Прикладные задачи анализа данных

ИСКУССТВО ВИЗУАЛИЗАЦИИ

Часть 2. Прикладная

Дьяконов А.Г.

Московский государственный университет
имени М.В. Ломоносова (Москва, Россия)

Цели визуализации

- анализ
- иллюстрация слов
- рассказ (story-telling)

нахождение закономерностей
детекция наличия выбросов/аномалий
проверка данных на логичность, полноту и т.п.
придумывание признаков (деформаций, комбинаций, индикаторов)

На данные надо обязательно смотреть!

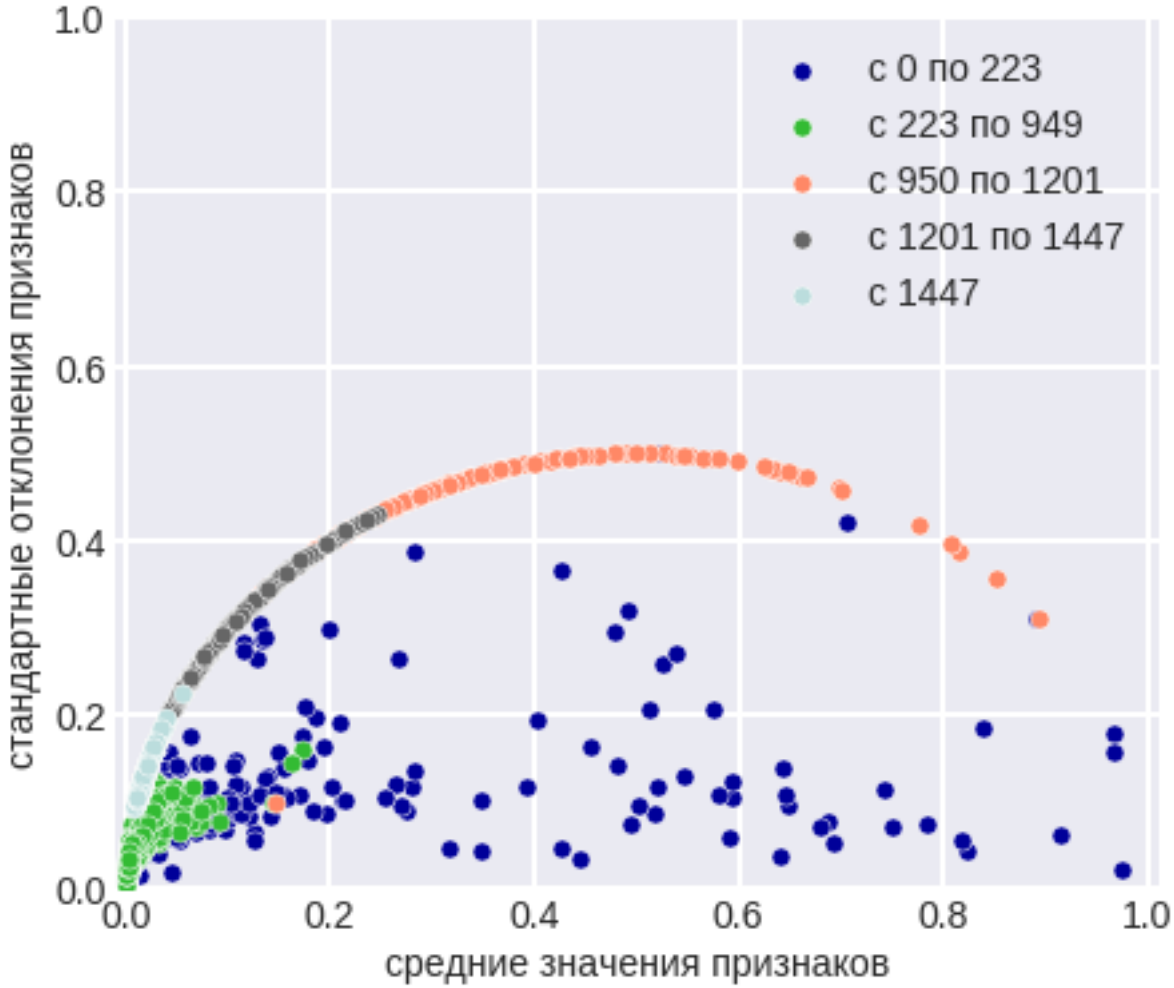
Способы визуализации

- статистики признаков

ЗАДАЧА BIOLOGICAL RESPONSE

Чётко видны группы

Фантастика: дугообразная зависимость у трёх групп признаков!



ВОПРОС: Какие это признаки?

ОТВЕТ: это были бинарные признаки!

У них std зависит от mean (поскольку $x_i^2 = x_i$)!

[0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0]

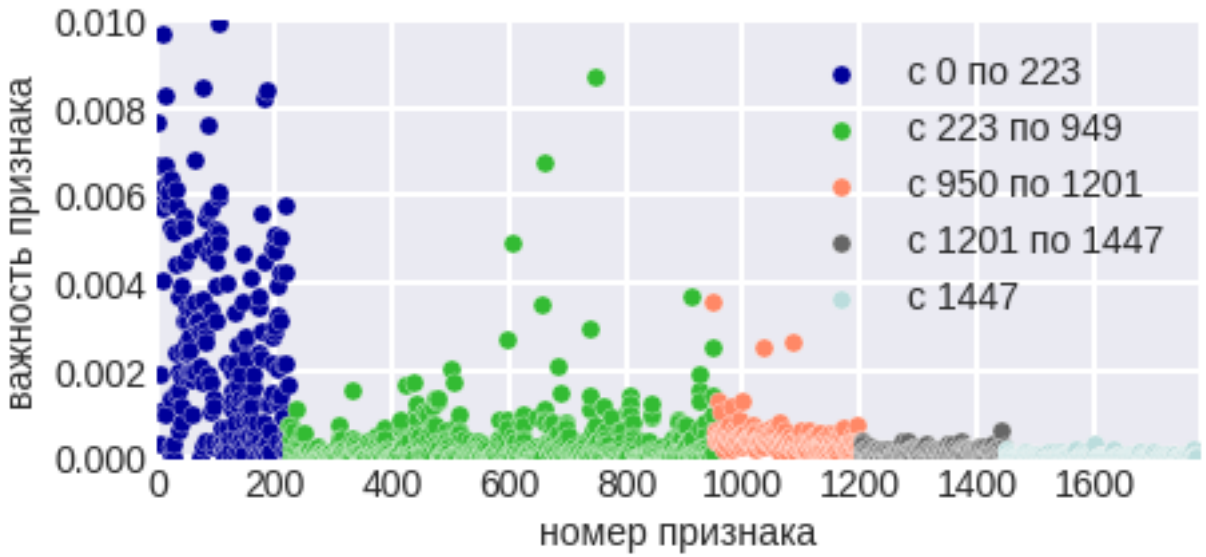
$$\text{mean}\{x_i\}_{i=1}^{n} = \frac{1}{n}\sum_{l=1}^{n} x_i \equiv p$$

$$\text{std}\{x_i\}_{i=1}^{n} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \frac{1}{n}\sum_{l=1}^{n} x_i \right)^2} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - p)^2} =$$

$$= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i^2 - 2px_i + p^2)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - 2px_i + p^2)} =$$

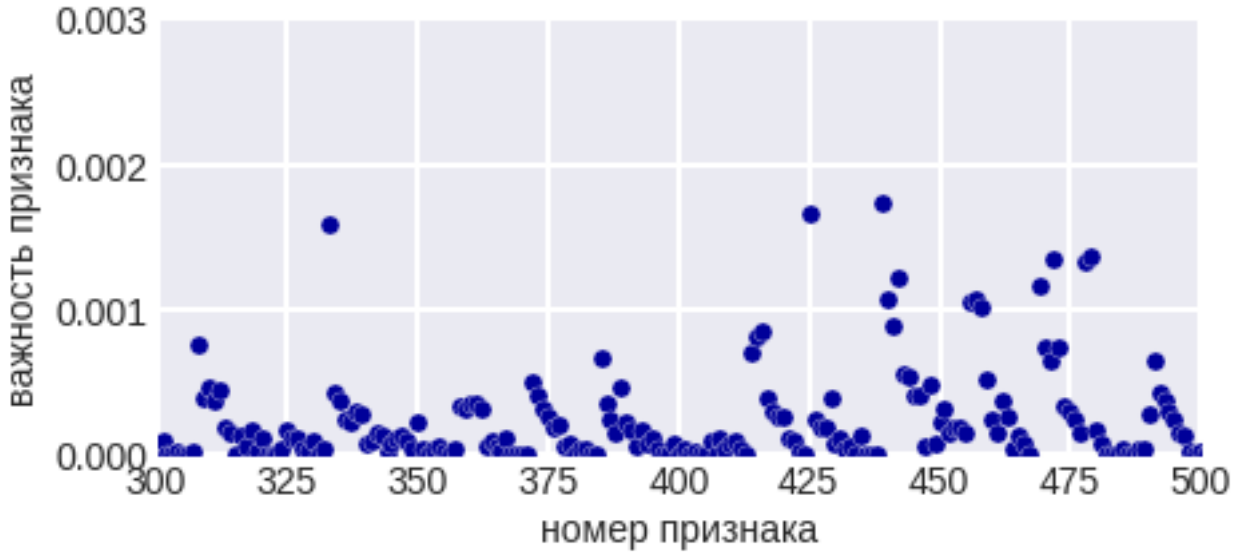
$$= \sqrt{\frac{1-2p}{n}\sum_{i=1}^{n} x_i + p^2} = \sqrt{(1-2p)p + p^2} = \sqrt{p - p^2} = \sqrt{p(1-p)}$$

Важности признаков с точки зрения RF.



Потом: целые группы признаков можно удалять без существенной потери качества

Увеличение картинки



Есть подгруппы признаков!

Меняйте масштаб!

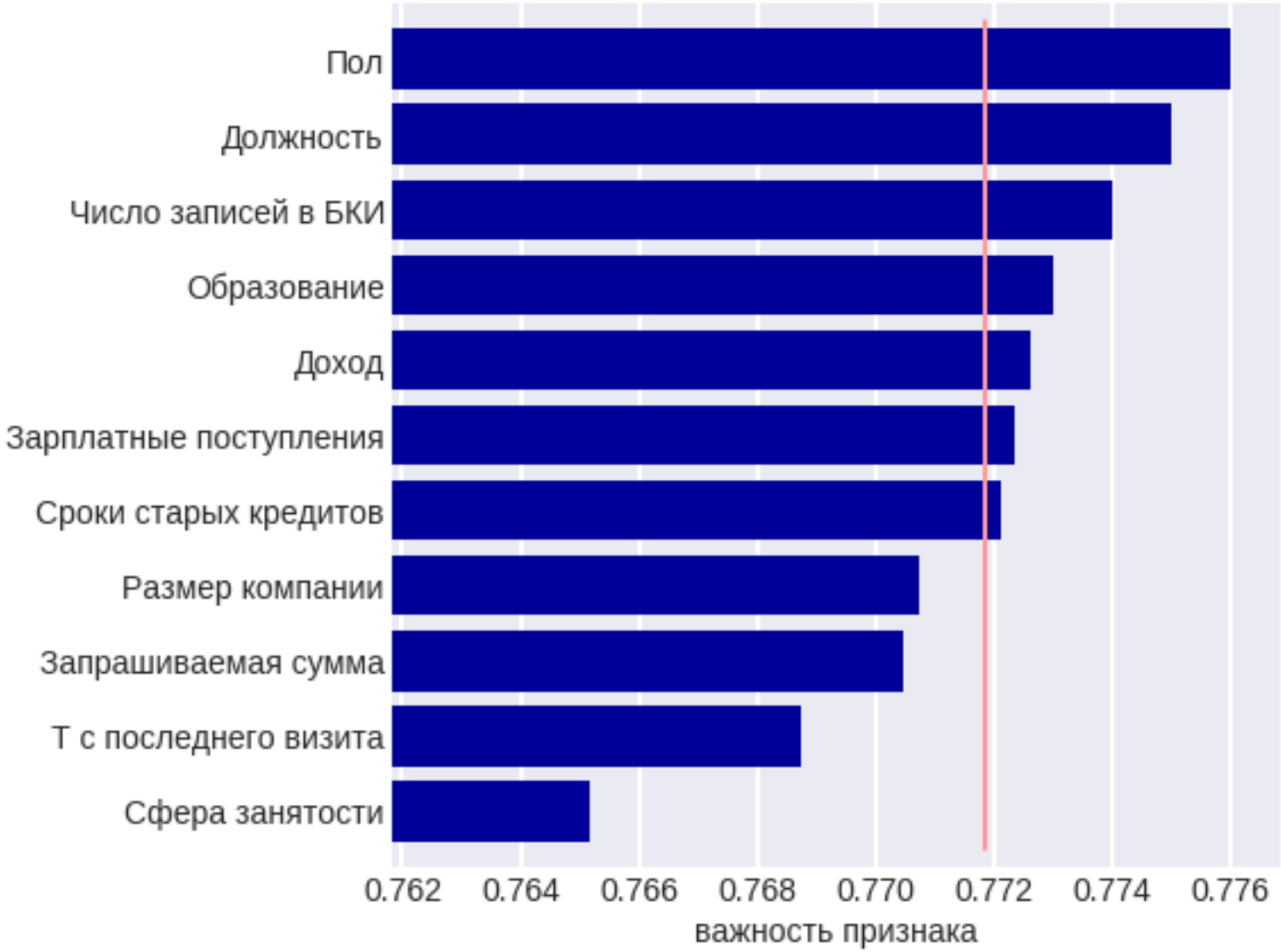
Аналогично: исследование сложности «классификации» объектов

Исследование частей выборки(фолдов)

Подозрительная унимодальная зависимость!

Что значит?

Как правильно показывать важности признаков



Сортировка, среднее значение, вертикальная ориентация

Что часто делается в начале задачи

Задача «Give Me Some Credit»

Статистика признаков

Анализ отдельных признаков: значения должны быть на отрезке [0,1], но есть неожиданные значения + Наны.

Значения

%	Age	#30-59	%	Доход	#o1	#90	#	#60	# в сем
82404 от 0 до 1 потом … Есть дроби!!!	0 (1), 21-109	0-13, 96, 98	70097 до 1 потом… Есть дроби!!!	целые	0-58	0-17, 96, 98	0-26, 32, 54	0-9, 96, 98	0-10, 13, 20

Уникальных значений

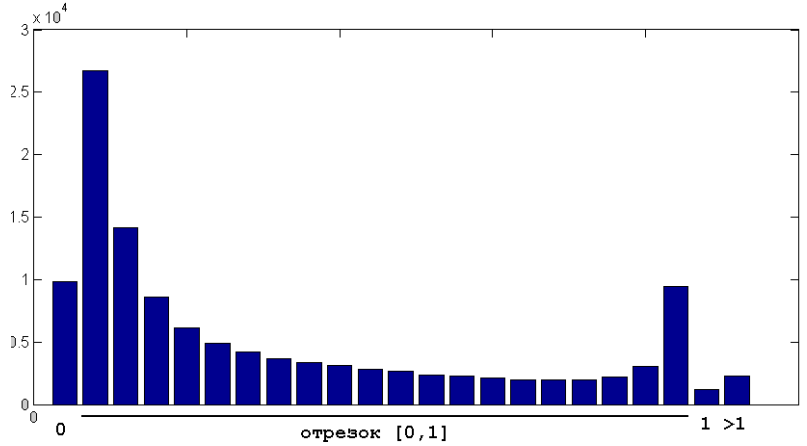
1	2	3	4	5	6	7	8	9	10
84500	86	16	78795	11866	54	19	26	12	13

Нанов

				19831					2630 если тут, то в 5

Аук, Аук через плотность

0.7807	- 0.6356	0.6910	0.5266	- 0.5782	- 0.5484	0.6613	- 0.5383	0.6247	0.5482
0.7631	0.6836	0.7077	0.5046	0.6327	0.5518	0.7347	0.5621	0.6525	0.6071
0.7815	0.6329	0.6910	0.5364	0.5554	0.5497	0.6613	0.5432	0.6247	0.5499

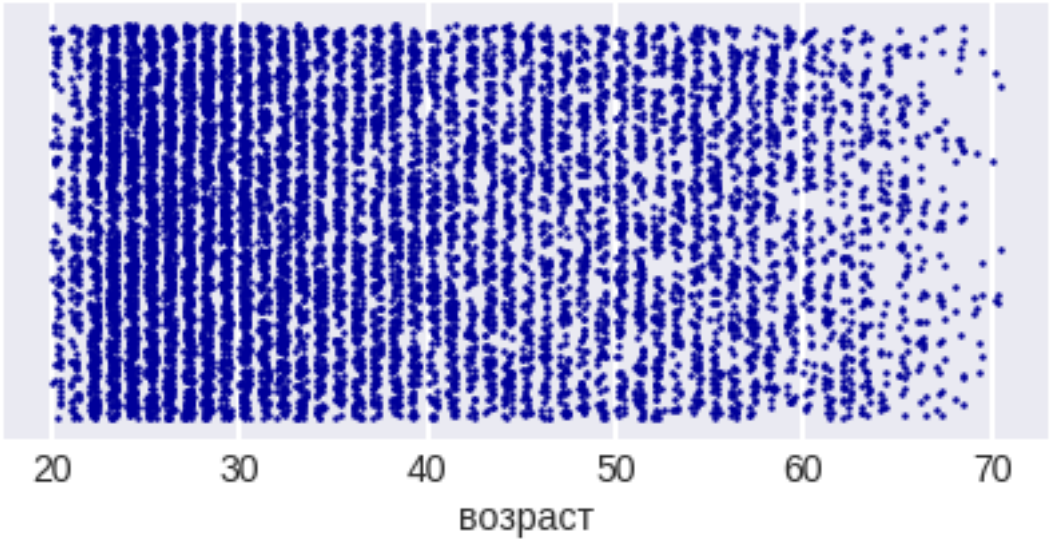


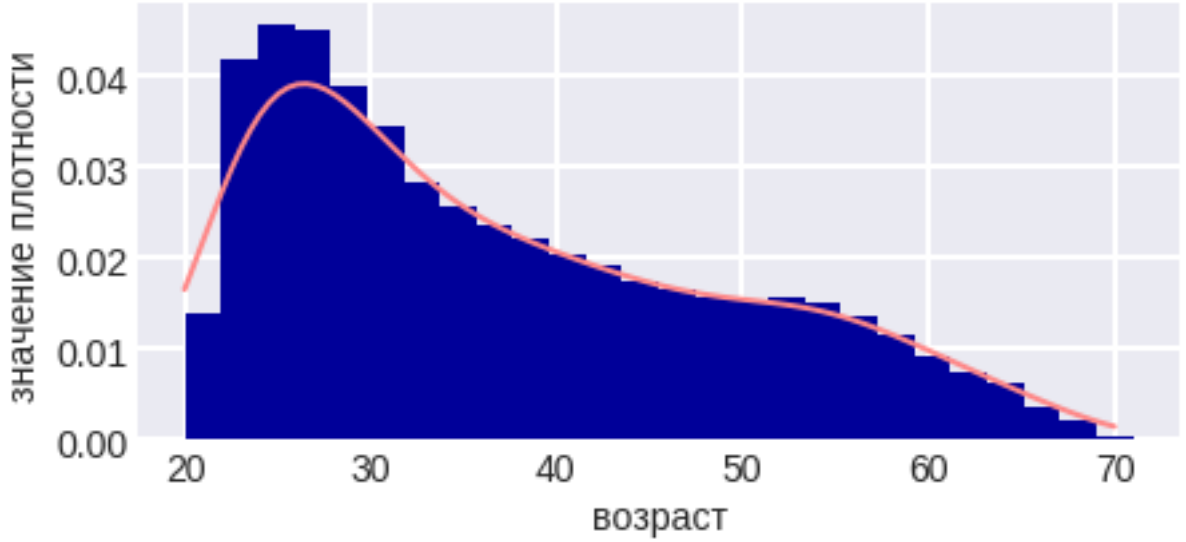
Смотрим на сами признаки

```
for name in data.columns:
    if data[name].nunique()<8:
        u = data[name].unique()
    else:
        u = data[name].unique()[:8]
    if type(data[name].tolist()[0]) is str:
        print ('%25s %10d %10s %10s %s' % (name, data2[name].nunique(), '', 'str', str(u)))
    elif type(data2[name].tolist()[0]) is pd.tslib.Timestamp:
        print ('%25s %10d %10s %10s %s' % (name, data2[name].nunique(), '', 'time',  ''))
    else:
        print ('%25s %10d %10.2f %10.2f %s' % (name, data2[name].nunique(), data2[name].mean(),
                                data2[name].std(), str(u)))
```

```
                    Класс          4       2.20       0.97 [1 2 3 4]
                    Номер       8404    7442.45     269.63 [5001 5002 ...]
                  Вес, т         124      38.27       7.30 [ 41.1  44.4  ...]
                   Начало       8404                  time
          Количество, шт         45      63.78       5.13 [ 66.  61.  ...]
```

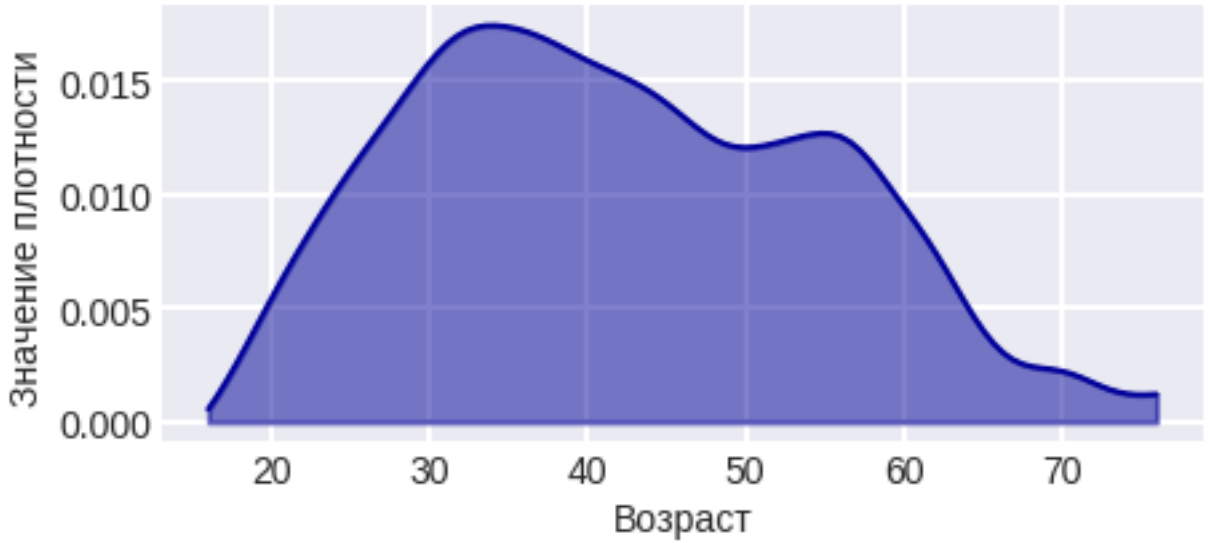
Визуализация отдельных признаков





Недаром нужны гистограммы!

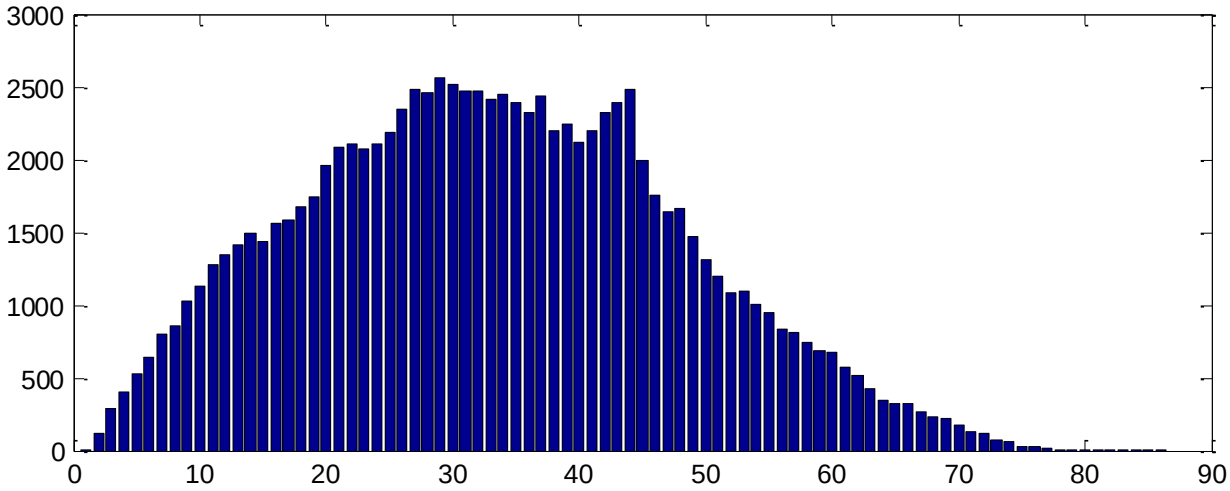
ЗАДАЧА «М-магазин»



Распределение возраста покупателей

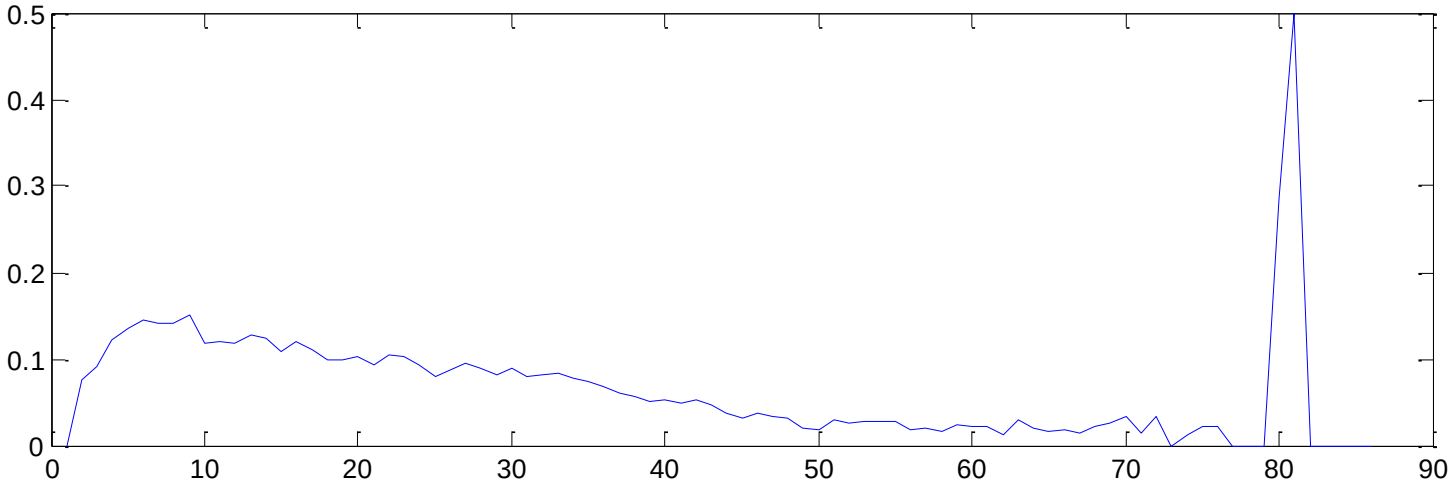
Так обычно выглядит распределение!

ЗАДАЧА «ТКС»



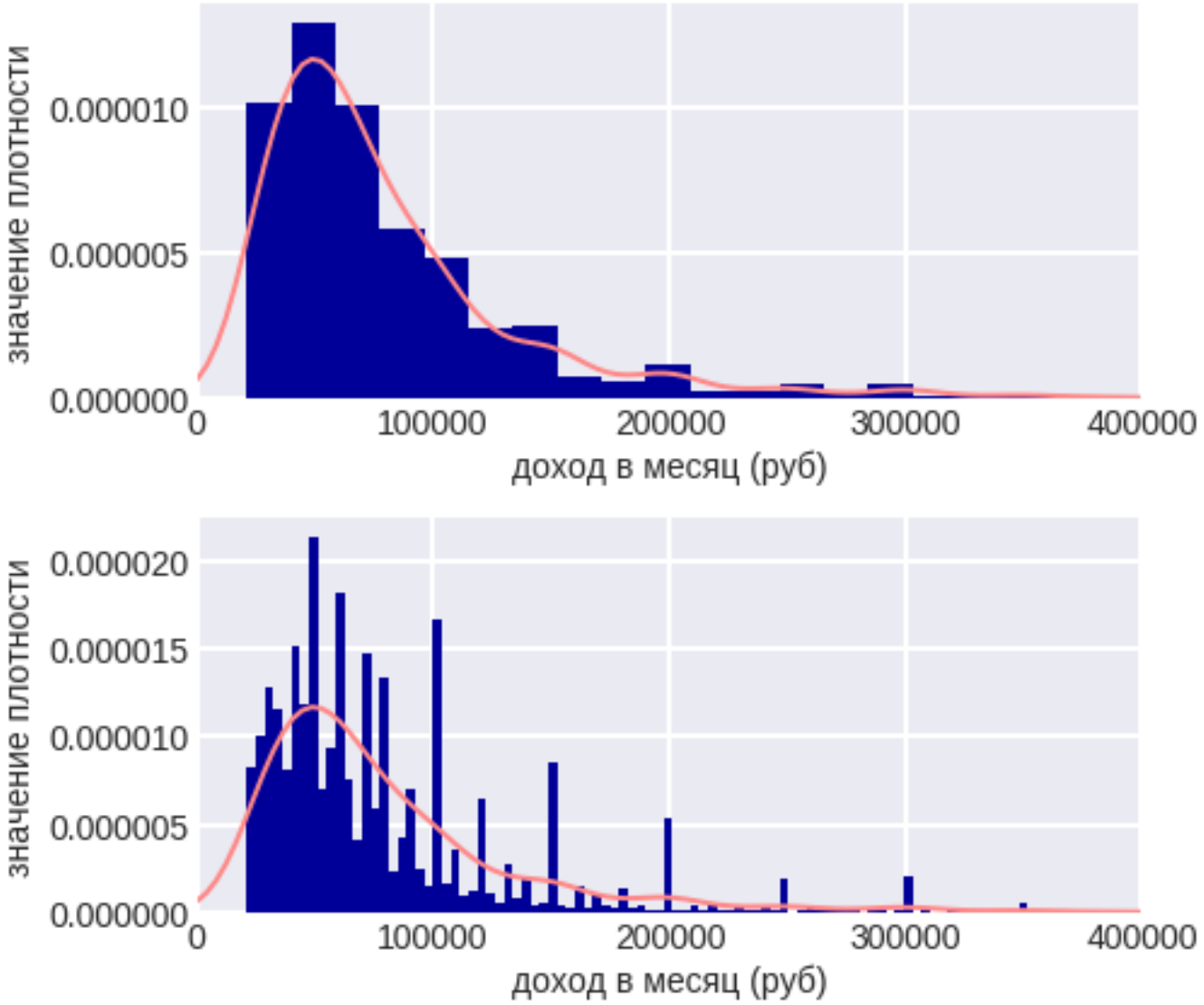
Распределение по возрасту

Что значит?



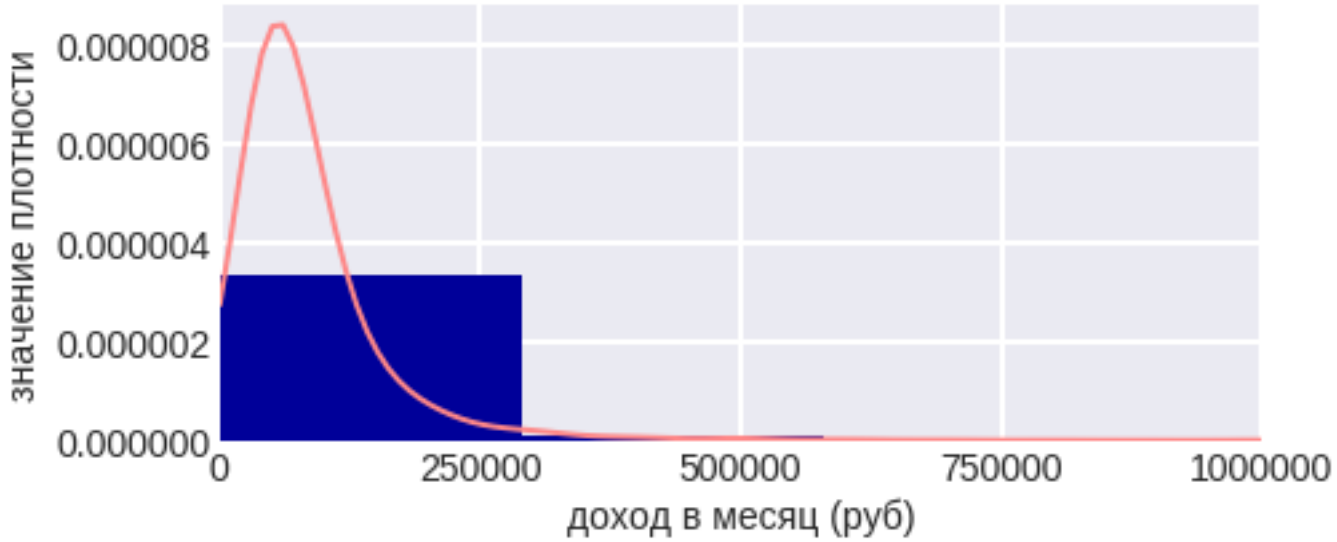
Отношение плотностей – есть явный выброс!

Проблемы визуализаторов – параметры по умолчанию



увеличили число бинов

Проблемы визуализаторов – выбросы



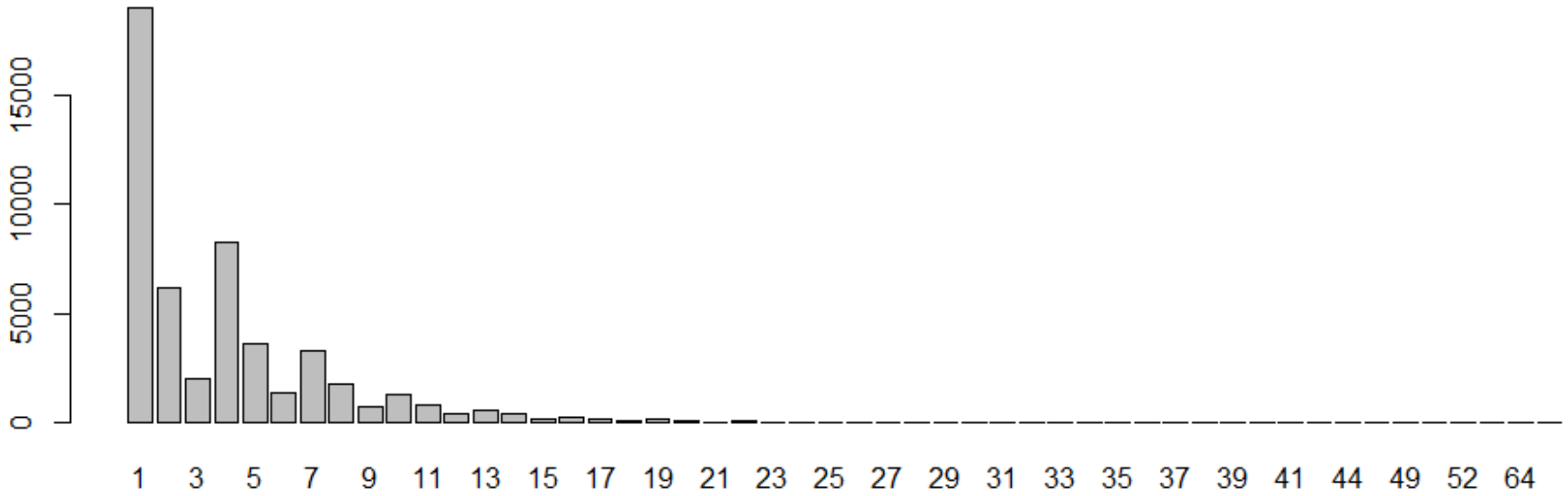
Что будет если не устранять выбросы…

```
def make_clips(data, name):
            return (data[name].clip(lower=data[name].quantile(0.01),
                  upper=data[name].quantile(0.99)).values)
```

Ещё раз о параметрах по умолчанию: «Liberty»

Что интересного в распределении целевого признака?

a transformed count of hazards or pre-existing damages



Ещё раз о параметрах по умолчанию: «Liberty»

Выбирать:

число бинов

ширина столбцов

Визуализация отдельных признаков

Приёмы

- **взять подвыборку**
- **менять число бинов!**
- **самому выбирать бины!**

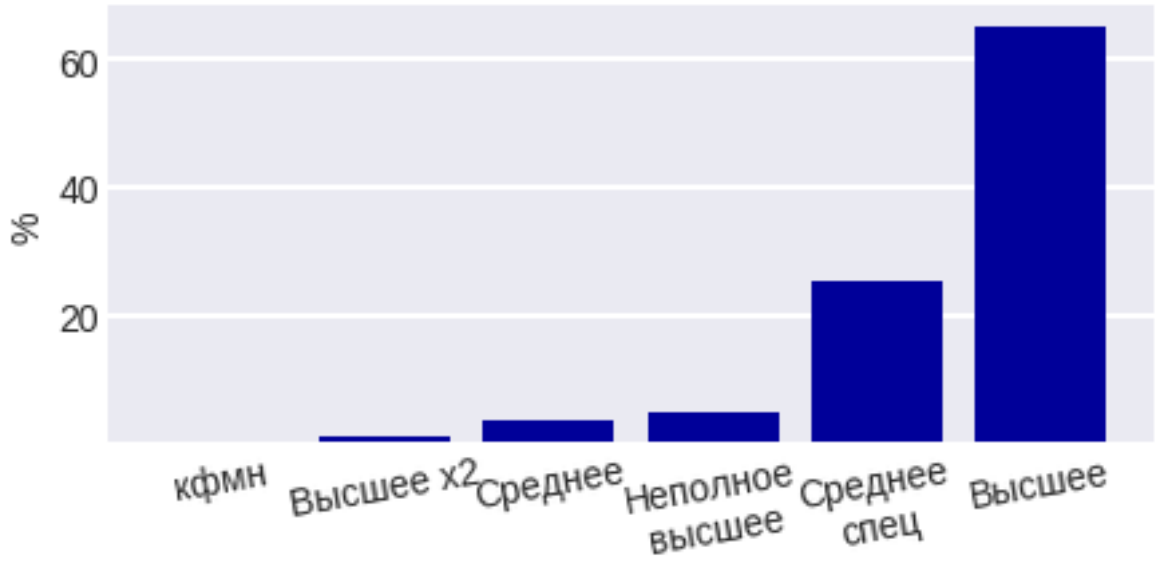
Зачем

- **логичность признака**
- **типичные значения**
- **области типичных значений**
- **преобразования признака**

Сравнение:

- **при разных значениях целевого**
- **на обучении и контроле**

Визуализация категориальных признаков

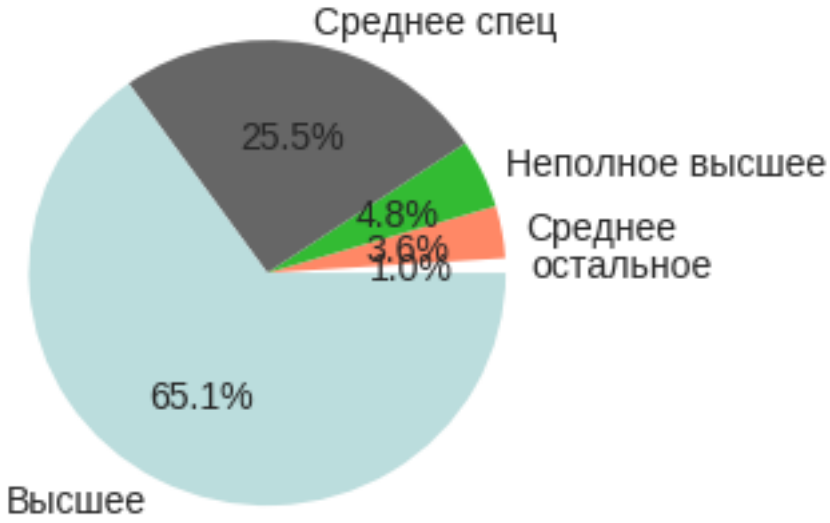


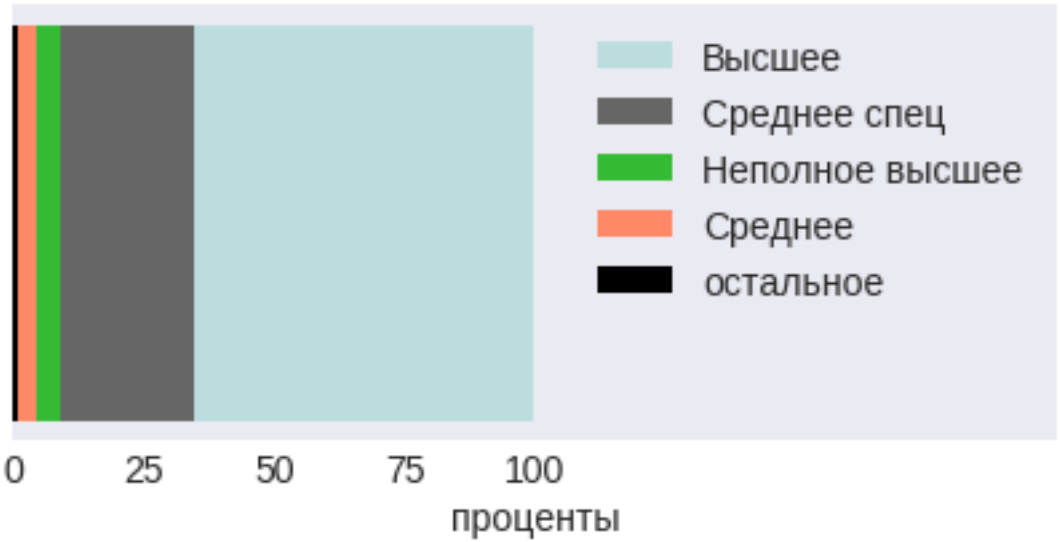
не видно мелкие категории

категорий может быть много

Как быть?

Визуализация категориальных признаков





Визуализация категориальных признаков

Не использовать 3D-эффекты

Мелкие категории → «остальное»

Площадь всех категорий = 100%

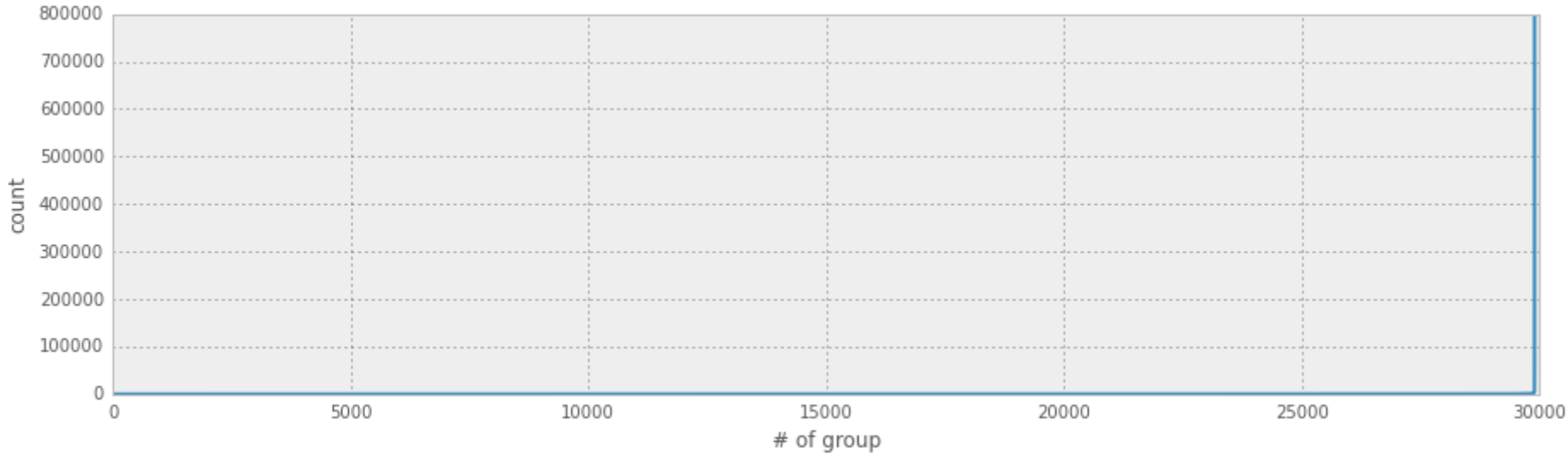
Диаграмма-пирог – не рекомендуется

Когда информации для визуализации мало – таблицы!

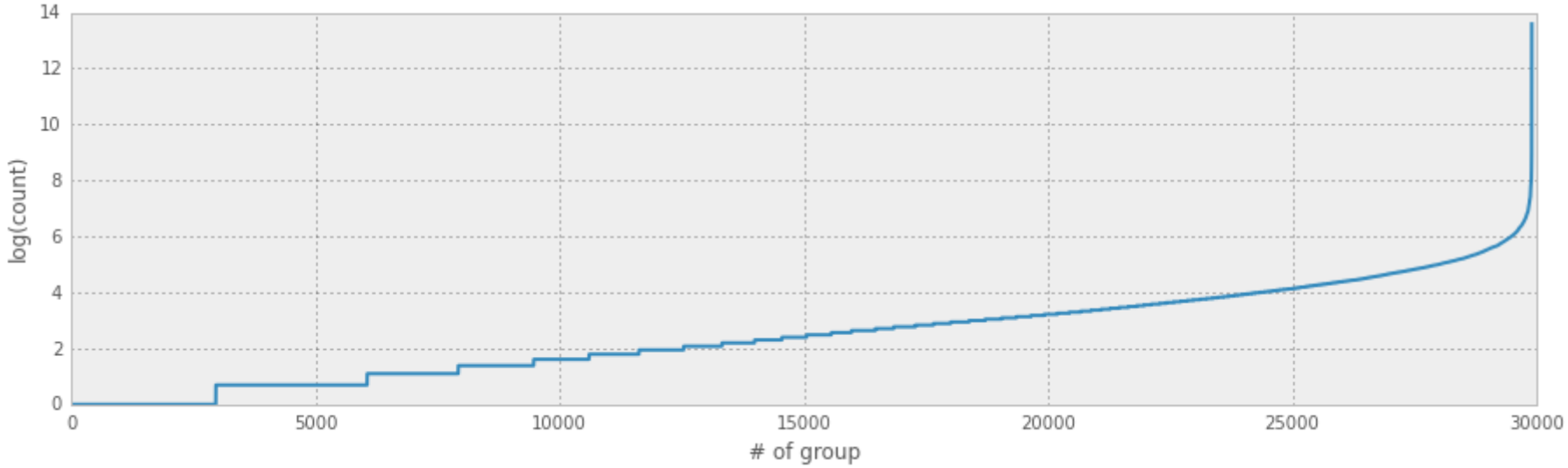
Образование	%
Высшее	65.1
Среднее спец	25.5
Неполное высшее	4.8
Среднее	3.6
Высшее х2	0.8
кфмн	0.2

Можно ещё логарифмировать…

Зачем ещё нужно логарифмирование



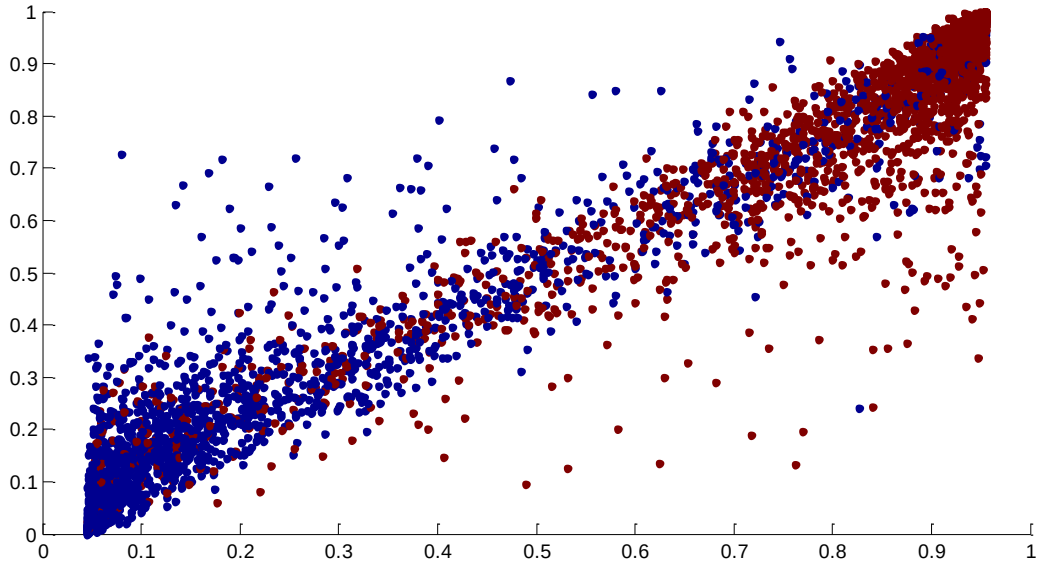
число представителей одной из ~30000 групп в выборке

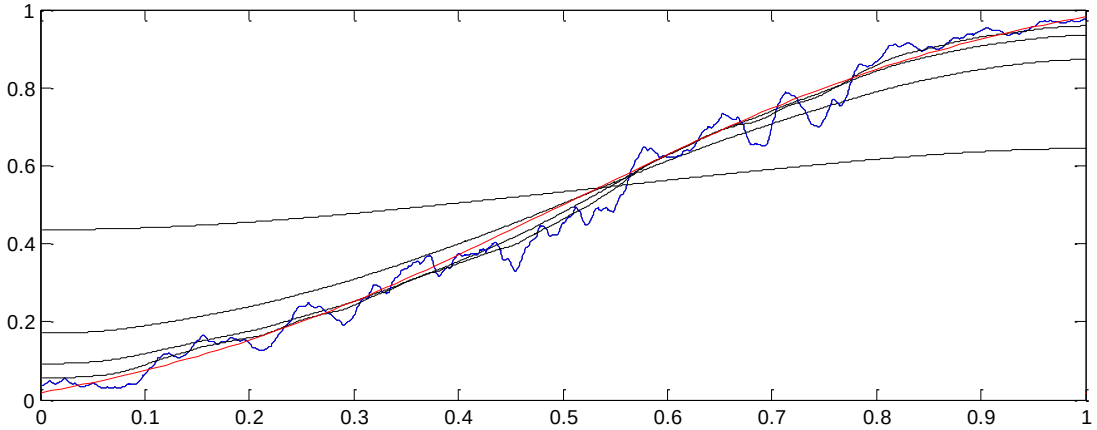


логарифм этого числа

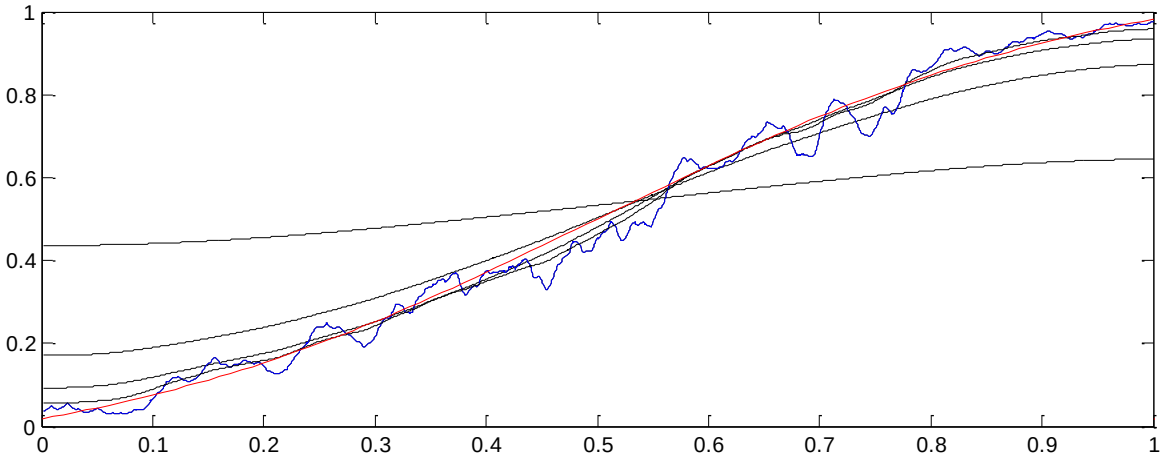
Реальная прикладная задача

«Biological Response»





«Deformation of Random Forests»

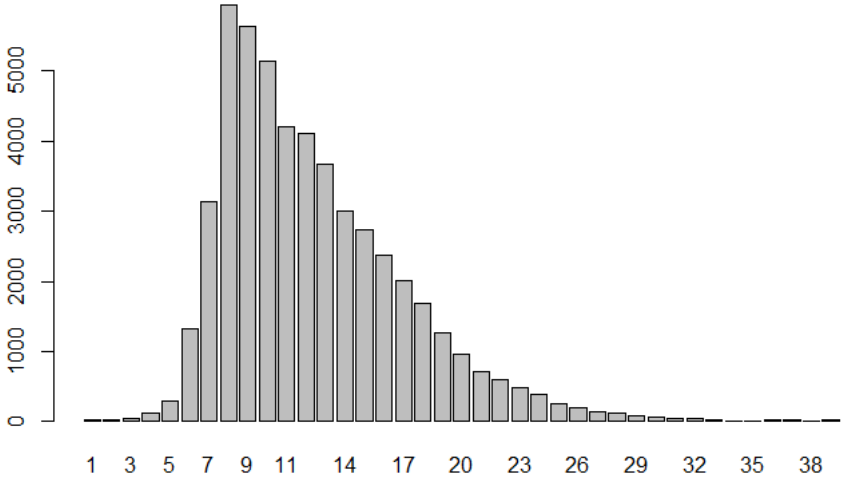


$$\beta\left(\frac{1}{1+e^{-\alpha(x-0.5)}}-0.5\right)+0.5$$

Распределения на признаках – природа признаков

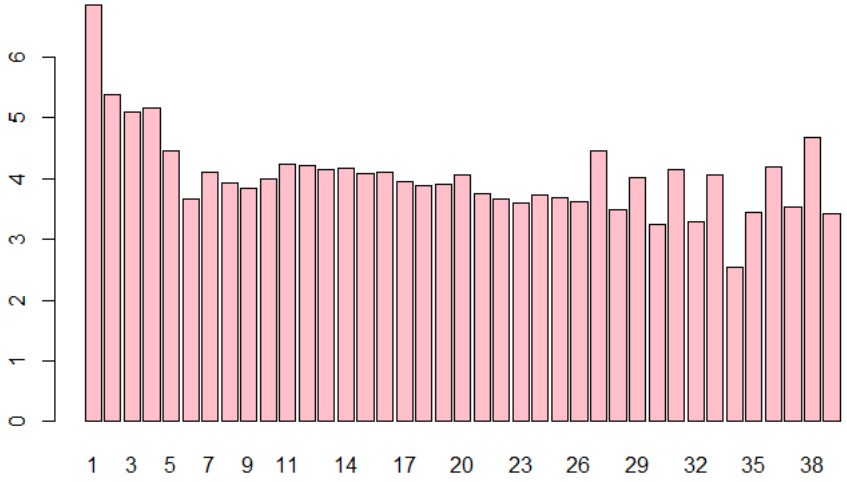
Задача «Liberty»

Целочисленный признак – вещественный или категориальный?




```
barplot(table(train[,21]))
```

Распределение значений признака

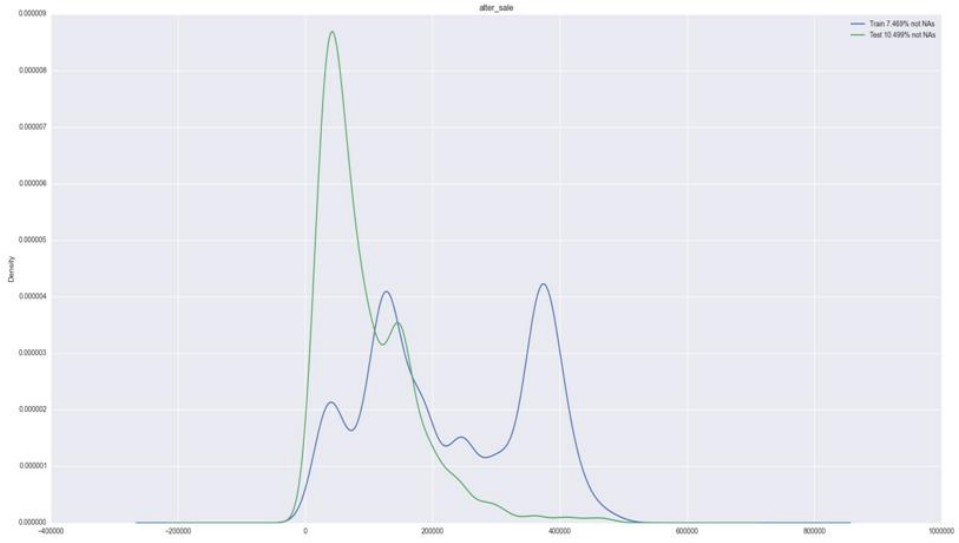



```
barplot(tapply(train$Hazard, train[,34], mean),
        col='pink')
```

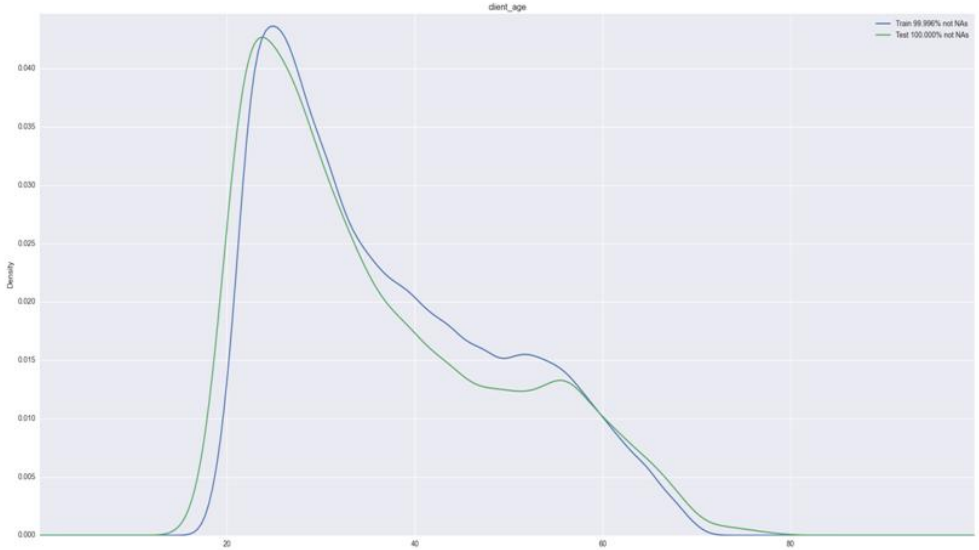
Среднее цели на значениях признака

Как распределение меняется при переходе к контролю

смотреть как меняются распределения
обучение – контроль



Есть существенные изменения



Нет изменений

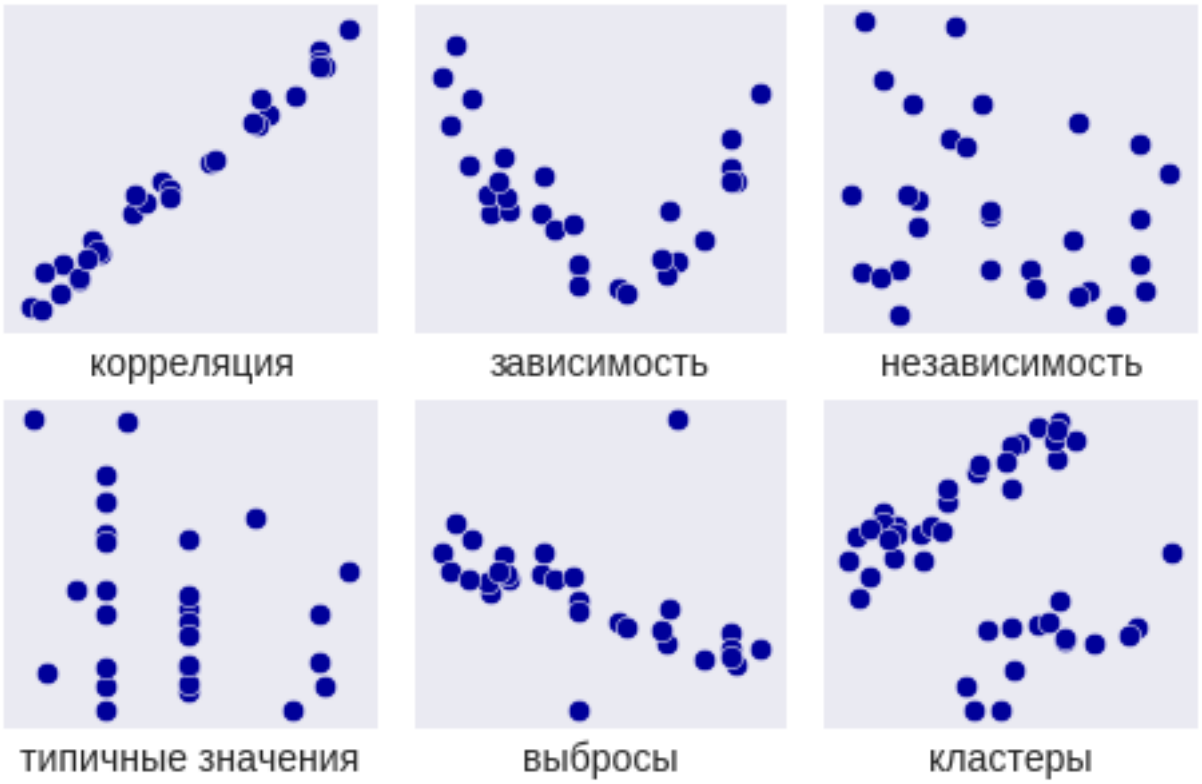
История про о-трэвел и волшебный признак.

Визуализация пары признаков

Самый распространённый способ –
диаграмма рассеивания («скатерплот»)

А что на диаграмме рассеивания 2х признаков можно увидеть?

Что можно увидеть в данных («признак» – «признак»)



Что можно увидеть в данных («признак» – «признак»)

корреляцию

при правильном масштабе и небольшом шуме

зависимость признаков

при малом шуме и «достаточно равномерном» распределении

независимость признаков

часто это «ложное видение»

типичные значения

сложно при большом объёме данных

выбросы

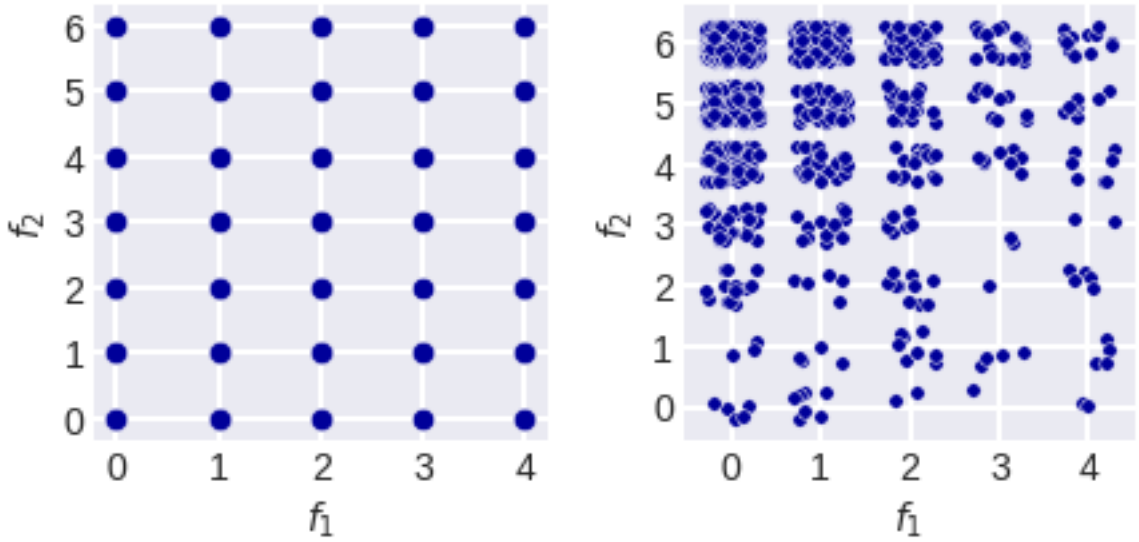
при правильном масштабе

кластеры

при правильном масштабе

Диаграммы рассеивания дискретных признаков

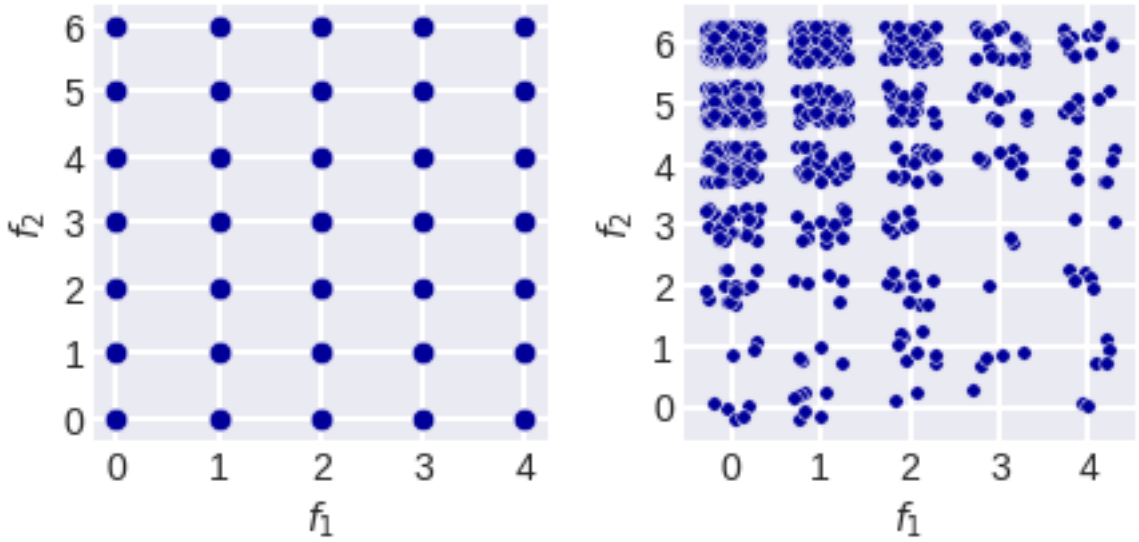
Зачем нужен Jitter



Что видно?

Диаграммы рассеивания дискретных признаков

Зачем нужен Jitter



Что видно?

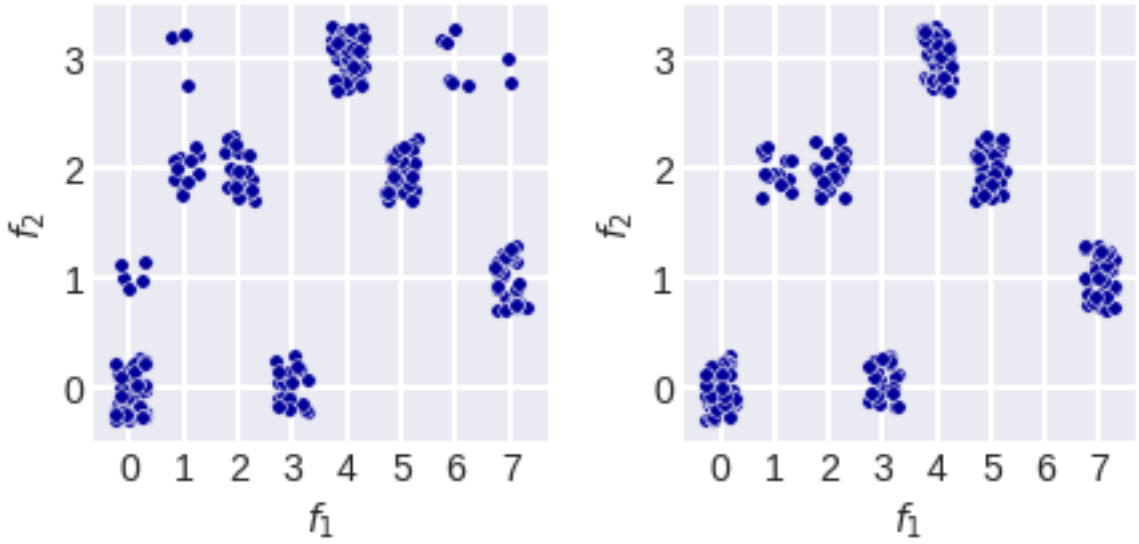
«Треугольная зависимость»

(т.е. ~~взаимная~~ нумерация имеет смысл)

Сводная таблица

	0	1	2	3	4	5	6
0	5	3	13	24	59	152	405
1	7	4	5	14	25	56	154
2	2	8	10	8	16	21	60
3	1	4	1	2	9	10	21
4	2	4	5	2	7	8	12

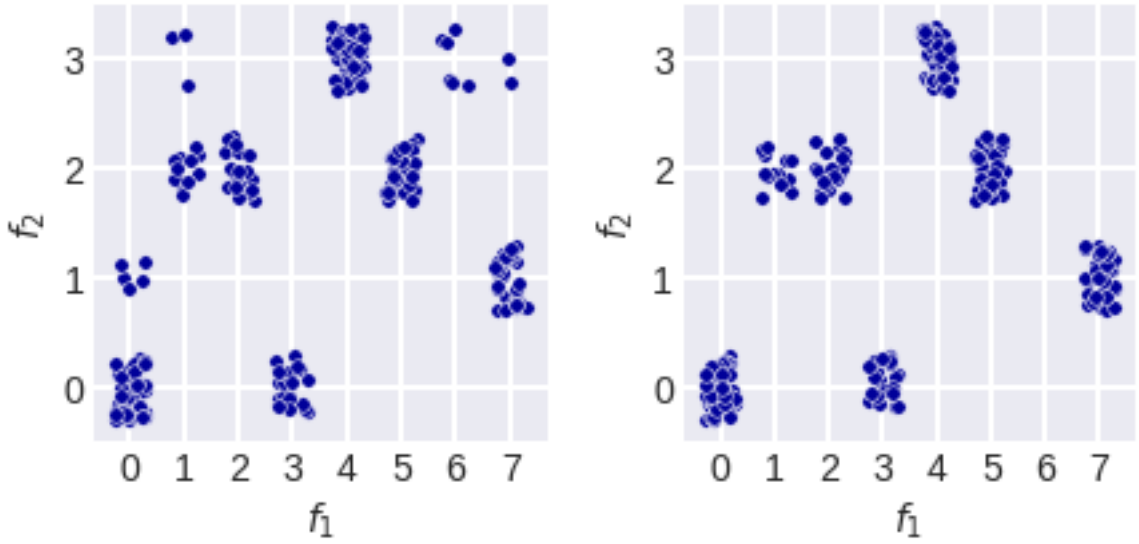
Диаграммы рассеивания дискретных признаков



Справа – после удаления маленьких кластеров!

Что здесь видно?

Диаграммы рассеивания дискретных признаков



Один признак – уточнение другого!

«Liberty»

Из задачи «Liberty»

Верхняя треугольная зависимость

```
table(train$T2_V6,train$T2_V14)
```

```
     1     2     3     4     5     6     7
1 9840  1463   831   376   106    28    17
2  485 21233  3957  4137  1440   396   128
3   79   141  2570   794   431   106    41
4   30    66    22  1180   204   175    75
5    9    15     7     3   212    58    60
6    0     6     0     1     2    96    53
7    0     4     1     4     2     0   115
```

Обоснование необходимости использования пар признаков

```
table(train$T2_V11,train$T2_V13)
```

```
     A     B     C     D     E
N 10160   323   803   513  2260
Y   100   191  6704  4571 25374
```

```
tapply(train$Hazard,
       list(train$T2_V11, train$T2_V13),
       mean)
```

```
        A        B        C        D        E
N 3.876378 5.099071 4.574097 5.518519 3.946460
Y 3.810000 4.319372 4.231653 4.175016 3.942815
```

Из задачи «RedHat»

```
people[:5]
```

	people_id	char_1	group_1	char_2	date	char_3	char_4	char_5	char_6	char_7	char_8	char_9	char_10
0	ppl_100	type 2	group 17304	type 2	2021-06-29	type 5	type 5	type 5	type 3	type 11	type 2	type 2	True
1	ppl_100002	type 2	group 8688	type 3	2021-01-06	type 28	type 9	type 5	type 3	type 11	type 2	type 4	False
2	ppl_100003	type 2	group 33592	type 3	2022-06-10	type 4	type 8	type 5	type 2	type 5	type 2	type 2	True
3	ppl_100004	type 2	group 22593	type 3	2022-07-20	type 40	type 25	type 9	type 4	type 16	type 2	type 2	True

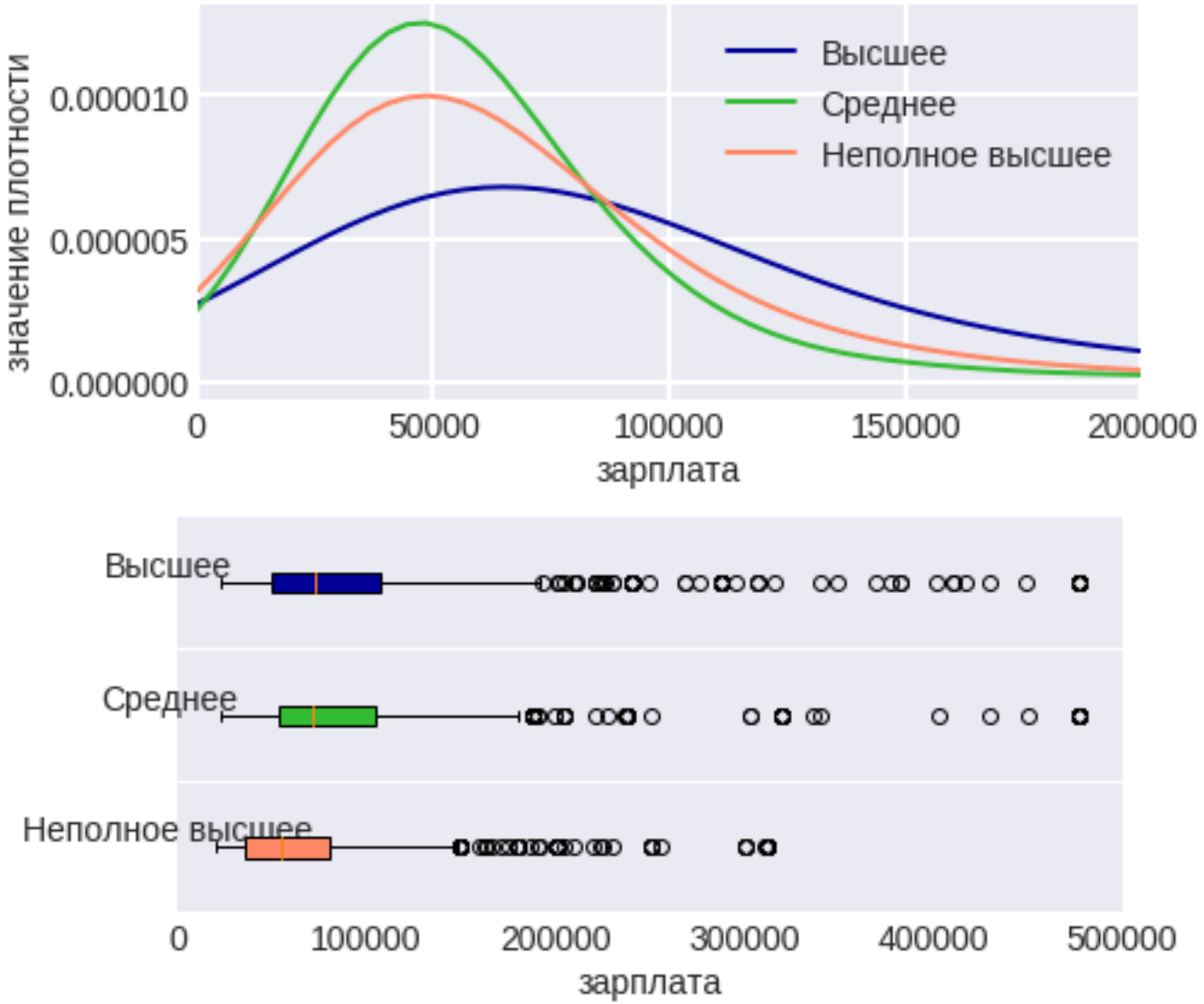
По таблице объект-признак сложно увидеть, что один категориальный признак – уточнение другого

```
pd.crosstab(people.char_1, people.char_2)
```

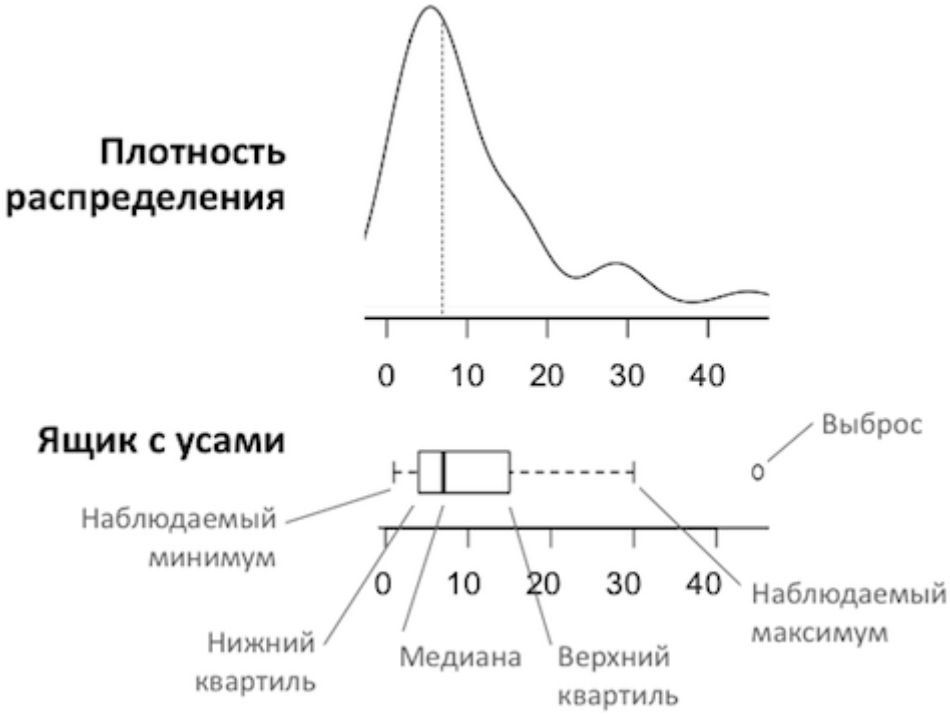
char_2	type 1	type 2	type 3
char_1			
type 1	15251	0	0
type 2	0	77314	96553

Как использовать это знание?

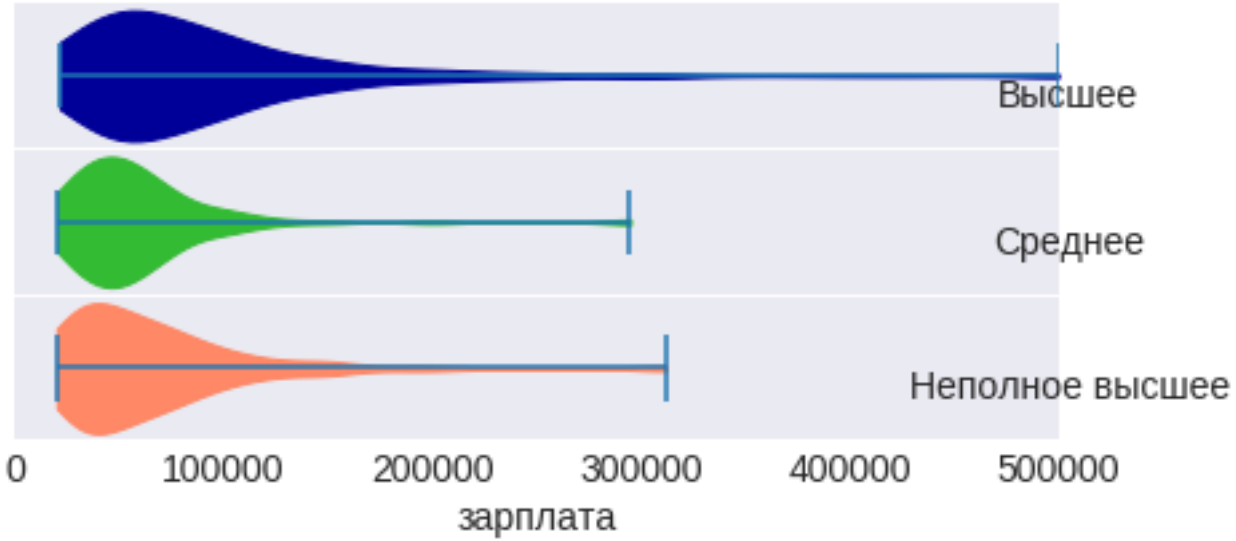
Пара «вещественный признак – категориальный»



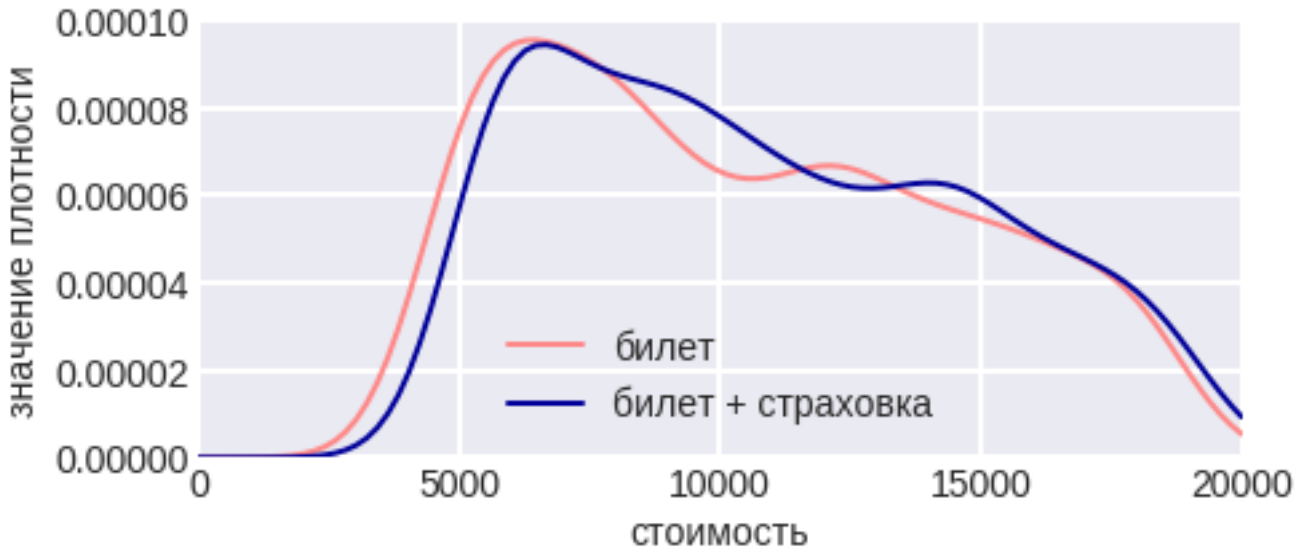
Ящик с усами (box-plot)



Всё это не очень наглядно...



ЗАДАЧА «О-Т»

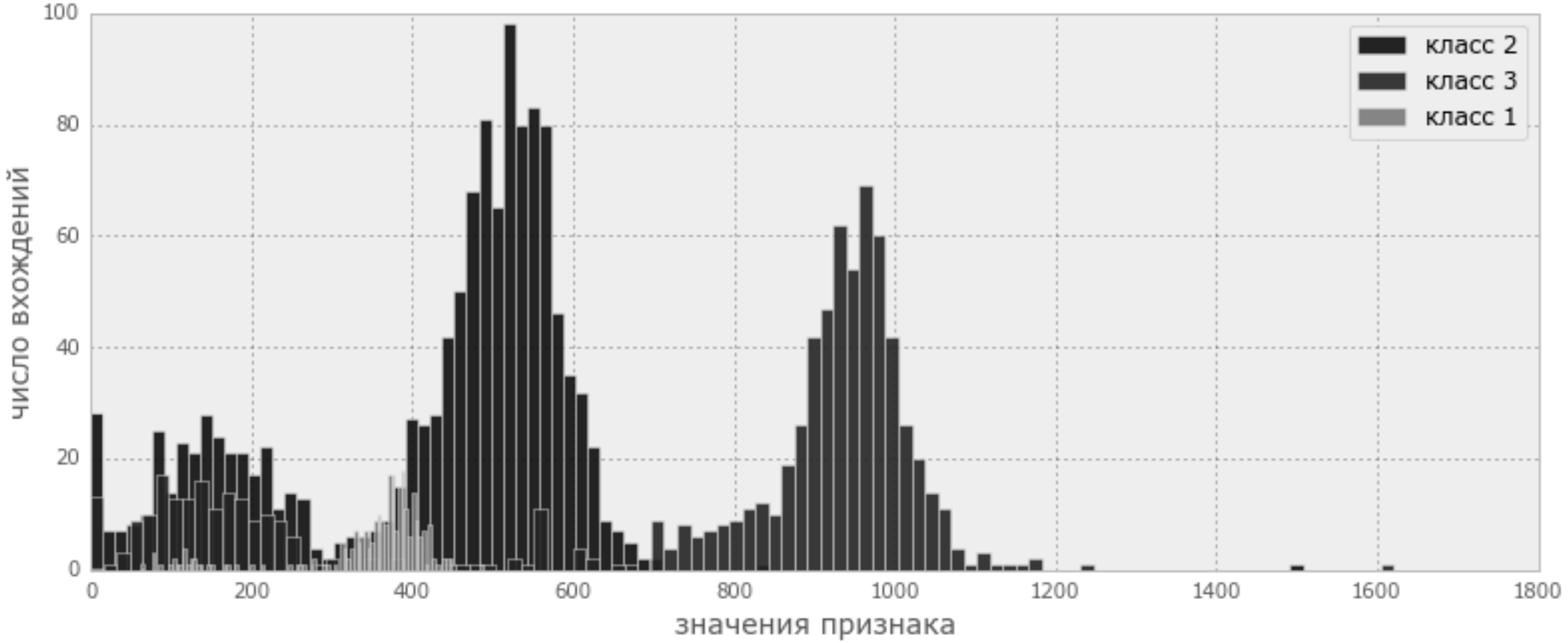


Всегда ставьте под сомнение свои выводы!

Как распределена цель на признаках

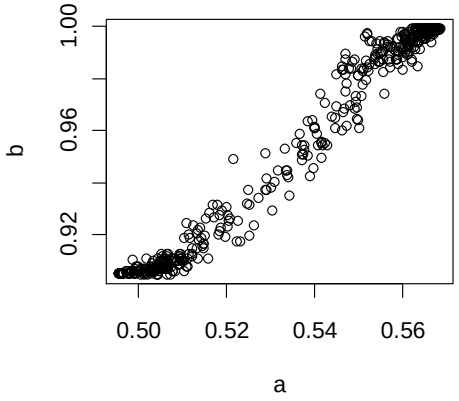
Чем плох рисунок?

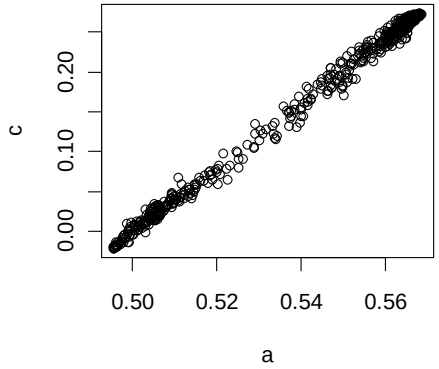
Как распределена цель на признаках



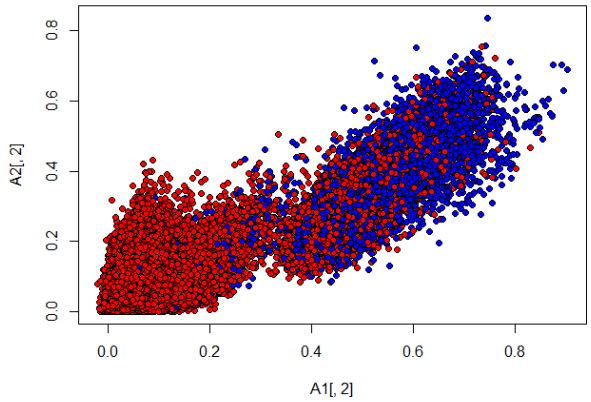
Вот чем…

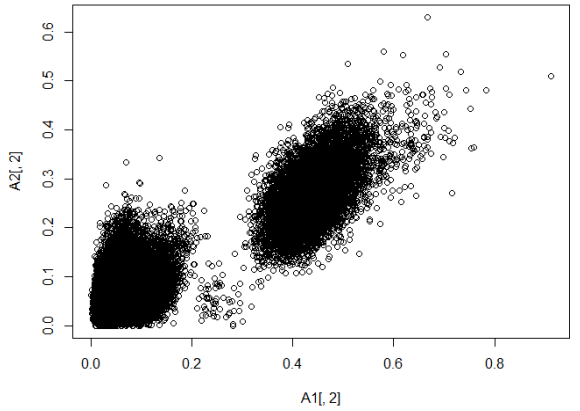
Визуализация ответов двух алгоритмов





Как найти ошибку используя бенчмарк…

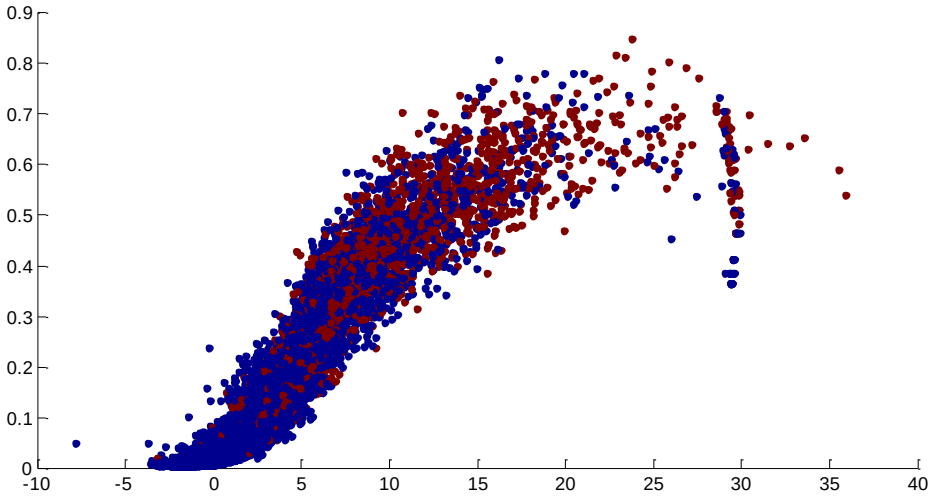




Совет: создавайте бенчмарк!

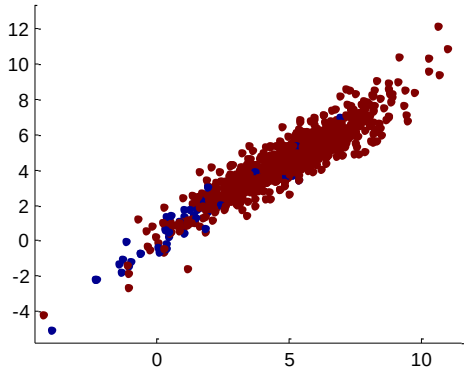
Ещё о визуализации «алгоритм-алгоритм»

Задача скоринга

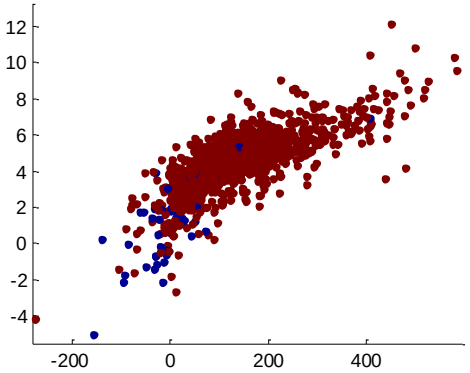


Мой – горизонталь и RF – вертикаль

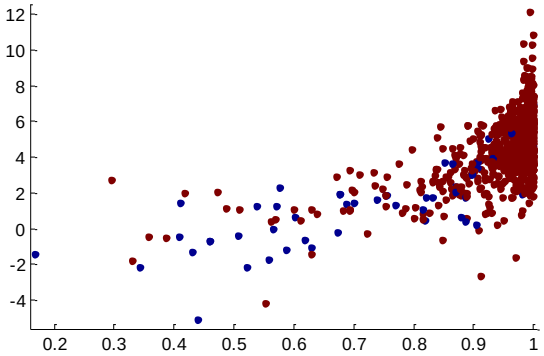
В задаче AMAZON



Разные LIBLINEAR

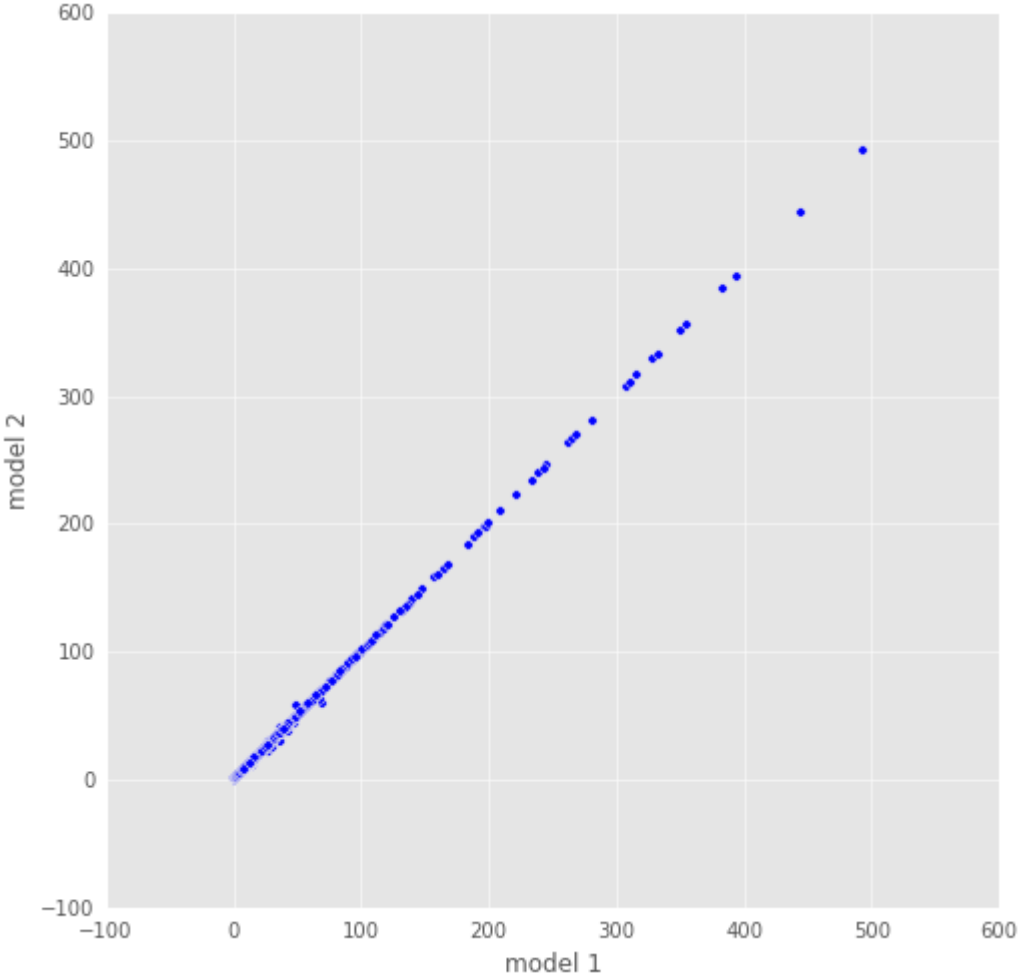


LIBLINEAR и PERCEPTRON

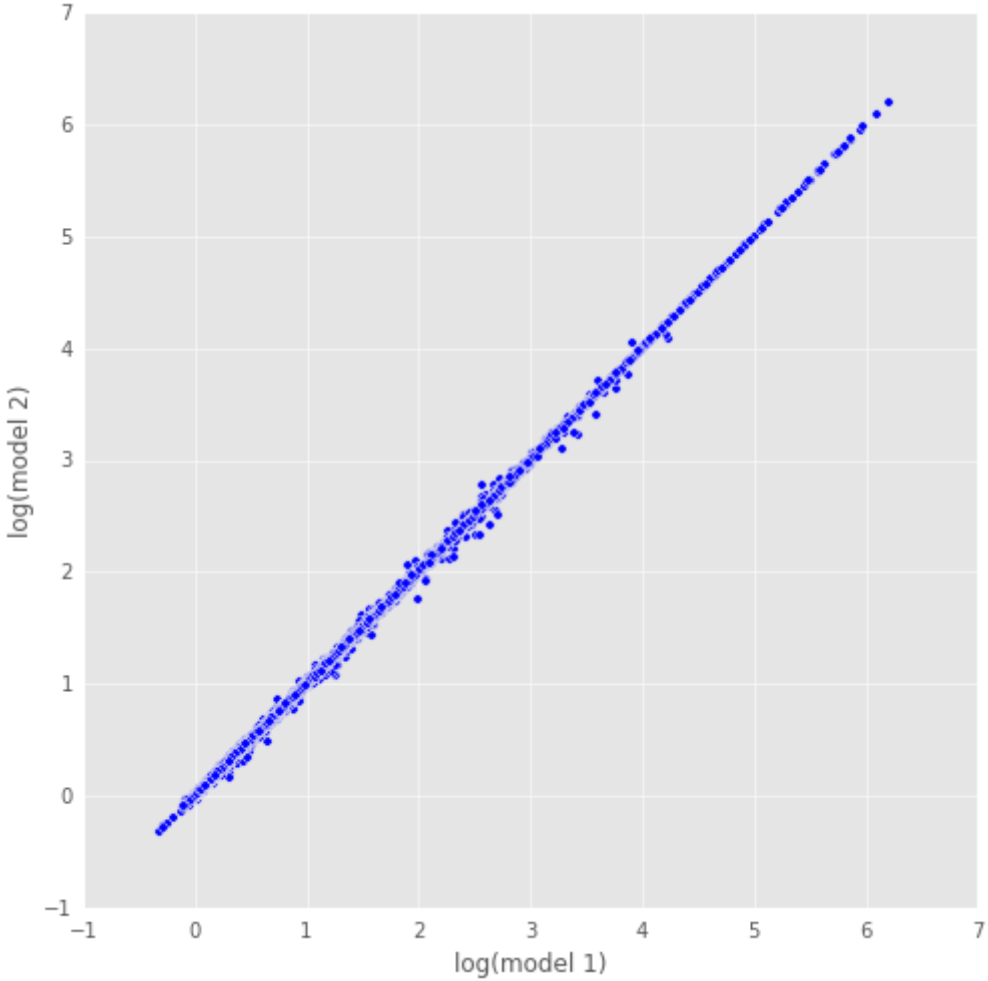


kNN и LIBLINEAR

Ещё о визуализации «алгоритм-алгоритм»

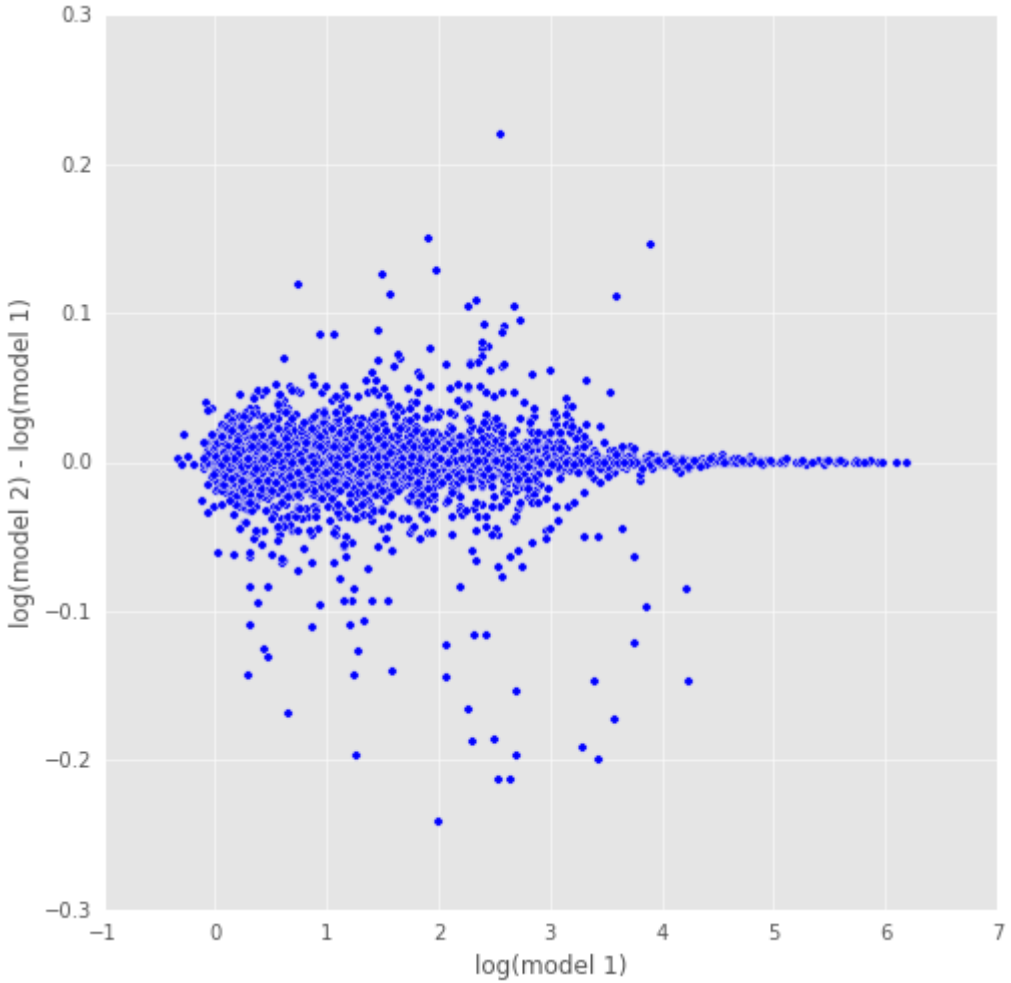


Две модели



Опять логарифмирование шкал!

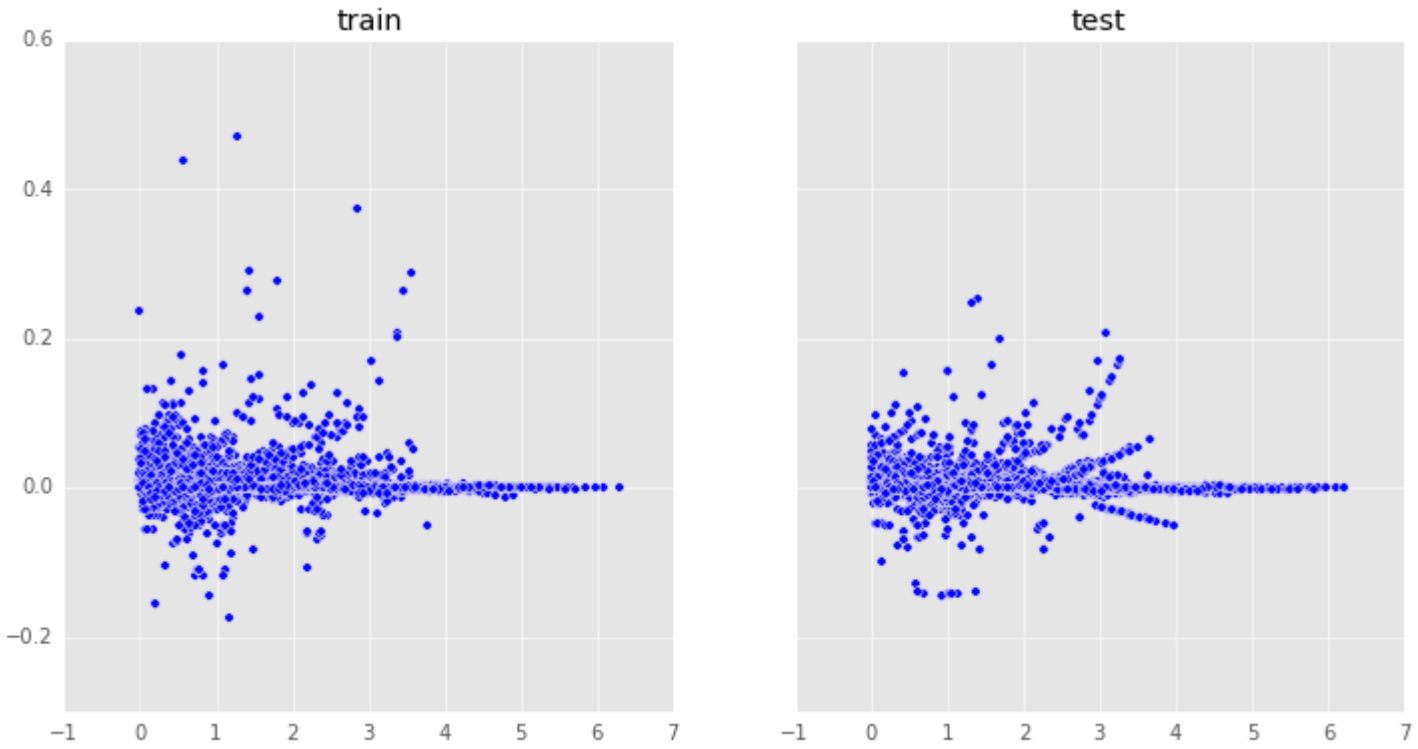
Ещё о визуализации «алгоритм-алгоритм»



Опять смотрим разницу ответов

Наблюдение: при больших значениях модели работают идентично!

Ещё о визуализации «алгоритм-алгоритм»

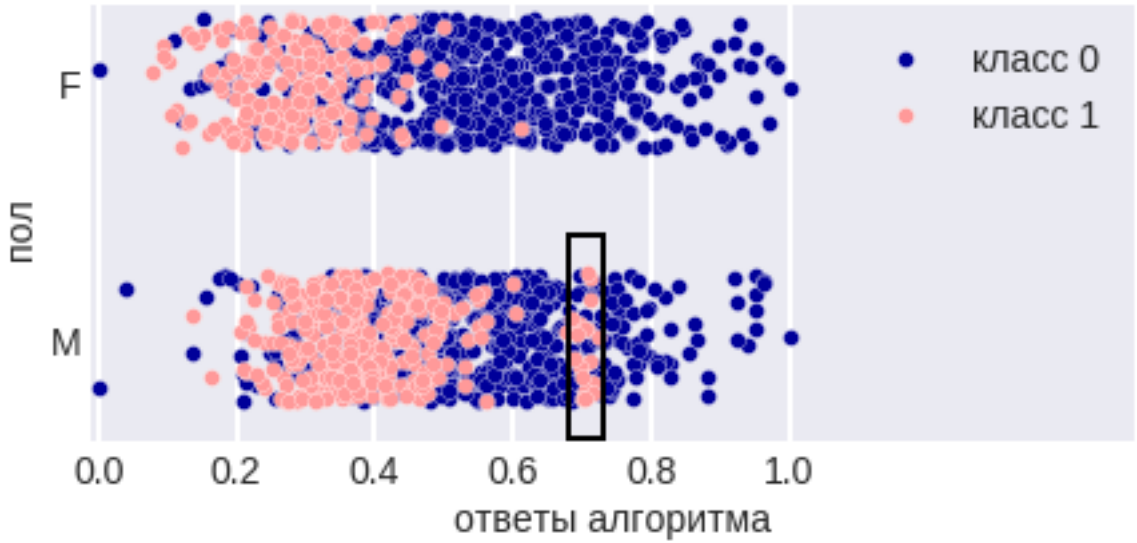


На контроле подозрительные линии…

Что это может значить?

Что делать?

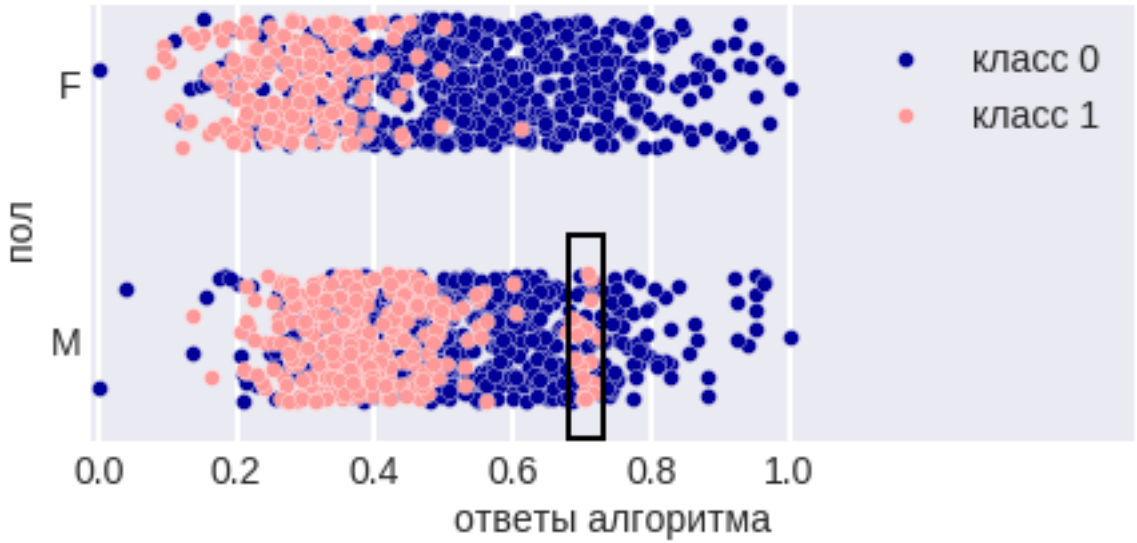
Ответы алгоритма – признак



Что видно?

Задача «~Analytics»

Ответы алгоритма – признак



Что видно:

- **зона неверных ответов (почему?)**
- **порог зависит от значения признака «пол»**

Но: распределения ответов зависит от контрольной выборки

Что надо проверить найдя закономерность?

Что надо проверить найдя закономерность?

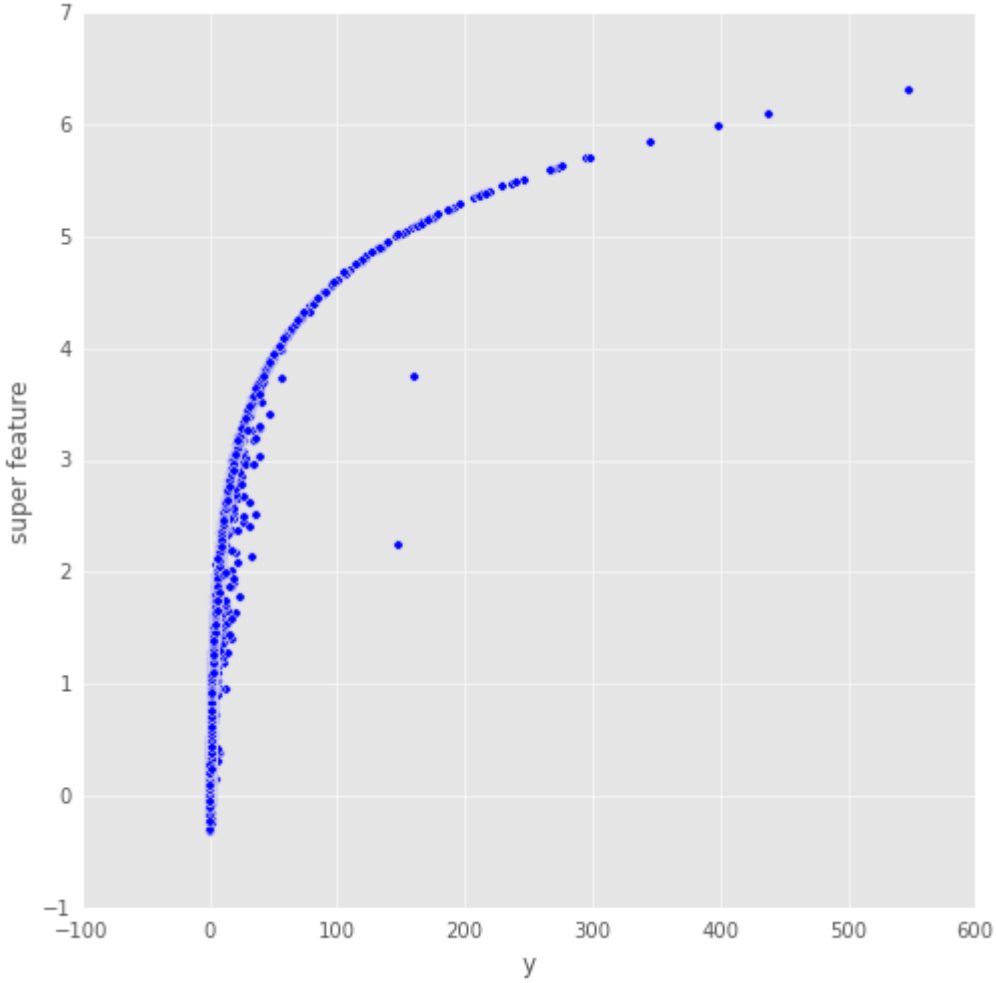
Что «контроль» ложится на обучение!

На практике нет гарантий одинаковости распределений гарантирует, даже если это гарантирует заказчик.

Примеры: рёбра в соцсети, заказы, разнесённые по времени (что-то приходится на праздники) и т.д.

Визуализация «алгоритм – признак»

Что сделать, чтобы картинка стала понятнее?



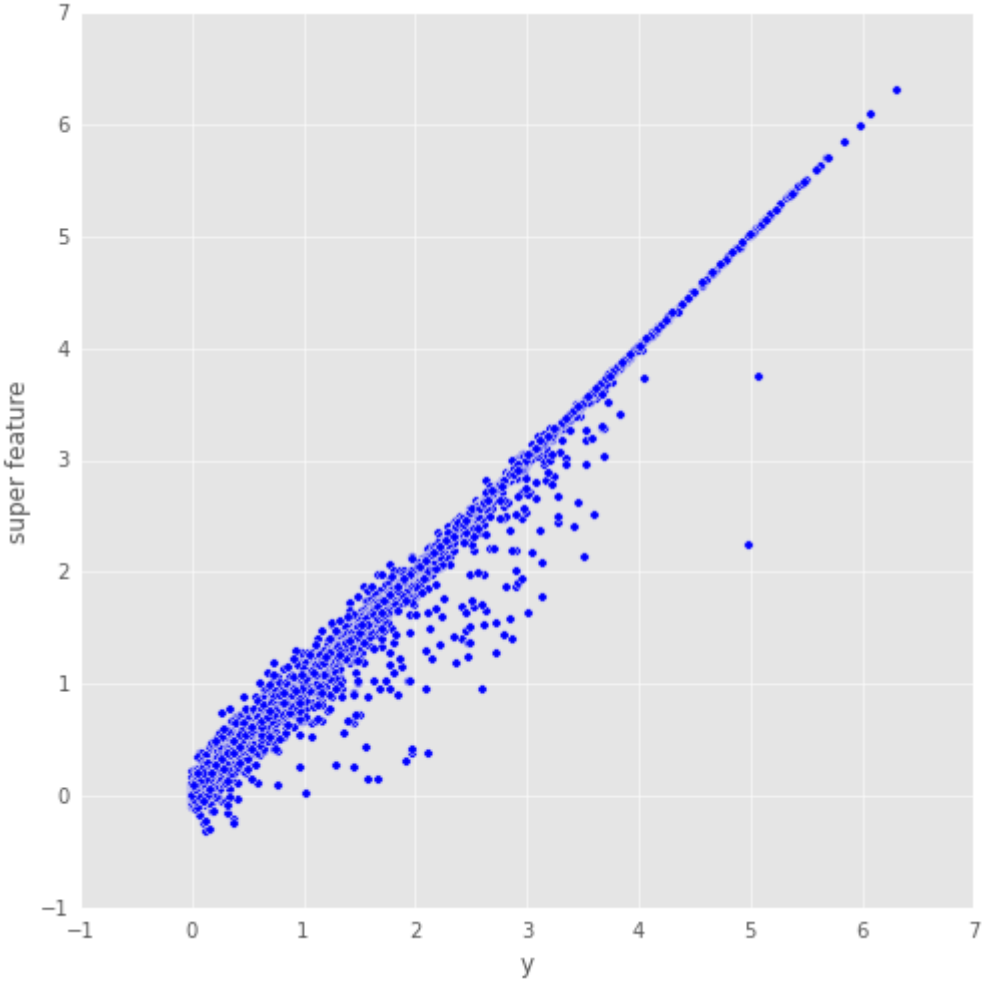
целевой признак и комбинация 2х признаков

Заметим, что эта комбинация строится как почти ответ…

```
plt.scatter((y2), np.log(train2.mnk.values) + train2.tmp.values)
```

Логарифмирование целевого признака

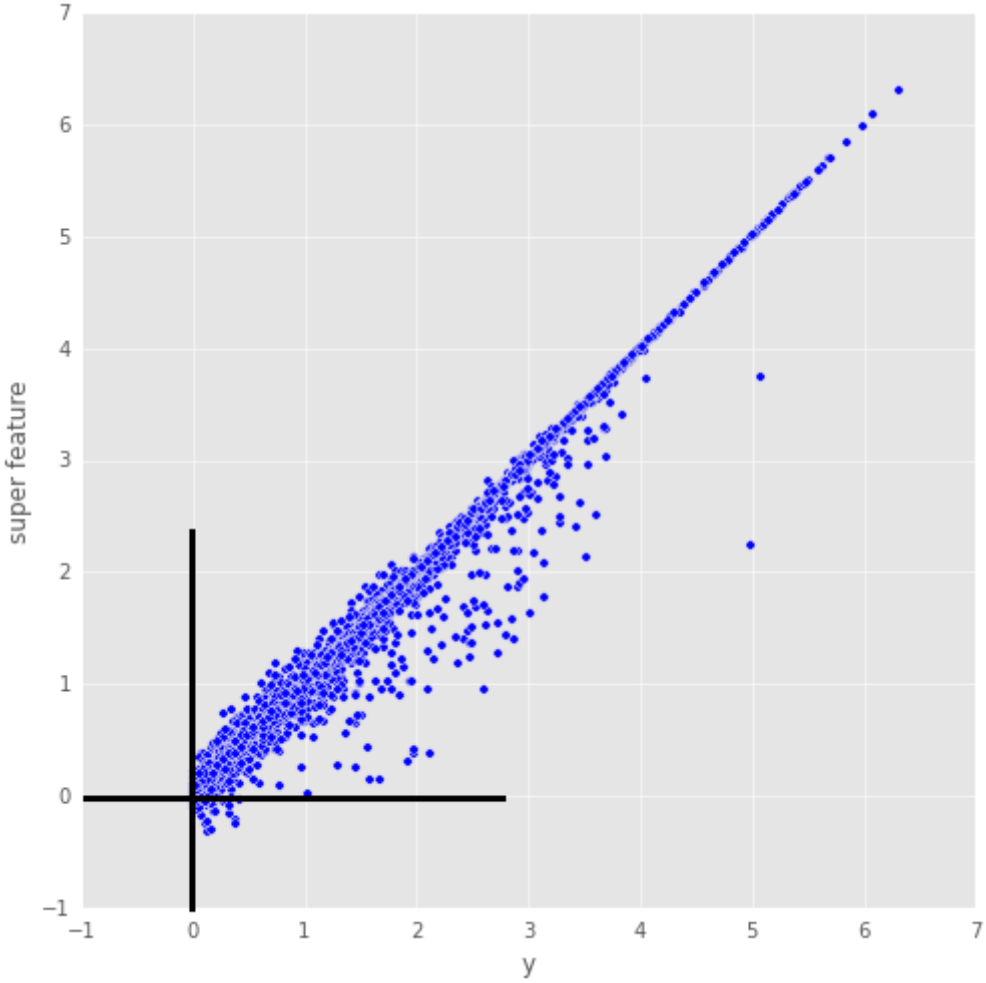
Что ещё сделать, чтобы картинка стала понятнее?



целевой признак и комбинация 2х признаков

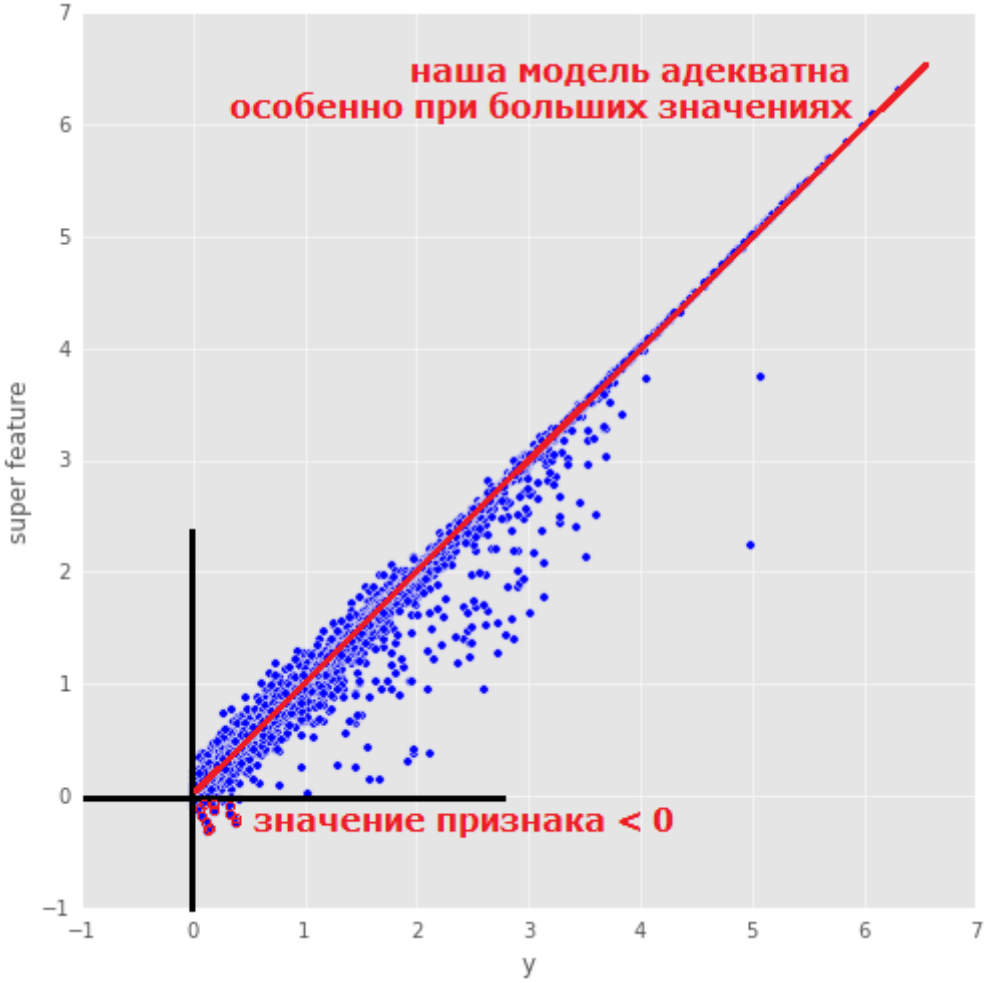
```
plt.scatter(np.log(y2), np.log(train2.mnk.values) + train2.tmp.values)
```

Логарифмирование целевого признака



Что видно на графике?

Логарифмирование целевого признака

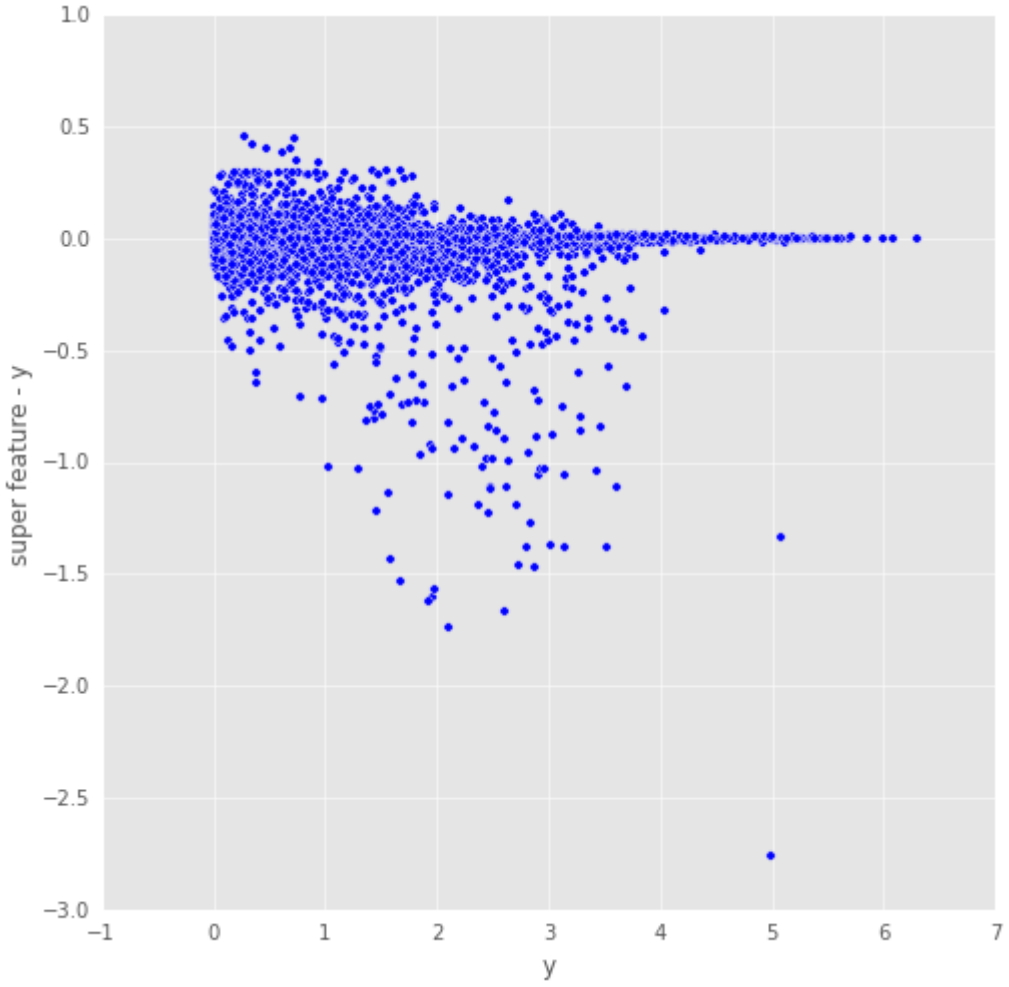


Правильный ответ всегда > 0

А наш супер-признак может принимать отрицательные значения!!!

Вывод: `maximum(f, 0)`

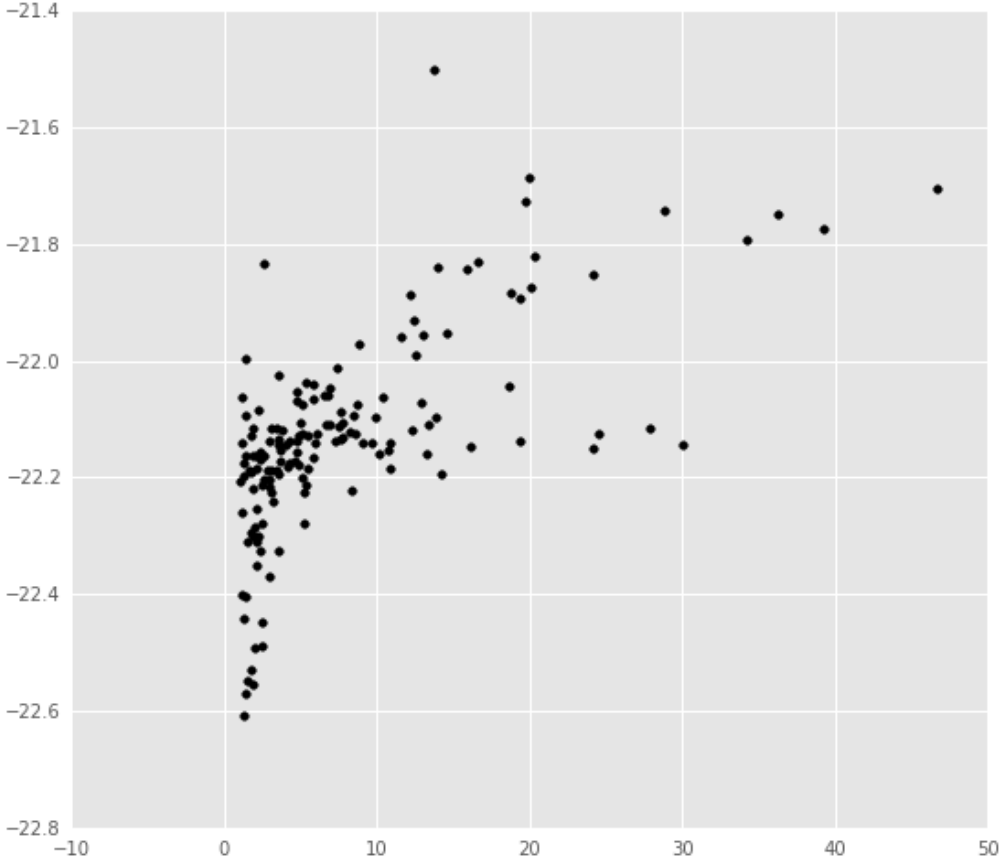
Разница признака и целевого признака



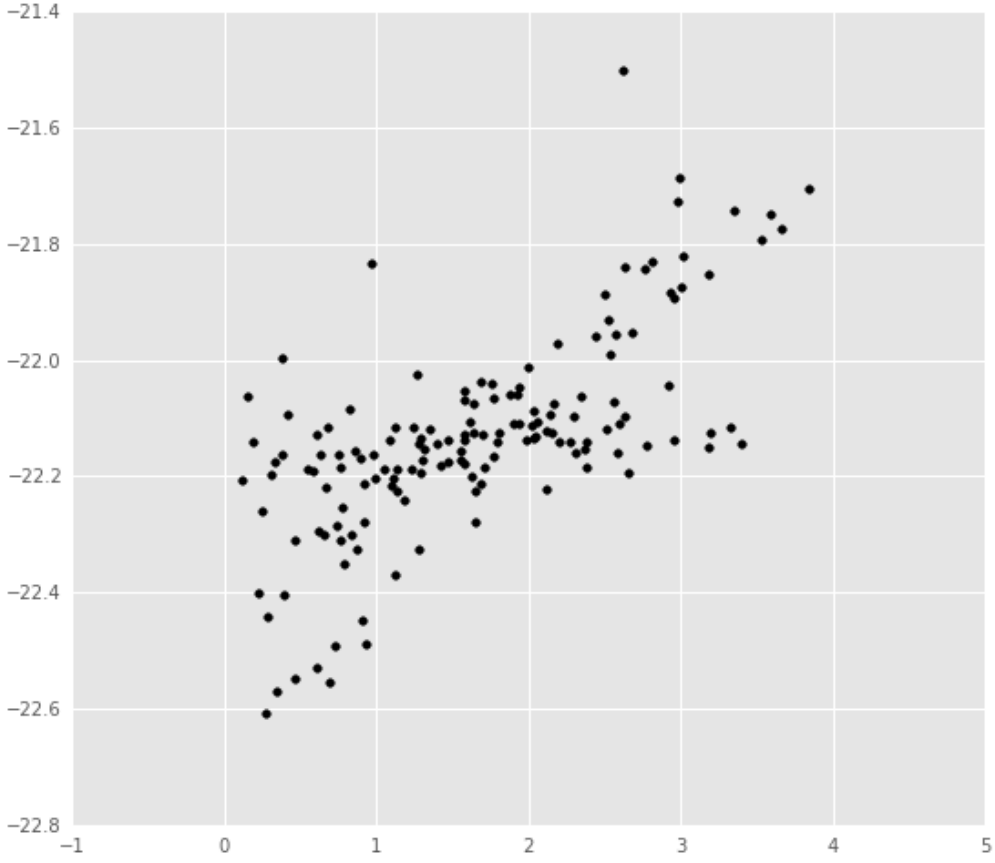
Если построили «почти ответ» – полезно посмотреть на ошибку

```
plt.scatter(np.log(y2), np.log(train2.mnk.values) + train2.tmp.values - np.log(y2))
```

Необходимость логарифмирования можно не заметить на меленьких выборках



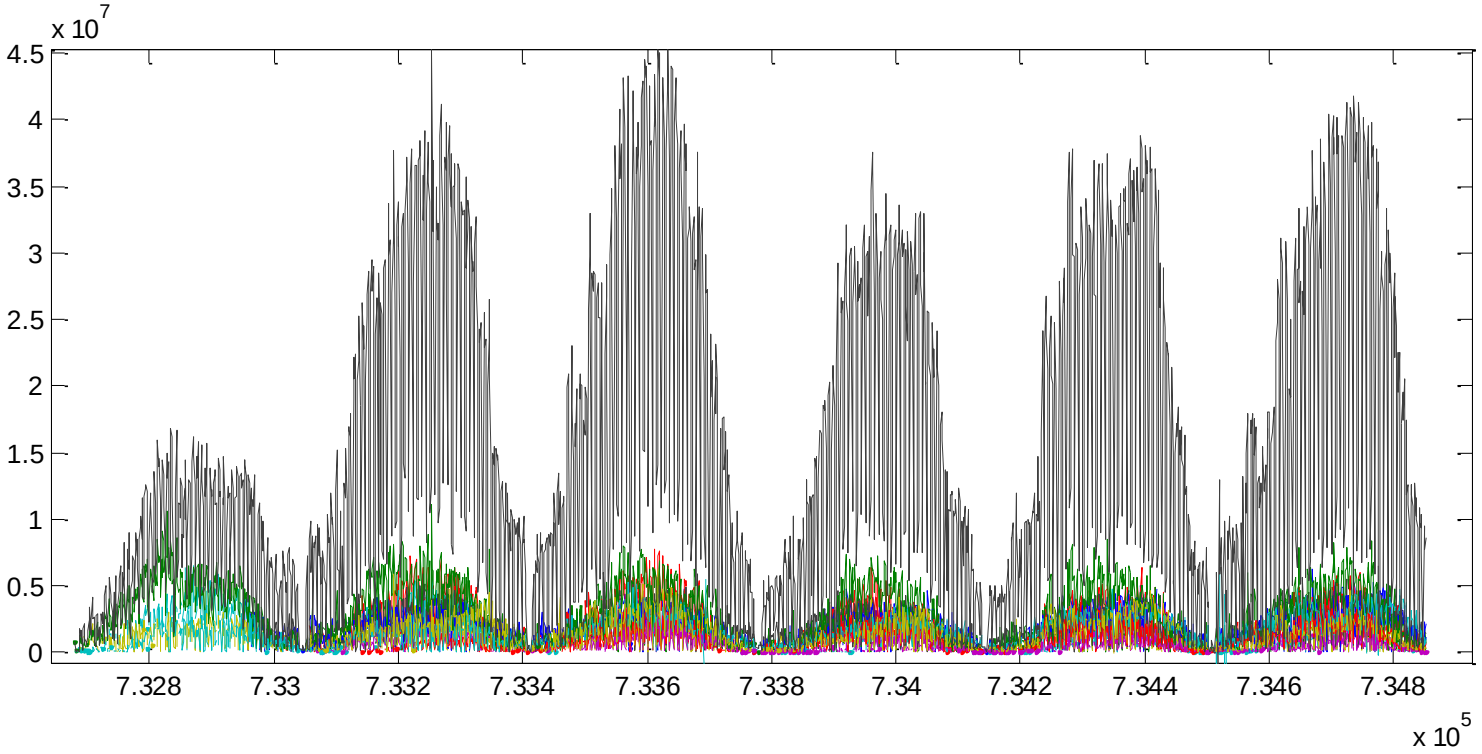
До логарифмирования



после

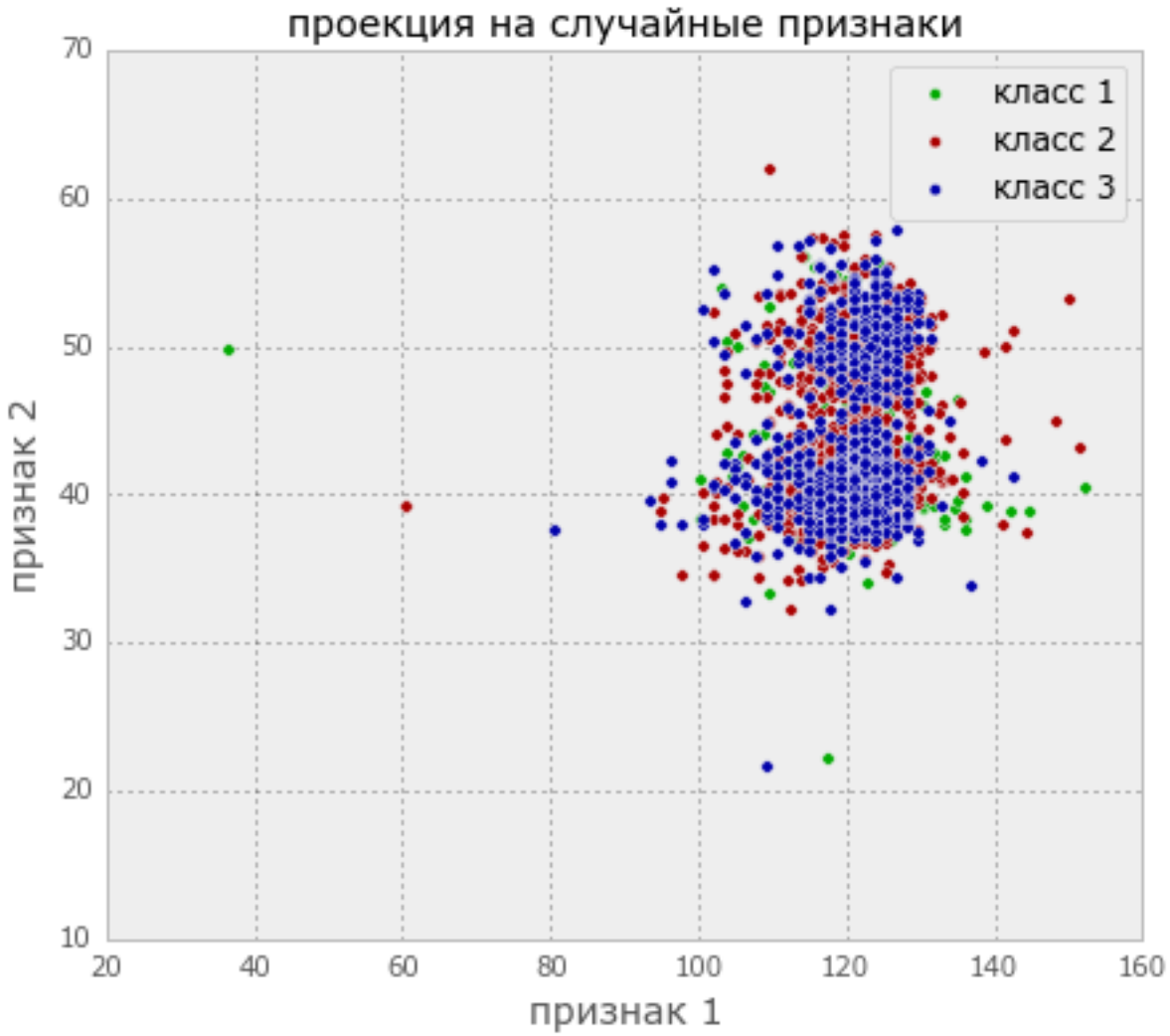
Агрегация (по дням недели)

прогнозирование временного ряда (продажи)

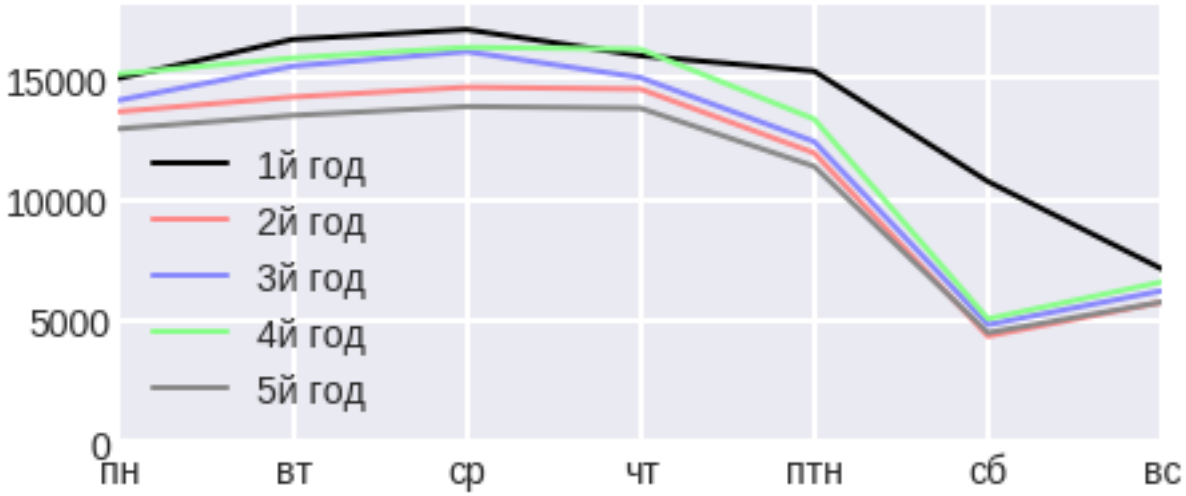


Есть отрицательные значения – выбросы вниз (!?).

Смотрим на пары признаков
если есть время
признаков немного
есть интересные сочетания



Агрегация (по дням недели)



Первый год нетипичен!

Остальные – очень похожи… осталось научиться прогнозировать «уровень недели».

Агрегация

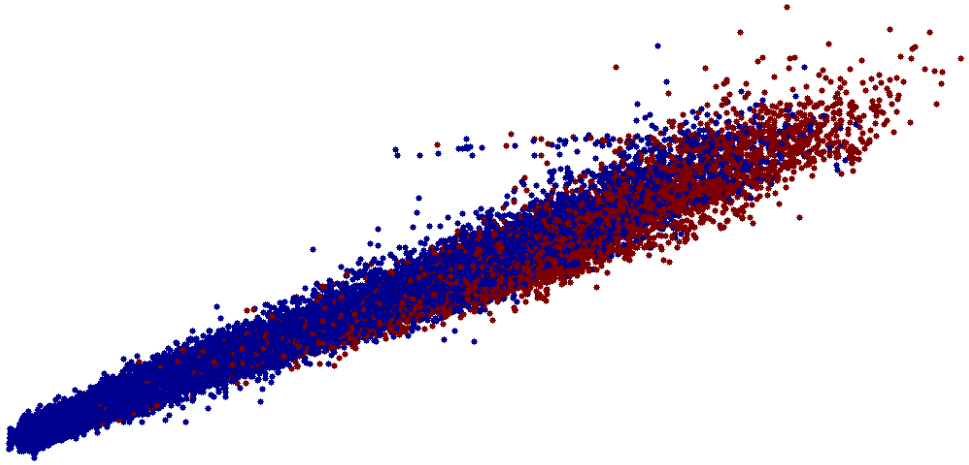
Типичная ошибка:
что агрегировать

- **все покупки (проблема оптовиков)**
- **средние покупки всех пользователей**

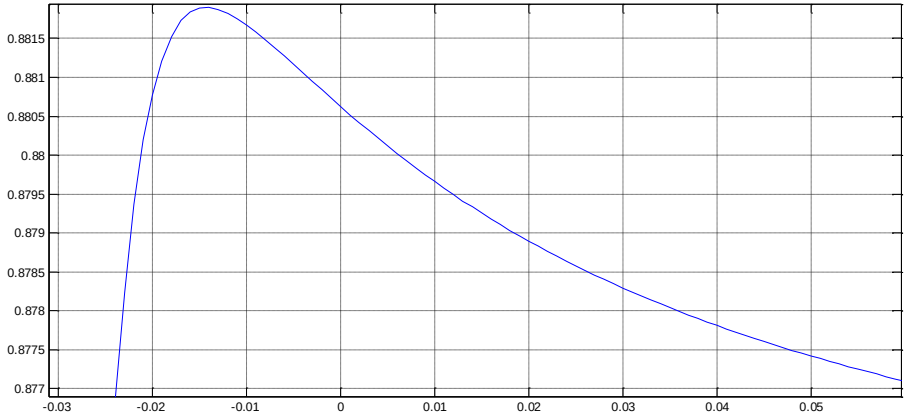
Прошлый год – задача Сбербанка
«мужские» / «женские» товары

Одномерная визуализация: качество алгоритма от параметра

Задача скоринга



Байес и (RF+GBM)



Коэффициент в линейной комбинации. Лучше вычитать!

Удивительно, но при визуализации:

- гладкость
- монотонность или унимодальность
- м.б. + явные выбросы

Если этого нет:

- ищем ошибку

3D-визуализации

Третий признак

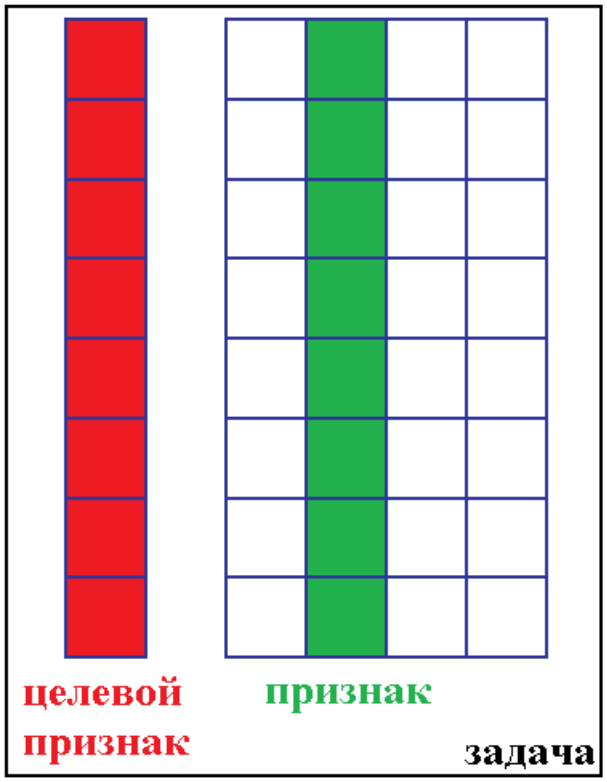
- **цвет**
- **размер**
- **форма**

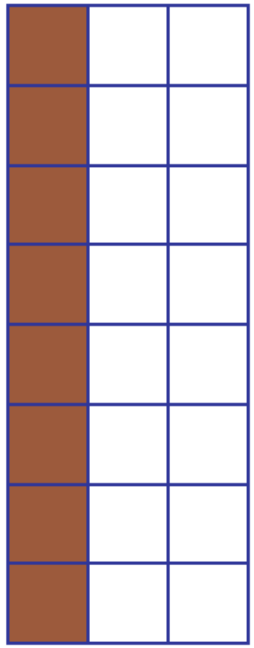
Практически не делают!

Иногда, если объектов мало и можно интерактивно вращать

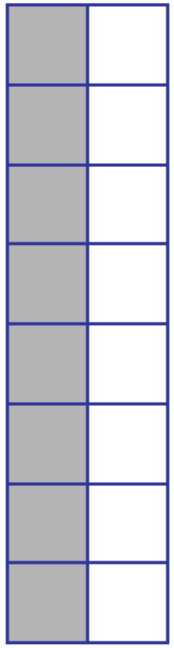
Что можно визуализировать:

«Всё вертикальное»





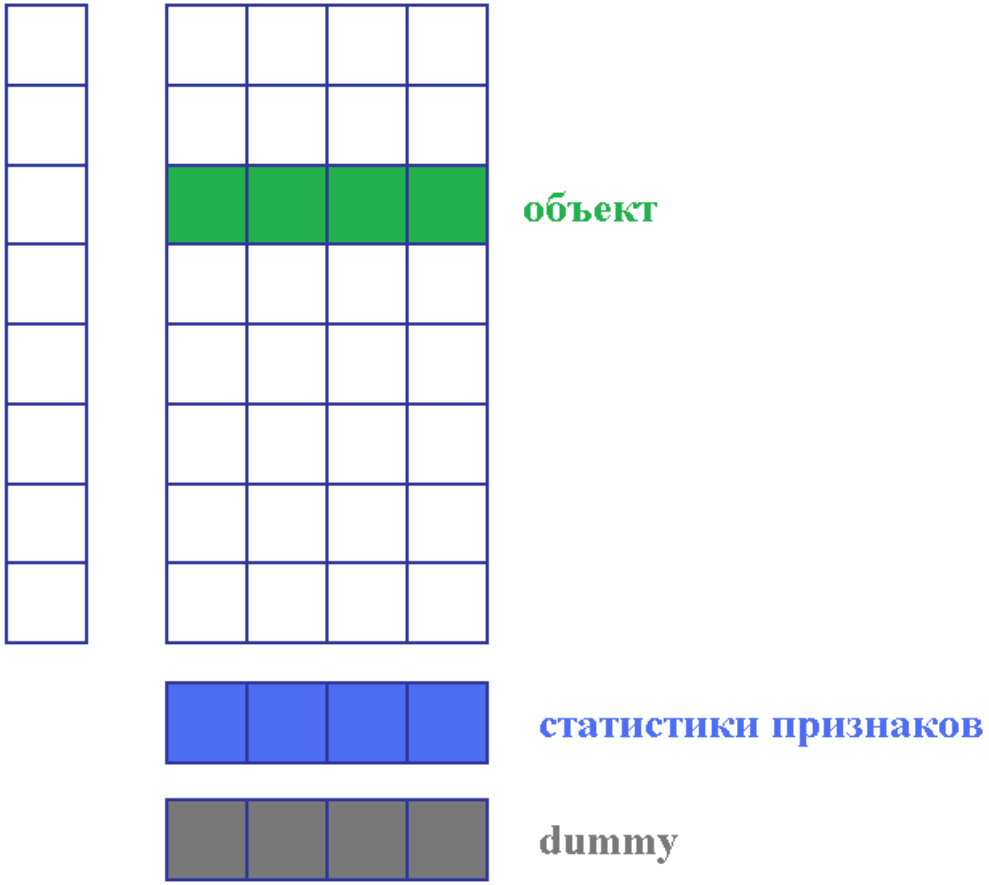
ответы алгоритма



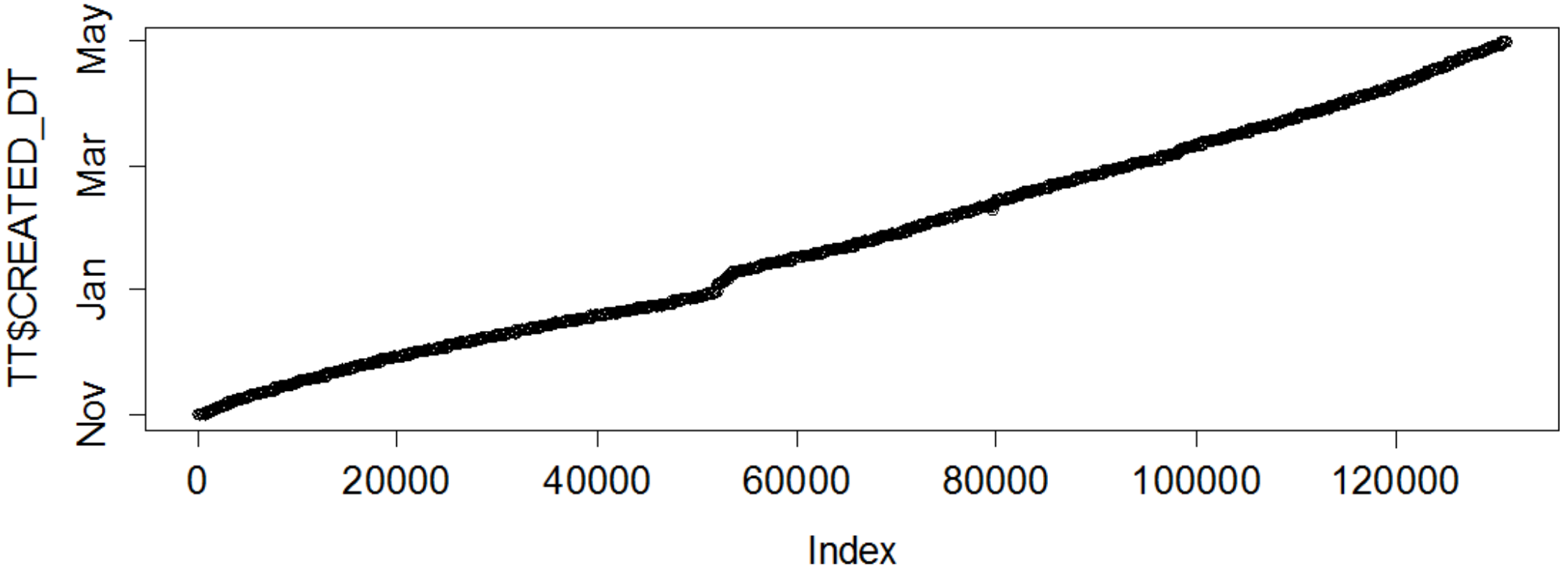
dummy-признаки (номера объектов, групп, rand)

Что можно визуализировать:

«Всё горизонтальное» (реже)



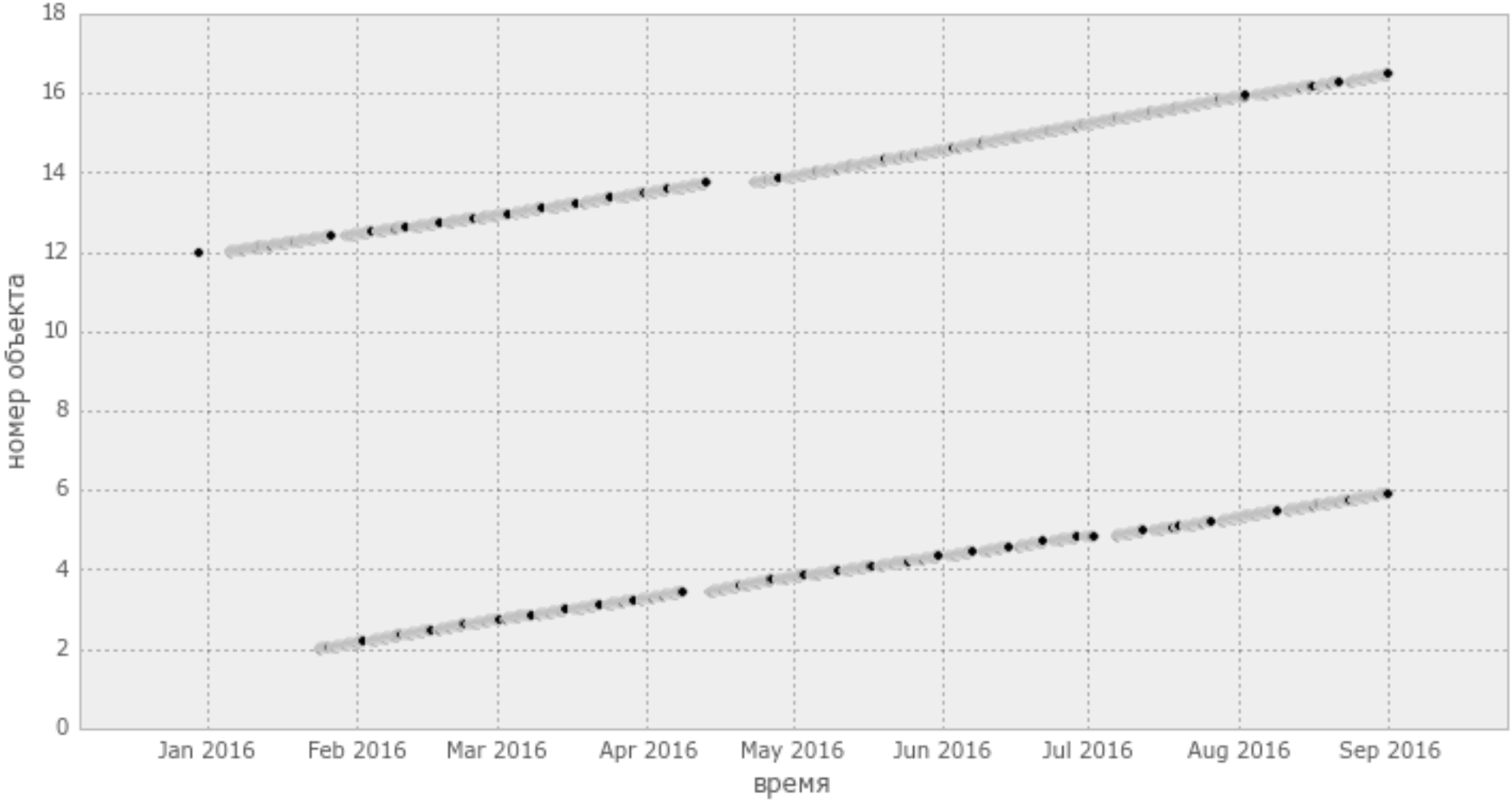
Пример dummy-визуализации



Сделайте график «id – время»:

- простая проверка на монотонность
- видны «подозрительные периоды»

Пример dummy-визуализации



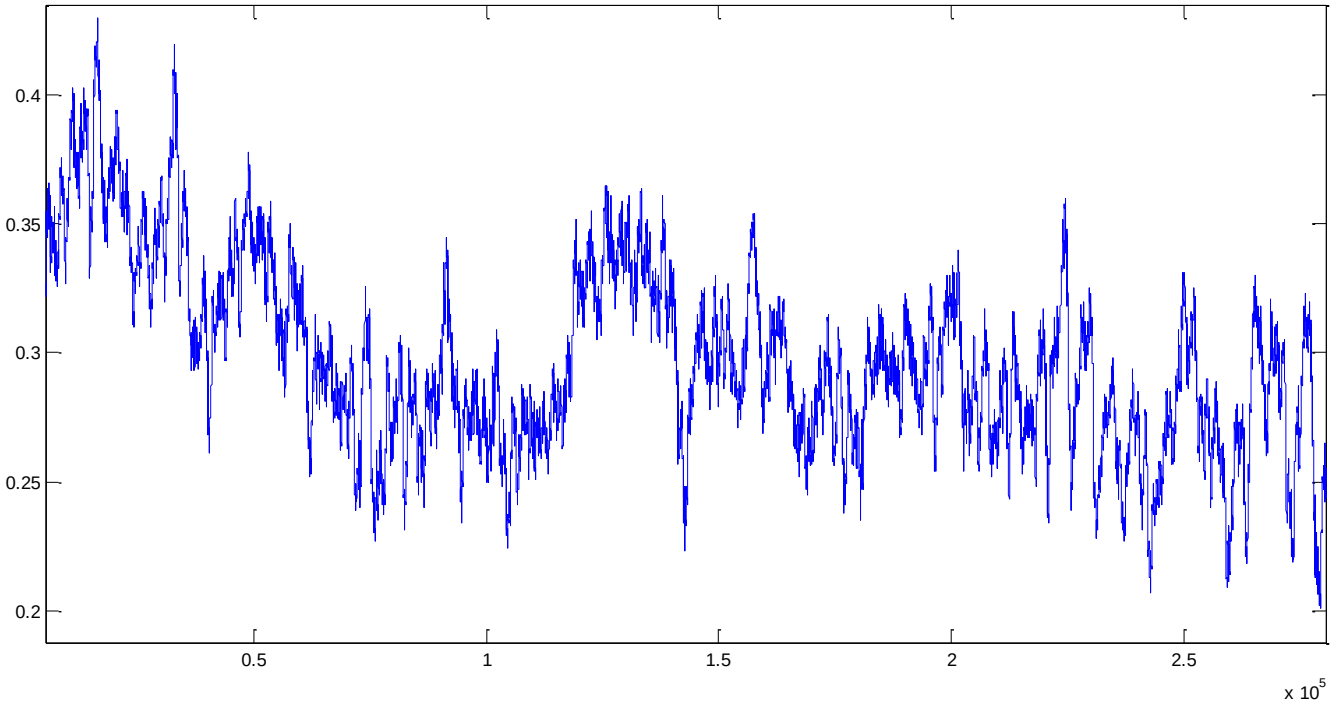
Случай из жизни: время – номер объекта

Видна двойная нумерация, периоды непоявления объектов

При раскраске по другим признакам видно больше!

Пример dummy-визуализации

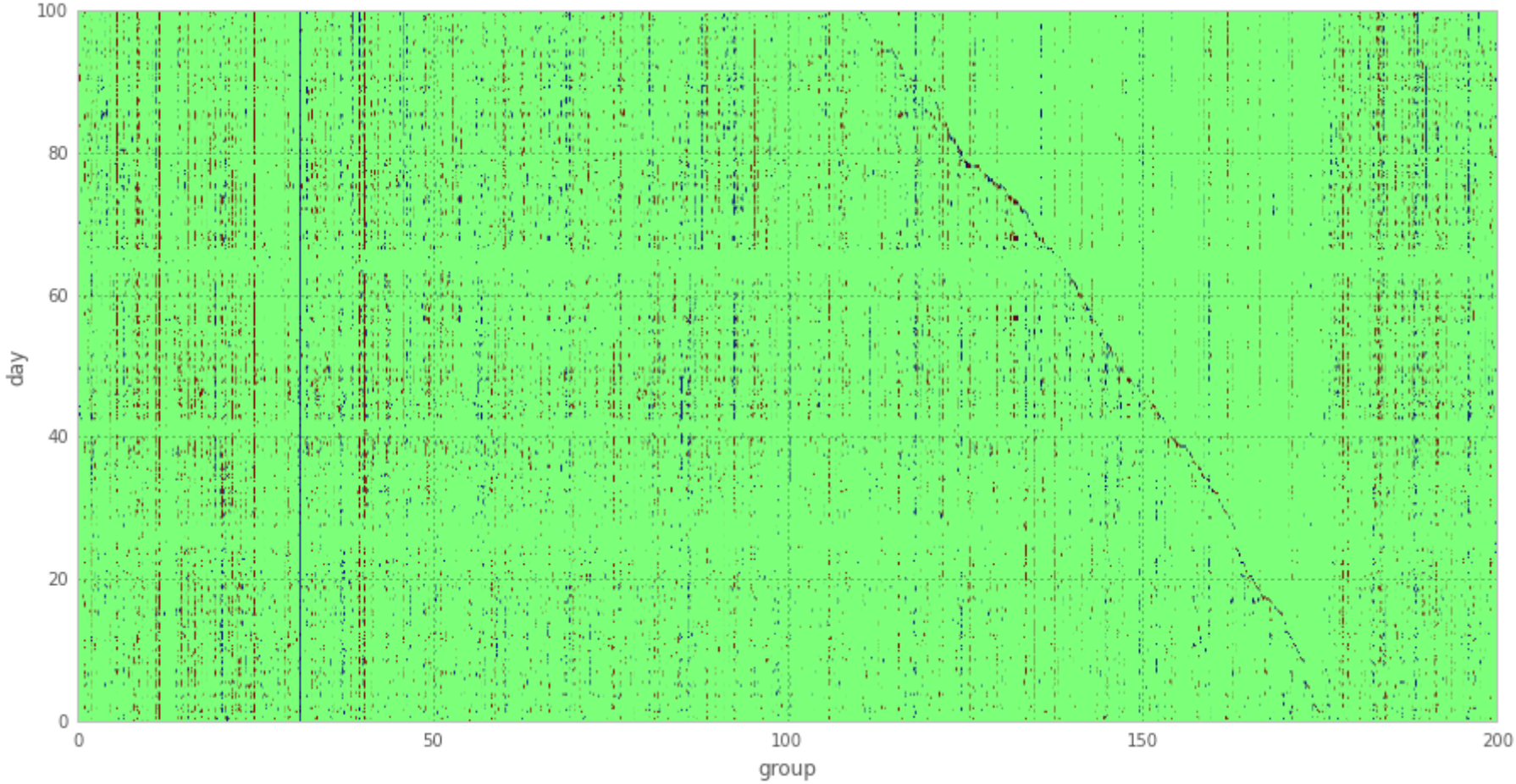
Как меняется цель со временем



Применяется сглаживание окном

Некоторые кейсы

Визуализация данных (RedHat) КУДА



по горизонтали – разные группы,

по вертикали – дни (подряд),

салатовый цвет – нет взаимодействия,

красный / синий – класс 1 / 0

Что за подозрительная полоса?

Визуализация данных (RedHat)

Группы упорядочены так:

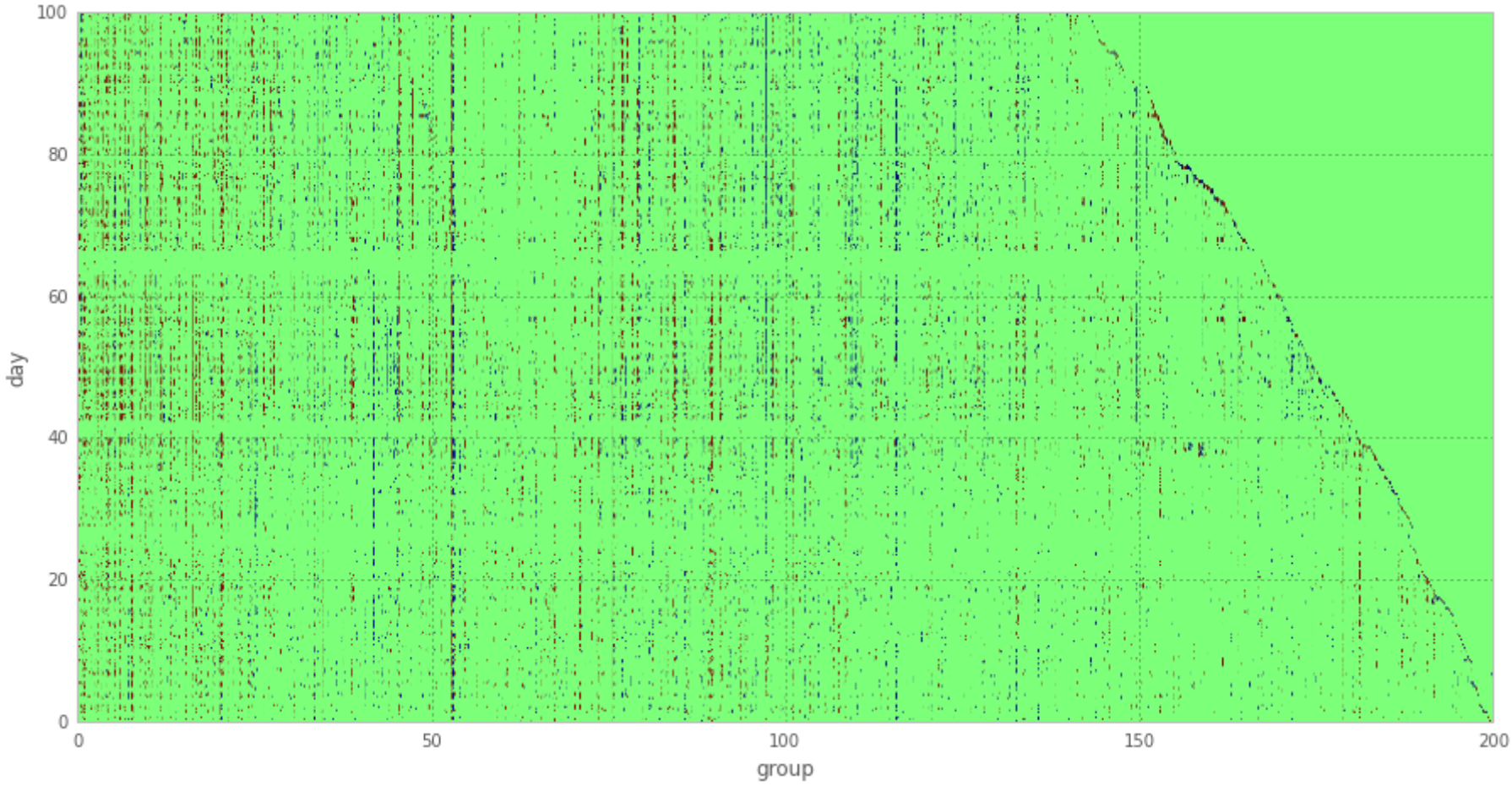
```
group_date2.columns[:10]

'group 1000', 'group 10006', 'group 1001', 'group 1002', 'group
10021', 'group 10025', 'group 10032', 'group 10036', 'group 1004',
```

это лексикографический порядок!

Теперь сделаем в обычном порядке…

```
data_train.group_1 = data_train.group_1.map(lambda x: int(x[6:]))
```

Визуализация данных (RedHat)

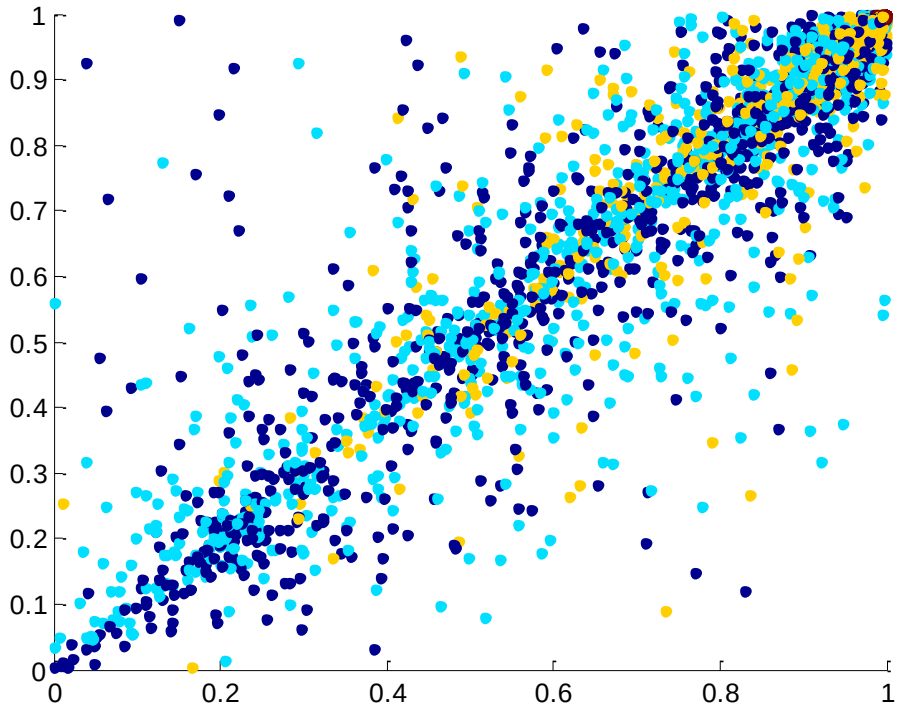


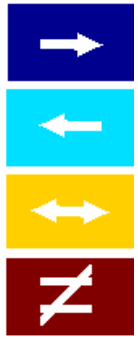
теперь понятнее… группы, видимо, идут в порядке появления последние – которые добавлялись в дни сбора выборки

Задача «Причина-следствие»

Метод: «ручная деформация пространств»

```
% метод, основанный на полиномиальном приближении
[f fn] = cause_f_polyfit(Xs);
scatter(f(:,1), f(:,2), 20, Ys(:,2), 'filled')
```

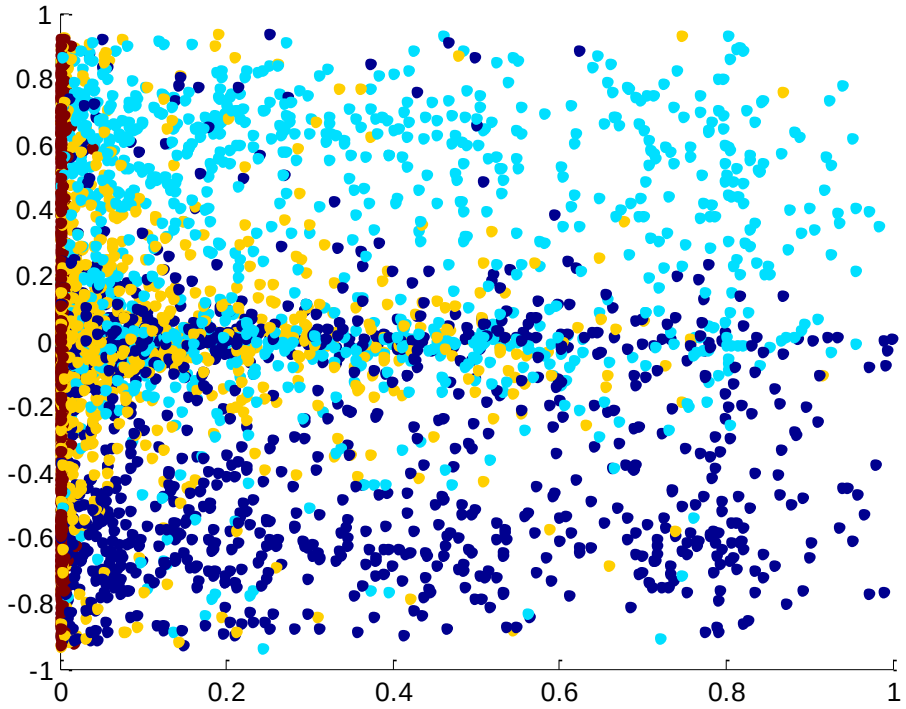




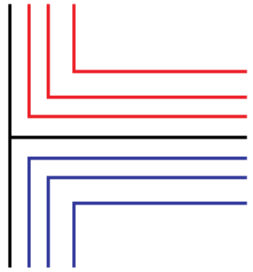
Кстати: хорошая задача – пример «новой науки»

Алгебраические выражения над признаками

```
scatter(1-0.5*(f(:,1)+f(:,2)),fn21(:,1)-fn21(:,2), 20, Ys(:,2), 'filled')
```

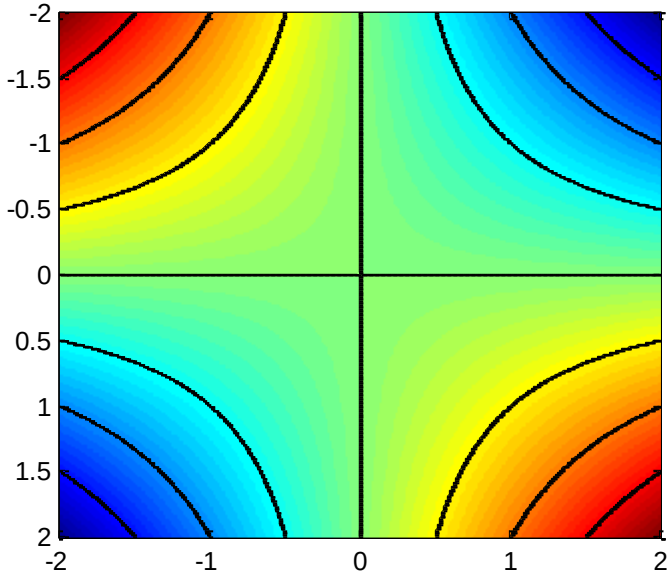


А теперь надо «уголками откусывать классы»:

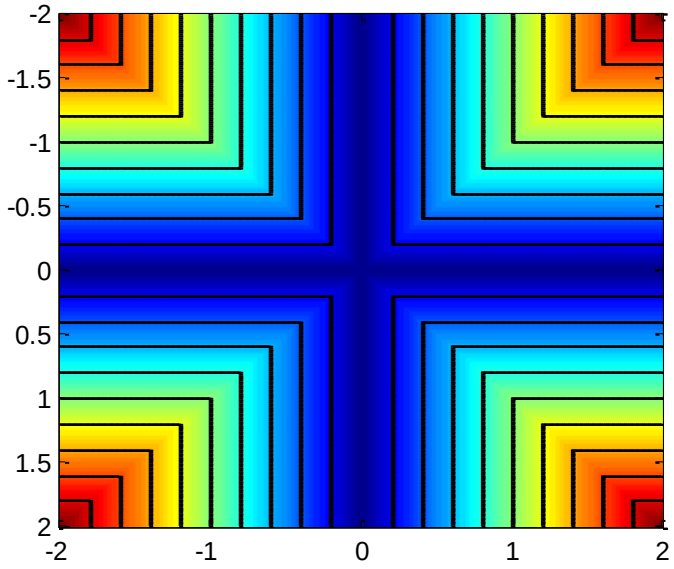


Какие функции «откусывают уголками»

$$z = y \cdot x$$

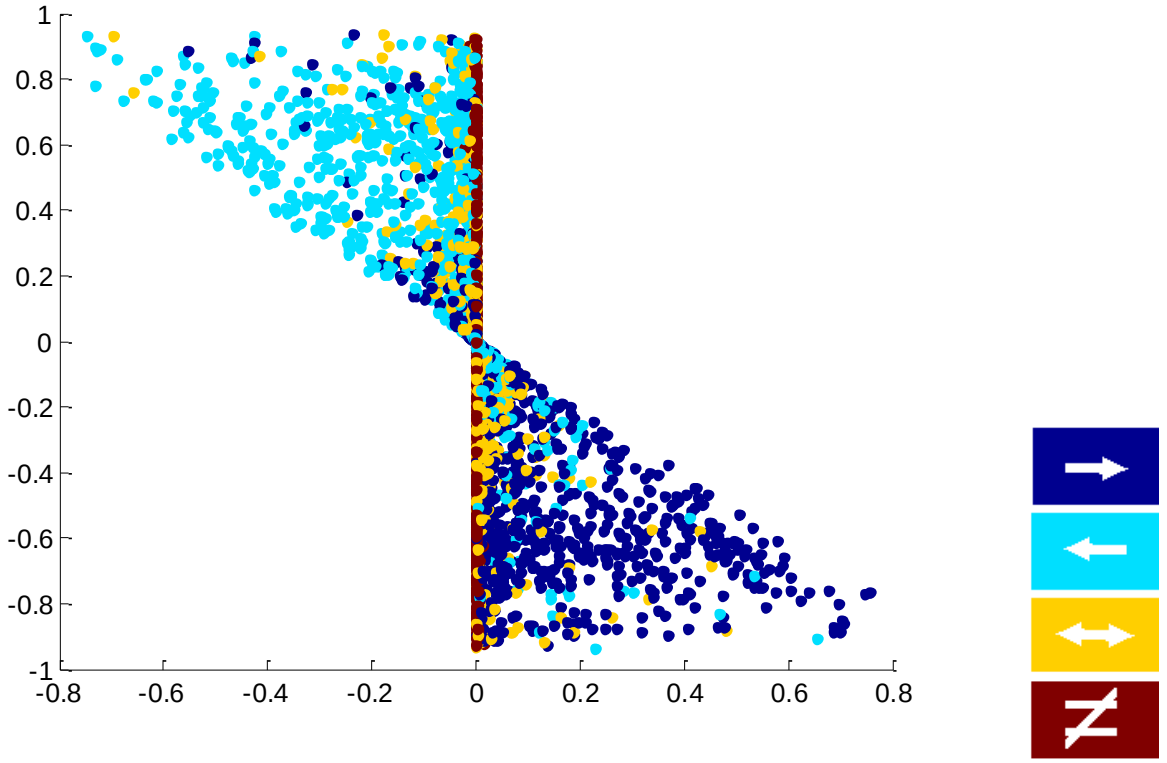


$$z = \min(|y|,|x|)$$



Алгебраические выражения над признаками

```
a = -(1-0.5*(f(:,1)+f(:,2))).*(fn21(:,1)-fn21(:,2))
scatter(a,fn21(:,1)-fn21(:,2), 20, Ys(:,2), 'filled')
```

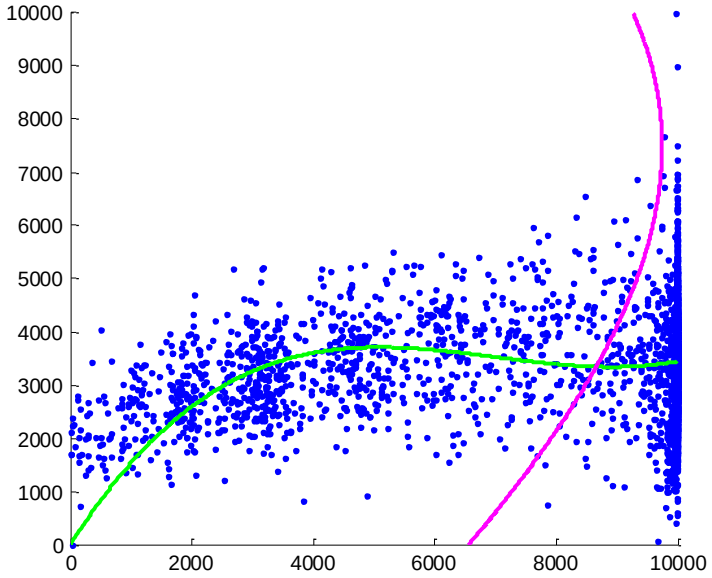


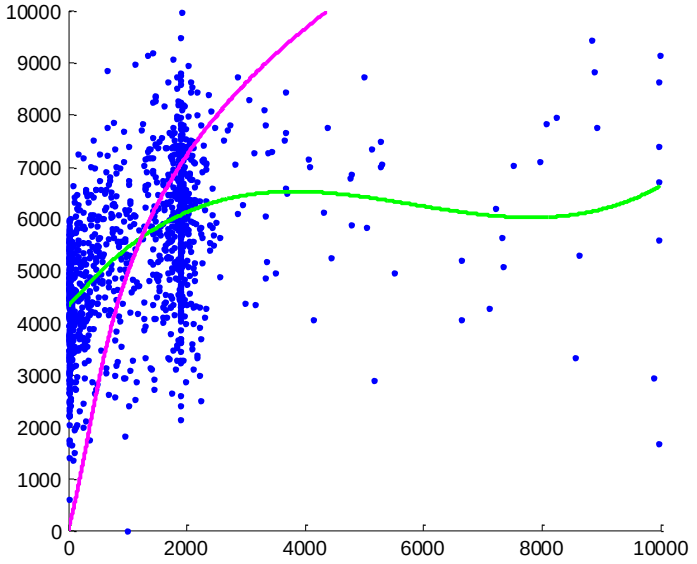
И здесь мы видим разделяемость синих и голубых!

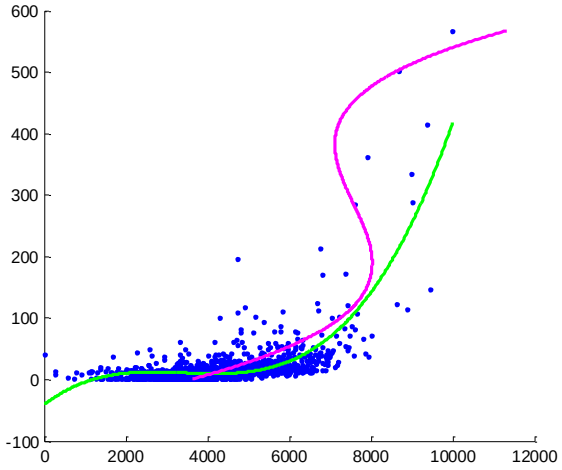
Получается алгоритм неплохого качества.

Ещё один приём: посмотреть как метод «работает»

Полиномиальная регрессии (deg=3) сразу от 2х переменных…

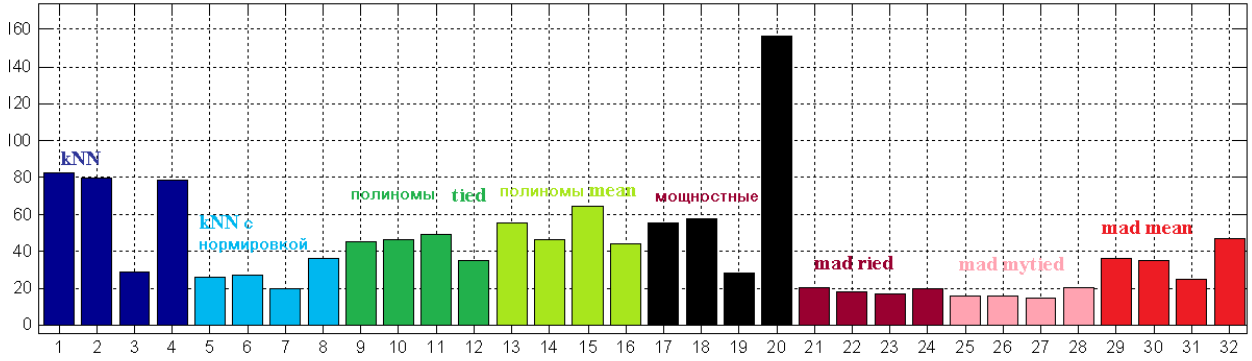




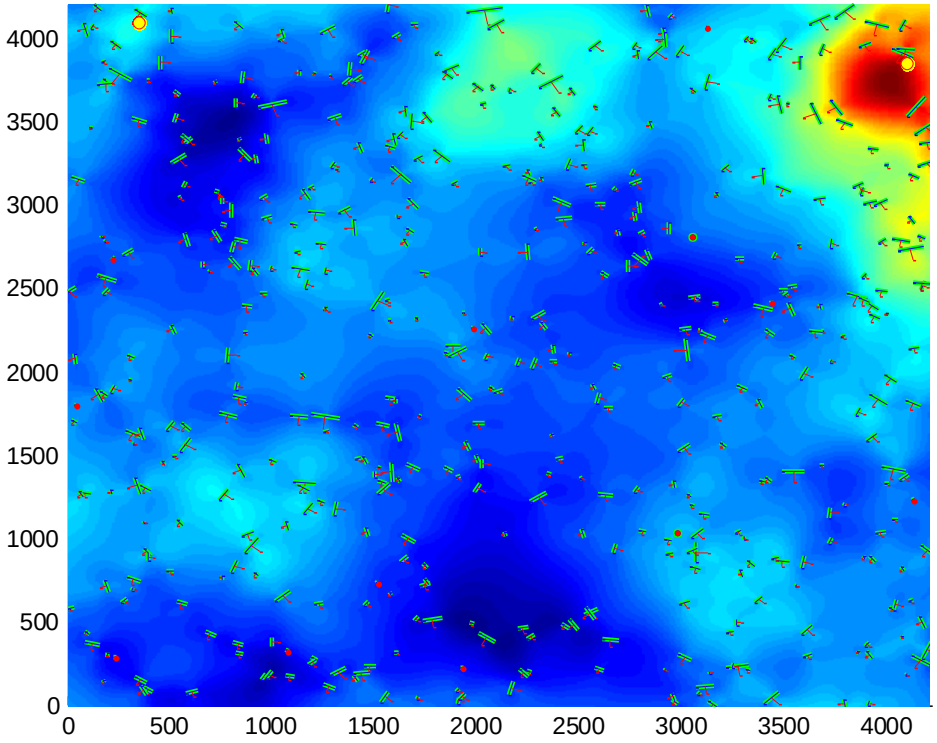


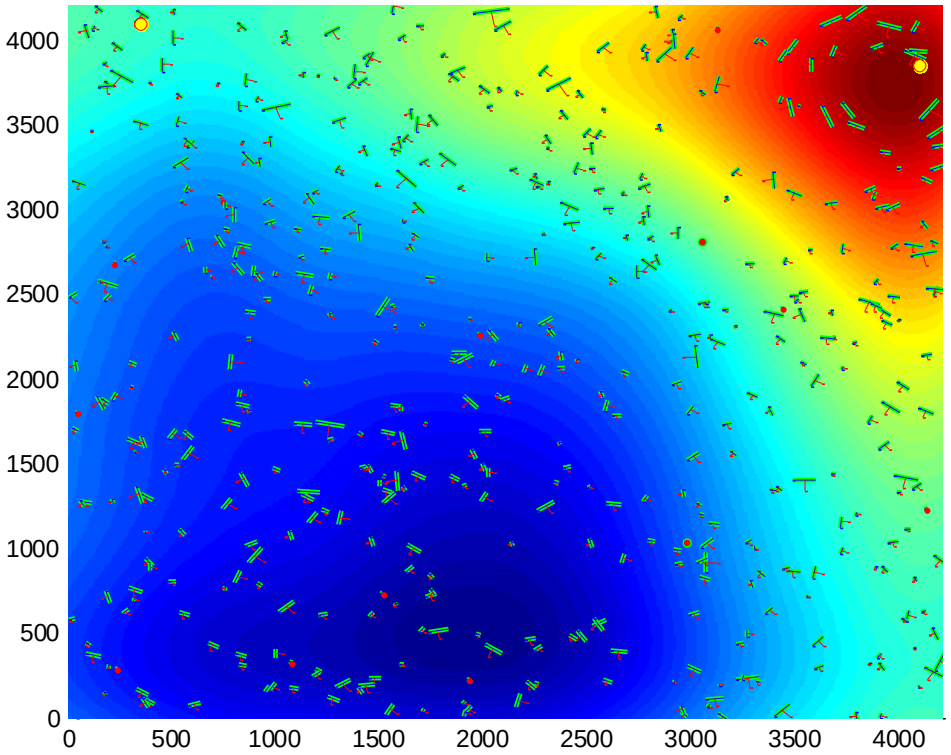
Ответы алгоритмов – как признаки

Построено несколько методов –
их ответы как признаки,
потом с помощью RF «качество алгоритмов».



Задача про чёрные дыры



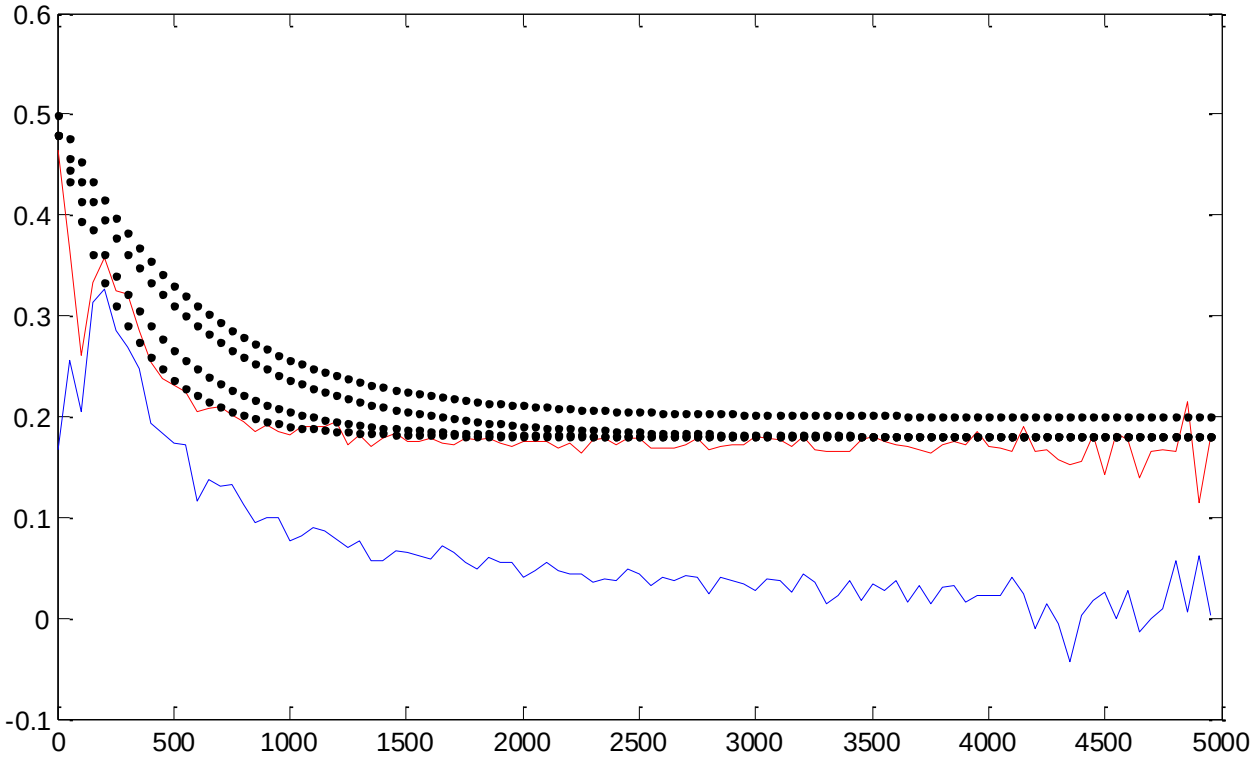


Какая связь между рисунками?

Ответ:

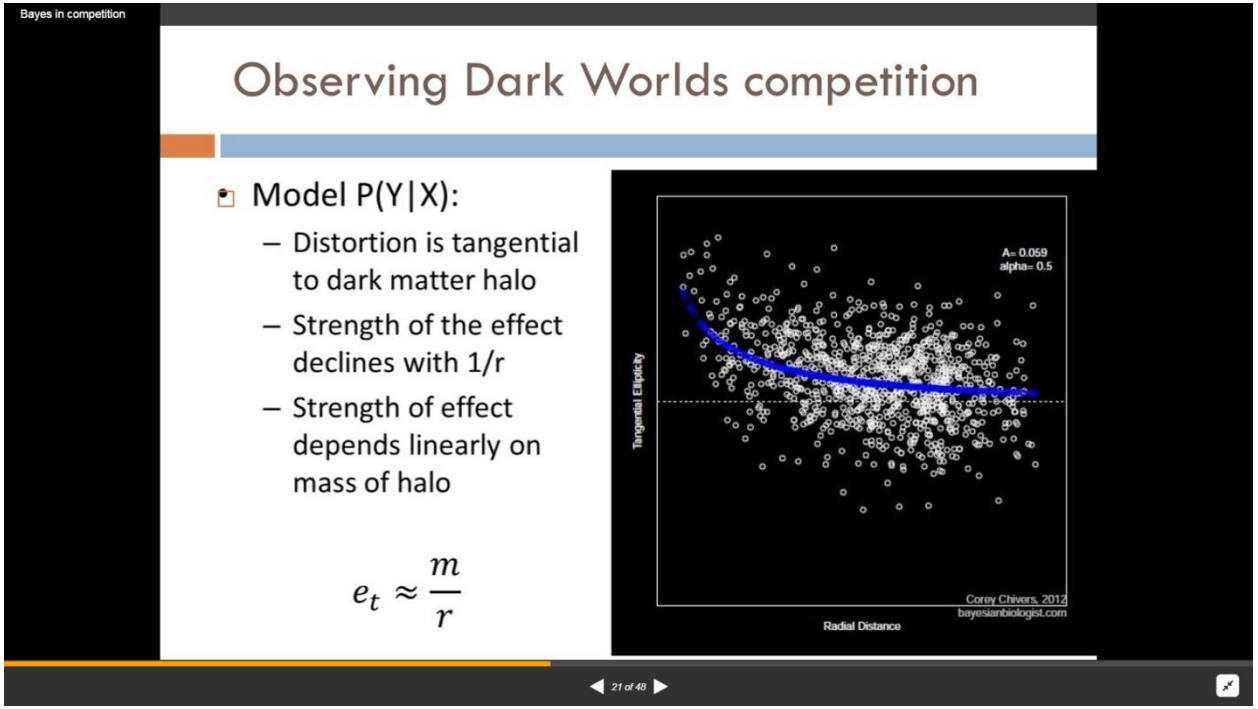
«Плотность» и её сглаженный аналог.

Средний профиль плотности(красный):



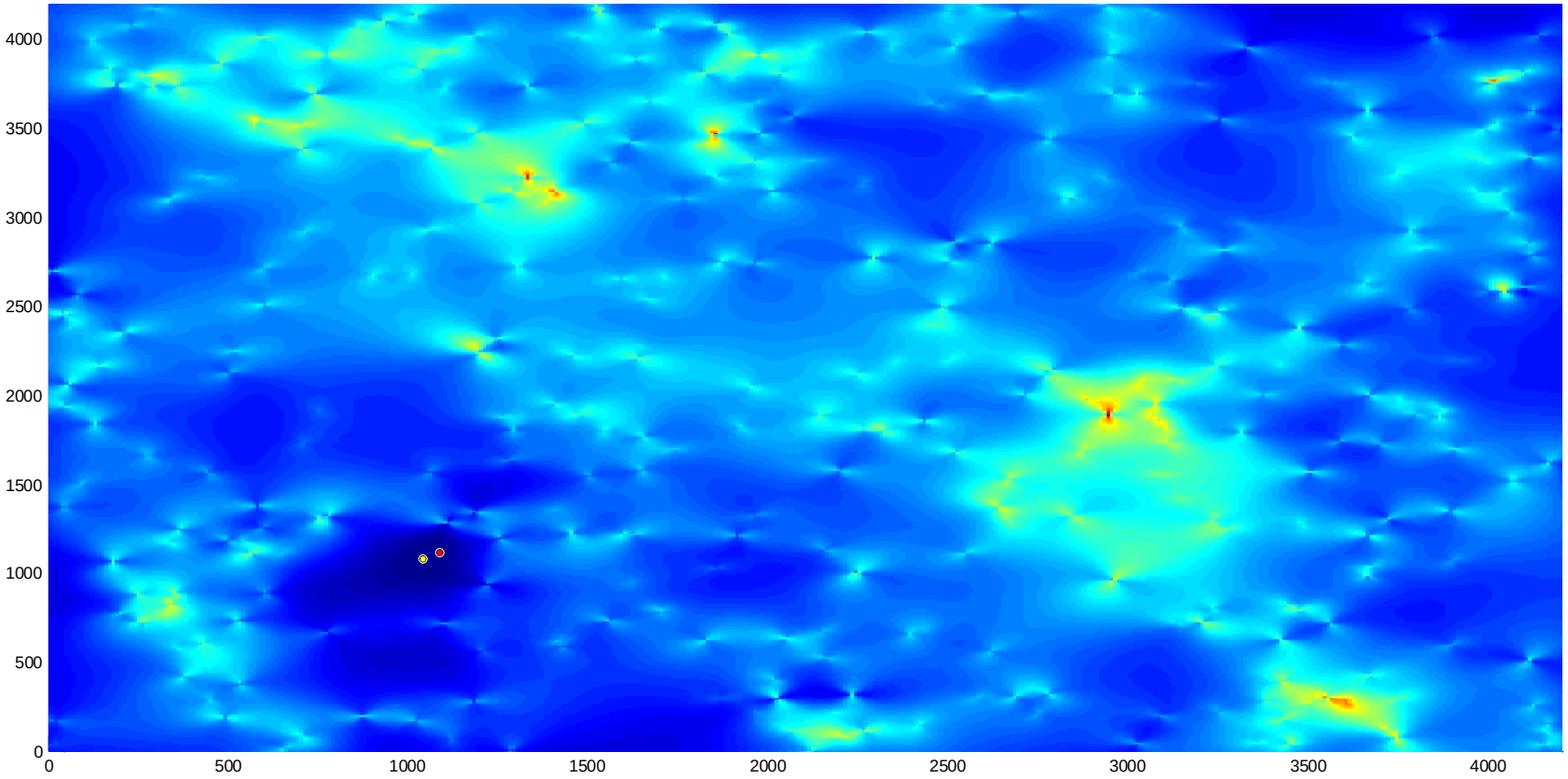
и методы его приближения

Owen Zhang



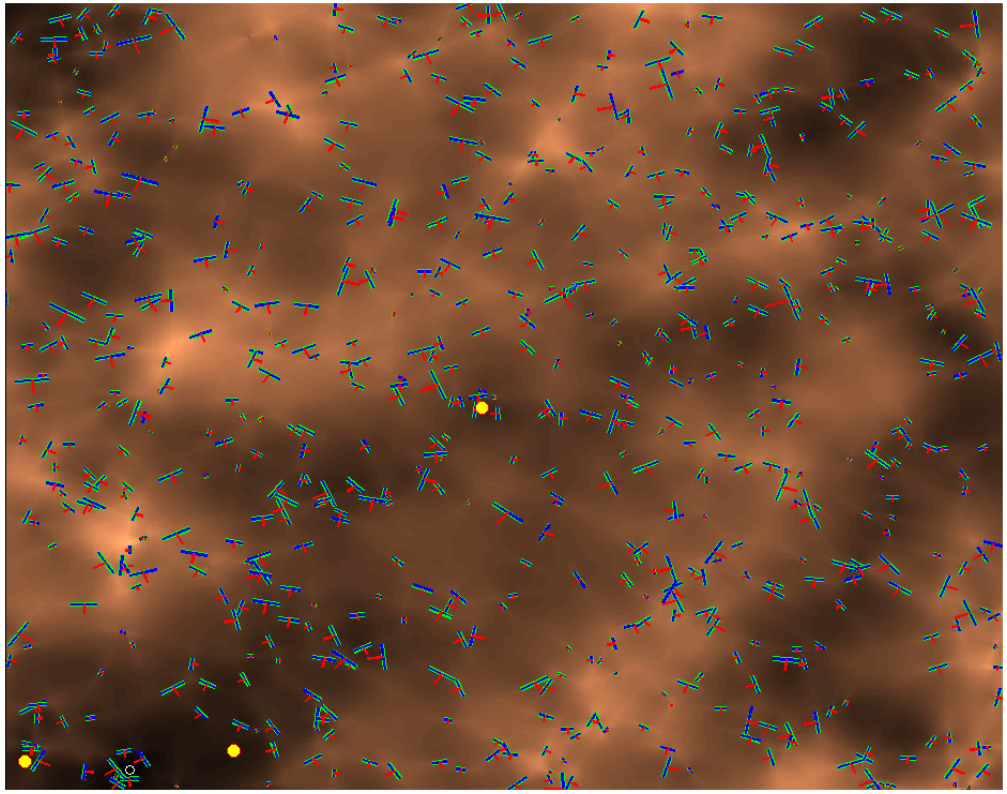
Также использовал визуализацию для создания модели

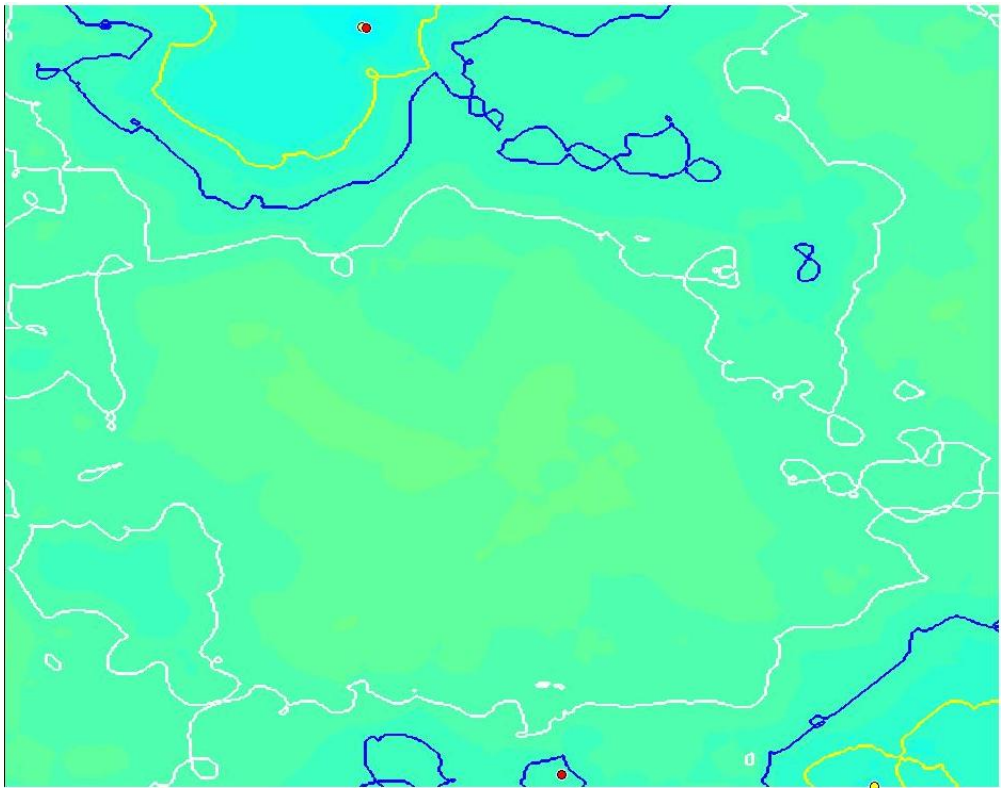
Другой способ:



разумно решать комбинацией двух

Трудности большого числа дыр:

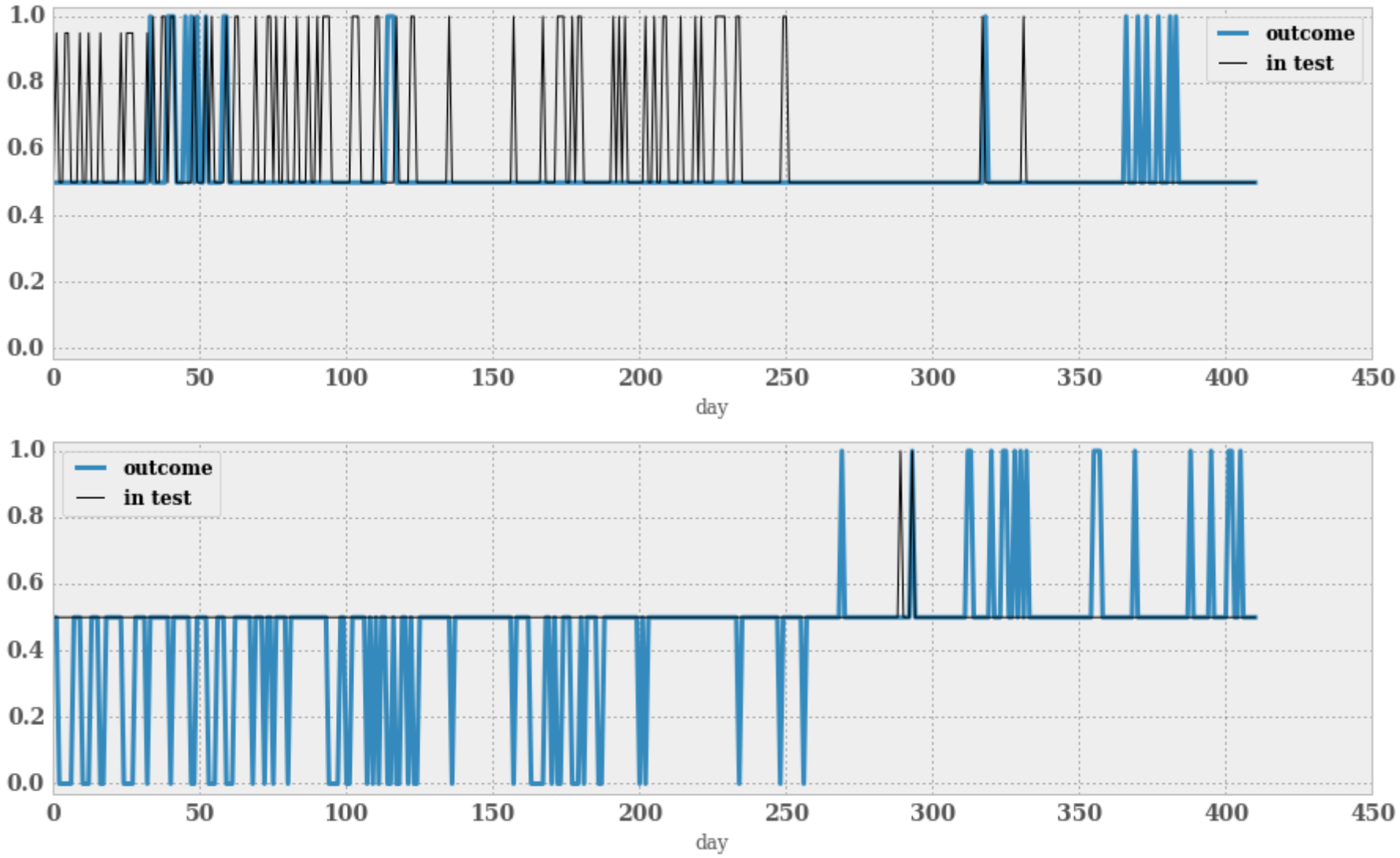




переход к линиям уровня

Главное – выбор эффективной визуализации.

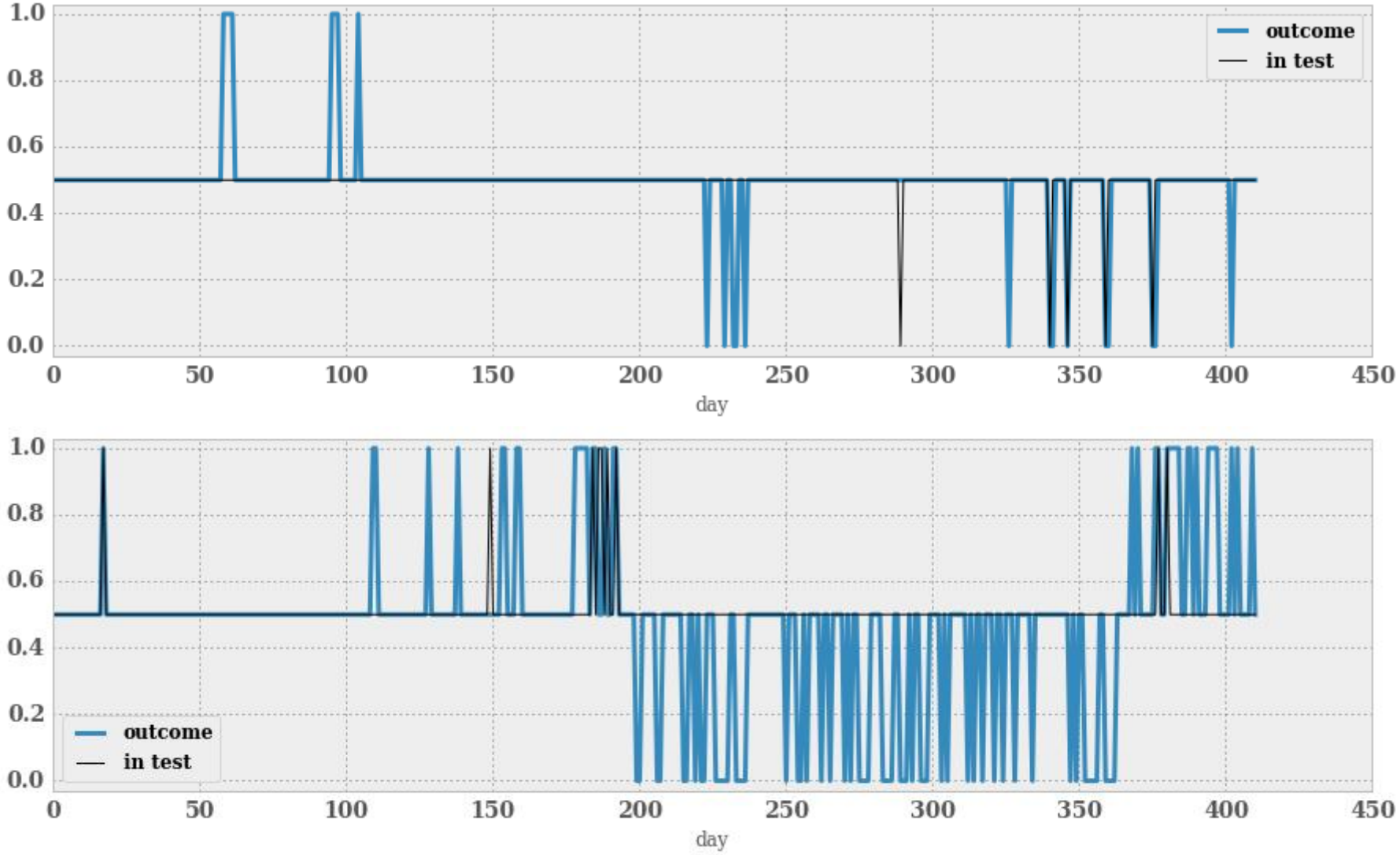
Задача «RedHat»



Как ведут себя представители групп по дням

Каждый график – для отдельной группы

Задача «RedHat»



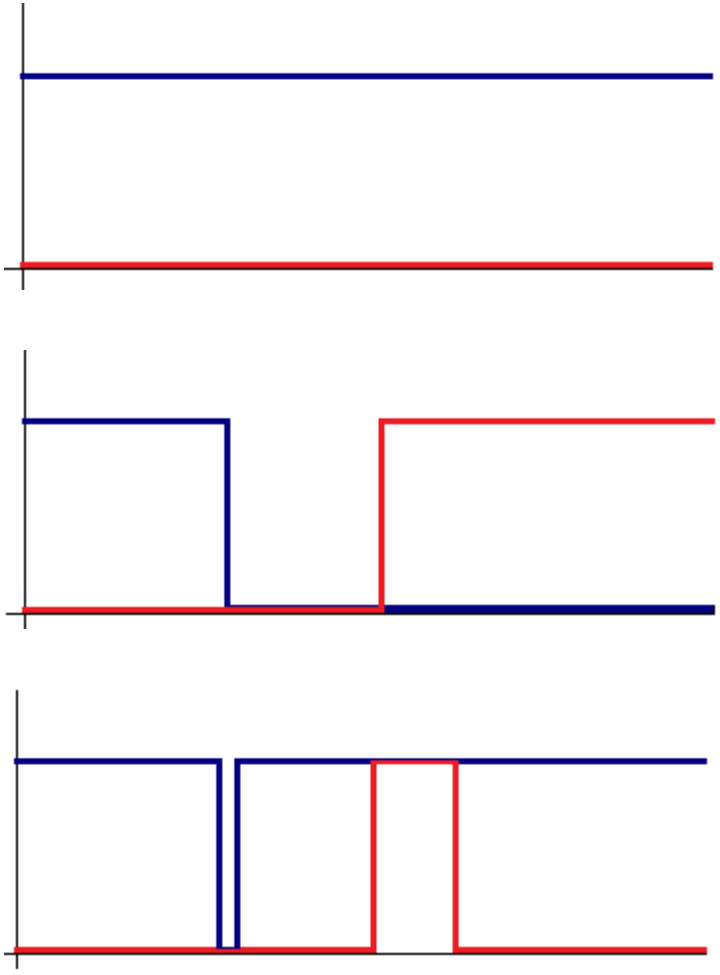
Как ведут себя представители групп по дням

Каждый график – для отдельной группы

Задача «RedHat»

Что видим?

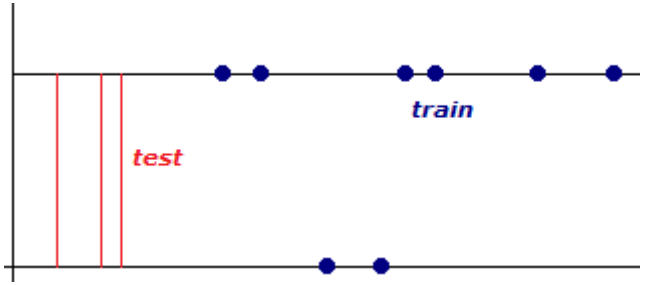
целевой признак кусочно-константный



Причём, максимум 2 «перепада»

Обучение и контроль распределены случайно…

Нет такого…



Задача «RedHat»

Подобные закономерности сложно увидеть в таблице…

	people_id	activity_id	date_x	activity_category	char_1_x	char_2_x	char_3_x	char_4_x	char_5_x	char_6_x	char_7_x	char_8_x	cha
189103	ppl_99966	act2_1740163	2022-09-23	type 2	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9
189103	ppl_99966	act2_1882139	2022-09-24	type 4	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9
189103	ppl_99966	act2_3544055	2022-09-27	type 2	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9
189103	ppl_99966	act2_4300471	2022-09-24	type 2	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9
189103	ppl_99966	act2_4353827	2022-09-24	type 2	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9
189103	ppl_99966	act2_4367217	2022-09-23	type 4	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9
189103	ppl_99966	act2_4459718	2022-09-24	type 4	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.999	-1.9

Так не видно…

Задача «RedHat»

	people_id	date_x	activity_category	outcome
189103	ppl_99966	2022-09-23	type 2	1
189103	ppl_99966	2022-09-24	type 4	0
189103	ppl_99966	2022-09-27	type 2	0
189103	ppl_99966	2022-09-24	type 2	0
189103	ppl_99966	2022-09-24	type 2	0
189103	ppl_99966	2022-09-23	type 4	1
189103	ppl_99966	2022-09-24	type 4	0

убрали лишние столбцы

А так?

Задача «RedHat»

	people_id	date_x	activity_category	outcome
189103	ppl_99966	2022-09-23	type 2	1
189103	ppl_99966	2022-09-23	type 4	1
189103	ppl_99966	2022-09-24	type 4	0
189103	ppl_99966	2022-09-24	type 2	0
189103	ppl_99966	2022-09-24	type 2	0
189103	ppl_99966	2022-09-24	type 4	0
189103	ppl_99966	2022-09-27	type 2	0

сделали сортировку по времени

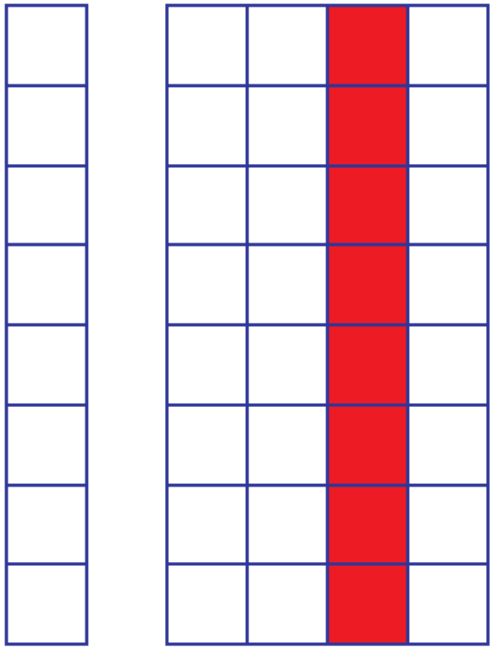
А так?

Полезные операции: группировка и сортировка!

нормировка и tiedrank

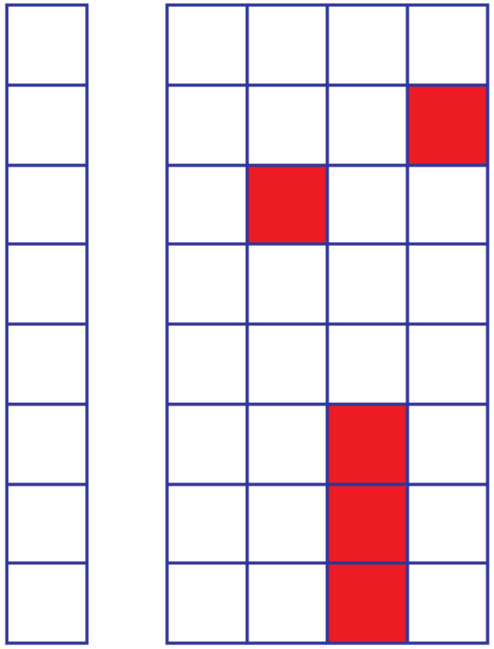
Что есть в данных:

- шумовые признаки



удалить

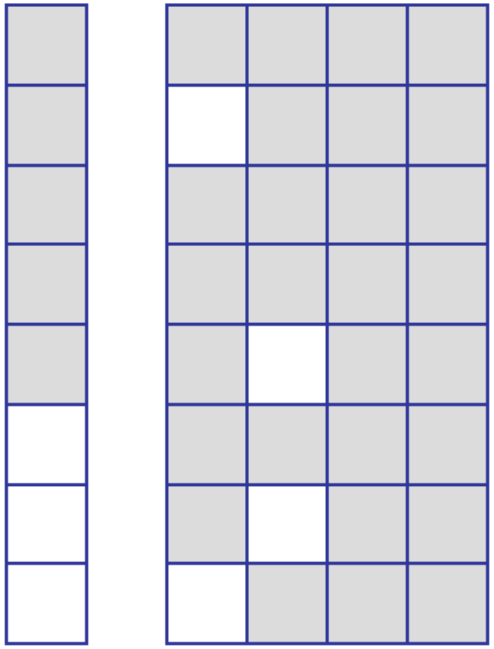
- шумовые значения



причины:

«ошибки из-за невнимательности», «особые режимы»

-пропуски:



причины:

«нет значения», «не знаем значения»

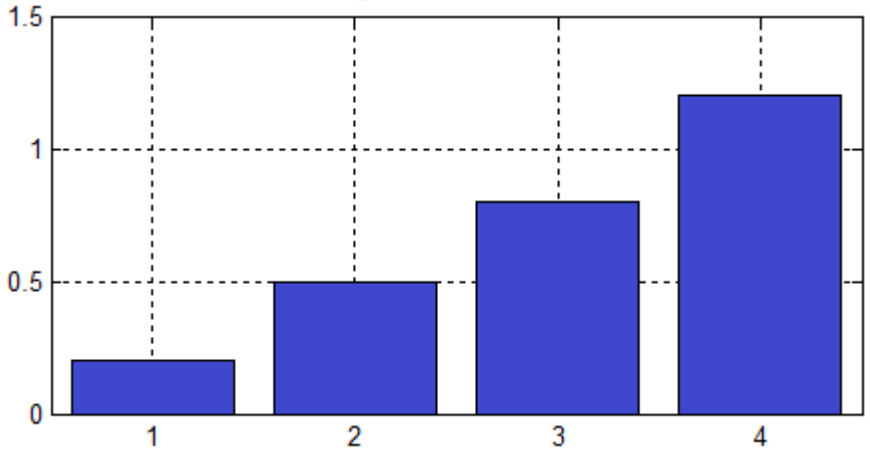
метод:

+dummy!!!

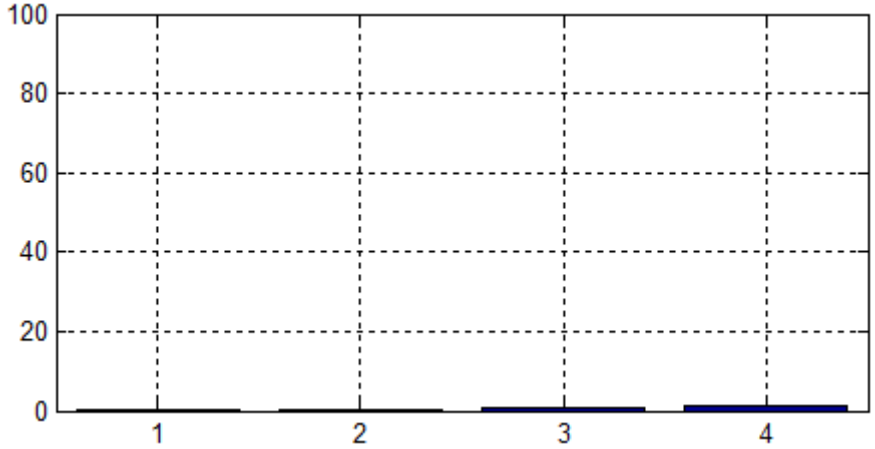
Про рекомендации к визуализации

Процент женщин в парламенте

«неправильно»



«правильно»



А если это процент убитых в Битцевском парке?

Про рекомендации к визуализации

24 июл в 12:25 Поделиться Мне нравится 8

масштаб отвратительный

24 июл в 12:43 | Ответить

Визуализация для профессионала

- где объективный возможный минимум наблюдаемых значений,
- где объективный возможный максимум наблюдаемых значений,
- какое ожидаемое среднее у наблюдаемых значений,
- какие отклонения наблюдаемых значений статистически значимы.