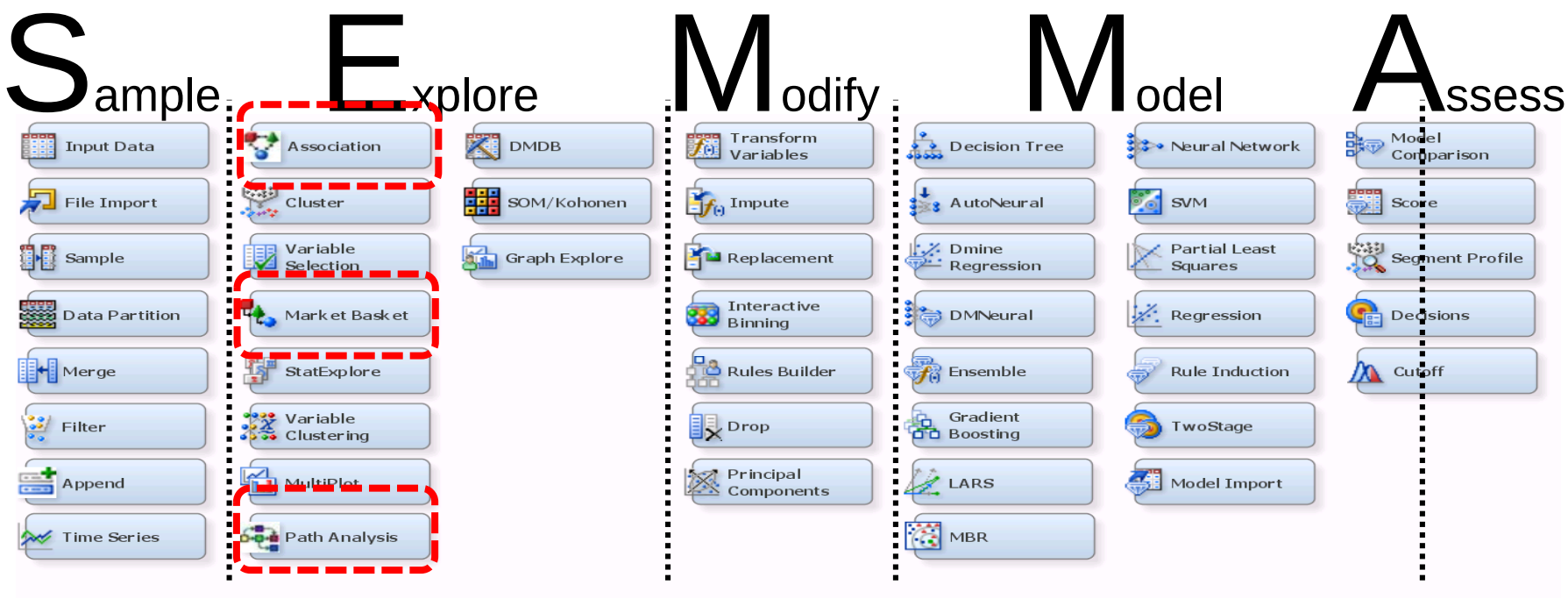
SAS ENTERPRISE MINER

ПОИСК АССОЦИАТИВНЫХ ПРАВИЛ

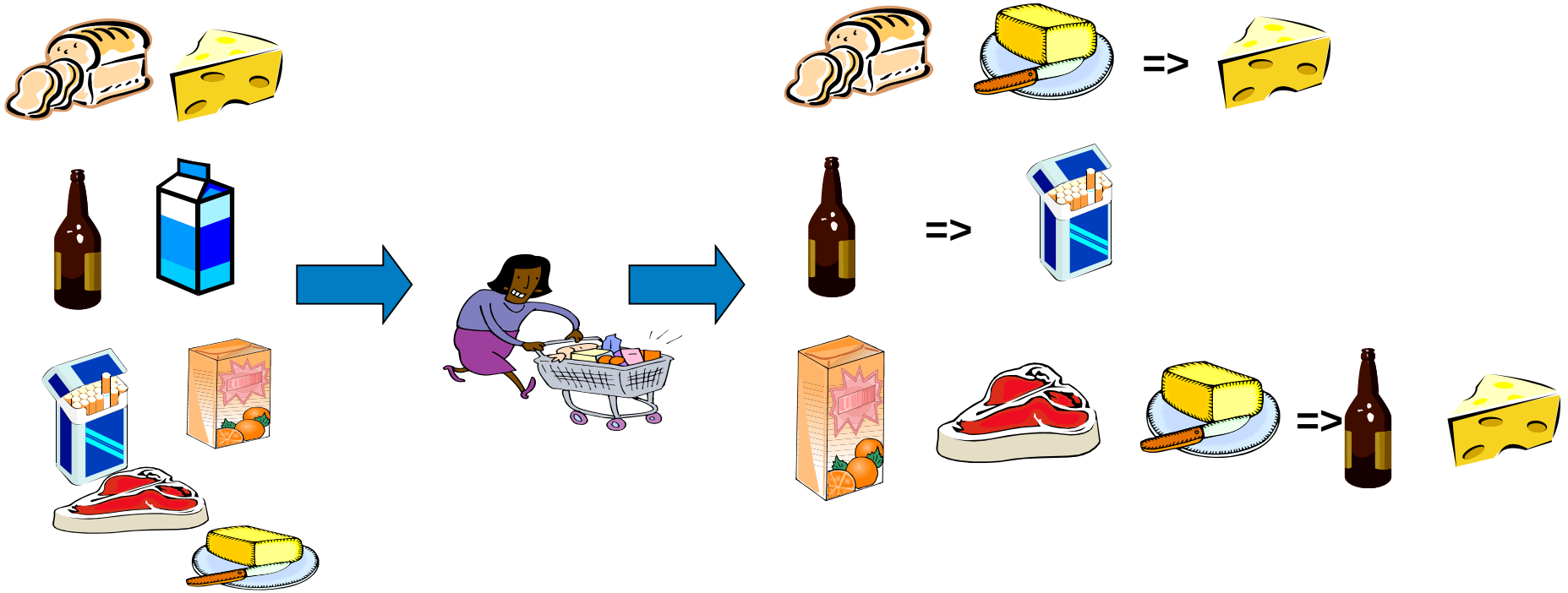
КОНЦЕПЦИЯ SEMMA



ТИПОВАЯ ПРИКЛАДНАЯ ЗАДАЧА: АНАЛИЗ «КОРЗИНЫ ПОКУПАТЕЛЯ»

Ассортимент супермаркета

Интересные правила



Задача Определить интересные правила в предпочтениях покупателей при выборе товара

АССОЦИАТИВНЫЙ АНАЛИЗ

- Правила с семантикой:
  - в s% случаев ЕСЛИ верно A и B и C, ТО с достоверностью c будет верно D и E
  - A&B&C=>D&E, где A,B,C,D,E – (различные!) предикаты, s – поддержка (support), c – достоверность (confidence)
- Основная задача:
  - найти все интересные правила, с заданными ограничениями по s и c (возможно задание дополнительных ограничений на предикаты и сами правила)
- Основной математический аппарат:
  - дискретная математика, математическая логика, комбинаторная оптимизация (на основе метода «ветвей и границ» - вариации полного перебора с отсевом подмножеств допустимых решений, заведомо не содержащих оптимальных решений).

АССОЦИАТИВНЫЙ АНАЛИЗ

- Тип моделей:
  - Как правило, «описательный» (descriptive) Data mining => одна из задач - наглядное представление правил
- Тип обучения:
  - «без учителя» (unsupervised) => тренировочный набор не размечен
- Типы правил:
  - Булевы!!!
  - Числовые – нужна дискретизация, интервалы как булевы предикаты
  - Иерархические – если определена иерархия для значений атрибутов
  - Временные – как правило, семантика «в s случаях если произошло A и B, то потом случится C и D с вероятностью c»)
  - Пространственные – предикаты определяют пространственные связи между объектами, например «рядом», «далеко» и т.п.

АССОЦИАТИВНЫЙ АНАЛИЗ

- Прикладные задачи:
  - «Экономические»: анализ корзины, маркетинг
  - «Безопасность» и Web usage mining: модели поведения пользователя
  - Text mining: поиск ключевых слов, характеристик и тематик
  - Биоинформатика, медицина
- Задачи анализа:
  - Поиск самих правил
  - Поиск исключений (из правил)
  - Выделение признаков (на основе правил)
  - Классификация и прогнозирование (на базе правил)

БУЛЕВЫ АССОЦИАТИВНЫЕ ПРАВИЛА

- Опр Найденные правила называются **интересными правилами**
- Опр Набор атрибутов X∪ Y называется **часто встречаемым** если supp(X∪Y)>=minsupp

$$I = \{i_1, i_2, ..., i_n\} - набор\ \ атрибутов$$

$$\textbf{\textit{Ассоциативное правило}}\ X \Rightarrow Y$$

$$X \subseteq I, Y \subseteq I, X \cap Y = \{\}$$

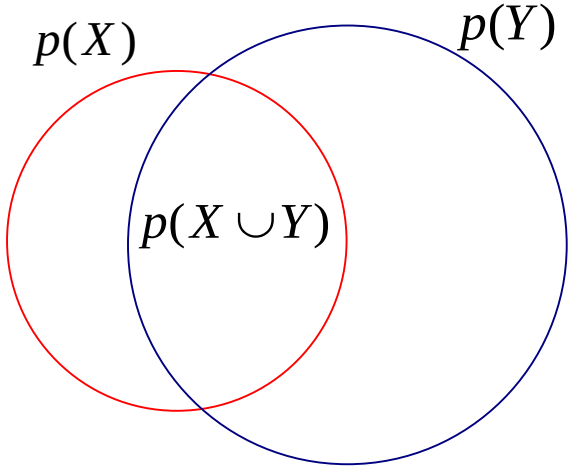
$$support(X \Rightarrow Y) = p(X \cup Y)$$

$$confidence\,(X \Rightarrow Y) = p(Y \mid X) = \frac{p(X \cup Y)}{p(X)}$$

$$Задача: найти\ все\ ассоциативные\ правила\ \ с$$

$$support \geq MinSup\ и\ confidence\ >= MinConf$$

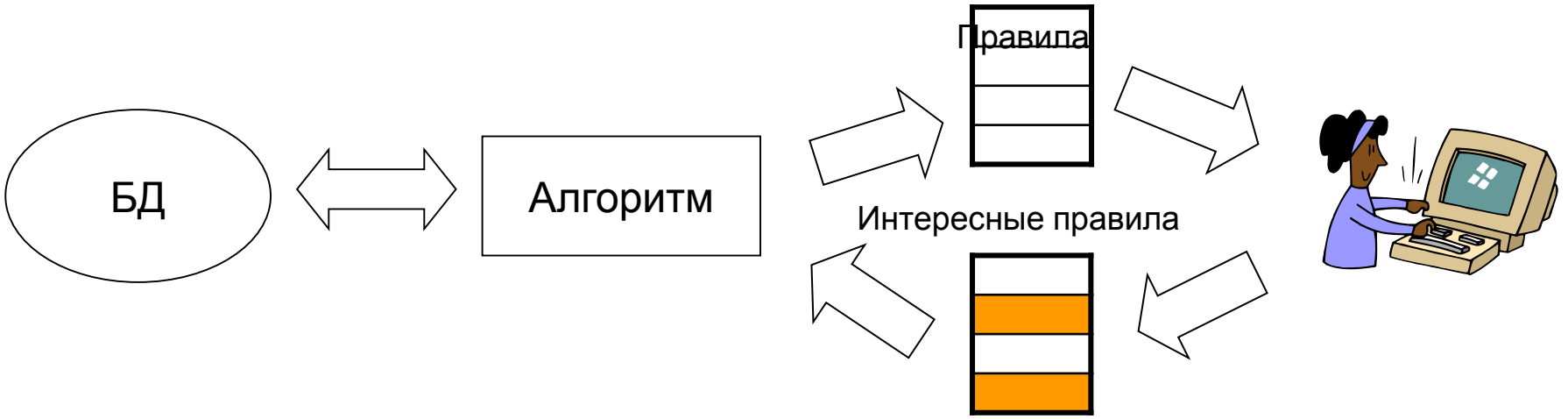
Множество транзакций



Популярные алгоритмы: Apriori, FP-tree

ИНТЕРЕСНОСТЬ

- Объективная
- Субъективная (на основе информации, заданной экспертом)
  - «Полезная» (Actionable)
  - «Неожиданная» (Unexpected)



КРИТИКА ДОСТОВЕРНОСТИ И ПОДДЕРЖКИ

- Пример: (Aggarwal & Yu, PODS98)
  - Среди 5000 студентов:
    - 3000 играют баскетбол, 3750 любят черный хлеб
    - 2000 и то и другое
  - *basketball* $\Rightarrow$ *bread* [40%, 66.7%] вводит в заблуждение, поскольку процент любителей хлеба 75% выше support 66.7%.
  - *basketball* $\Rightarrow$ *not bread* [20%, 33.3%] более полезное, хотя support и confidence ниже

	basketball	not basketball	sum(row)
bread	2000	1750	3750
not bread	1000	250	1250
sum(col.)	3000	2000	5000

КРИТИКА ПОДДЕРЖКИ И ДОСТОВЕРНОСТИ

- Пример:
  - X и Y: положительно коррелированны,
  - X и Z, отрицательно коррелированны
  - support и confidence больше у X=>Z
- Нужна мера «зависимости» типа

$$\frac{P(A \cap B)}{P(A)P(B)}$$

- P(B|A)/P(B) называется lift для
  - A => B

X	1	1	1	1	0	0	0	0
Y	1	1	0	0	0	0	0	0
Z	0	1	1	1	1	1	1	1

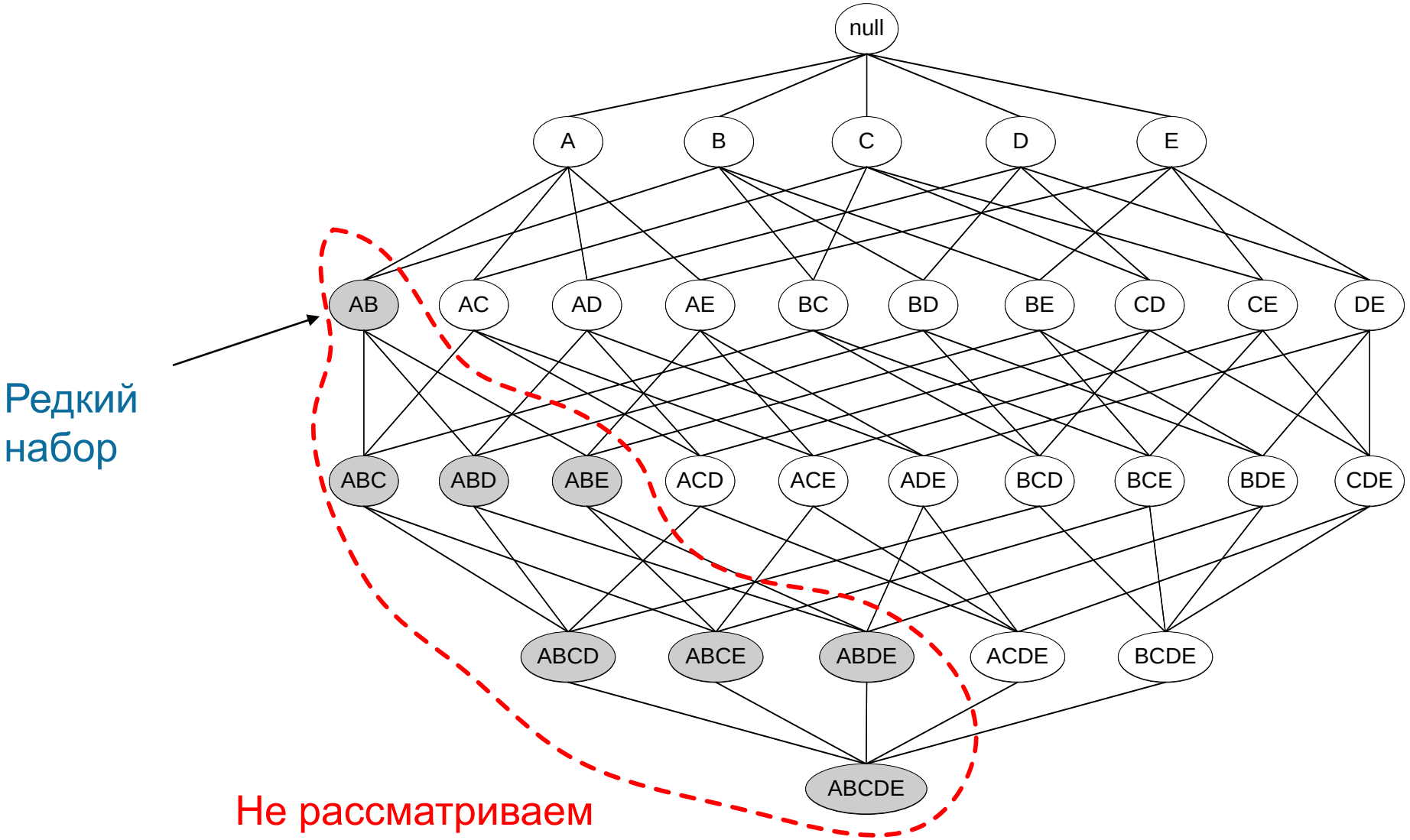
Rule	Support	Confidence
X=>Y	25%	50%
X=>Z	37.50%	75%

Itemset	Support	Interest
X,Y	25%	2
X,Z	37.50%	0.9
Y,Z	12.50%	0.57

АЛГОРИТМ APRIORI

- Основной принцип (анти-монотонность):
  - Любое подмножество часто встречаемого набора является часто встречаемым набором
- Формально:
  - Поддержка любого набора элементов не может превышать минимальной поддержки всех его подмножеств
  - Необходимое условие частой встречаемости k-элементного набора – частая встречаемость всех его (k-1)-элементных подмножеств
- Этапы алгоритма:
  - Генерация множества часто встречаемых наборов (supp >= minsupp): метод «ветвей и границ» - направленный перебор от простых (коротких) наборов к сложным (длинным) с отсечением
  - Генерация правил по найденным наборам (conf >= minconf)

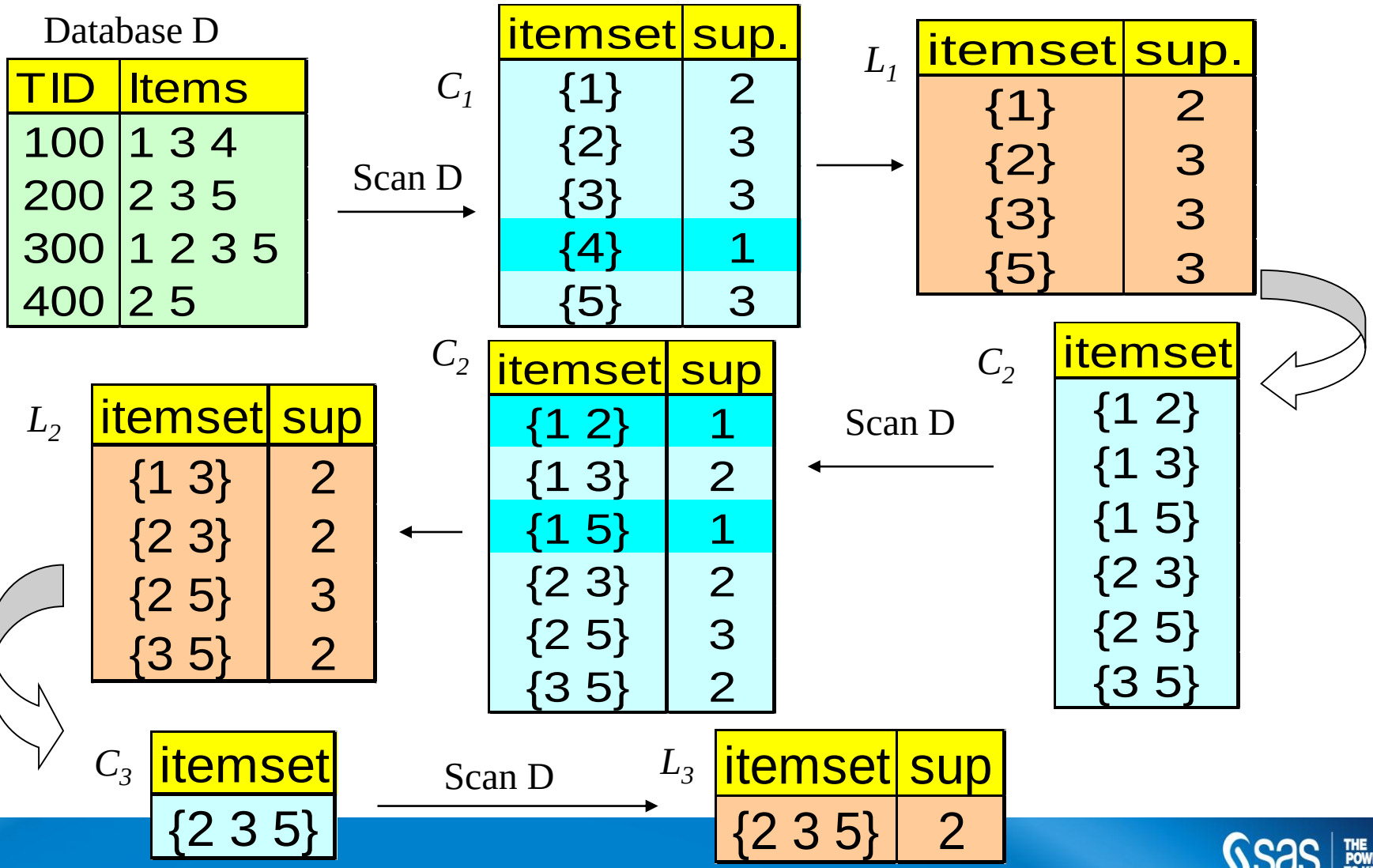
ИДЕЯ МЕТОДА ВЕТВЕЙ И ГРАНИЦ ДЛЯ APRIORI



ПРИМЕР ГЕНЕРАЦИИ КАНДИДАТОВ

- L_3={*abc, abd, acd, ace, bcd*}
- Объединение: L_3*L_3
  - *abcd* = *abc* + *abd*
  - *acde* = *acd* + *ace*
- Удаление:
  - *acde* удален, т.к. *ade* не в L_3
- C_4={*abcd*}

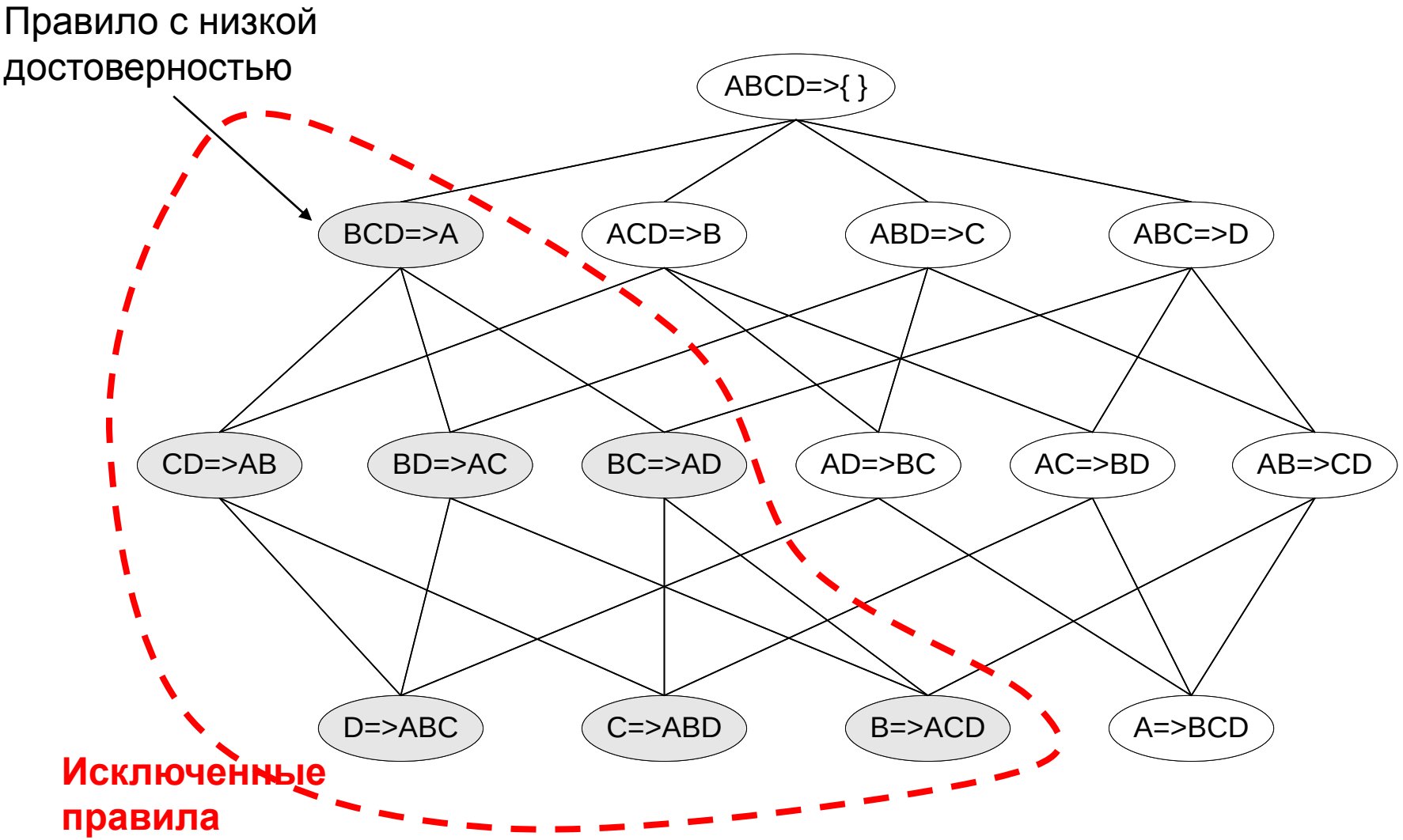
ПРИМЕР ГЕНЕРАЦИИ ЧАСТЫХ НАБОРОВ



ГЕНЕРАЦИЯ ПРАВИЛ

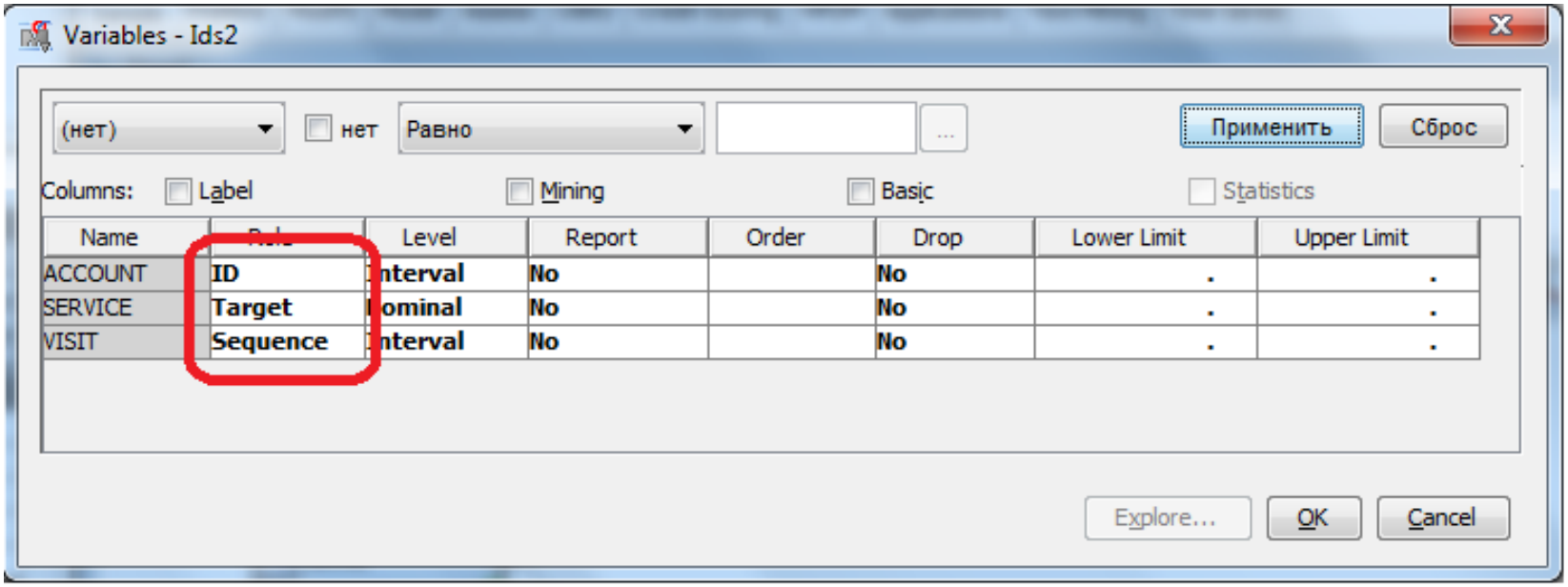
- Критерий:
  - conf(X=>Y) = P(Y|X) = support({X,Y}) / support(X)
  - conf(X=>Y)>=minconf
  - все support известны с 1-го этапа
- Принцип:
  - Если правило {A} => {B, C} интересно, то и {A, B} => {C} интересно
- Доказательство:
  - conf({A}=>{B, C}) = supp({A, B, C}) / support({A})>=minconf
  - conf({A, B}=>{C}) = supp({A, B, C}) / support({A, B})
  - support({A, B}) <= supp({A})
  - conf({A, B}=>{ C})>=minconf
- Алгоритм:
  - Для каждого часто встречаемого набора проверять правила на интересность, начиная со случая, когда в правой части правила находится один атрибут и постепенно добавлять/убавлять атрибуты в/из правую/левой часть(и).

МЕТОД ВЕТВЕЙ И ГРАНИЦ ДЛЯ ГЕНЕРАЦИИ ПРАВИЛ



НАСТРОЙКА ИСТОЧНИКА ДАННЫХ ДЛЯ АССОЦИАТИВНОГО АНАЛИЗА

- Роли переменных:
  - ID – идентификатор транзакции (может быть несколько строк с одинаковым ID(
  - Target – переменная, содержащая имя Item
  - Sequence (или Time) – номер события в цепочке (опционально)
- Роль источник: Transactions



АССОЦИАТИВНЫЙ АНАЛИЗ ДЛЯ ПОИСКА ПОСЛЕДОВАТЕЛЬНОСТЕЙ СОБЫТИЙ

Два этапа:

1. Обычный поиск частых эпизодов без учета последовательностей
2. Формирование многоместных правил вида **A=>B=>C=>D** с учетом последовательности

Дополнительные настройки:

- Роли time или sequence
- Ограничение на размер временного окна при поиске
- Меньшая поддержка чем у частых эпизодов

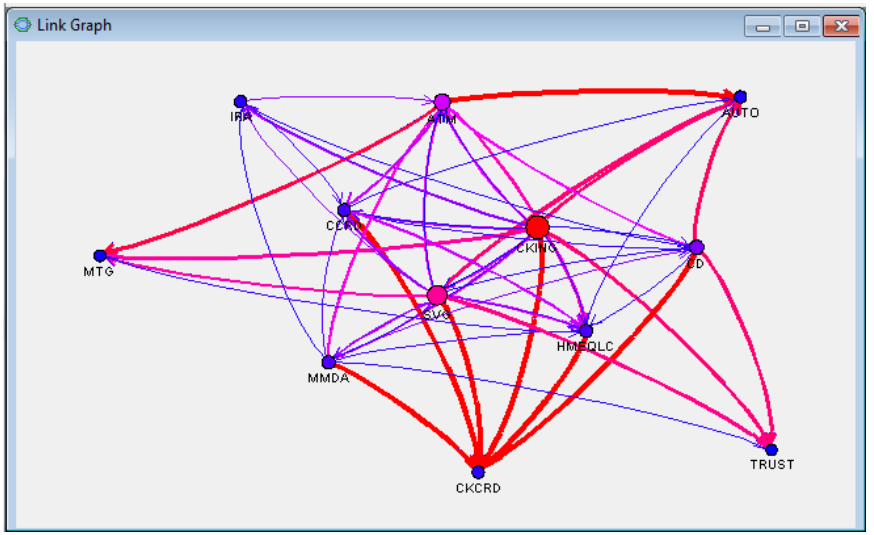
ПРЕДСТАВЛЕНИЕ РЕЗУЛЬТТАТОВ АССОЦИАТИВНОГО АНАЛИЗА

Rules Table

NITEMS	COUNT	SUPPORT	CONF	PSEUDOLIFT	RULE
2	4329	54.17	63.15	1.02	CKING ==> SVG
2	2892	36.19	42.19	1.10	CKING ==> ATM
2	2053	25.69	41.53	1.08	SVG ==> ATM
3	1986	24.85	45.88	1.19	CKING ==> SVG ==> ATM
2	1709	21.39	55.61	1.45	ATM ==> ATM
2	1677	20.99	24.46	1.00	CKING ==> CD
3	1546	19.35	53.46	1.39	CKING ==> ATM ==> ATM
2	1316	16.47	19.20	1.17	CKING ==> HMEQLC
2	1256	15.72	25.40	1.04	SVG ==> CD
2	1245	15.58	18.16	1.04	CKING ==> MMDA
2	1187	14.85	17.32	1.12	CKING ==> CCRD
3	1139	14.25	26.21	1.07	CKING ==> SVG ==> CD

- Таблица правил

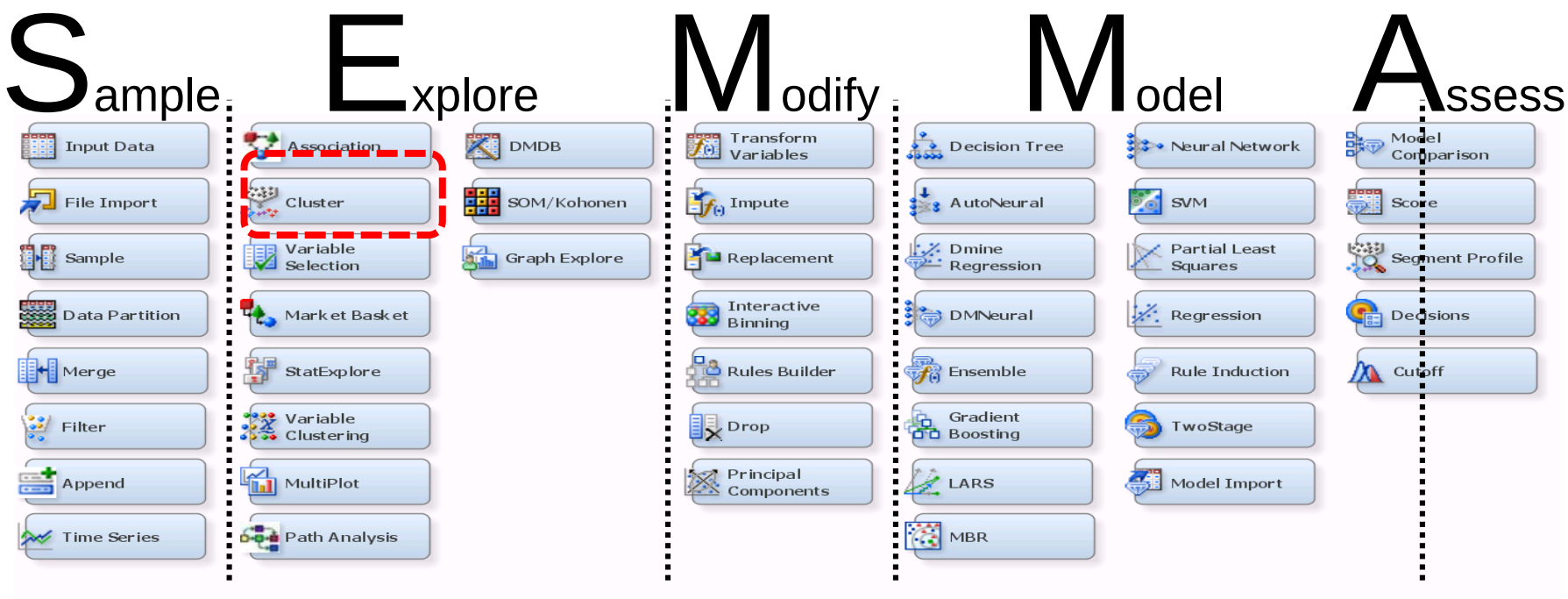
- Граф связей:



SAS ENTERPRISE MINER

КЛАСТЕРИЗАЦИЯ

КОНЦЕПЦИЯ SEMMA



ЧТО ЕСТЬ КЛАСТЕР?

- Кластер: группа «похожих» объектов
  - «похожих» между собой в группе (внутриклассовое расстояние)
  - «не похожих» на объекты других групп
  - Определение неформальное, формализация зависит от метода
- Кластерный анализ
  - Разбиение множество объектов на группы (кластеры)
- Тип моделей:
  - «описательный» (descriptive) Data mining => одна из задач - наглядное представление кластеров
  - «прогнозный» (predictive) Data mining => разбиение на кластеры, а затем «классификация» новых объектов
- Тип обучения:
  - всегда «без учителя» (unsupervised) => тренировочный набор не размечен

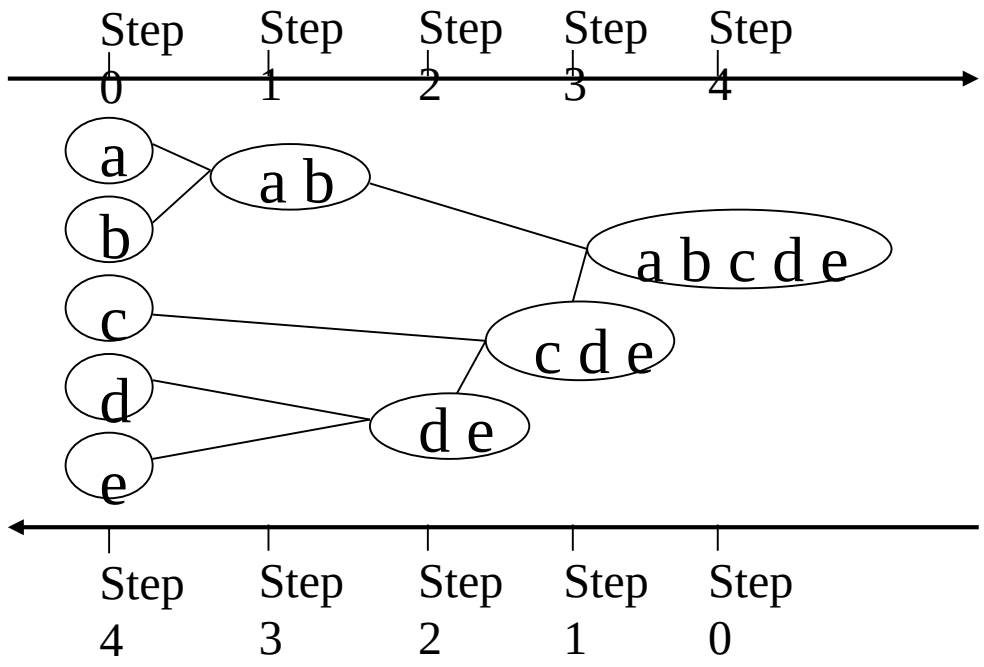
ПРИМЕНЕНИЕ МЕТОДОВ КЛАСТЕРИЗАЦИИ В ЗАДАЧАХ АНАЛИЗА ДАННЫХ

- Кластеризация ради кластеризации:
  - Выявление и описание групп (человек не способен «осознать» более 10 объектов в одной задаче, как обработать выборку с миллионами?)
  - «Сжатие» информации (особенно в обработке мультимедиа)
  - Построение различных поисковых индексов (сравниваем не со всеми, а начинаем с прототипов кластеров)
- Мощнейшее средство предобработки данных:
  - Дискретизация
  - Уменьшение размера выборки (от больших объемов к «реальным»)
  - Обработка пропущенных значений (инициализируем и итерационно «улучшаем» пропуски)
  - Поиск исключений и артефактов (что не в кластере, то под «подозрением»)

КАЧЕСТВО КЛАСТЕРИЗАЦИИ

- Хороший метод кластеризации находит кластеры
  - с высоким «внутриклассовым» сходством объектов
  - и низким «межклассовым» сходством объектов
- Оценка качества кластеризации (нет понятия «точность»)
  - необходима, так как влияет на выбор параметров метода
  - определяется либо экспертом – субъективная величина
  - либо «перекрестной» проверкой целевой функции кластеризации
- Качество кластеризации зависит:
  - от метода кластеризации
  - от меры сходства (или расстояния)

ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ

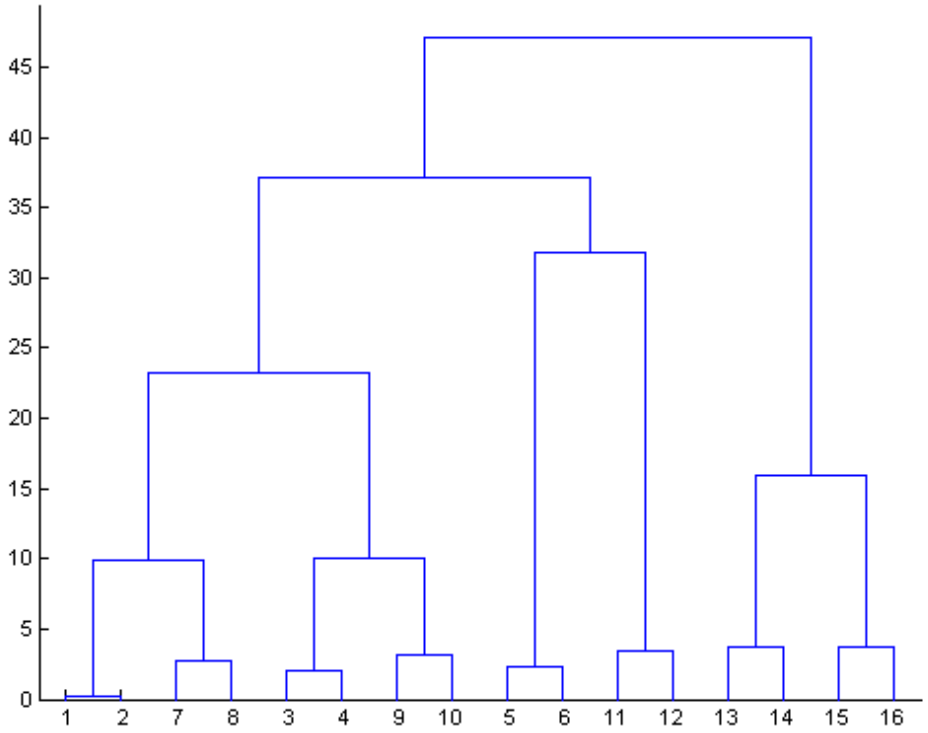


восходящая
agglomerative

нисходящая
divisive

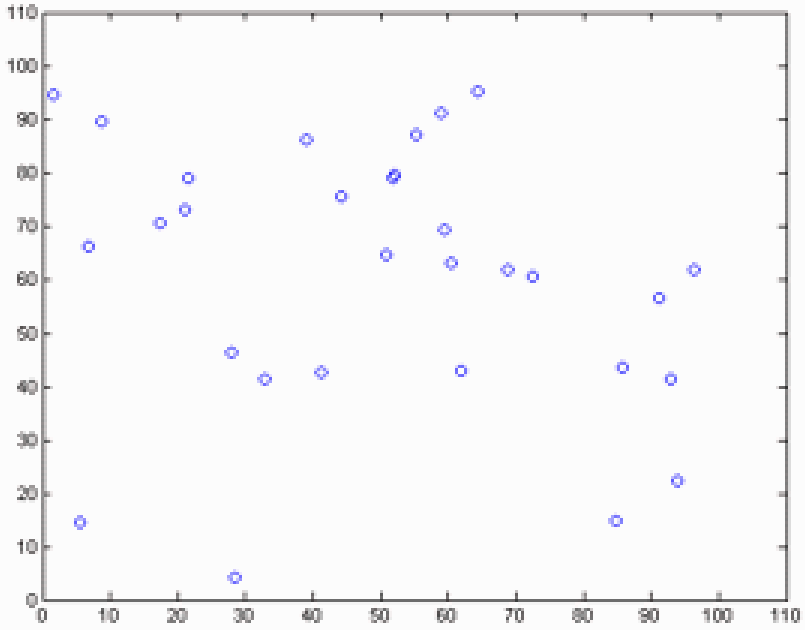
- Используется только матрица сходства (различия) и не требуется дополнительных параметров (например, числа кластеров)
- «Пошаговое» объединение ближайших кластеров (восходящая) или разбиение наиболее удаленных (нисходящая)

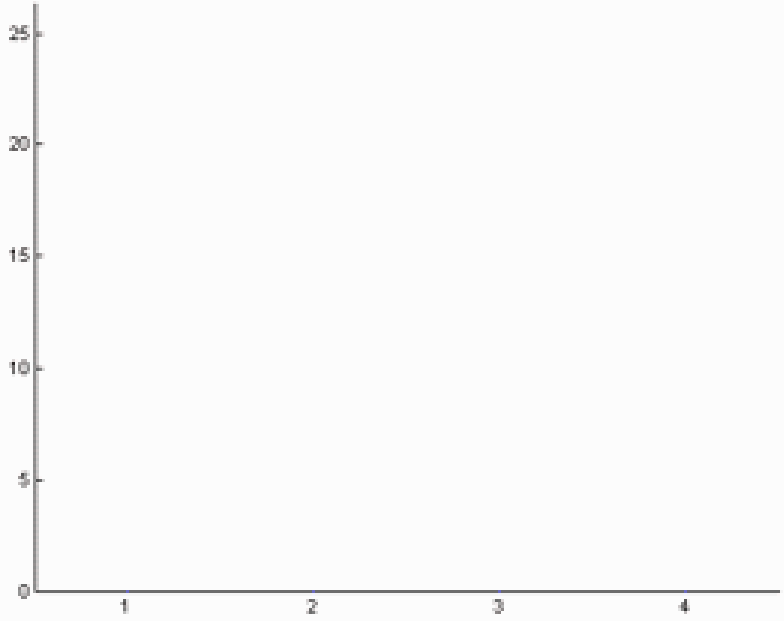
ПРЕДСТАВЛЕНИЕ ИЕРАРХИЧЕСКИХ КЛАСТЕРОВ - ДЕНДРОГРАММА



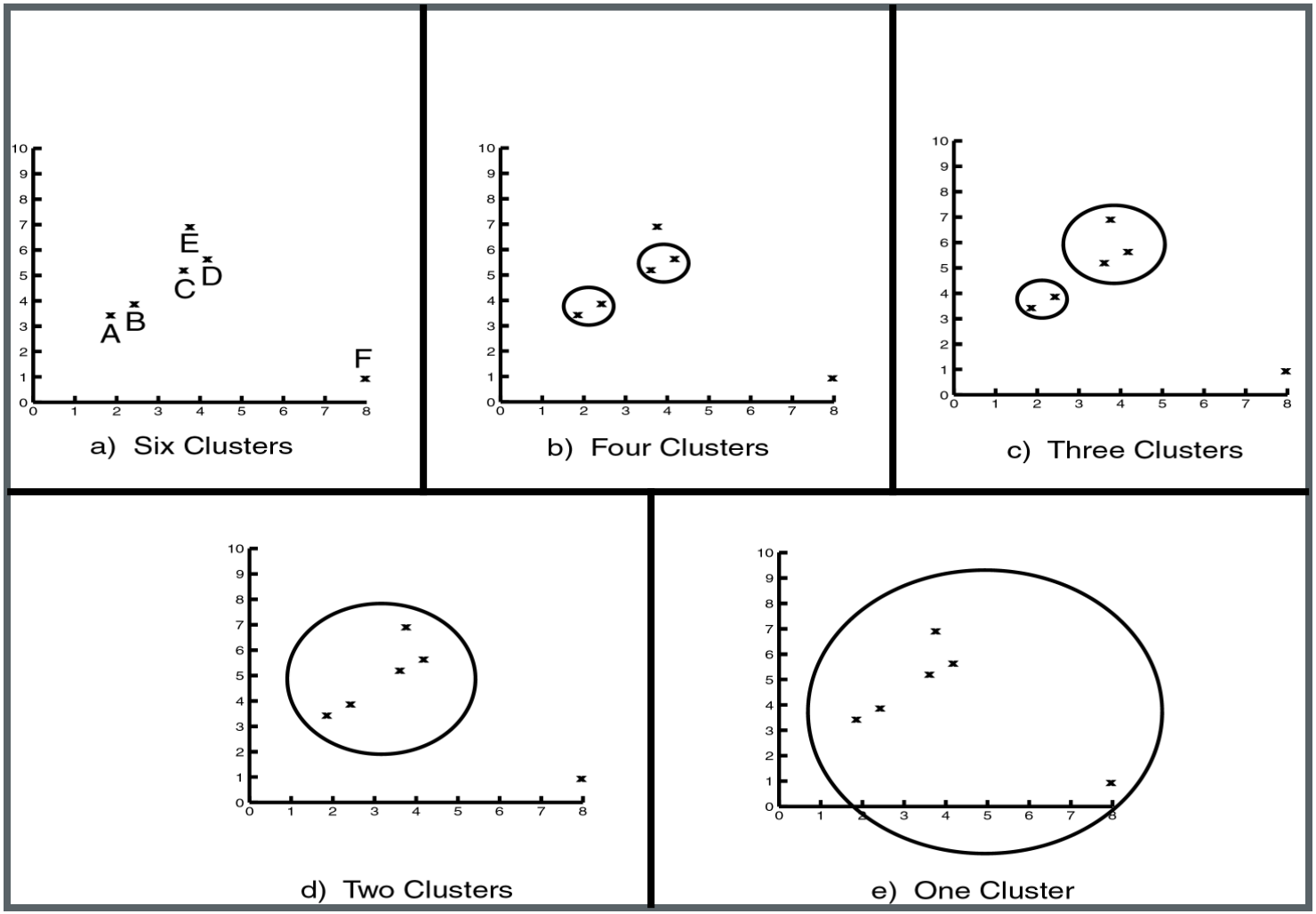
- бинарное дерево, описывающее все шаги разбиения
- Корень – общий кластер, листья - элементы
- «Высота» ветвей (до пересечения) – порог расстояния «склейки» («разделения»)
- Результат кластеризации – «срез» дендрограммы

ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ - DEMO



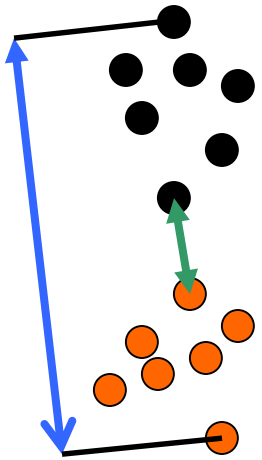


УРОВНИ КЛАСТЕРИЗАЦИИ



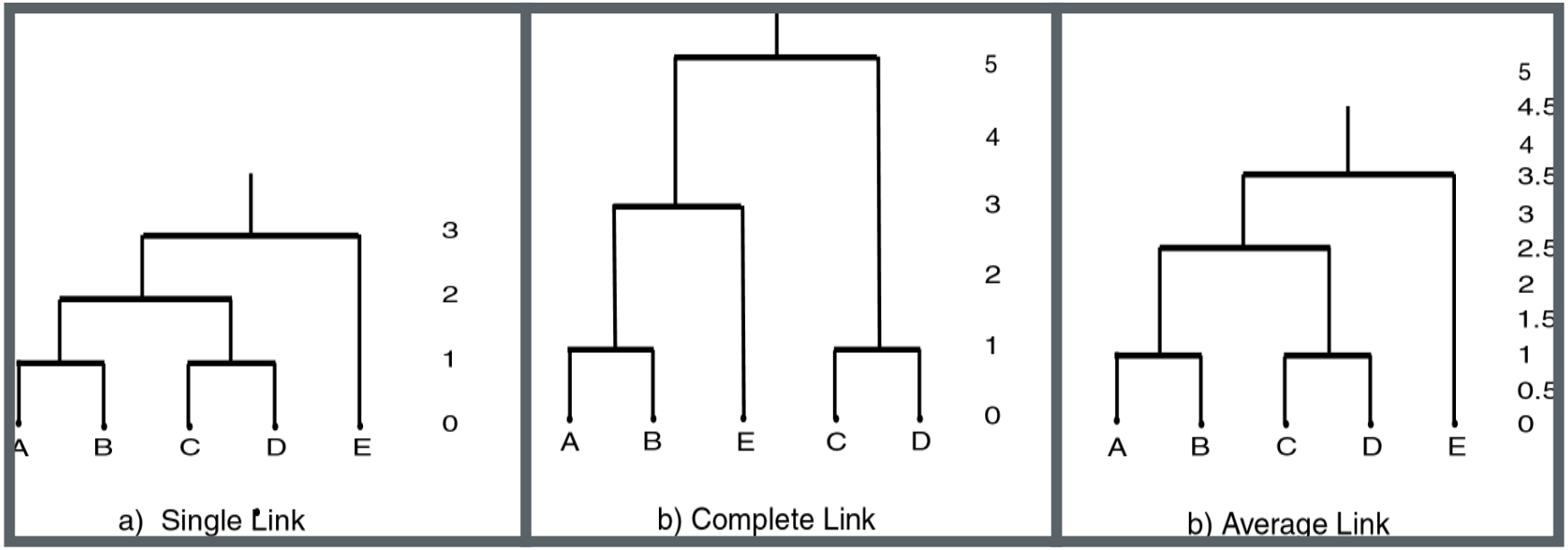
ОЦЕНКА БЛИЗОСТИ КЛАСТЕРОВ

- Расчет расстояния на основе попарных расстояний между элементами различных кластеров:
  - **Полное связывание**: наибольшее попарное расстояние. Дает компактные сферические кластеры.
  - **Среднее связывание**: усредненное попарное расстояние. Редко используется.
  - **Единственное связывание**: наименьшее попарное расстояние. Дает «растянутые» кластеры сложной формы.
  - **Центроидное связывание***:* расстояние между центрами (мат. ожидание) кластеров.
  - Другие методы (например **метод Ward'a** – минимизирует внутрикластерные дисперсии или другую целевую функцию)



Пример

	A	B	C	D	E
A	0	1	2	2	3
B	1	0	2	4	3
C	2	2	0	1	5
D	2	4	1	0	3
E	3	3	5	3	0



КЛАСТЕРИЗАЦИЯ НА ОСНОВЕ СТРОГОЙ ГРУППИРОВКИ (PARTITIONING):

- Основная задача:
  - Найти такое разбиение *C* исходного множества *X* из *N* объектов на *K* непересекающихся подмножеств C_k, покрывающих *X*, чтобы внутриклассовое расстояние было минимальным:

$$\min_{C_i \cap C_j = \varnothing, \bigcup C_i = X} \Sigma_{i=1}^{K} \Sigma_{x \in C_i} \Sigma_{x' \in C_i} d(x, x')$$

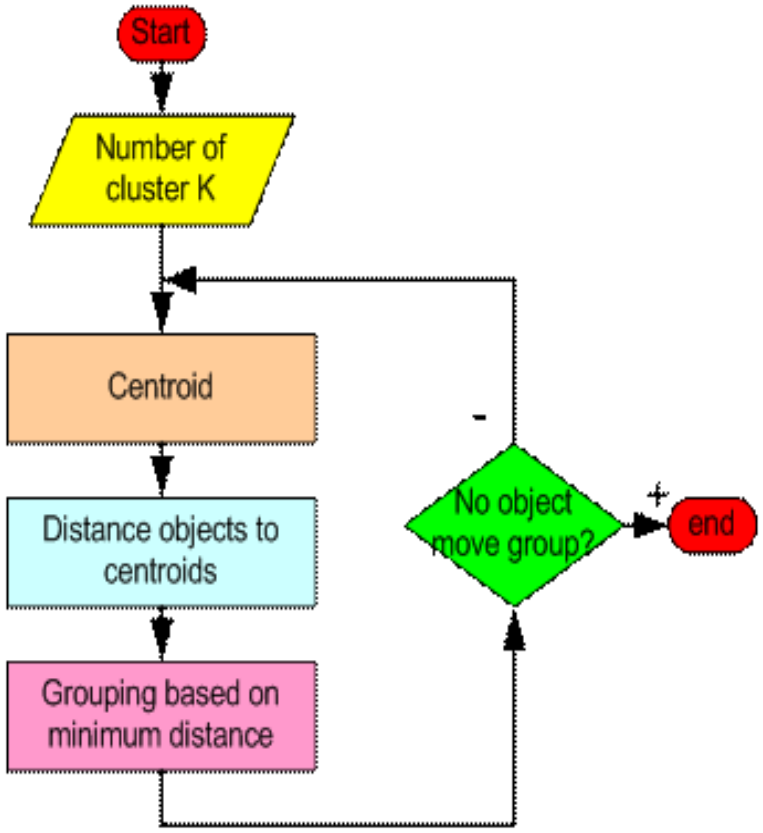
- Точное решение – перебор с отсечением
  - метод «ветвей и границ», но число комбинаций неприемлемо даже для 100 объектов:

$$S(N,K) = \frac{1}{K!} \sum_{i=1}^{K} (-1)^{K-i} \binom{K}{i} K^N$$

- Эвристические методы:
  - K-means (прототип кластера – мат. ожидание *m*), K-medoids (прототип кластера – средний элемент)
  - ищется локальный минимум

$$\min_{C_i \cap C_j = \varnothing, \bigcup C_i = X} \Sigma_{i=1}^{K} \Sigma_{x \in C_i} d(m_i, x)$$

МЕТОД K-MEANS В ENTERPRISE MINER



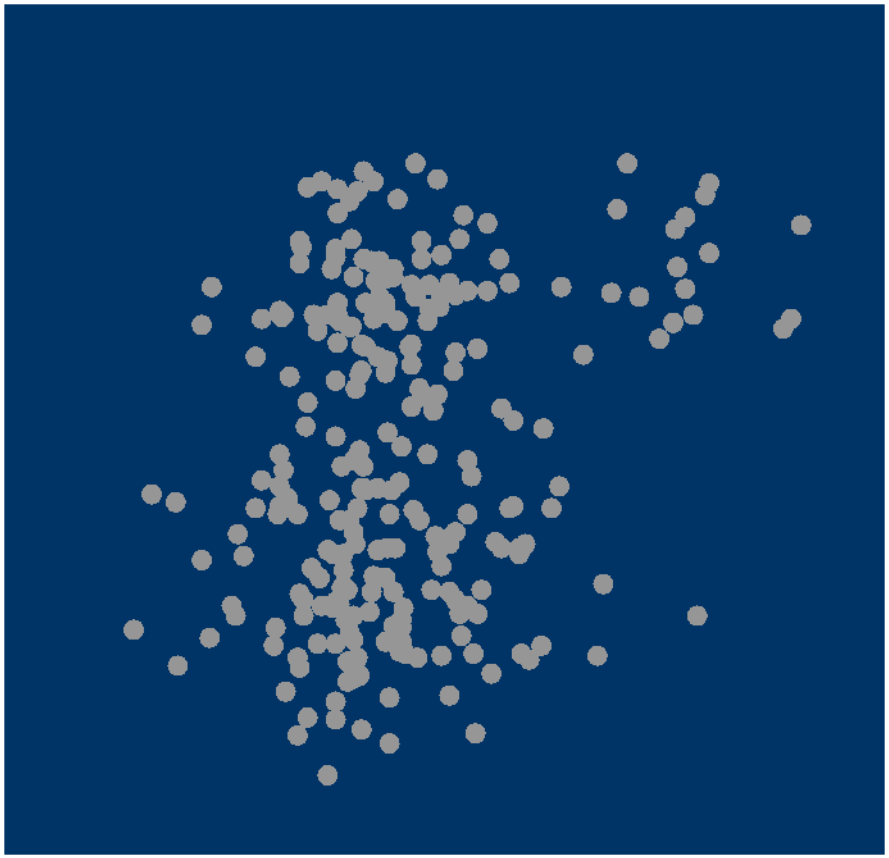
- Шаг 0. Инициализация:
  - произвольное разбиение на заданное число кластеров K (где значение K выбирается по CCC на основе иерархической кластеризации) по
- Шаг 1. Поиск центров: $m_i = \sum_{x \in C_i} x \Big/ \|C_i\|$
  - Для всех K кластеров
- Шаг 2. Расчет расстояний до центров:
  - Для всех N объектов и K кластеров $d(m_i, x) = \sum_{x \in C_i} x \Big/ \|C_i\|$
- Шаг 3. Выбор ближайшего кластера:

$$x \in C_i \Leftrightarrow i = \min_j d(m_j, x)$$

- Если были перестановки, то Шаг 1.

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

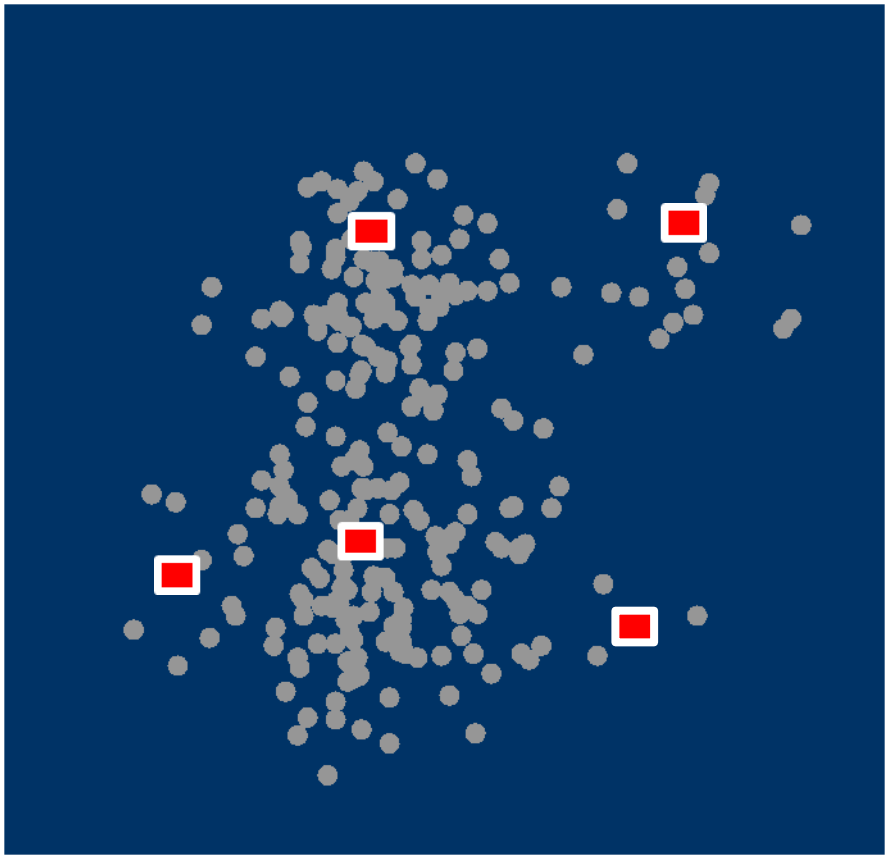
Training Data



1. **Выбор переменных.**
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

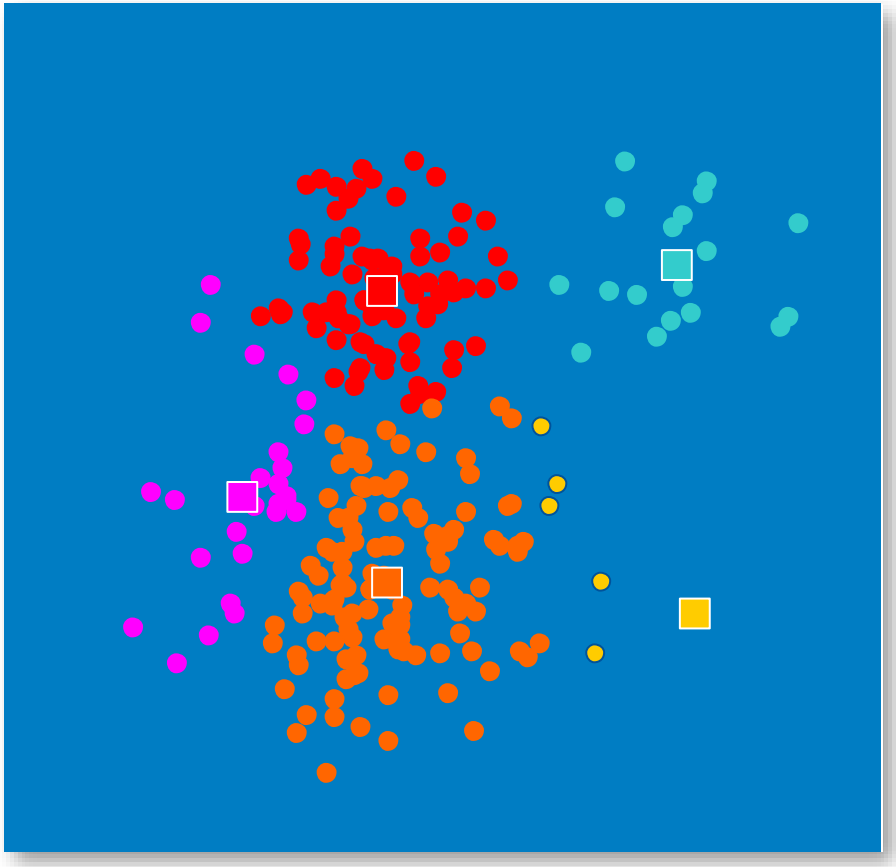
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

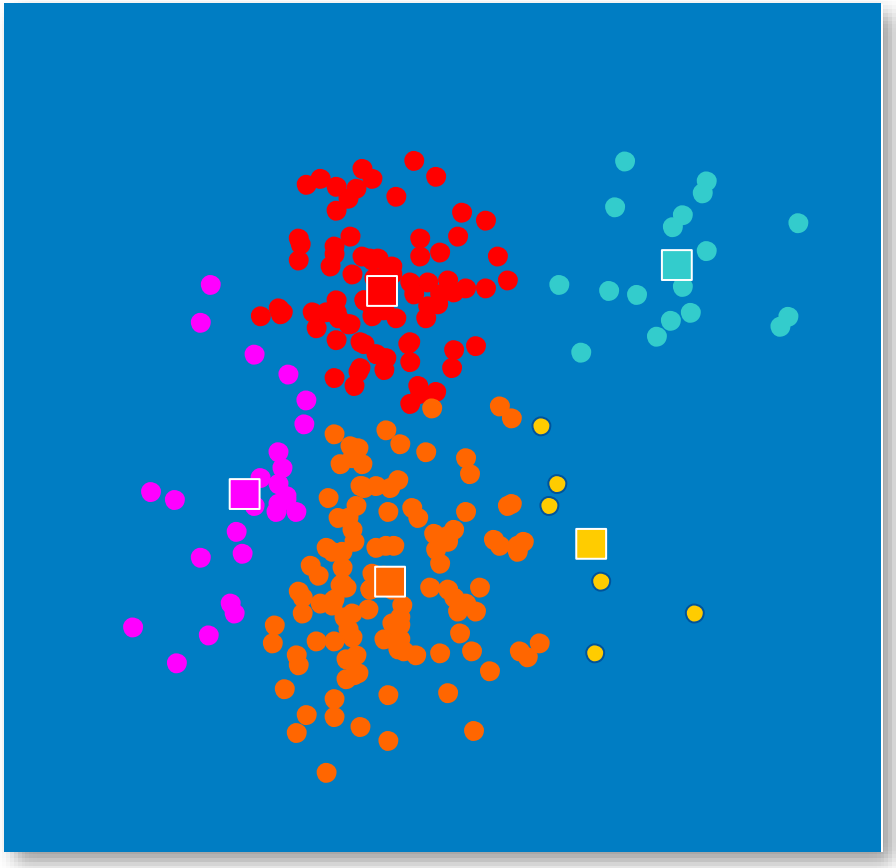
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. **Выбор ближайшего центра для каждого примера.**
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

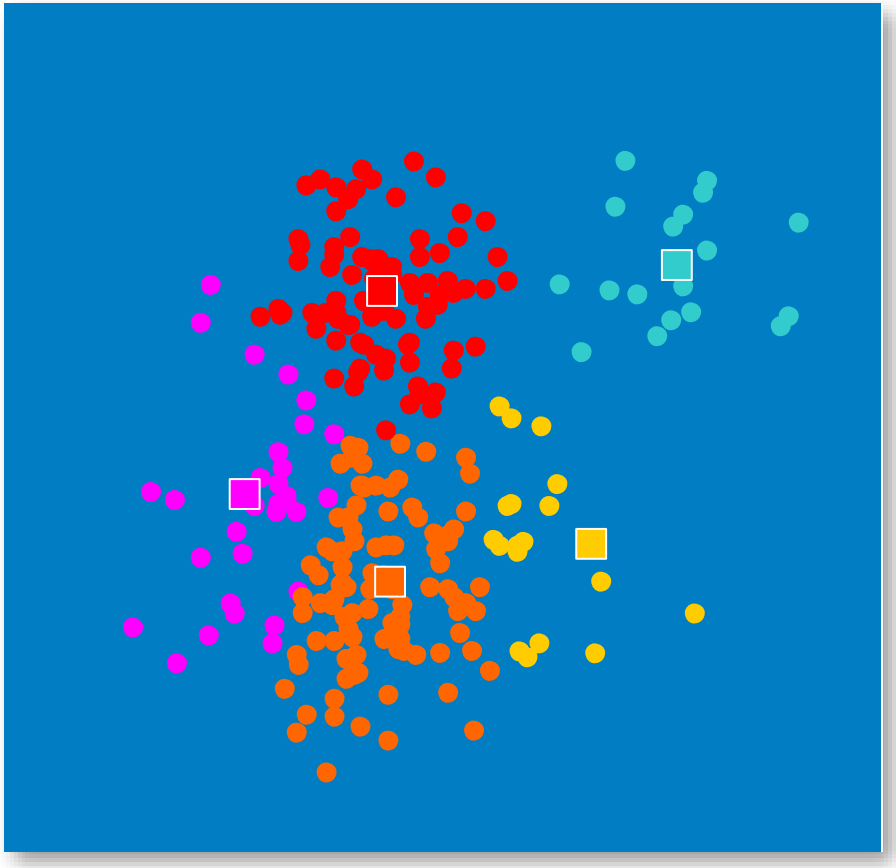
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. **Пересчет центров.**
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

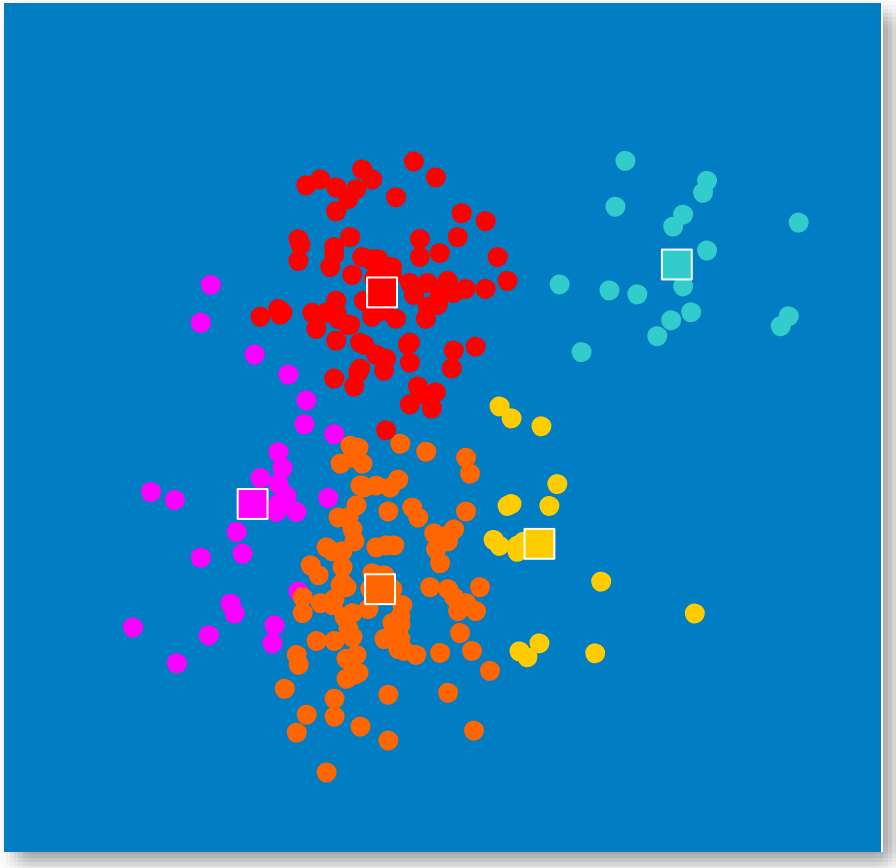
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. **Выбор ближайшего центра для каждого примера.**
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

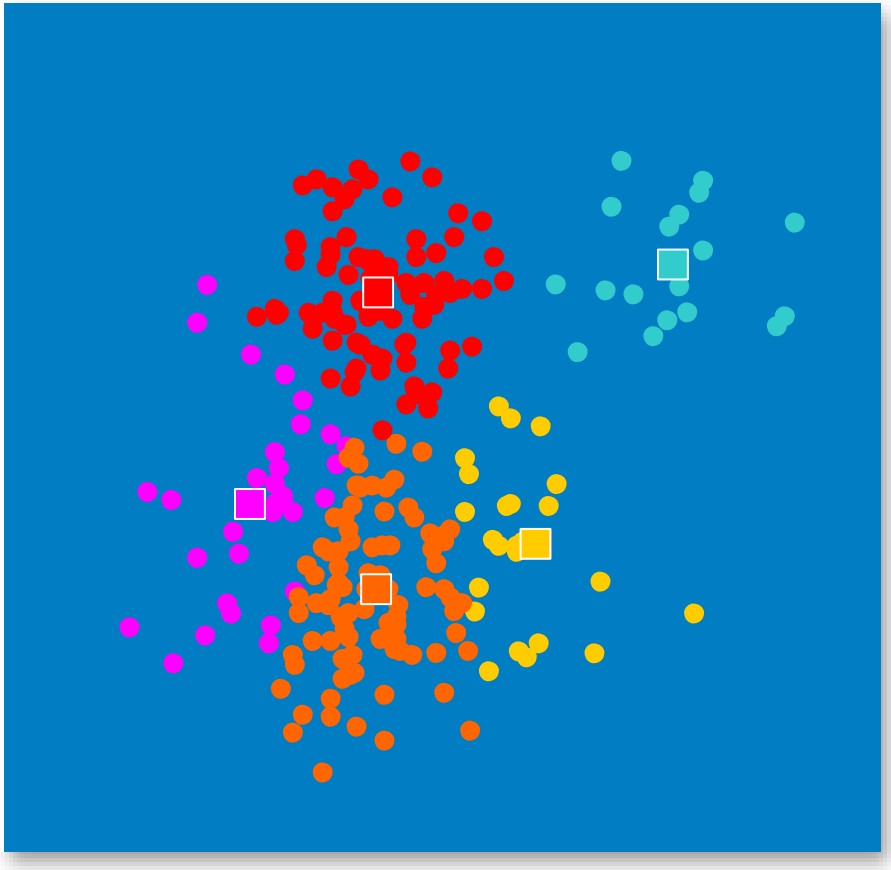
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. **Пересчет центров.**
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

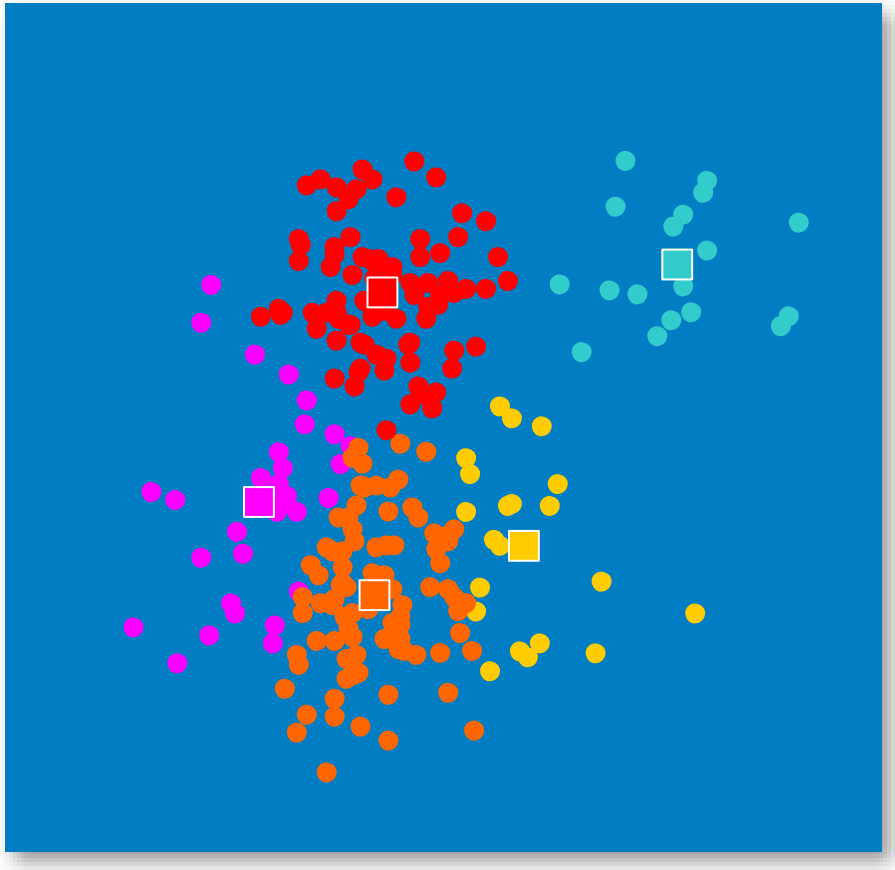
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. **Выбор ближайшего центра для каждого примера.**
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

Training Data



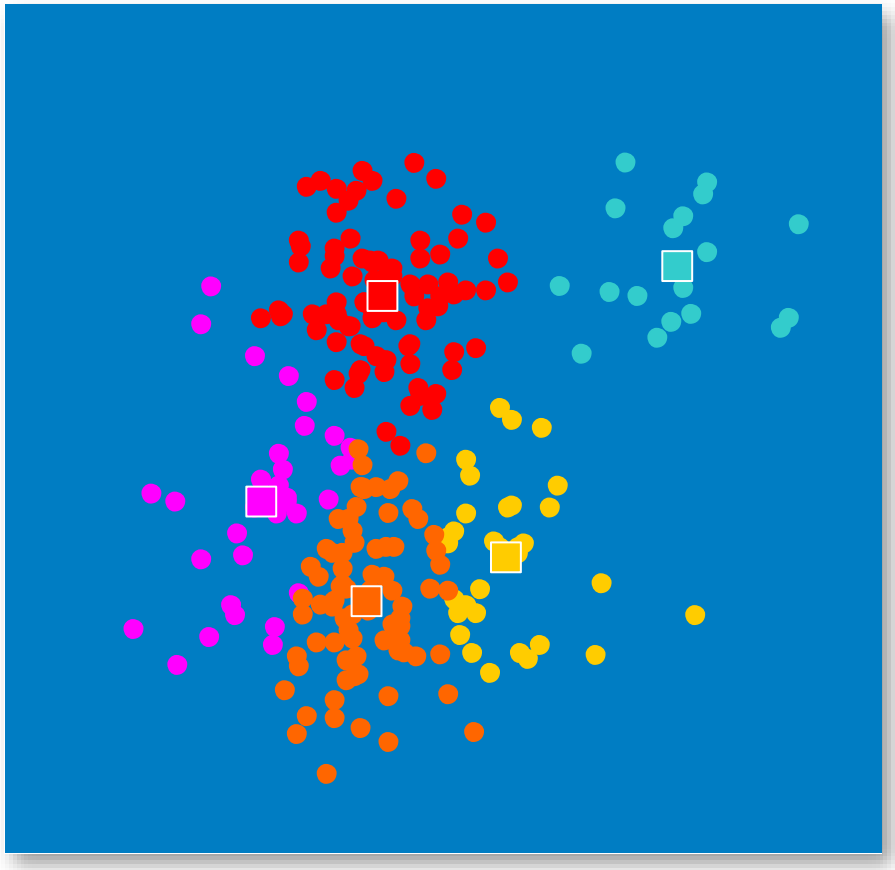
1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. **Пересчет центров.**
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. **Выбор ближайшего центра для каждого примера.**
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

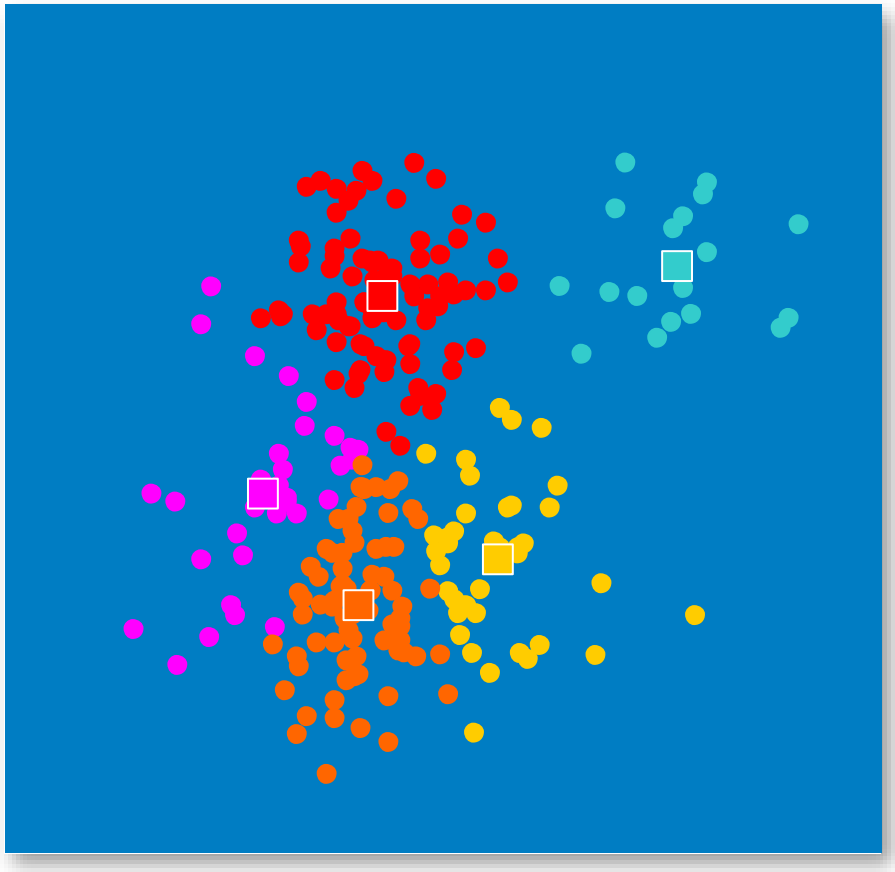
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. **Пересчет центров.**
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

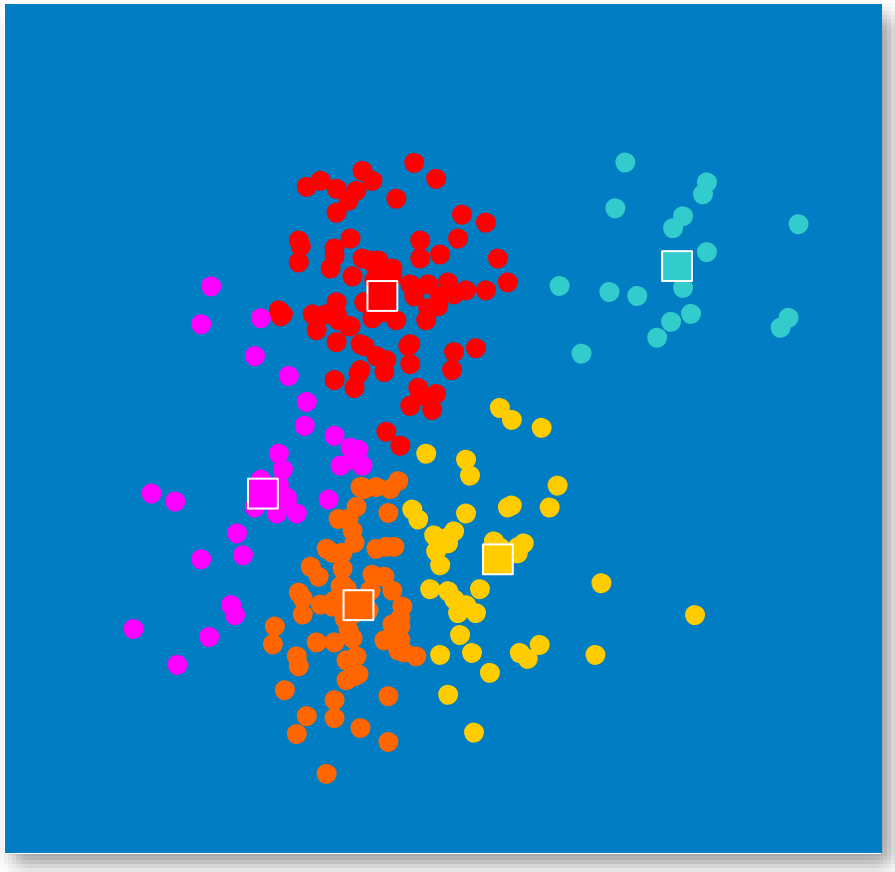
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. **Выбор ближайшего центра для каждого примера.**
4. Пересчет центров.
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

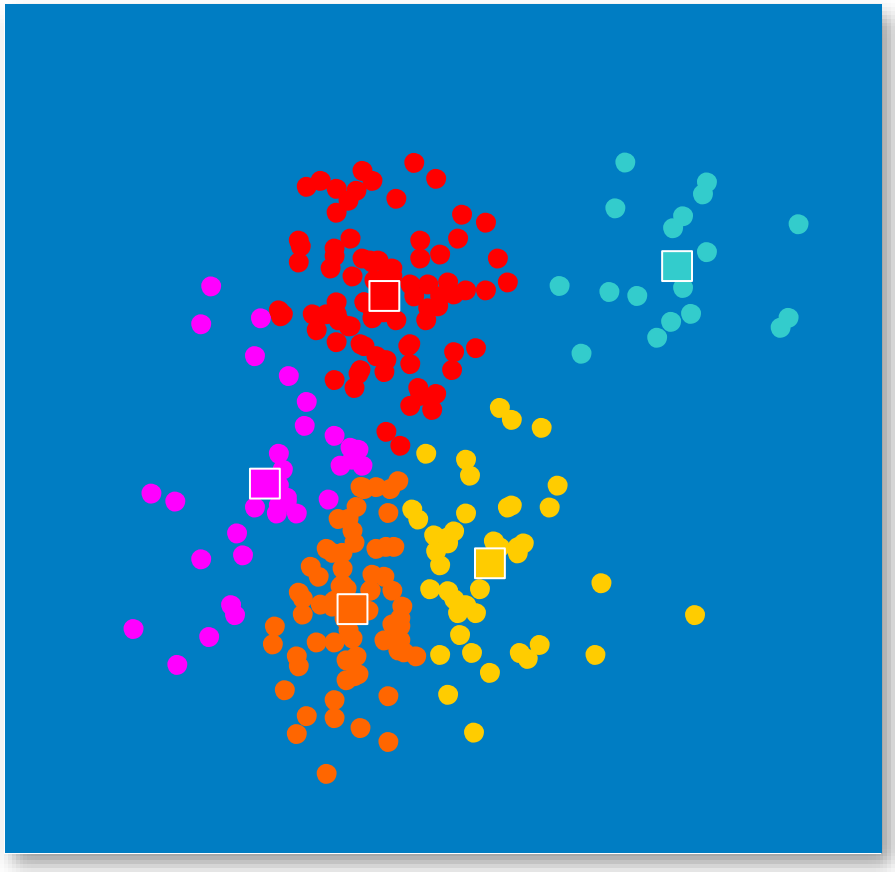
Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. **Пересчет центров.**
5. Переход на шаг 3 пока процесс не сошелся

АЛГОРИТМ КЛАСТЕРИЗАЦИИ K-MEANS

Training Data



1. Выбор переменных.
2. Выбор *k* центров кластеров.
3. Выбор ближайшего центра для каждого примера.
4. **Пересчет центров.**
5. Переход на шаг 3 пока процесс не сошелся

ОПРЕДЕЛЕНИЕ ЧИСЛА КЛАСТЕРОВ

- SAS Cubic Clustering Criterion (CCC) (Sarle, 1983)
  - Основная идея: сравнение R^2 (для отображения матрицы данных с помощью индикаторной матрицы в прототипы кластеров) для заданной модели кластеризации с $E(R^2)$ для равномерно распределенного множества прототипов кластеров (как наихудший возможный вариант):

$$CCC = \ln\left[\frac{1-E(R^2)}{1-R_{clust}{}^2}\right] \times K$$

ОПРЕДЕЛЕНИЕ ЧИСЛА КЛАСТЕРОВ

1. Случайно выбираются центры для большого (50 по умолчанию) числа кластеров
2. Все наблюдения объединяются в эти случайные кластеры
3. Решается задач восходящей иерархической кластеризации, на каждом шаге считается CCC
4. По определенным правилам выбирается оптимальное число кластеров:
   - Первый локальный пик …