# 3.2 Data Cleaning

| * Real-world data tends to be incomplete, noisy, and inconsistent. Data cleaning routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data. * Trong data mining, data cleaning (hoặc data cleansing) là một bước quan trọng nhằm đảm bảo rằng dữ liệu được sử dụng trong quá trình phân tích là chính xác, đầy đủ và nhất quán. |
| --- |

# 

## 3.2.1 Missing Values

| **Vấn đề**: Tưởng tượng rằng bạn cần phân tích dữ liệu bán hàng và khách hàng của AllElectronics. Bạn nhận thấy rằng *nhiều bản ghi không có giá trị được ghi lại cho một số thuộc tính như thu nhập của khách hàng*. Bạn có thể làm gì để điền vào các giá trị còn thiếu cho thuộc tính này? Hãy cùng xem qua các phương pháp sau.    (Dữ liệu mẫu cho phần Missing Values)   1. **Bỏ qua bản ghi (Ignore the tuple)**  * Thường được thực hiện khi nhãn lớp bị thiếu. * Phân tích một tập dữ liệu về nhân viên của một công ty, trong đó mỗi bản ghi đại diện cho một nhân viên và có một nhãn lớp để chỉ ra rằng khách hàng này có họ tên hay không. Tuy nhiên, trong một số bản ghi, nhãn lớp này bị thiếu. Nếu mục tiêu của bạn là xây dựng một mô hình phân loại để tính toán điểm GPA, bạn có thể bỏ qua các bản ghi không có nhãn lớp, vì các bản ghi này không cung cấp thông tin cần thiết. Mặc dù việc bỏ qua bản ghi có thể làm mất một phần dữ liệu, nhưng trong trường hợp này, những bản ghi không có nhãn lớp không giúp ích cho quá trình tính toán, nên việc bỏ qua chúng là hợp lý. * Phương pháp này không thực sự hiệu quả, trừ khi bản ghi chứa nhiều thuộc tính có giá trị bị thiếu. * Phân tích dữ liệu nhân viên của một công ty, với các thuộc tính như tên, vị trí công việc, thu nhập, số năm kinh nghiệm, điểm GPA. Một vài bản ghi chỉ thiếu một hoặc hai giá trị như vị trí làm việc hoặc số năm kinh nghiệm. Trong trường hợp này, việc bỏ qua toàn bộ bản ghi chỉ vì thiếu một vài giá trị không phải là cách hiệu quả, vì hầu hết các thuộc tính khác vẫn có giá trị và có thể hữu ích cho phân tích. Tuy nhiên, nếu một bản ghi thiếu nhiều thuộc tính quan trọng như cả tên, thu nhập thì bỏ qua bản ghi đó sẽ hợp lý hơn, vì dữ liệu thiếu quá nhiều và không đủ thông tin để thực hiện phân tích có giá trị. Trong trường hợp dữ liệu thiếu quá ít, việc bỏ qua bản ghi sẽ dẫn đến mất mát dữ liệu quan trọng mà vẫn có thể hữu ích cho các phương pháp dự đoán hoặc phân tích khác. * Kém hiệu quả khi tỷ lệ giá trị bị thiếu trên mỗi thuộc tính có sự khác nhau đáng kể. * Phân tích dữ liệu về nhân viên công ty . * Ở một số bản ghi, thuộc tính tên chỉ thiếu ở khoảng 12% các trường hợp. * Thuộc tính mức độ số năm kinh nghiệm lại bị thiếu ở 25% các bản ghi. * Nếu bạn áp dụng phương pháp bỏ qua các bản ghi có giá trị bị thiếu, thì những bản ghi có thiếu thuộc tính "số năm kinh nghiệm" sẽ bị loại bỏ với số lượng lớn (25% dữ liệu bị loại), trong khi những bản ghi thiếu thuộc tính "chiều cao" chỉ chiếm một phần rất nhỏ (12%). Trong trường hợp này, tỷ lệ thiếu giá trị giữa các thuộc tính rất khác nhau, và việc bỏ qua toàn bộ các bản ghi có giá trị thiếu sẽ dẫn đến mất mát dữ liệu không đồng đều. Điều này có thể làm sai lệch phân tích, vì bạn đang loại bỏ quá nhiều dữ liệu chỉ dựa trên một vài thuộc tính có tỷ lệ giá trị bị thiếu cao hơn. * Bằng cách bỏ qua bản ghi, chúng ta không sử dụng các giá trị của các thuộc tính còn lại trong bản ghi đó. * Phân tích dữ liệu nhân viên của một công ty. Mỗi bản ghi bao gồm các thuộc tính như tên, vị trí công việc, thu nhập, số năm kinh nghiệm. Một số bản ghi thiếu thông tin về số năm kinh nghiệm của nhân viên, nhưng các thuộc tính khác như tên, vị trí công việc, lương vẫn được điền đầy đủ. Nếu bạn chọn phương pháp bỏ qua toàn bộ những bản ghi này chỉ vì thiếu số năm kinh nghiệm, bạn sẽ bỏ lỡ các thông tin quan trọng từ những thuộc tính khác. Bằng cách bỏ qua toàn bộ bản ghi chỉ vì thiếu một thuộc tính, bạn không tận dụng được các thuộc tính còn lại mà có thể vẫn hữu ích cho việc phân tích. * **Ignore the tuple:** * This is usually done when the class label is missing * This method is not very effective, unless the tuple contains several attributes with missing values      * It is especially poor when the percentage of missing values per attribute varies considerably * By ignoring the tuple, we do not make use of the remaining attributes’ values in the tuple  1. **Điền thủ công (Fill in manually)**  * Tốn thời gian và có thể không khả thi khi phải xử lý một tập dữ liệu lớn với nhiều giá trị bị thiếu. * Bạn đang làm việc với một tập dữ liệu gồm 100.000 nhân viên của một công ty, bao gồm các thuộc tính như tên,vị trí công việc, thu nhập, số năm kinh nghiệm. Trong đó, có khoảng 20% bản ghi thiếu giá trị ở một hoặc nhiều thuộc tính, đặc biệt là tên và thu nhập. Nếu bạn cố gắng điền thủ công các giá trị bị thiếu này, bạn sẽ phải kiểm tra từng bản ghi, liên hệ trực tiếp với nhân viên hoặc sử dụng các nguồn khác để xác minh thông tin và điền giá trị. Điều này sẽ tiêu tốn rất nhiều thời gian và nguồn lực, đặc biệt khi số lượng bản ghi lớn và số lượng giá trị bị thiếu nhiều. Với quy mô tập dữ liệu lớn như vậy, phương pháp điền thủ công trở nên không khả thi. Các công ty thường phải tìm cách tự động hóa quy trình này hoặc sử dụng các phương pháp khác như ước tính giá trị bị thiếu dựa trên dữ liệu hiện có thay vì điền từng bản ghi một cách thủ công.  1. **Sử dụng biến hằng số toàn cục**  * Thay thế tất cả các giá trị thuộc tính bị thiếu bằng cùng một hằng số. * Phương pháp này đơn giản, nhưng nó không hoàn toàn đáng tin cậy. * Giả sử bạn có một tập dữ liệu về nhân viên của một công ty bao gồm các thuộc tính như tên, vị trí công việc, thu nhập. Nhiều bản ghi thiếu thông tin về thu nhập, vì vậy bạn quyết định thay thế tất cả các giá trị bị thiếu này bằng hằng số “Unknown” (Không xác định). Mặc dù phương pháp này đơn giản, nhưng nó có một số vấn đề. Khi bạn thực hiện phân tích dữ liệu, chương trình khai thác dữ liệu có thể hiểu nhầm rằng “Unknown” là một giá trị hợp lệ và có thể đưa ra kết luận không chính xác. Ví dụ, nếu bạn đang thực hiện phân tích trình độ nhân viên dựa trên thu nhập, tất cả những nhân viên đó có thu nhập bị thay thế bởi “Unknown”, có thể bị nhóm lại thành một phần riêng, mặc dù trên thực tế, họ không có điểm chung nào ngoài việc thiếu dữ liệu. * Do đó, mặc dù việc sử dụng hằng số để thay thế giá trị bị thiếu là một phương pháp đơn giản, nó không hoàn toàn đáng tin cậy vì có thể dẫn đến các kết quả phân tích sai lệch.      1. **Sử dụng thước đo khuynh hướng trung tâm (Use a measure of central tendency)**  * Các biện pháp về xu hướng trung tâm chỉ ra giá trị “trung bình” của phân phối dữ liệu. Ví dụ: Trung bình, Trung vị.. * Trong trường hợp thuộc tính là thuộc tính số: * Đối với phân phối dữ liệu chuẩn (đối xứng), có thể sử dụng giá trị trung bình. (Đối với phân phối dữ liệu lệch, chúng ta nên sử dụng trung vị. * Ví dụ: * Giả sử bạn đang làm việc với một tập dữ liệu về nhân viên bao gồm các thuộc tính như tên, vị trí công việc, điểm trung bình (GPA), số năm kinh nghiệm và thu nhập. Một số bản ghi bị thiếu giá trị của thuộc tính GPA. Bạn có thể điền giá trị bị thiếu bằng cách sử dụng điểm trung bình GPA cho toàn bộ tập nhân viên. Khi gặp bản ghi của một sinh viên có giá trị GPA bị thiếu, bạn sẽ thay thế giá trị này bằng giá trị trung bình của thuộc tính GPA.      1. **Sử dụng giá trị trung bình hoặc trung vị của thuộc tính cho tất cả các mẫu thuộc cùng một lớp với bản ghi cho trước (Use the attribute mean or median for all samples belonging to the same class as the given tuple)**  * Nếu phân loại khách hàng theo “năm kinh nghiệm”, chúng ta có thể thay thế giá trị bị thiếu bằng giá trị thu nhập trung bình của những khách hàng trong cùng danh mục “năm kinh nghiệm” với danh mục đã cho. * Nếu phân phối dữ liệu cho một lớp nhất định bị lệch, giá trị trung vị sẽ là lựa chọn tốt hơn. * Ví dụ: * Giả sử bạn đang làm việc với một tập dữ liệu về nhân viên bao gồm các thuộc tính như tên, vị trí công việc, điểm trung bình (GPA), số năm kinh nghiệm và thu nhập. Một số bản ghi bị thiếu giá trị của thuộc tính số lương. Bạn có thể điền giá trị bị thiếu bằng cách sử dụng điểm trung bình của số lương của toàn bộ nhân viên có cùng số năm kinh nghiệm với đối tượng bị thiếu .      1. **Sử dụng giá trị có xác suất lớn nhất để điền vào giá trị còn thiếu**  * Điều này có thể được xác định bằng hồi quy, các công cụ dựa trên suy luận sử dụng công thức Bayesian hoặc phương pháp quy nạp cây quyết định. * Ví dụ, bằng cách sử dụng các thuộc tính nhân viên khác trong tập dữ liệu của mình, bạn có thể xây dựng cây quyết định để dự đoán các giá trị còn thiếu cho thu nhập của họ. * **Use the most probable value to fill in the missing value** * This may be determined with regression, inference-based tools using a Bayesian formalism, or decision tree induction     Tổng quan về các phương pháp trên:   * Các phương pháp 3 đến 6 có thể gây ra thiên lệch cho dữ liệu — giá trị được điền vào có thể không chính xác. * Phương pháp số 6 là một chiến lược phổ biến, sử dụng nhiều thông tin từ dữ liệu hiện có nhất để dự đoán giá trị bị thiếu. Bằng cách xem xét các giá trị của các thuộc tính khác trong việc ước tính giá trị bị thiếu, có cơ hội cao hơn rằng mối quan hệ giữa thu nhập và các thuộc tính khác sẽ được bảo toàn. * Trong một số trường hợp, giá trị bị thiếu có thể không phải là một lỗi trong dữ liệu. Mặc dù chúng ta có thể cố gắng hết sức để làm sạch dữ liệu sau khi thu thập, nhưng việc thiết kế quy trình nhập liệu và cơ sở dữ liệu tốt sẽ giúp giảm thiểu số lượng giá trị bị thiếu hoặc lỗi ngay từ đầu.  1. Xem bên trên, dưới cùng p1   **2.Fill in manually**   * Time-consuming and may not be feasible when dealing with a large dataset with many missing values.   **3.Use a global constant**   * Replace all missing attribute values with the same constant   **4. Use a measure of central tendency**   * Measures of central tendency indicate the “middle” value of data distribution(For example: Mean, Median. * For normal (symmetric) data distributions, the mean can be used * For skewed data distribution, we should employ the median   **5.Use the attribute mean or median for all samples belonging to the same class as the given tuple**   * if classifying customers according to year of exp, we may replace the missing value with the mean income value for customers in the same year of exp category as that of the given tuple * If the data distribution for a given class is skewed, the median value is a better choice.   **6. Use the most probable value to fill in the missing value**   * This may be determined with regression, inference-based tools using a Bayesian formalism, or decision tree induction   **Conclusion:**   * Methods 3 through 6 bias the data—the filled-in value may not be correct. * Method 6 is a popular strategy that uses the most information from the existing data to predict the missing value. By considering the values of other attributes in estimating the missing value, there is a higher chance that the relationships between income and the other attributes will be preserved. * In some cases, a missing value may not be an error in the data. While we can do our best to clean the data after collection, designing good data entry processes and databases will help minimize the number of missing values or errors from the outset.   (Tìm hiểu và trình bày bởi Bach Nguyen) |
| --- |

# 

## 3.2.2 Noisy Data

| **Dịch sách:(ai cần thì đọc nha, phần tóm tắt chứa ý chính)**  "Nhiễu là gì?" Nhiễu là lỗi ngẫu nhiên hoặc giá trị khác thường một biến đo lường. Trong Chương 2, chúng ta đã thấy một số kỹ thuật mô tả thống kê cơ bản (ví dụ: hộp boxplots và biểu đồ phân tán), và các phương pháp trực quan hóa dữ liệu có thể được sử dụng để nhận diện các ngoại lệ (outliers), mà có thể đại diện cho nhiễu. Với một thuộc tính số như giá cả, làm thế nào chúng ta có thể "làm mượt" dữ liệu để loại bỏ nhiễu? Hãy xem các kỹ thuật làm mượt dữ liệu dưới đây.  **Phân đoạn (Binning):** Phương pháp phân đoạn làm mượt giá trị dữ liệu đã được sắp xếp bằng cách tham khảo "láng giềng" của nó, tức là các giá trị xung quanh nó. Các giá trị đã được sắp xếp được phân bố vào một số "giỏ" hoặc các đoạn (bins). Vì phương pháp phân đoạn tham khảo các giá trị láng giềng, nó thực hiện làm mượt cục bộ. Hình 3.2 minh họa một số kỹ thuật phân đoạn. Trong ví dụ này, dữ liệu về giá cả trước tiên được sắp xếp và sau đó phân chia thành các đoạn có tần suất bằng nhau với kích thước 3 (tức là mỗi đoạn chứa ba giá trị). Trong làm mượt bằng trung bình đoạn (bin means), mỗi giá trị trong một đoạn được thay thế bằng giá trị trung bình của đoạn đó. Ví dụ, trung bình của các giá trị 4, 8 và 15 trong Đoạn 1 là 9. Do đó, mỗi giá trị gốc trong đoạn này được thay thế bằng giá trị 9.  Tương tự, phương pháp làm mượt bằng trung vị đoạn (bin medians) có thể được áp dụng, trong đó mỗi giá trị trong đoạn được thay thế bằng trung vị của đoạn. Trong làm mượt bằng biên đoạn (bin boundaries), giá trị nhỏ nhất và lớn nhất trong một đoạn được xác định là các biên đoạn. Mỗi giá trị trong đoạn sau đó được thay thế bằng giá trị biên gần nhất. Nói chung, càng mở rộng kích thước của đoạn thì hiệu ứng làm mượt càng lớn. Ngoài ra, các đoạn có thể có độ rộng bằng nhau, nơi phạm vi khoảng giá trị trong mỗi đoạn là không đổi. Phân đoạn cũng được sử dụng như một kỹ thuật phân loại và sẽ được thảo luận thêm ở Phần 3.5.  **Hồi quy (Regression):** Làm mượt dữ liệu cũng có thể được thực hiện bằng hồi quy, một kỹ thuật khớp các giá trị dữ liệu theo một hàm số. Hồi quy tuyến tính bao gồm việc tìm ra đường "tốt nhất" để khớp hai thuộc tính (hoặc biến) để một thuộc tính có thể được sử dụng để dự đoán thuộc tính kia. Hồi quy tuyến tính bội là một phần mở rộng của hồi quy tuyến tính, nơi nhiều hơn hai thuộc tính tham gia và dữ liệu được khớp với một bề mặt đa chiều. Hồi quy sẽ được mô tả thêm ở Phần 3.4.5.  **Phân tích ngoại lệ (Outlier analysis):** Các ngoại lệ có thể được phát hiện thông qua phân cụm, ví dụ, nơi các giá trị tương tự được tổ chức thành các nhóm, hoặc "cụm". Trực quan, các giá trị nằm ngoài các cụm có thể được coi là ngoại lệ. Chương 12 được dành riêng cho chủ đề phân tích ngoại lệ.  Nhiều phương pháp làm mượt dữ liệu cũng được sử dụng để phân loại dữ liệu (một dạng chuyển đổi dữ liệu) và giảm kích thước dữ liệu. Ví dụ, các kỹ thuật phân đoạn được mô tả trước đó làm giảm số lượng giá trị khác biệt trên mỗi thuộc tính. Điều này hoạt động như một dạng giảm kích thước dữ liệu cho các phương pháp khai phá dữ liệu dựa trên logic, chẳng hạn như suy diễn cây quyết định, trong đó lặp lại việc so sánh giá trị trên dữ liệu đã được sắp xếp. Hệ thống phân cấp khái niệm là một dạng phân loại dữ liệu cũng có thể được sử dụng để làm mượt dữ liệu. Hệ thống phân cấp khái niệm cho giá cả, ví dụ, có thể ánh xạ các giá trị giá thực thành rẻ, giá vừa phải, và đắt, do đó giảm số lượng giá trị dữ liệu cần được xử lý bởi quá trình khai phá. Phân loại dữ liệu sẽ được thảo luận ở Phần 3.5. Một số phương pháp phân loại (ví dụ, mạng nơron) có cơ chế làm mượt dữ liệu tích hợp. Phân loại là chủ đề của Chương 8 và 9.  **Tóm tắt:**  Dữ liệu nhiễu (Noise): Nhiễu là lỗi ngẫu nhiên hoặc sự biến thiên trong một biến đo lường. Nó có thể được xác định bằng các kỹ thuật mô tả thống kê như biểu đồ hộp (boxplots), biểu đồ phân tán (scatter plots), và trực quan hóa dữ liệu.   * Kỹ thuật làm mượt dữ liệu (Data Smoothing Techniques):   + Phân đoạn (Binning): Dữ liệu được sắp xếp và chia thành các đoạn (bins). Các kỹ thuật làm mượt bao gồm:     - Trung bình đoạn (Bin means): Thay thế mỗi giá trị trong đoạn bằng giá trị trung bình của đoạn. * Ví dụ, với một nhóm gồm các giá trị (4, 8, 6), giá trị trung bình là 6, và tất cả các giá trị trong nhóm sẽ được thay bằng 6   + - Trung vị đoạn (Bin medians): Thay thế giá trị bằng trung vị của đoạn. * Ví dụ, với nhóm gồm các giá trị (4, 8, 6), giá trị trung vị là 6, và tất cả các giá trị trong nhóm sẽ được thay bằng 6   + - Biên đoạn (Bin boundaries): Thay thế giá trị bằng biên nhỏ nhất hoặc lớn nhất trong đoạn. * Ví dụ, nếu một nhóm có các giá trị (15, 19, 22, 26), có 15 và 26 là giá trị nhỏ nhất và lớn nhất làm ranh giới. ta thấy 19 gần 15 hơn thì thay 19 thành 15; 22 gần 26 hơn thì ta thay 22 thành 26. => Ta được nhóm (15,15,26,26)      * + Hồi quy (Regression): Làm mượt dữ liệu bằng cách khớp các giá trị dữ liệu với một hàm số. Hồi quy tuyến tính giúp khớp hai thuộc tính để dự đoán giá trị. Hồi quy tuyến tính bội mở rộng cho nhiều thuộc tính hơn. * Ví dụ: hồi quy tuyến tính * i * Chúng ta muốn tìm mối quan hệ giữa tuổi và thu nhập. Hồi quy tuyến tính có thể giúp xác định một đường thẳng có dạng y = wx + b, trong đó w là hệ số và b là hằng số * Sau khi áp dụng hồi quy tuyến tính, có thể thu được hàm y = 800x + 10,000, cho biết rằng mỗi năm tuổi thêm vào sẽ làm tăng thu nhập trung bình khoảng 800 đơn vị. Phương trình này có thể được dùng để dự đoán thu nhập của một người dựa trên độ tuổi của họ      * + Phân tích ngoại lai (Outlier Analysis): Các ngoại lai có thể được phát hiện bằng phương pháp phân cụm (clustering). Các giá trị nằm ngoài cụm có thể được coi là ngoại lai. * Ví dụ: Chi tiêu hàng tháng (đô la): 200, 250, 220, 210, 205, 230, 400 * Giả sử chúng ta có dữ liệu về chi tiêu hàng tháng của một khách hàng. Nhìn vào dãy số trên, có thể dễ dàng nhận thấy rằng chi tiêu 400 đô la là rất khác biệt so với các giá trị còn lại (dao động quanh 200-250 đô la). Điểm dữ liệu này có thể được coi là ngoại lai và có thể cần kiểm tra lại xem nó có phải là một lỗi nhập liệu hoặc một sự kiện bất thường (ví dụ: mua sắm lớn).     Q1 = 205  Q2 = 220  Q3 = 250  IQR = 45  [137.5 ; 317.5]   * Phân loại và giảm kích thước dữ liệu (Discretization and Data Reduction): Các phương pháp làm mượt như phân đoạn có thể giúp giảm số lượng giá trị khác biệt trong dữ liệu, phục vụ cho các phương pháp khai phá dữ liệu như suy diễn cây quyết định.   **Tiếng anh:**   * Noisy Data (Noise): Noise refers to random errors or variations in a measured variable. It can be identified using statistical description techniques such as boxplots, scatter plots, and data visualization methods. * Data Smoothing Techniques:   + Binning: Data is sorted and divided into bins. Smoothing techniques include:     - Bin means: Each value in the bin is replaced by the mean value of the bin.     - Bin medians: Each value is replaced by the median value of the bin.     - Bin boundaries: Each value is replaced by the nearest boundary (smallest or largest) within the bin.   + Regression: Data smoothing by fitting data values to a function. Linear regression fits two attributes to predict one value, while multiple linear regression extends this to more attributes.   + Outlier Analysis: Outliers can be detected through clustering, where similar values are grouped into clusters. Values that fall outside the clusters can be considered outliers. * Discretization and Data Reduction: Smoothing methods like binning help reduce the number of distinct values in the data, which is useful for data mining methods like decision tree induction. |
| --- |

# 

## 3.2.3 Data Cleaning as a Process

| **1. Discrepancy detection**   * **This is the first step in the data cleaning process, aiming to identify discrepancies caused by data entry errors, intentional errors, outdated data, inconsistent data representations, and other system errors.** * **How can we proceed with discrepancy detection ?** * **Use any knowledge you may already have regarding properties of the data or metadata:** This is where we can make use of the knowledge we gained about our data. For example, what are the data type and domain of each attribute? What are the acceptable values for each attribute? The basic statistical data descriptions are useful here to grasp data trends and identify anomalies. For example, find the mean, median, and mode values. Are the data symmetric or skewed? What is the range of values? Do all values fall within the expected range? What is the standard deviation of each attribute?... * **Rule Checking:**   **unique rules** (each attribute value must be different)  **consecutive rules** (no missing values between the lowest and highest values)  **null rules** (specifies the use of blanks, question marks, special characters, or other strings that may indicate the null condition)   * **Using Supporting Tools:** There are a number of different commercial tools that can aid in the discrepancy detection step.   **2. Data transformations**   * Some data inconsistencies may be corrected manually using external references. For example, errors made at data entry may be corrected by performing a paper trace. * **Most errors, however, will require data transformations**. * **Commercial tools can assist in the data transformation step**. * **Data migration tools:** are designed to move data from one system or environment to another. This process is crucial during system upgrades, cloud migrations, or when consolidating data from multiple sources. Some tools: AWS DataSync, Microsoft Azure Data Factory, Talend * **ETL (extraction/transformation/loading) tools:** are used to extract data from various sources, transform it into a suitable format, and load it into a target system, such as a data warehouse. These tools are essential for data integration and preparation for analysis. Some tools: Informatica PowerCenter, Talend Open Studio, Apache Hadoop   **1. Phát hiện sự sai lệch**   * **Đây là bước đầu tiên của quy trình làm sạch dữ liệu, nhằm tìm ra các sai lệch do lỗi nhập liệu, lỗi cố ý, dữ liệu lỗi thời, biểu diễn dữ liệu không nhất quán, và các lỗi hệ thống khác.** * **Làm thế nào để chúng ta tiến hành phát hiện sai lệch ?** * **Sử dụng bất kỳ kiến ​​thức nào bạn có thể đã có về các thuộc tính của dữ liệu hoặc siêu dữ liệu:** Đây là lúc chúng ta có thể sử dụng kiến thức mà chúng ta đã học về dữ liệu trong Chương 2. Ví dụ, loại dữ liệu và miền giá trị của mỗi thuộc tính là gì? Giá trị chấp nhận được cho mỗi thuộc tính là gì? Các mô tả dữ liệu thống kê cơ bản đã được thảo luận trong Mục 2.2 rất hữu ích ở đây để nắm bắt xu hướng dữ liệu và xác định các dị thường. Ví dụ, tìm giá trị trung bình, trung vị, và mode. Dữ liệu có đối xứng hay lệch không? Phạm vi giá trị là gì? Tất cả các giá trị có nằm trong phạm vi mong đợi không? Độ lệch chuẩn của mỗi thuộc tính là gì?...   *Ví dụ: Kiểm tra thông tin sản phẩm trong hệ thống quản lý kho hàng*  *Mô tả tình huống:*   * *Một công ty sử dụng hệ thống quản lý kho hàng để theo dõi các sản phẩm trong kho. Cơ sở dữ liệu chứa các thuộc tính cho mỗi sản phẩm như "Mã sản phẩm", "Tên sản phẩm", "Số lượng tồn kho", "Giá bán", và "Ngày hết hạn".*   *1. Sử dụng kiến thức về các thuộc tính:*   * *Mã sản phẩm: Thuộc tính "Mã sản phẩm" phải là một chuỗi ký tự duy nhất, không được trùng lặp trong cơ sở dữ liệu. Nếu có hai sản phẩm có cùng "Mã sản phẩm", điều này có thể cho thấy dữ liệu đã bị nhập sai hoặc trùng lặp.* * *Giá bán: Thuộc tính "Giá bán" phải là một số dương. Nếu có giá trị âm (ví dụ: "-100") hoặc giá trị bằng 0 cho một sản phẩm có trong kho, điều này có thể là sai lệch do lỗi nhập liệu.* * *Số lượng tồn kho: Thuộc tính "Số lượng tồn kho" phải là một số nguyên không âm. Nếu xuất hiện các giá trị như "-10" hoặc "3.5", đây có thể là lỗi dữ liệu.*   *2. Sử dụng kiến thức về siêu dữ liệu (metadata):*   * *Kiểm tra giá trị hợp lệ của "Ngày hết hạn": Siêu dữ liệu cho biết rằng "Ngày hết hạn" phải là một ngày trong tương lai đối với các sản phẩm có hạn sử dụng. Nếu có sản phẩm có "Ngày hết hạn" nằm trong quá khứ (ví dụ: "01/01/2023" khi hiện tại là tháng 9 năm 2024), đây là một sai lệch cần được kiểm tra.* * *Ràng buộc tham chiếu với bảng nhà cung cấp: Siêu dữ liệu cho biết mỗi sản phẩm phải có một "Mã nhà cung cấp" hợp lệ tham chiếu đến bảng "Nhà cung cấp". Nếu có sản phẩm có "Mã nhà cung cấp" là "NV999", nhưng mã này không tồn tại trong bảng "Nhà cung cấp", đây là một sai lệch về tính toàn vẹn tham chiếu.*      * **Kiểm tra các quy tắc duy nhất, quy tắc liên tiếp, và quy tắc null:**   **quy tắc duy nhất** (giá trị của mỗi thuộc tính phải khác nhau)  *Ví dụ: Trong một cơ sở dữ liệu khách hàng, thuộc tính "Mã khách hàng" phải có giá trị duy nhất cho mỗi khách hàng. Không được có hai khách hàng có cùng một mã.*    **quy tắc liên tiếp** (Một quy tắc liên tiếp nói rằng không có giá trị nào bị thiếu giữa các giá trị thấp nhất và cao nhất cho thuộc tính, và tất cả các giá trị cũng phải duy nhất )  *Ví dụ: Trong một rạp chiếu phim, các ghế ngồi được đánh số liên tiếp từ 1 đến N trong mỗi hàng. Quy tắc liên tiếp yêu cầu rằng không có số ghế nào bị bỏ qua hoặc bị thiếu*.    **quy tắc null** (chỉ định việc sử dụng khoảng trắng, dấu chấm hỏi, ký tự đặc biệt, hoặc các chuỗi khác có thể biểu thị tình trạng null  *Ví dụ: Nếu thuộc tính "Số điện thoại" không có giá trị, hãy sử dụng một ký tự đặc biệt như "NA" (Not Available) hoặc "?" để biểu thị giá trị bị thiếu.*     * **Sử dụng công cụ hỗ trợ:** Có một số công cụ thương mại khác nhau có thể hỗ trợ trong bước phát hiện sai lệch. Các công cụ này dựa vào kỹ thuật phân tích cú pháp và khớp mờ khi làm sạch dữ liệu từ nhiều nguồn khác nhau. Các công cụ kiểm tra dữ liệu tìm kiếm các sai lệch bằng cách phân tích dữ liệu để khám phá các quy tắc và mối quan hệ, và phát hiện dữ liệu vi phạm các điều kiện như vậy. Chúng là các biến thể của các công cụ khai thác dữ liệu. Ví dụ, chúng có thể sử dụng phân tích thống kê để tìm mối tương quan, hoặc phân cụm để xác định các giá trị ngoại lệ.   **2. Biến đổi dữ liệu**   * Một số sự không nhất quán trong dữ liệu có thể được sửa chữa thủ công bằng cách sử dụng các tài liệu tham khảo bên ngoài. Ví dụ, các lỗi xảy ra trong quá trình nhập dữ liệu có thể được sửa chữa bằng cách kiểm tra lại các tài liệu gốc. Tuy nhiên, hầu hết các lỗi sẽ cần đến các bước biến đổi dữ liệu. * Các công cụ thương mại có thể hỗ trợ trong bước biến đổi dữ liệu này. * Công cụ di chuyển dữ liệu (Data Migration Tools): Các công cụ này được thiết kế để di chuyển dữ liệu từ hệ thống hoặc môi trường này sang hệ thống hoặc môi trường khác. Quá trình này rất quan trọng trong các trường hợp nâng cấp hệ thống, di chuyển dữ liệu lên đám mây, hoặc khi hợp nhất dữ liệu từ nhiều nguồn khác nhau. Một số công cụ phổ biến bao gồm: * AWS DataSync: * Microsoft Azure Data Factory: * Talend * Công cụ ETL (Extract, Transform, Load): Các công cụ ETL được sử dụng để trích xuất dữ liệu từ nhiều nguồn khác nhau, biến đổi dữ liệu thành định dạng phù hợp, và tải dữ liệu vào hệ thống đích, chẳng hạn như kho dữ liệu. Những công cụ này rất cần thiết cho việc tích hợp dữ liệu và chuẩn bị dữ liệu cho các phân tích. Một số công cụ phổ biến bao gồm: * Informatica PowerCenter * Apache Hadoop: |
| --- |
|  |

Sorted data for price (in dollars): 4, 6, 7, 10, 13, 14, 14, 17, 20, 22, 26, 35

| Partition into (equal-frequency) bins:  Bin 1: 4, 6, 7, 10  Bin 2: 13, 14, 14, 17  Bin 3: 20, 22, 26, 35 |
| --- |