Глава 2: Входни данни, концепции, екземпляри и атрибути

Компоненти на входните данни за обучение

Концепцията представлява ***модел*** или ***правило***, което алгоритъмът се опитва да научи от данните. Това може да бъде:

- **Classification (Класификация)** – предсказване на дискретна категория (напр. дали имейлът е спам или не).
- **Association (Асоцииране)** – намиране на връзки между различни елементи (напр. в анализ на потребителско поведение – „Клиентите, които купуват хляб, често купуват и масло“).
- **Clustering (Клъстериране)** – групиране на сходни елементи без предварително зададени категории.
- **Numeric Prediction (Числова прогноза)** – предсказване на числова стойност (напр. прогнозиране на цената на жилище).

Примерът(example) представлява *индивидуална наблюдавана стойност* (наблюдение) в набора от данни:

- **Relations (Релации)** – свързани обекти в база от данни.
- **Flat Files (Плоски файлове)** – таблици с редове и колони (напр. CSV файлове).
- **Recursion (Рекурсия)** – случаи, при които примерите са свързани помежду си в йерархични или рекурсивни структури.

Атрибутът измерва различни характеристики на един пример. Разделяме ги в няколко вида:

- **Nominal (Номинален)** – категории без подредба (напр. цвят на кола: червен, син, зелен).

- **Ordinal (Ординален)** – категории с подредба, но без равномерна разлика между тях (напр. оценка: нисък, среден, висок).
- **Interval (Интервален)** – числови стойности със смислена разлика между тях, но без истинска нулева стойност (напр. температура в градуси Целзий).
- **Ratio (Отношение)** – числови стойности с истинска нулева точка (напр. тегло, височина, време).

Подготовка на входните данни- критичен етап в процеса на машинно обучение. Ако данните не са правилно обработени, моделът може да даде неточни резултати. Основните аспекти на този процес включват:

- **ARFF (Attribute-Relation File Format)** – стандартен формат за файлове, използван в софтуера *WEKA* за представяне на данни. Съдържа описание на атрибутите и самите данни.
- **Sparse Data (Редки данни)** – случаи, при които повечето стойности са нули или липсващи.
- **Attributes (Атрибути)** – Изборът на правилните характеристики за модела е от решаващо значение. Често се налага трансформиране на данните, за да станат по-информативни.
- **Missing and Inaccurate Values (Липсващи и неточни стойности)** – обработка на пропуснати данни и шум в данните. Входните данни често съдържат липсващи или погрешни стойности. Те могат да бъдат коригирани чрез методи като запълване със средни стойности, премахване на

грешни записи или използване на модели за предсказване на липсващите данни.

- **Unbalanced Data (Небалансирани данни)** – когато някои класове са представени с много повече примери от други (напр. 95% от данните са за "не-спам", а само 5% за "спам"). Моделът може да игнорира редките случаи. За справяне с това се използват техники като балансиране на класовете, претегляне на примерите или генериране на нови данни чрез методи като SMOTE.
- **Getting to Know Your Data (Запознаване с данните)** – анализиране и визуализиране на входните данни преди обучение.

За да работи машинното обучение, то използва входни данни, които се състоят от няколко основни елемента – **Компоненти на входните данни**:

- **Concepts (Концепции):** Видовете знания, които можем да извлечем от данните. **Цел:** Да разберем и опишем ясно какво означава дадена концепция. Например, ако анализираме данни за времето, концепция може да бъде „дали ще вали или не“.
- **Instances (Екземпляри):** Това са отделните примери в набора от данни. Всеки екземпляр представлява един запис или случай. **Важно**: Понякога има връзки между примерите. Например, при анализ на пациентски данни, здравословното състояние на един човек може да бъде свързано с неговото минало лечение.
- **Attributes (Атрибути):** Характеристики, които измерват различни аспекти на даден екземпляр. **Видове атрибути**:

- Категорични (номинални) – Например „цвят на кола“ (червен, син, зелен).
- Числови – Например „височина в сантиметри“ или „температура в градуси“.

Concept (Концепция): Това е знанието, което трябва да бъде научено от алгоритъма. **Concept description (Описание на концепция)**: Това е резултатът (изходът) от процеса на обучение.

Стилове на обучение

Classification Learning (Класификационно обучение)

- Предсказване на дискретен клас- **конкретни категории (етикети), към които се отнасят обектите** в класификационни задачи
- **Примери за проблеми:** данни за времето, контактни лещи, ириси, преговори за трудови договори.
- Класификационното обучение е **supervised (контролирано)**.
- Алгоритъмът получава **реални изходни стойности** (етикети).
- Изходната стойност се нарича **class (клас) на примера**.
- Успехът се измерва върху **test data (тестови данни)**, където класовете са известни.
- На практика успехът често се измерва **субективно**.

Асоциативно обучение -Прилага се, когато **няма предварително зададен клас**, а всяка структура се счита за „интересна“. Разлика спрямо класификационното обучение:

  - Може да предсказва **стойността на всеки атрибут, а не само класа**.
  - Може да предсказва **повече от един атрибут едновременно**.
- **В резултат:**

- **Много повече асоциативни правила, отколкото класификационни.**
- **Затова са необходими ограничения**, като минимално покритие и минимална точност.
- **Пр.** в търговията на дребно може да се използва асоциативно обучение, за да се открият зависимости между продукти, които често се купуват заедно. Ако клиент закупи хляб, има голяма вероятност да закупи и мляко. Тези асоциации се представят като правила:"Ако клиентът купи хляб, тогава ще купи и мляко".
- Разликата спрямо класификационното обучение е, че тук няма предварително зададен клас, а се откриват асоциации между атрибути. Също така, можете да предсказвате повече от един атрибут едновременно.

Clustering (Клъстериране)

- Откриване на групи от подобни обекти.
- Клъстерирането е **unsupervised (неконтролирано)**.
- **Класът на даден пример не е известен предварително.**
- Успехът често се измерва **субективно**.
- Пр. в маркетинга може да се използва клъстериране, за да се открият групи от потребители, които имат подобни навици за пазаруване. Клъстерирането ще групира потребителите в различни сегменти, като "млади хора, които често купуват спортни стоки" или "пенсионери, които предпочитат книги".
- Клъстерирането е неконтролирано обучение (unsupervised), защото не знаем предварително какви групи ще открием. Успехът на алгоритъма зависи от това как ще се интерпретират тези групи.

Numeric Prediction (Числова прогноза)

- Вариант на класификационното обучение, при който **“class” (клас) е числова стойност**.
- Нарича се още **regression (регресия)**.

- Обучението е **supervised (контролирано)**.
- Алгоритъмът получава **целеви стойности (target value)**.
- Успехът се измерва върху **тестови данни**.
- Пр: В икономиката може да се използва числова прогноза за предсказване на стойността на акциите на базата на исторически данни. Вместо да се класифицират акциите в категории (например "расте" или "пада"), числовата прогноза дава конкретна стойност, например прогноза за стойността на акциите след определен период.
- Това е вид регресия (regression), където целевата стойност е числова, а успехът на модела се измерва чрез точността на предсказаните стойности върху тестови данни.

Instance (Екземпляр):

- **Конкретен пример** за концепцията, която трябва да бъде класифицирана, асоциирана или клъстерирана.
- **Независим отделен обект**, който има **предварително дефиниран набор от атрибути**.
- Входът за алгоритъма за обучение е **набор от екземпляри (dataset, набор от данни)**.
- Представя се като **единична релация/плосък файл (flat file)**.
- **Ограничения:**
  - Липса на **връзки между обектите**.
  - **Най-често срещаната форма в практическото извличане на данни.**

Generating a Flat File (Генериране на плосък файл)

- Процесът на **“flattening” (изравняване)** се нарича **denormalization (денормализация)**.
- **Обединяват се няколко релации в една.**
- Може да се направи с всяко крайно множество от крайни релации.
- **Проблем:** Връзки без предварително зададен брой обекти.

- **Денормализацията може да доведе до фалшиви зависимости**, отразяващи структурата на базата данни.

Recursion (Рекурсия) - **функция или процес извиква сам себе си**, докато не достигне някакво крайно условие

- **Неограничените релации изискват рекурсия**, ,защото имаме данни, които са свързани в **непредвиден брой нива**.
- Подходящите техники са известни като **Inductive Logic Programming (ILP, Индуктивно логическо програмиране)**.
- Пример за ILP метод: **Quinlan's FOIL rule learner (FOIL – алгоритъм за индукция на правила)**.
- **Основни проблеми:**

(a)Шум в данните (noise) - Неточни или противоречиви данни могат да доведат до грешни изводи.

(b) Изчислителна сложност (computational complexity) - При големи обеми от данни рекурсията може да стане бавна и ресурсоемка.

Multi-Instance Concepts (Концепции с множество екземпляри)

- Всеки индивидуален пример съдържа **„чувал" (bag)** от екземпляри.
- Всички екземпляри се описват чрез **същите атрибути**.
- Един или повече екземпляри могат да определят класификацията на примера.
- Целта на обучението остава същата – създаване на описание на концепцията.
- **Важни реални приложения:**
  - Прогнозиране на активността на лекарства (drug activity prediction).
  - Класификация на изображения (image classification).
- **Примери:**

    - Лекарството се разглежда като „чувал“ от различни геометрични конфигурации на молекулите.
    - Изображението се представя като „чувал“ от компоненти на изображението.

Attribute (атрибут)

- Всеки екземпляр се описва чрез предварително зададен **набор от характеристики (features), наречени атрибути.**
- **Но:** Броят на атрибутите може да варира в практиката.
- **Възможно решение:** Използване на **флаг „irrelevant value” (нерелевантна стойност)**.
- **Свързан проблем:** Съществуването на даден атрибут може да зависи от стойността на друг атрибут.

Възможни типове атрибути (Levels of Measurement – Нива на измерване):

- **Nominal (Номинален):** Категории без подредба (цвят на кола: червен, син, зелен).
- **Ordinal (Ординален):** Категории с подредба, но без равномерна разлика между тях (оценка: нисък, среден, висок).
- **Interval (Интервален):** Числови стойности със смислена разлика, но без абсолютна нулева точка (температура в градуси Целзий).
- **Ratio (Отношение):** Числови стойности с абсолютна нулева точка (тегло, височина, време).

Нива на измерване (Levels of Measurement)

Nominal Levels of Measurement (Номинални нива на измерване)

- **Стойностите са просто символи** – служат **само като етикети или имена**.
- **Nominal** идва от латинската дума за **„име“**.

- **Пример:** Атрибут „outlook“ (прогноза) в данните за времето: Възможни стойности: **„sunny“ (слънчево), „overcast“ (облачно), „rainy“ (дъждовно)**.
- **Няма връзка или подредба между стойностите** (няма ред, няма мерки за разстояние).
- Единствената възможна операция е **проверка за равенство**.

Ordinal Levels of Measurement (Ординални нива на измерване)

- **Добавя ред към стойностите**, но **няма дефинирано разстояние между тях**. **Пример:** Атрибут „temperature“ (температура) в данните за времето: Възможни стойности: **„hot“ (горещо) > „mild“ (умерено) > „cool“ (хладно)**.
- **Събиране и изваждане на стойности не е смислено. Пример за правило:** Ако **temperature < hot → play = yes (играем = да)**.
- **Разликата между номинални и ординални стойности не винаги е ясна** (например атрибутът „outlook“).

Interval Quantities (Интервални величини)

- **Редът е дефиниран**, но **стойностите са измерени в равномерни интервали**. **Пр:** Атрибут „temperature“ (температура), измерена в градуси по Фаренхайт; Атрибут „year“ (година).
- **Разликата между две стойности има смисъл**.
- **Но:** Нулата **не е дефинирана като абсолютна стойност**.
- **Сумиране и умножение не са смислени операции**.

Ratio Quantities (Отношения величини)

- **Дефинирана е абсолютна нулева точка**. **Пр:** Атрибут „distance“. Разстоянието между един обект и себе си е **0**.

- **Отношения величини се третират като реални числа**.
- **Всички математически операции са позволени**.
- **Но:** Винаги ли има „естествена“ нулева точка?

    - **Отговорът зависи от научните знания.**
    - Например **Фаренхайт първоначално не е знаел за най-ниската възможна температура**

Типове атрибути, използвани на практика

- **Много алгоритми за Data Mining работят само с два типа измервания: номинални и ординални.**
- **Други алгоритми работят само с ratio (отношения) величини.**
- **Номиналните атрибути** се наричат още **“categorical” (категориални), “enumerated” (изброими) или “discrete” (дискретни)**.
    - Но: **“enumerated” и “discrete” предполагат ред**, което не винаги е вярно.
    - **Специален случай:** Dichotomy (дихотомия) → **Boolean (булев) атрибут** (напр. „да“/„не“).
- **Ординалните атрибути** понякога се кодират като **“numeric” (числови) или “continuous” (непрекъснати)**.
    - Но: **“continuous” предполага математическа непрекъснатост**, което не винаги е вярно.

Metadata (Метаданни)

- **Метаданните** съдържат **информация за данните**- данни за данните.
- **Теоретично:** Може да се използват за **ограничаване на пространството за търсене** на алгоритъма.
- **Примери:**
    - **Дименсионални ограничения** (напр. математически изрази трябва да са дименсионално правилни).
    - **Циклични подредби** (напр. градуси в компаса).
    - **Частични подредби** (напр. релации между обобщения и специализации).

Подготовка на входните данни

- **Денормализацията не е единственият проблем** при подготовката на данни за обучение.
- **Проблем:** Различни източници на данни (напр. отдел продажби, отдел фактуриране на клиенти и др.).
- **Разлики:**
  - Различни стилове на съхранение на записи.
  - Кодиране на данни.
  - Различни времеви периоди.
  - Агеграция на данни.
  - Различни първични ключове.
  - Различни видове грешки.
- **Данните трябва да бъдат събрани, интегрирани и почистени.**
- **“Data warehouse” (Хранилище на данни):** Единна и последователна точка за достъп до данните.
- **Понякога са необходими външни данни („overlay data“).**
- **Критично:** Видът и нивото на агрегиране на данните

Допълнителни типове атрибути

  - **Форматът ARFF** поддържа и **“string” (низови) атрибути**: Подобни на **номинални**, но **списъкът с възможни стойности не е предварително зададен**.
  - **Поддържа и “date” (дата) атрибути**: Използва формата **ISO-8601: yyyy-MM-dd-THH:mm:ss**.

Relational Attributes (Релационни атрибути)

- Позволяват представяне на **multi-instance (многоекземплярни) проблеми** във формат ARFF.
- Всяка стойност на релационен атрибут представлява **отделен „чувал“ от екземпляри**.
- Всеки „чувал“ съдържа **едни и същи атрибути**.

Sparse Data (Рехави/Разредени данни)

- В някои приложения **повечето стойности на атрибутите са нули**.
- **Решение:** Използване на **компактен начин за съхранение**.
- **Пример:** Броене на думи при категоризация на текстове.
- **Форматът ARFF поддържа разредено съхранение на данни.**
- Някои алгоритми работят **много по-ефективно** със sparse data.

Missing Values (Липсващи стойности)

- Често се маркират с **извън обхвата стойности** за даден атрибут.
- **Видове липсващи стойности:**
  - **“Unknown” (неизвестна)**
  - **“Unrecorded” (незаписана)**
  - **“Irrelevant” (нерелевантна)**
- **Причини:**
  - Проблеми с оборудването.
  - Промени в експерименталния дизайн.
  - Комбиниране на различни набори от данни.
  - Невъзможност за измерване.
- **Липсваща стойност може да носи собствено значение!**
  - Пример: Липсващ тест в медицински преглед може да бъде значещ.

Unbalanced Data (Небалансирани данни)

- Проблем, при който **единият клас е много по-често срещан от останалите**.
- **Пример:** Откриване на рядко заболяване.
- **Решение:** Използване на техники, които **отчитат неравните разходи за грешна класификация**.