

# Машинное обучение

Лекция 8. Подводим итоги первой половины курса



# Содержание лекции

## I. Напоминание изученного

- Стандартные задачи и модели
- Линейные модели
- Решающие деревья и ансамбли деревьев
- Оценка качества

## II. Ответы на вопросы

## I. Напоминание изученного

# Стандартные задачи и методы

# Классификация



*Iris setosa*



*Iris versicolor*



*Iris virginica*

Вход (обучающая выборка):

Признаки N объектов с известными классами

Выход:

Классификатор (алгоритм, прогнозирующий классы новых объектов по их признакам)

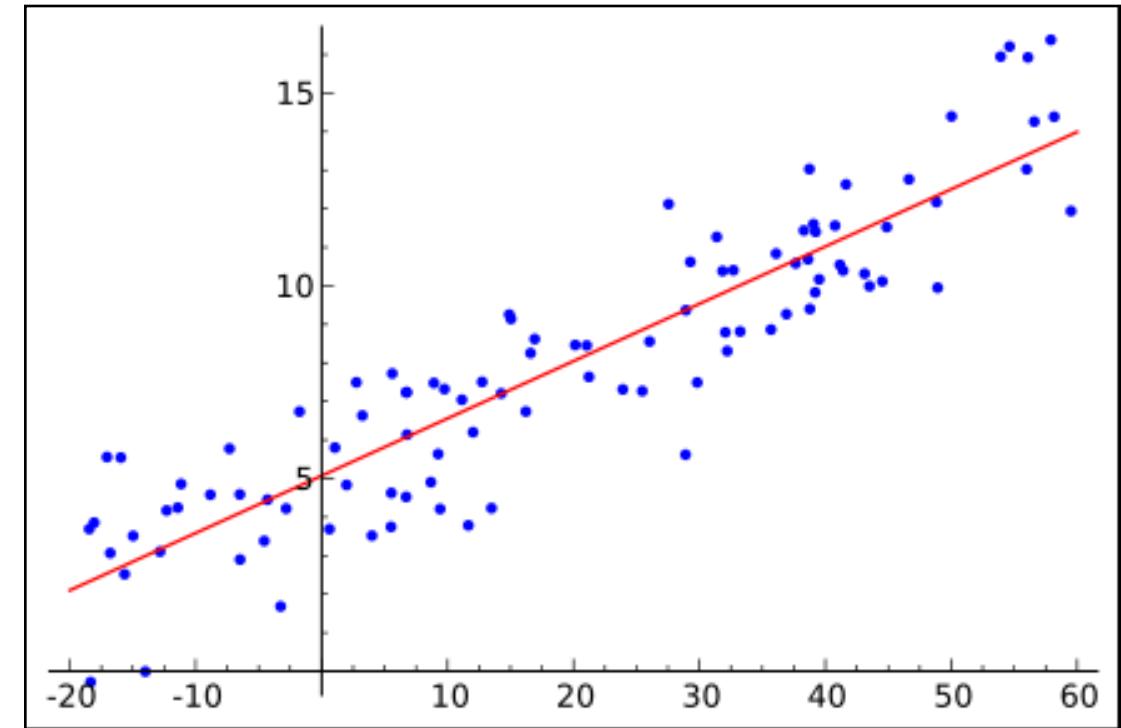
# Регрессия

**Вход (обучающая выборка):**

Признаки  $N$  объектов с известными значениями прогнозируемого вещественного параметра объекта

**Выход:**

Алгоритм, прогнозирующий значение вещественной величины по признакам объекта



# Кластеризация

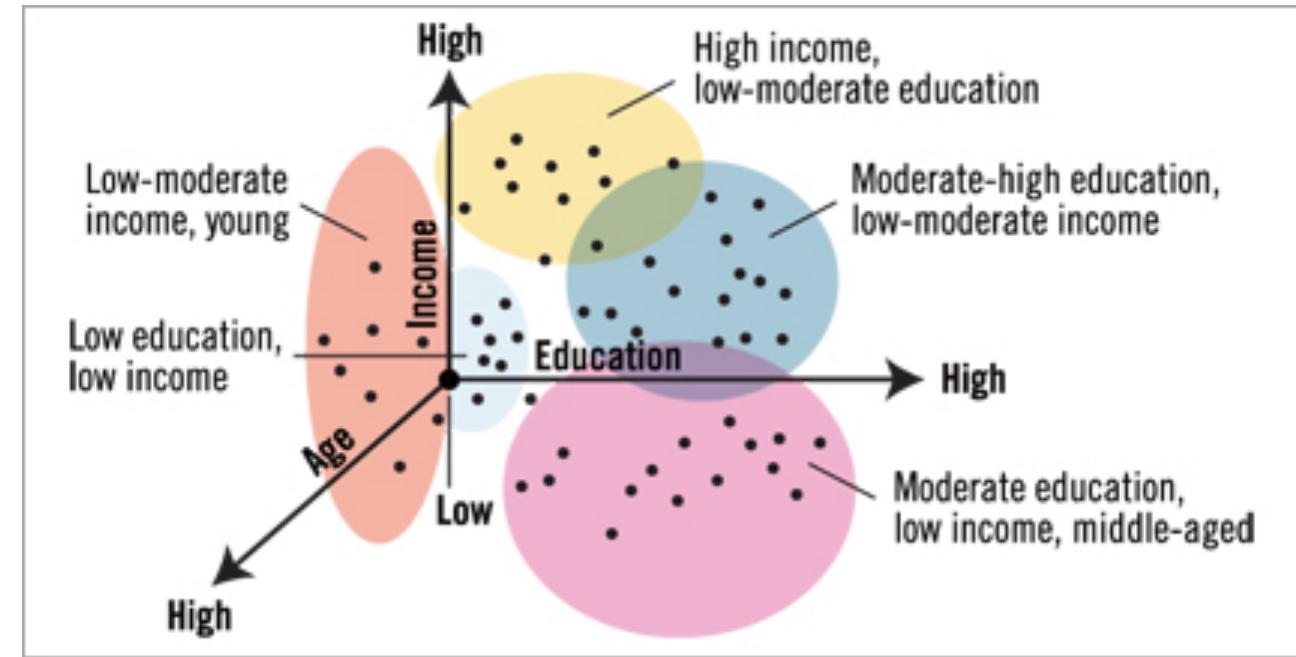
Вход (обучающая выборка):

Признаки  $N$  объектов

Выход:

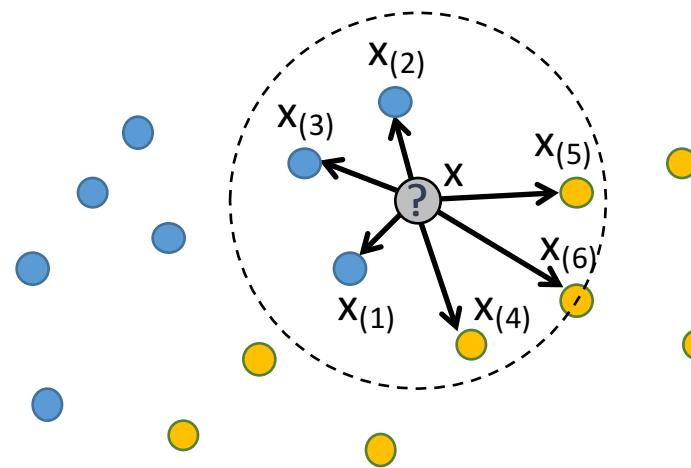
Найденные в выборке классы (кластеры), метки кластеров для объектов из обучающей выборки и алгоритм отнесения новых объектов к кластеру

Пример: сегментация рынка



# Взвешенный kNN

Пример классификации ( $k = 6$ ):



Веса можно определить как функцию от соседа или его номера:

$$w(x_{(i)}) = w(i)$$

или как функцию расстояния:

$$w(x(i)) = w(d(x, x_{(i)}))$$

$$Z_{\text{blue}} = \frac{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)})}{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)}) + w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}$$

$$Z_{\text{yellow}} = \frac{w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)}) + w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}$$

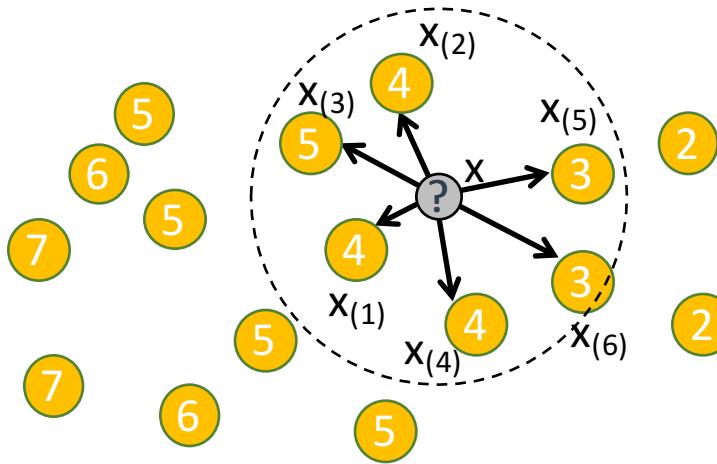
$$\text{?} = \operatorname{argmax}_{\circlearrowleft} Z_{\circlearrowleft}$$

$$\text{if } Z_{\text{yellow}} > Z_{\text{blue}} : \quad \text{?} = \text{yellow}$$

$$\text{if } Z_{\text{yellow}} < Z_{\text{blue}} : \quad \text{?} = \text{blue}$$

# Взвешенный kNN для регрессии

Пример ( $k = 6$ ):



Веса можно определить как функцию от соседа или его номера:

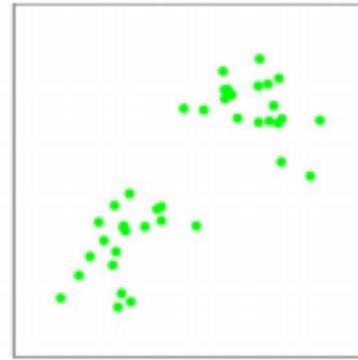
$$w(x_{(i)}) = w(i)$$

или как функцию расстояния:

$$w(x(i)) = w(d(x, x_{(i)}))$$

$$\text{?} = \frac{4 \cdot w(x_{(1)}) + 4 \cdot w(x_{(2)}) + 5 \cdot w(x_{(3)}) + 4 \cdot w(x_{(4)}) + 3 \cdot w(x_{(5)}) + 3 \cdot w(x_{(6)})}{w(x_{(1)}) + w(x_{(2)}) + w(x_{(3)}) + w(x_{(4)}) + w(x_{(5)}) + w(x_{(6)})}$$

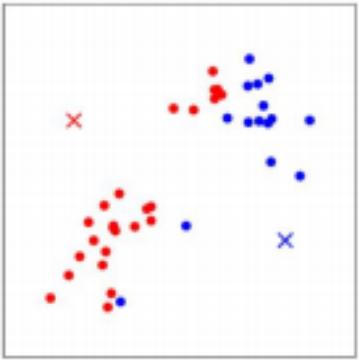
# Простой алгоритм кластеризации: kMeans



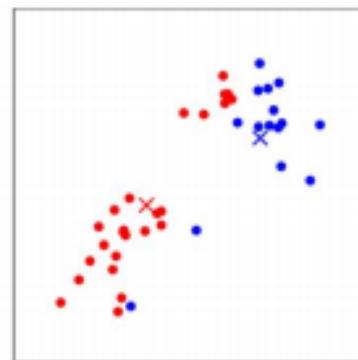
(a)



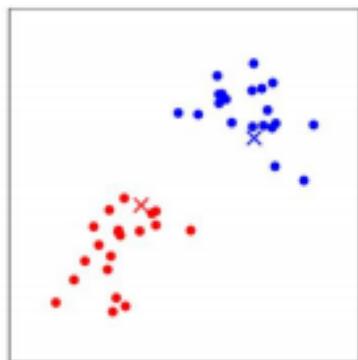
(b)



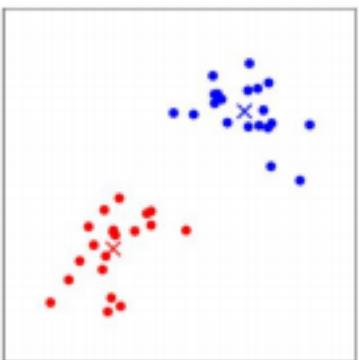
(c)



(d)



(e)

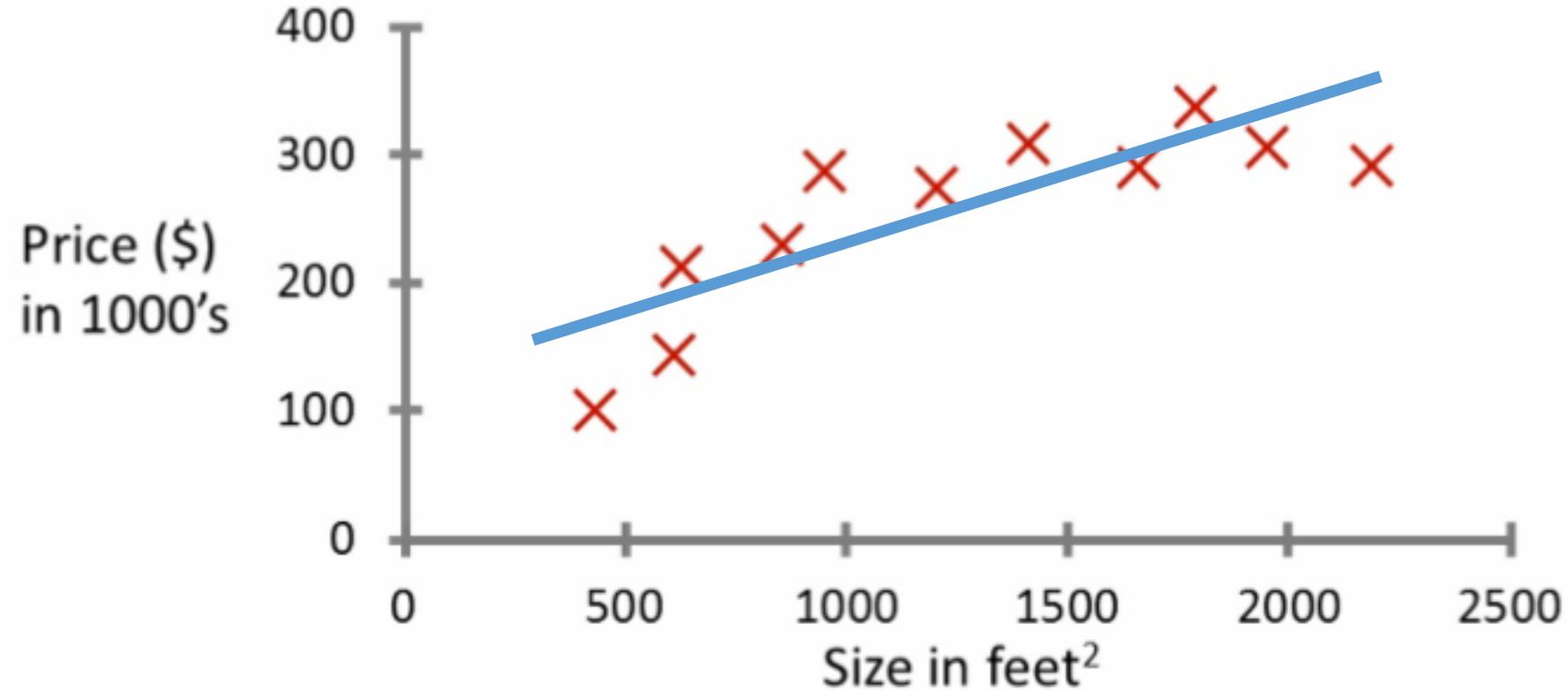


(f)

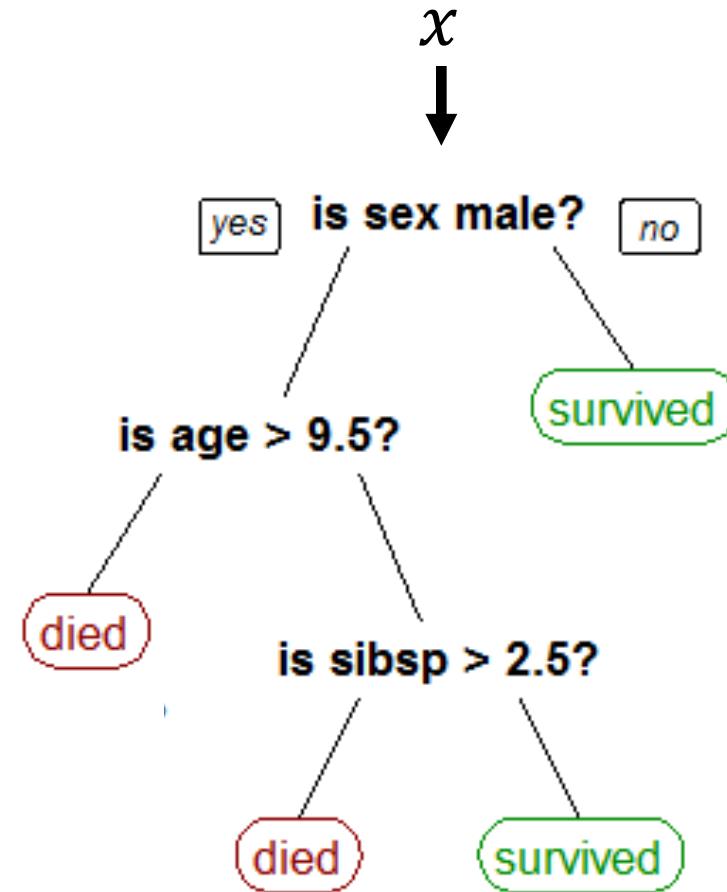
## Напоминание: часто используемые методы

- Линейные модели
- Решающие деревья
- Ансамбли решающих деревьев

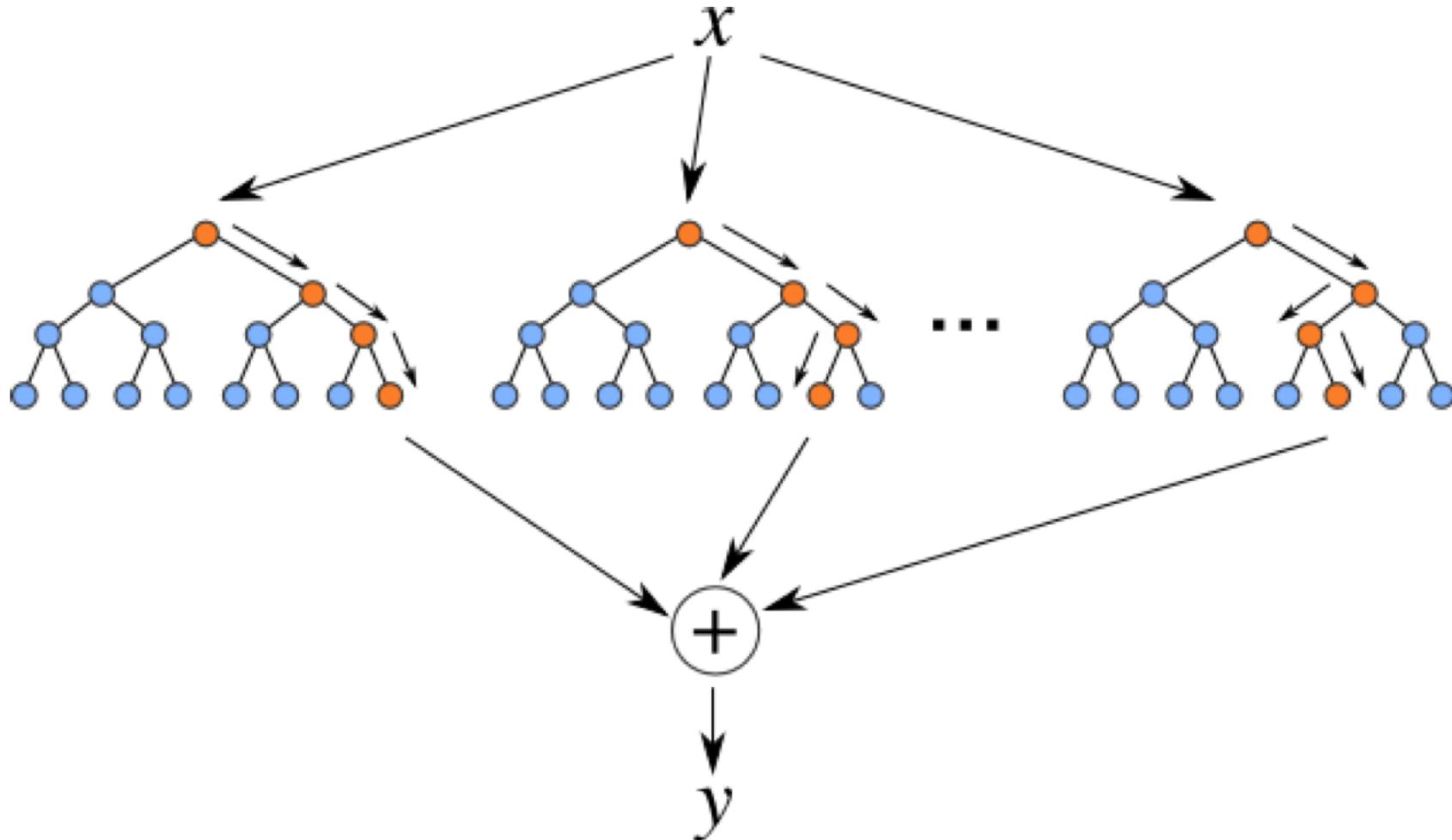
# Линейные модели



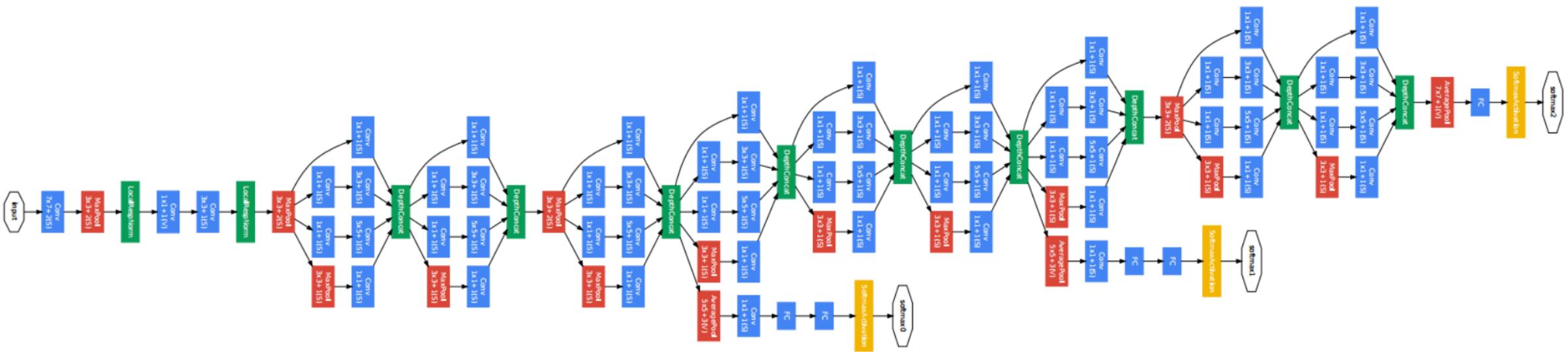
# Решающие деревья



# Ансамбли решающих деревьев

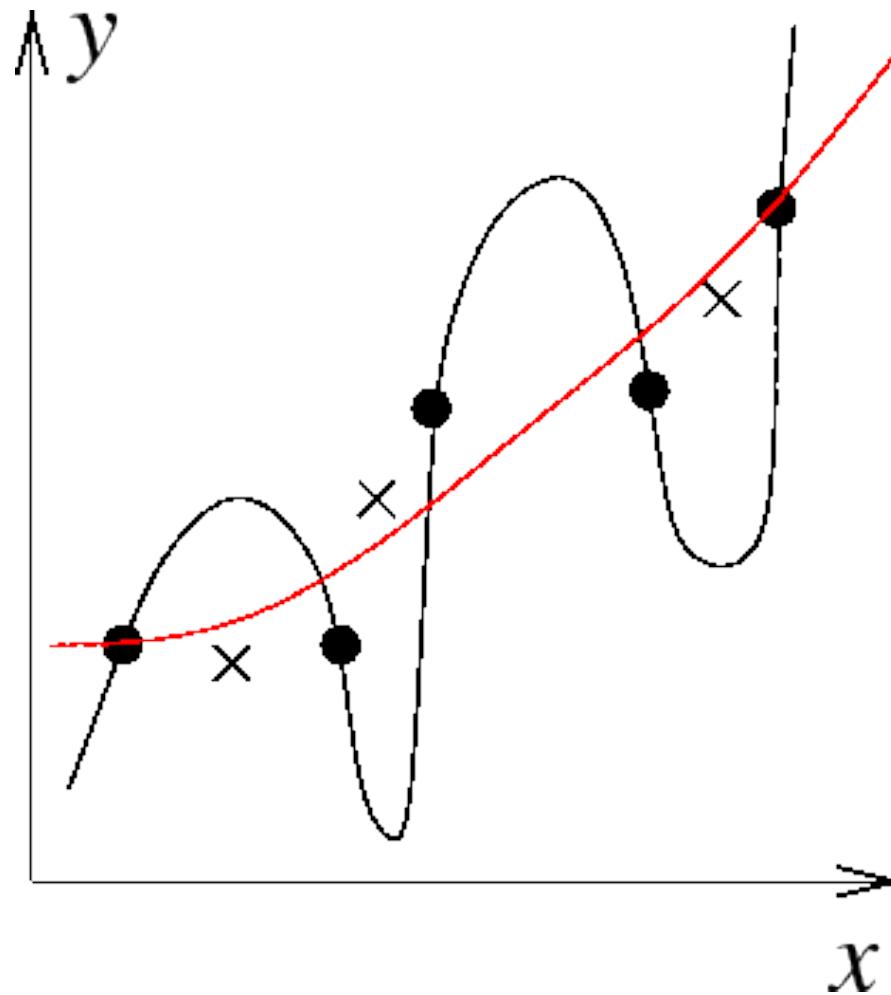


# Нейронные сети



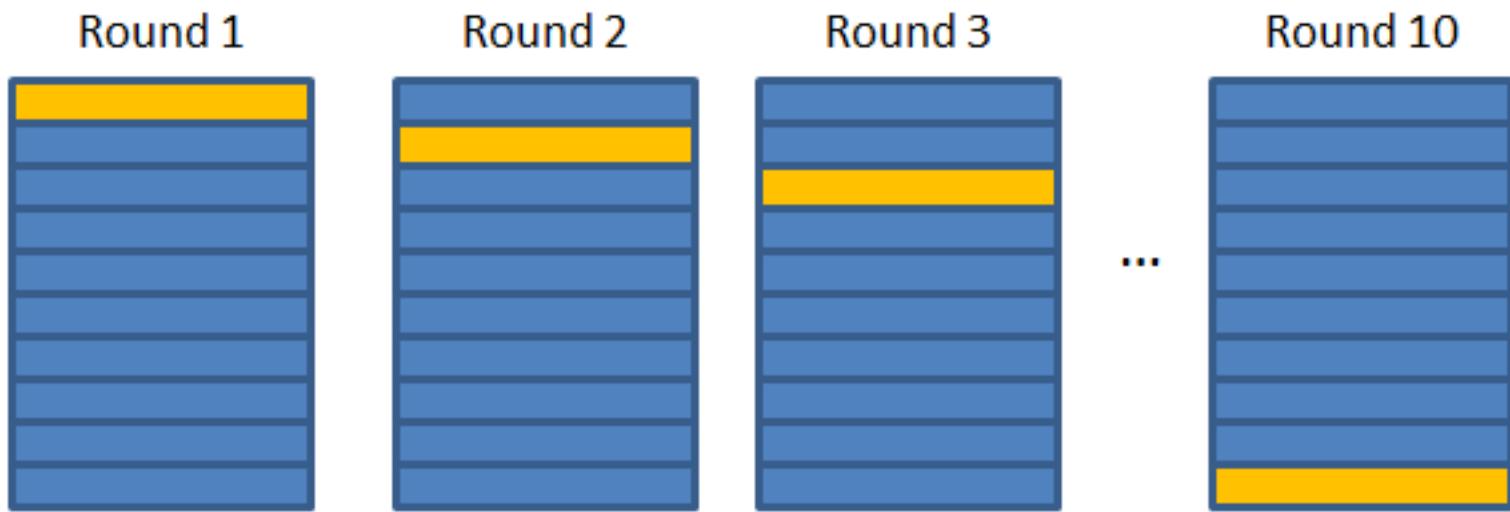
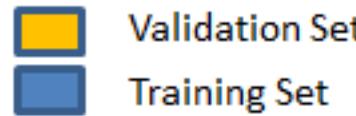
# GoogLeNet

# Переобучение на примере регрессии



# Кросс-валидация

K-Fold cross validation:



На картинке  $k = 10$ . Другие частые варианты – 3 и 5.

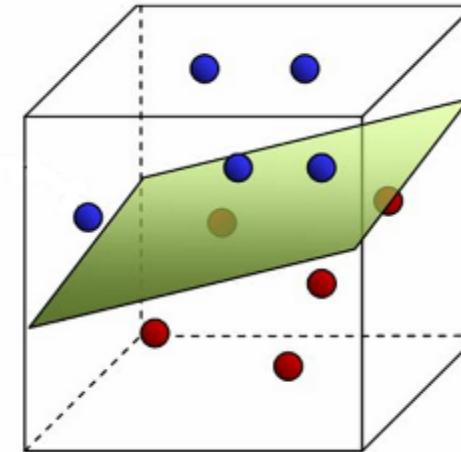
# Линейные модели

# Формализуем линейный классификатор

$$a(x) = \begin{cases} 1, & \text{если } f(x) > 0 \\ -1, & \text{если } f(x) \leq 0 \end{cases}$$

$$f(x) = w_0 + w_1 x_1 + \cdots + w_d x_d = w_0 + \langle w, x \rangle$$

Геометрическая интерпретация:  
разделяем классы плоскостью



## Формализуем линейный классификатор

$$a(x) = \begin{cases} 1, & \text{если } f(x) > 0 \\ -1, & \text{если } f(x) \leq 0 \end{cases}$$

Если добавляем  $x_{(0)} = 1$ , то:

$$\cancel{f(x) = w_0 + \langle w, x \rangle}$$

$$f(x) = \langle w, x \rangle$$

## Отступ (margin)

Отступом алгоритма  $a(x) = \text{sign}\{f(x)\}$  на объекте  $x_i$  называется величина

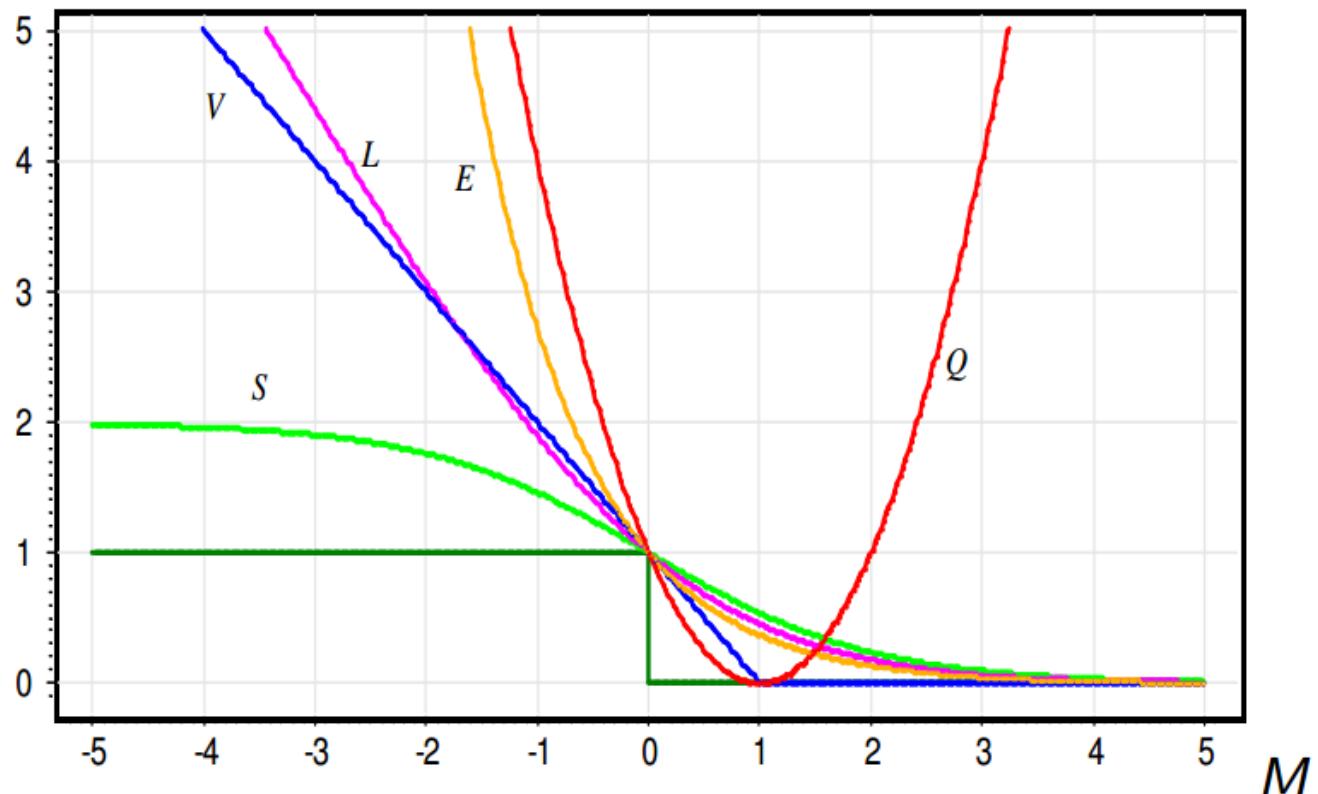
$$M_i = y_i f(x_i)$$

( $y_i$  - класс, к которому относится  $x_i$ )

$$\begin{aligned} M_i \leq 0 &\Leftrightarrow y_i \neq a(x_i) \\ M_i > 0 &\Leftrightarrow y_i = a(x_i) \end{aligned}$$

# ФУНКЦИЯ ПОТЕРЬ

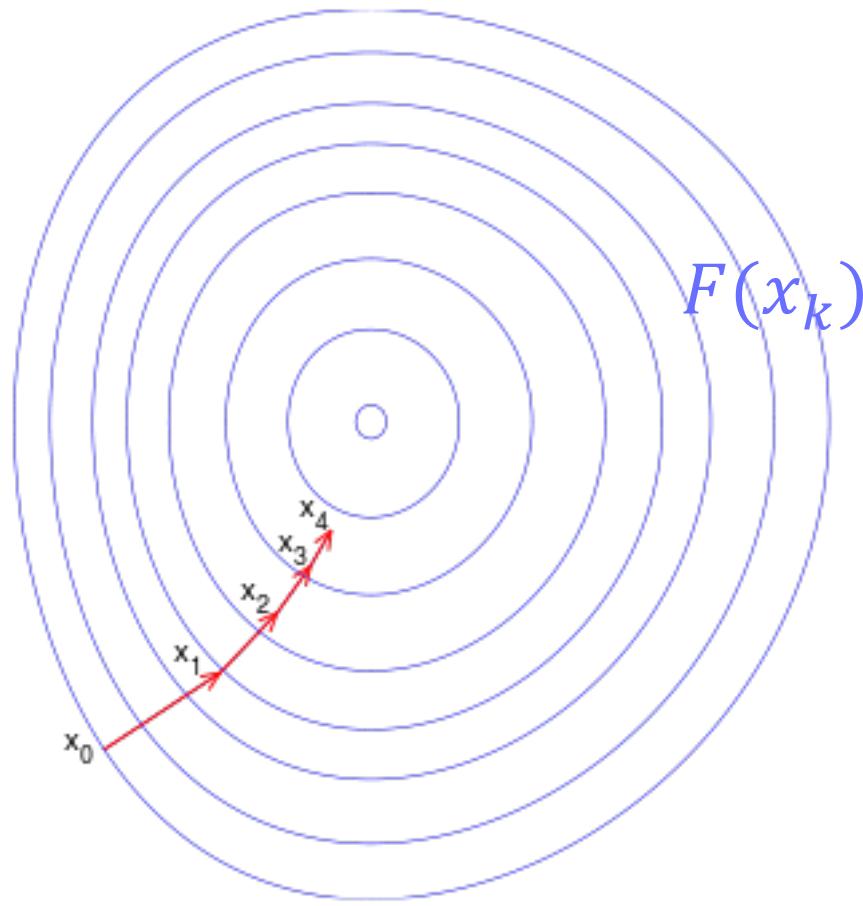
$$Q(w) = \sum_{i=1}^{\ell} [M_i(w) < 0] \leq \tilde{Q}(w) = \sum_{i=1}^{\ell} \mathcal{L}(M_i(w)) \rightarrow \min_w;$$



$$\begin{aligned} Q(M) &= (1 - M)^2 \\ V(M) &= (1 - M)_+ \\ S(M) &= 2(1 + e^M)^{-1} \\ L(M) &= \log_2(1 + e^{-M}) \\ E(M) &= e^{-M} \end{aligned}$$

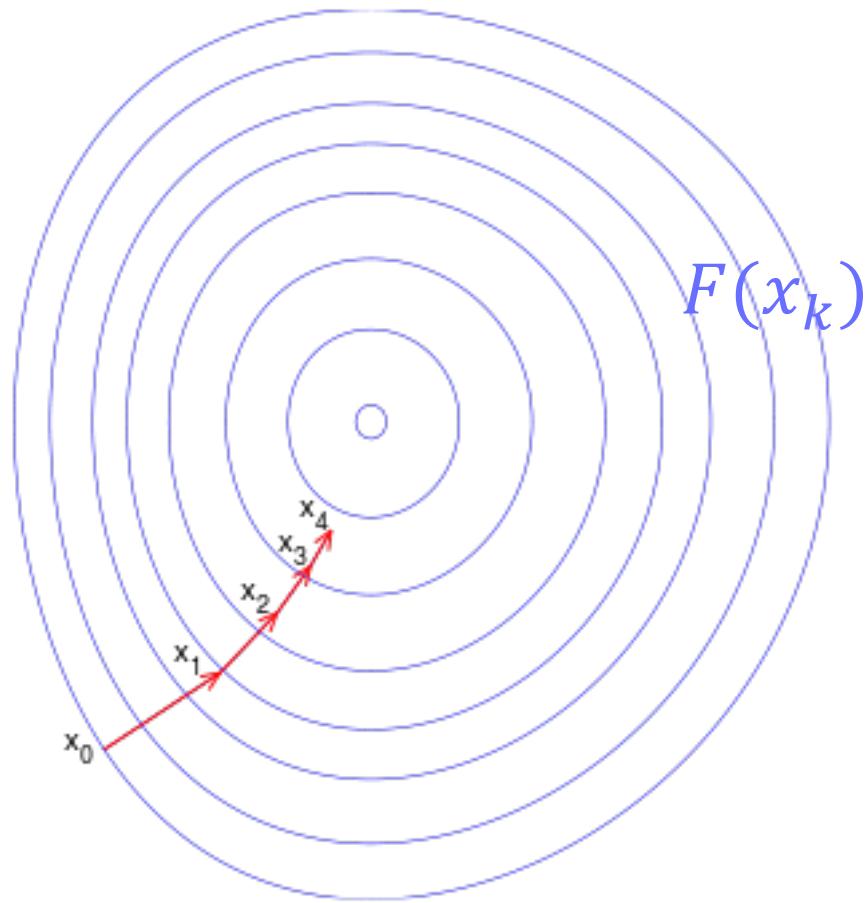
# Градиентный спуск (GD, Gradient Decent)

$$x_{k+1} = x_k - \gamma_k \nabla F(x_k)$$



# Градиентный спуск (GD, Gradient Decent)

$$x_{k+1} = x_k - \gamma_k \nabla F(x_k)$$



$$\nabla_w \tilde{Q} = \sum_{i=1}^l \nabla L(M_i) = \sum_{i=1}^l L'(M_i) \frac{\partial M_i}{\partial w}$$

$$M_i = y_i \langle w, x_i \rangle \Rightarrow \frac{\partial M_i}{\partial w} = y_i x_i$$

$$\nabla \tilde{Q} = \sum_{i=1}^l y_i x_i L'(M_i)$$

$$w_{k+1} = w_k - \gamma_k \sum_{i=1}^l y_i x_i L'(M_i)$$

## Стохастический градиент (SGD)

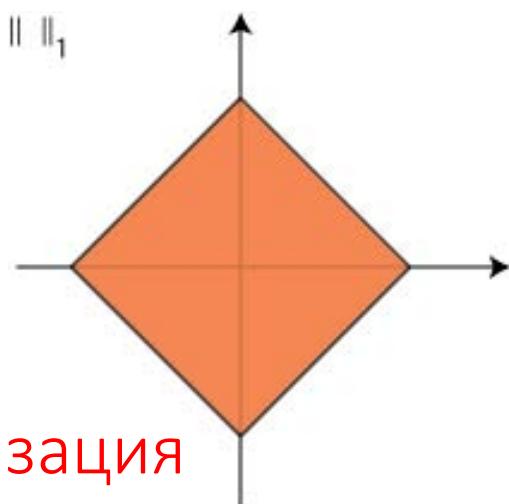
$$w_{k+1} = w_k - \gamma_k \sum_{i=1}^l y_i x_i L'(M_i)$$

$$w_{k+1} = w_k - \gamma_k y_i x_i L'(M_i)$$

$x_i$  – случайный элемент обучающей выборки

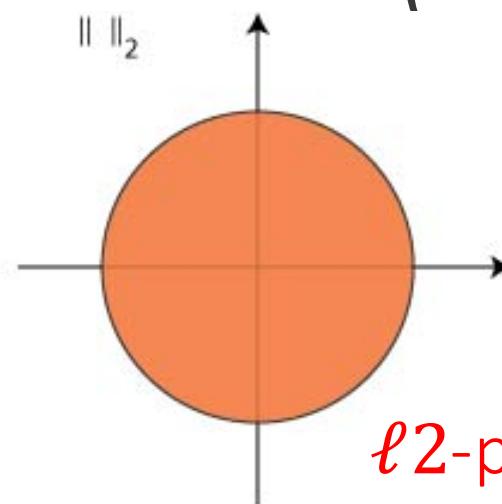
# Регуляризация

$$\left\{ \begin{array}{l} \tilde{Q} = \sum_{i=1}^l L(M_i) \rightarrow \min \\ \sum_{n=1}^d |w_n| \leq \tau \end{array} \right.$$



$\ell 1$ -регуляризация

$$\left\{ \begin{array}{l} \tilde{Q} = \sum_{i=1}^l L(M_i) \rightarrow \min \\ \sum_{n=1}^m {w_n}^2 \leq \tau \end{array} \right.$$

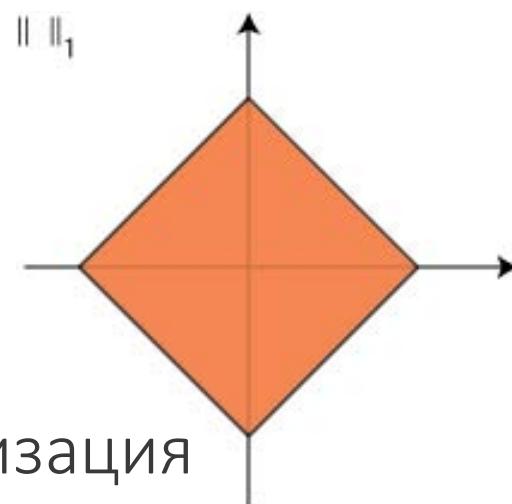


$\ell 2$ -регуляризация

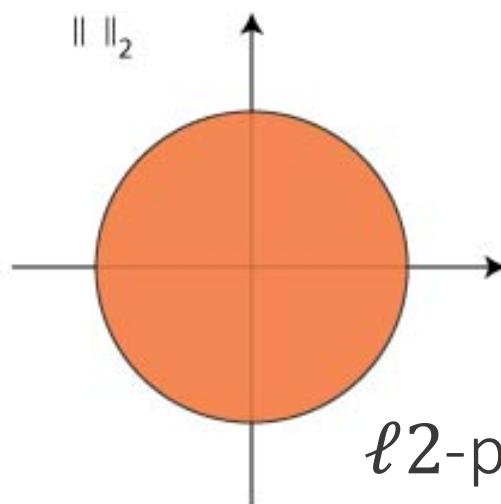
# Регуляризация

$$\sum_{i=1}^l L(M_i) + \gamma \sum_{n=1}^d |w_n| \rightarrow \min$$

$$\sum_{i=1}^l L(M_i) + \gamma \sum_{n=1}^d {w_n}^2 \rightarrow \min$$



$\ell 1$ -регуляризация



$\ell 2$ -регуляризация

# Общий случай

$$Q = \sum_{i=1}^{\ell} L(y_i, f(x_i)) + \gamma V(w) \rightarrow \min_w$$

Функция потерь

Коэффициент  
регуляризации

Регуляризатор

## Упражнение:

Как будет меняться качество на обучающей и на тестовой выборке с ростом коэффициента регуляризации в SVM и в логистической регрессии в sklearn? Выясните, почему результат такой.

# Стандартные линейные классификаторы

Классификатор	Функция потерь	Регуляризатор
SVM (Support vector machine, метод опорных векторов)	$L(M) = \max\{0, 1 - M\} = (1 - M)_+$	$\sum_{k=1}^m w_k^2$
Логистическая регрессия	$L(M) = \log(1 + e^{-M})$	Обычно $\sum_{k=1}^m w_k^2$ или $\sum_{k=1}^m  w_k $

# Обязательно ли функция потерь – функция от отступа?

Пример:

$$y_i \in \{0, 1\} \quad Q = - \sum_{i=1}^{\ell} y_i \ln p_i + (1 - y_i) \ln(1 - p_i) \rightarrow \min_w$$

$$p_i = \sigma(\langle w, x_i \rangle) = \frac{1}{1 + e^{-\langle w, x_i \rangle}}$$

Упражнение:

Показать, что это та же оптимизационная задача, что и в логистической регрессии

# Линейные модели в регрессии

$$a(x) = \langle w, x \rangle + w_0$$

$$Q = \sum_{i=1}^l L(y_i, a(x_i)) + \gamma V(w) \rightarrow \min_w$$

$$L(y_i, a(x_i)) = (y_i - a(x_i))^2 \quad L(y_i, a(x_i)) = |y_i - a(x_i)|$$

$$V(w) = \|w\|_{l2}^2 = \sum_{n=1}^d w_n^2$$

$$V(w) = \|w\|_{l1} = \sum_{n=1}^d |w_n|$$

# Линейные модели в регрессии

$$a(x) = \langle w, x \rangle + w_0$$

$$Q = \sum_{i=1}^l L(y_i, a(x_i)) + \gamma V(w) \rightarrow \min_w$$

Гребневая регрессия  
(Ridge regression):

$$V(w) = \|w\|_{l2}^2 = \sum_{n=1}^d w_n^2$$

LASSO (least absolute  
shrinkage and selection  
operator):

$$V(w) = \|w\|_{l1} = \sum_{n=1}^d |w_n|$$

# Линейная регрессия

$$a(x) = \langle w, x \rangle + w_0$$

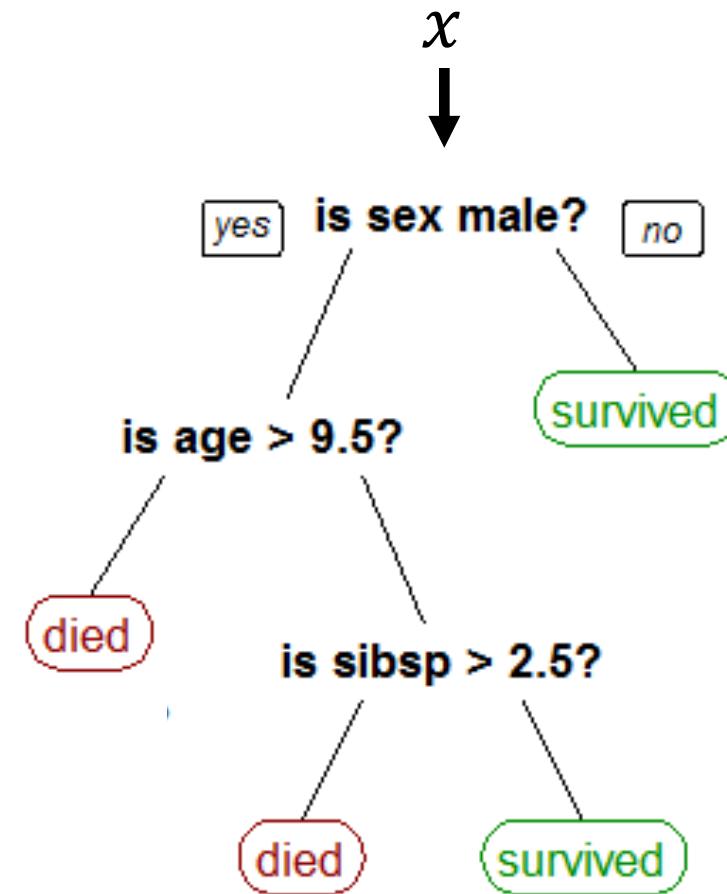
$$Q = \sum_{i=1}^l L(y_i, a(x_i)) \rightarrow \min_w$$

$$L(y_i, a(x_i)) = (y_i - a(x_i))^2$$

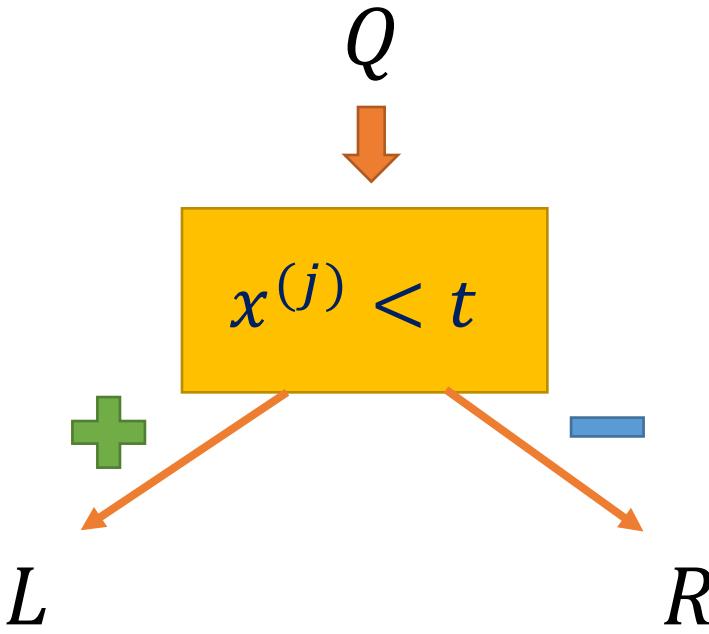
А без регуляризатора и с квадратичными потерями  
получаем привычную нам линейную регрессию

Решающие деревья и ансамбли деревьев

# Решающее дерево



# Выбор разбиения



$$G(j, t) = \frac{|L|}{|Q|} H(L) + \frac{|R|}{|Q|} H(R) \rightarrow \min_{j,t}$$

$H(R)$  - мера «неоднородности» множества  $R$

# Критерии построения разбиений

$H(R)$  – мера «неоднородности» множества  $R$

Пусть мы решаем задачу классификации на 2 класса,  
 $p_0, p_1$  – доли объектов классов 0 и 1 в  $R$

1) Misclassification criteria:  $H(R) = 1 - \max\{p_0, p_1\}$

2) Entropy criteria:  $H(R) = -p_0 \ln p_0 - p_1 \ln p_1$

3) Gini criteria:  $H(R) = 1 - p_0^2 - p_1^2 = 2p_0p_1$

# Критерии построения разбиений

$H(R)$  – мера «неоднородности» множества  $R$

Пусть мы решаем задачу классификации на  $K$  классов,  
 $p_1, \dots, p_K$  – доли объектов классов 1, ...,  $K$  в  $R$

1) Misclassification criteria:  $H(R) = 1 - p_{max}$

2) Entropy criteria:

$$H(R) = - \sum_{k=1}^K p_k \ln p_k$$

3) Gini criteria:

$$H(R) = \sum_{k=1}^K p_k(1 - p_k)$$

# Критерии построения разбиений

$H(R)$  – мера «неоднородности» множества  $R$

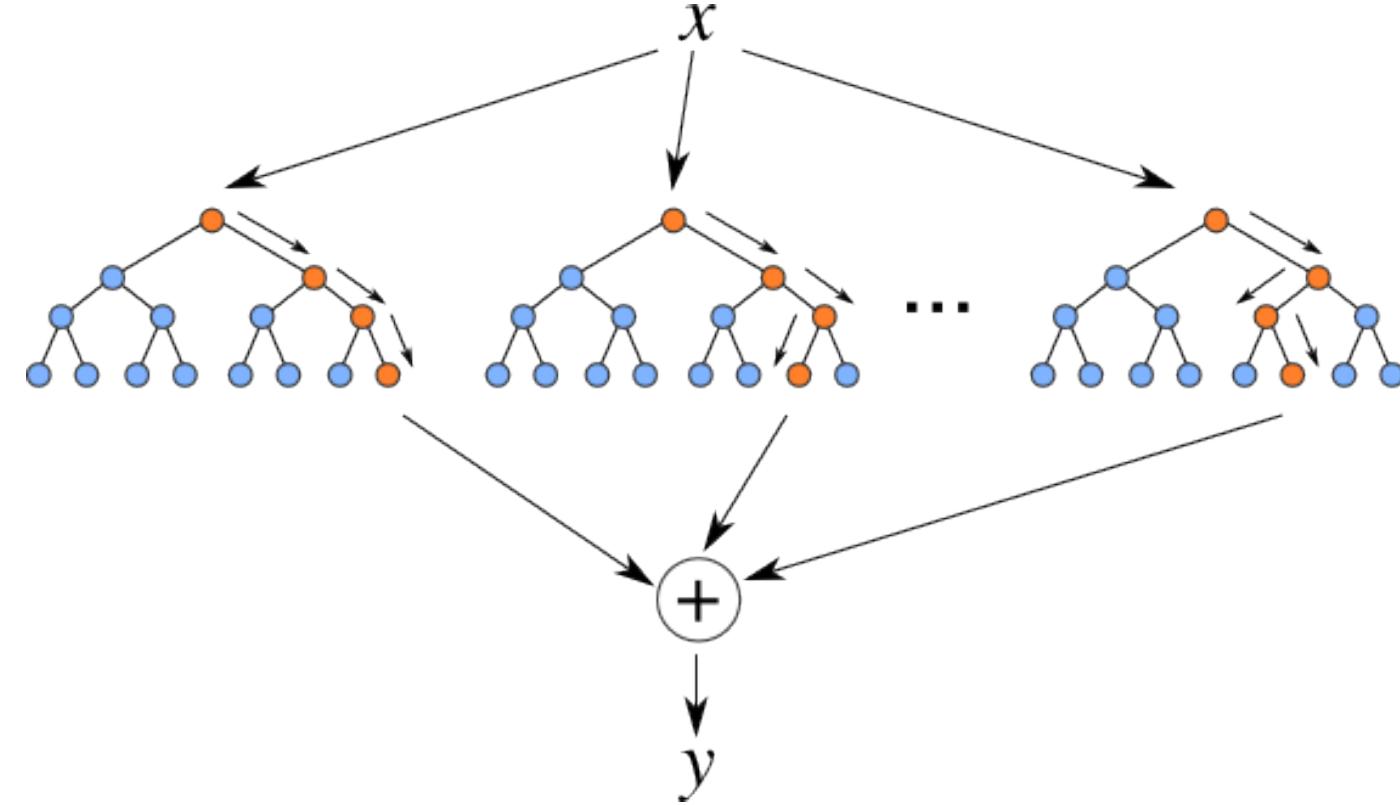
Чтобы решать задачу регрессии, достаточно взять среднеквадратичную ошибку в качестве  $H(R)$ :

$$H(R) = \frac{1}{|R|} \sum_{x_i \in R} (y_i - \bar{y})^2$$

$$\bar{y} = \frac{1}{|R|} \sum_{x_i \in R} y_i$$

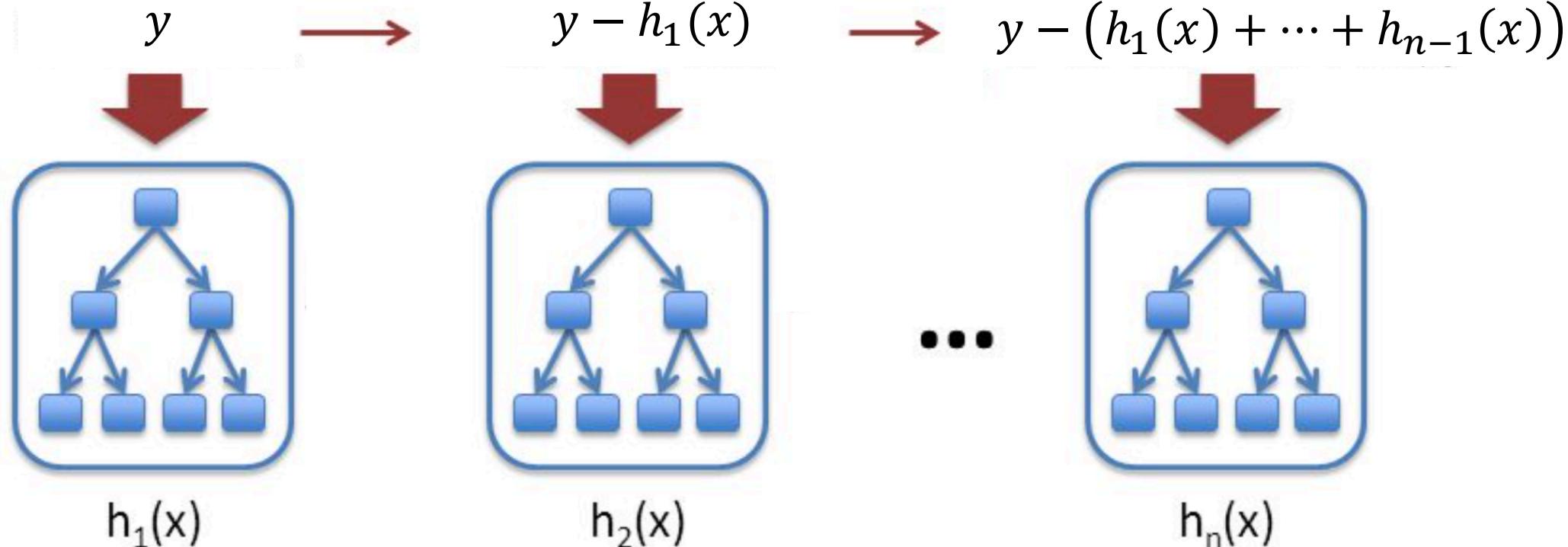
# Random Forest

1. Генерируем  $M$  выборок на основе имеющейся
2. Строим на них деревья с randomизированными разбиениями в узлах: выбираем  $k$  случайных признаков и ищем наиболее информативное разбиение по ним
3. При прогнозе усредняем ответ всех деревьев



# Идея Gradient Boosted Decision Trees (GBDT)

$$h(x) = h_1(x) + \dots + h_n(x)$$

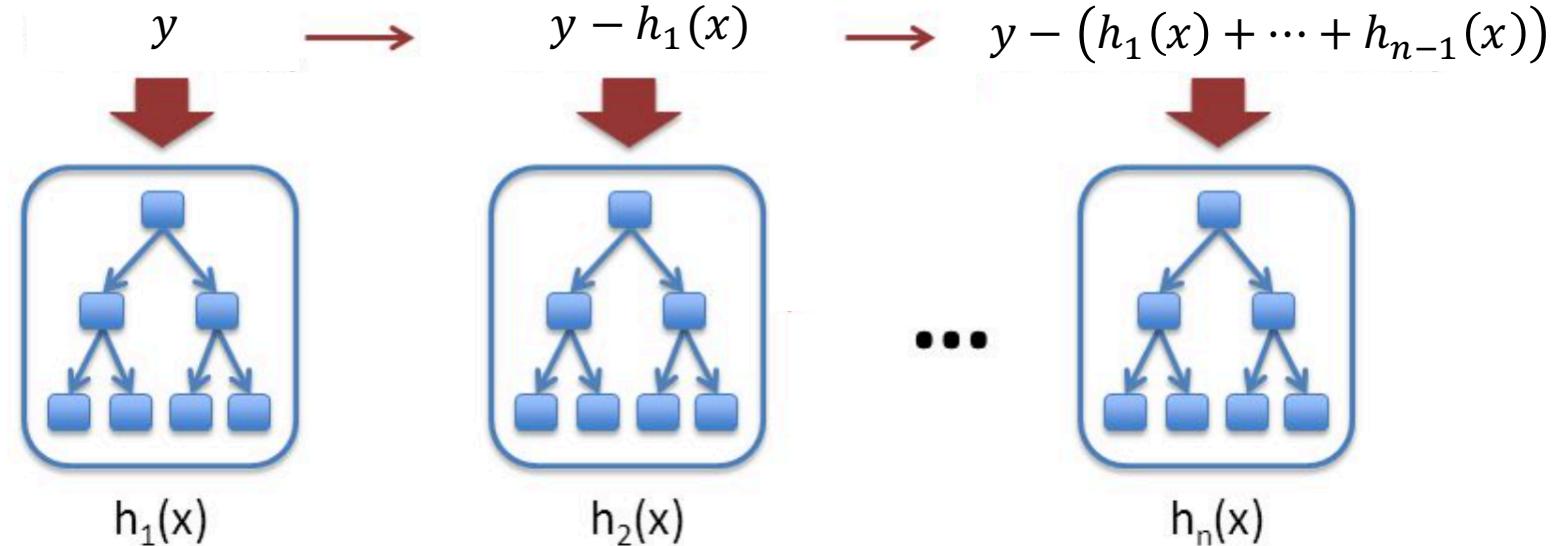


# Gradient Boosted Decision Trees

- Каждое новое дерево  $h_k(x)$  обучаем на ответы  $y_i - h_i$   
 $h_i$  - прогноз всей композиции на  $i$ -том объекте на предыдущей итерации
- Коэффициент  $\alpha_k$  перед новым деревом подбираем с помощью численной оптимизации ошибки

# Идея Gradient Boosted Decision Trees

$$a_n(x) = h_1(x) + \cdots + h_n(x)$$



## Аналогия с численной оптимизацией

Нам нужно минимизировать ошибку:

$$Q(\hat{y}, y) = \sum_{i=1}^l (\hat{y}_i - y_i)^2 \rightarrow \min \quad \hat{y}_i = a(x_i)$$

Если бы мы подбирали ответы  $\hat{y}$  итеративно, можно было бы это делать градиентным спуском

Но нам нужно подобрать не ответы, а функцию  $a(x)$

# Градиентный бустинг и градиент

В бустинге

$$a(x) = \sum_{t=1}^T \beta_t h_t(x)$$

**Идея:** будем каждый следующий алгоритм выбирать так, чтобы он приближал антиградиент ошибки

$$h_t(x) \approx -\frac{\partial Q(\hat{y}, y)}{\partial \hat{y}}$$

## Градиентный бустинг и градиент

Если  $h_t(x) \approx -\frac{\partial Q(\hat{y}, y)}{\partial \hat{y}}$  и  $Q(\hat{y}, y) = \sum_{i=1}^l (\hat{y}_i - y_i)^2$

$$h_t(x_i) \approx -\frac{\partial Q(\hat{y}_i, y_i)}{\partial \hat{y}_i} = -2(\hat{y}_i - y_i) \propto y_i - \hat{y}_i$$

## GBM с квадратичными потерями

1. Обучаем первый базовый алгоритм  $h_1$ ,  $\beta_1 = 1$
2. Повторяем в цикле по  $t$  от 2 до  $T$ :

обучаем  $h_t$  на ответы  $y_i - a_{t-1}(x_i)$

выбираем  $\beta_t$

## GBM с квадратичными потерями

1. Обучаем первый базовый алгоритм  $h_1$ ,  $\beta_1 = 1$
2. Повторяем в цикле по  $t$  от 2 до  $T$ :

обучаем  $h_t$  на ответы  $y_i - a_{t-1}(x_i)$

выбираем  $\beta_t$

Стратегии выбора  $\beta_t$ :

- всегда равен небольшой константе
- как в методе наискорейшего спуска
- уменьшая с ростом  $t$

## GBM с произвольными потерями

1. Обучаем первый базовый алгоритм  $h_1$ ,  $\beta_1 = 1$
2. Повторяем в цикле по  $t$  от 2 до  $T$ :

обучаем  $h_t$  на  $-\frac{\partial Q(\hat{y}_i, y_i)}{\partial \hat{y}_i} = -\frac{\partial L(\hat{y}_i, y_i)}{\partial \hat{y}_i}$

выбираем  $\beta_t$

$$\text{Здесь } Q(\hat{y}, y) = \sum_{i=1}^l L(\hat{y}_i, y_i) \quad \hat{y}_i = a_{t-1}(x_i)$$

# Оценка качества

# Метрики качества регрессии

- MAE
- RMSE
- MAPE
- SMAPE
- logloss

# Метрики качества классификации

- Accuracy
- Precision
- Recall
- F-measure
- ROC-AUC

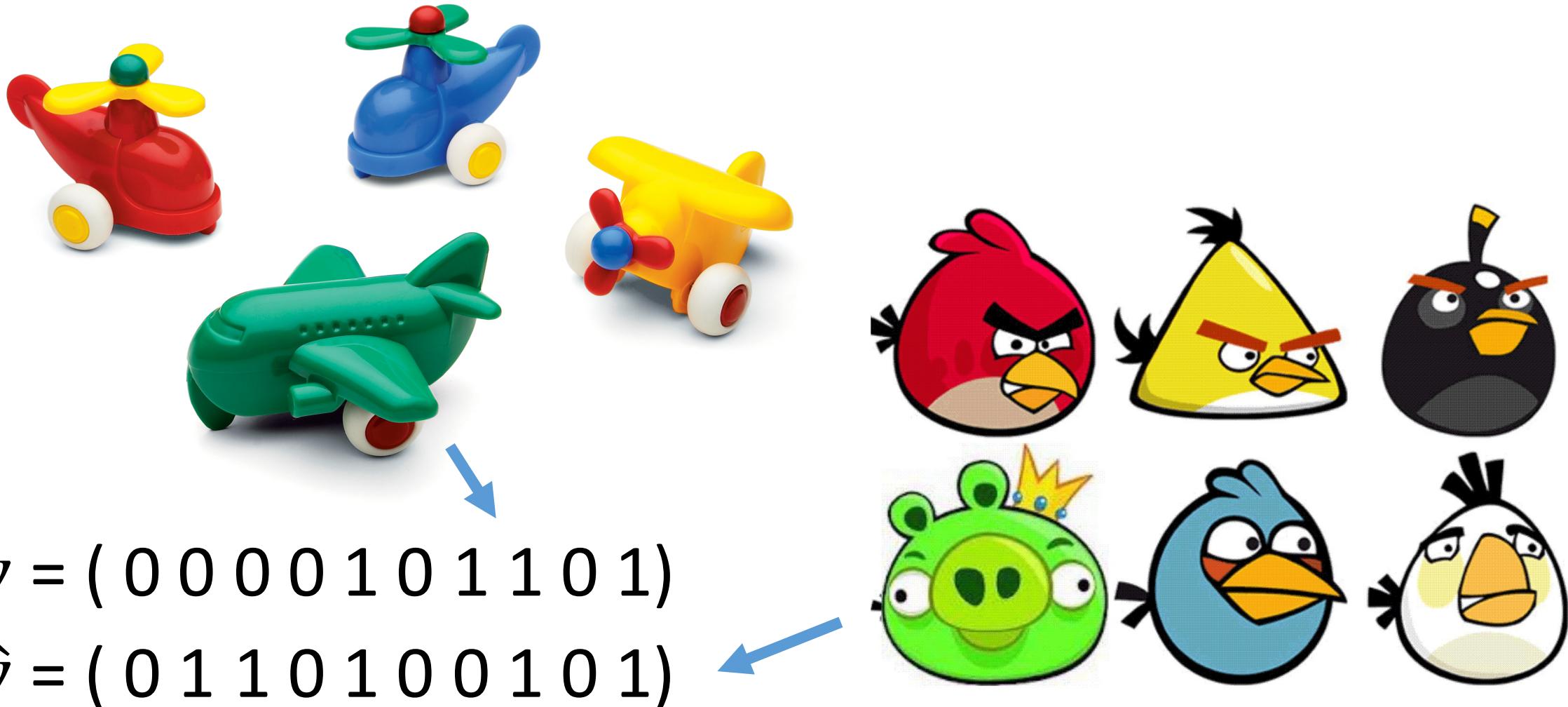
# Precision & Recall

- Precision – точность
- Recall - полнота

# Сбитые самолеты



# Сбитые самолеты



Precision

Precision – точность выстрелов:

Количество сбитых самолётов

---

Количество выстрелов

$$y = (0\ 0\ 0\ 0\ \textcolor{red}{1}\ 0\ 1\ \textcolor{red}{1}\ 0\ 1)$$

$$\hat{y} = (\textcolor{blue}{0}\ \textcolor{blue}{1}\ \textcolor{blue}{1}\ 0\ 1\ 0\ 0\ \textcolor{blue}{1}\ 0\ 1)$$



Recall

Recall – «полнота» сбивания самолетов:

**Количество сбитых самолётов**

---

Общее количество самолётов

$$y = (0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1)$$

$$\hat{y} = (0\ 1\ 1\ 0\ \textcolor{red}{1}\ 0\ 0\ 1\ 0\ 1)$$



# Связь с True Positive, False Positive и др.

		Actual Class	
		Yes	No
Predicted Class	Yes	True Positive	False Positive
	No	False Negative	True Negative

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

## F-measure (F-score, F1)

- Среднее геометрическое между precision и recall
- Значение F-measure ближе к меньшему из precision, recall

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

# Метрики качества классификации

- Accuracy
- Precision
- Recall
- F-measure
- ROC-AUC

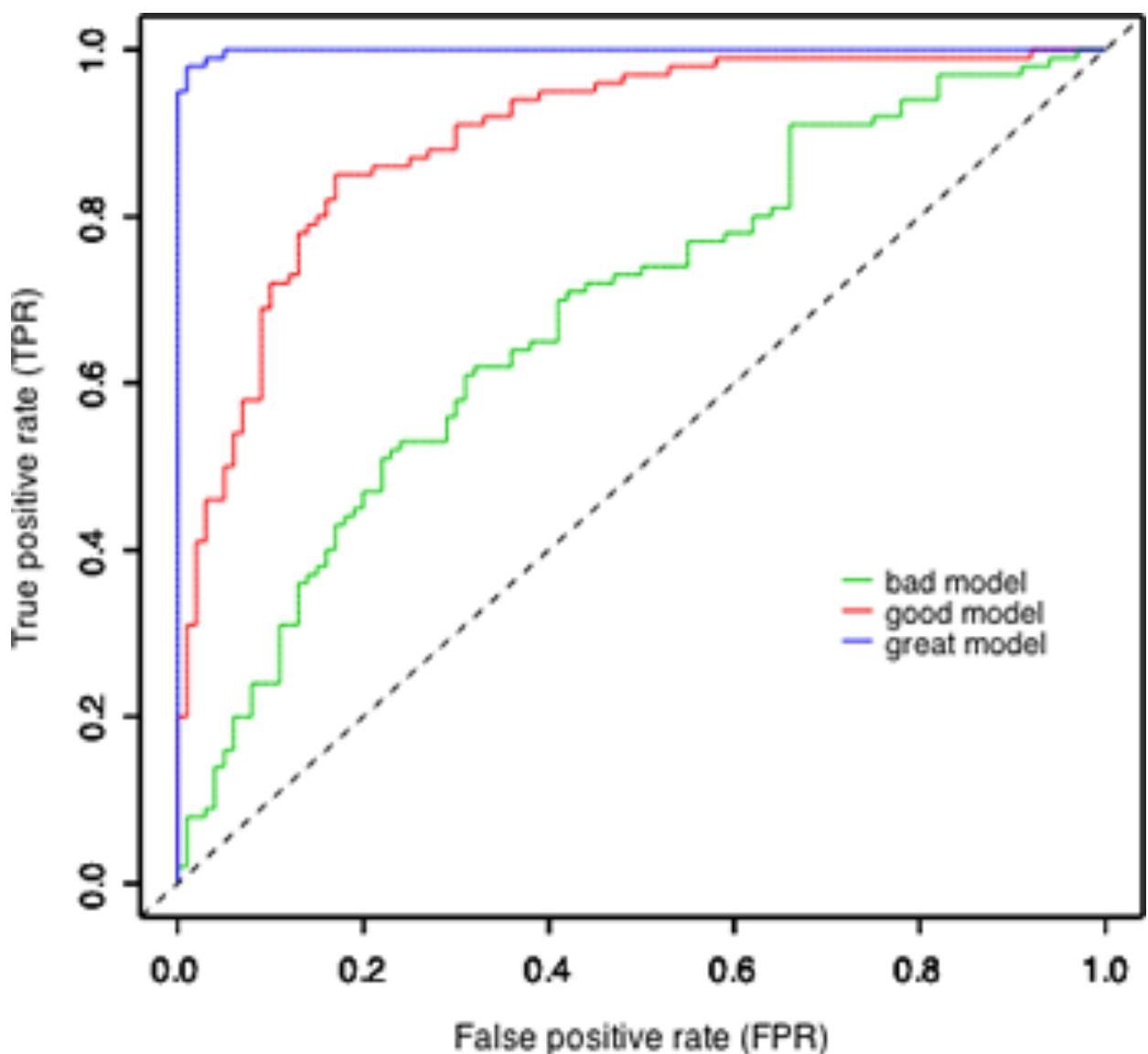
# ROC

		Actual Class	
		Yes	No
Predicted Class	Yes	True Positive	False Positive
	No	False Negative	True Negative

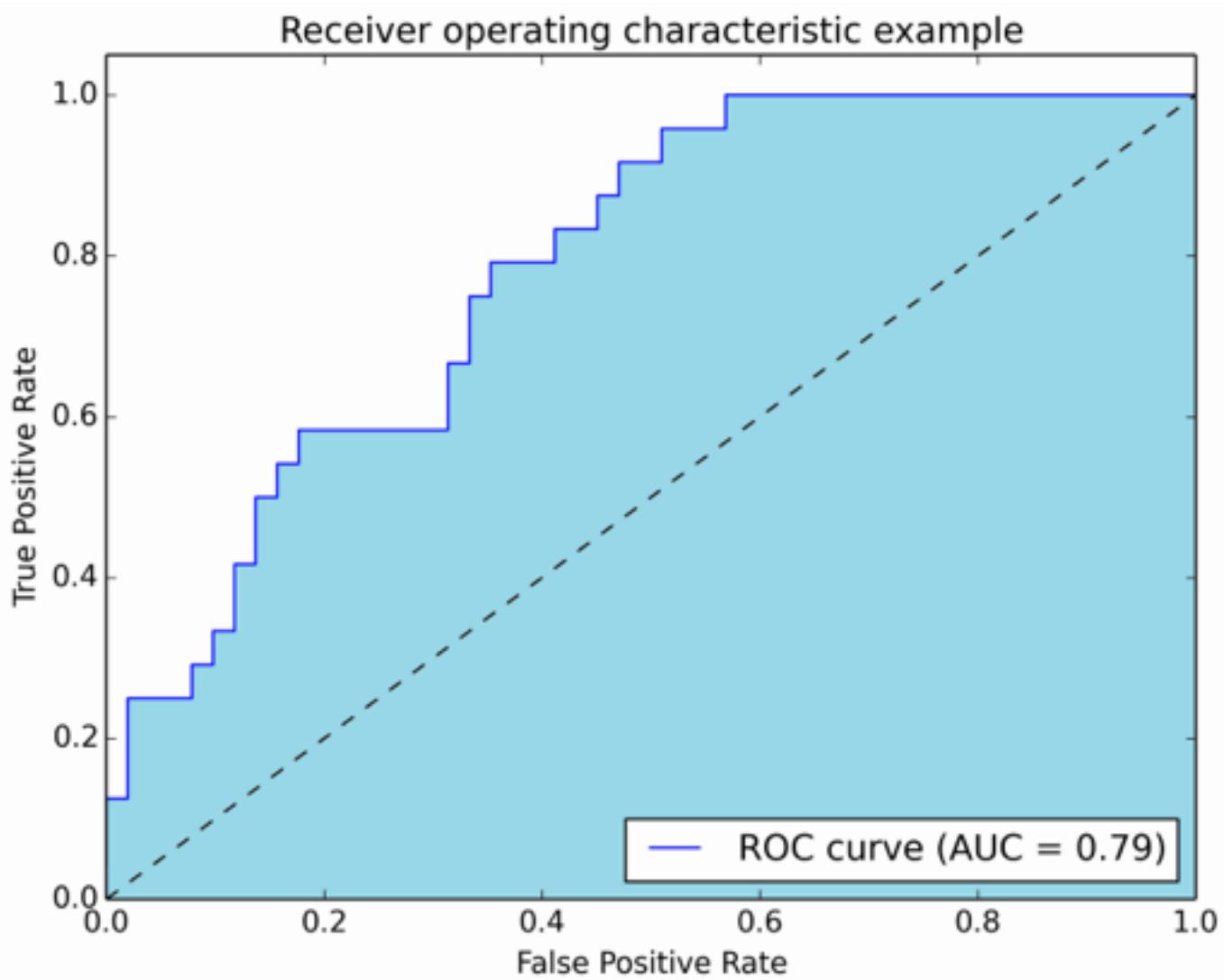
$$TPR = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

$$FPR = \frac{\text{False positives}}{\text{False positives} + \text{True negatives}}.$$

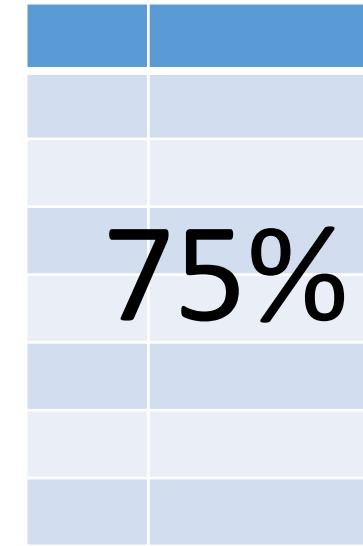
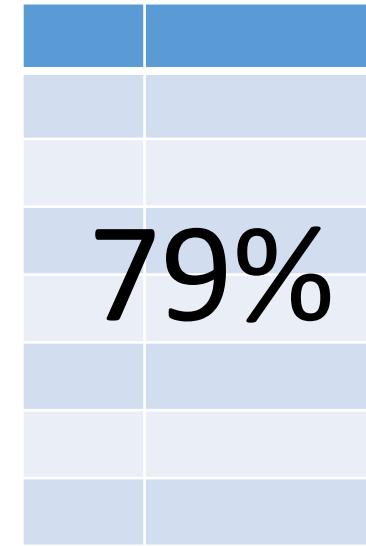
# ROC



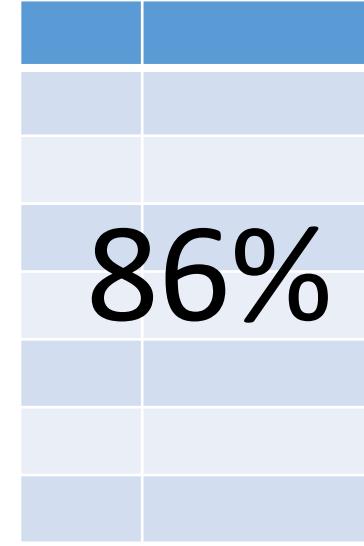
# ROC-AUC



# Проблема: разброс качества на разных данных

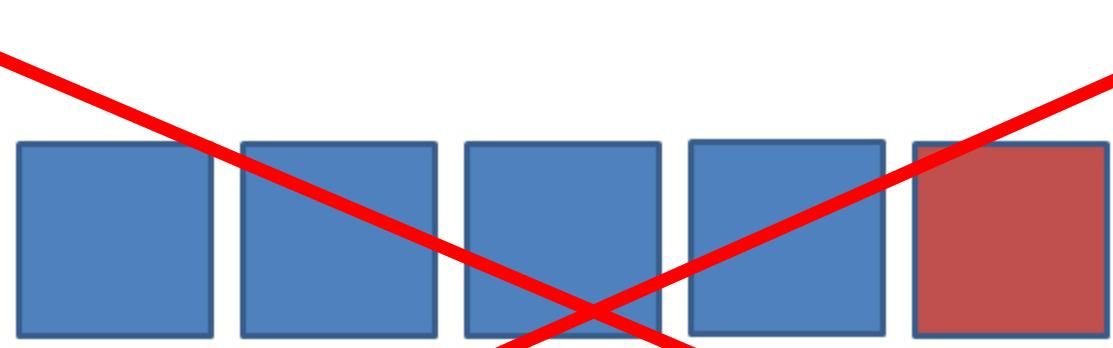


...



## Шаг 1: усреднение качества в CV

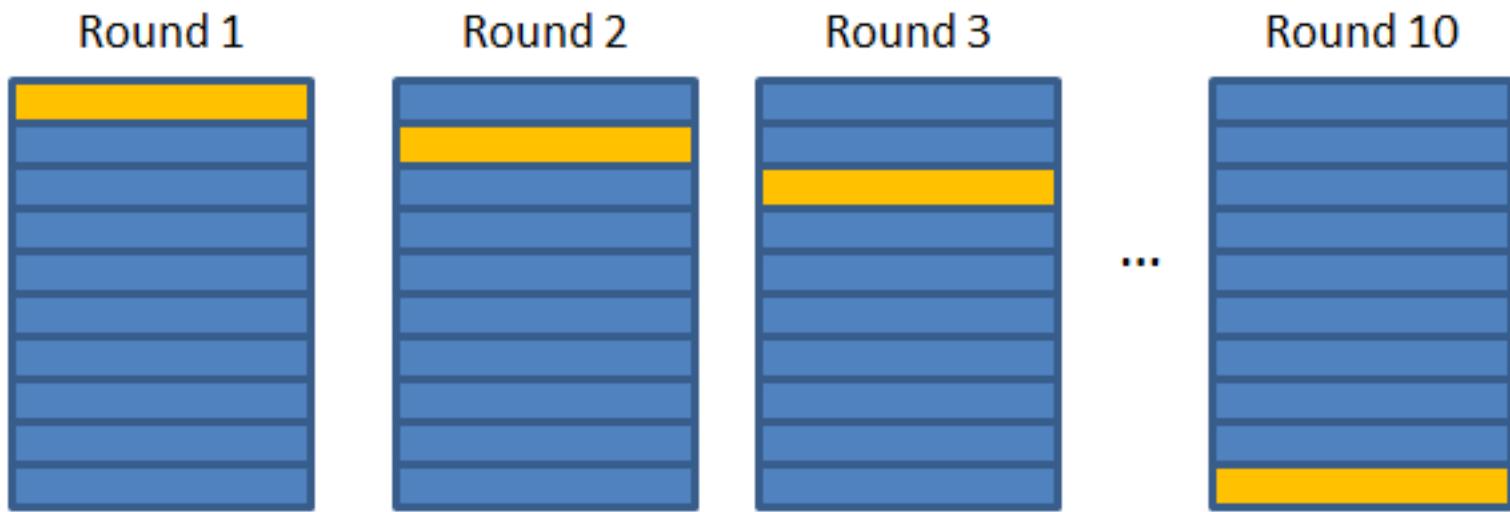
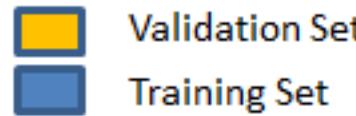
Если есть проблема со стабильностью модели, точно нужно избегать оценок на одном фиксированном датасете



Нужно использовать оценку качества в кросс-валидации

# Кросс-валидация

K-Fold cross validation:



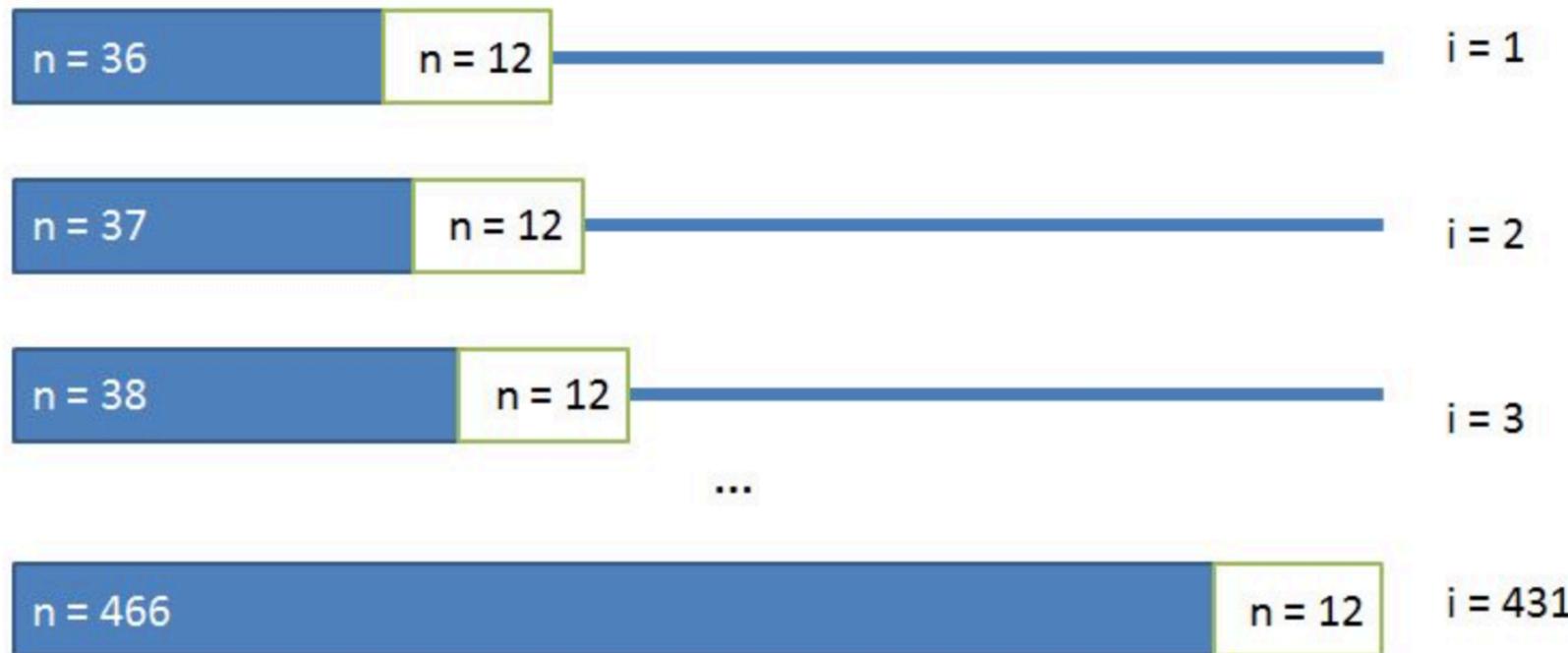
На картинке  $k = 10$ . Другие частые варианты – 3 и 5.

# Предупреждение: будьте осторожны с CV

N = 478 (month-end data)

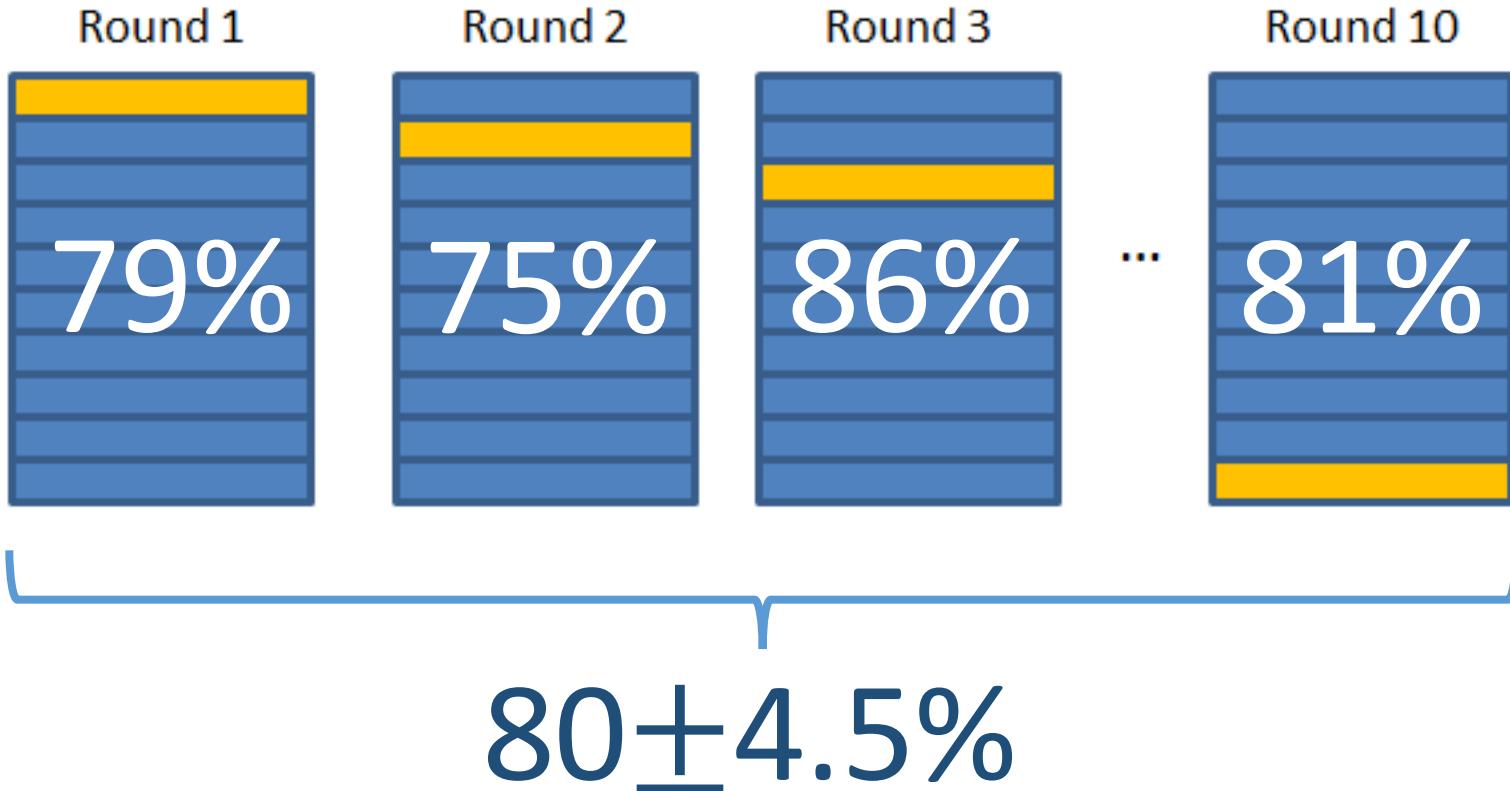
June 1967

March 2007

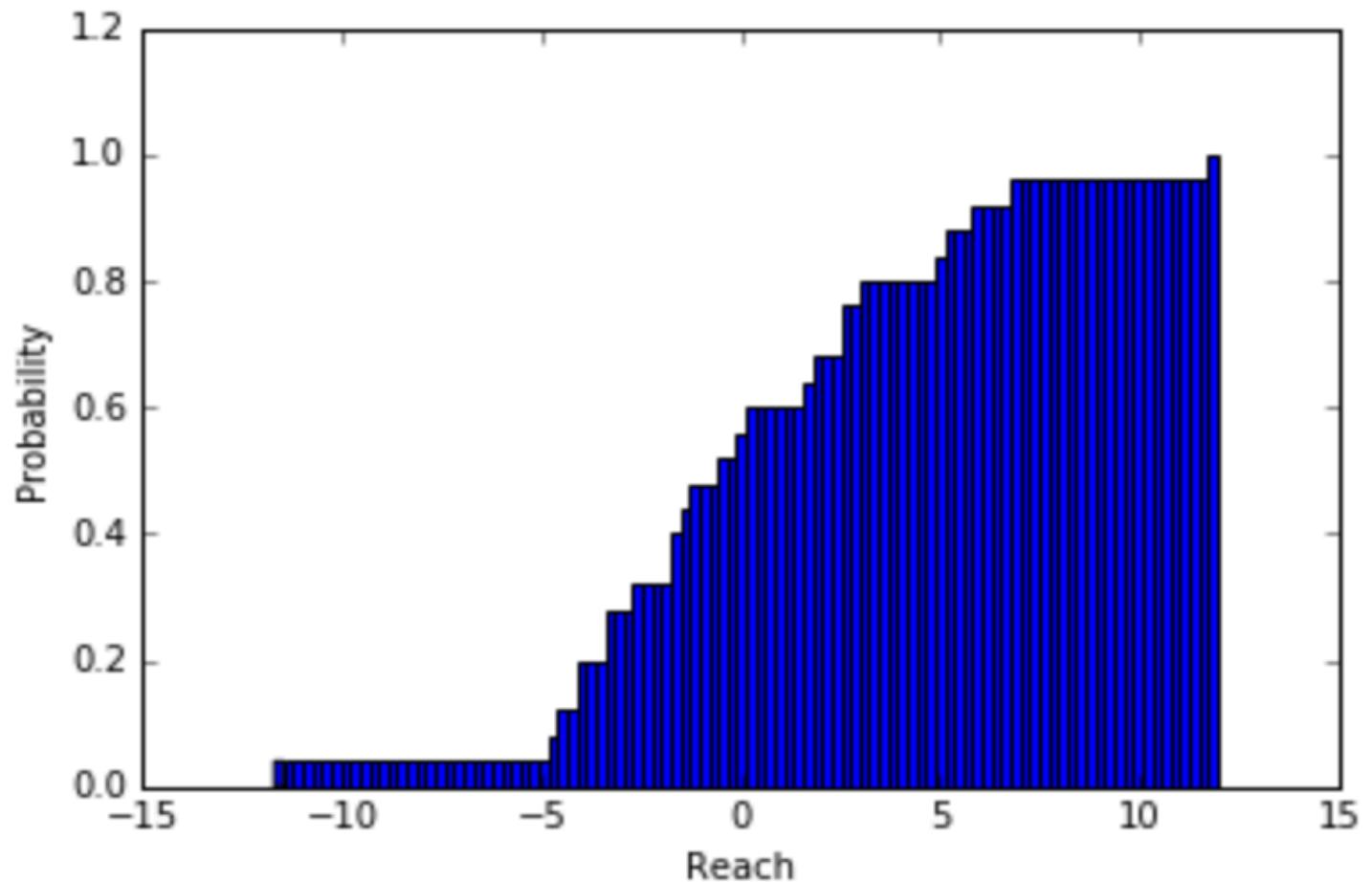


## Шаг 2: учет разброса и распределения в CV

 Validation Set  
 Training Set



Шаг 2: учет разброса и распределения в CV



## Шаг 3: анализ топа важных признаков

На одном фолде:

0.211268 Номер  
0.147105 Ширина  
0.128326 Вес  
0.0954617 Параметр 1  
0.0688576 Высота  
0.057903 Параметр 2  
0.0438185 Параметр 3

На другом:

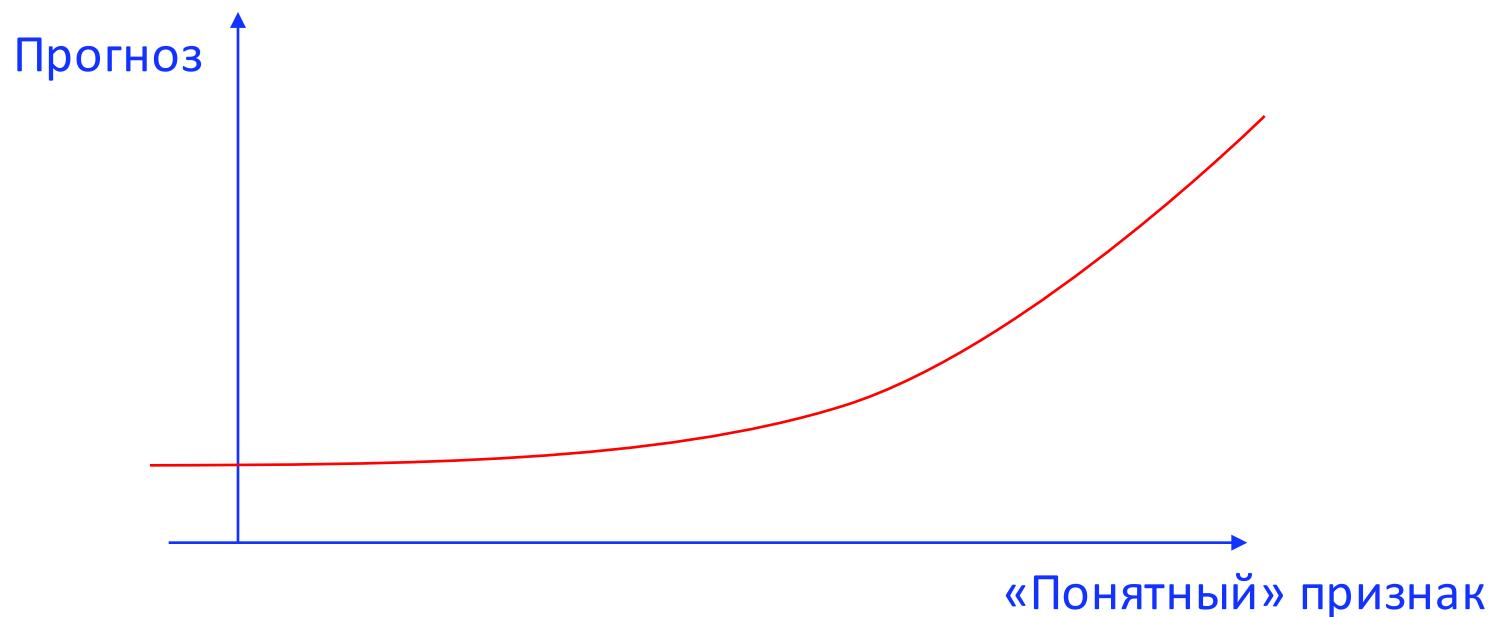
0.285714 Номер  
0.163265 Параметр 1  
0.122449 Высота  
0.102041 Параметр 4  
0.0816327 Параметр 5  
0.0816327 Вес  
0.0612245 Параметр 2

...

...

## Шаг 4: Анализ зависимости от признаков

Если зависимость от каких-то признаков должна иметь понятный вид, можем поменять их (построить «искусственные» примеры) и посмотреть, как ведет себя прогноз



## Шаг 5: Уменьшение разброса

- Вариант 1: нахождение допущенных ошибок
- Вариант 2: более устойчивые модели

# Онлайновая оценка качества

Допустим, на исторических данных качество алгоритма высокое, а будет ли оно высоким в реальности?

Идеи:

1. А/В тест
2. Оценка статзначимости результата

## A/B тест

1. Случайным образом делим клиентов на равные группы
2. Измеряем целевые метрики (например, доход с каждой группы клиентов) в каждой группе за длительный период времени
3. Получаем какое-то число для каждой группы
4. Что дальше?

# Статистическая значимость: пример



# Статистическая значимость: пример



Одна кривая отличается от других на 10%  
Но разбиение на самом деле – случайное

## Постановка задач

- I. Что прогнозировать и как использовать модель
- II. Как оценивать качество (онлайн и оффлайн)
- III. Выборка, признаки и модель

# Содержание лекции

## I. Напоминание изученного

- Стандартные задачи и модели
- Линейные модели
- Решающие деревья и ансамбли деревьев
- Оценка качества

## II. Проблемы и ошибки при анализе данных

## III. Ответы на вопросы

## **II. Ответы на вопросы**

## Постановка задач

- I. Что прогнозировать и как использовать модель
- II. Как оценивать качество (онлайн и оффлайн)
- III. Выборка, признаки и модель

## Возможные проблемы с таргетом

1. Решение об использовании прогнозов принимается теми, кто не понимает их смысл
2. В качестве таргета используются недостоверные данные
3. Таргет выбран «как-нибудь», чтобы проверить, можно ли прогнозировать «хоть что-то» и уверенностью, что другой таргет тоже получится прогнозировать
4. Есть «утечка» (leak) таргета
5. На новых данных изменился масштаб таргета

# Возможные проблемы с оценкой качества

1. Качество долгое время оптимизируется на одной и той же тестовой выборке
2. Качество оценивается на примерах, которые были в обучении
3. У метрики есть две разные записи и разные участники проекта имеют ввиду разное
4. При оценке качества модель заглядывает «в будущее»
5. Метрика качества не связана с желаемым эффектом
6. Оффлайн и онлайн качество связаны слабо

## Возможные проблемы с оценкой качества

7. Онлайн-качество не оценивается или не проводится корректный А/В тест
8. Решения принимаются по результатам не статзначимых измерений
9. Эксперименты останавливаются по достижению статзначимости
10. Обучающая и валидационная выборки непоказательны для оценки качества (пример с «галочкой»)

# Возможные проблемы с выборкой

1. Выборка слишком маленькая на обучении/тесте
2. Выборка строится по недостоверным данным
3. Логгирование данных существенно изменилось
4. Классы в выборке сильно несбалансированы
5. Соотношение классов в выборке сильно отличается от действительности
6. Смысл отдельных полей в данных понят неправильно
7. В отдельных полях есть выходы за допустимые диапазоны и другие ошибки

## Возможные проблемы с признаками

1. Категориальные признаки не используются (поленились кодировать)
2. Кодирование категориальных признаков приводит к переобучению
3. Признаки повторяют друг друга или сильно скоррелированы
4. В признаках «нет сигнала»
5. Признаков значительно больше, чем объектов
6. Есть признаки, по своему смыслу приводящие к переобучению (функции от таргета/номера объектов из выборки и т.п.)

## Возможные проблемы с моделью

1. Не заметили переобучение/недообучение модели
2. Выбрана модель, плохо работающая на данных данной природы (пример – решающее дерево на большом количестве разреженных признаков)
3. Ответы модели интерпретируются неправильно (например: прогнозы принимают за вероятности, хотя это не так)
4. Для построения модели выбран инструмент, плохо подходящий для использования в продакшене

# Признаки в анализе текстов

- Dataset: 20news\_groups
- Электронные письма, разбитые по 20 темам (классам)
- Попробуем придумать классификатор, который различает две темы:
  - **auto** и **politics.mideast**

# Извлечение текстовых признаков

- Пример письма 1:

From: carl\_f\_hoffman@cup.portal.com

Newsgroups: rec.autos

Subject: 1993 Infiniti G20

Message-ID: <78834@cup.portal.com>

Date: Mon, 5 Apr 93 07:36:47 PDT

Organization: The Portal System (TM)

Lines: 26

I am thinking about getting an Infiniti G20.

In consumer reports it is ranked high in many  
catagories including highest in reliability index for compact cars.  
(Mitsubishi Galant was second followed by Honda Accord)

# Извлечение текстовых признаков

- Пример письма 2:

From: Bob.Waldrop@f418.n104.z1.fidonet.org (Bob Waldrop)

Subject: Celebrate Liberty! 1993

Message-ID: <1993Apr5.201336.16132@dsd.es.com>

Followup-To:talk.politics.misc

Announcing... Announcing... Announcing... Announcing...

CELEBRATE LIBERTY!  
1993 LIBERTARIAN PARTY NATIONAL CONVENTION  
AND POLITICAL EXPO

THE MARRIOTT HOTEL AND THE SALT PALACE

SALT LAKE CITY, UTAH

# Текстовые признаки: bag-of-words



the world of **TOTAL**

**all about the company**

Our energy exploration, production, and distribution operations span the globe, with activities in more than 100 countries.

At TOTAL, we draw our greatest strength from our fast-growing oil and gas reserves. Our strategic emphasis on natural gas provides a strong position in a rapidly expanding market.

Our expanding refining and marketing operations in Asia and the Mediterranean Basin complement already solid positions in Europe, Africa, and the U.S.

Our growing specialty chemicals sector adds balance and profit to the core energy business.

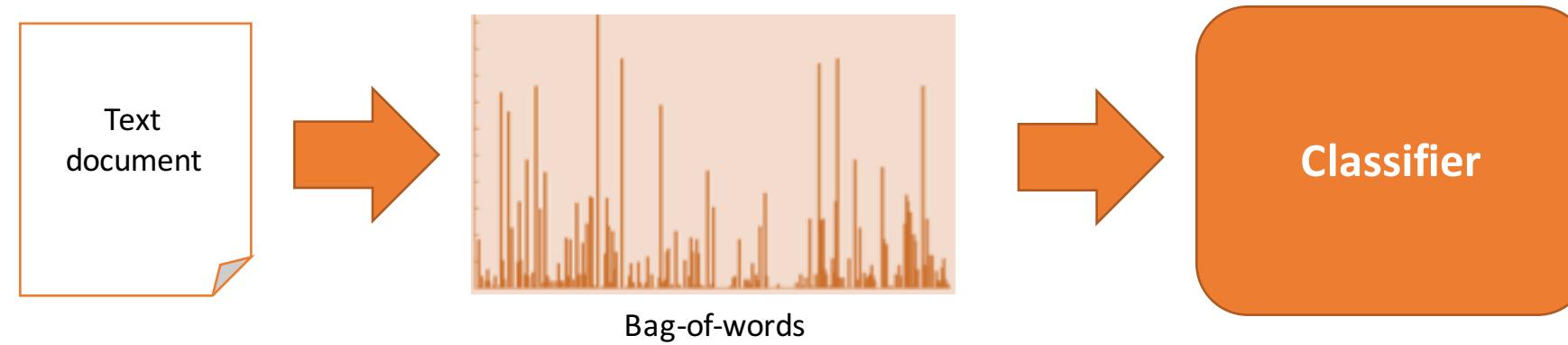
**All About The Company**

- Global Activities
- Corporate Structure
- TOTAL's Story
- Upstream Strategy
- Downstream Strategy
- Chemicals Strategy
- TOTAL Foundation
- Homepage



aardvark	0
about	2
all	2
Africa	1
apple	0
anxious	0
...	
gas	1
...	
oil	1
...	
Zaire	0

# Простой классификатор текстов



# Взвешивание частот слов в текстах

Term Frequency

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Inverse Document Frequency

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|}$$

# Про байесовскую классификацию

- Байесовский классификатор:

$$a(x) = \operatorname{argmax}_y P(x|y)P(y)$$

- **Обучение:** оценить по выборке  $P(x|y)$  и  $P(y)$
- Оценивать  $P(x|y)$  как функцию **многих** переменных **затруднительно** –  
нужно много данных

# Наивный байесовский классификатор

1. Байесовский классификатор:

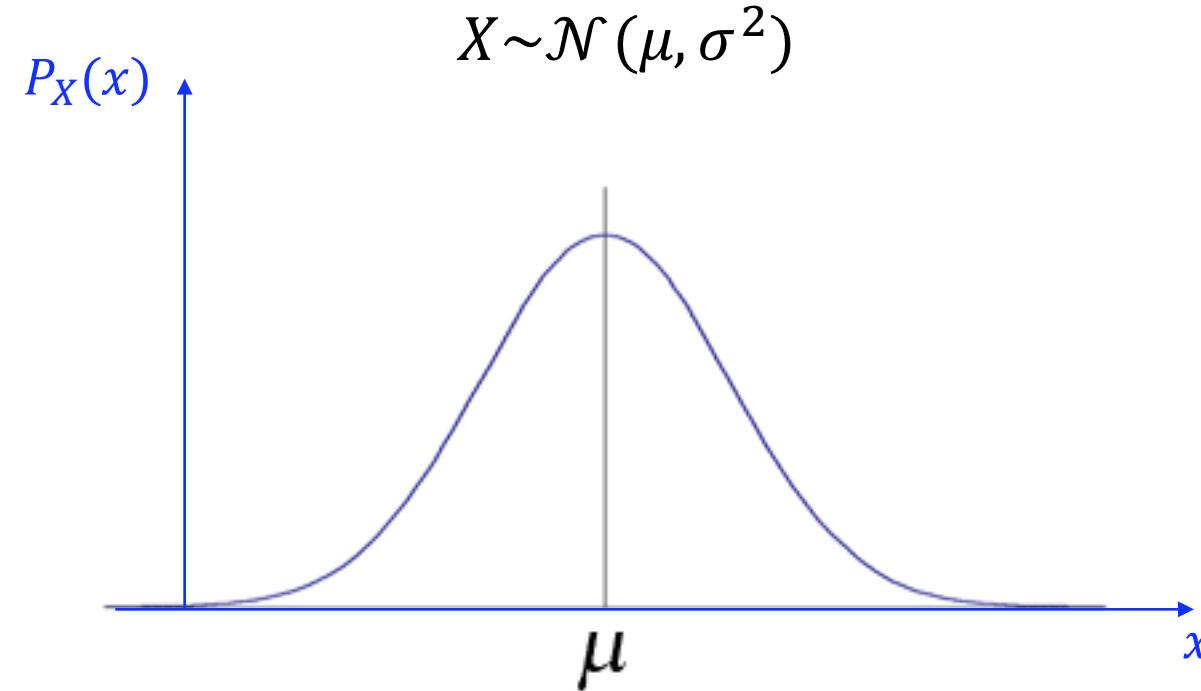
$$a(x) = \operatorname{argmax}_y P(y|x) = \operatorname{argmax}_y P(x|y) P(y)$$

2. С «наивной» гипотезой:

$$P(x|y) = P(x_{(1)}|y) P(x_{(2)}|y) \dots P(x_{(N)}|y)$$

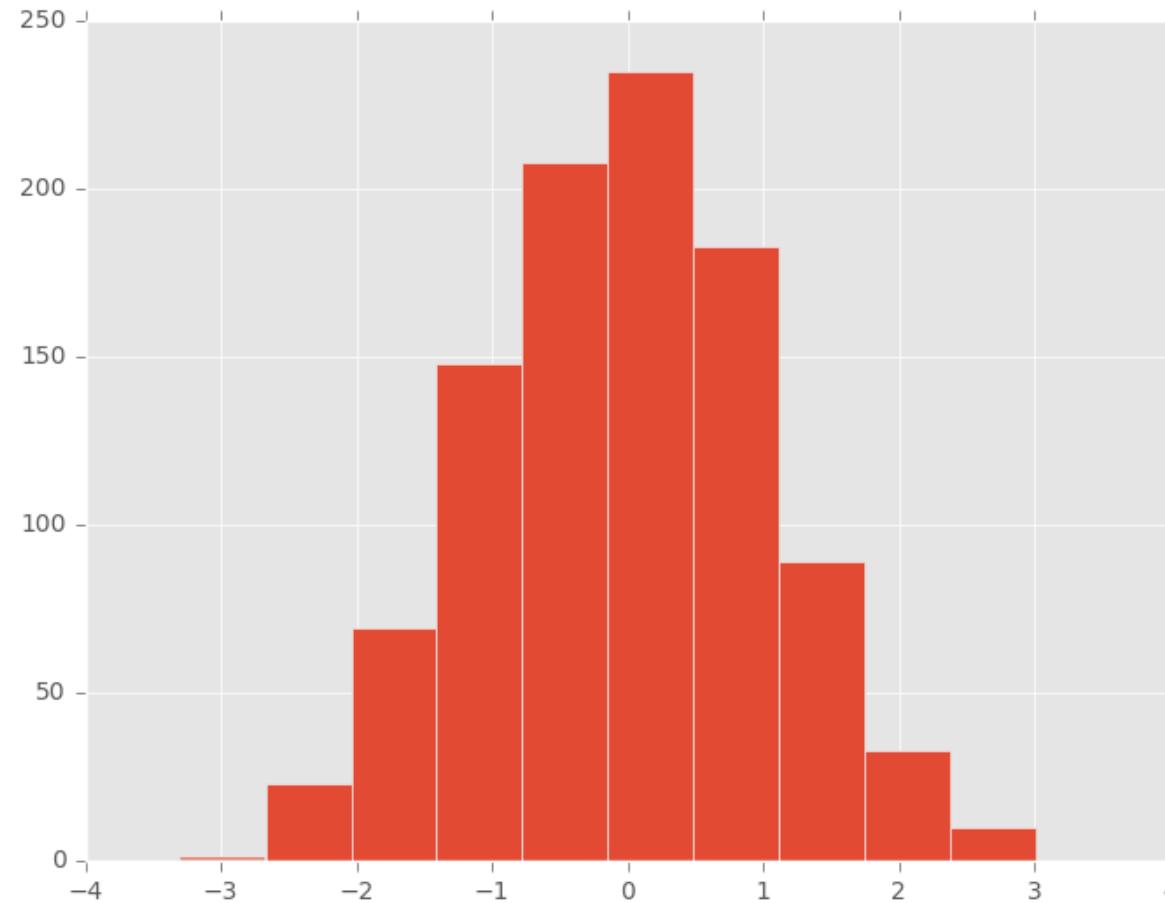
$x_{(k)}$  – k-ый признак объекта x

## Пример: нормальное распределение



$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

# Пример: нормальное распределение



## Пример: нормальное распределение

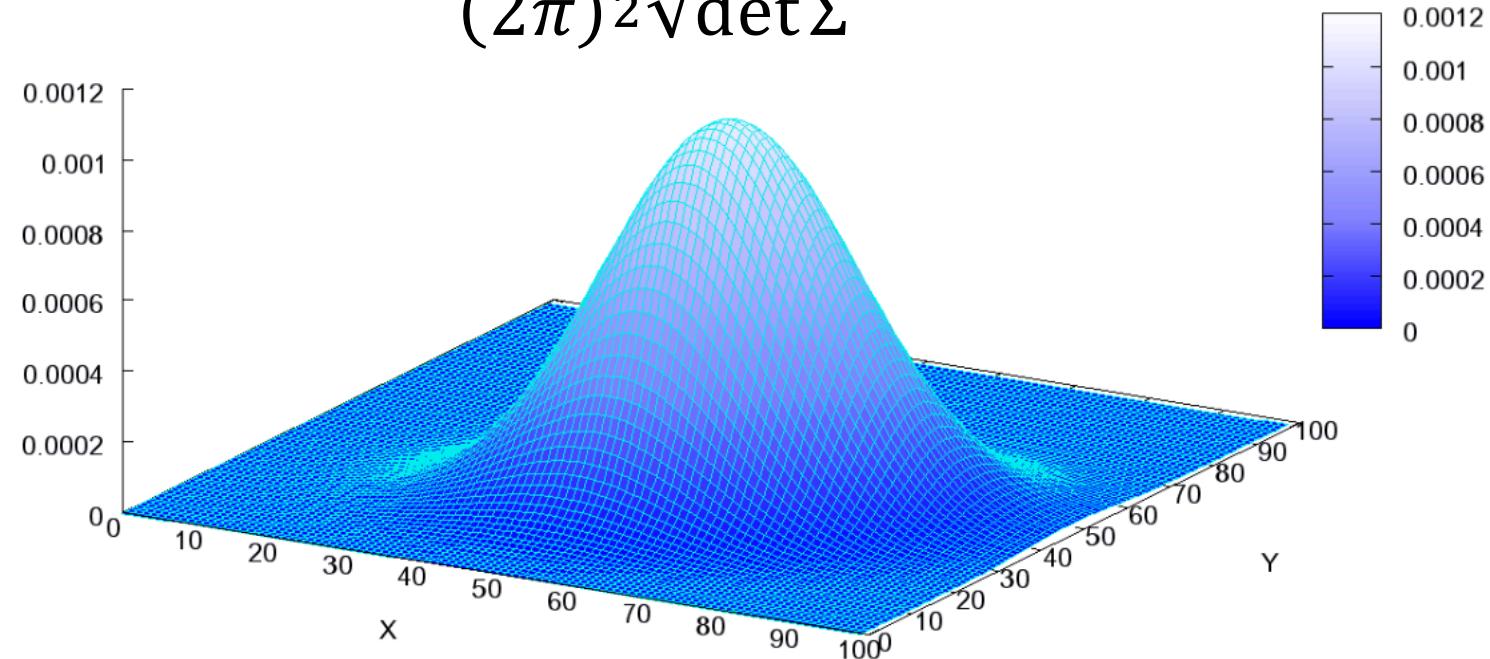
$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N x_i \quad \sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \hat{\mu})^2$$

несмешенный вариант оценки для дисперсии:

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \hat{\mu})^2$$

# Про нормальный дискриминантный анализ

$$p(x) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}$$



Параметры: вектор средних  $\mu$  и матрица ковариаций  $\Sigma$

## Нормальный дискриминантный анализ

Он же – квадратичный дискриминантный анализ (QDA)

$$a(x) = \operatorname{argmax}_y p(x|y)P(y)$$

$$p(x|y) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma_y}} e^{-\frac{1}{2}(x-\mu_y)^T \Sigma_y^{-1} (x-\mu_y)}$$

# Нормальный дискриминантный анализ

**Обучение:**

Для каждого класса  $y$  читаем по выборке:

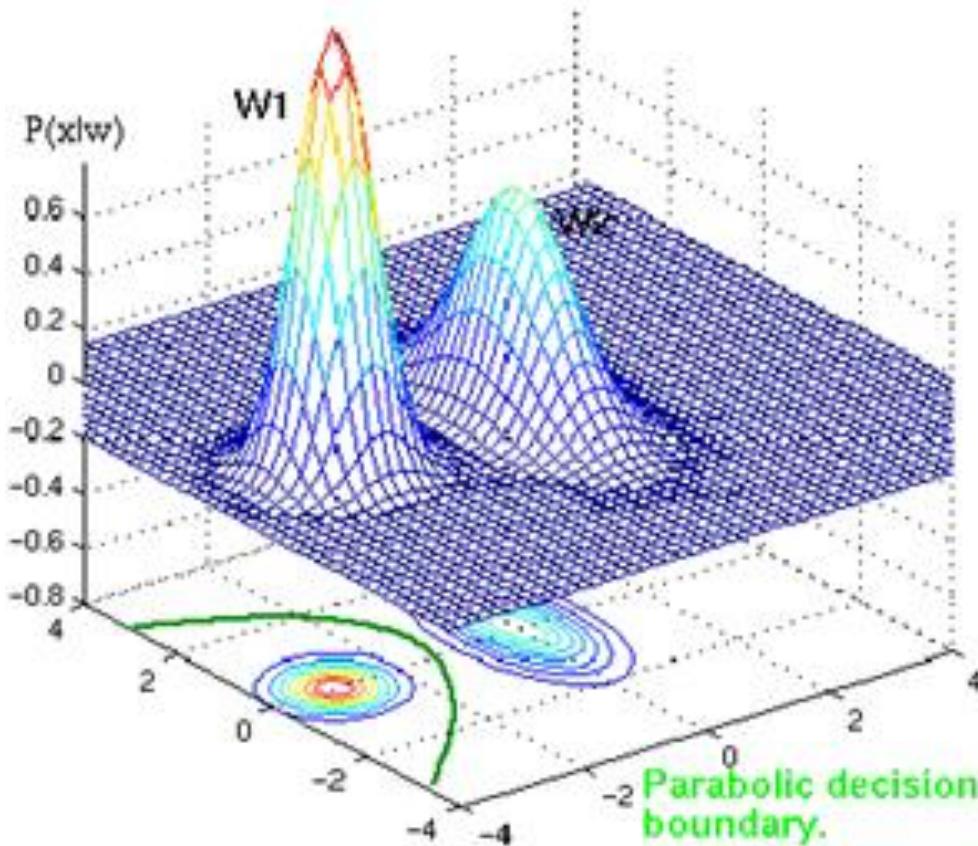
$$\mu_y = \frac{1}{l_y} \sum_{i:y_i=y} x_i \quad \Sigma_y = \frac{1}{l_y - 1} \sum_{i:y_i=y} (x_i - \mu_y)(x_i - \mu_y)^T$$

**Применение:**

На объекте с вектором признаков  $x$  отвечаем классом  $y$ , для которого наибольшее значение принимает следующее выражение:

$$\frac{l_y}{l} \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma_y}} e^{-\frac{1}{2}(x-\mu_y)^T \Sigma_y^{-1} (x-\mu_y)}$$

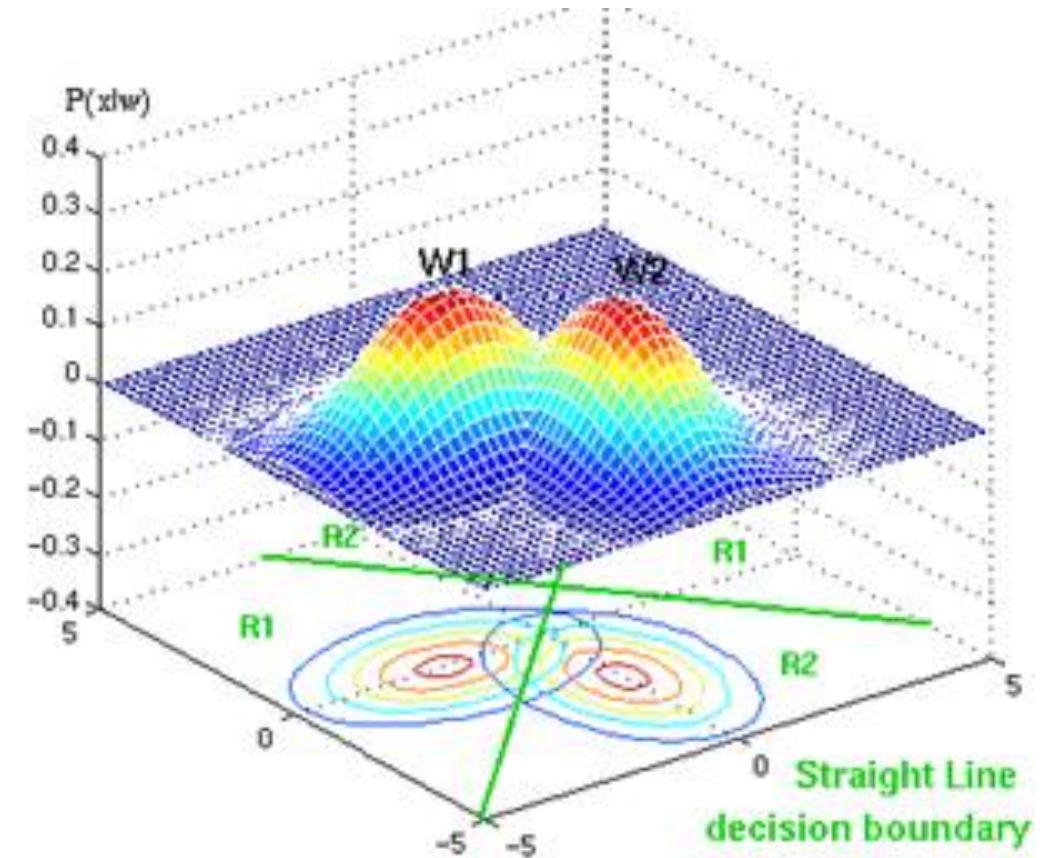
# Нормальный дискриминантный анализ



Обычно разделяющая  
поверхность имеет  
параболическую,  
гиперболическую или  
эллиптическую форму

# Вырождение в две прямые

В случае двух одинаковых по форме гауссиан, по-разному ориентированных в пространстве, разделяющая поверхность может принять вид двух прямых



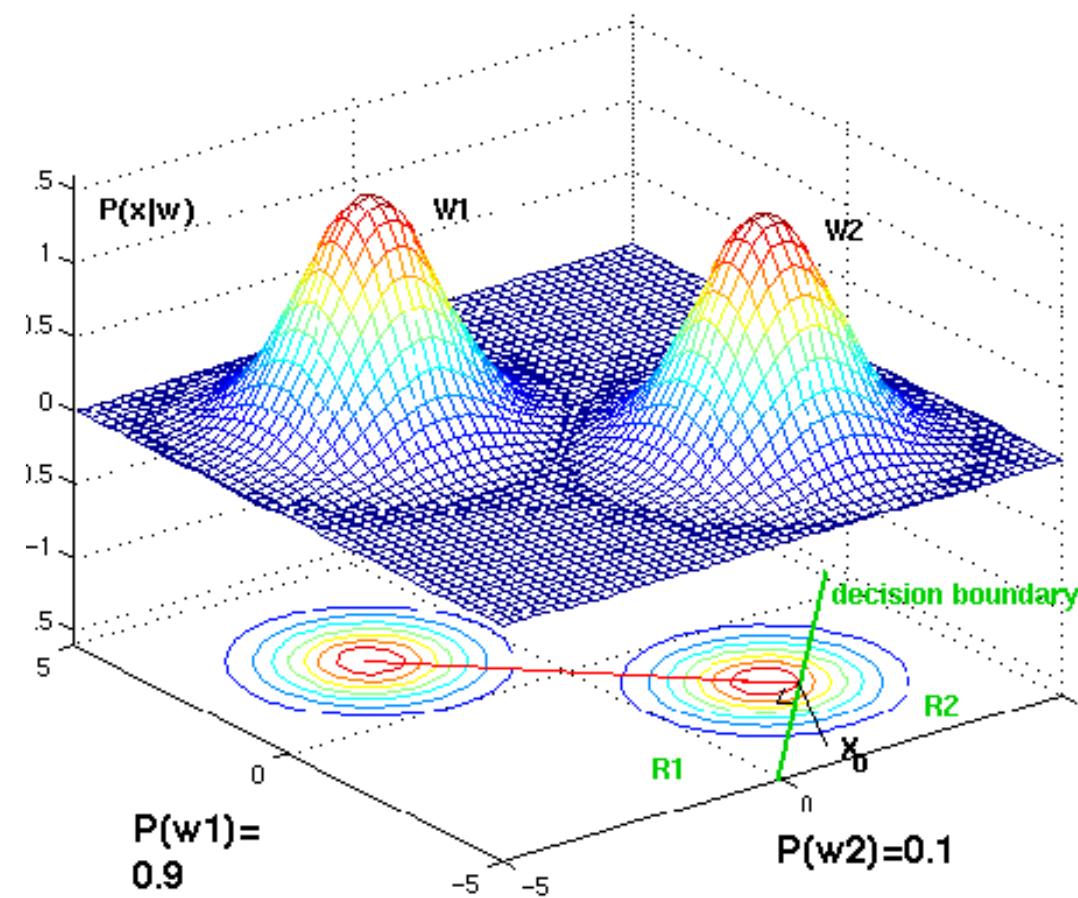
# Линейный дискриминантный анализ

Он же – линейный дискриминант Фишера:

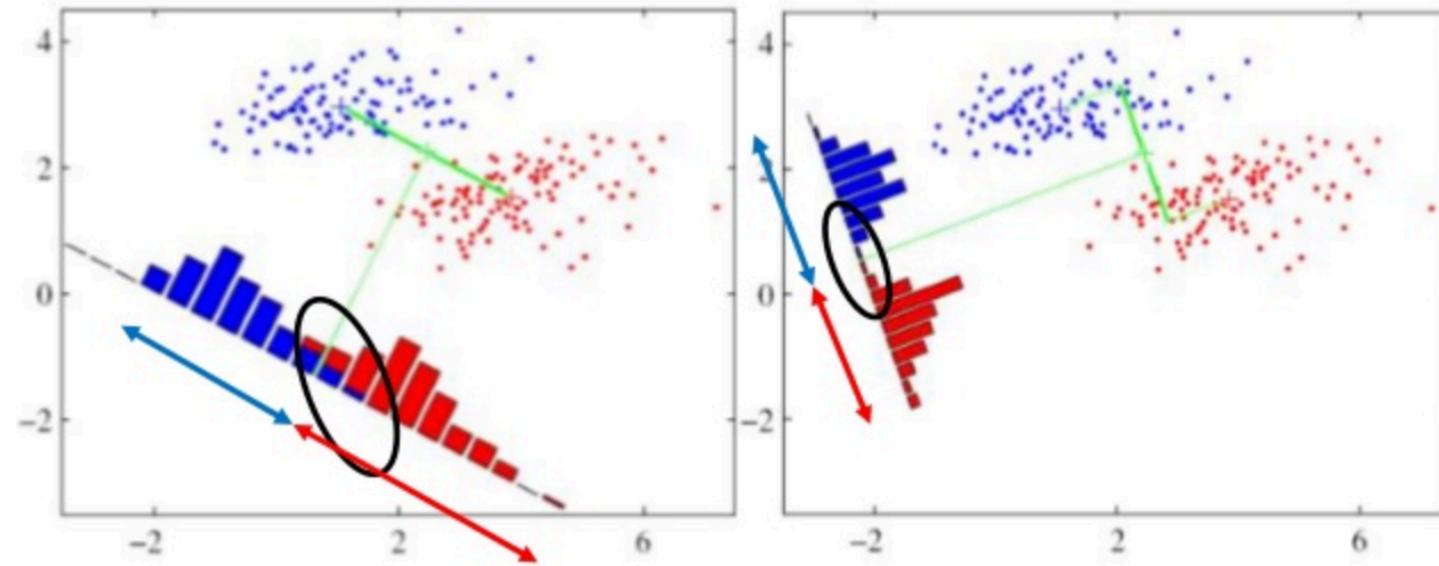
$$a(x) = \operatorname{argmax}_y p(x|y)P(y)$$

$$p(x|y) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma}} e^{-\frac{1}{2}(x-\mu_y)^T \Sigma^{-1} (x-\mu_y)}$$

# Линейный дискриминантный анализ



## LDA: другой вариант получения



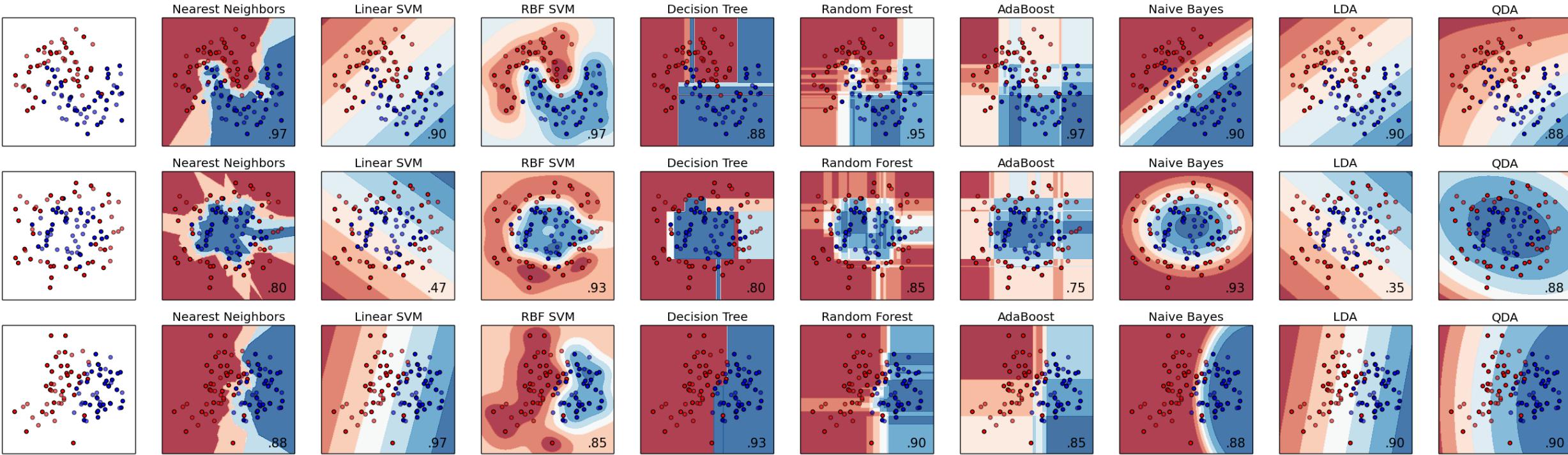
$$\frac{(m_1 - m_2)^2}{\sigma_1^2 + \sigma_2^2} \rightarrow \min$$

## Упражнение

Как будет выглядеть обучение и применение алгоритма в случае LDA?

(Переписать аналогично QDA)

# Сравнение классификаторов



[http://scikit-learn.org/0.16/auto\\_examples/classification/plot\\_classifier\\_comparison.html#example-classification-plot-classifier-comparison-py](http://scikit-learn.org/0.16/auto_examples/classification/plot_classifier_comparison.html#example-classification-plot-classifier-comparison-py)

# Резюме

## I. Напоминание изученного

- Стандартные задачи и модели
- Линейные модели
- Решающие деревья и ансамбли деревьев
- Оценка качества

## II. Ответы на вопросы

## Что нас ждет дальше

- Обучение без учителя (unsupervised learning)
- Прикладные области (рекомендации, предиктивная аналитика, анализ текстов)
- Нейронные сети и deep learning

# Спасибо за внимание



[info@applieddatascience.ru](mailto:info@applieddatascience.ru)



[https://t.me/joinchat/B10lThC96v0BQCvs\\_joNew](https://t.me/joinchat/B10lThC96v0BQCvs_joNew)



[https://github.com/vkantor/ml2018jan\\_feb](https://github.com/vkantor/ml2018jan_feb)