Models)

У выбросов мало соседей

У обычных точек много соседей

Пример: расстояние до k-го соседа ~ мера нетипичности

Часто: специфические метрики (Махалобиса)

Почему-то самые популярные...

Local Outlier (LOF)

Пусть $\rho_k(z)$ – расстояние до k -го БС

будем использовать расстояние

$$\rho'(x, z) = \max[\rho(x, z), \rho_k(z)]$$

$$r_k(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} \rho'(x, x_i)$$

обратная величина к этой называется local reachability density (LRD)

оценка «выбросовости» (relative distance score):

$$\frac{1}{k} \sum_{x_i \in N_k(x)} \frac{r_k(x)}{r_k(x_i)}$$

Local cluster approach

Делаем кластеризацию

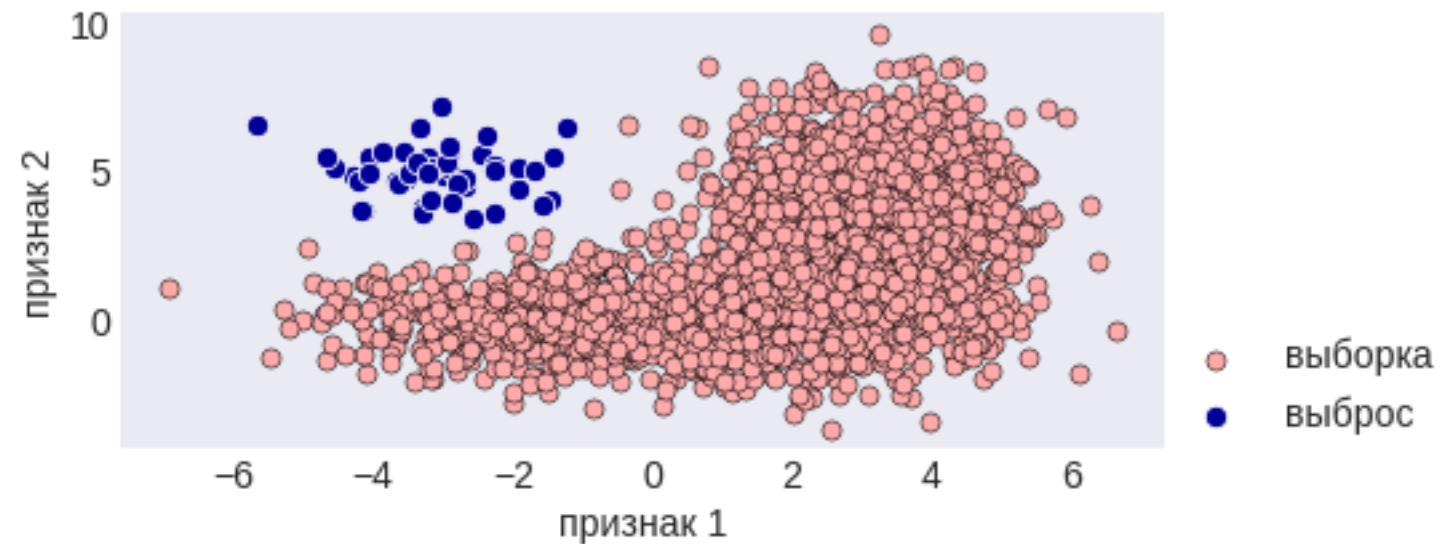
Оценка «выбросовости» ~ расстояние Махалобиса до ближайшего кластера

$$\sqrt{(x - \mu)^T \Sigma^{-1} (x - \mu)}$$

матрица ковариаций оценивается по соответствующему кластеру

Методы подмены задачи

**делаем кластеризацию
маленькие кластеры – выбросы**



Методы машинного обучения

- `sklearn.svm.OneClassSVM`
- `sklearn.ensemble.IsolationForest`
- `sklearn.covariance.EllipticEnvelope`

`sklearn.svm.OneClassSVM`

**Задача классификации с одним классом (поиска новизны, а не выбросов,
т.к. подстраивается под выборку)**

`kernel` – ядро (`linear` – линейное,
`poly` – полиномиальное,
`rbf` – радиальные базисные функции,
`sigmoid` – сигмоидальное,
своё заданное)

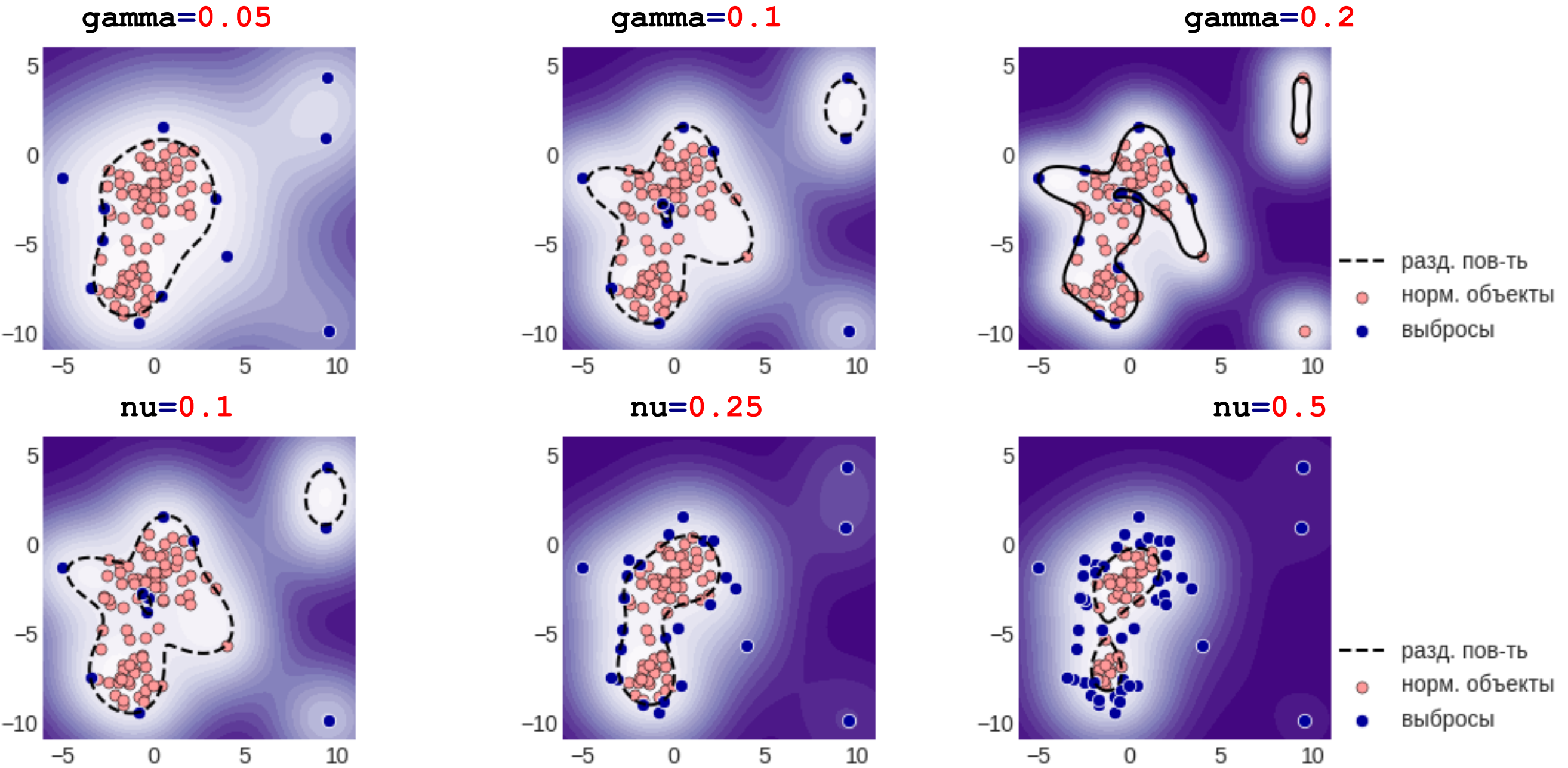
`nu` – верхняя граница на % ошибок и нижняя на % опорных векторов
(0.5 по умолчанию)

`degree` – степень для полиномиального ядра

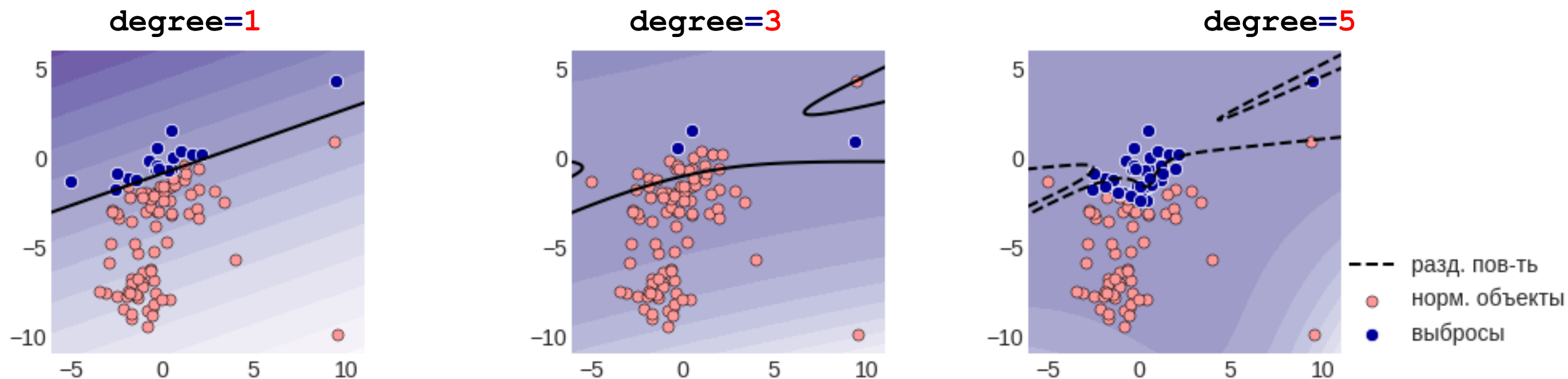
`gamma` – коэффициент для функции ядра (`1/n_features` по умолчанию)

`coef0` – параметр в функции полиномиального или сигмоидального ядра

OneClassSVM: `svm.OneClassSVM (nu=0.05, kernel="rbf", gamma=0.1)`



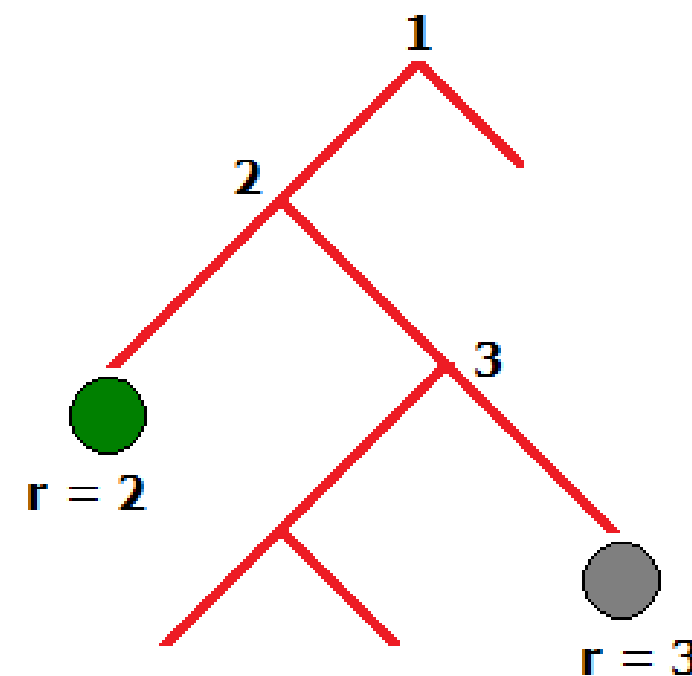
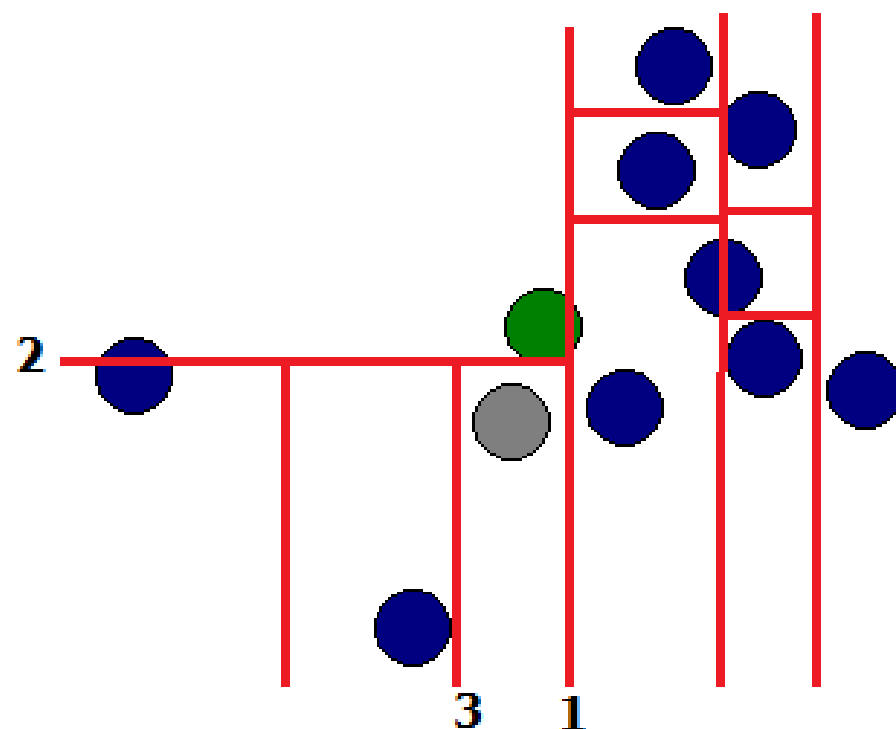
OneClassSVM: `svm.OneClassSVM(nu=0.2, kernel="poly", degree=5)`



Изолирующий лес

Состоит из деревьев, каждое дерево строится до исчерпаниия выборки

- Выбирается случайный признак и случайное расщепление
- Для каждого объекта мера его нормальности – среднее арифметическое глубин листьев, в которые он попал (изолировался)



`sklearn.ensemble.IsolationForest`

`n_estimators` – число деревьев

`max_samples` – объём выборки для построения одного дерева
(если вещественное число, то процент всей выборки)

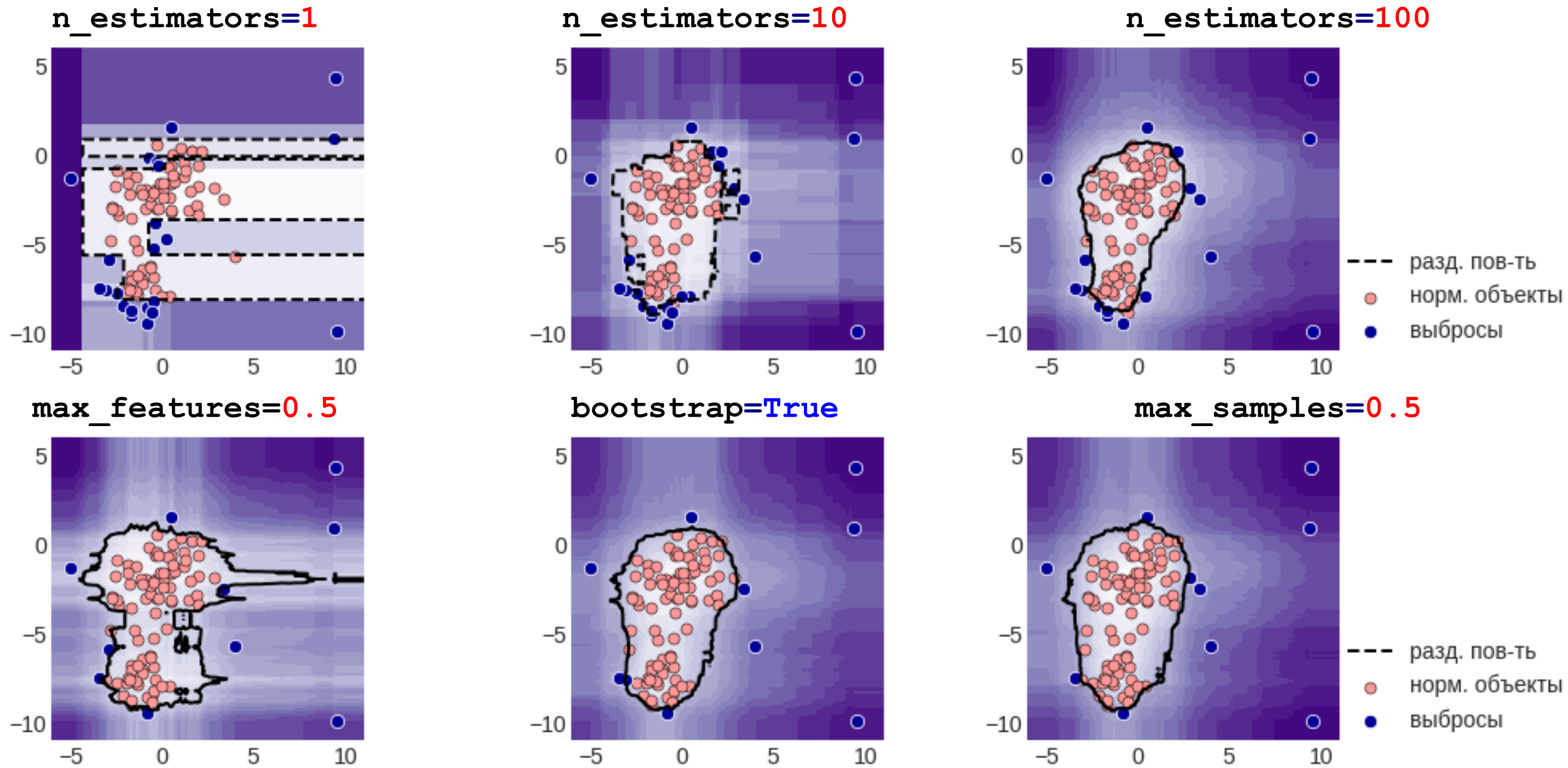
`contamination` – доля выбросов в выборке (для выбора порога)

`max_features` – число признаков, которые используются при построении одного дерева

`bootstrap` – включение режима бутстрепа при формировании подвыборки

```
clf = IsolationForest(n_estimators=100,  
                      max_samples='auto',  
                      contamination='auto',  
                      max_features=1.0,  
                      bootstrap=False,  
                      n_jobs=None,  
                      random_state=1,  
                      verbose=0,  
                      warm_start=False)
```

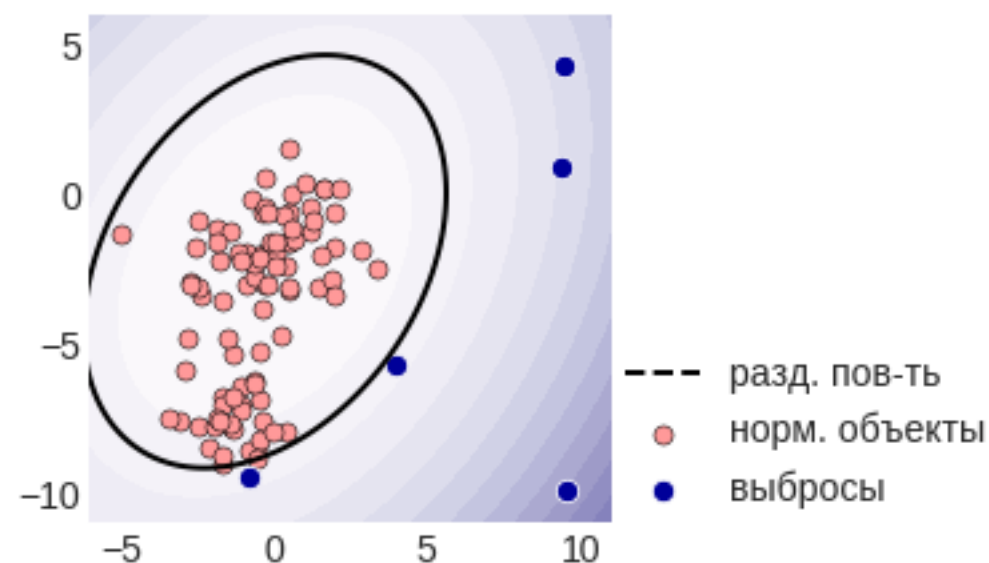
sklearn.ensemble.IsolationForest



Эллипсоидальная аппроксимация данных

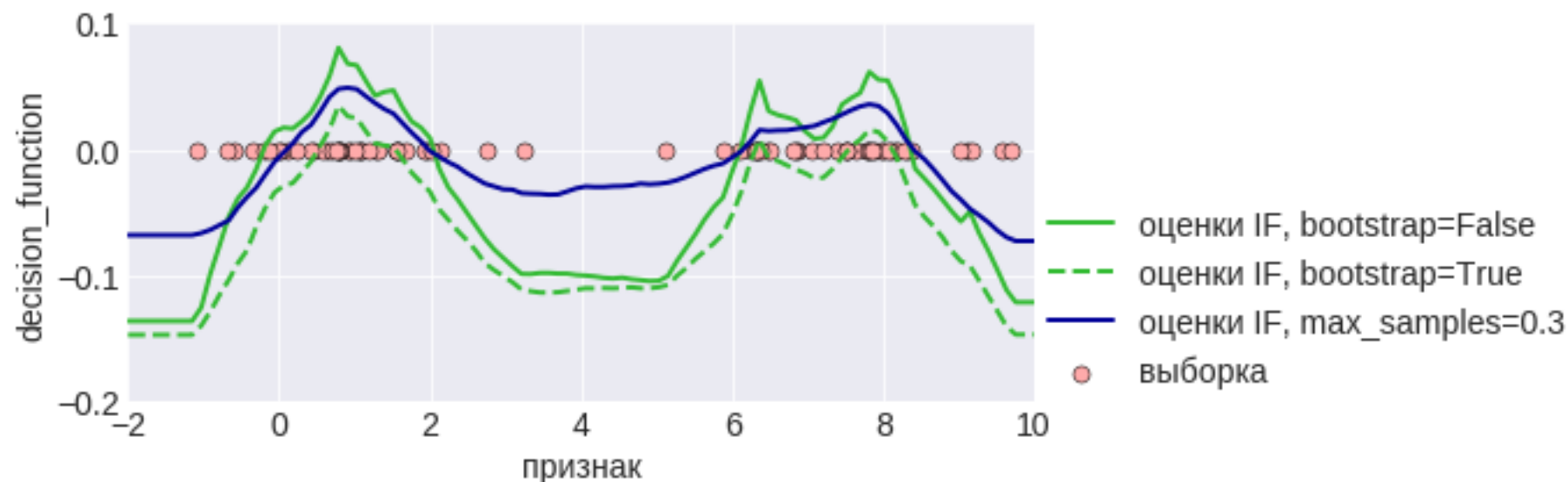
По расстоянию Махаланобиса – степень новизны
– не подходит для мультимодальных данных

```
sklearn.covariance.EllipticEnvelope(store_precision=True,  
                                     assume_centered=False,  
                                     support_fraction=None,  
                                     contamination=0.1,  
                                     random_state=None)
```



Резюме

- Для SVM нужно однородное признаковое пространство, только одно подходящее ядро
 - Леса опять лучше!



```
clf = IsolationForest(n_estimators=1000, max_samples=1.0, contamination=0.1,  
                      max_features=1.0, bootstrap=False, n_jobs=1,  
                      random_state=None, verbose=0)  
  
clf.fit(X)  
a = clf.decision_function(np.linspace(-2, 10, 100)[:, np.newaxis])
```


Ансамбли алгоритмов

Feature Bagging

- **выбираем подмножество признаков**
- **решаем в нём задачу детектирования выбросов**
- **усредняем оценки ненормальности по разным подмножествам признакового пространства**

Rotated Bagging

Перед применением алгоритма – случайный поворот в признаковом пространстве

Часто перебираются разные нормировки признаков

Интересно: результат ансамбля не обязательно среднее... может быть максимум!

История из практики

Задача

**~ 15 датчиков сложного оборудования (насос)
за несколько лет с большой дискретизацией
известно ~ 10 поломок
часть поломок скрыта**

Решение

IF в специальном признаковом пространстве

Итог

Не очень хорошее обнаружение скрытых поломок

История из практики

Заказчик

Выяснение: как работает алгоритм

Итог: алгоритм хороший!

**Он не выявляет качественно все поломки,
но выявляет «нетипичное поведение»,
а это, как оказалось:**

- **некорректный запуск оборудования**
- **слишком позднее выключение**
- **работа в редких режимах**

Приём

**Как и для кластеризации,
можно детекторы аномалий использовать как генераторы признаков**

$$X = [X, IF(X)]$$

- не подглядываем в целевые значения
- интерпретация признака – насколько объект типичен

Итоги

Практически очень важная задача

статистика

модели

итерации

метрики

подмена задачи

ML