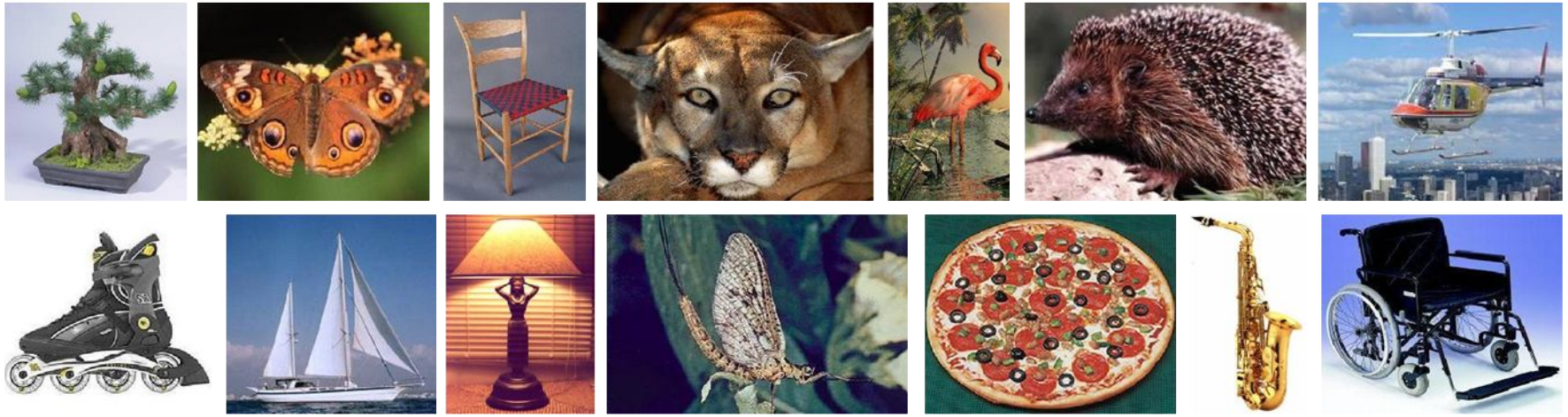
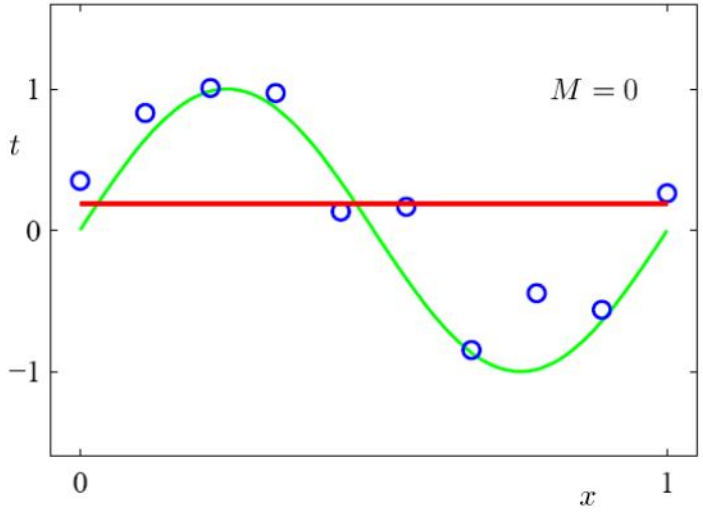
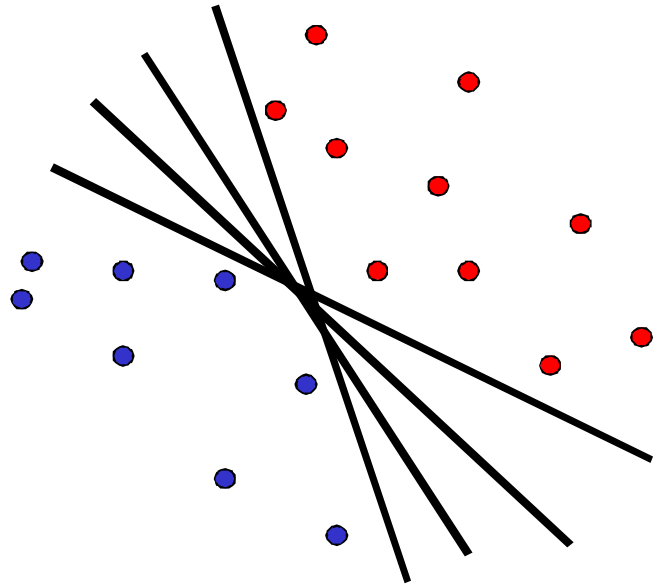
Классификация изображений







Slides adapted from Fei-Fei Li, Rob Fergus, Antonio Torralba, Jean Ponce and Svetlana Lazebnik

Общая информация

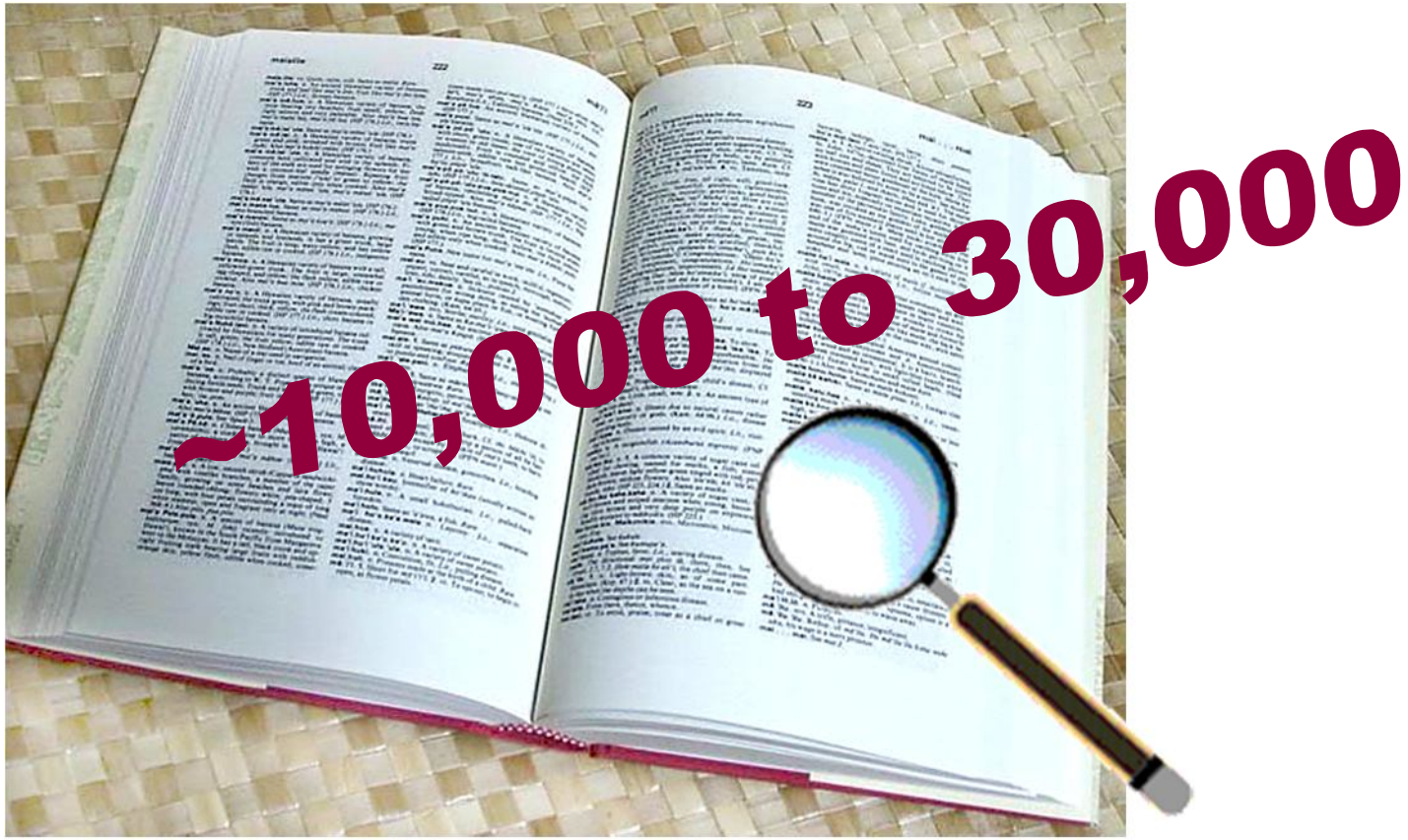
Microsoft Research

Этот курс подготовлен и читается при поддержке Microsoft Research

Microsoft Research

- Страница курса http://courses.graphicon.ru/main/vision

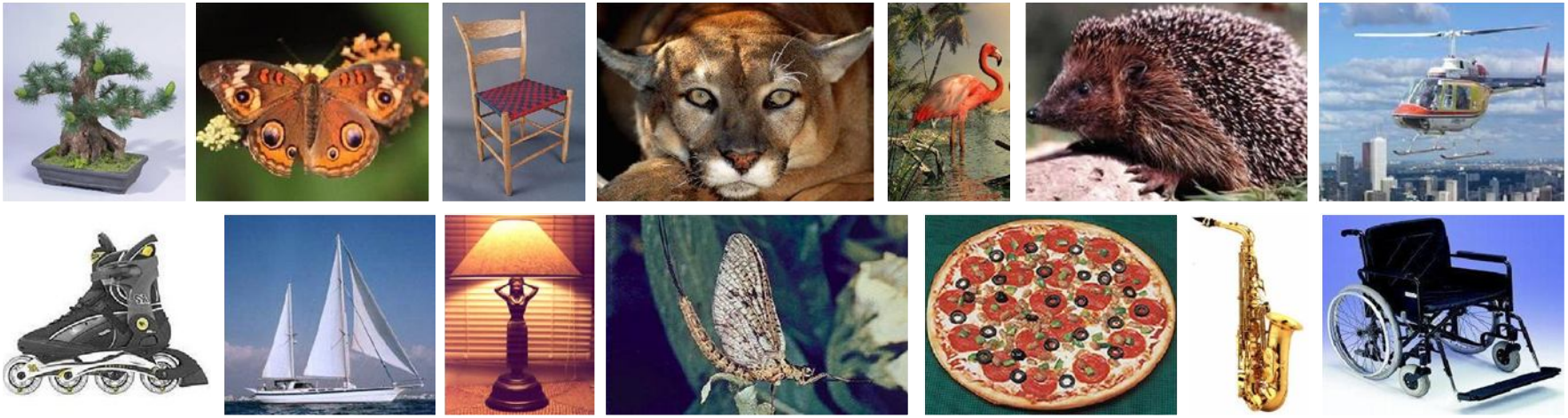
Категории объектов



• Объектов бесконечное множество, а сколько существует классов объектов?
• 1500-3000 основных существительных, ~10 подкатегорий
• Выделяют и больше – до 100 000 категорий

Biederman 1987

Классификация изображений

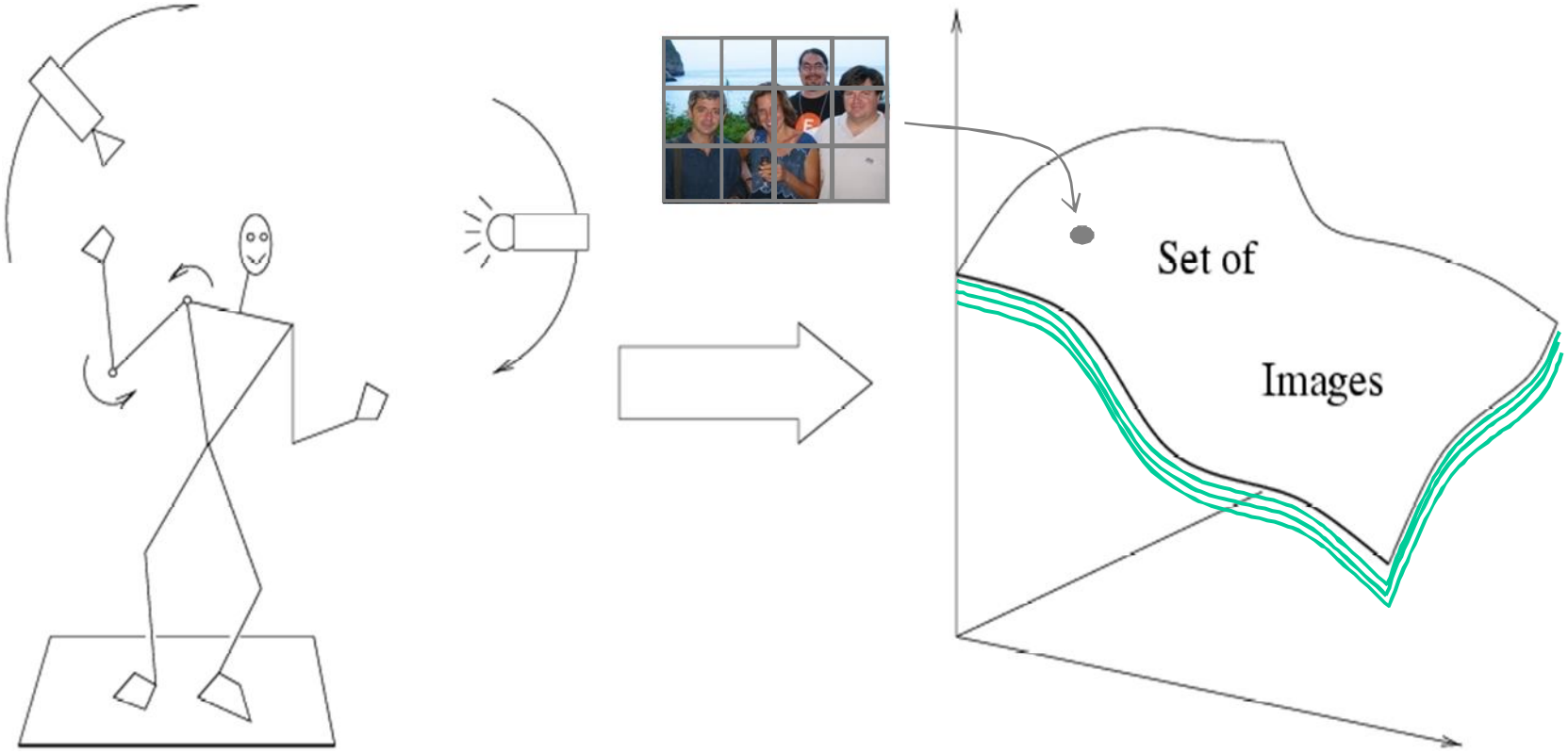


- Будем рассматривать изображения, на которых запечатлён один объект
  - Или определённого «класса», например «вечер», «пейзаж»

- Задача «классификации изображений»
  - Вопрос: есть ли на изображении объект заданного класса?
  - Вопрос: относится ли изображение к заданному классу?

- Есть и более сложные постановки задачи распознавания изображений

Изменчивость изображений



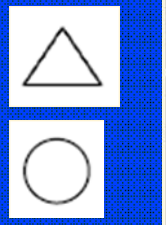
Внешние факторы:

Положение камеры
Освещение
Внутренние параметры

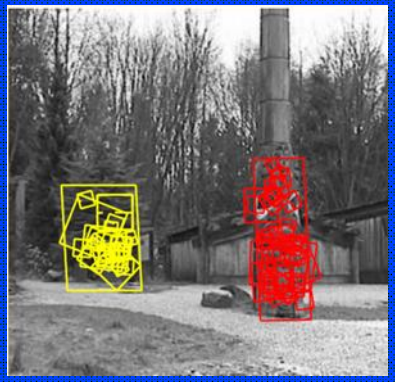
Внутренние факторы:

Внутриклассовая изменчивость

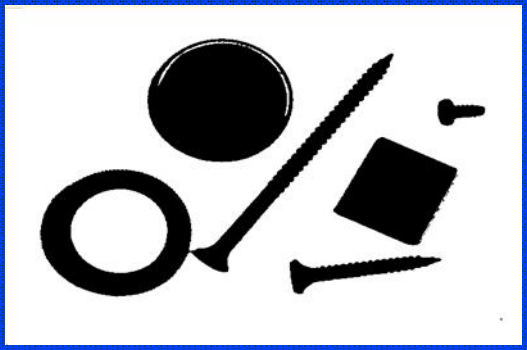
Ранее



Поиск шаблонов



Использование локальных особенностей

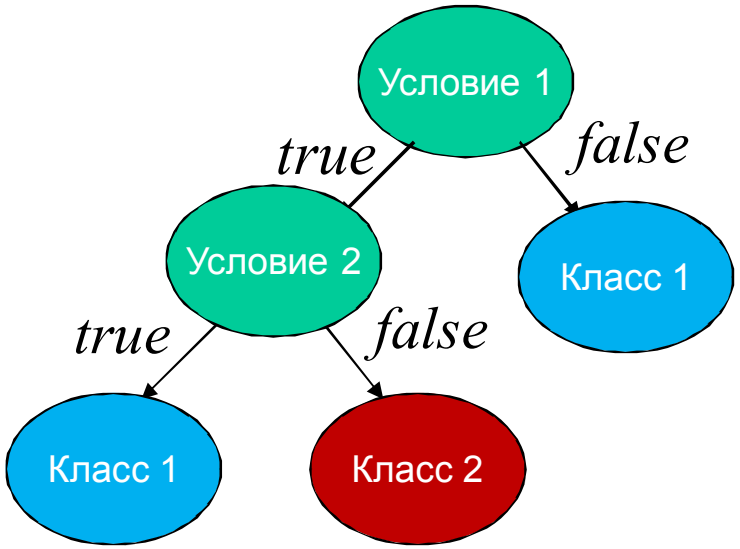


Инвариантные признаки для выделенных сегментов

- Все эти методы нашли своё применение, но решить задачу распознавания не получилось

Что нам мешает?

- Все правила подбираются вручную

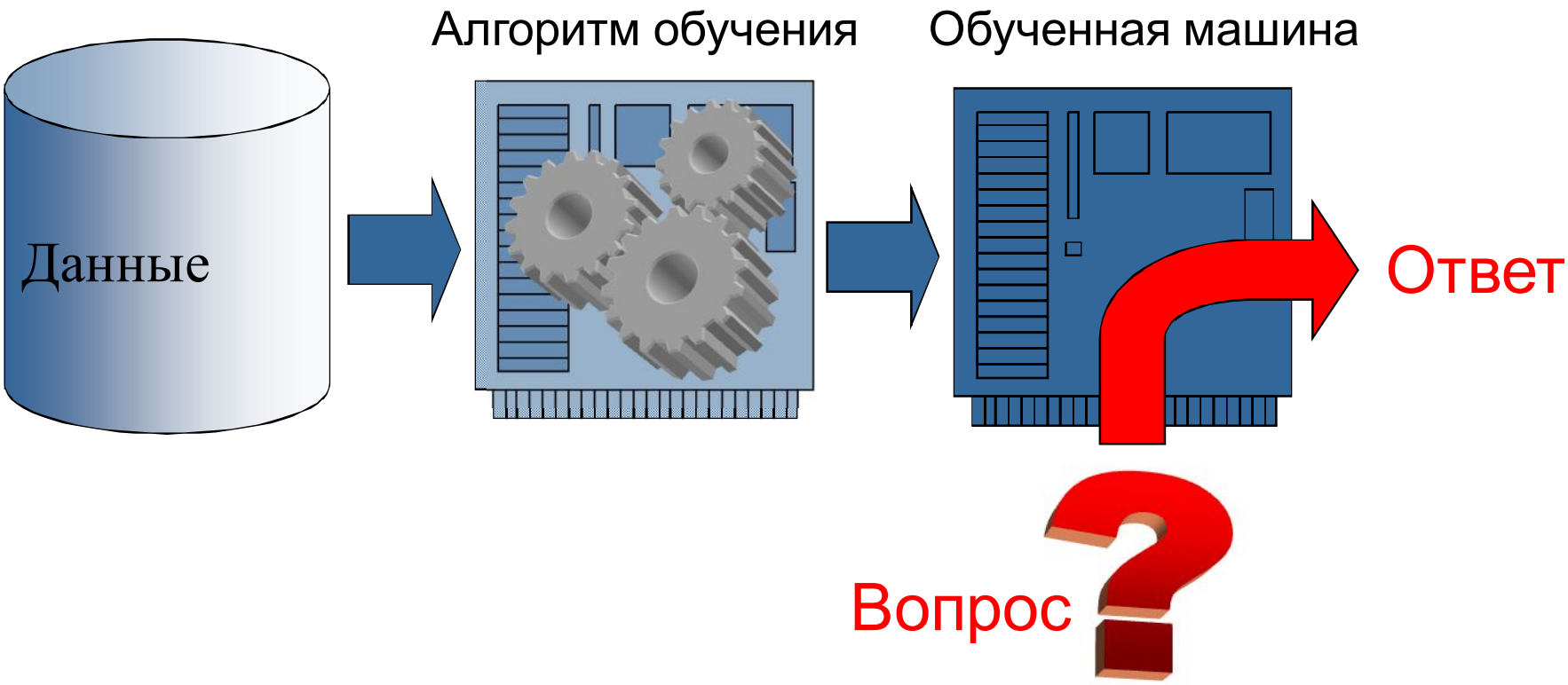


Строим вручную дерево решений

- Что в результате:
  - Можем перебрать немного комбинаций
  - Плохо умеем подбирать пороги
  - Можно использовать только «осмысленные» признаки
  - Маленькое количество признаков

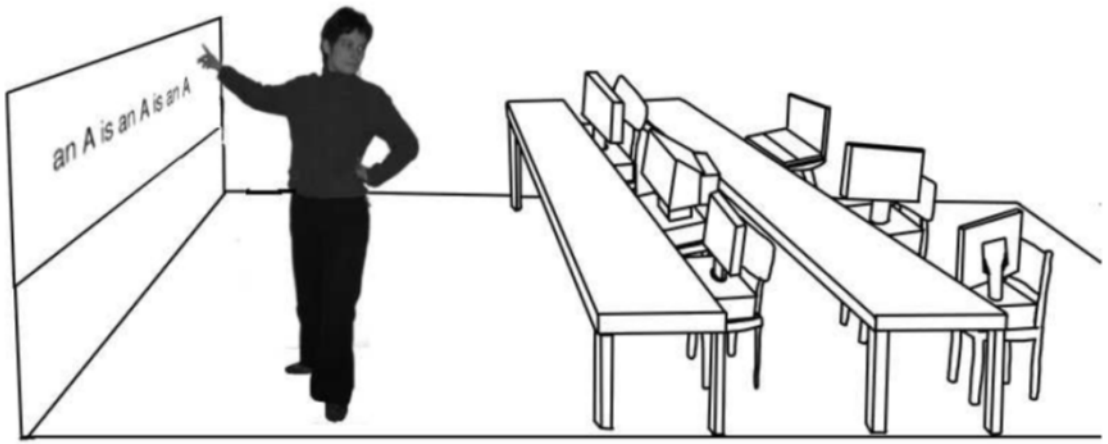
Что нам мешает?

- Что бы нам хотелось?



Что такое машинное обучение?

- Обучение ≠ «заучивание наизусть»
  - Заучить наизусть – для машины не проблема
  - Мы хотим научить машину делать выводы!
  - Машина должна корректно работать на новых данных, которые мы ей раньше не давали
  - По конечному набору обучающих данных машина должна научиться делать выводы на новых данных



Microsoft Research

Машинное обучение!

- Разные методы исследовались много лет
  - Нейронные сети…

- В середине 90х – начале 2000х было предложено несколько мощных методов
- Их появление привело к взрывному прогрессу в области анализа данных и компьютерного зрения в частности

- На примере ряда задач компьютерного зрения мы рассмотрим основные классы методов распознавания с машинным обучением

Microsoft Research

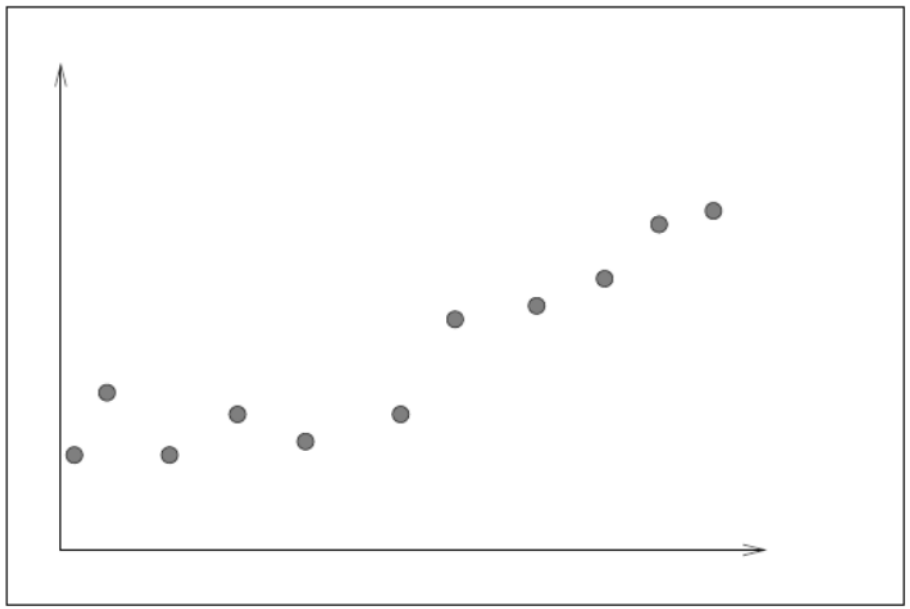
План

- Рассмотрим общую постановку задачи классификации и один из ключевых методов
- Затем рассмотрим один из этапных методов классификации изображений

Microsoft Research

Задача машинного обучения

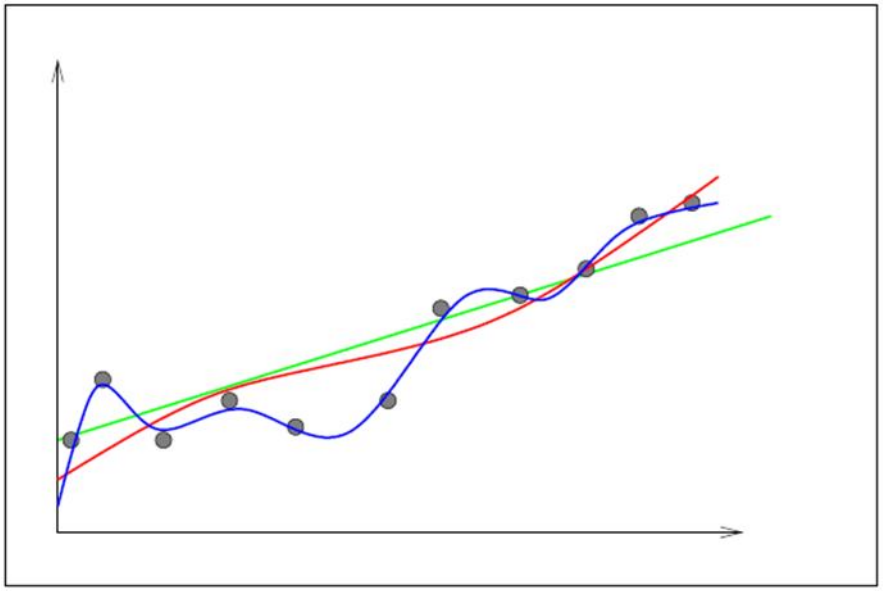
- В нашем распоряжении есть **конечное число** данных – обучающая выборка
- Каждый элемент описываем набором признаков *x* («вектор-признак»)
- Для каждого вектора параметров *x* известен ответ *y*



Microsoft Research

Задача машинного обучения

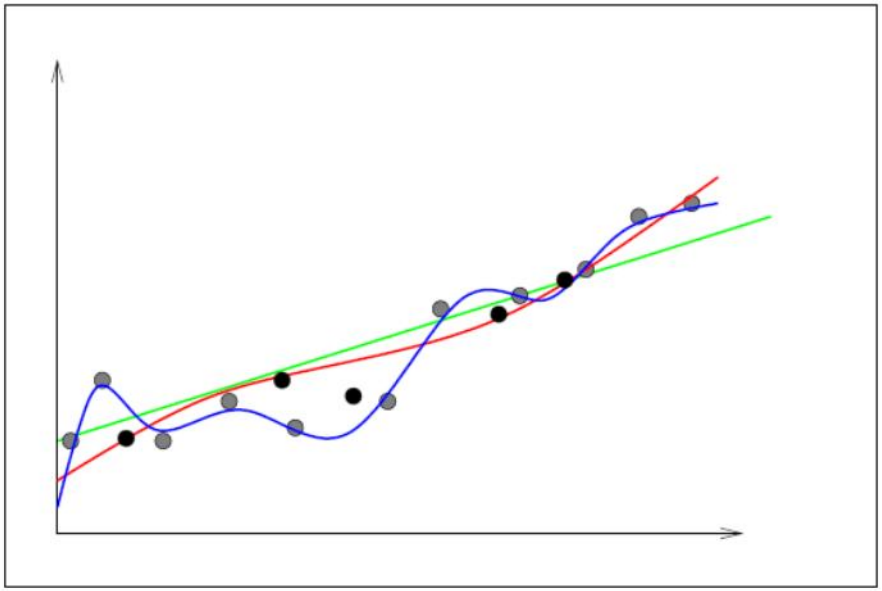
- Требуется сконструировать функцию y=***f(x)*** от вектора признаков ***x***, которое выдает ответ *y* для любого возможного наблюдения ***x***



Microsoft Research

Задача машинного обучения

- Построенное правило должно хорошо работать на новых данных



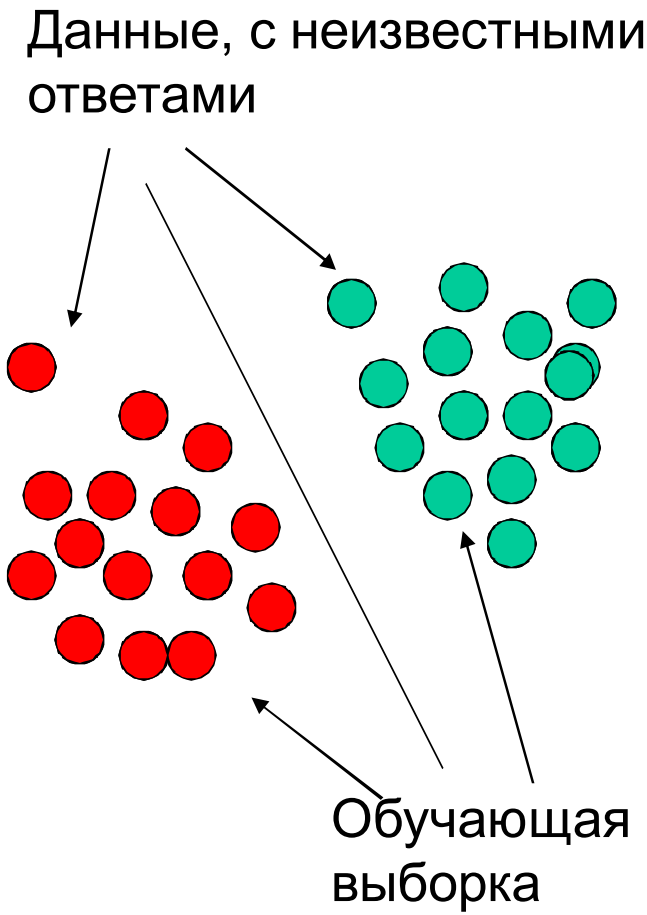
Обучение ≠ «заучивание наизусть»

- заучить наизусть – для машины не проблема
- мы хотим научить машину делать выводы на новых данных!

Статистические основы

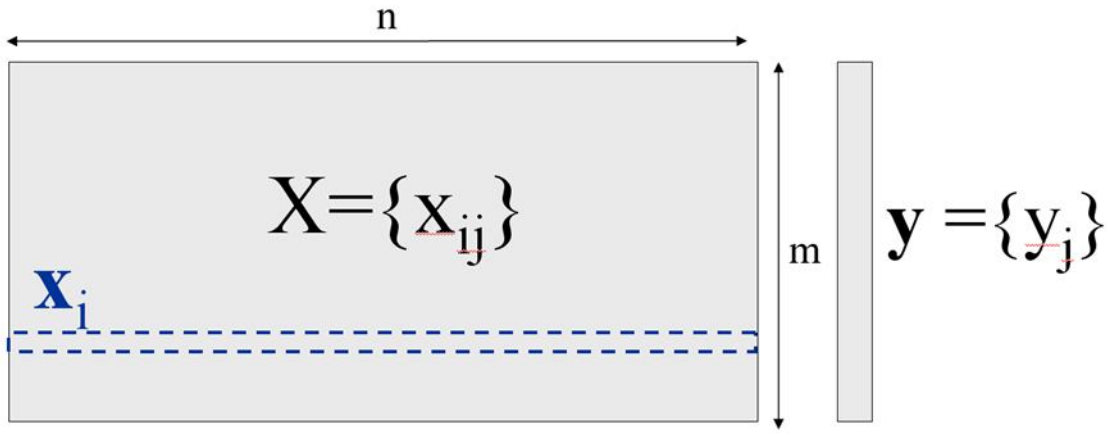
Microsoft Research

- Нас интересует качество работы алгоритма на новых данных
  - надо связать имеющиеся данные с теми, которые придется обрабатывать в будущем
- Выход: использование теории вероятностей и мат. статистики
  - Значения признаков, внутренние состояния системы считаем случайными величинами
  - Будем считать, что имеющиеся данные и данные, которые придется обрабатывать в будущем одинаково распределены



Обучающая выборка

- Каждое наблюдение будем представлять в виде вектора признаков $X=\{x_i\}$ и известного ответа (выхода) $y=\{y_j\}$



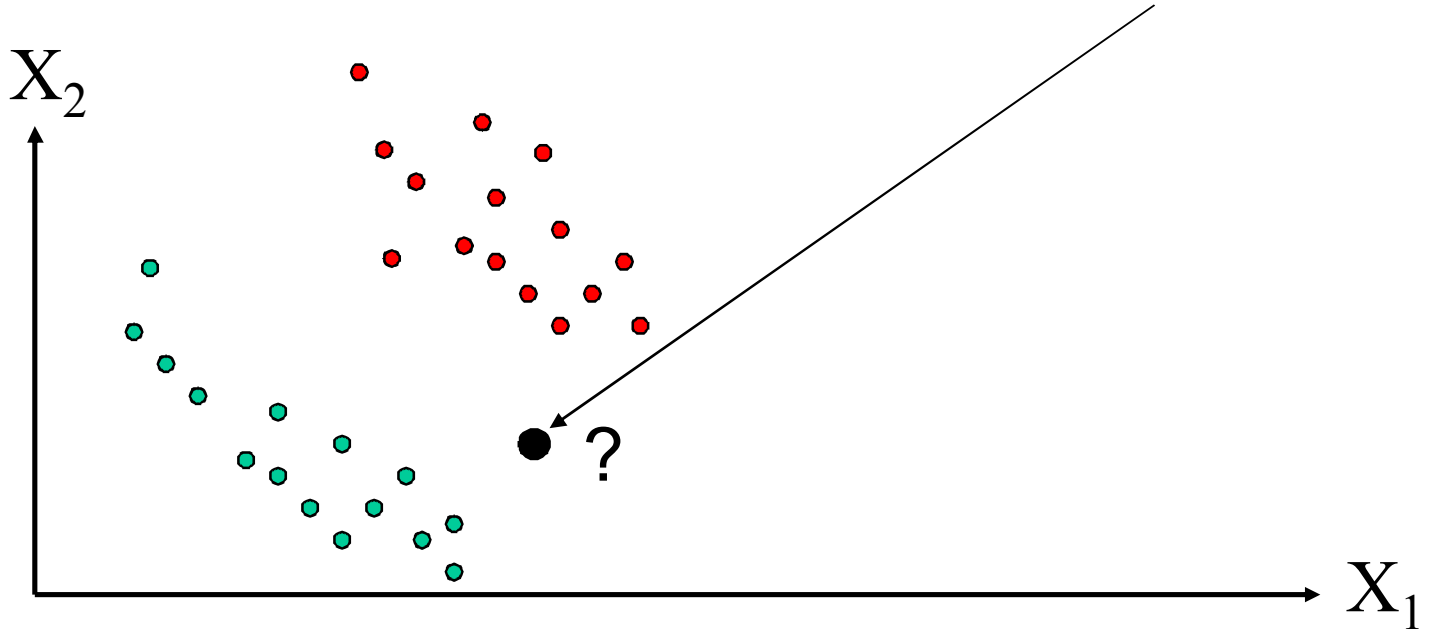
Бинарная классификация

- Дана обучающая выборка

$$X_m = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_m, y_m)\} \quad (\mathbf{x}_i, y_i) \in \mathbf{R}^{\mathbf{m}} \times Y, \; Y = \{-1, +1\}$$

- Объекты независимы и взяты из некоторого неизвестного распределения $(\mathbf{x}_i, y_i) \square \mathrm{P}(\mathbf{x}, y)$

- Цель: для всех новых значений ***x*** оценить значение argmax $\mathrm{P}(y \mid \mathbf{x})$



Многоклассовая классификация

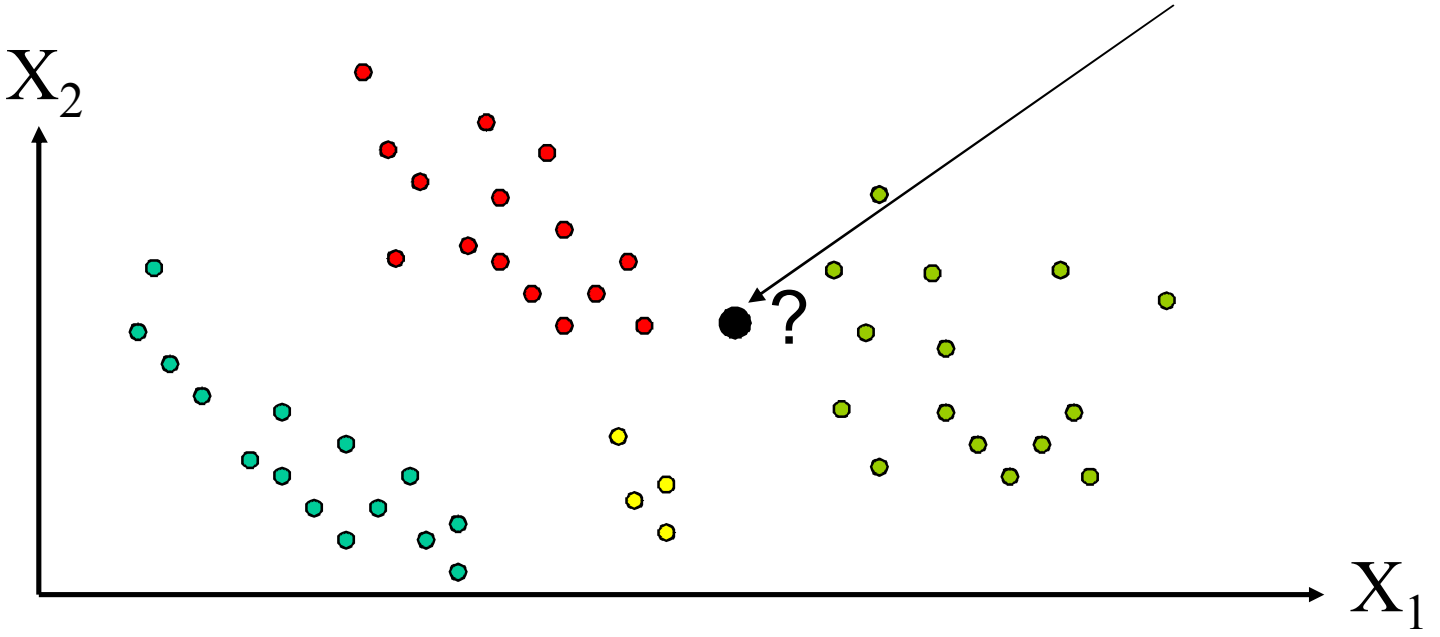
- Дана обучающая выборка

$$X_m = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_m, y_m)\} \quad (\mathbf{x}_i, y_i) \in \mathbf{R}^m \times Y, \; Y = \{1, ..., K\}$$

- Объекты независимы и взяты из некоторого неизвестного распределения

$$(\mathbf{x}_i, y_i) \sim P(\mathbf{x}, y)$$

- цель: для всех новых значений X оценить значения $P(y \mid \mathbf{x})$



Регрессия

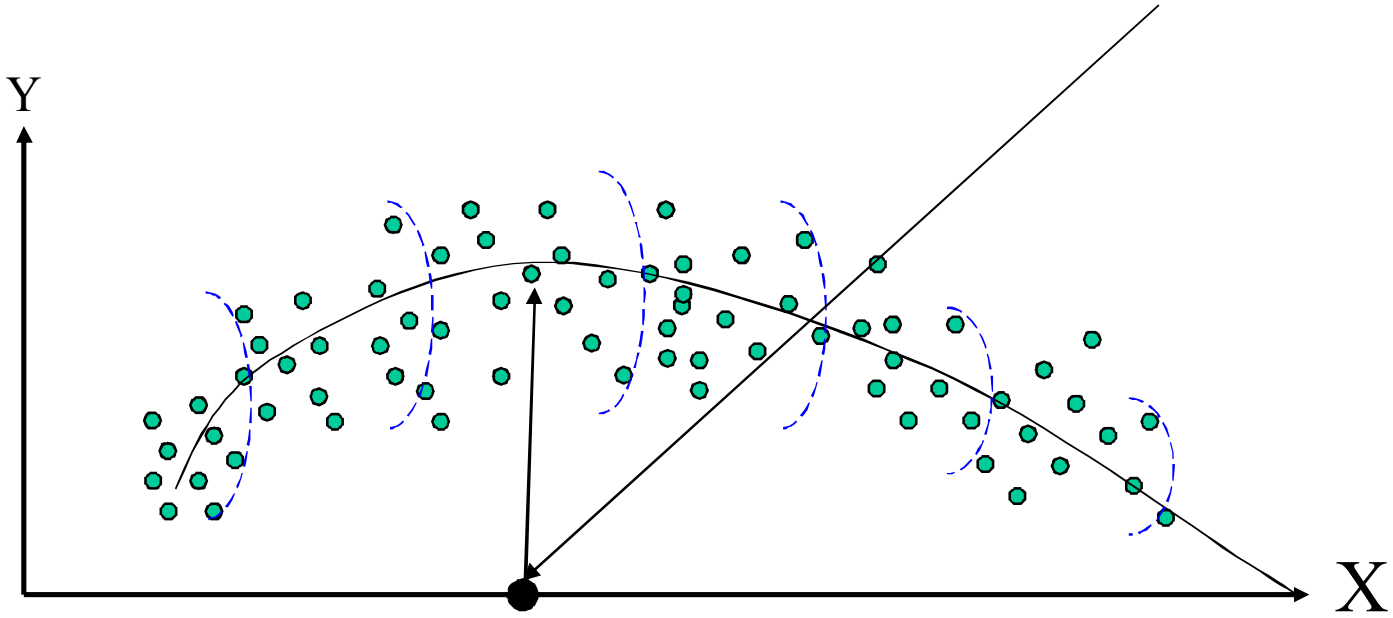
- Дана обучающая выборка

$$X_m = \left\{ \left(\mathbf{x}_1, y_1 \right), ..., \left(\mathbf{x}_m, y_m \right) \right\} \qquad (\mathbf{x}_i, y_i) \in \mathbf{R}^{\mathbf{m}} \times Y, \; Y = \mathbf{R}$$

- Объекты независимы и взяты из некоторого неизвестного распределения

$$(\mathbf{x}_i, y_i) \; \square \; \mathrm{P}(\mathbf{x}, \; y)$$

- Цель: для всех новых значений X оценить значение: $\mathrm{E}\left(y \mid \mathbf{x} \right)$

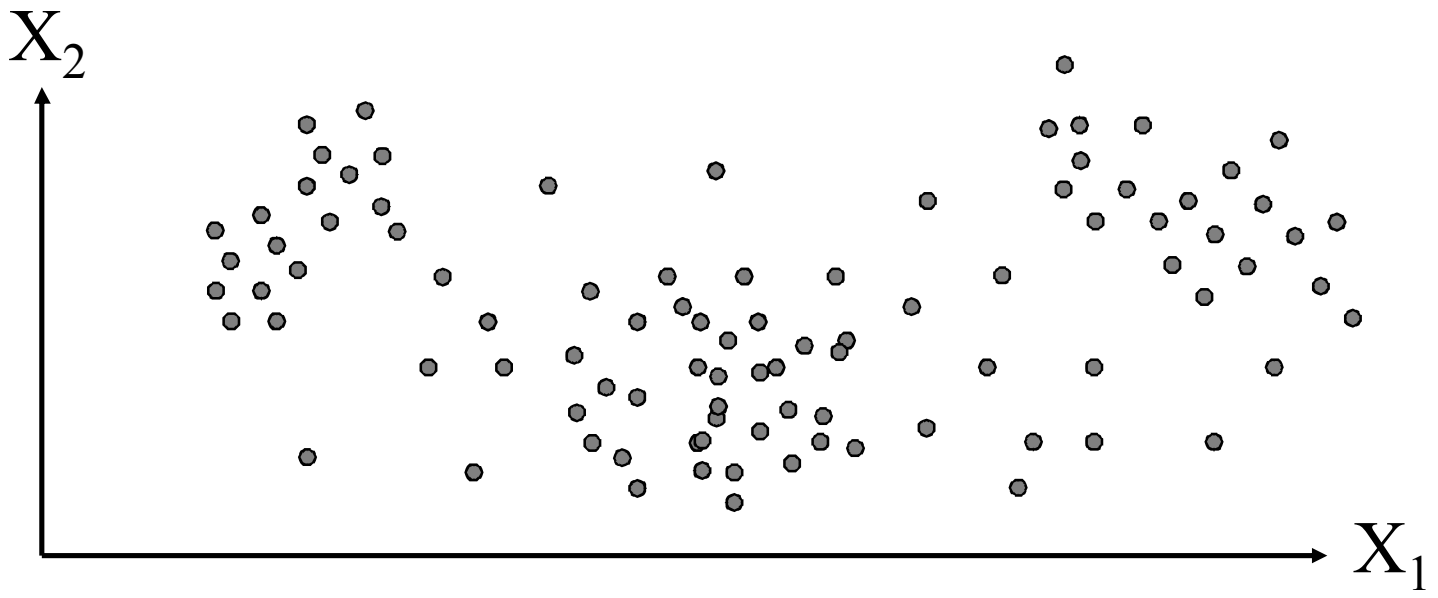


Microsoft Research

Кластеризация

- Дана обучающая выборка
$$X_m = \{\mathbf{x}_1, ..., \mathbf{x}_m\} \qquad \mathbf{x}_i \in \mathbf{R}^{\mathbf{m}}$$
- Объекты независимы и взяты из некоторого неизвестного распределения $\mathbf{x}_i \sim P(\mathbf{x})$

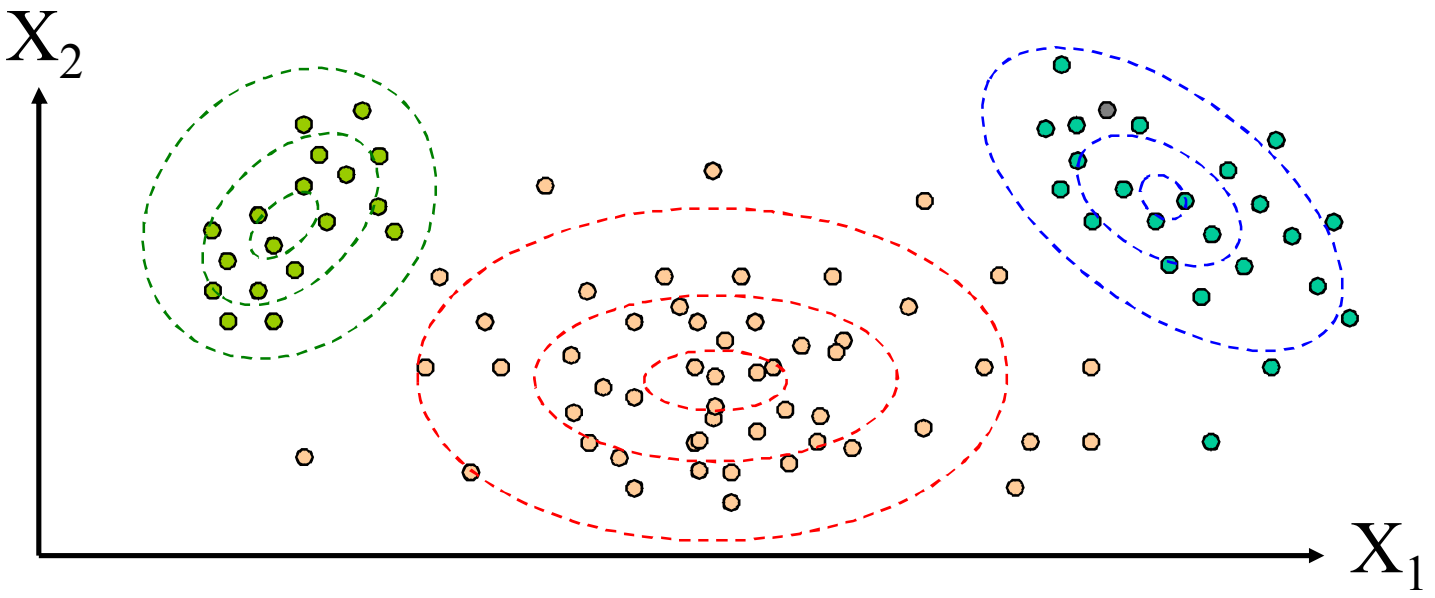
- Разбить множество объектов на классы (кластеры) на основе некоторой меры сходства объёктов



Кластеризация

- Дана обучающая выборка
$$X_m = \{\mathbf{x}_1, ..., \mathbf{x}_m\} \quad \mathbf{x}_i \in \mathbf{R}^m$$
- Объекты независимы и взяты из некоторого неизвестного распределения $\mathbf{x}_i \sim P(\mathbf{x})$

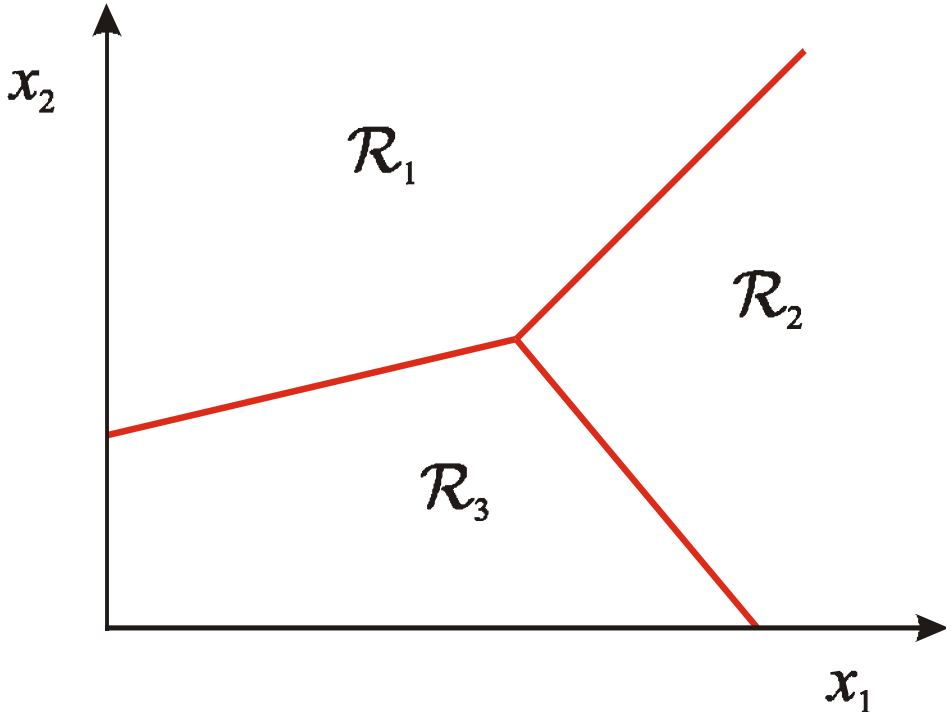
- цель: для всех новых значений *x* оценить значения $P(\mathbf{x})$



Классификация

- Хотим построить функцию y=f(x) – *решающее правило или классификатор*
- Любое *решающее правило* делит пространство на *решающие регионы* разделенные *решающими границами*

Decision



Microsoft Research

Классификация

- Будем выбирать функции ***f*** из **параметрического семейства *F*** (т.е. будем выбирать подходящий набор параметров)
- Введем некоторую **функцию потерь *L(f(x), y)***,
  - в случае классификации используют $L(f(\mathbf{x}), y) = I[y \neq f(\mathbf{x})]$, где $f(\mathbf{x})$ - предсказанный класс.
- Задача обучения состоит в том, чтобы найти набор параметров классификатора f, при котором потери для новых данных будут минимальны
- «Метод классификации» = параметрическое семейство F+ алгоритм оценки параметров по обучающей выборке

Microsoft Research

Общий риск

- Общий риск – математическое ожидание потерь:

$$R(f) = \mathrm{E}\Big(L\big(f(\mathbf{x}), y\big)\Big) = \int_{\mathbf{x},y} L(f(\mathbf{x}), y)dP$$

  - рассчитать невозможно, поскольку распределение ***P*** неизвестно

Microsoft Research

Эмпирический риск

- Пусть $X^m = \{\mathbf{x_1}, ..., \mathbf{x_m}\}$- обучающая выборка

- Эмпирический риск (ошибка тренировки):

$$R_{emp}\left(f, X^m\right) = \frac{1}{m}\sum_{i=1}^{m} L(f(x_i), y_i)$$

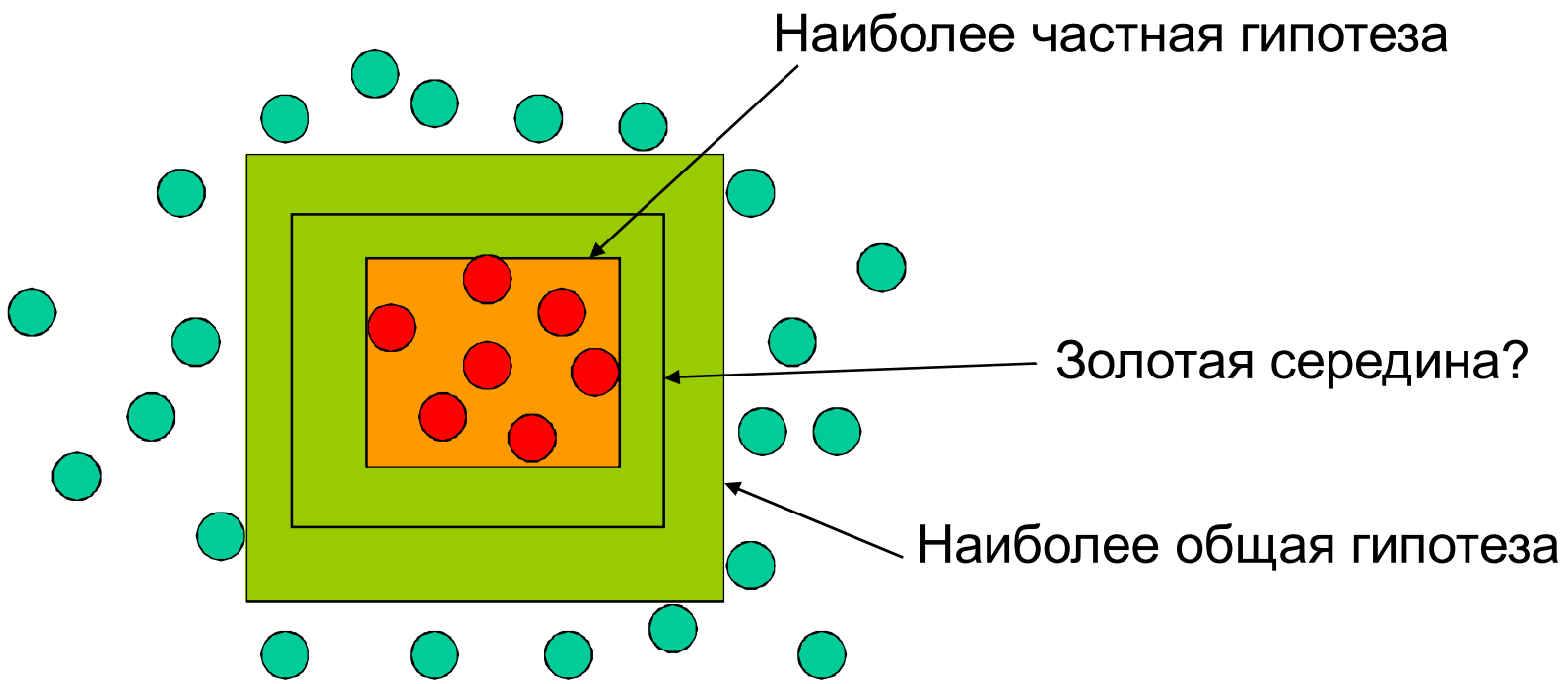
- Метод минимизации эмпирического риска:

$$f = \arg\min_{f \in F} R_{emp}\left(f, X^m\right)$$

Замечание

Microsoft Research

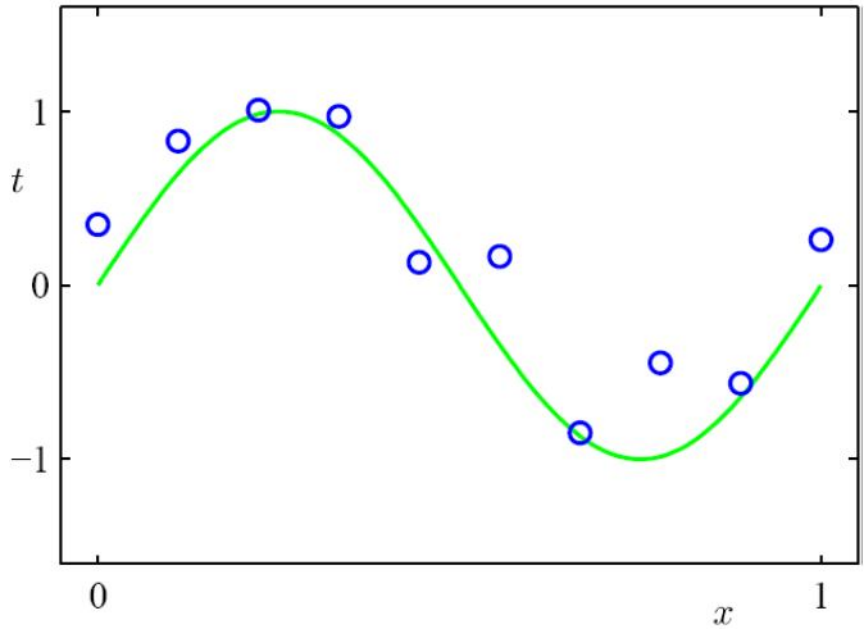
- Гипотез, имеющих нулевой эмпирический риск может также существовать неограниченное количество:



Microsoft Research

Явление переобучения

- Искусственный пример: задача регрессии
  - На самом деле $t = \sin(2\pi x) + \epsilon$, ϵ - нормально распределенный шум
  - Но мы этого не знаем
  - Есть обучающая выборка, требуется восстановить зависимость



Microsoft Research

Явление переобучения

- Будем выбирать целевую зависимость среди параметризованного множества - полиномов порядка M

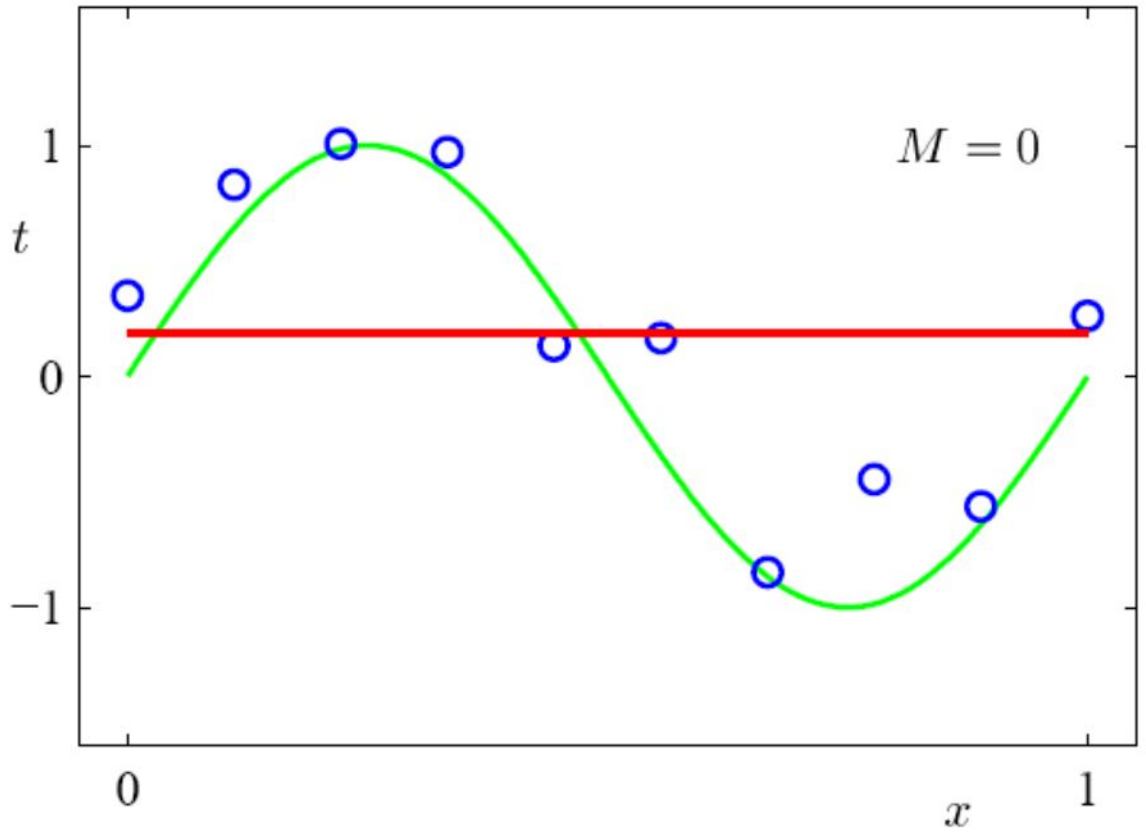
$$y(x, \mathbf{x}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \mathbf{w}^T \phi_M(x)$$

- Введем функция потерь $L((x,t),y) = \frac{1}{2}(y(x,\mathbf{w}) - t)^2$

- Среди множества полиномов будем выбирать тот, который приносит наименьшие суммарные потери на обучающей выборке

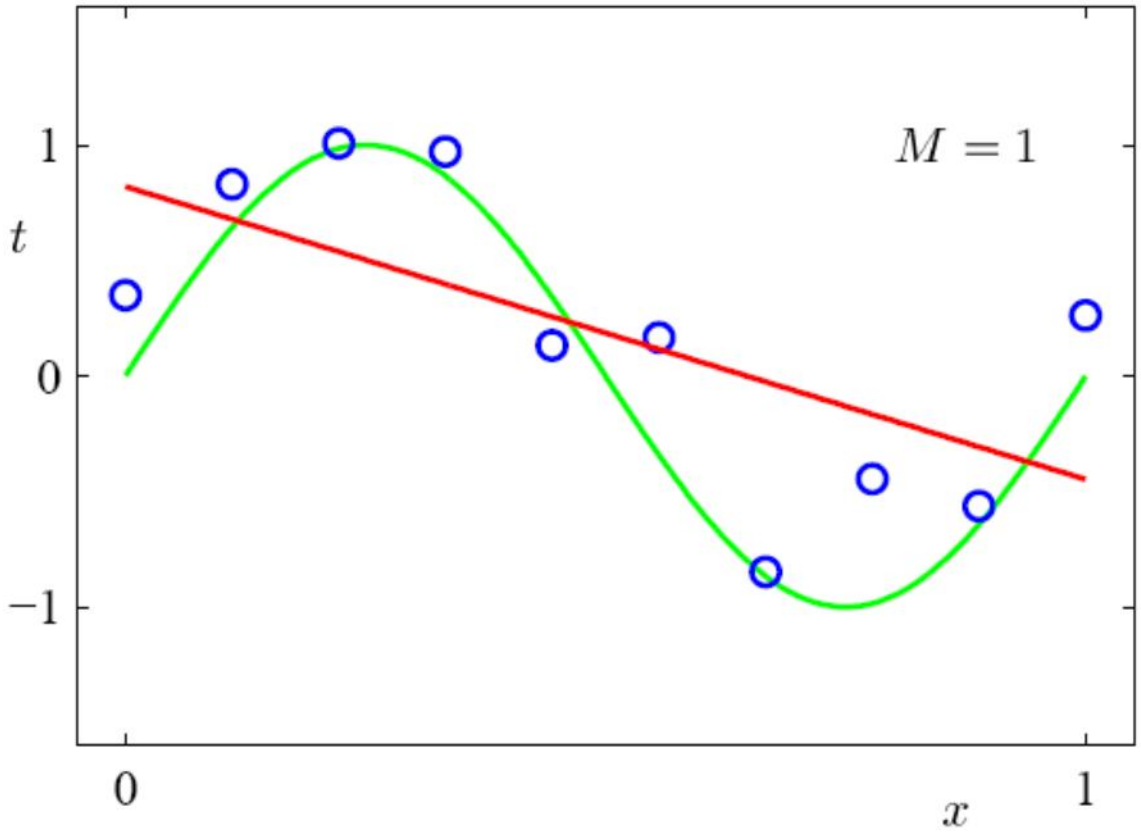
Явление переобучения

Microsoft Research



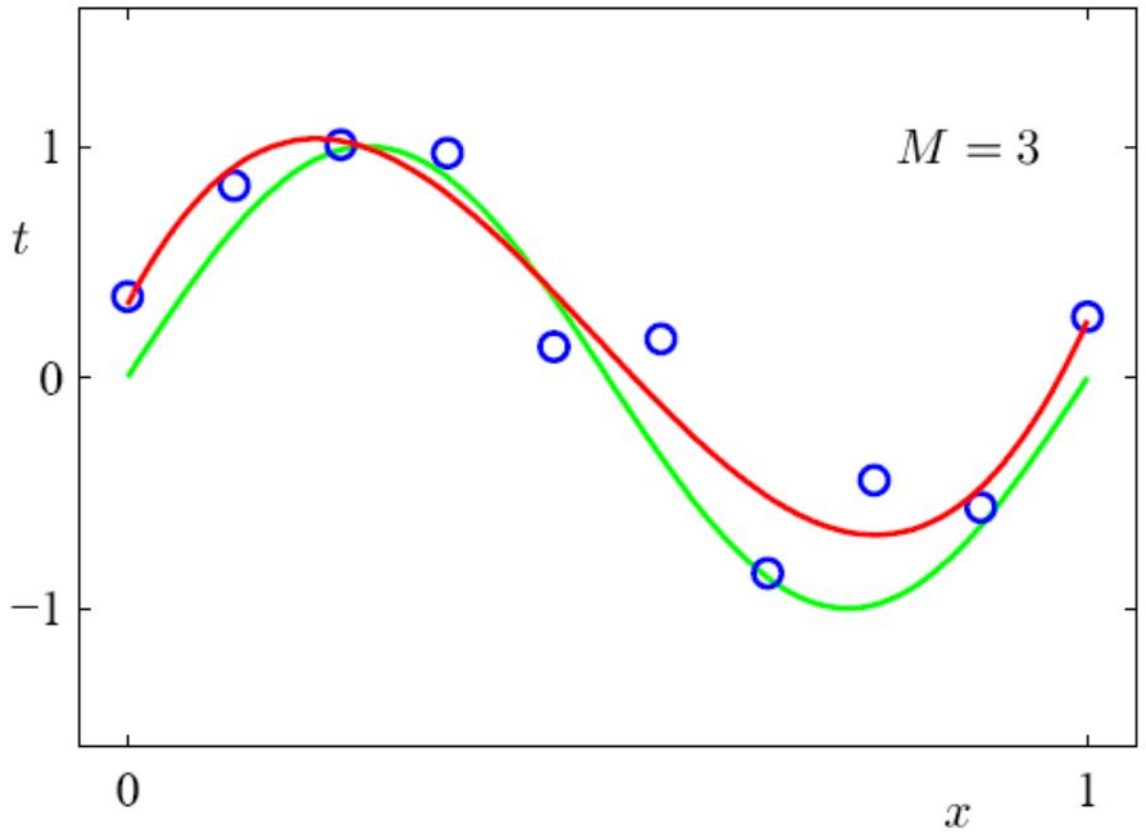
Явление переобучения

Microsoft Research



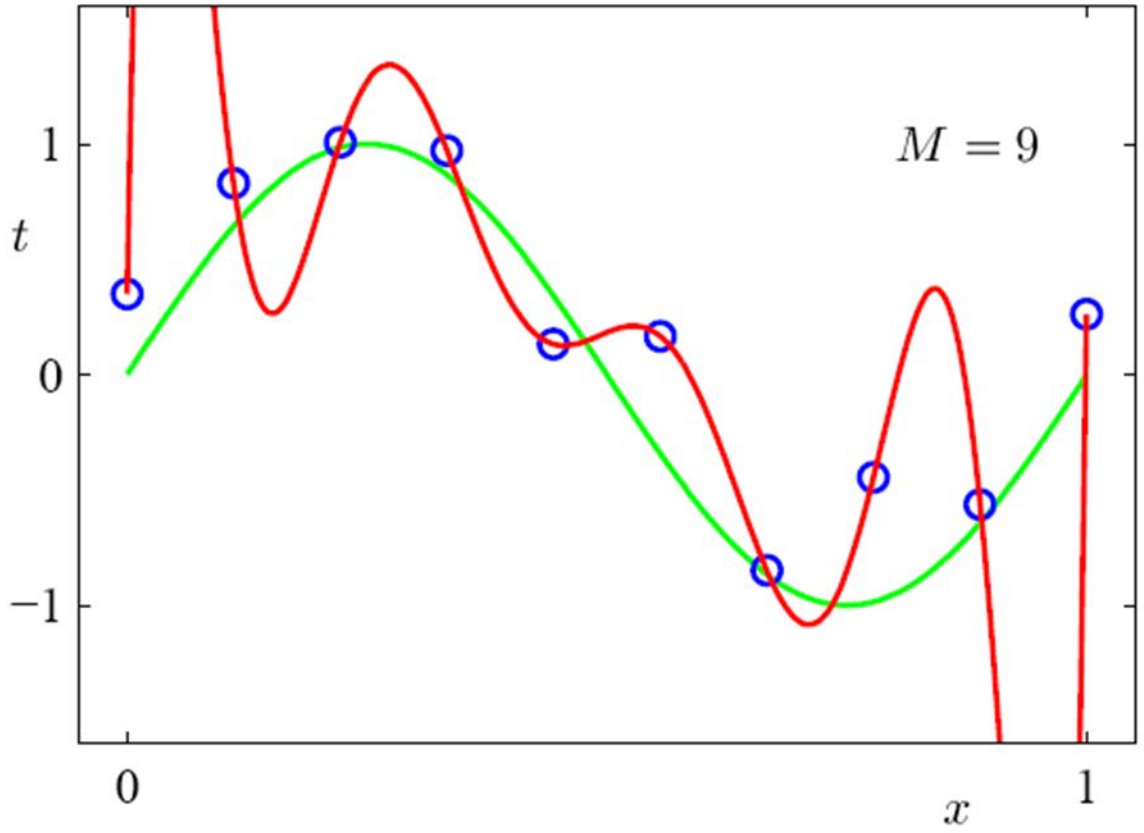
Явление переобучения

Microsoft Research



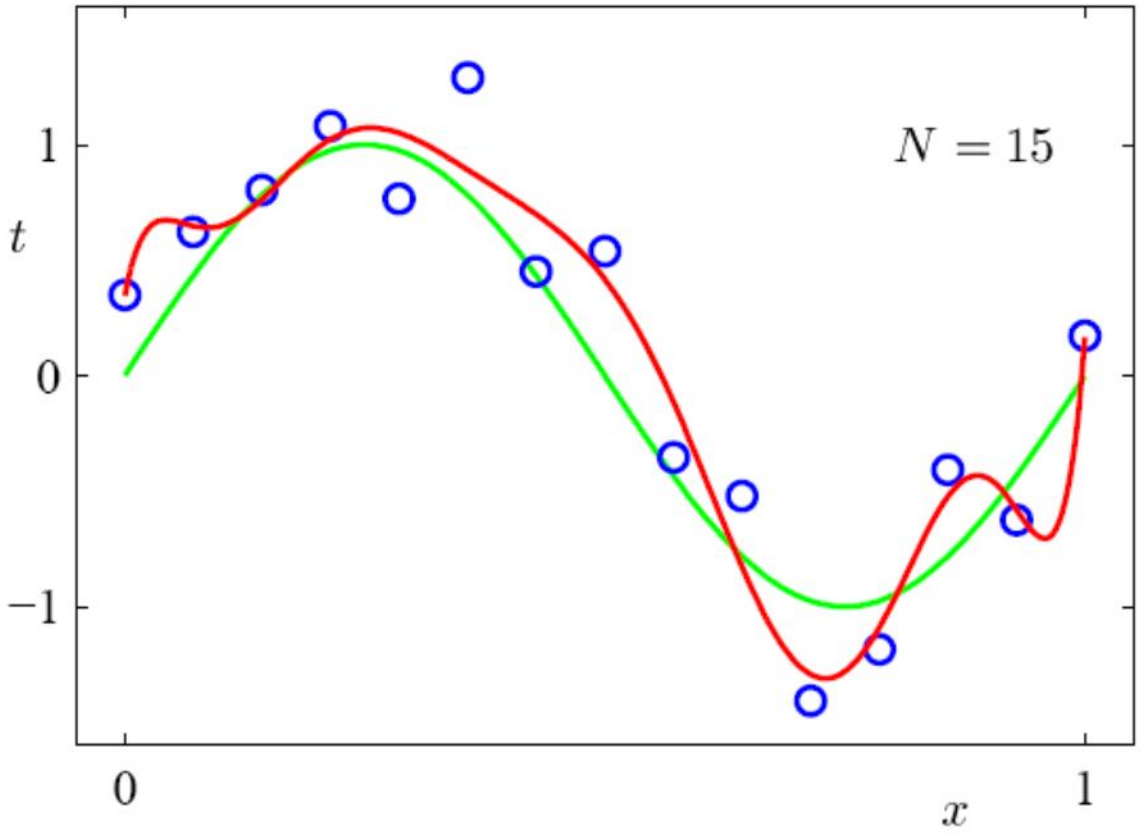
Явление переобучения

Microsoft Research



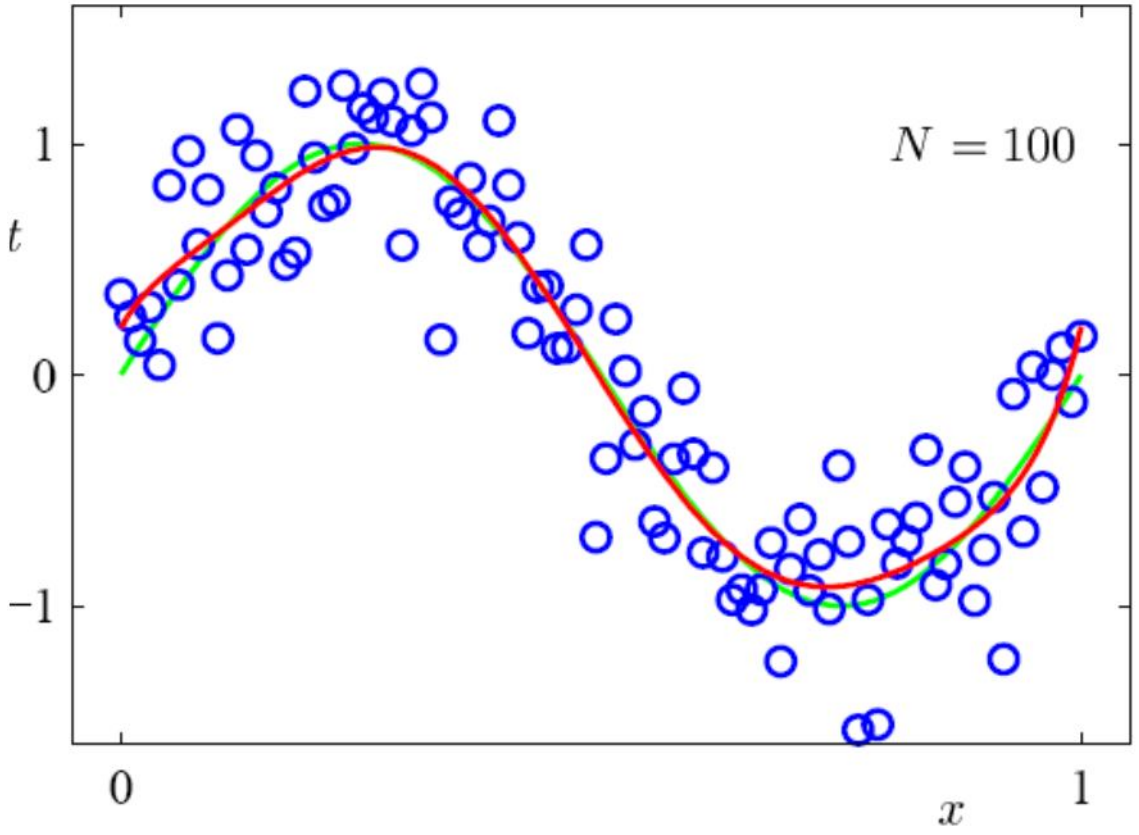
Явление переобучения

Microsoft Research



Явление переобучения

Microsoft Research



Microsoft Research

Явление переобучения

- Причина – гипотеза хорошо описывает свойства не объектов в целом, но только лишь объектов из обучающей выборки:
  - Слишком много степеней свободы параметров модели алгоритма (слишком сложная модель)
  - Шум в данных
  - Плохая обучающая выборка

Microsoft Research

Теория Вапника-Червоненкиса

- Теория, предложенная русским математиком Владимиром Вапником, для оценки обобщающей способности алгоритмов

- Основные результаты теории:
  - Оценка сложности (емкости) параметрического семейства функций
  - Оценка качества алгоритма через эмпирический риск и сложность модели

Microsoft Research

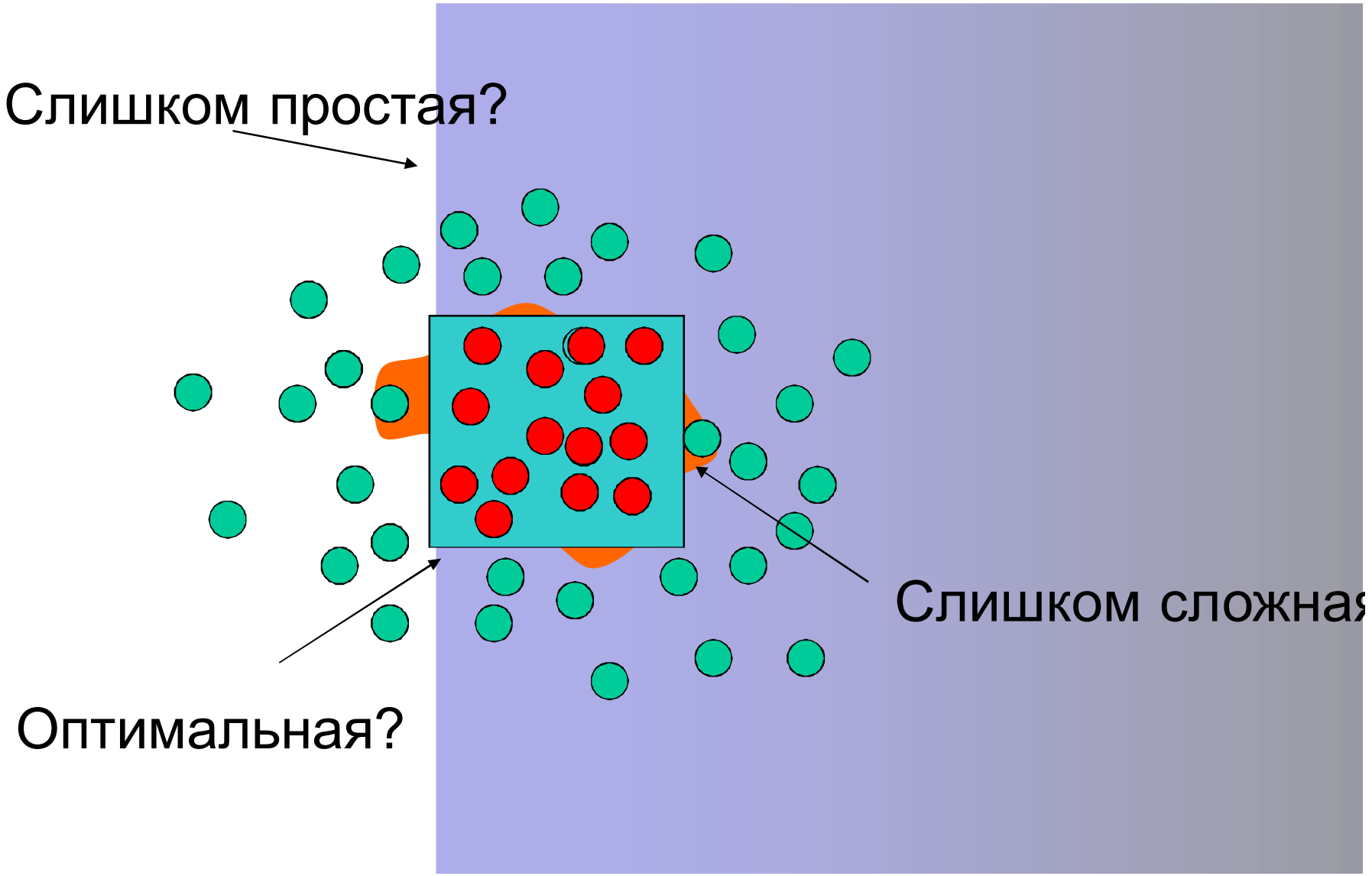
Принцип структурной минимизации риска

- Основная идея - «Выбрать модель наиболее простую из достаточно точных»
- Пусть есть последовательность вложенных параметрических семейств возрастающей сложности

$$F_1 \subset F_2 \subset \ldots \subset F_h = F$$

- Выберем семейство с минимальной сложностью, но обеспечивающее нужную точность

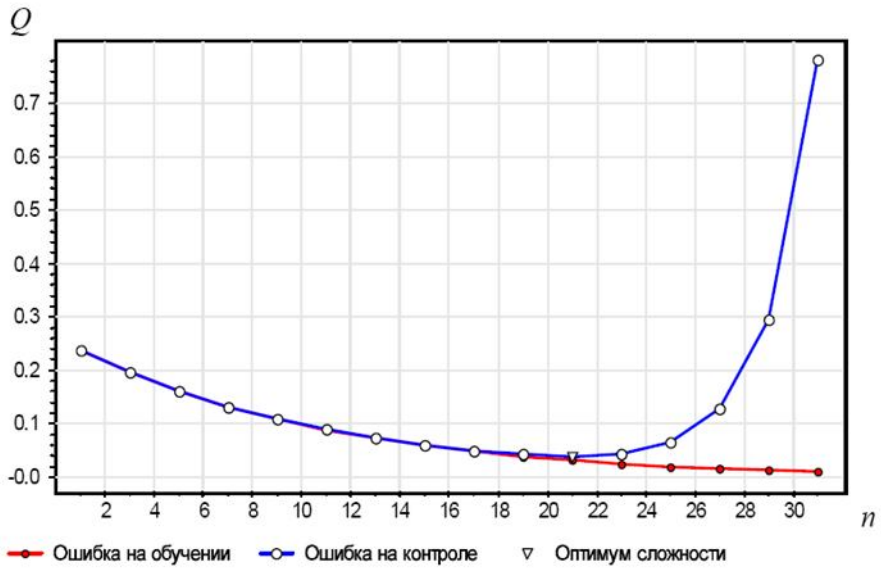
Иллюстрация



Microsoft Research

Практический вывод из VC теории

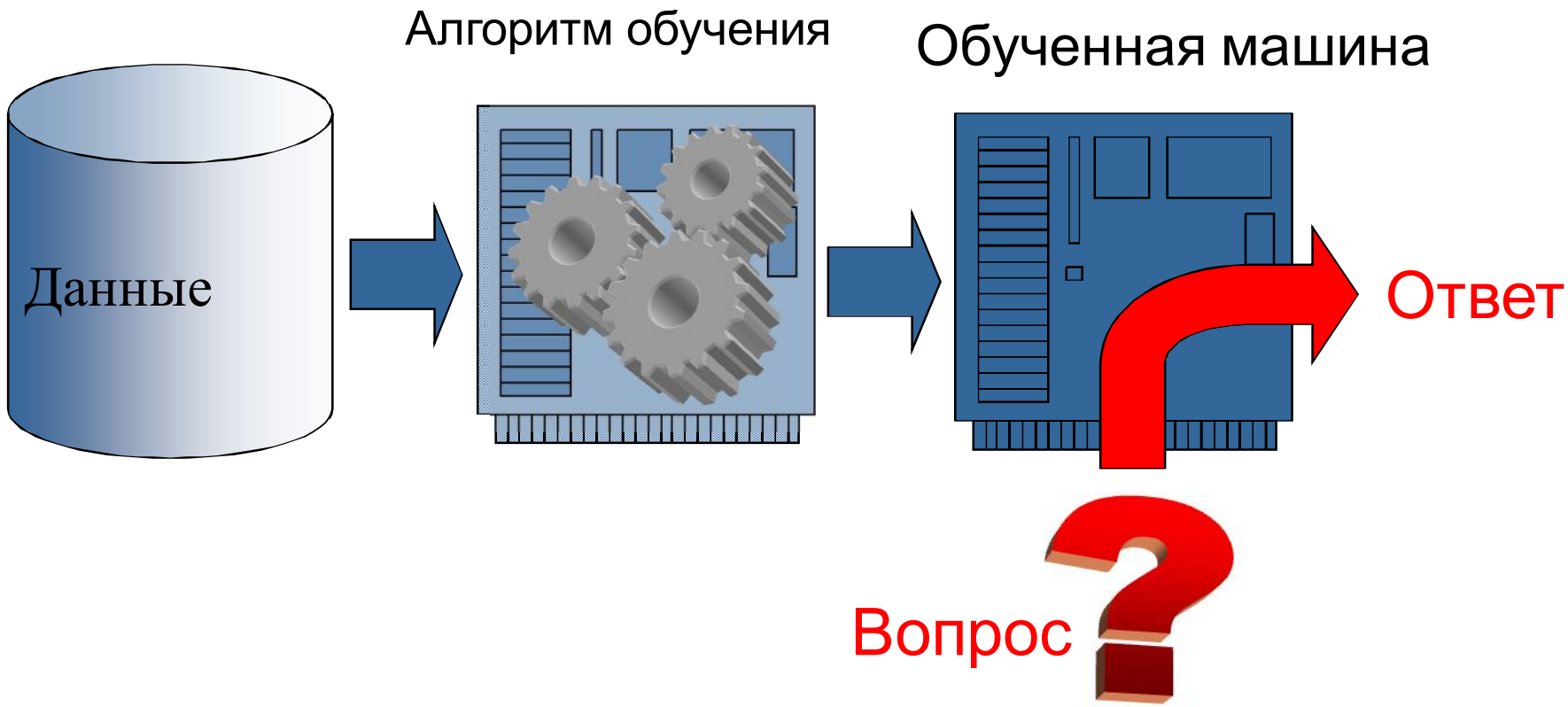
- Требуется баланс между сложностью модели, обеспечивающей низкий эмпирический риск и простотой, обеспечивающей способность к обобщению



Microsoft Research

Оценка классификаторов

- Предположим, что обучать классификатор мы уже умеем
- Как же оценить, насколько хорошо он обучился?
- Нужны количественные оценки качества обучения



Microsoft Research

Оценка общего риска

- Общий риск:

$$R(f, X) = P_{X_m}\left(f(x) \neq y\right) = \int_X P(x)\left[f(x) \neq y\right]dx$$

  - Его минимизация для нас является основной целью
  - Однако, напрямую его посчитать невозможно (требует вычислений на неограниченном множестве)

Удерживание

Microsoft Research

- Оценим общий риск ошибкой на некотором конечном подмножестве X не пересекающимся с обучающей выборкой:

$$R(f, X) \sim P\left(f(x) \neq y \mid X^c\right) = \frac{1}{c}\sum_{j=1}^{c}\left[f(x_j) \neq y_j\right]$$

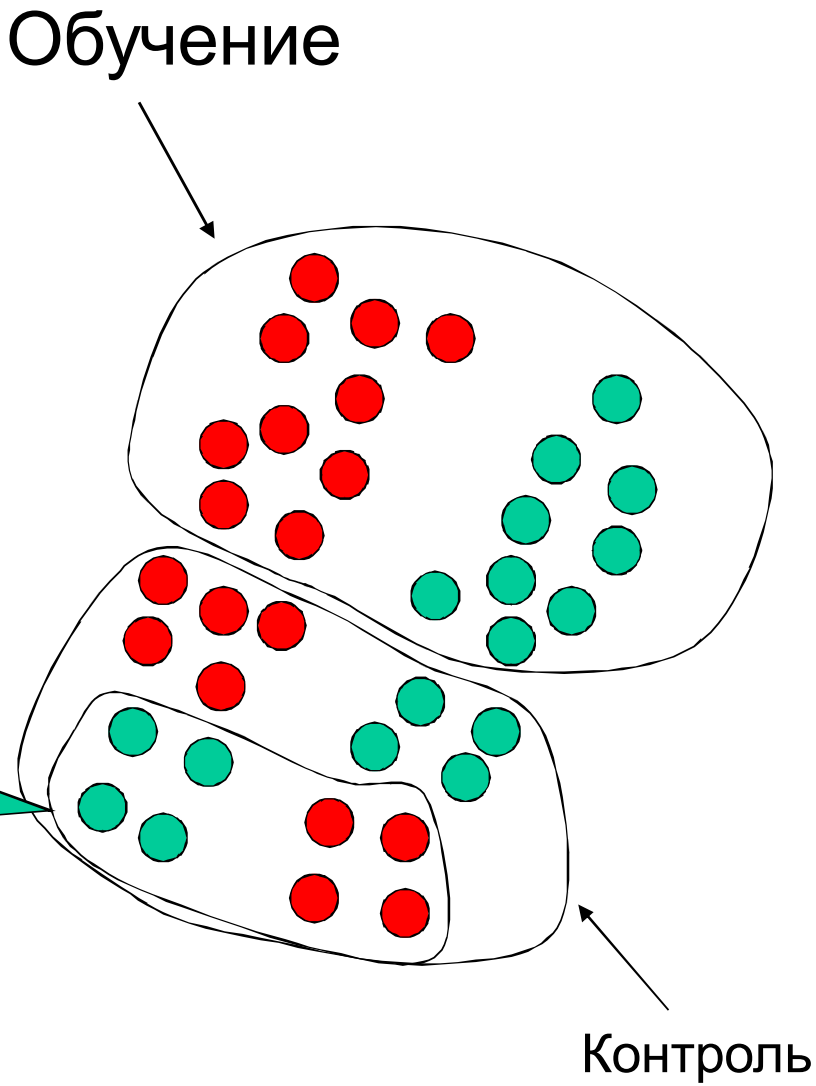
Microsoft Research

Удерживание

- Пусть, имеется набор данных $X^k = \{x_1, ..., x_k\}$ с известными ответами
- Разобьем $X^l \bigcup X^c = X^k : X^l \bigcap X^c = 0$
- Будем использовать для обучения X^l , а для контроля X^c
- То есть: $P(f(x) \neq y) \approx P\left(f(x) \neq y \mid X^c\right)$

Характеристики «удерживания»

- Быстро и просто рассчитывается
- Некоторые «сложные» прецеденты могут полностью попасть в только одну из выборок и тогда оценка ошибки будет смещенной



Ошибка произойдет не по вине классификатора, а из-за разбиения!

Microsoft Research

Повторное удерживание

- Если разбиение на контроль и обучение может быть не устойчивым, то почему бы не провести его много раз и не усреднить?
  - Такой методикой мы частично избавимся от проблемы «сложных прецедентов»;
  - НО, вероятность того, что какие-то прецеденты ни разу не попадут в контрольную выборку всё равно велика;
  - Процесс становиться сильно рандомизированным;

Microsoft Research

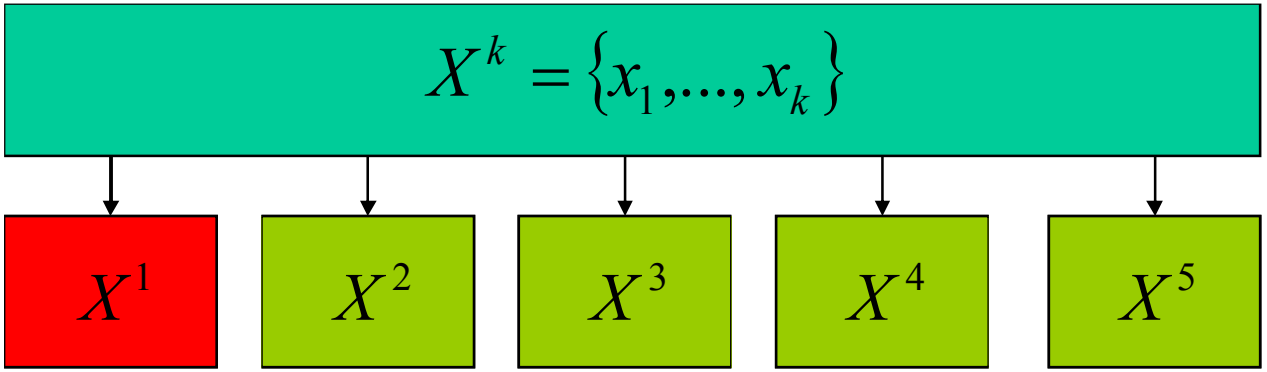
Скользящий контроль

- Разделим выборку на d непересекающихся частей и будем поочередно использовать одно из них для контроля а остальные для тренировки
  - Разбиваем: $\{X^i\}_1^d : X^i \cap X^j = 0, i \neq j$

    $\bigcup_{i=1}^{d} X^i = X^k$
  - Приближаем риск:

$$P(f(X^k) = y^*) \approx \frac{1}{d} \sum_{i=1}^{d} P(f(X^i) \neq y^* \mid \bigcup_{i \neq j} X^i)$$

Иллюстрация



Контроль

Обучение

Результат считается как средняя ошибка по всем итерациям

Microsoft Research

Свойства

- В пределе равен общему риску
- Каждый прецедент будет один раз присутствовать в контрольной выборке
- Обучающие выборки будут сильно перекрываться (чем больше сегментов, тем больше перекрытие)
  - Если одна группа «сложных прецедентов» попала полностью в один сегмент, то оценка будет смещенной

Microsoft Research

5-2 контроль

- 5-2 cross validation
- Некоторый компромисс:
  - Разделим выборку случайным образом пополам
  - Обучим на одной половине, протестируем на другой, и наоборот
  - Повторим этот эксперимент пять раз и усредним результат

- Свойства:
  - Каждый из прецедентов будет учувствовать в контрольных выборках на каждом из 5 этапов

Microsoft Research

Обучение параметров алгоритма

- У самого алгоритма обучения есть параметры, которые приходится как-то подбирать
  - Число деревьев в лесу
  - Глубина дерева

- Простая схема подбора параметров:
  - N раз повторяем схему обучения со скользящим контролем
  - Выбираем параметры, при которых обучение оказалось наилучшим

Microsoft Research

Виды ошибок

- Измерения ошибки как «вероятности выдать неверный ответ» может быть не всегда достаточно
  - 15% ошибки при постановке диагноза может означать как и то что, 15 % больных будут признаны здоровыми (и возможно умрут от отсутствия лечения), так и то, что 15% здоровых больными (и деньги на лечение будут потрачены зря)

- При неравнозначности ошибок для разных классов вводят понятие ошибки первого и второго рода и замеряют их по отдельности

Microsoft Research

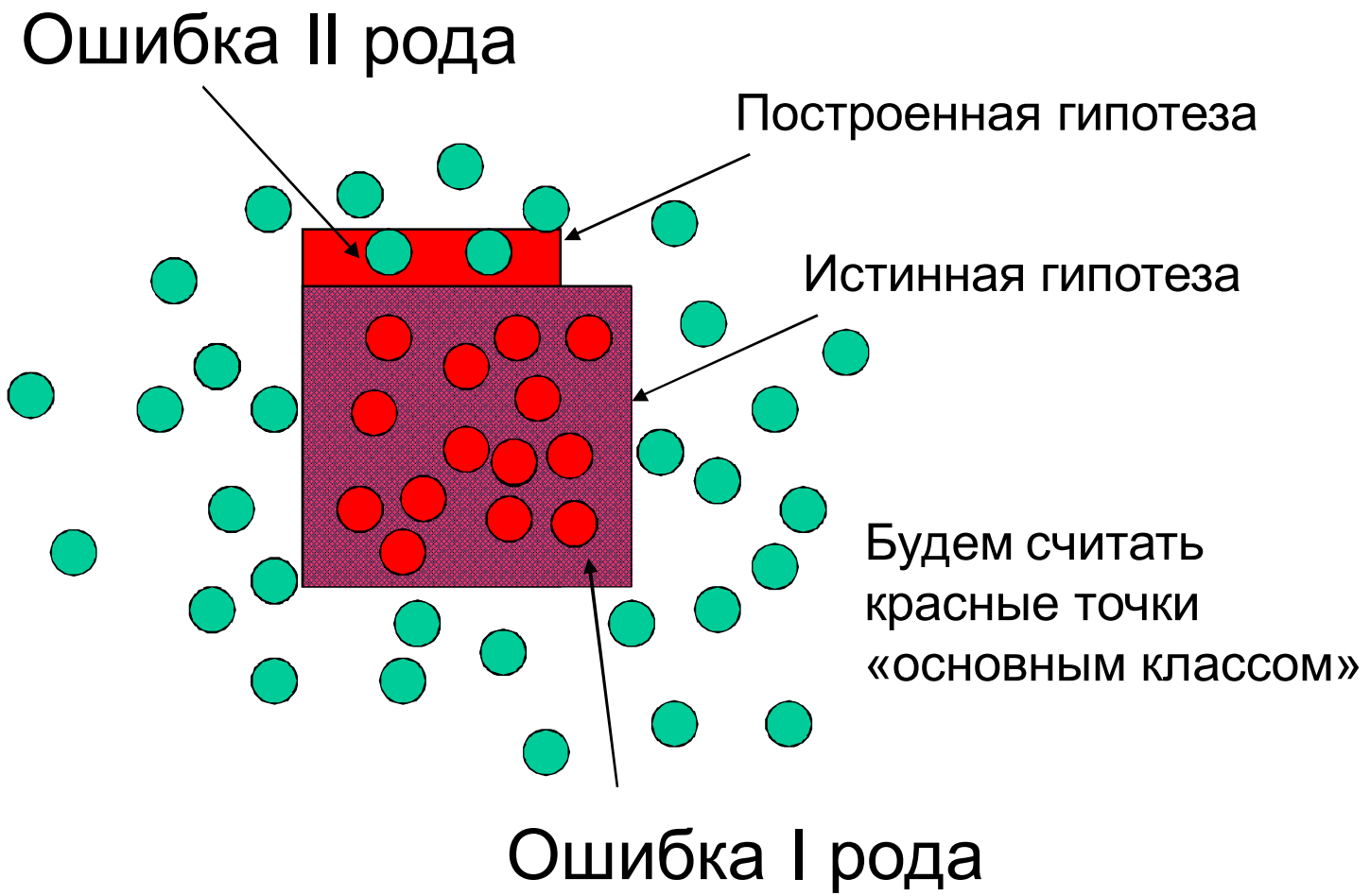
Ошибки I и II рода

- Пусть, существует «основной класс»
  - Обычно, это класс, при обнаружении которого, предпринимается какое-либо действие;
  - Например, при постановке диагноза основным классом будет «болен», а вторичным классом «здоров».

- Ошибка первого рода равна вероятности принять основной класс за вторичный
  - Вероятность «промаха», когда искомый объект будет пропущен

- Ошибка второго рода равна вероятности принять вторичный класс за основной
  - Вероятность «ложной тревоги», когда за искомый объект будет принят «фон»

Ошибки I и II рода



Microsoft Research

Ошибки I и II рода

- Что считать основным классом зависит полностью от прикладной специфики

- Особенно важно оценивать ошибки I и II рода раздельно при несбалансированности классов:
  - Пусть $P(y=+1)=0.01; P(y=-1)=0.99$
  - Тогда при ошибке II рода 0 и ошибке I рода 0.5

    $$P(f(x)=-1 \mid y=+1)=0.5$$

  - Общая ошибка всего лишь

    $$P(a(x) \neq y)=0.005$$

Microsoft Research

Чувствительность vs Избирательность

- **Чувствительность** – вероятность дать правильный ответ на пример основного класса

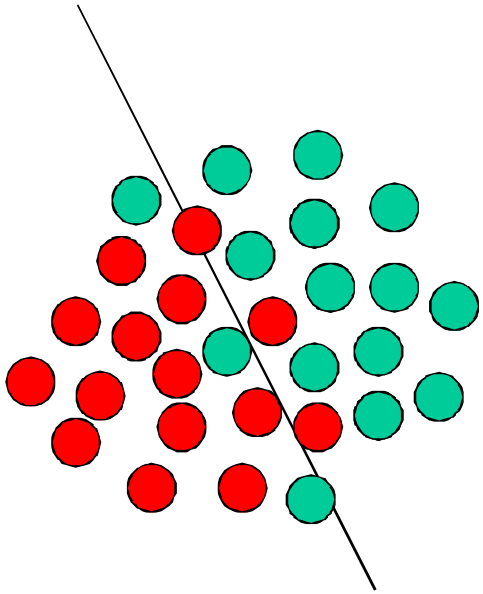
$$sensitivity = P(f(x) = y \mid y = +1)$$

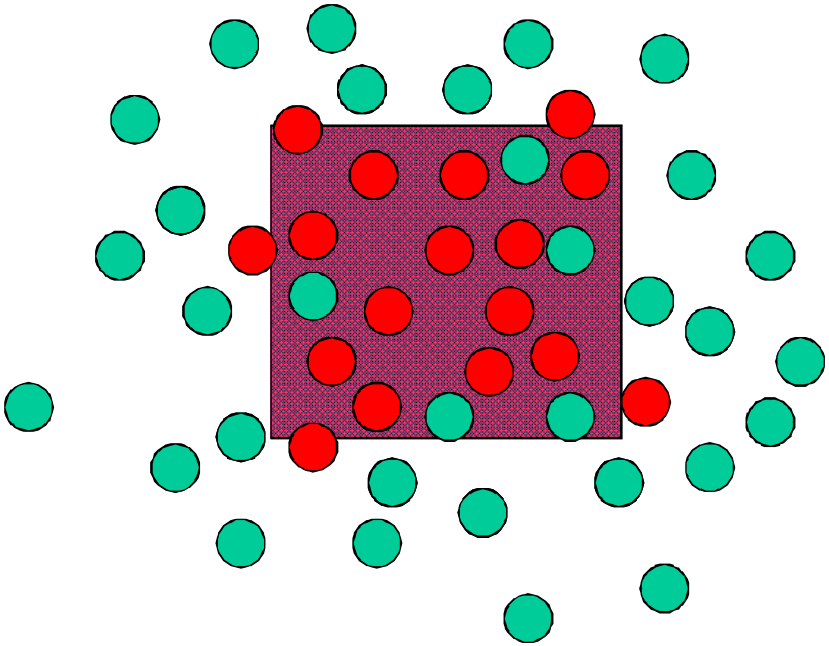
  - Также *уровень обнаружения (detection rate)*
- **Избирательность** – вероятность дать правильный ответ на пример вторичного класса

$$specificity = P(f(x) = y \mid y = -1)$$

Регулировка баланса

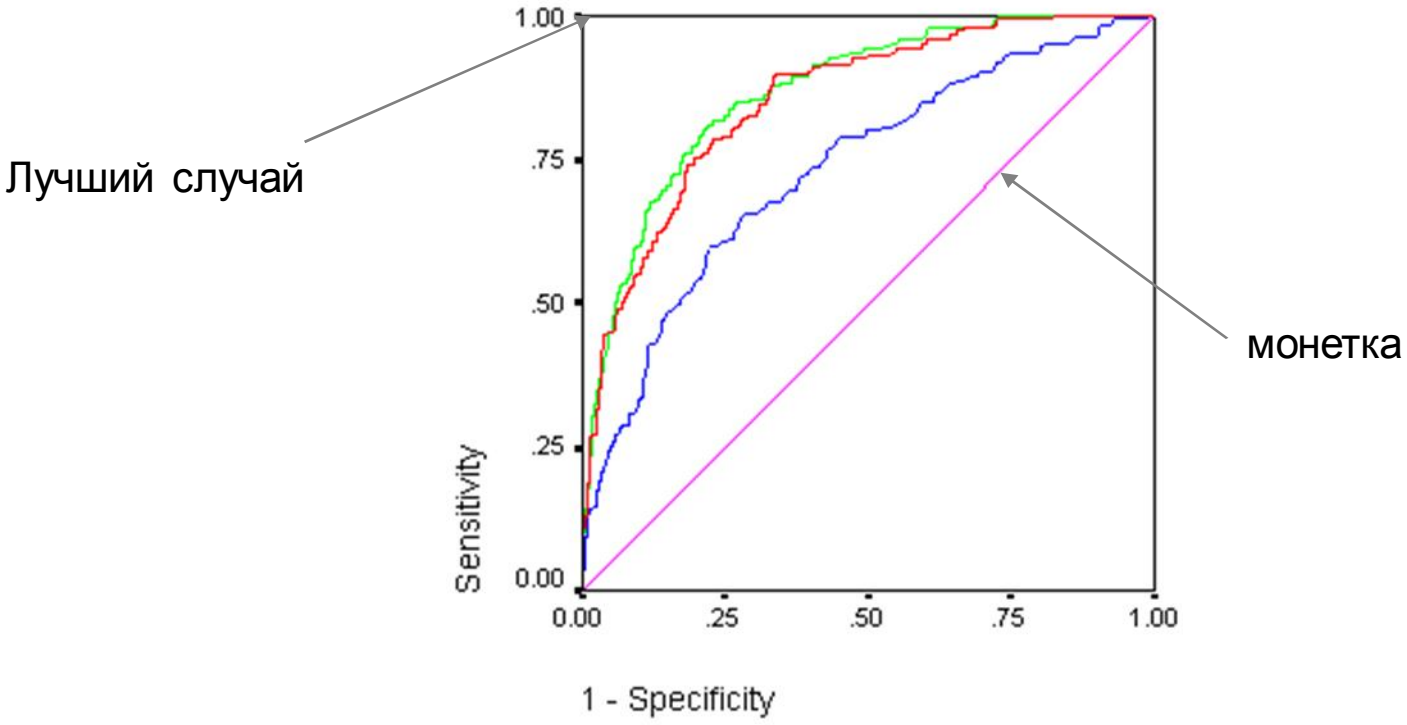
- Почти все алгоритмы классификации допускают регулировку соотношения ошибки I и II рода за счет варьирования некоторого параметра





ROC кривая

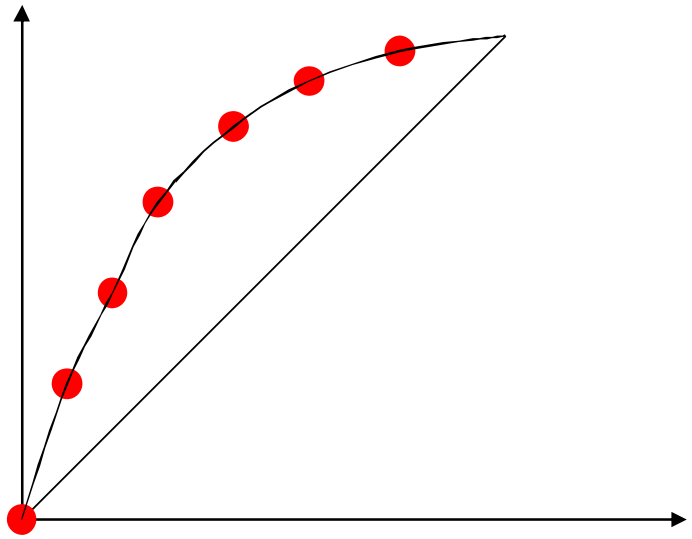
- ROC – Receiver Operating Characteristic curve
  - Кривая, отражающая зависимость чувствительности и ошибки второго рода



ROC кривая - Построение

- Для различных значений параметра строится таблица ошибок
  - Сам параметр в таблице не участвует!
  - Классификатор строится и оценивается на разных выборках!
- По таблице строиться набор точек в плоскости sensitivity/FP
  - Каждая строка таблицы - точка
- По точкам строиться кривая

Sensitivity	False Positive
0.0	0.0
0.25	0.5
0.5	0.8
…	…
1.0	1.0



Microsoft Research

Анализ ROC кривой

- Площадь под графиком – AUC
  - Дает некоторый объективный показатель качества классификатора
  - Позволяет сравнивать разные кривые

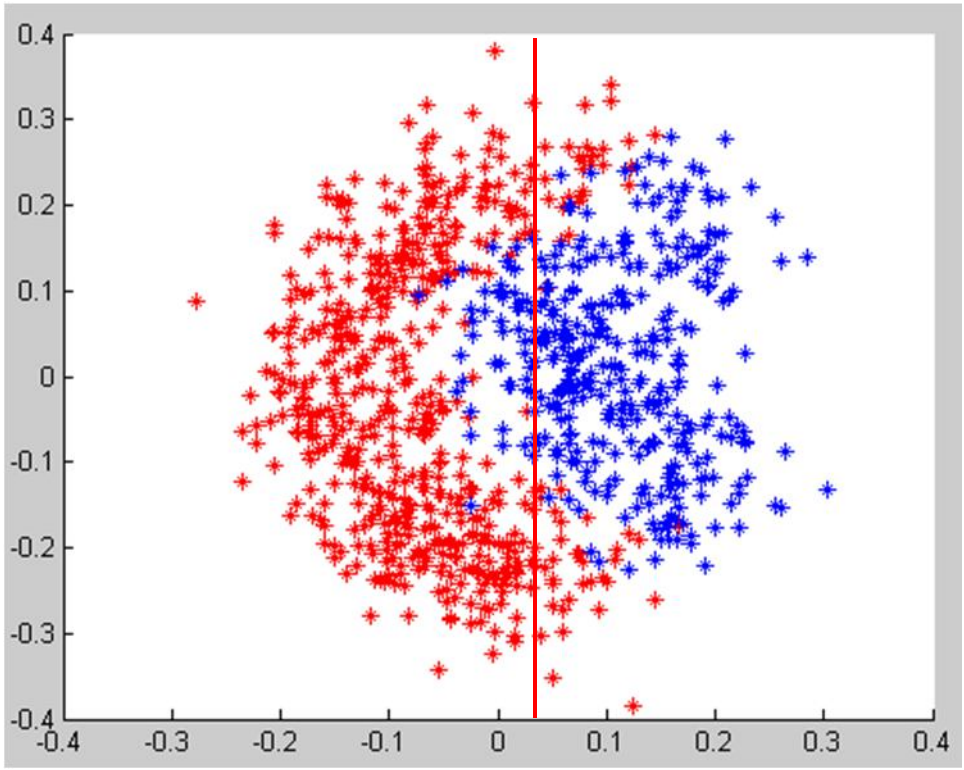
- Соблюдение требуемого значения ошибок I и II рода
  - Зачастую, для конкретной задачи существуют рамки на ошибку определенного рода. С помощью ROC можно анализировать возможность текущего решения соответствовать требованию

Пример

- Данные – точки на плоскости

- Параметрическое семейство – порог по оси X

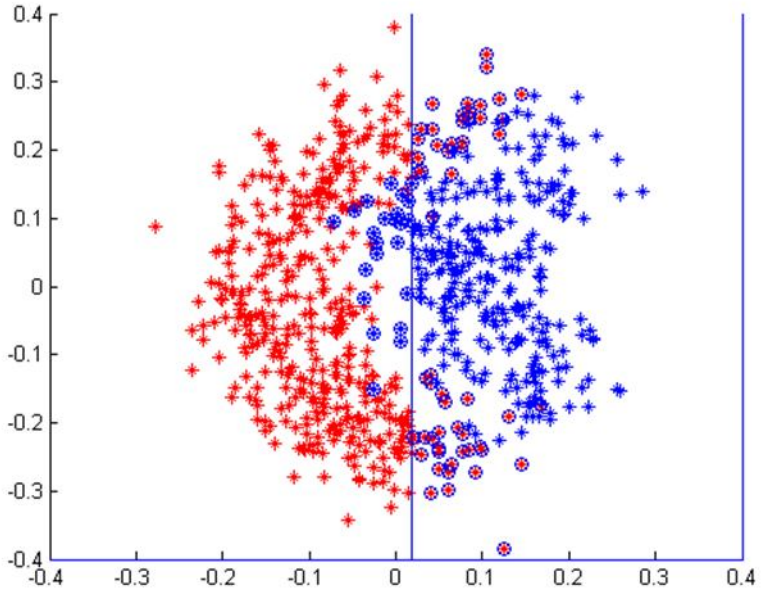
$$a(x^1, x^2) = \begin{cases} +1, x_1 > \Theta \\ -1, x_1 \le \Theta \end{cases}$$



Удерживание

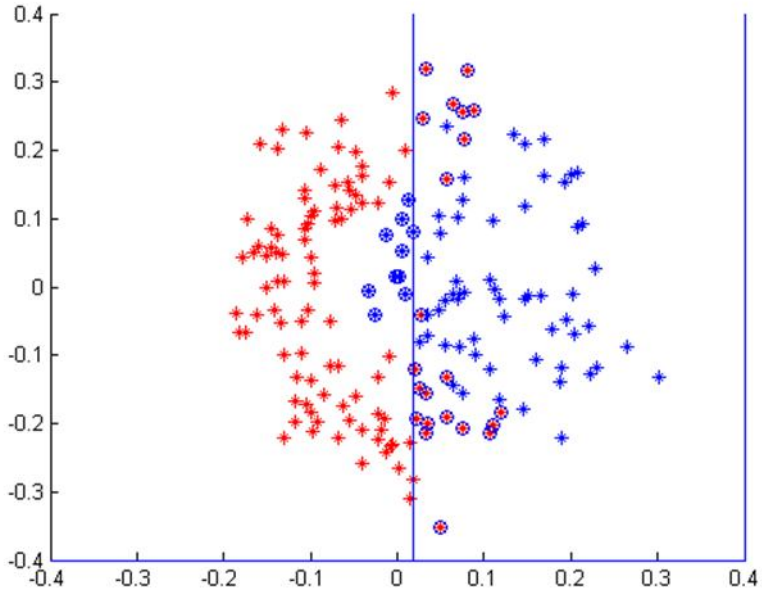
Microsoft Research

Ошибка: 0.1133



Тренировочная выборка

Ошибка: 0.1433



Контрольная выборка

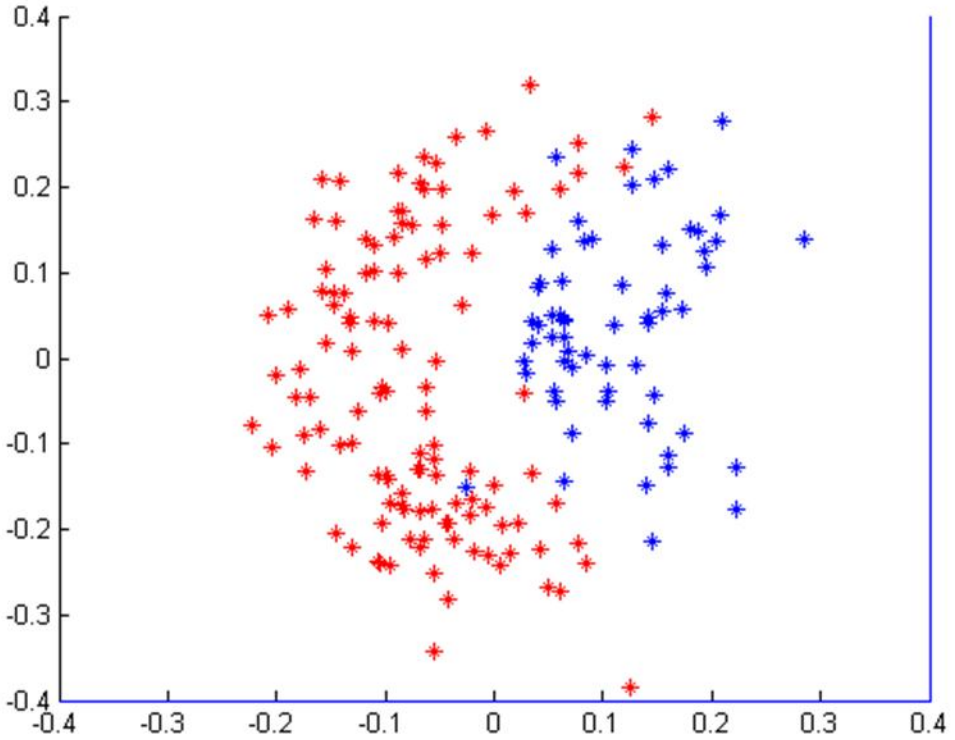
Повторное удерживание

- Тренировочная ошибка:
  - *{0.1097 0.1236 0.1208 0.1250 0.1250}*
  - Среднее = *0.1208*

- Ошибка на контроле
  - *{0.1833 0.1222 0.1333 0.1222 0.1167}*
  - Среднее = *0.1356*

Скользящий контроль: разбиение

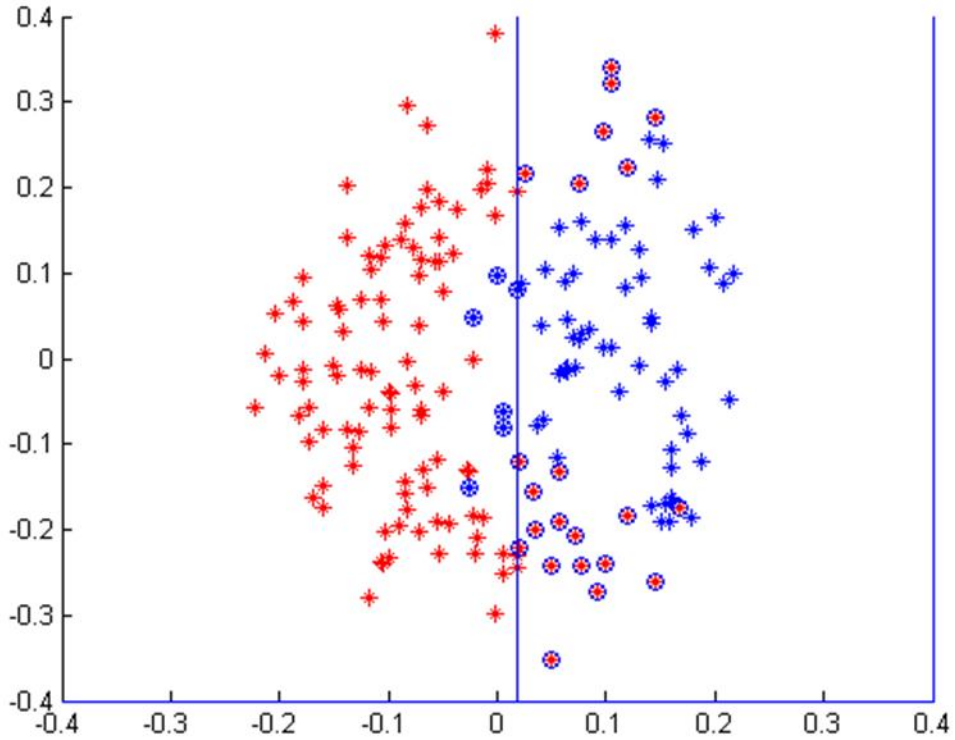
Microsoft Research



Microsoft Research

Скользящий контроль:

Итеративно измеряем ошибку



Microsoft Research

Скользящий контроль

- Тренировочная ошибка:
  - *{0.1236 0.1208 0.1250 0.1097 0.1306}*
  - Среднее = *0.1219*

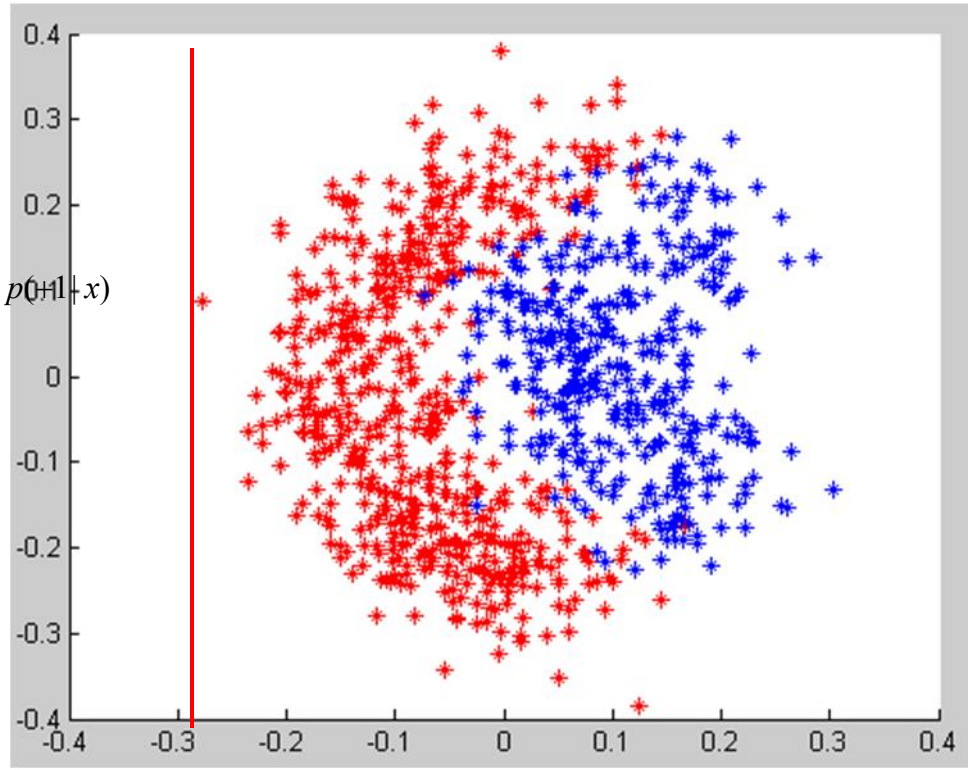
- Ошибка на контроле
  - *{0.1500 0.1333 0.1222 0.1778 0.1000}*
  - Среднее = *0.1367*

Построение ROC: таблица

Microsoft Research

- Меняем порог и оцениваем ошибку

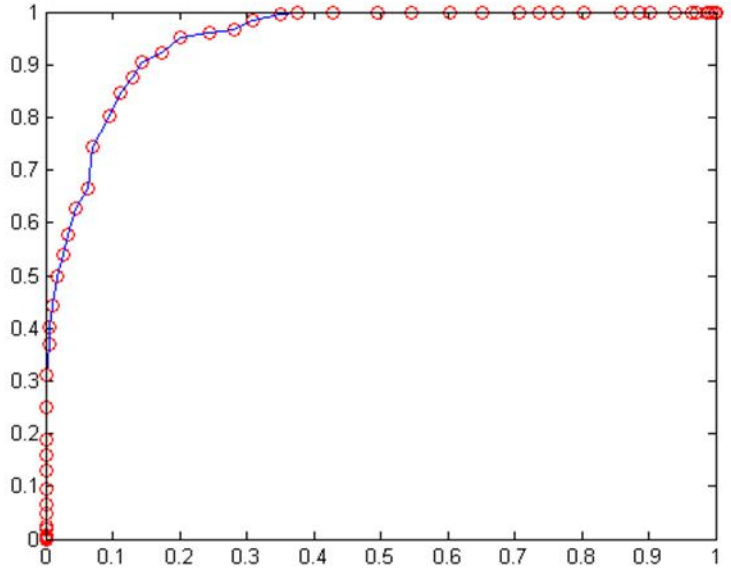
Sensitivity	False Positive
0.0	0.0
0.25	0.5
0.5	0.8
…	…
1.0	1.0



Microsoft Research

Построение ROC-кривой

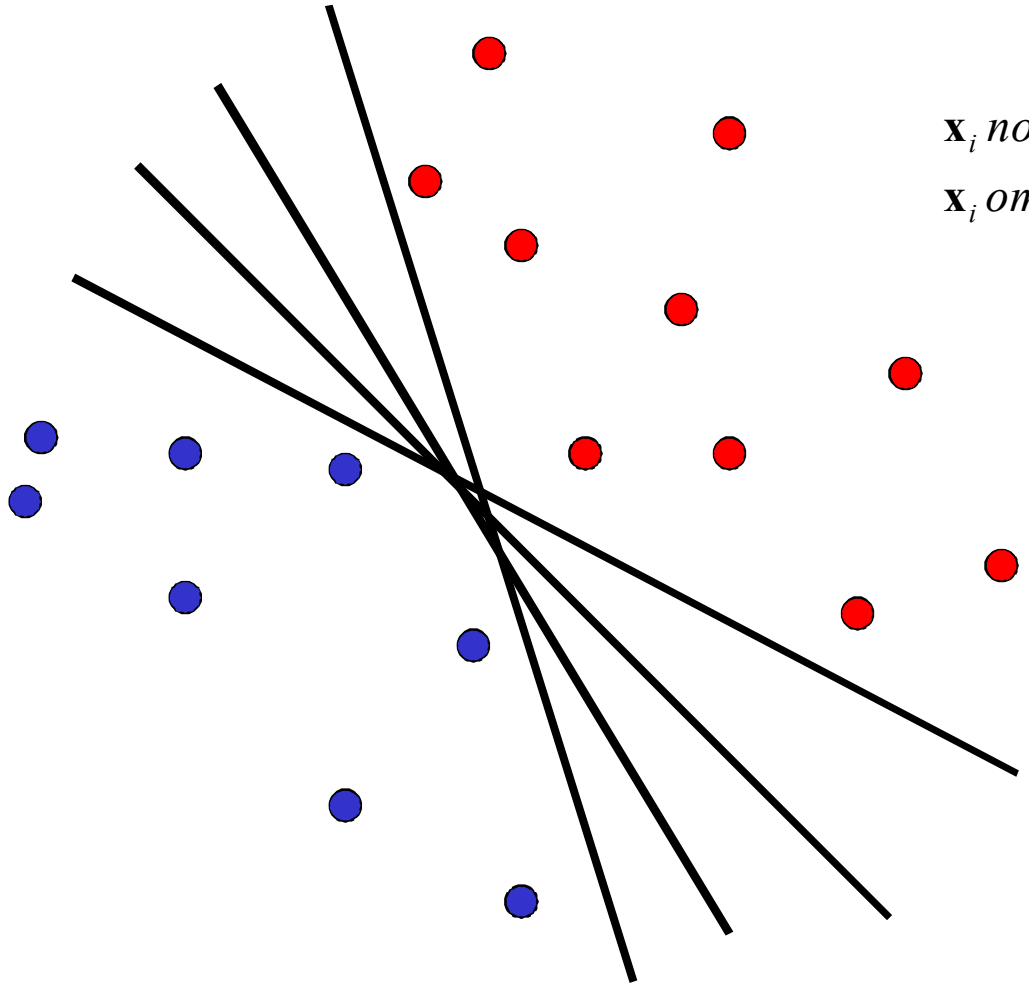
- По таблице строим точки
- Точки интерполируем кривой



Microsoft Research

Линейный классификатор

- Найдем линейную функцию (гиперплоскость) и разделим положительные {y=+1} и отрицательные {y=-1} примеры



$$\mathbf{x}_i \text{ положительные}: \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 0$$

$$\mathbf{x}_i \text{ отрицательные}: \quad \mathbf{x}_i \cdot \mathbf{w} + b < 0$$

Какая гиперплоскость наилучшая?

Метод опорный векторов (MOB)

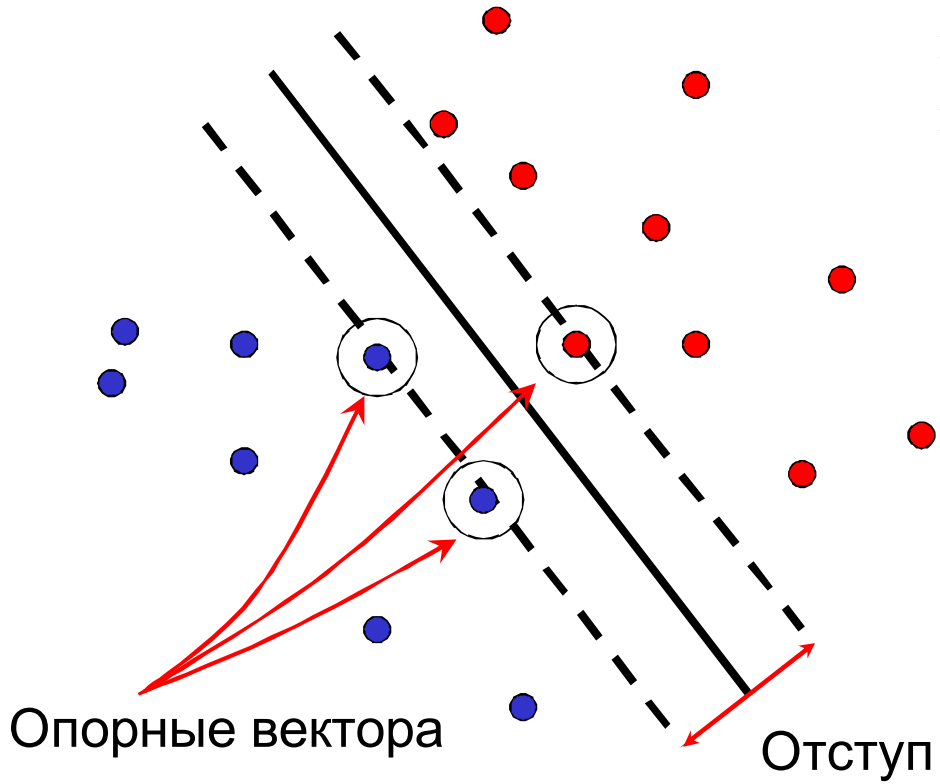
Microsoft Research

- Support Vector Machine (SVM)
- Найдем гиперплоскость, максимизирующую *отступ* между положительными и отрицательными примерами

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Метод опорный векторов

- Найдем гиперплоскость, максимизирующую *отступ* между положительными и отрицательными примерами



$\mathbf{x}_i$ *положительные* $(y_i = 1)$: $\mathbf{x}_i \cdot \mathbf{w} + b \geq 1$

$\mathbf{x}_i$ *отрицательные* $(y_i = -1)$: $\mathbf{x}_i \cdot \mathbf{w} + b \leq -1$

Для опорных векторов, $\mathbf{x}_i \cdot \mathbf{w} + b = \pm 1$

Расстояние от точки до гиперплоскости: $\dfrac{|\mathbf{x}_i \cdot \mathbf{w} + b|}{\|\mathbf{w}\|}$

Поэтому отступ равен $2 / \|\mathbf{w}\|$

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Microsoft Research

Поиск гиперплоскости

1. Максимизируем $2/\|\mathbf{w}\|$
2. Правильно классифицируем все данные:

$$\mathbf{x}_i \textit{ позитивные } (y_i = 1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$$

$$\mathbf{x}_i \textit{ негативные } (y_i = -1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$$

- *Квадратичная оптимизационная задача*:

- Минимизируем $\frac{1}{2}\mathbf{w}^T\mathbf{w}$

  При условии $y_i(\mathbf{w}\cdot\boldsymbol{x}_i+b) \geq 1$

- Решается с помощью методом множителей Лагранжа

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Поиск гиперплоскости

Microsoft Research

- Решение: $\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$

Обученные веса

Опорные вектора

- Для большей части векторов вес = 0!
- Все вектора, для которых вес >0 называются опорными
- Определяется только опорными векторами

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Microsoft Research

Поиск гиперплоскости

- Решение: $\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$

  $b = y_i - \mathbf{w} \cdot \mathbf{x}_i$ для любого опорного вектора

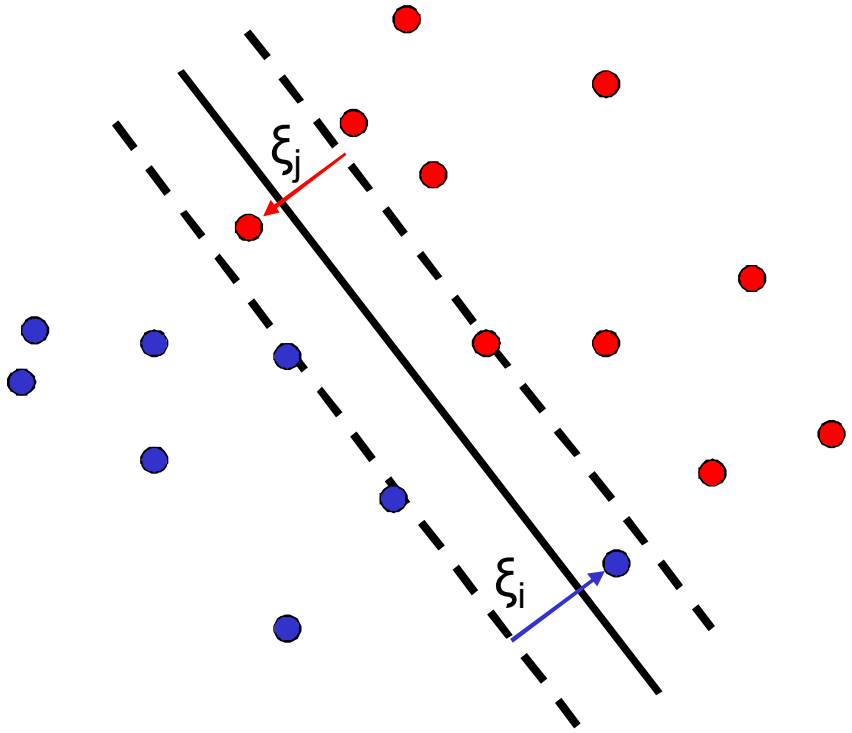
- Решающая функция:

$$\mathbf{w} \cdot \mathbf{x} + b = \sum_i \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b$$

- Решающая функция зависит от скалярных произведений (inner product) от тестового вектора $\boldsymbol{x}$ и опорных векторов $\boldsymbol{x}_i$
- Решение оптимизационной задачи также требует вычисления скалярных произведений $\boldsymbol{x}_i \cdot \boldsymbol{x}_j$ между всеми парами векторов из обучающей выборки

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Реальный случай



Вводим дополнительные «slack» переменные: $\xi = (\xi_1, \ldots, \xi_n)^T$

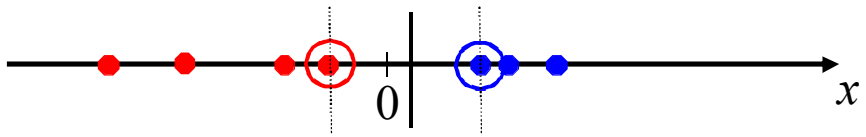
Минимизируем $\frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}\xi_i$

При условии $y_i(wx_i + b) \geq 1 - \xi_i$

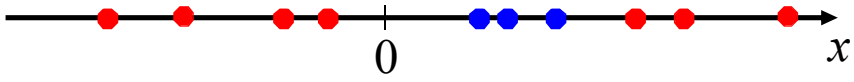
C – параметр регуляризации

Нелинейные МОВ

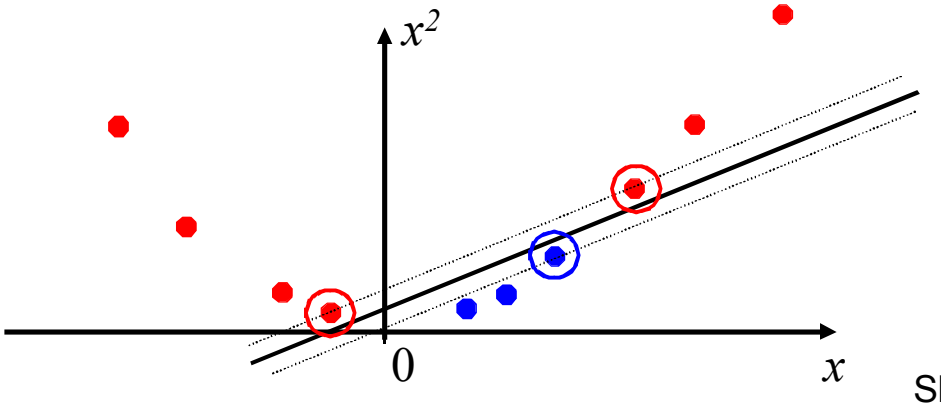
- На линейно разделимых данных МОВ работает отлично:



- Но на более сложных данных не очень:



- Можно отобразить данные на пространство большей размерности и разделить их линейно там:

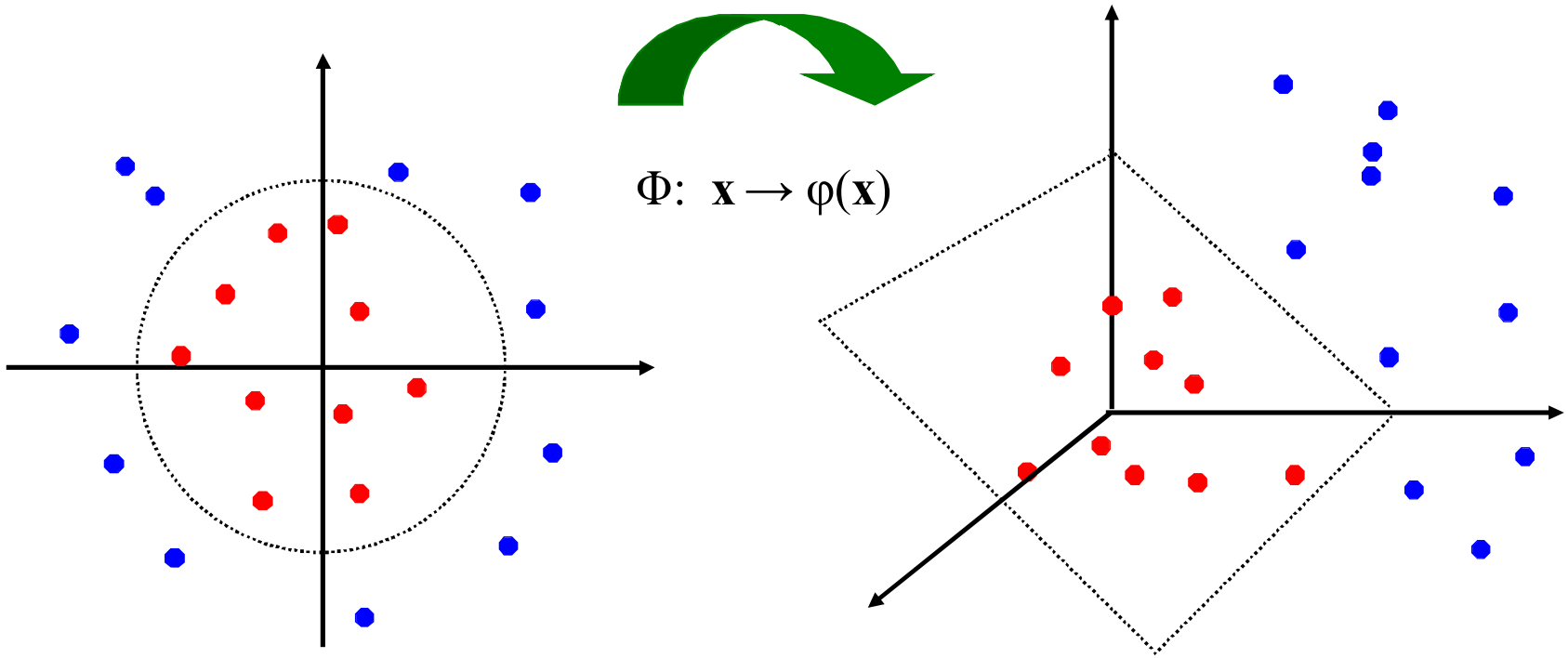


Slide credit: Andrew Moore

Нелинейные МОВ

Microsoft Research

- Идея: отображение исходного пространства параметров на какое-то многомерное пространство признаков (feature space) где обучающая выборка линейно разделима:



Slide credit: Andrew Moore

Нелинейные МОВ

Microsoft Research

- Вычисление скалярных произведений в многомерном пространстве вычислительно сложно
- *The kernel trick:* вместо прямого вычисления преобразования $\varphi(\mathbf{x})$, мы определим ядровую функцию K:

$$K(\mathbf{x}_i, \mathbf{x}_{jj}) = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)$$

- Чтобы все было корретно, ядро должно удовлетворять условию Мерсера (*Mercer's condition*)
  - Матрица $K(x_i,x_j)$ должна быть неотрицательно определенной
- С помощью ядра мы сможем построить нелинейную решающую функцию в исходном пространстве:

$$\sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b$$

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Microsoft Research

Пример ядра

- Полиномиальное: $K(x, y) = ((\mathbf{x}, y) + c)^d$

- Пусть d=2, x=(x_1,x_2):

$$K(x, y) = ((\mathbf{x}, y) + c)^2 = (x_1 y_1 + x_2 y_2 + c)^2 = \varphi(x) \cdot \varphi(y)$$

$$\varphi(x) = (x_1^2, x_2^2, \sqrt{2}x_1 x_2, \sqrt{2c}x_1, \sqrt{2c}x_2, c)$$

- Т.е. из 2х мерного в 6и мерное пространство

Microsoft Research

Использование MOB

1. Выбрать признаки для изображения («мешок слов»)
2. Выбрать ядро для этого вектор-признака
3. Вычислить матрицу значений ядра для каждой пары примеров из обучающей выборки
4. Загрузить матрицу в библиотеку SVM для получения весов и опорных векторов
5. Во время вывода: вычислить значения ядра для тестового образца и каждого опорного вектора и взять взвешенную сумму для получения значения решающей функции

Microsoft Research

Многоклассовые MOB

- Нет специальной формулировки MOB для случая многих классов
- На практике MOB для нескольких классов получается путем комбинации нескольких двухклассовых MOB
- Один против всех
  - Обучение: обучаем MOB для каждого класса против всех остальных
  - Вывод: применим все MOB к образцу и назначим класс, MOB для которого выдал наиболее достоверное решение
- Один против одного:
  - Обучение: обучим MOB для каждой пары классов
  - Вывод: каждый MOB голосует за классы, выбираем класс с наибольшим числом голосов

Microsoft Research

MOB - резюме

- Плюсы
  - Много доступных библиотек: http://www.kernel-machines.org/software
  - Мощный и гибкий подход на основе ядер
  - На практике работает очень хорошо, даже для маленьких обучающих выборок

- Минусы
  - Нет прямых многоклассовых метод, нужно объединять двухклассовые MOB
  - Вычисления, память
    - При обучении нужно строить полную матрицу ядра для всех примеров
    - Обучение занимает много времени для больших задач

Microsoft Research

- Почти всё готово для классификации изображений

Классификация текстов

- Представление документа без порядка: частоты слов из словаря («мешок слов») Salton & McGill (1983)

Классификация текстов

- Представление документа без порядка: частоты слов из словаря («мешок слов») Salton & McGill (1983)

Классификация текстов

- Представление документа без порядка: частоты слов из словаря («мешок слов») Salton & McGill (1983)

«Визуальные слова»

- Что в нашем случае «слово» и «словарь»?
- «Визуальное слово» – часто повторяющийся фрагмент изображения («visual word»)
- В изображении визуальное слово может встречаться только один раз, может ни разу, может много раз

Sivic, J. and Zisserman, A. Video Google: A Text Retrieval Approach to Object Matching in Videos. ICCV 2003

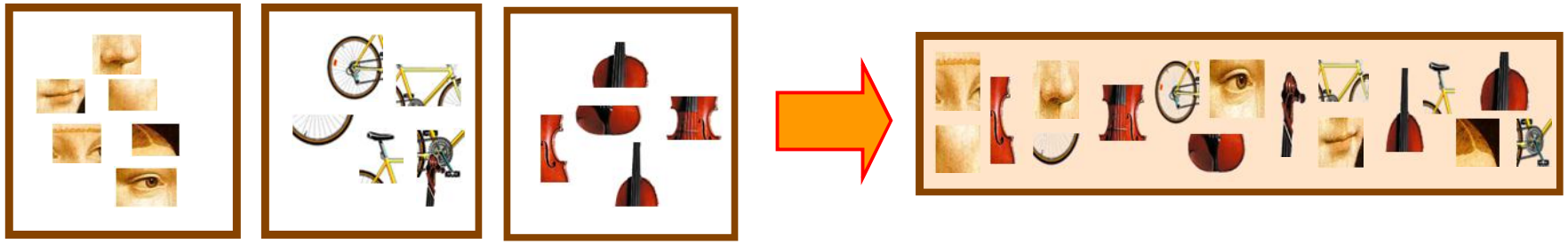
Microsoft Research

«Визуальный словарь»

- Словарь – набор фрагментов, часто повторяющихся в коллекции изображений

- Как составить словарь?
  - Составить большой список всех фрагментов по всей коллекции
  - Разделить весь список на похожие группы
  - Будем считать все фрагменты в одной группе – «экземплярами» одного и того же слова

«Мешок визуальных слов»



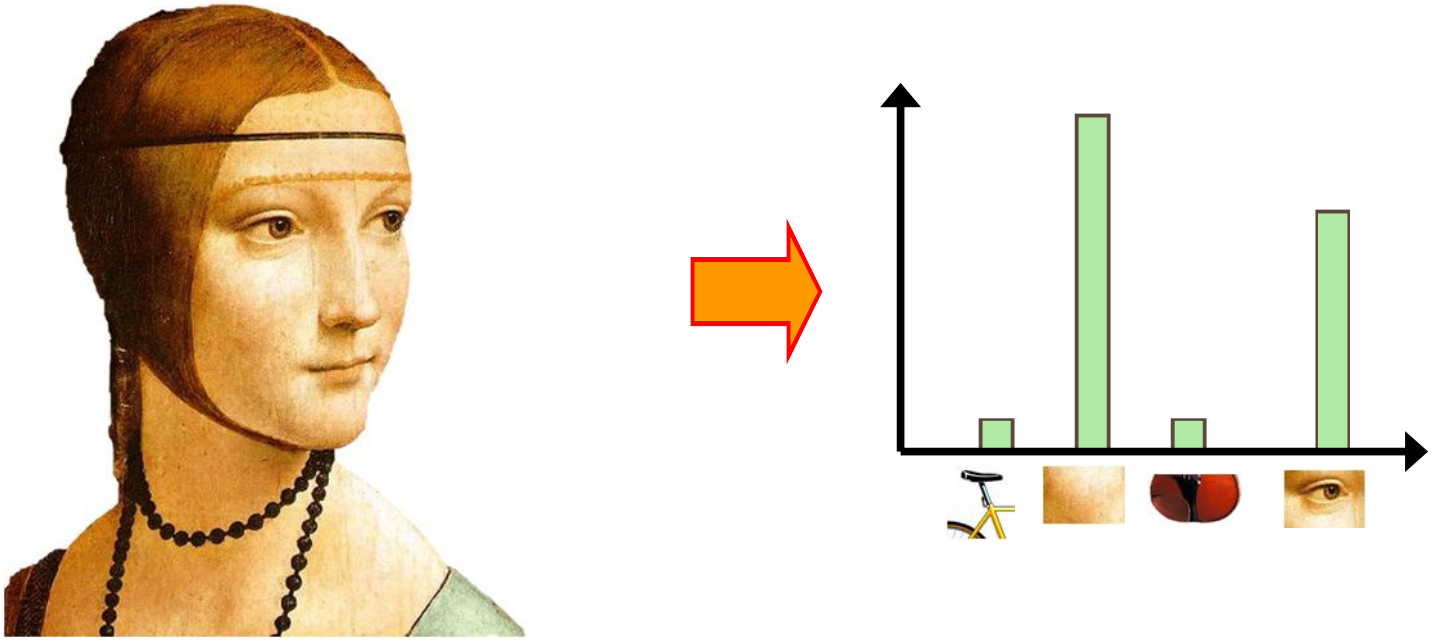
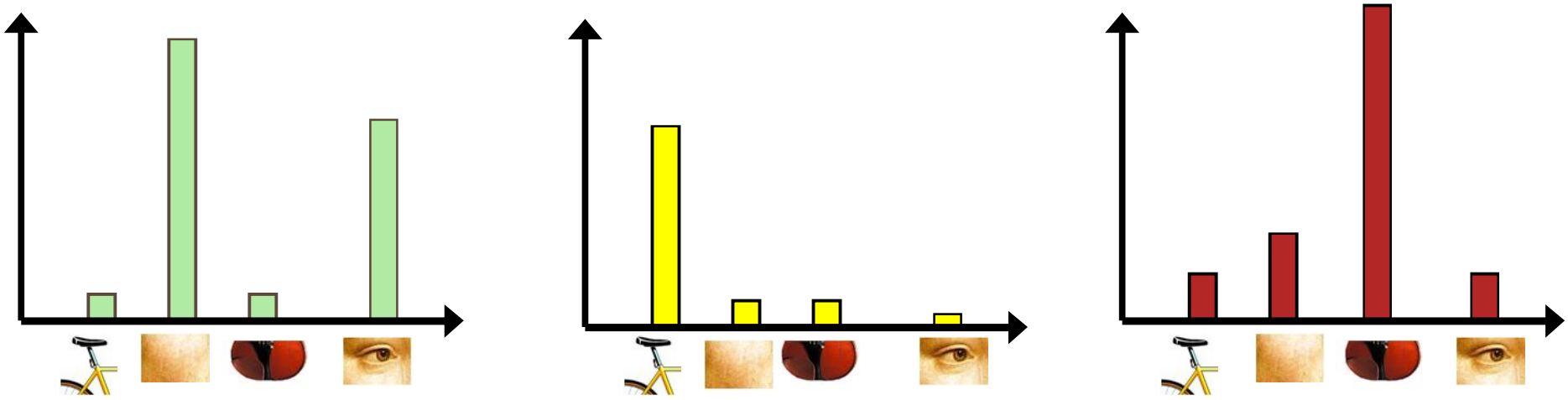


Схема метода «мешок слов»

1. Извлечение особенностей
2. Обучить “визуальный словарь”
3. Квантуем особенности по словарю
4. Описываем картинку частотами «визуальных слов»



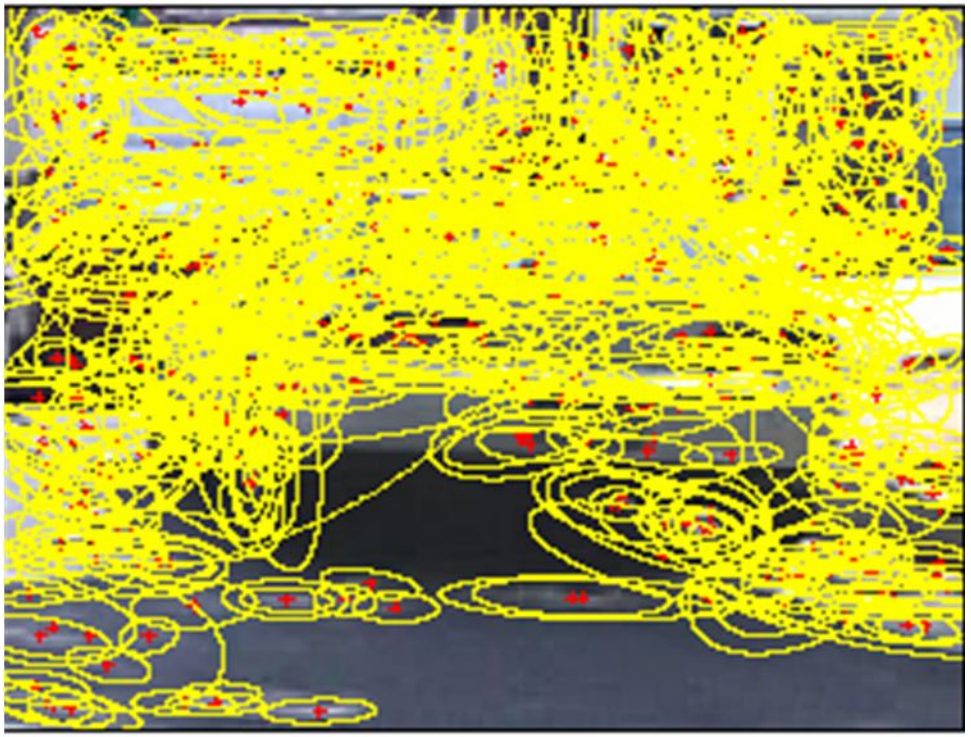
1. Извлечение особенностей

- Регулярная сетка
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005

1. Извлечение особенностей

Microsoft Research

- Регулярная сетка
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
- Характерные точки
  - Csurka et al. 2004
  - Fei-Fei & Perona, 2005
  - Sivic et al. 2005

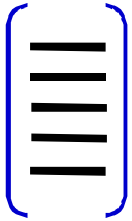


1. Извлечение особенностей

Microsoft Research

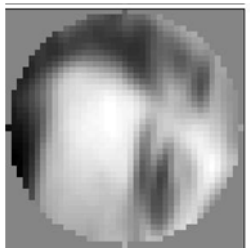
- Регулярная сетка
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
- Характерные точки
  - Csurka et al. 2004
  - Fei-Fei & Perona, 2005
  - Sivic et al. 2005
- Другие методы
  - Случайный выбор (Vidal-Naquet & Ullman, 2002)
  - Сегменты (Barnard et al. 2003)

1. Извлечение особенностей

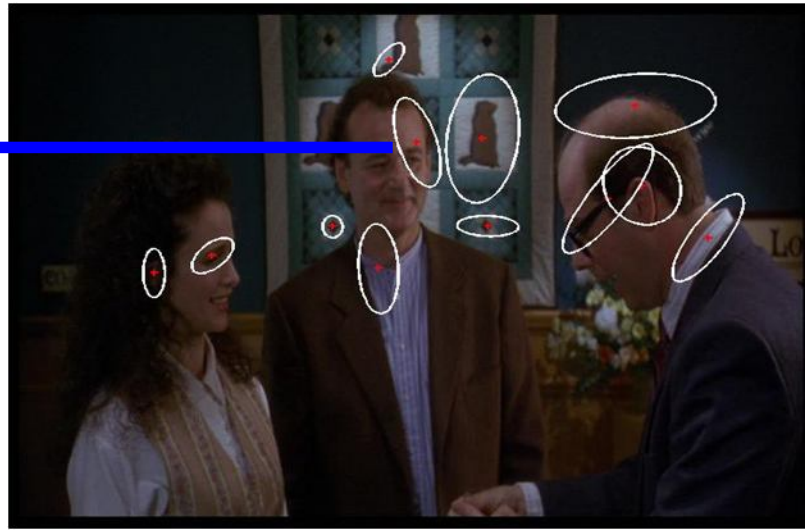


Вычисление SIFT

[Lowe'99]



Нормализация области



Найденные области

[Mikojaczyk and Schmid '02]

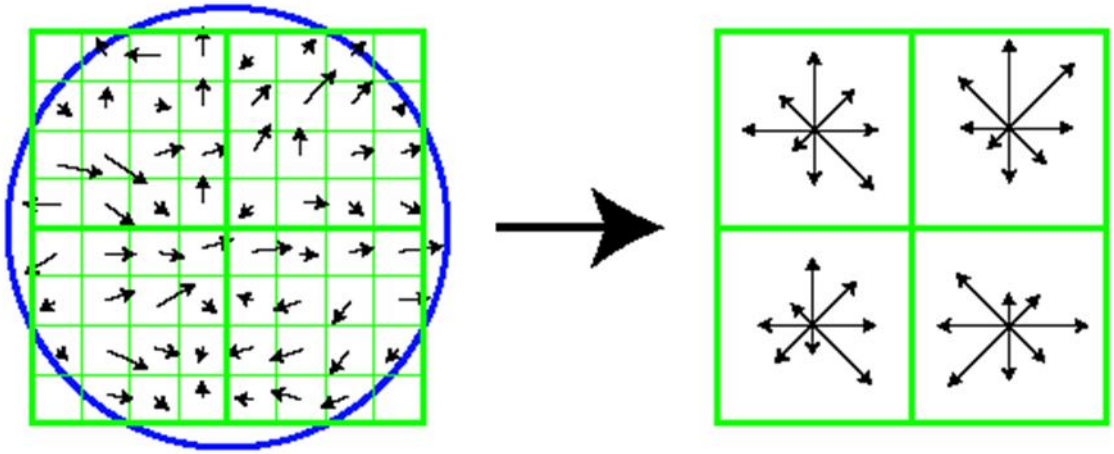
[Mata, Chum, Urban & Pajdla, '02]

[Sivic & Zisserman, '03]

Slide credit: Josef Sivic

Гистограммы градиентов SIFT

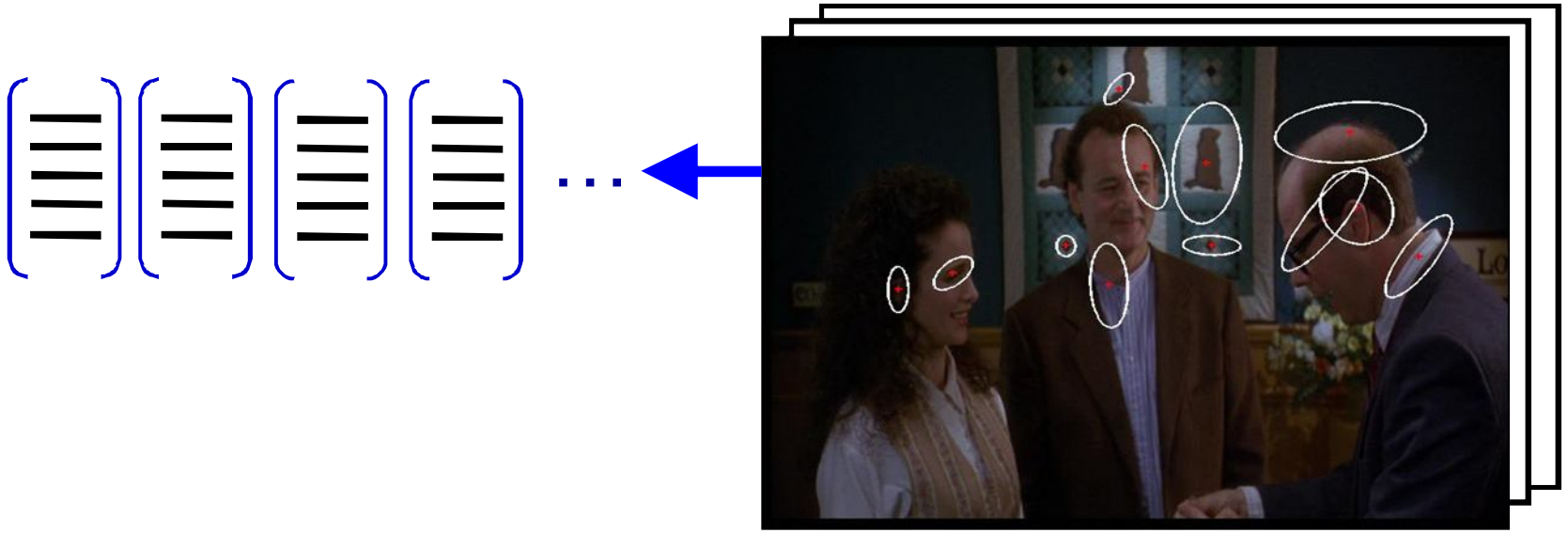
Microsoft Research



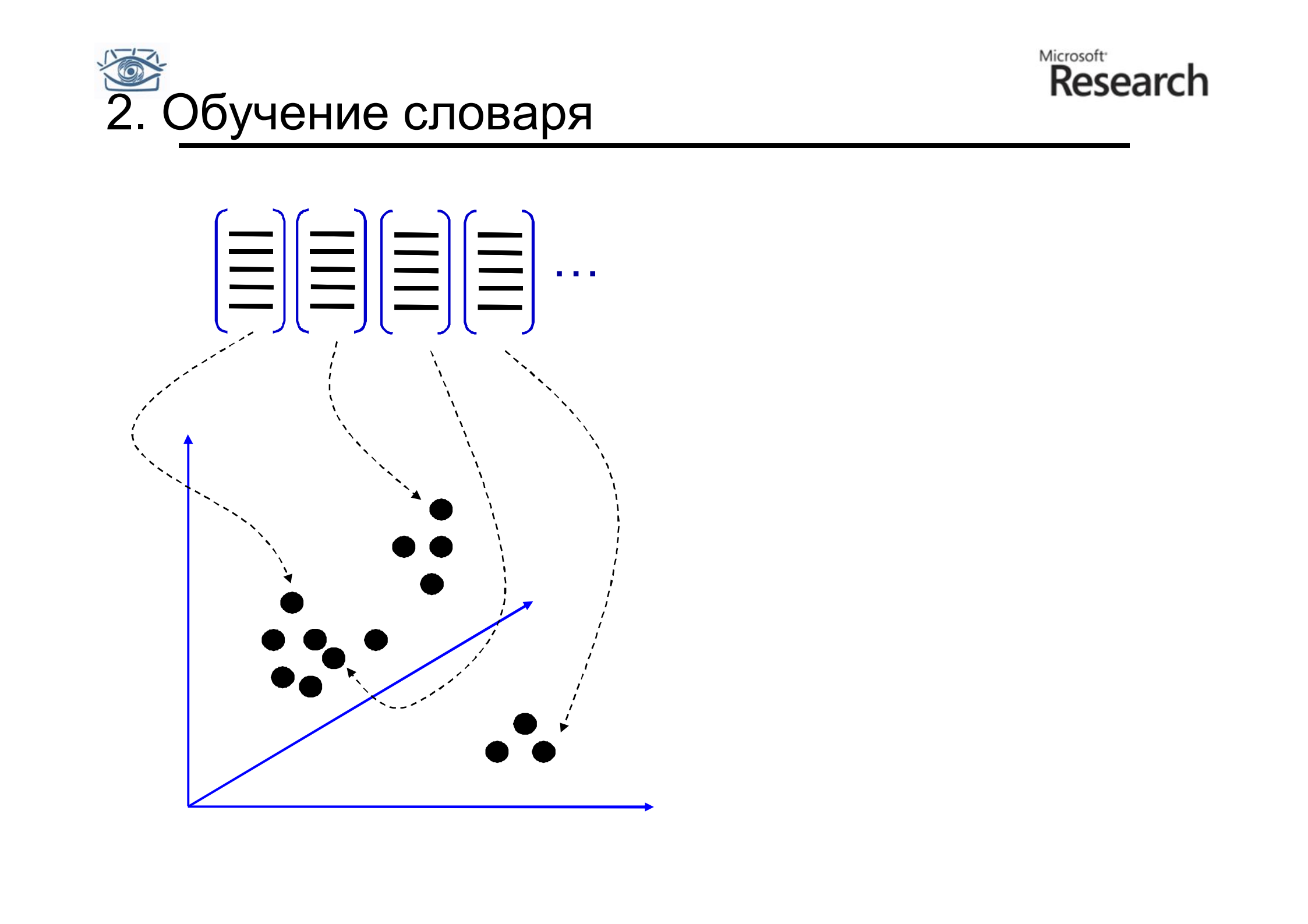
- Вычисляем градиент в каждой точке
- Строим гистограммы по окрестностям
- Обычно – сетка 4x4, в каждой гистограмма с 8ю ячейками
- Стандартная длина вектора-дескриптора – 128 (4*4*8)
- Устойчив к изменениям освещенности и небольшим сдвигам
- **Считается одним из самых эффективных дескрипторов для поиска изображений**

Microsoft Research

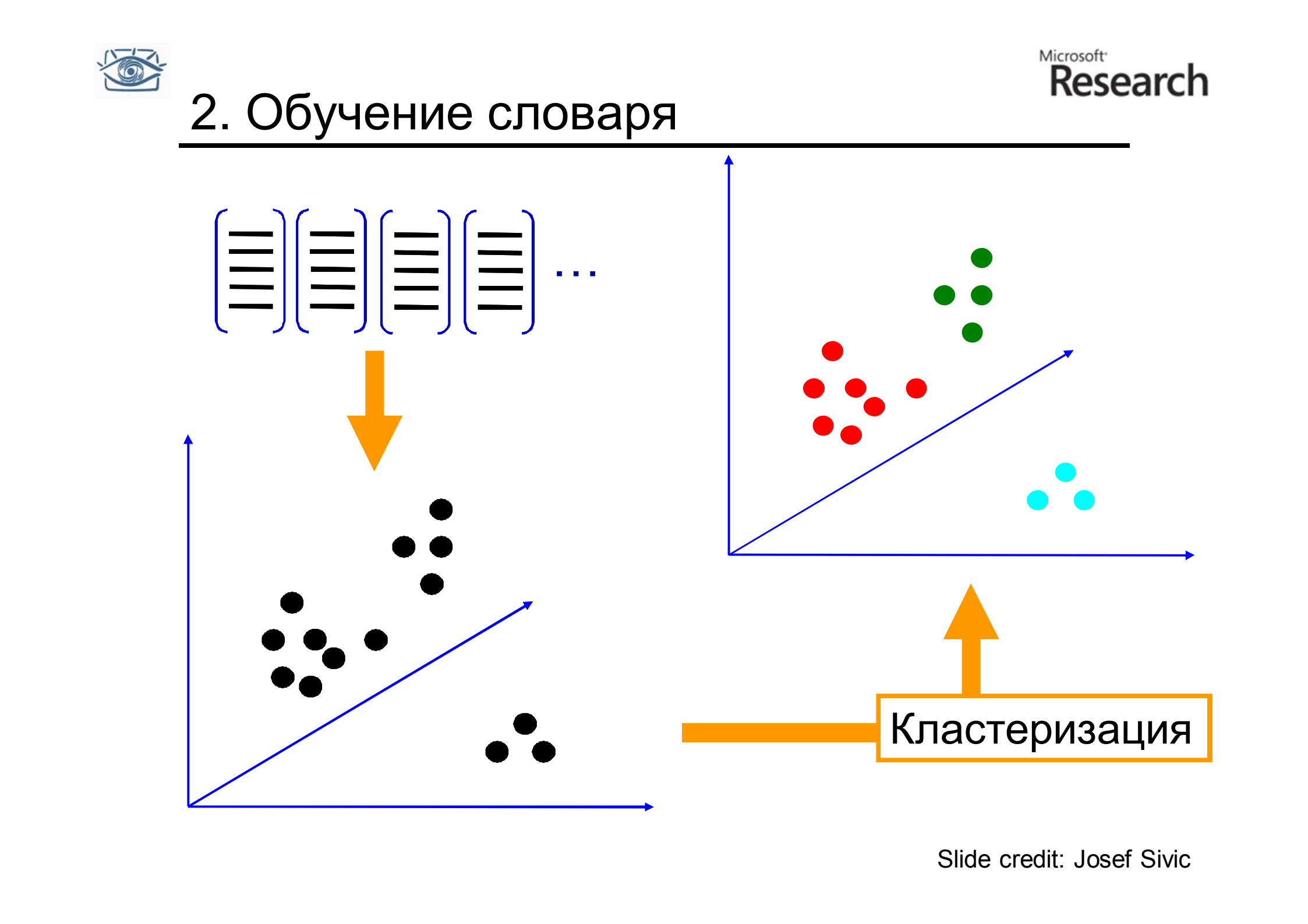
1. Извлечение особенностей



Неупорядоченный набор особенностей!



2. Обучение словаря


2. Обучение словаря
Microsoft Research
...
Кластеризация
Slide credit: Josef Sivic

Microsoft Research

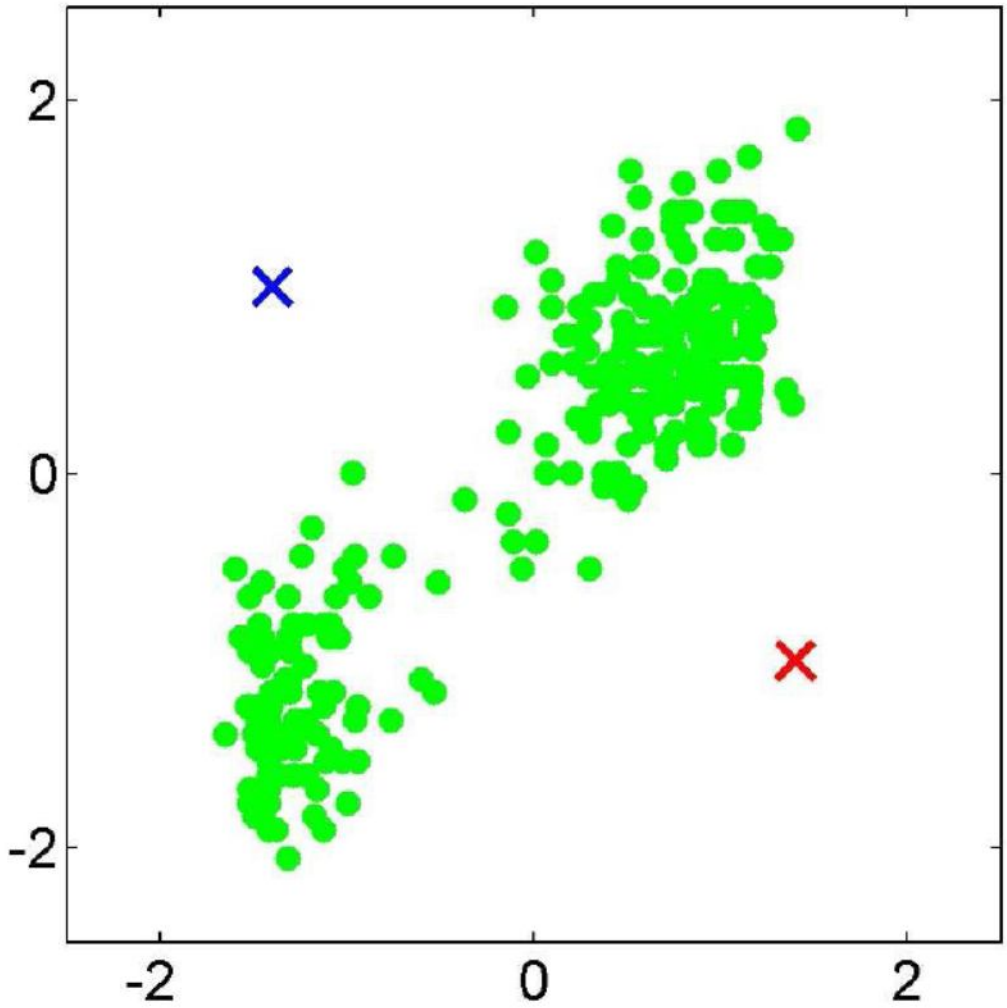
Кластеризация K-средними

- Минимизируем сумму квадратов Евклидовых расстояний между точками x_i и ближайшими центрами кластеров m_k

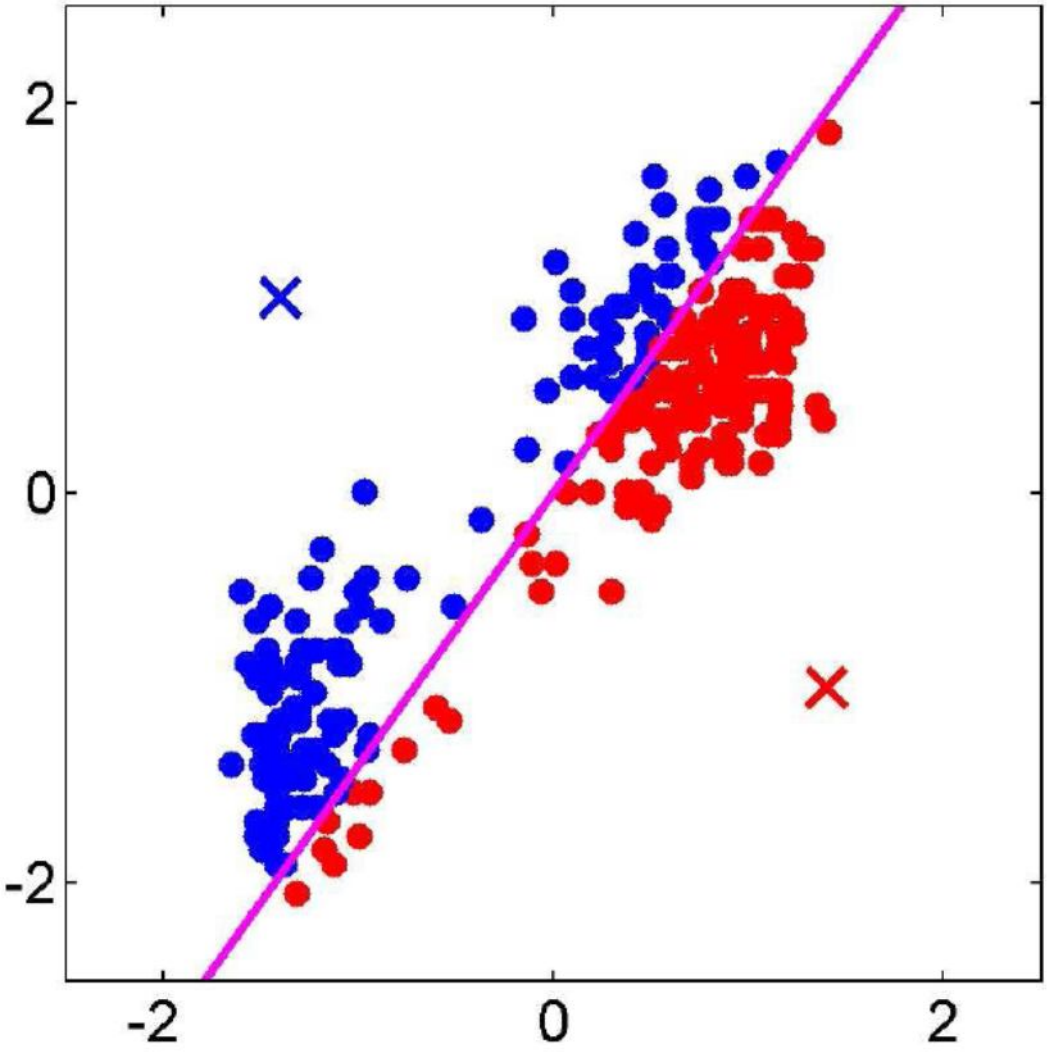
$$D(X,M) = \sum_{\text{cluster } k} \sum_{\substack{\text{point } i \text{ in} \\ \text{cluster } k}} (x_i - m_k)^2$$

- Алгоритм:
- Случайно инициализируем K центров кластеров
- Повторяем до сходимости:
  - Назначаем каждую точку ближайшему центру
  - Пересчитываем центр каждого кластера как среднее всех назначенных точек

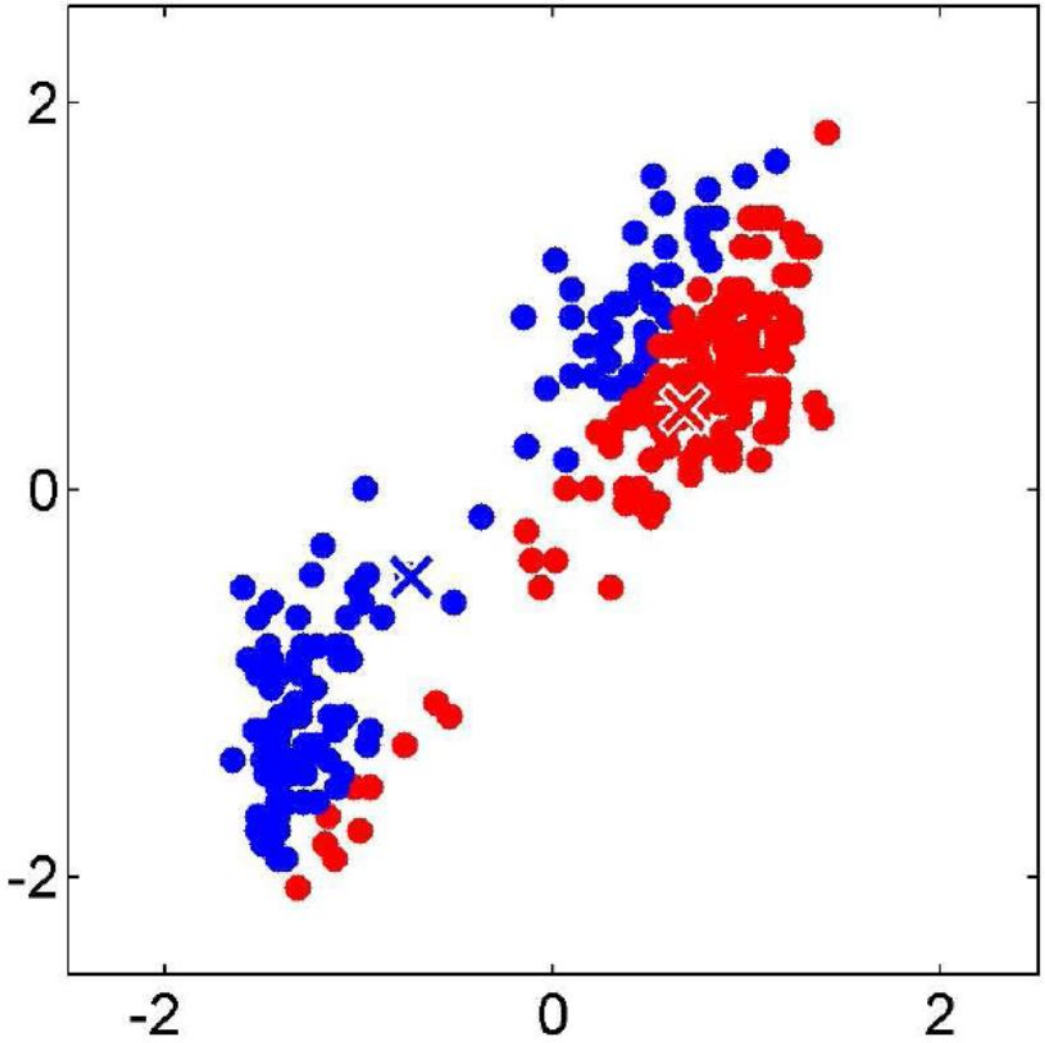
Иллюстрация



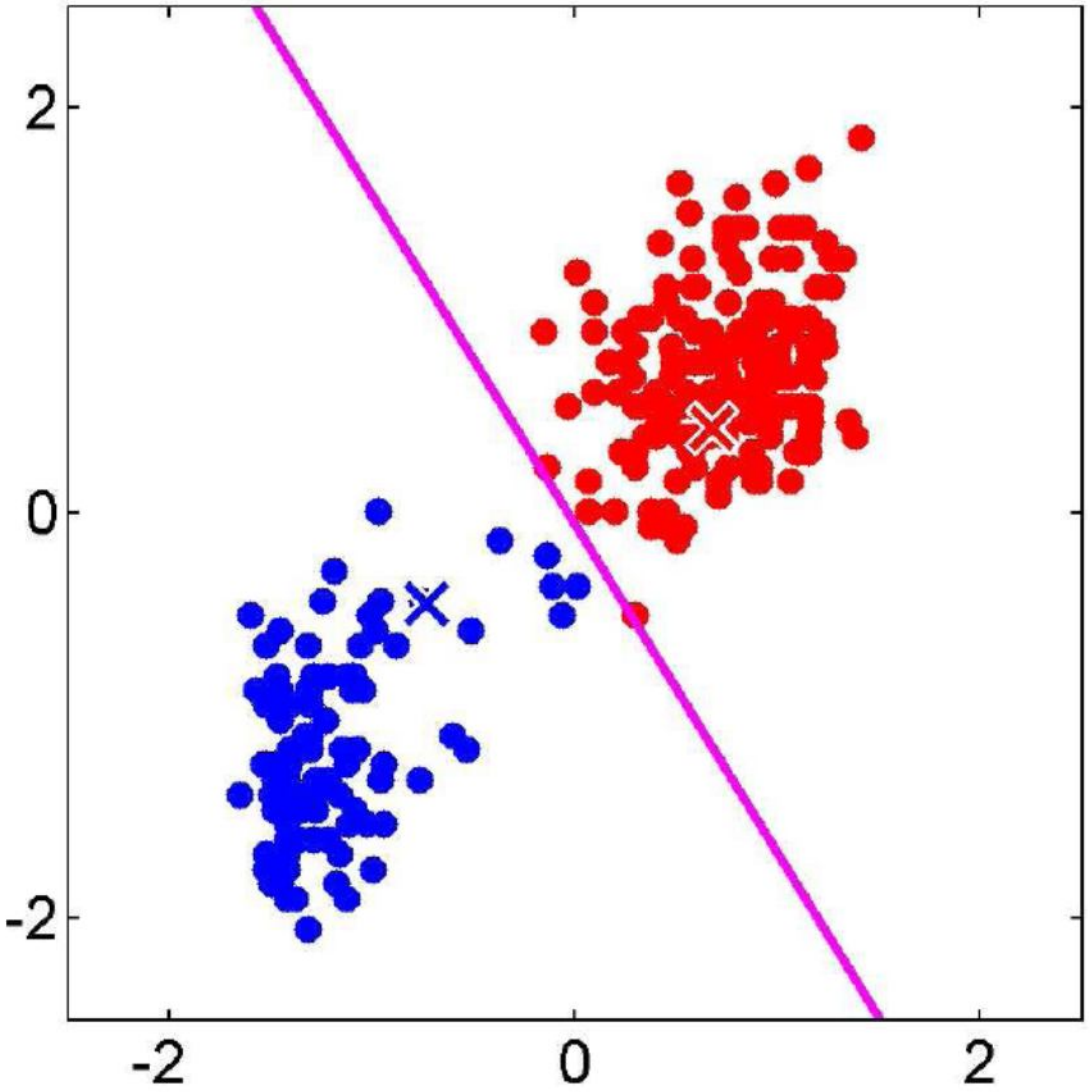
Иллюстрация



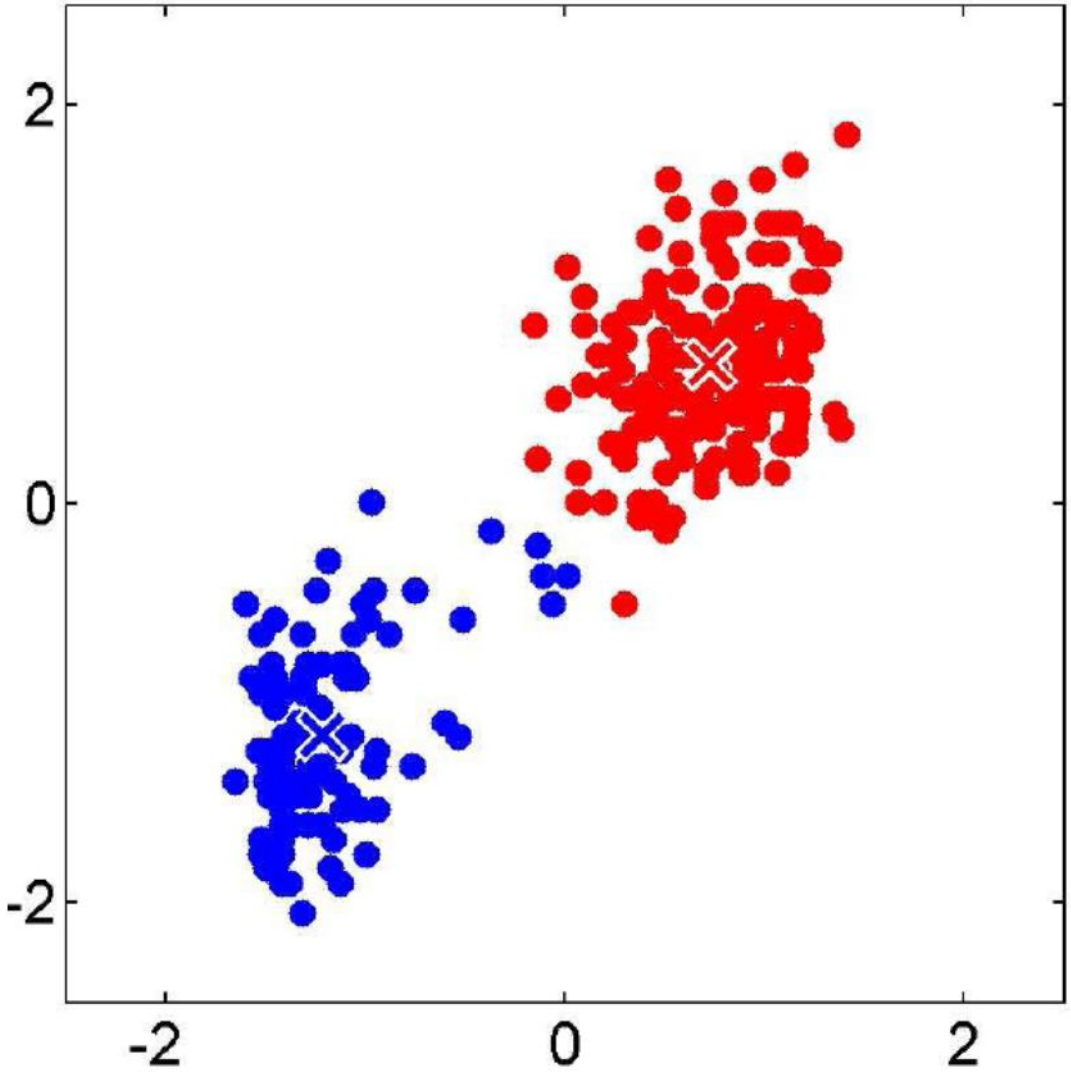
Иллюстрация



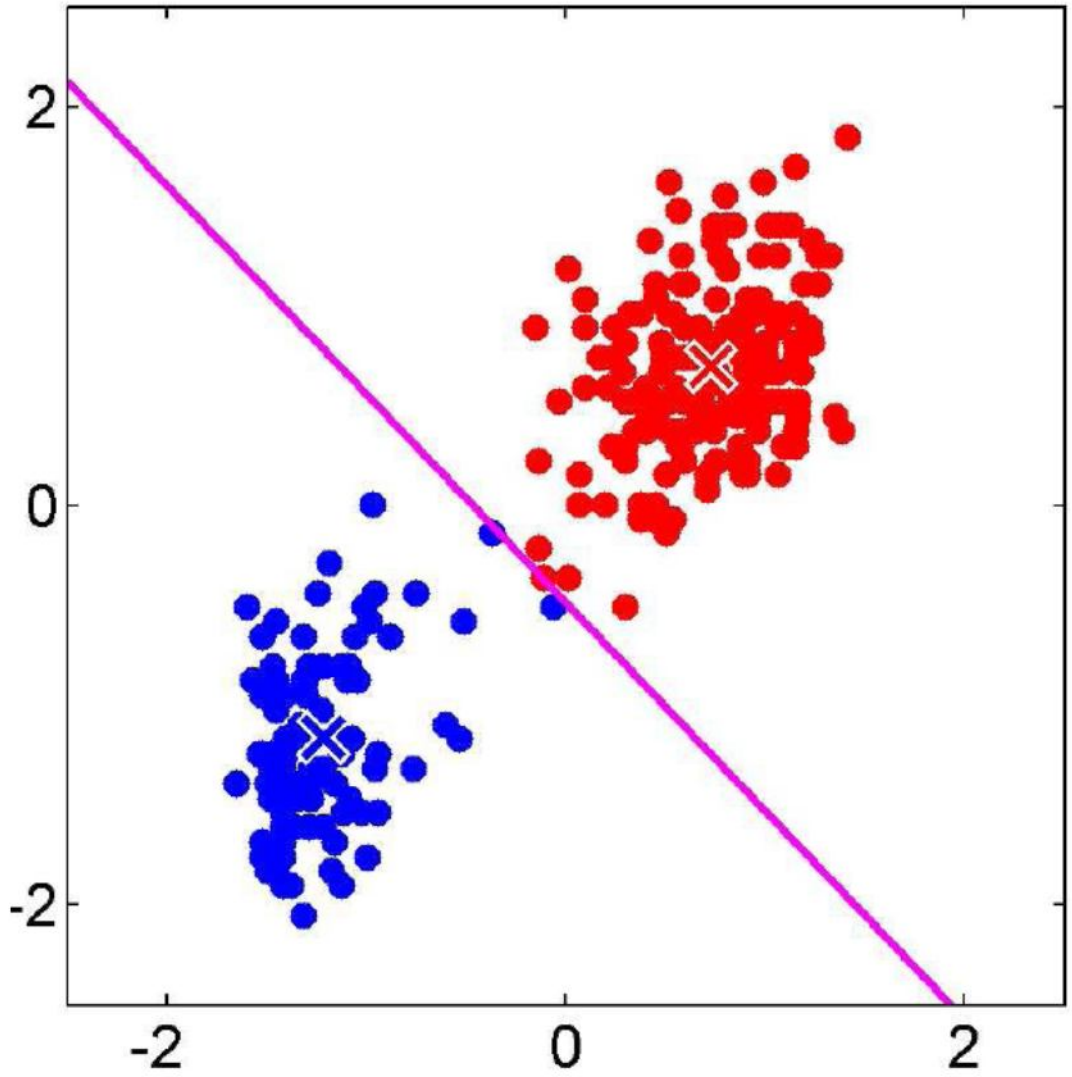
Иллюстрация



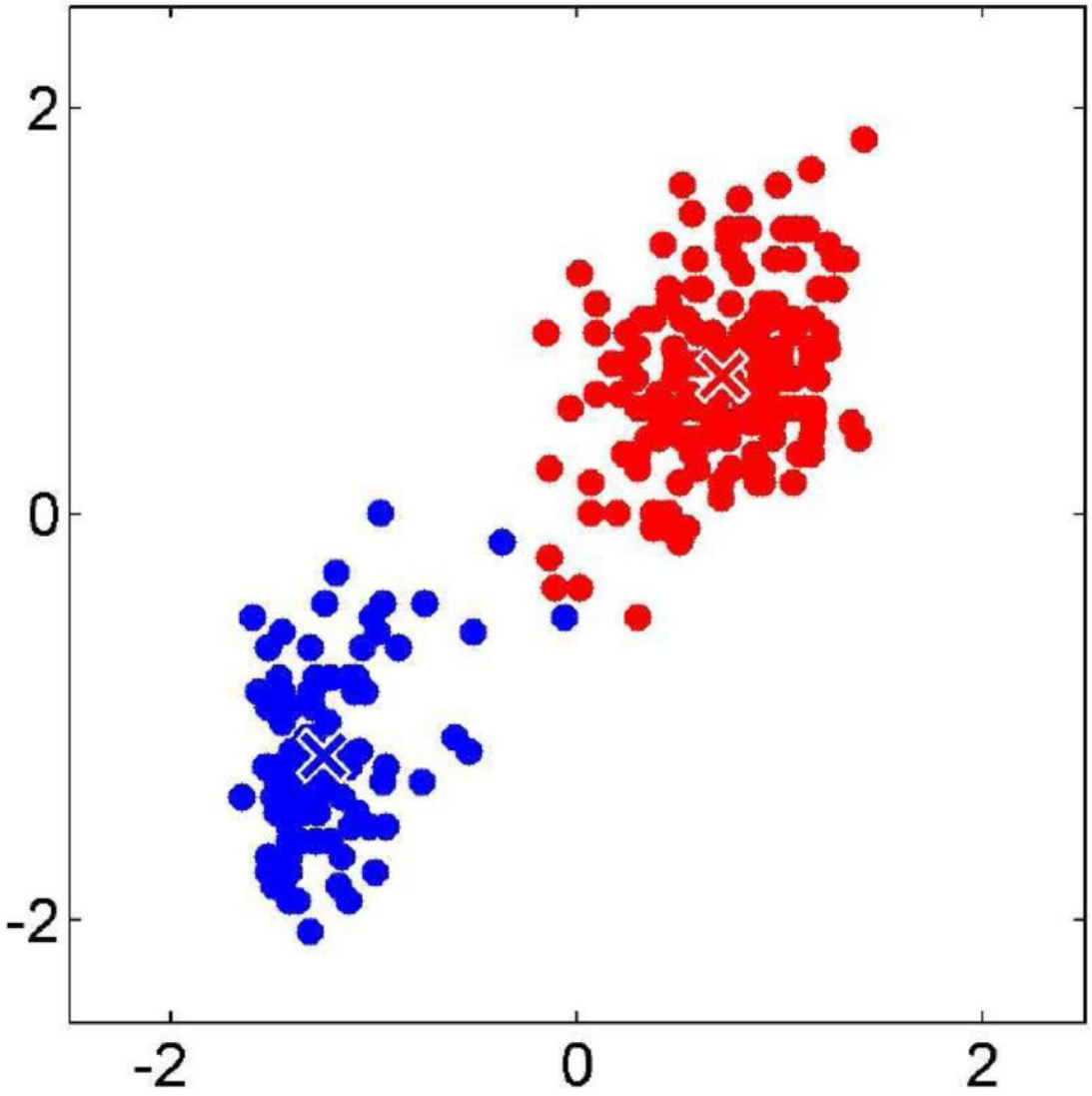
Иллюстрация



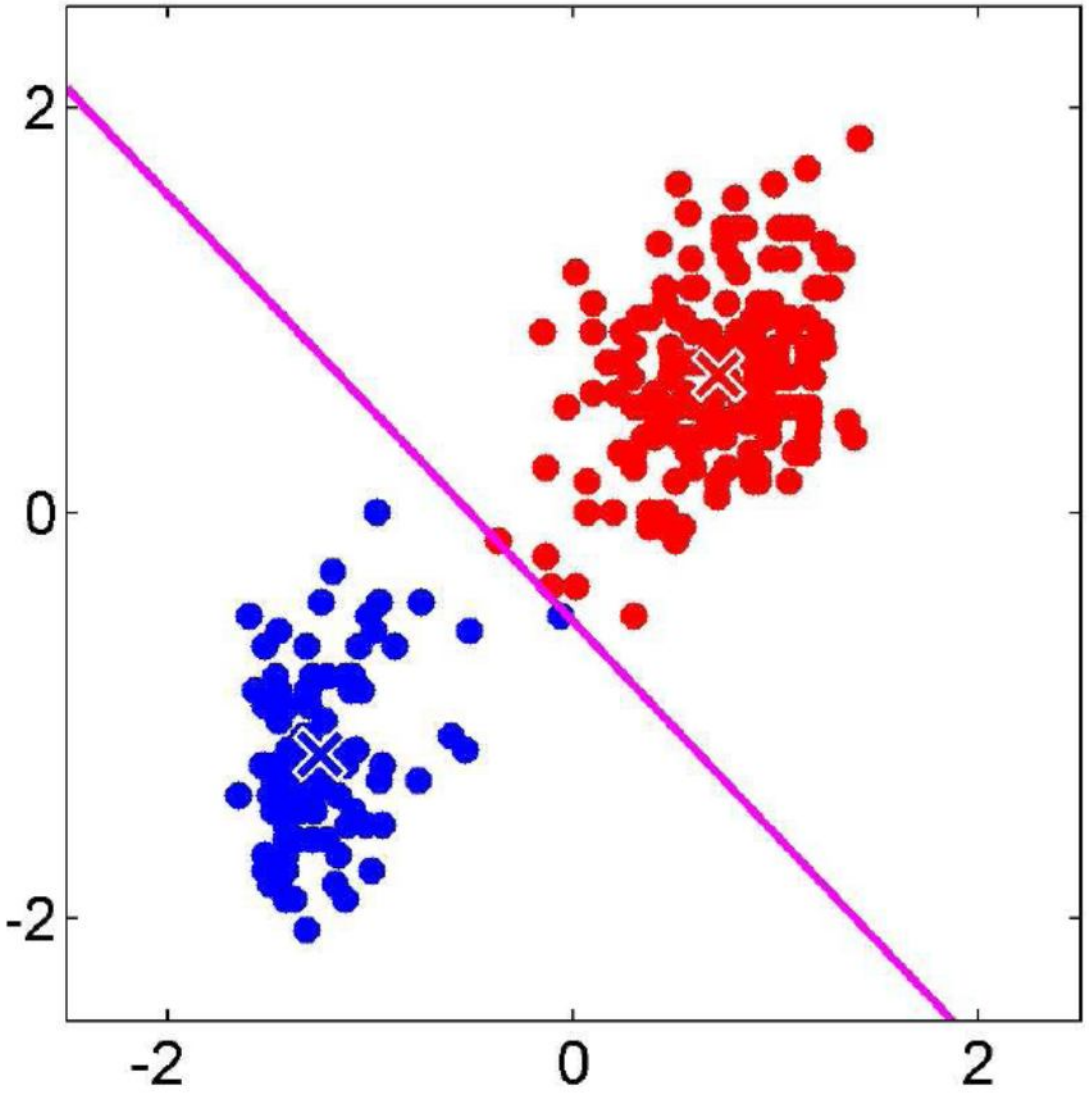
Иллюстрация



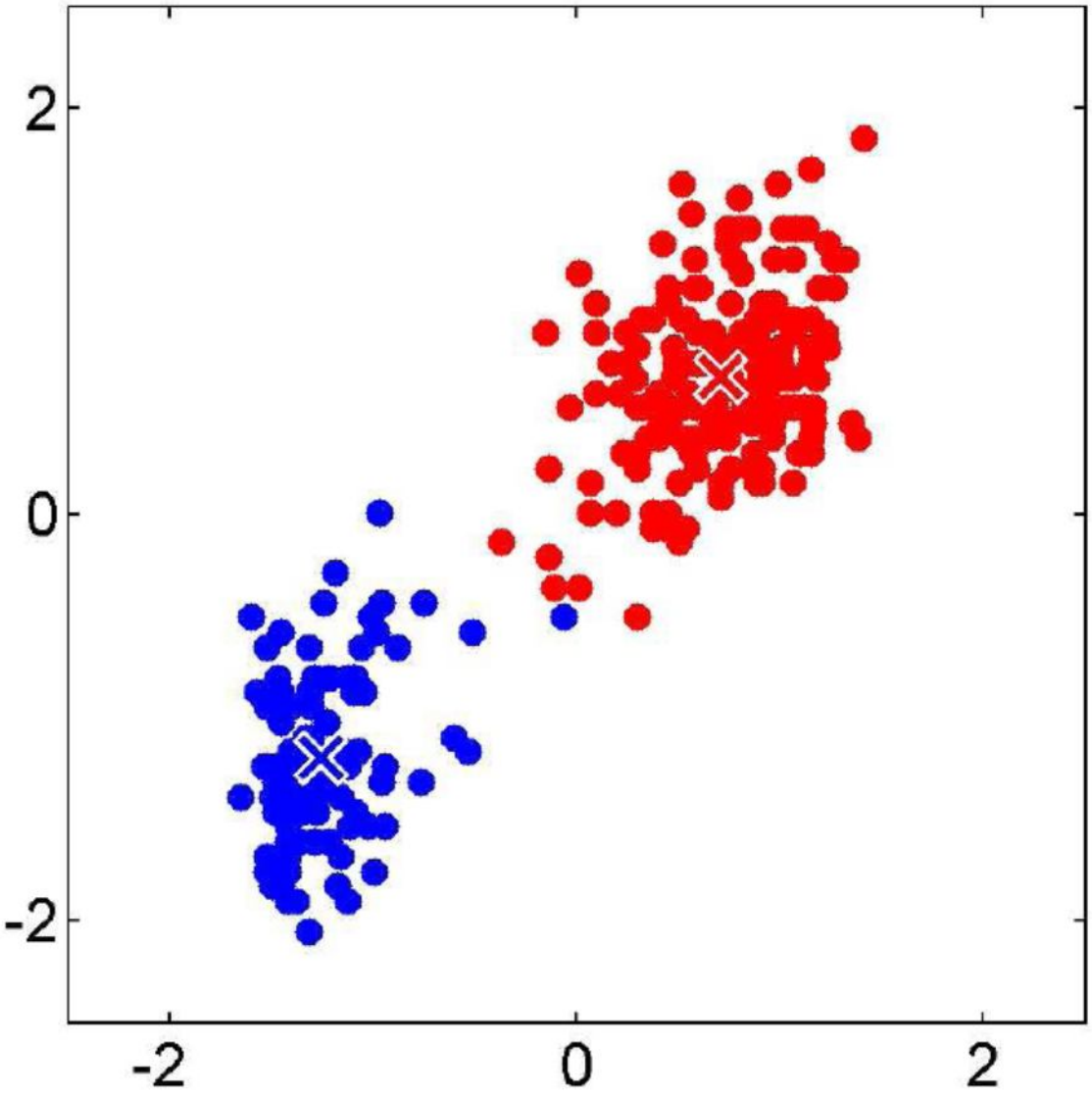
Иллюстрация



Иллюстрация

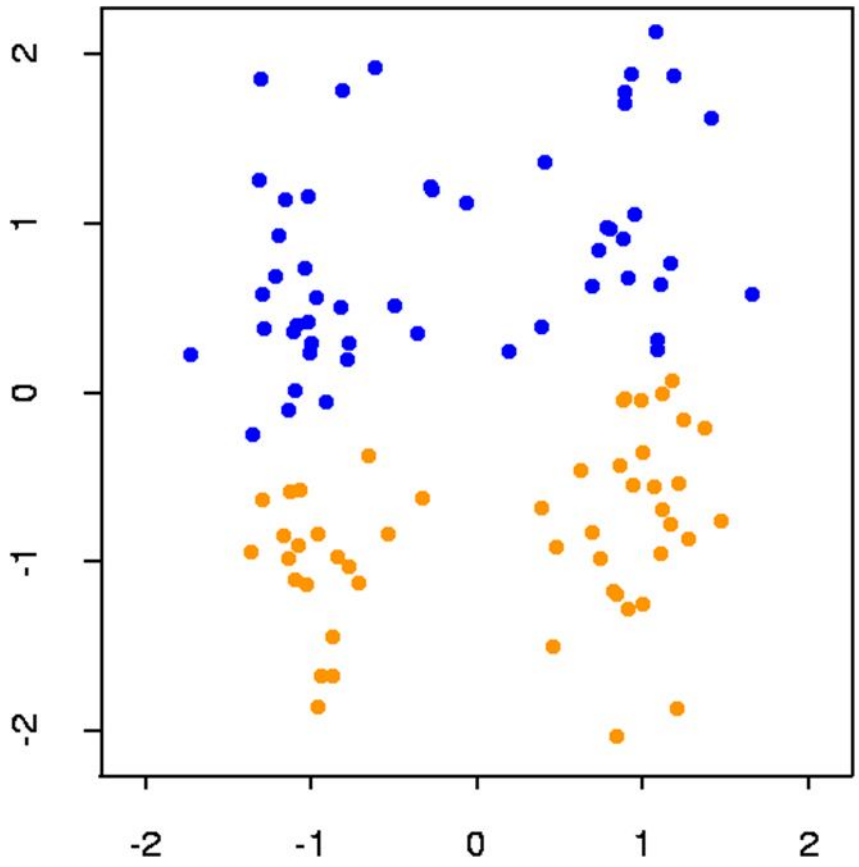


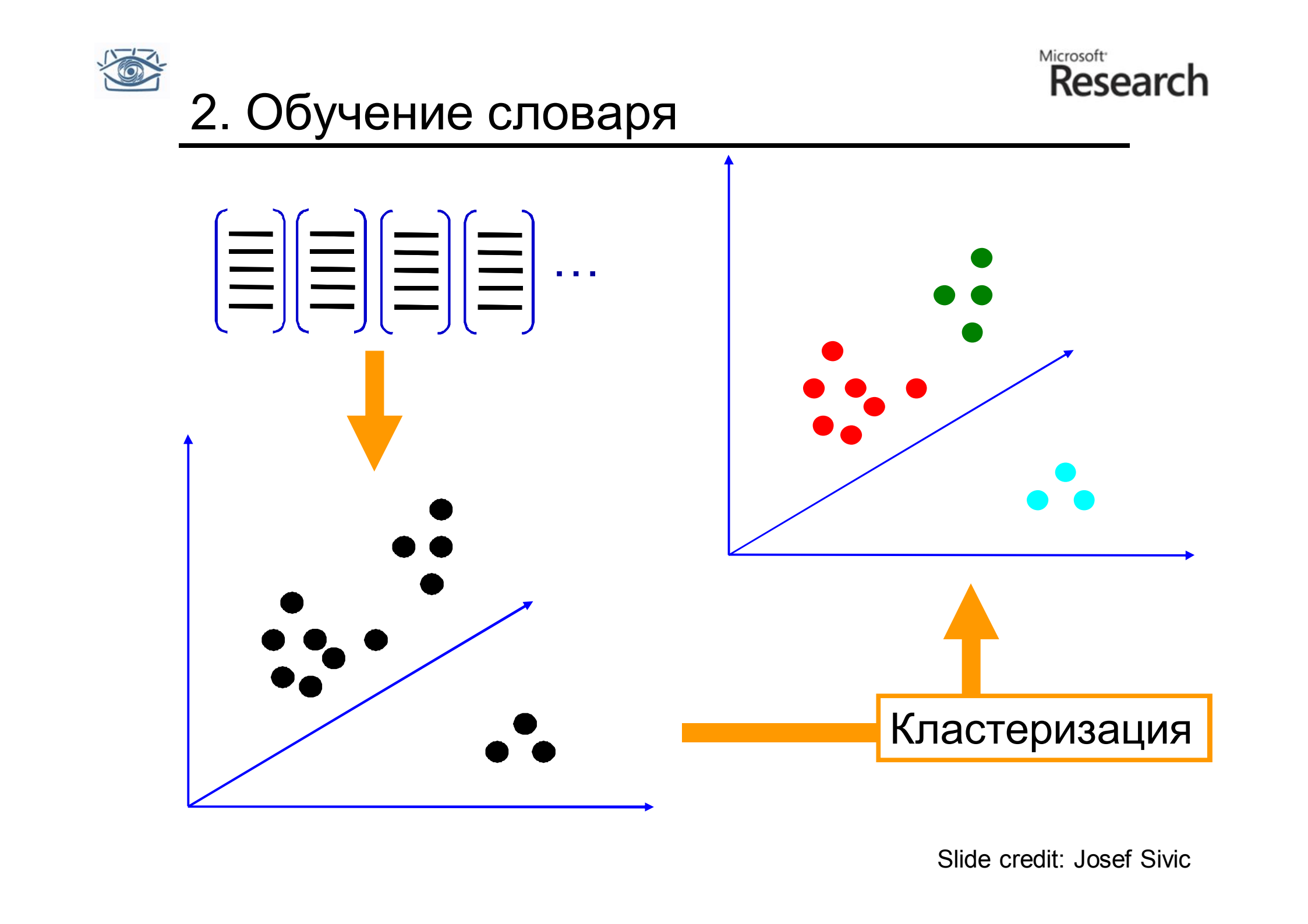
Иллюстрация



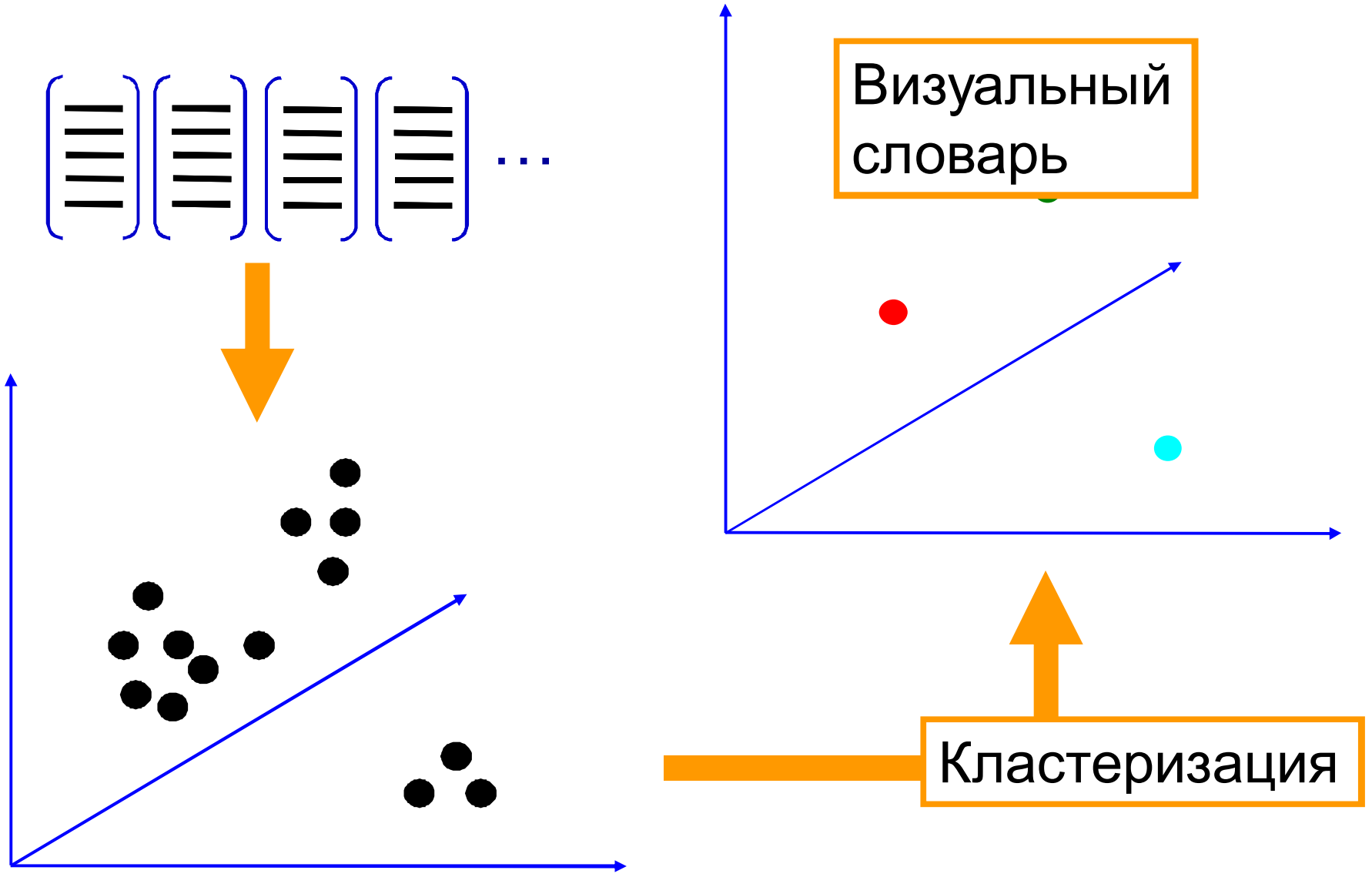
Алгоритм К-средних

- Однопараметрический
  - Требует знания только о количестве кластеров
- Рандомизирован
  - Зависит от начального приближения
- Не учитывает строения самих кластеров
- Часто применяется




2. Обучение словаря
Microsoft Research
...
Кластеризация
Slide credit: Josef Sivic

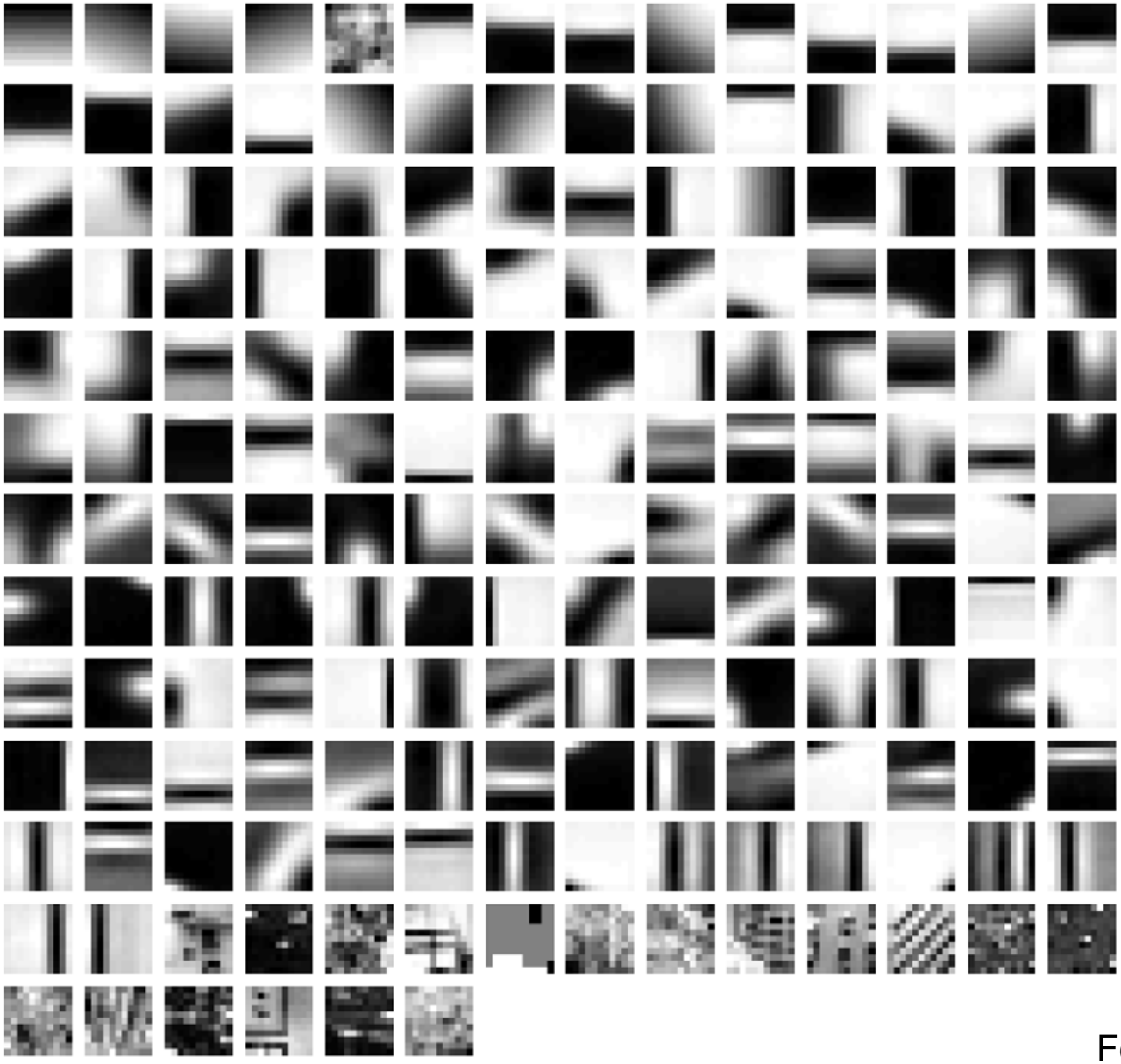
2. Обучение словаря



Slide credit: Josef Sivic

Microsoft Research

Пример словаря



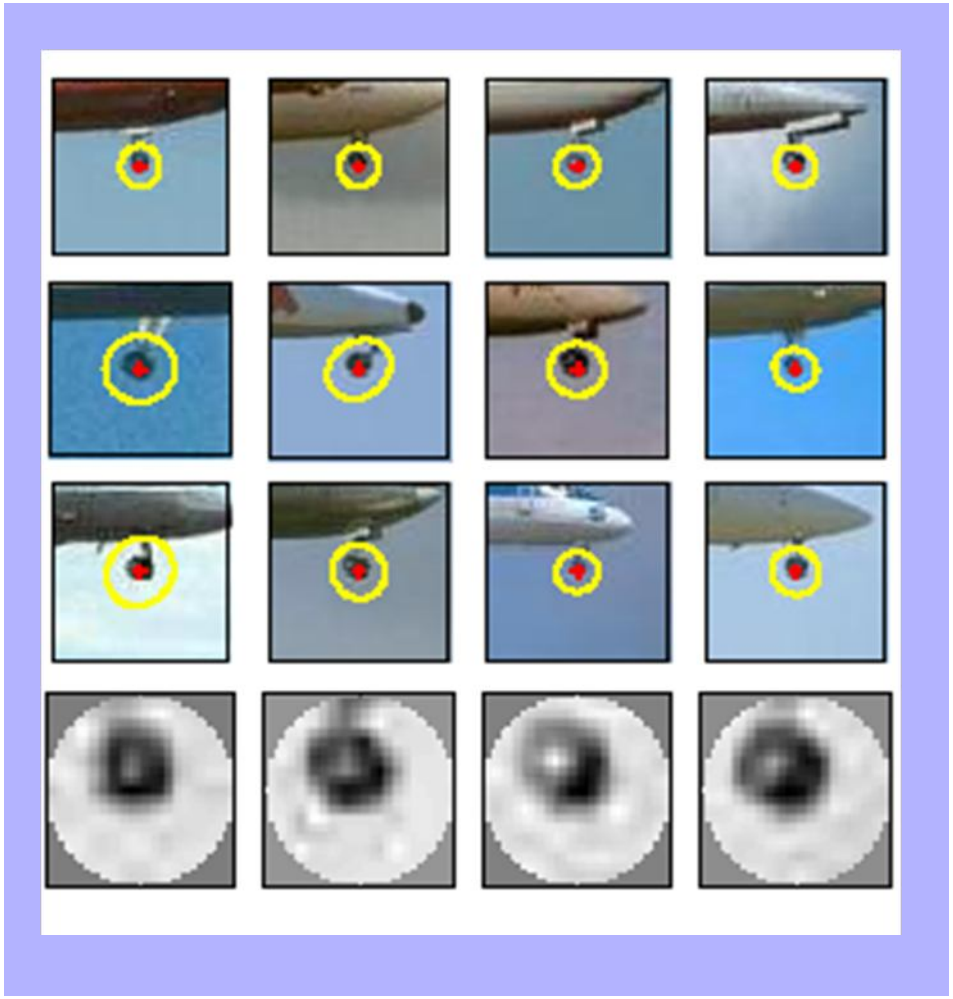
Fei-Fei et al. 2005

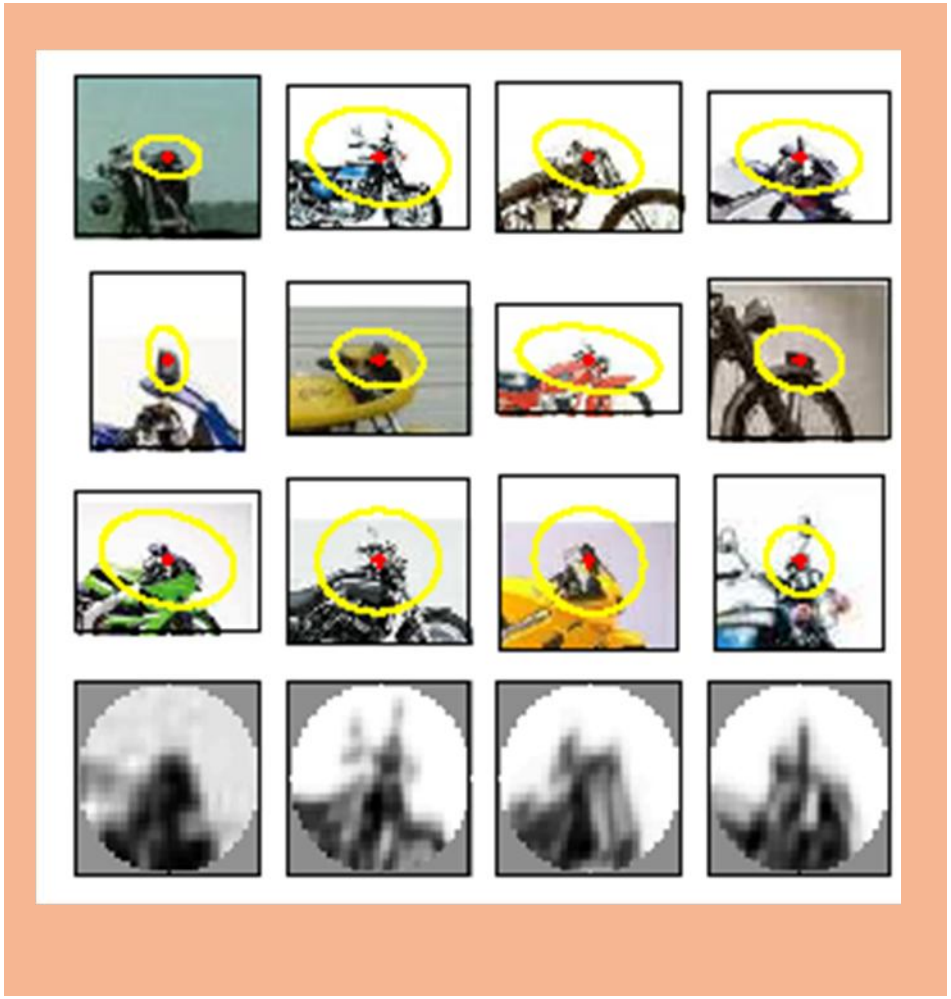
Microsoft Research

Квантование

- Теперь мы можем для каждого фрагмента изображения найти слово из словаря
  - Находим ближайшее по дескриптору слово в словаре
- «Квантование»
  - Потому что фактически сопоставляем каждому фрагменту изображения число от 1 до N, где N – размер словаря

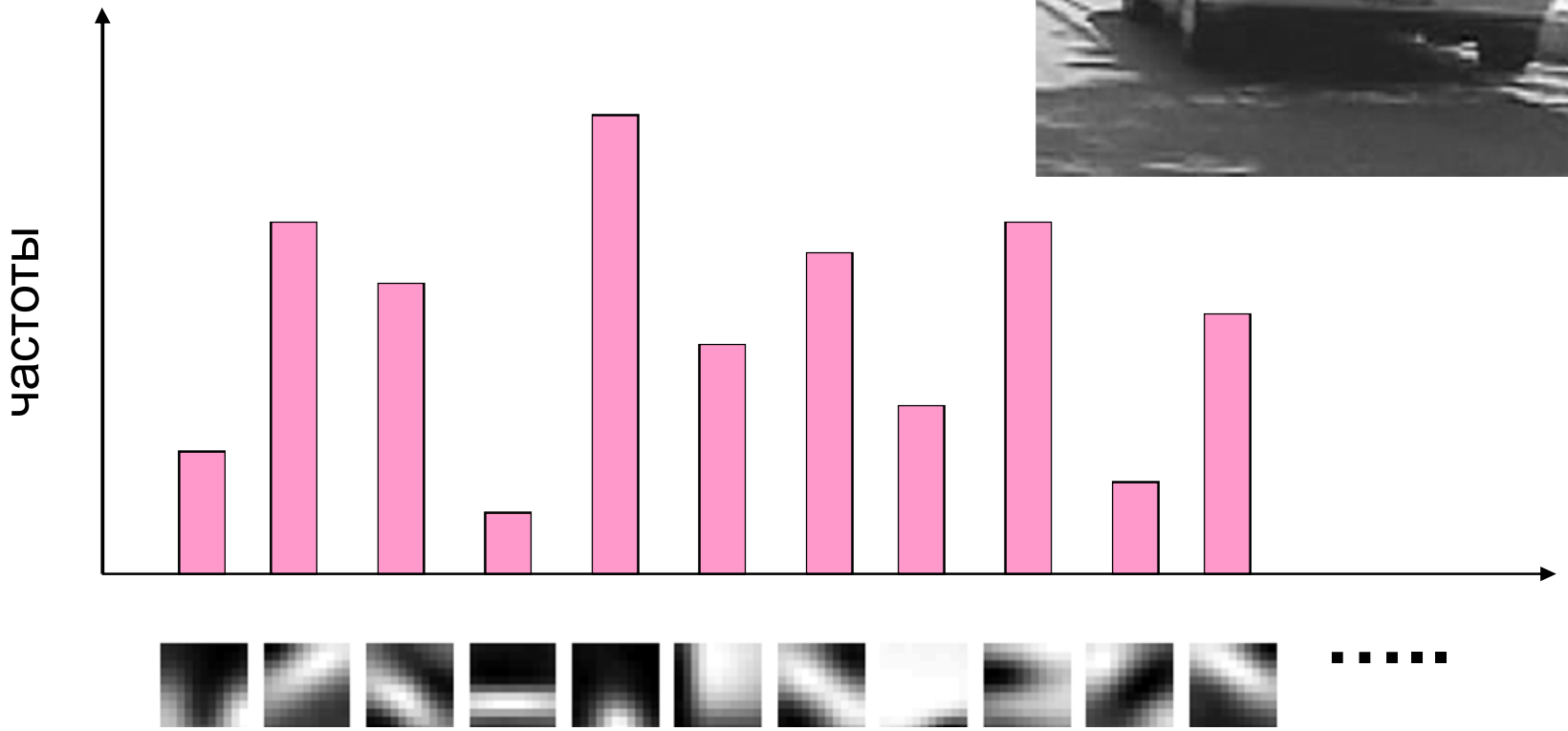
Примеры визуальных слов





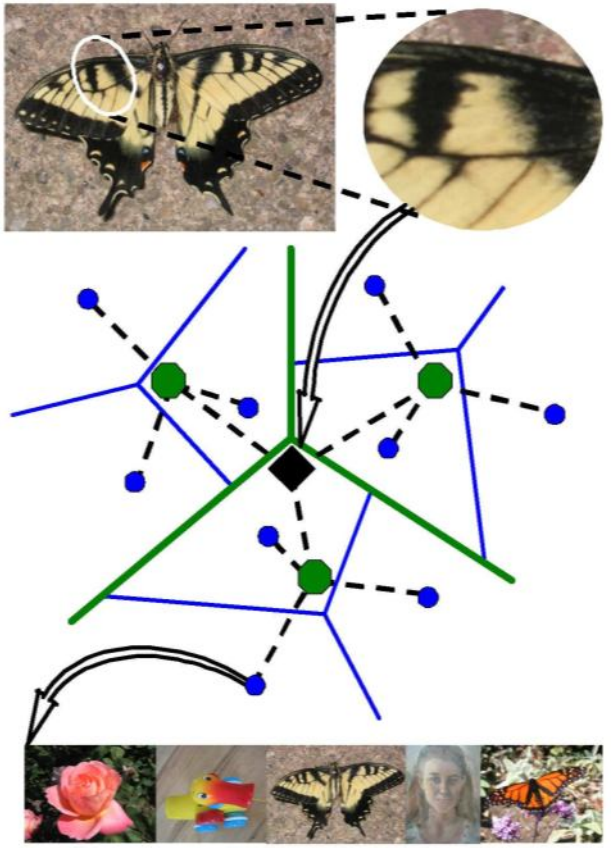
Sivic et al. 2005

3. Мешок слов



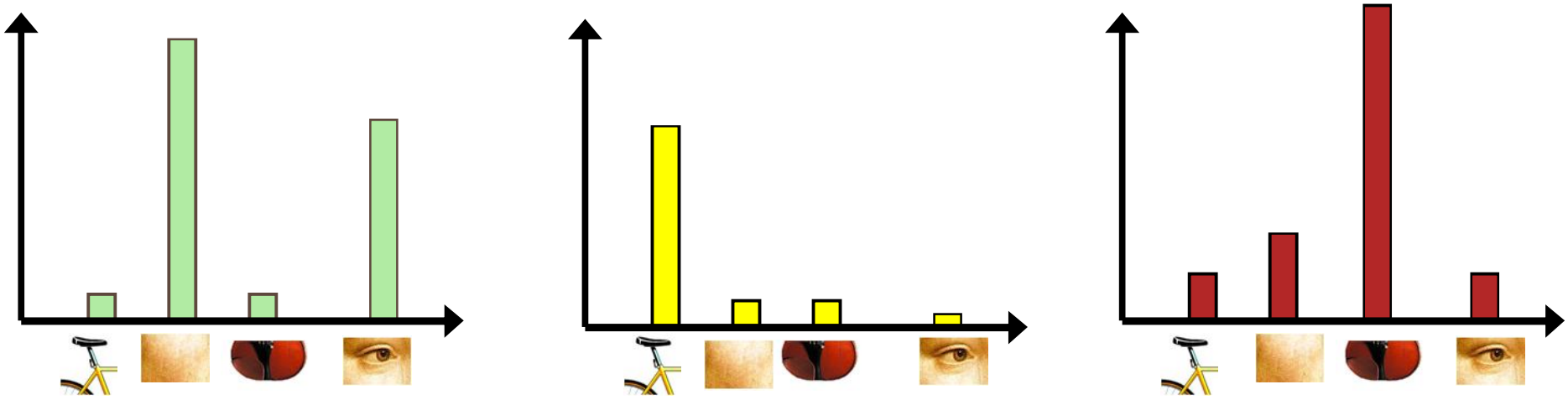
Визуальные словари

- Как выбрать размер словаря?
  - Маленький: слова не могут описать все особенности
  - Большой: переобучение
- Вычислительная сложность
  - Деревья словарей (Nister & Stewenius, 2006)
  - Approximate nearest-neighbour
  - Хеширование



Классификация изображений

- Как мы будем распознавать изображение, если представление изображения в виде сумки слов уже получено?



Классификация изображений

Microsoft Research

- Воспользуемся методом SVM
- Поскольку вектор-признак – гистограмма, нам потребуются специальные ядра
- Ядро пересечения гистограмм:

$$I(h_1, h_2) = \sum_{i=1}^{N} \min(h_1(i), h_2(i))$$

- Обобщенное гауссово ядро:

$$K(h_1, h_2) = \exp\left(-\frac{1}{A} D(h_1, h_2)^2\right)$$

- *D* может быть евклидовым расстоянием, χ^2 , и т.д.

J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, Local Features and Kernels for Classifcation of Texture and Object Categories: A Comprehensive Study, IJCV 2007

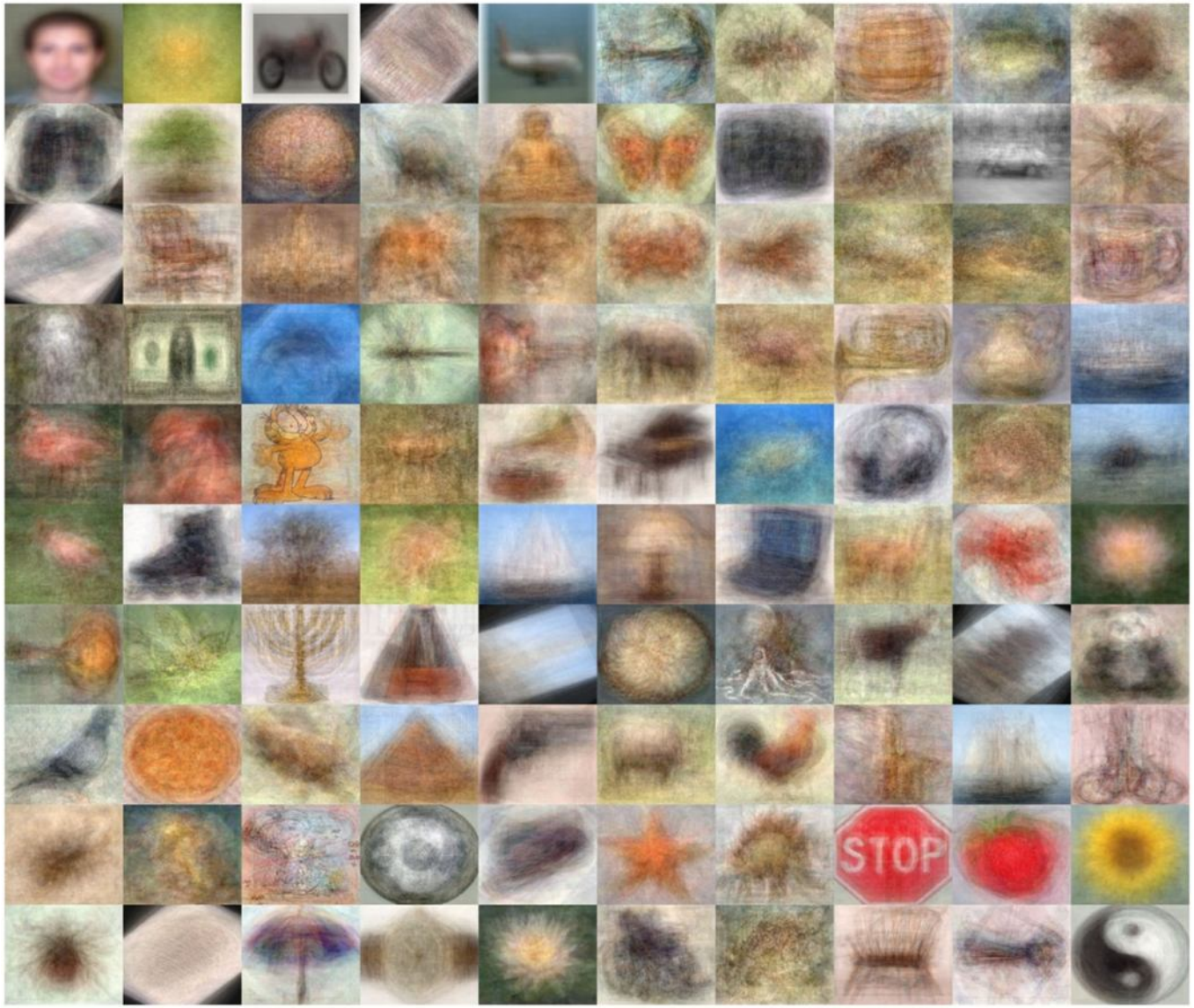
Microsoft Research

Сравнение ядровых функций

	PASCAL (mean EER)	TRECVID (mean InfAP)
Linear	0.874	0.041
HI	0.909	0.052
Gaussian RBF	0.892	0.075
Laplacian RBF	**0.921**	**0.087**
Sub-linear RBF	**0.922**	**0.084**
χ^2 RBF	**0.925**	**0.083**

Caltech-101

Microsoft Research



Microsoft Research

Резюме метода:

- Обучение алгоритма:
  - Собираем множество фрагментов
  - Кластеризуем и строим словарь
  - Квантуем каждый фрагмент по словарю
  - Считаем «мешки слов» для каждого изображения
  - Обучаем SVM на мешках слов

- Классификация изображения:
  - Выбираем фрагменты из изображения
  - Квантуем каждый фрагмент по словарю
  - Строим «мешок слов» для изображения
  - Применяем классификатор SVM

Microsoft Research

Резюме лекции

- Машинное обучение позволяет использовать большое количество неинтуитивных простых признаков
- Чем больше признаков и сложнее метод – тем больше нам нужно данных для обучения
- Идеального решения задачи не существуют – нам нужно выбирать оптимальный баланс параметров

Microsoft Research

На следующей лекции

- Поиск объектов в изображениях