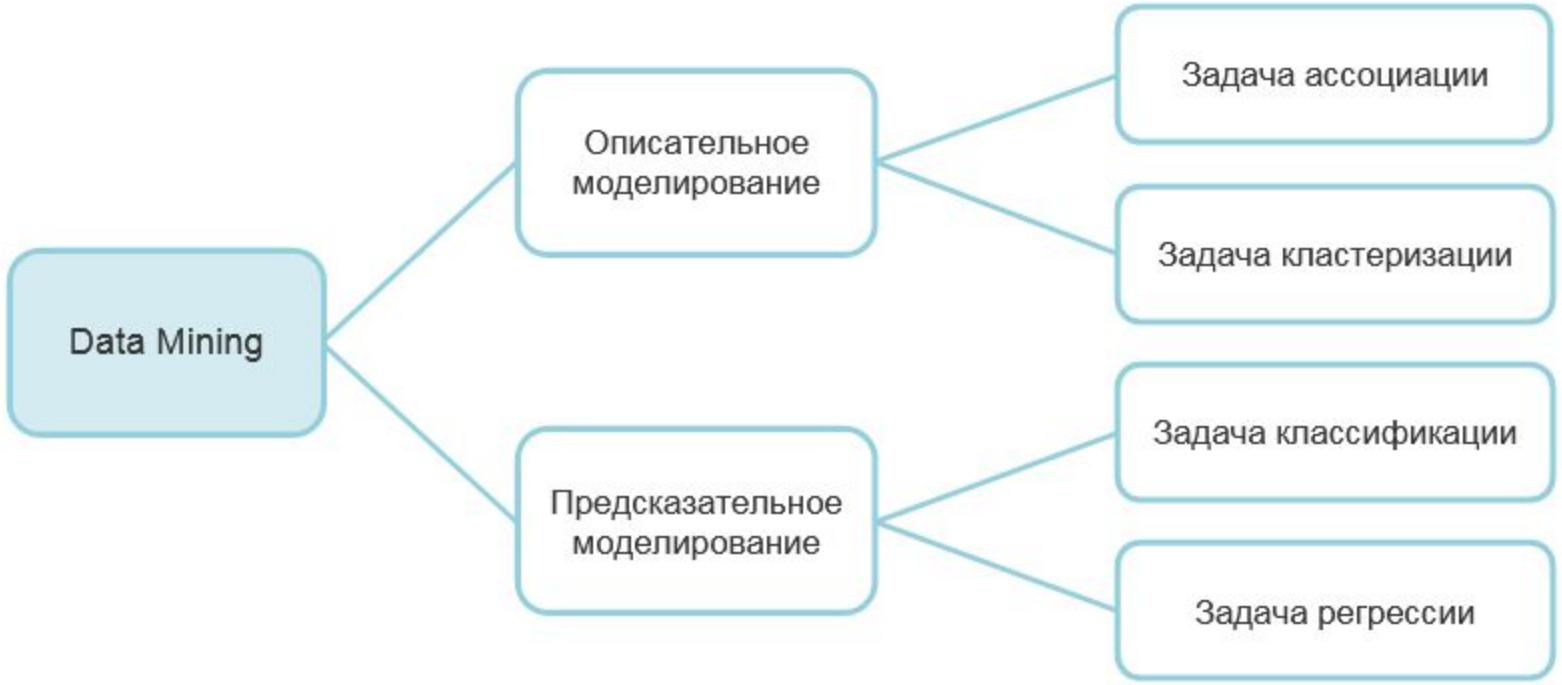
Data Mining

Data Mining — обнаружение в «сырых» данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности.

Модели и задачи Data Mining

- Data Mining – совокупность большого числа различных методов обнаружения знаний.
- В современной бизнес-аналитике принято выделять в Data Mining описательные (дескриптивные) и предсказательные модели.



Модели и задачи Data Mining

- Описательная аналитика позволяет выполнить описание множества объектов в виде кластеров, правил, шаблонов поведения, групп. Отве
  - *Какова структура клиентской базы?*
  - *Какой профиль идеального клиента?*
  - *Какие есть взаимосвязи между характеристиками клиентов?*
  - *Какие события происходят одновременно?*

- Предсказательная аналитика отвечает на следующие вопросы:
  - *Откликнется ли клиент на данную маркетинговую кампанию?*
  - *Какой размер прибыли будет в следующем месяце?*
  - *Какие из потенциальных клиентов вероятно совершат приобретение услуги в следующем месяце?*
  - *Какой прогнозируемый спрос на товар на следующий период планирования?*

и так далее.

Data Mining — это не один метод, а совокупность большого числа различных методов обнаружения знаний. Существует несколько условных классификаций задач Data Mining:

1 **Классификация** – это установление зависимости дискретной выходной переменной от входных переменных.

2 **Регрессия** – это установление зависимости непрерывной выходной переменной от входных переменных.

3 **Кластеризация** – это группировка объектов (наблюдений, событий) на основе данных, описывающих свойства объектов. Объекты внутри кластера должны быть похожими друг на друга и отличаться от других, которые вошли в другие кластеры.

4 **Ассоциация** – выявление закономерностей между связанными событиями.

Ассоциация

Ассоциация – выявление закономерностей между связанными событиями. Примером такой закономерности служит правило, указывающее, что из события X следует событие Y. Такие правила называются *ассоциативными*.

На примерах применения ассоциативных правил могут быть следующие задачи:

- выявление наборов товаров, которые в супермаркетах часто покупаются вместе или никогда не покупаются вместе;
- определение доли клиентов, положительно относящихся к нововведениям в их обслуживании;
- определение профиля посетителей веб-ресурса;
- определение доли случаев, в которых новое лекарство показывает опасный побочный эффект…

Базовые понятия теории ассоциативных правил:

- *Предметный набор* — это непустое множество объектов, появившихся в одной транзакции.
- *Транзакция* — некоторое множество событий, происходящих совместно.

Типичная транзакция —приобретение клиентом товара в супермаркете.

Вопрос: является ли покупка одного товара следствием или причиной покупки другого товара, то есть связаны ли данные события? Эту связь и устанавливают *ассоциативные правила*.

Формальная запись ассоциативных правил

$$I = \{i_1, i_2, \ldots, i_n\}$$

$$D = \{t_1, t_2, \ldots, t_m\}$$

$$X \Rightarrow Y, \text{ где } X, Y \subseteq I.$$

ID транзакции	молоко	хлеб	масло
1	1	1	0
2	0	0	1
3	0	0	0
4	1	1	1
5	0	1	0

Методы поиска ассоциативных правил

- Алгоритм AIS. Первый алгоритм, предложенный Agrawal, Imielinski and Swami сотрудниками IBM Almaden в 1993 году. В алгоритме AIS кандидаты множества наборов генерируются и подсчитываются "на лету", во время сканирования базы данных.
- Алгоритм SETM. Создание этого алгоритма было мотивировано желанием использовать язык SQL для вычисления часто встречающихся наборов товаров. Формирует кандидатов "на лету", основываясь на преобразованиях базы данных.
- Алгоритм Apriori. Работа данного алгоритма состоит из нескольких этапов, каждый из этапов состоит из следующих шагов:
  - формирование кандидатов;
  - подсчет кандидатов.

Алгоритм Apriori

- Выявление частых наборов объектов — операция, требующая большого количества вычислений, а следовательно, и времени. Алгоритм Apriori описан в 1994 г. Срикантом Рамакришнан (Ramakrishnan Srikant) и Ракешом Агравалом (Rakesh Agrawal). Алгоритм использует одно из свойств поддержки, гласящее: поддержка любого набора объектов не может превышать минимальной поддержки любого из его подмножеств:

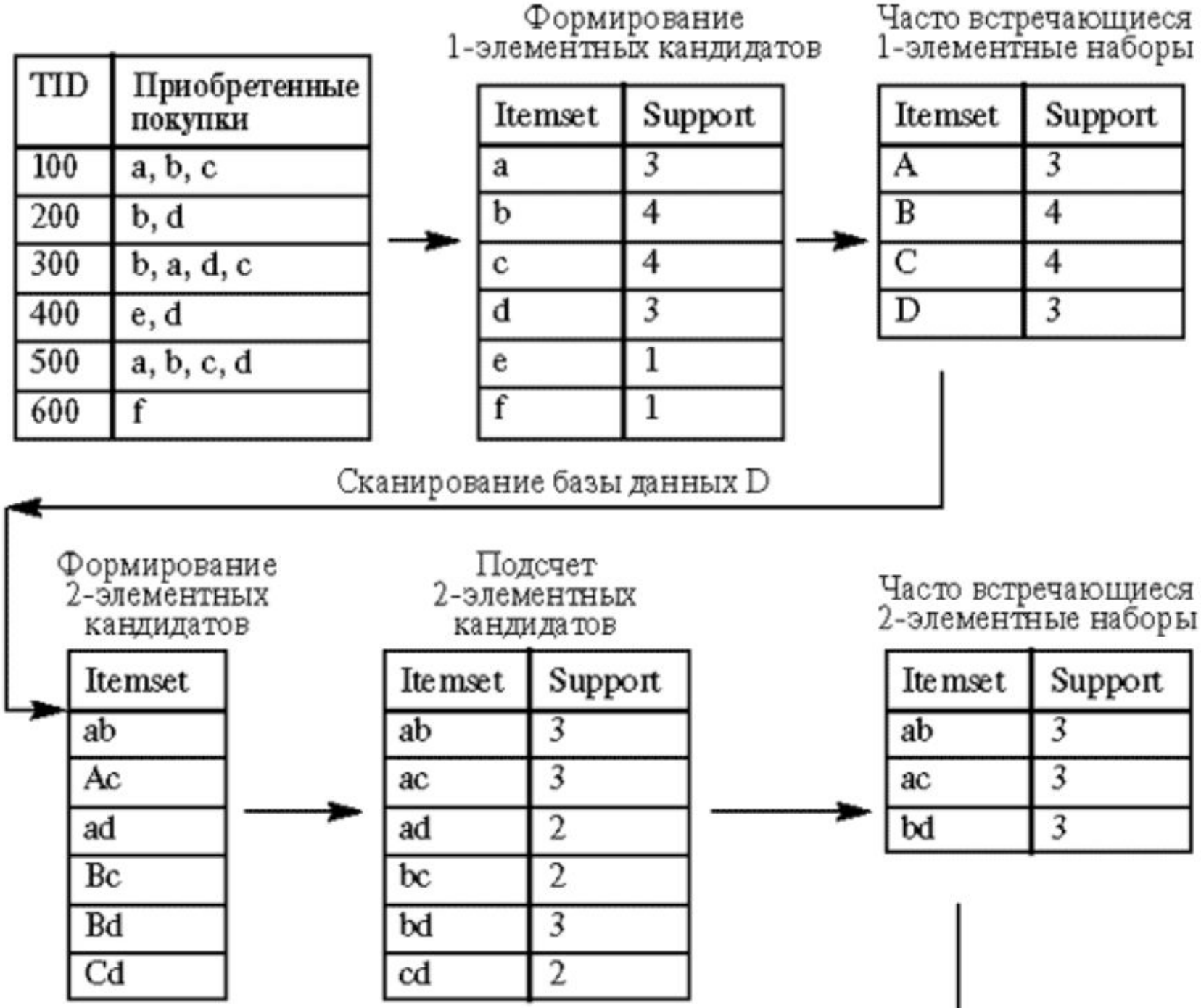
$$\mathrm{Supp}_F \leq \mathrm{Supp}_E \text{ при } E \subset F$$

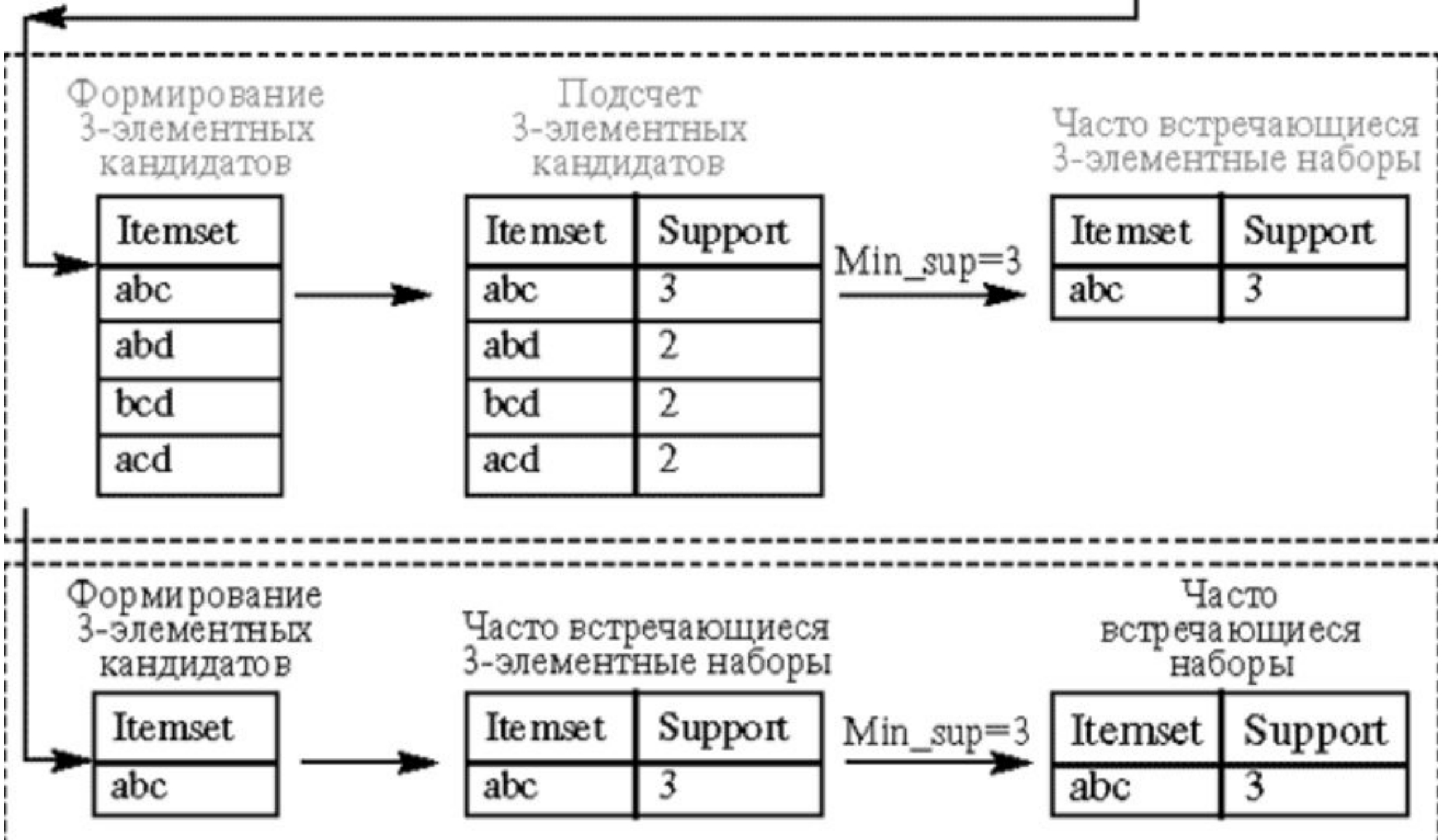
Например, поддержка 3-объектного набора **{соки, вода, чипсы}** будет всегда меньше или равна поддержке 2-объектных наборов **{соки, вода}, {вода, чипсы}, {соки, чипсы}.**

Это объясняется тем, что любая транзакция, содержащая **{соки, вода, чипсы},** содержит также и наборы **{соки, вода}, {вода, чипсы}, {соки, чипсы},** причем

- Алгоритм Apriori определяет часто встречающиеся наборы за несколько этапов. На i -м этапе определяются все часто встречающиеся i -элементные наборы. Каждый этап состоит из двух шагов: формирования кандидатов (candidate generation) и подсчета поддержки кандидатов (candidate counting).
- Рассмотрим i -й этап:
  - На ***шаге формирования кандидатов*** алгоритм создает множество кандидатов из i -элементных наборов, чья поддержка пока не вычисляется.
  - На ***шаге подсчета кандидатов*** алгоритм сканирует множество транзакций, вычисляя поддержку наборов-кандидатов.
- После сканирования отбрасываются кандидаты, поддержка которых меньше ***определенного пользователем минимума***, и сохраняются только часто встречающиеся i -элементные наборы.

- Разновидности алгоритма Apriori, являющиеся его оптимизацией, предложены для сокращения количества сканирований базы данных, количества наборов-кандидатов или того и другого: AprioriTID и AprioriHybrid.
- Алгоритм DHP, также называемый алгоритмом хеширования.
- Алгоритм PARTITION - разбиения (разделения) заключается в сканировании транзакционной базы данных путем разделения ее на непересекающиеся разделы, каждый из которых может уместиться в оперативной памяти.
- Алгоритм DIC, Dynamic Itemset Counting разбивает базу данных на несколько блоков, каждый из которых отмечается так называемыми "начальными точками" (start point), и затем циклически сканирует базу данных.


TID
Приобретенные покупки
100
a, b, c
200
b, d
300
b, a, d, c
400
e, d
500
a, b, c, d
600
f
Формирование 1-элементных кандидатов
Itemset
Support
a
3
b
4
c
4
d
3
e
1
f
1
Часто встречающиеся 1-элементные наборы
Itemset
Support
A
3
B
4
C
4
D
3
Сканирование базы данных D
Формирование 2-элементных кандидатов
Itemset
ab
Ac
ad
Bc
Bd
Cd
Подсчет 2-элементных кандидатов
Itemset
Support
ab
3
ac
3
ad
2
bc
2
bd
3
cd
2
Часто встречающиеся 2-элементные наборы
Itemset
Support
ab
3
ac
3
bd
3


Формирование 3-элементных кандидатов
Itemset
abc
abd
bcd
acd
Подсчет 3-элементных кандидатов
Itemset
Support
abc
3
abd
2
bcd
2
acd
2
Min_sup=3
Часто встречающиеся 3-элементные наборы
Itemset
Support
abc
3
Формирование 3-элементных кандидатов
Itemset
abc
Часто встречающиеся 3-элементные наборы
Itemset
Support
abc
3
Min_sup=3
Часто встречающиеся наборы
Itemset
Support
abc
3

№	Транзакция
1	Сливы, салат, помидоры
2	Сельдерей, конфеты
3	Конфеты
4	Яблоки, морковь, помидоры, картофель, конфеты
5	Яблоки, апельсины, салат, конфеты, помидоры
6	Персики, апельсины, сельдерей, помидоры
7	Фасоль, салат, помидоры
8	Апельсины, салат, морковь, помидоры, конфеты
9	Яблоки, бананы, сливы, морковь, помидоры, лук, конфеты

- Ассоциативное правило состоит из двух наборов предметов, называемых **условие** и **следствие**, записываемых в виде **X → Y**, что читается следующим образом: «Из X следует Y» или «Если условие, то следствие».
- **Условие может ограничиваться только одним предметом**.

помидоры → салат.

- Ассоциативные правила описывают связь между **условием** и **следствием**.

Эта связь характеризуется двумя показателями:

поддержкой (support) и **достоверностью** (confidence).

Поддержка ассоциативного правила S—
это отношение числа транзакций, которые содержат как условие, так и следствие к общему количеству транзакций.
Например, для ассоциации A → B можно записать:

$$\mathrm{supp}(X) = \frac{|\{t \in T; X \subseteq t\}|}{|T|}$$

$$S(A \to B) = P(A \cap B) = \frac{\text{количество транзакций, содержащих A и B}}{\text{общее количество транзакций}}$$

Достоверность ассоциативного правила A → B — это мера точности правила.
Определяется как отношение количества транзакций, содержащих и условие, и следствие, к количеству транзакций, содержащих только условие:

$$\mathrm{conf}(X \Rightarrow Y) = \mathrm{supp}(X \cup Y) / \mathrm{supp}(X)$$

$$C(A \to B) = P(A \mid B) = P(A \cap B) / P(A) = \frac{\text{количество транзакций, содержащих A и B}}{\text{количество транзакций, содержащих только A}}$$

- При поиске ассоциативных правил используются дополнительные показатели, позволяющие оценить значимость правила. Можно выделить объективные и субъективные меры значимости правил.
- Объективными являются такие меры, как **поддержка** и **достоверность**, которые могут применяться независимо от конкретного приложения.
- Субъективные меры связаны со специальной информацией, определяемой пользователем в контексте решаемой задачи. Такими субъективными мерами являются **лифт** (lift) и **левередж** (от англ. leverage — плечо, рычаг), **улучшение** (improvement) и др.

- **Лифт** (оригинальное название — интерес) вычисляется следующим образом:

$$L(A \rightarrow B) = C(A \rightarrow B) / S(B)$$

$$lift = \frac{\text{Confidence}}{\text{Expected confidence}}$$

- Лифт > 1 указывает, что условие и следствие чаще встречаются в транзакциях вместе, чем по отдельности.
- Лифт=1 указывает, что условие и следствие на появление друг друга не влияют.
- Лифт < 1, указывают на то, что условие и следствие встречаются в транзакциях чаще по отдельности, чем вместе.

По-другому lift можно определить как отношение confidence к expected confidence, т.е. отношение достоверности правила, когда оба элемента покупаются вместе к достоверности правила, когда один из элементов покупался (неважно, со вторым или без).

Другой мерой значимости правила, является **левередж**:

$$T(A \rightarrow B) = S(A \rightarrow B) - S(A)S(B)$$

- **Левередж** — это разность между наблюдаемой частотой, с которой условие и следствие появляются совместно (то есть поддержкой ассоциации), и произведением частот появления (поддержек) условия и следствия по отдельности.

- Рассмотрим еще одну характеристику ассоциативного правила, которую можно считать мерой полезности.

Она называется **улучшением** (improvement) и вычисляется подобно **левереджу**, только берется не разность, а отношение наблюдаемой частоты и частот

$$I(A \to B) = \frac{S(A \to B)}{S(A) \cdot S(B)}$$

Улучшение показывает, полезнее ли правило случайного угадывания.

- Если $I(A \to B) > 1$, это значит, что вероятнее предсказать наличие набора B с помощью правила, чем угадать случайно.

Некоторые популярные меры:

- Полное доверие (англ. All-confidence)
- Коллективная мощь (англ. Collective strength)
- Убедительность (англ. Conviction)

$$\mathrm{conv}(X \Rightarrow Y) = \frac{1 - \mathrm{supp}(Y)}{1 - \mathrm{conf}(X \Rightarrow Y)}.$$

В общем виде Conviction — это «частотность ошибок» правила.
Чем результат выше 1, тем лучше.

Выводы

- Задачей поиска ассоциативных правил является определение *часто встречающихся наборов объектов* в большом множестве наборов.
- Результаты решения задачи представляются в виде ассоциативных правил, условная и заключительная часть которых содержит наборы объектов.
- Основными характеристиками ассоциативных правил являются *поддержка*, *достоверность* и *улучшение*.
- *Поддержка* (support) показывает, какой процент транзакций поддерживает данное правило.
- *Достоверность* (confidence) показывает, какова вероятность того, что из наличия в транзакции набора условной части правила следует наличие в ней набора заключительной части.
- *Улучшение* (improvement) показывает, полезнее ли правило случайного угадывания.

- Задача *поиска ассоциативных правил* решается в два этапа:
  - На первом выполняется поиск всех частых наборов объектов.
  - На втором из найденных частых наборов объектов генерируются ассоциативные правила.
- Алгоритм *Apriori* использует одно из свойств поддержки, гласящее: поддержка любого набора объектов не может превышать ми [illegible] бого из его подмножеств. $\mathrm{Supp}_F \leq \mathrm{Supp}_E$ при $E \subset F$

Интерпретация ассоциативных правил

Все множество ассоциативных правил можно разделить на три вида:

1. Полезные правила – содержат информацию, которая ранее была неизвестна, но имеет логичное объяснение. Такие правила могут быть использованы для принятия решений.
2. Тривиальные правила – содержат информацию, которая уже известна. При анализе рыночных корзин в правилах с самой высокой поддержкой и достоверностью окажутся товары-лидеры продаж. Практическая ценность таких правил крайне низка.
3. Непонятные правила – содержат информацию, которая не может быть объяснена. Это или аномальные значения, или глубоко скрытые знания. Необъяснимость правил может привести к непредсказуемым результатам, требуется дополнительный анализ.

- Технология *Data Mining* не может дать ответы на те вопросы, которые не были заданы. Она не может заменить аналитика, а всего лишь дает ему мощный инструмент для облегчения и улучшения его работы.
- *Data Mining* достаточно часто делает множество ложных и не имеющих смысла открытий. Многие специалисты утверждают, что *Data Mining-средства* могут выдавать огромное количество статистически недостоверных результатов. Чтобы этого избежать, необходима проверка адекватности полученных моделей на тестовых данных.
- Успешный анализ требует качественной предобработки данных. По утверждению аналитиков и пользователей баз данных, процесс предобработки может занять до 80% процентов всего *Data Mining-процесса*.