COLLECTING DATA

+ Is the data correct and sufficient? Collecting: Garbage in -> Garbage out

+ Ways to collect data:

    1. Company, organization: ok use it but nó có thể sai, thiếu, lỗi thời -> cần tìm hiểu về quá trình crawling (nguồn gốc), và ngữ cảnh của dữ liệu

    2. Out there (online): isValid?, noise, có chứa dữ liệu rác và bản quyền

    3. Not yet available:

        + Pros: hiểu được ngữ cảnh

        + Cons: Tốn thời gian và tiền bạc

+ Privacy: kiểm tra "robots.txt" file để xem có được phép crawling data?, not advisable to send too many request to the site in a short time.

+ Sử dụng web API 'more official" than parse HTML. JSON đơn giản và dễ dàng khai thác hơn XML

BASIC STATISTICS

+ Population: all the members, Parameters

    - Issues: nếu tập dữ liệu là population -> không cần thiết vì không có tính phổ quát, không có giá trị bởi lẽ quá đầy đủ. Quá lớn để có thể tính toán

+ Sample: subset drawn -> random sample, Statistics

    - Issues: Từ những phân tích của tập sample mà suy ra cho tập population (suy diễn 1-1) -> đơn giản, dễ sai vì sample không đại diện tất cả cho population

→ lấy mẫu thường là phương pháp hiệu quả và tiết kiệm hơn. Áp dụng các phương pháp lấy mẫu chính xác có thể giảm số lượng người tham gia cần thiết, do đó tiết kiệm thời gian và tài nguyên.

+ Inference: rút ra kết luận về tham số của một quần thể, tương đương với toàn bộ quần thể, dựa trên thông tin thu được từ một mẫu.

→ Suy luận dựa trên ba khía cạnh cơ bản của thống kê: lý thuyết xác suất, phân bố mẫu và suy luận thống kê.

+ Sampling

- Random: là phương pháp lý tưởng nhất nhưng khó thực hiện, xác xuất được chọn là như nhau, unbias bởi vì là random và same probability
- Representative: lấy mẫu ở các thuộc tính cụ thể với số lượng là bằng nhau nhưng bao nhiêu thuộc tính là đủ?, bias bởi vì dựa trên kinh nghiệm mà lấy mẫu, lấy mẫu ở một thời điểm.
- Stratified: giống như representative nhưng lấy mẫu theo một tỉ lệ -> đảm bảo mẫu đại diện cho sự đa dạng và cho phép phân tích theo nhóm phụ, bias bởi dựa trên kinh nghiệm lấy mẫu

- Convenience: lấy mẫu dựa trên sự thuận tiện, tính sẵn có (close to hand), bias bởi vì các mẫu thu nhập không đại diện cho quần thể được quan tâm
- Snowball: các cá nhân hoặc mục được chọn dựa trên giới thiệu từ những người khác trong mẫu -> hữu dụng khi dân số khó tiếp cận, một số mẫu hiếm. Bias bởi vì không phải random và những người tham gia có thể giới thiệu những người trùng nhau
- Lỗi lấy mẫu phát sinh khi một mẫu không phải là mẫu "đại diện" tốt nhất. Những lỗi này xuất hiện tự nhiên do các mẫu luôn phụ thuộc vào biến động ngẫu nhiên, và luôn có thể có một mức độ không chắc chắn nào đó. Tuy nhiên, mức độ của những lỗi này phụ thuộc vào kích thước mẫu và độ biến động của tổng thể. Nói chung, các tổng thể nhỏ hơn có khả năng tạo ra lỗi lấy mẫu cao hơn.
- Sai lệch lấy mẫu xảy ra do sự sai lệch có hệ thống khỏi tính ngẫu nhiên trong quá trình lựa chọn mẫu. Nói đơn giản, khi một số phần tử mẫu có khả năng được chọn cao hơn những phần tử khác hoặc khi phương pháp lấy mẫu có sai sót. Một số ví dụ về sai lệch lấy mẫu bao gồm sai lệch lựa chọn, sai lệch đo lường và sai lệch đáp ứng. Những sai lệch này có thể làm giảm độ chính xác của phân tích vì chúng có thể dẫn đến việc đại diện quá mức hoặc không đủ của một số đặc điểm hoặc nhóm phụ trong mẫu.

+ Type of data:

  - Ratio: sắp xếp, khoảng cách bằng nhau, include zero

  - Interval: sắp xếp, khoảng cách bằng nhau

+ Independent variables: can change -> (direct effect) dependent variables

+ Research design:

  - Experimental design: divides different groups -> compares the groups on one or more variables of interest.

  - Quasi-experimental research design: a experiment occurs outside of the lab, in a naturally occurring setting.

  - Correlational research designs: determine how strongly different variables are related to each other

+ Distribution: Dự báo được tương lai, kiểm tra được hiện trạng của dữ liệu.

  - Central tendency: mean, median and mode

  - Variability: range, variable and std

+ Mean, Median, Mode (No Mode, Single, Mutimodal): được dùng để kiểm tra skewed of data (hình dạng của tập dữ liệu)

  - Negative direction: mean < median < mode

  - Normal: mean = median = mode

  - Positive direction: mean > median > mode

+ Outlier: cần viết dòng này nếu tập dữ liệu không theo phân phối chuẩn 'Giả định phân phối chuẩn' -> 1.5IQR

→ Median có giá trị hơn mean bởi vì outliers ảnh hưởng rất nhiều tới giá trị mean

+ Range: max_value - min_value -> xác định được khoảng cách giữa giá trị lớn nhất và bé nhất

+ Midrange: (max_value + min_value) / 2

+ IQR = Q3 - Q1

+ Dùng độ lệch chuẩn thay cho phương sai: Một trong những lý do là độ lệch chuẩn được biểu thị bằng cùng đơn vị với giá trị trung bình, trong khi phương sai được biểu thị bằng đơn vị bình phương → dễ hiểu và trực quan hơn nhiều so với phương sai.

+ Low variance là lý tưởng bởi vì nó cho phép bạn dự đoán thông tin về tổng thể tốt hơn dựa trên dữ liệu mẫu. High varicance có nghĩa là các giá trị ít nhất quán hơn, do đó khó dự đoán hơn.

DATA VISUALIZATION

+ Features:

- Indicators: most important information
- Simplicity: clear information -> understand at hand immediately
- Brevity: message is short and clear, no unnecessary information
- Originality: types of data -> offers readers a new perspective
- Colour: most important pieces, clear and easy to understand.

+ Comparison plots: multiple variables, variables over time

- Line chart: visualizing trends, smaller time periods, vertical bar charts -> better choice -> avoid too many lines, adjust the scale.
- Bar chart: vertical bar charts: so sánh với số lượng biến ít; horizontal: số lượng biến nhiều, tên biến dài -> khác với histogram bởi đây bar là so sánh các biến còn histogram là sự phân bố của một biến
- Radar charts: at the same scale, nhìn phát biết ngay -> display ten factors or fewer on one radar chart -> easier to read. With multiple variables -> multiple radar

+ Relation plots: relationships among variables

- Scatter: whether a correlation exists, multiple groups using different colors
- Bubble: scatter and a third numerical variable -> size of the dots
- Correlogram: scatter and histograms (diagonals)
- Heatmap: visualizing multivariate data, correlation between variables.

+ Composition plots: a part of a whole

- Pie chart: not compare bởi vì so sánh giữa các slice rất khó.
- Donut: more space-efficient, further devide groups into sub-groups
- Stacked bar: compare total across each bar or show a percentage each group -> 100% easier to see relative differences in each groups
- Stacked area: trends for part of a whole relationship
- Venn: logical relations, circle size -> importance of a group, intersection

+ Distribution plots: a single numerical variable.

- Histogram: xác định ở đâu giá trị tập trung nhiều nhất, dễ xác định outliers
- Density: a variation of a histogram but smoother distributions, is better when the distribution shape for histograms heavily depends on the number of bins
- Box plot: multiple statistical measurements, show data outliers, compare for multiple variables or groups.
- Violin plot: combination of box plots and density plots

+ Geo plots:

- Dot map: same size and value, not meant to be counted, use different color to show multiple categories
- Choropleth Map: colored to encode a variable -> good way to show how a variable varies across a geographic area
- Connection map: a certain number of connections between two locations

DATA PREPROCESSING

+ Why need?

- Nó cải thiện độ chính xác và độ tin cậy. Xử lý dữ liệu trước giúp loại bỏ các giá trị dữ liệu bị thiếu hoặc không nhất quán do lỗi của con người hoặc máy tính, điều này có thể cải thiện độ chính xác và chất lượng của một tập dữ liệu, làm cho nó trở nên đáng tin cậy hơn.
- Nó làm cho dữ liệu nhất quán. Khi thu thập dữ liệu, có thể có các bản sao dữ liệu và việc loại bỏ chúng trong quá trình xử lý trước có thể đảm bảo các giá trị dữ liệu để phân tích nhất quán, giúp tạo ra kết quả chính xác.
- Nó tăng khả năng đọc của thuật toán dữ liệu. Xử lý trước nâng cao chất lượng dữ liệu và giúp các thuật toán học máy dễ dàng đọc, sử dụng và diễn giải dữ liệu hơn.
  ➔ Right decision based on accurate data.

+ Why is preparing data so urgent and time-consuming?

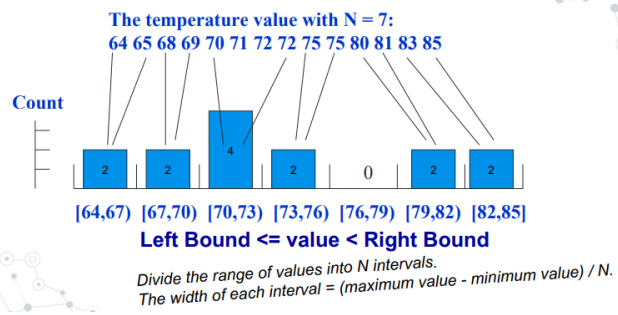
- Cấp bách:
  - Cần ra quyết định kịp thời dựa trên dữ liệu.
  - Ứng dụng thời gian thực đòi hỏi dữ liệu chính xác ngay lập tức.
  - Chuẩn bị chậm khiến tổ chức mất lợi thế cạnh tranh.
- Mất thời gian:
  - Dữ liệu khổng lồ, phức tạp, nhiều định dạng.
  - Dữ liệu thực tế thường lộn xộn, nhiều lỗi, thiếu giá trị.
  - Chuyển đổi, xử lý dữ liệu phức tạp, cần chuyên gia.
  - Cần hiểu biết chuyên môn về lĩnh vực phân tích.

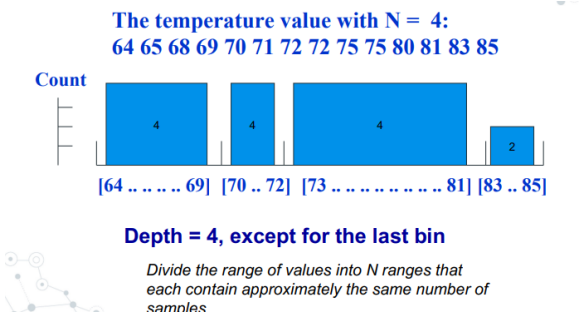
+ Data Cleaning

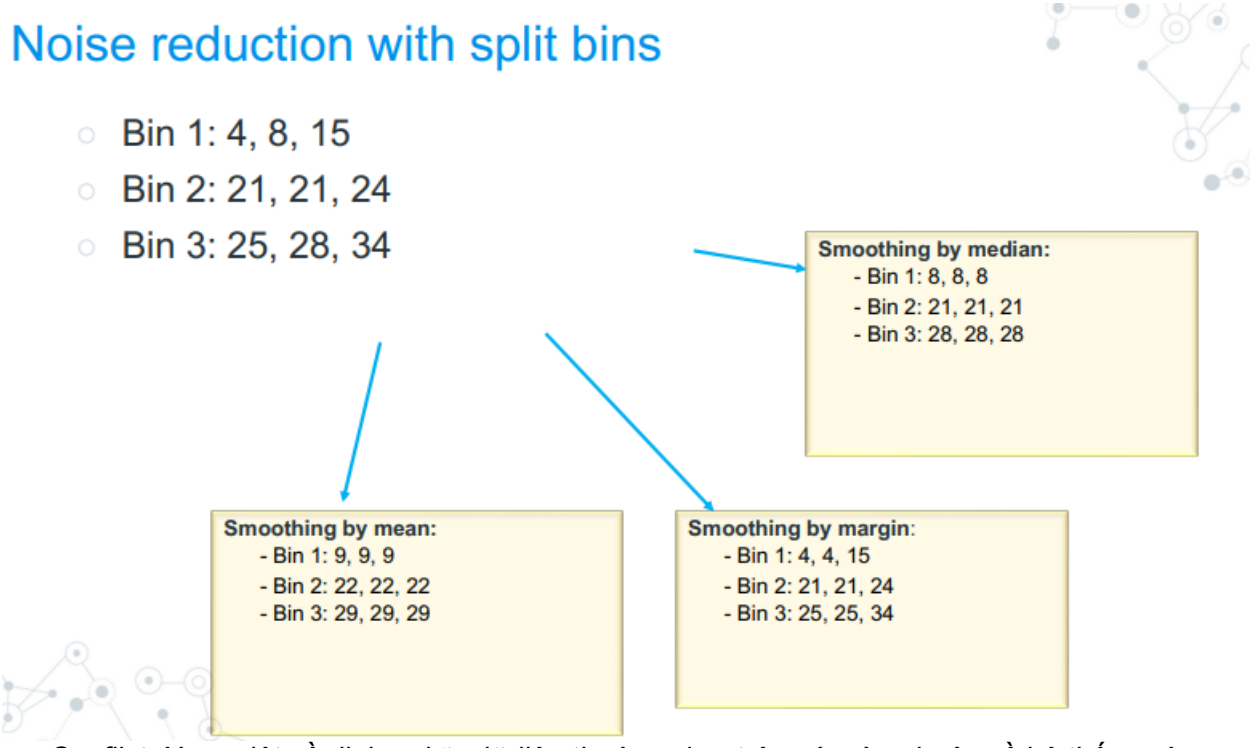
- Why? đảm bảo bạn có dữ liệu chất lượng cao nhất. Điều này không chỉ ngăn ngừa sai sót mà còn ngăn ngừa sự thất vọng của khách hàng và nhân viên, tăng năng suất và cải thiện phân tích dữ liệu và ra quyết định.
- Missing value: Không có phương pháp nào đúng, căn bản việc điền dữ liệu là đã sai
  - Xóa: phổ biến nhưng không hiệu quả, đặc biệt nếu tỉ lệ missing value cao -> gây mất dữ liệu.
  - Thu thập lại: tốn thời gian và tiền bạc
  - Fill manually: vô ích và không khả thi

- Fill automatically: a common constant or statistic (mean, median, mode)

- Noise reduction: có thể dẫn đến dự đoán sai, mô hình thiên vị và những hiểu biết sai lầm -> xác định nhờ 1.5IQR, range, quá trình preprocessing
  - Regression: is used to fit an equation to a dataset
  - Clustering: Groups are formed from the data having similar value.
  - Binning method:
    - Equal-width: not good for skewed data
    - Equal-depth: Không có sự nhất quán -> chung giá trị ở 2 bins -> điểm kì dị







- Conflict: Xung đột về định nghĩa dữ liệu thường dựa trên các ràng buộc về hệ thống và quy trình trong quá khứ. Three conflict patterns:
  - Different Context: (What this often sounds like in an organization.) “That definition is close. But for our department, we mean ABC.”
  - Overloaded Terms: “There are two types of this thing … except for this other thing, which is sort of a hybrid of the other two.”
  - Name Conflicts: “That’s not what that term means at all. It means XYZ.”

+ Data Integration:

- Why:
  - Thấu hiểu toàn diện: Bằng cách kết hợp dữ liệu từ nhiều nguồn, bạn có được cái nhìn toàn diện và chính xác hơn về doanh nghiệp, khách hàng hoặc lĩnh vực nghiên cứu của mình.
  - Phát hiện ra các mối liên hệ và xu hướng ẩn: Dữ liệu tích hợp tiết lộ các mối liên hệ và xu hướng có thể vô hình trong các tập dữ liệu riêng lẻ, cho phép ra quyết định và dự đoán tốt hơn.
  - Chất lượng dữ liệu được cải thiện: Quá trình lựa chọn bao gồm việc xác định và giải quyết các mâu thuẫn, lỗi và sự dư thừa, đảm bảo phân tích đáng tin cậy hơn.
  - Hiệu quả tăng cường: Tích hợp dữ liệu vào một kho lưu trữ trung tâm giúp hợp lý hóa việc truy cập, quản lý và phân tích, tiết kiệm thời gian và tài nguyên.
  - Tuân thủ và quản trị: Lựa chọn và tích hợp dữ liệu hiệu quả giúp các tổ chức đáp ứng các yêu cầu về quy định và duy trì kiểm soát đối với thông tin nhạy cảm.
- Quy trình lựa chọn dữ liệu:
  - Xác định mục tiêu: Nêu rõ ràng các mục tiêu cụ thể bạn muốn đạt được với dữ liệu.
  - Xác định nguồn dữ liệu liên quan: Xác định nguồn dữ liệu bên trong và bên ngoài nào chứa dữ liệu cần thiết.
  - Đánh giá chất lượng dữ liệu: Đánh giá mức độ chính xác, đầy đủ, nhất quán và liên quan của các nguồn dữ liệu tiềm năng.
  - Lựa chọn dữ liệu: Chọn các phần tử dữ liệu cụ thể phù hợp với mục tiêu của bạn và đáp ứng các tiêu chuẩn chất lượng.
  - Làm sạch và xử lý trước dữ liệu: Giải quyết bất kỳ mâu thuẫn, lỗi hoặc giá trị nào bị thiếu để đảm bảo tính toàn vẹn của dữ liệu.
  - Tích hợp dữ liệu: Kết hợp dữ liệu từ nhiều nguồn thành một định dạng thống nhất, thường sử dụng các công cụ và kỹ thuật tích hợp dữ liệu.
  - Xác nhận dữ liệu tích hợp: Kiểm tra kỹ lưỡng về độ chính xác và nhất quán trong tập dữ liệu tích hợp.
  - Ghi chép quá trình lựa chọn dữ liệu: Ghi lại các quyết định và quy trình để tham khảo và bảo trì trong tương lai.
- Mistakes:
  - Chia sẻ dữ liệu không an toàn: việc chia sẻ quá mức và rò rỉ dữ liệu
  - Dữ liệu trùng lặp: tạo nên những hố dữ liệu cồng kềnh và tốn kém
  - Phương pháp quy trình kém: dữ liệu chỉ hữu ích cho mức độ câu hỏi được đặt ra cho nó.
  - Dữ liệu không đầy đủ: thu thập dữ liệu cần thiết để truy vấn có thể khó khăn hoặc yêu cầu các quy trình ETL kéo dài.
  - Dữ liệu di chuyển với tốc độ khác nhau: tốc độ của luồng dữ liệu, có thể thay đổi đáng kể tùy thuộc vào lưu trữ, quyền truy cập vào dữ liệu riêng biệt và cách dữ liệu nhập vào data lake được xử lý.
- Process: metadata is the contextual information that helps you understand raw data. Metadata helps you make data discoverable, accessible, trustworthy, and valuable.
  - Schema matching: metadata
  - Eliminate redundant and duplicate: repeated records and correlation analysis
  - Resolve inconsistencies: weight is measured in kilograms or pounds.
- Extract, transform, and load (ETL) is the process of combining data from multiple sources into a large, central repository called a data warehouse. ETL uses a set of business rules to

clean and organize raw data and prepare it for storage, data analytics, and machine learning (ML).

+ Data reduction:

- Why:
  - Tiết kiệm dung lượng lưu trữ.
  - Tăng tốc xử lý, phân tích, truy xuất dữ liệu.
  - Dễ dàng quản lý, chia sẻ dữ liệu.
  - Tuân thủ quy định bảo mật, riêng tư.
  - Cải thiện chất lượng dữ liệu.
  - đơn giản hóa mô hình học máy, giảm chi phí mô hình hóa và giảm thời gian xử lý và huấn luyện.
  - Việc có quá nhiều biến hoặc thuộc tính dữ liệu trong một bộ dữ liệu có thể làm nghẹt tốc độ xử lý song song nhanh chóng của một thuật toán học máy.
- Giảm dữ liệu có thể mất thông tin?
  - Đè nén dữ liệu ảnh, âm thanh làm giảm chất lượng.
  - Giảm chiều dữ liệu có thể loại bỏ thông tin không cần thiết.
  - Tổng hợp dữ liệu có thể bỏ qua chi tiết cá nhân.
  - Lọc dữ liệu có thể loại bỏ thông tin quý giá.
- Làm thế nào để giảm thiểu mất thông tin?
  - Chọn phương pháp phù hợp với mục tiêu phân tích và mức độ mất thông tin chấp nhận được.
  - Đánh giá và điều chỉnh phương pháp nếu cần.
  - Giữ bản sao dữ liệu gốc.
  - Theo dõi quá trình giảm dữ liệu.
  - Ghi chép chi tiết quá trình giảm dữ liệu, bao gồm phương pháp và rủi ro mất thông tin.
  - Thường xuyên đánh giá tác động của việc giảm dữ liệu đến kết quả phân tích và điều chỉnh phương pháp nếu cần.
- Method:
  - Aggregation: tổng hợp dữ liệu cấp thấp thành dữ liệu cấp cao -> giảm chiều, tang sự thú vị cho mẫu
  - Dimensionality reduction: remove irrelevant attribute -> giảm thời gian xử lý và huấn luyện cho mô hình học máy
  - Data compression: giảm phần cứng lưu trữ, thời gian truyền dữ liệu và bang thông liên lạc – Waveket transforms, PCA
  - Numerosity Reduction: thay thế dữ liệu gốc bằng một dạng dữ liệu nhỏ hơn đại diện cho dữ liệu gốc với ít nhiều độ trung thực.
    - Regression model
    - Histogram: Divide the data into bins and the height of the column is the number of objects in each bin. Store only the average of each bin
  - Clustering: Thực hành này sử dụng các thuộc tính dữ liệu để xây dựng một tập hợp các cụm trong đó dữ liệu được phân chia. Các điểm tương đồng và khác biệt giữa các đối tượng dữ liệu dẫn đến vị trí và khoảng cách khác nhau giữa các cụm và các đối tượng nói chung.
  - Sampling: Use a much smaller random sample set instead of a large data set.

        - Trong trường hợp SRSWR, tất cả các số ngẫu nhiên đều được chấp nhận ngay cả khi được lặp lại nhiều lần. Trong trường hợp SRSWOR, nếu bất kỳ số ngẫu nhiên nào được lặp lại thì số đó sẽ bị bỏ qua và nhiều số khác sẽ được rút ra.
    - Discretization: Chuyển đổi miền giá trị thuộc tính (liền kề) bằng cách chia miền giá trị thành các khoảng. Lưu nhãn của dãy thay vì giá trị thực. Thích hợp cho dữ liệu số liên tục.
        - One-hot encoding: tang số chiều nhưng giải quyết được vấn đề chuyển đổi số của các cột dữ liệu nominal -> tang khả năng multicollinear (correlation cao)

+ Data transformation:

- Why:
    - Cơ cấu tổ chức tốt hơn: Dữ liệu được chuyển đổi dễ sử dụng hơn cho cả con người và máy tính.
    - Chất lượng dữ liệu được cải thiện: Dữ liệu xấu tiềm ẩn một số rủi ro. Chuyển đổi dữ liệu có thể giúp tổ chức của bạn loại bỏ các vấn đề về chất lượng và giảm khả năng giải thích sai.
    - Truy vấn nhanh hơn: Bằng cách chuẩn hóa dữ liệu và lưu trữ hợp lý trong nhà kho, tốc độ truy vấn và công cụ BI có thể được tối ưu hóa - dẫn đến giảm ma sát cho phân tích.
    - Quản lý dữ liệu đơn giản hơn: Một phần lớn của việc chuyển đổi dữ liệu là theo dõi siêu dữ liệu và nguồn gốc.
    - Sử dụng rộng rãi hơn: Chuyển đổi giúp bạn tận dụng tối đa dữ liệu của mình bằng cách chuẩn hóa và làm cho nó dễ sử dụng hơn.
- Method
    - Smoothing: the process of removing noise from the data.
        - Binning age values (e.g., 20-25, 26-30, 31-35) to reduce variability and create smoother trends.
    - Integration: summarizing or integrating data.
        - Merging customer data from different databases into a single customer relationship management (CRM) system.
    - Generalization: replacing low-level concepts with high-level concepts.
        - Grouping products into categories (e.g., electronics, clothing, books) instead of individual items.
    - Normalization: attribute data should be returned to a small range of values like 0 to 1.
        - Normalizing income data to compare individuals with different salary ranges.
    - Attribute construction: new properties are created and added to a given set of properties.
        - Calculating BMI (body mass index) from height and weight attributes.

BIG DATA

+ What is? Massive volume -> difficult to process using traditional techniques.

+ Characteristics:

- Volume: Khối lượng đề cập đến lượng dữ liệu bạn có
- Variety: Kiểu dữ liệu đề cập đến các loại dữ liệu lớn khác nhau. Đây là một trong những vấn đề lớn nhất mà ngành big data phải đối mặt vì nó ảnh hưởng đến hiệu suất. Việc quản lý đúng đắn sự đa dạng của dữ liệu bằng cách sắp xếp nó là rất quan trọng.

- Velocity: Tốc độ đề cập đến tốc độ xử lý dữ liệuTốc độ đề cập đến tốc độ xử lý dữ liệu. Tốc độ cao rất quan trọng cho hiệu suất của bất kỳ quy trình big data nào.
- Veracity: Tính chính xác đề cập đến độ chính xác của dữ liệu của bạn. Đây là một trong những đặc điểm quan trọng nhất của Big Data vì tính chính xác thấp có thể ảnh hưởng nghiêm trọng đến độ chính xác của kết quả của bạn.
- Value: Giá trị đề cập đến những lợi ích mà tổ chức của bạn thu được từ dữ liệu. Nó có phù hợp với các mục tiêu của tổ chức bạn không?
- Variability: số điểm không nhất quán trong dữ liệu. Tính biến đổi cũng có thể đề cập đến tốc độ tải dữ liệu lớn vào cơ sở dữ liệu của bạn không nhất quán.

+ Four components:

- Ingestion (collecting and preparing the data):
  - Thu thập và chuẩn bị dữ liệu bằng quy trình ETL (Extract, Transform, Load).
  - Xác định nguồn dữ liệu.
  - Quyết định thu thập dữ liệu theo đợt hay truyền trực tiếp.
  - Làm sạch, chỉnh sửa và tổ chức dữ liệu.
- Storage (storing the data):
  - Thực hiện bước cuối cùng của ETL: tải dữ liệu.
  - Lưu trữ dữ liệu trong kho dữ liệu hoặc kho dữ liệu lake, tùy theo nhu cầu.
- Analysis (analyzing the data)
  - Phân tích dữ liệu để tạo ra thông tin giá trị cho tổ chức.
  - Bốn loại phân tích dữ liệu lớn: dự báo, dự đoán, mô tả và chẩn đoán.
  - Sử dụng trí tuệ nhân tạo và thuật toán học máy để phân tích dữ liệu.
- Consumption (presenting and sharing the insights):
  - Chia sẻ thông tin sau phân tích với đối tượng không chuyên như các bên liên quan và quản lý dự án.
  - Sử dụng trực quan hóa dữ liệu và kể chuyện dữ liệu để truyền đạt hiệu quả.

+ Advantage:

- Ra quyết định sáng suốt hơn: Dữ liệu lớn giúp các doanh nghiệp và tổ chức đưa ra những quyết định sáng suốt hơn trong thời gian ngắn hơn. Dữ liệu lớn có thể xác định các xu hướng và mô hình mà nếu không có, các công ty sẽ bỏ lỡ, giúp họ tránh được sai lầm.
- Dịch vụ khách hàng dựa trên dữ liệu: Một tác động lớn khác của dữ liệu lớn đối với tất cả các ngành là ở bộ phận dịch vụ khách hàng.
- Tối ưu hóa hiệu quả: Các tổ chức sử dụng dữ liệu lớn để xác định những điểm yếu hiện có trong nội bộ. Ví dụ, Dữ liệu lớn đã giúp ngành sản xuất cải thiện đáng kể hiệu quả thông qua IoT và robot.
- Ra quyết định theo thời gian thực: Dữ liệu lớn đã thay đổi nhiều lĩnh vực bằng cách cho phép theo dõi theo thời gian thực, chẳng hạn như quản lý hàng tồn kho, tối ưu hóa chuỗi cung ứng, chống rửa tiền và phát hiện gian lận trong ngân hàng & tài chính.

DATA MODELING

+ Before: dựa trên các số liệu thống kê các biểu đồ -> xác định trước một vài nhánh trong ML có thể giải quyết vấn đề

+ After hypothesis:

- Nhiệm vụ của thuật toán học là tìm ra giả thuyết phù hợp nhất cho một vấn đề. -> hãy đưa ra nhiều giả thuyết nhất có thể.
- Học máy = quy trình lặp lại để tìm ra mức tổn thất tối thiểu cho dữ liệu cho trước.

+ After loss function design:

- What parameters to produce the lowest loss rate
- The process to optimize the function

+ Linear regression

◎ Assume that a line is fitted through the points (hypothesis)

$$f(x) = \beta_1 x + \beta_2$$

◎ The loss function is MSE (mean-squares error)

$$E(f) = \frac{1}{n}\sum_{k=1}^{n}(f(x_k) - y_k)^2 = \frac{1}{n}\sum_{k=1}^{n}(\beta_1 x_k + \beta_2 - y_k)^2$$

+ Nonlinear regression

◎ How with nonlinear regresstion? For example:

$$f(x) = \beta_2 \exp(\beta_1 x)$$

◎ The MSE function:

$$E(\beta_1, \beta_2) = \sum_{k=1}^{n}(\beta_2 \exp(\beta_1 x_k) - y_k)^2$$

+ Gradient descent:

- Direction vector: derivative of a function at a specific point gives the slope of the tangent line.
- Why is the tangent line considered as a direction vector?
  - Biểu thị hướng chứ không phải vị trí: Không giống như vectơ vị trí, chỉ định vị trí tuyệt đối của một điểm trong không gian, đường tiếp tuyến không xác định một điểm cụ thể. Thay vào đó, nó chỉ ra hướng chuyển động tại một điểm cụ thể trên đường cong. Hướng có thể được hình dung là góc mà đường cong "nghiêng" tại điểm đó.
  - Tiếp tuyến với đường cong: Đường tiếp tuyến “tiếp xúc” với đường cong tại một điểm và cùng hướng với đường cong tại điểm đó. Điều này làm cho nó thể hiện phù hợp hướng cục bộ dọc theo đường cong tại vị trí cụ thể đó.
  - Dùng trong tính toán: Các phép tính khác nhau liên quan đến đường cong, chẳng hạn như tìm vận tốc hoặc gia tốc trong các bài toán chuyển động, phụ thuộc vào hướng chuyển động. Vectơ chỉ phương của đường tiếp tuyến rất hữu ích làm đầu vào chính cho các phép tính này.
  - Đơn giản hóa việc biểu thị hướng: Trong khi toàn bộ đường cong cung cấp thông tin về đường đi tổng thể, thì đường tiếp tuyến ghi lại hướng tức thời tại một điểm cụ thể. Điều này cho phép phân tích và thao tác thông tin định hướng dễ dàng hơn mà không cần xử lý độ phức tạp của đường cong.

- Nếu bạn đứng tại một điểm nào đó, độ dốc của mặt đất trước mặt bạn sẽ phụ thuộc vào hướng bạn đang hướng tới. Để tính hệ số góc theo hướng nào, ta lấy đạo hàm theo hướng này ⇒ gọi là đạo hàm có hướng (directional derivative)
- Độ dốc (gradient) của f tại bất kỳ điểm nào cho bạn biết: hướng có hướng dốc nhất tính từ điểm đó so với mặt phẳng x, y; độ dốc của nó (độ dốc của ngọn đồi theo hướng đó)
- Gradient descent hoạt động bằng cách di chuyển xuống dưới về phía các điểm lõm hoặc thung lũng trong đồ thị để tìm giá trị nhỏ nhất. Trong mỗi lần lặp lại, gradient descent hạ thấp hàm chi phí theo hướng dốc nhất. Bằng cách điều chỉnh các tham số theo hướng này, nó tìm cách đạt đến giá trị tối thiểu của hàm chi phí và tìm ra các giá trị phù hợp nhất cho các tham số. Kích thước của mỗi bước được xác định bởi tham số α được gọi là Learning rate.
- Gradient cho chúng ta biết hướng đi lên dốc nhất và bằng cách di chuyển theo hướng ngược lại, chúng ta có thể tìm ra hướng đi xuống dốc nhất.
- Với gradient descent, chúng ta không bao giờ thực sự đạt được cực trị, mà chỉ đơn giản là tiếp cận nó dần dần. Vậy, tại sao chúng ta lại chọn gradient descent?

  Ưu điểm:
  - Tính linh hoạt: Chúng ta chỉ có thể giải hệ phương trình một cách rõ ràng cho một số ít mô hình. Ngược lại, chúng ta có thể áp dụng gradient descent cho bất kỳ mô hình nào mà chúng ta có thể tính toán gradient. Thường việc này khá dễ dàng và hiệu quả. Quan trọng hơn, nó thường có thể được thực hiện tự động, do đó các gói phần mềm như Theano và TensorFlow có thể giúp chúng ta không bao giờ phải tính toán đạo hàm riêng bằng tay.
  - Hiệu quả tính toán: Giải một hệ phương trình tuyến tính lớn có thể tốn kém, đắt hơn nhiều so với một lần cập nhật gradient descent. Do đó, gradient descent đôi khi có thể tìm ra giải pháp hợp lý nhanh hơn nhiều so với việc giải hệ tuyến tính. Vì vậy, gradient descent thường thực tế hơn so với việc tính toán các giải pháp chính xác, ngay cả đối với các mô hình mà chúng ta có thể suy ra được các giải pháp đó.
  - Khả năng mở rộng: Gradient Descent có thể mở rộng cho các tập dữ liệu lớn vì nó cập nhật các tham số cho từng ví dụ huấn luyện lần lượt.
  - Sự hội tụ: Gradient Descent có thể hội tụ về điểm cực tiểu toàn cục của hàm chi phí, với điều kiện tốc độ học được thiết lập phù hợp.

  Nhược điểm:

  - Cực tiểu cục bộ: Gradient descent có thể hội tụ về cực tiểu cục bộ thay vì cực tiểu toàn cục, đặc biệt nếu hàm chi phí có nhiều đỉnh và thung lũng.
  - Lựa chọn tốc độ học: Việc lựa chọn tốc độ học có thể ảnh hưởng đáng kể đến hiệu suất của gradient descent. Nếu tốc độ học quá cao, thuật toán có thể vượt quá mức tối thiểu và nếu nó quá thấp, thuật toán có thể mất quá nhiều thời gian để hội tụ.
  - Quá khớp: Gradient descent có thể quá khớp với dữ liệu huấn luyện nếu mô hình quá phức tạp hoặc tốc độ học quá cao. Điều này có thể dẫn đến hiệu suất tổng quát kém trên dữ liệu mới.
  - Tốc độ hội tụ: Tốc độ hội tụ của gradient descent có thể chậm đối với các tập dữ liệu lớn hoặc không gian có chiều cao, điều này có thể khiến thuật toán tốn nhiều chi phí tính toán.
  - Điểm yên ngựa: Trong không gian có chiều cao, gradient của hàm chi phí có thể có các điểm yên ngựa, có thể khiến gradient descent bị mắc kẹt trong một vùng cao nguyên thay vì hội tụ về một cực tiểu.

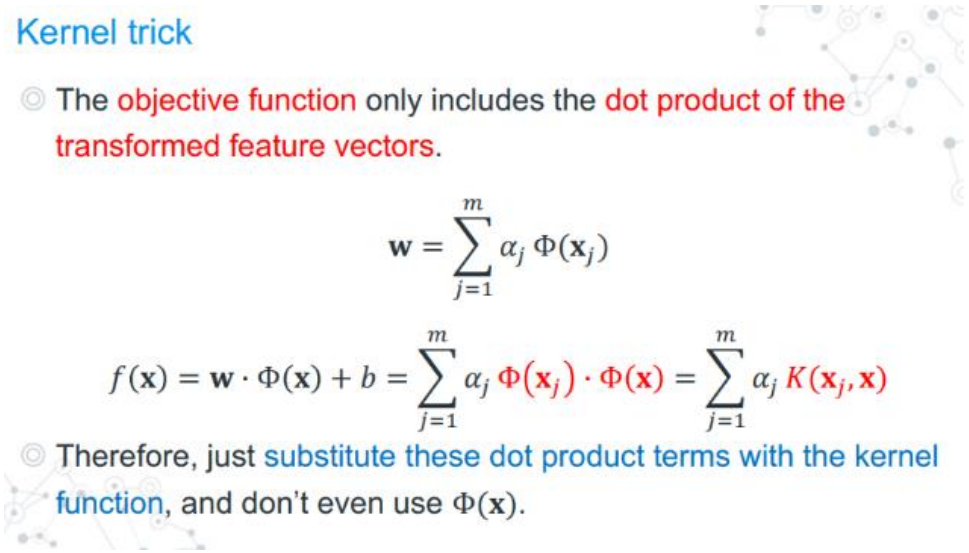
- Issues:
  - Learning rate:
    - Too low: requires many updates before reaching the minimum point
    - Just right: optimal -> reaches minimum point
    - Too large: causes drastic updates -> divergent behaviour (hành vi lệch hướng) -> nhảy qua nhảy lại quanh điểm minimum.
  - Starting point: sẽ dẫn đến rơi vào cực tiểu cục bộ chứ hum phải cực tiểu toàn cục
- Momentum: giúp vượt qua cực tiểu cục bộ
- Over-determined system: more constraints (equations) than unknown variables -> no solutions, approximate solutions to minimizing a given error.
- Under-determined system: more unknowns than constraints -> an infinite number of solutions, some choice of constraint must be made.
- Việc lựa chọn mô hình không đơn giản về việc giảm lỗi, đó là về sản xuất một mô hình có hiệu suất cao mức độ có thể giải thích được, khái quát hóa và dự đoán khả năng.
- Handle overfitting: k-fold validation, feature selection, data augmentation, ensembling

+ Classification:

- Challenge:
  - Ranh giới giữa các kiểu dữ liệu tạo thành một đa tạp phi tuyến tính khó mô tả.
  - Nếu dữ liệu mẫu chỉ thu được một phần của đa tạp, thì gần như chắc chắn nó sẽ thất bại trong việc mô tả dữ liệu tổng thể.
  - Dữ liệu có thể nằm trong không gian chiều cao hơn và việc trực quan hóa về cơ bản là không thể.
- Support Vector Machines
  - SVM (Máy học vectơ hỗ trợ) là một bài toán học máy có giám sát, trong đó chúng ta cố gắng tìm một siêu phẳng phân tách tốt nhất hai lớp dữ liệu. SVM thực hiện điều này bằng cách tìm khoảng cách biên tối đa giữa các siêu phẳng, có nghĩa là khoảng cách lớn nhất giữa hai lớp dữ liệu.
    - Vecto hỗ trợ (support vectors): Đây là những điểm gần nhất với siêu phẳng. Đường phân tách sẽ được xác định dựa trên các điểm dữ liệu này.
    - Khoảng cách biên (margin): Đây là khoảng cách giữa siêu phẳng và các quan sát gần nhất với siêu phẳng (vecto hỗ trợ). Trong SVM, khoảng cách biên lớn được coi là tốt. Có hai loại khoảng cách biên: biên cứng và biên mềm.
  - Siêu phẳng tốt nhất là siêu phẳng có khoảng cách lớn nhất với cả hai lớp dữ liệu, và đây chính là mục tiêu chính của SVM. Để tìm ra siêu phẳng này, SVM sẽ tìm kiếm các siêu phẳng khác nhau có khả năng phân loại tốt nhất các điểm dữ liệu, sau đó chọn ra siêu phẳng có khoảng cách xa nhất với các điểm dữ liệu, hay còn gọi là có biên phân cách lớn nhất.
  - SVM is considered one of the best algorithms because it can handle high-dimensional data, is effective in cases with limited training samples, and can handle non-linear classification using kernel functions.
  - Soft margin: Vi phạm lề biên có nghĩa là chọn một siêu phẳng có thể cho phép một số điểm dữ liệu nằm ở giữa khu vực lề biên hoặc ở phía không chính xác của siêu phẳng. Lỗi SVM bằng Lỗi biên + Lỗi phân loại. Biên càng lớn, lỗi biên càng nhỏ và ngược lại.
  - Nếu ai đó hỏi bạn mô hình nào tốt hơn, mô hình có biên tối đa và có 2 điểm phân loại sai hay mô hình có biên rất nhỏ nhưng tất cả các điểm đều được phân loại chính xác?

➔ Thật ra không có câu trả lời chính xác cho câu hỏi này, nhưng chúng ta có thể sử dụng Lỗi SVM = Lỗi biên + Lỗi phân loại để lý giải. Nếu bạn không muốn bất kỳ lỗi phân loại nào trong mô hình, bạn có thể chọn hình 2. Điều đó có nghĩa là chúng ta sẽ tăng 'c' để giảm Lỗi phân loại, nhưng nếu bạn muốn biên được tối đa hóa thì giá trị của 'c' nên được giảm thiểu. Đó là lý do tại sao 'c' là một siêu tham số và chúng ta tìm giá trị tối ưu của 'c' bằng cách sử dụng GridsearchCV và cross-validation.

- Ưu điểm của SVM
  - Hiệu quả hơn khi dữ liệu tuyến tính: SVM hoạt động tốt hơn khi dữ liệu có thể tách biệt bằng một đường thẳng.
  - Hiệu quả tốt hơn trong không gian đa chiều: SVM có thể xử lý dữ liệu trong không gian nhiều chiều phức tạp.
  - Giải quyết được các vấn đề phức tạp với hàm nhân: Sử dụng hàm nhân, SVM có thể giải quyết những vấn đề mà phân loại tuyến tính thông thường không giải quyết được.
  - Ít nhạy cảm với điểm ngoại lai: SVM ít bị ảnh hưởng bởi những điểm outlier so với một số thuật toán khác.
  - Ứng dụng trong phân loại ảnh: SVM được ứng dụng hiệu quả trong các bài toán phân loại ảnh.
- Nhược điểm của SVM
  - Chọn hàm nhân thích hợp khó khăn: Không dễ dàng để chọn lựa hàm nhân phù hợp cho từng bài toán cụ thể.
  - Hiệu quả trên tập dữ liệu lớn không tốt: SVM có thể không đạt hiệu quả tốt với các tập dữ liệu lớn.
  - Điều chỉnh tham số phức tạp: SVM có hai tham số quan trọng C và gamma cần được điều chỉnh, việc này có thể khó khăn và khó hình dung tác động của chúng.



Summary of the measurements

A\P	C	¬C	
C	TP	FN	P
¬C	FP	TN	N
	P'	N'	All

Measure	Formula
accuracy, recognition rate	$\frac{TP+TN}{P+N}$
error rate, misclassification rate	$\frac{FP+FN}{P+N}$
sensitivity, true positive rate, recall	$\frac{TP}{P}$
specificity, true negative rate	$\frac{TN}{N}$
precision	$\frac{TP}{TP+FP}$
F, F_1, F-score, harmonic mean of precision and recall	$\frac{2 \times precision \times recall}{precision + recall}$
F_β, where β is a non-negative real number	$\frac{(1+\beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$

Bài 1. Cho mẫu cụ thể của biến ngẫu nhiên hai chiều (X,Y) như sau:

X	2	3	4	5	6	7	8	9
Y	3	7	8	9	13	17	16	17

Viết hàm hồi quy tuyến tính mẫu của Y theo X.

Giải.

Ta có $n=8; \sum X_i^2 = 284; \overline{X} = 5,5; \overline{Y} = 11,25; \sum X_i Y_i = 582$.

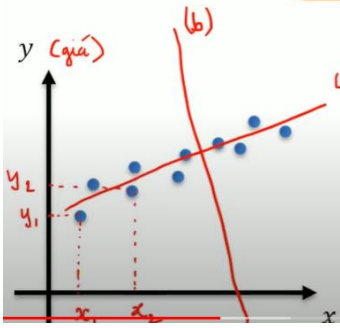
$$\hat{a} = \frac{\sum X_i Y_i - n\overline{X}\overline{Y}}{\sum X_i^2 - n\overline{X}^2} = \frac{582 - 8.5,5.11,25}{284 - 8.5,5^2} = \frac{29}{14} \text{ suy ra } \hat{b} = 11,25 - \frac{29}{14}.5,5 = -\frac{1}{7}.$$

Vậy hàm hồi quy mẫu là: $\hat{Y}_i = \frac{29}{14} X_i - \frac{1}{7}$.

	Population	Estimate Based on a Sample
Variance	$\sigma^2 = \frac{\Sigma(X-\mu)^2}{N}$	$s^2 = \frac{\Sigma(X-\bar{X})^2}{n-1}$
	where σ^2 is population variance, Σ is to sum, X is each score in the distribution, μ is the population mean, N is the number of cases in the population.	where s^2 is sample variance, Σ is to sum, X is each score in the distribution, $\bar{X}$ is the sample mean, n is the number of cases in the sample.
Standard Deviation	$\sigma = \sqrt{\frac{\Sigma(X-\mu)^2}{N}}$	$s = \sqrt{\frac{\Sigma(X-\bar{X})^2}{n-1}}$
	where σ is population standard deviation, Σ is to sum, X is each score in the distribution, μ is the population mean, N is the number of cases in the population.	where s is sample standard deviation, Σ is to sum, X is each score in the distribution, $\bar{X}$ is the sample mean, n is the number of cases in the sample.

Data x → Mô hình máy học $f_\theta(x)$ → dự đoán $\tilde{y} \approx$ dữ liệu thực y

$\mathcal{L}(\theta; x, y)$



Chọn $f_\theta(x_i) = \theta_1 x_i + \theta_0$, khi đó ta cần tìm:

$$\underset{\theta}{\text{agrmin}} \sum_{i=1}^{n} D(f_\theta(x_i), y_i) \cong \underset{\theta_1,\theta_0}{\text{agrmin}} \frac{1}{2n} \sum_{i=1}^{n} (\theta_1 x_i + \theta_0 - y_i)^2$$

Cần tính $\frac{\delta \mathcal{L}}{\delta \theta_1}$ và $\frac{\delta \mathcal{L}}{\delta \theta_0}$

$$\mathcal{L}(\theta_0, \theta_1) \cong \frac{1}{2n} \sum_{i=1}^{n} (\theta_1 x_i + \theta_0 - y_i)^2$$

$$\frac{\delta \mathcal{L}}{\delta \theta_0} = \frac{1}{n} \sum_{i=1}^{n} (\theta_1 x_i + \theta_0 - y_i)$$

$$\frac{\delta \mathcal{L}}{\delta \theta_1} = \frac{1}{n} \sum_{i=1}^{n} (\theta_1 x_i + \theta_0 - y_i) x_i$$

- Tính đạo hàm riêng $\frac{\delta \mathcal{L}}{\delta \theta}$
- Khởi tạo: θ_0, θ_1 ngẫu nhiên, $\alpha, \varepsilon > 0$ đủ nhỏ
- Lặp:
  - $\theta_0 \leftarrow \theta_0 - \alpha \frac{\delta \mathcal{L}}{\delta \theta_0}$
  - $\theta_1 \leftarrow \theta_1 - \alpha \frac{\delta \mathcal{L}}{\delta \theta_1}$
  - Nếu $\left|\frac{\delta \mathcal{L}}{\delta \theta_0}\right| < \varepsilon$ và $\left|\frac{\delta \mathcal{L}}{\delta \theta_1}\right| < \varepsilon$: dừng lặp
- θ là tham số để $\mathcal{L}$ đạt cực tiểu

11:11 / 32:02

Vector hóa công thức

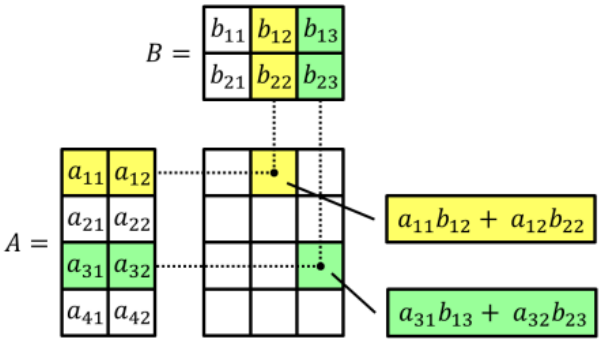
Đặt $\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$, $\bar{\mathbf{x}} = \begin{bmatrix} 1 \\ x \end{bmatrix}$, $X = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \end{bmatrix}$, $Y = [y_1 \quad y_2 \quad \dots \quad y_n]$

$$f_{\boldsymbol{\theta}}(x) = \boldsymbol{\theta}^T \bar{\mathbf{x}}$$

$$\frac{\delta \mathcal{L}}{\delta \theta_0} = \frac{1}{n} \sum_{i=1}^{n} (\theta_1 x_i + \theta_0 - y_i) = \frac{1}{n} X_1 (\boldsymbol{\theta}^T X - Y)^T$$

$$\frac{\delta \mathcal{L}}{\delta \theta_1} = \frac{1}{n} \sum_{i=1}^{n} (\theta_1 x_i + \theta_0 - y_i) x_i = \frac{1}{n} X_2 (\boldsymbol{\theta}^T X - Y)^T$$

- Minh họa tích ma trận AB của hai ma trận A và B:



Gradient descent

Hàm chi phí

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(y_i - (mx_i + b))^2$$

Chúng ta có thể tính toán độ dốc của hàm chi phí này là:

$$f'(m,b) = \begin{bmatrix} \frac{df}{dm} \\ \frac{df}{db} \end{bmatrix} = \begin{bmatrix} \frac{1}{N}\sum -x_i \cdot 2(y_i - (mx_i + b)) \\ \frac{1}{N}\sum -1 \cdot 2(y_i - (mx_i + b)) \end{bmatrix} = \begin{bmatrix} \frac{1}{N}\sum -2x_i(y_i - (mx_i + b)) \\ \frac{1}{N}\sum -2(y_i - (mx_i + b)) \end{bmatrix}$$

Weight
Bias

y = weight * x + bias

- Partial derivative:

$$\frac{\partial E}{\partial a} = -(y_0 - y)\frac{\partial y}{\partial b} = -(y_0 - y)\frac{\partial y}{\partial z}\frac{\partial z}{\partial a} \quad \text{(chain rule)}$$

$$\frac{\partial E}{\partial b} = -(y_0 - y)\frac{\partial y}{\partial b}$$

- Gradient descent:

$$a_{k+1} = a_k + \eta\frac{\partial E}{\partial a_k}$$

$$b_{k+1} = b_k + \eta\frac{\partial E}{\partial b_k}$$

Gradient Descent

Gradient Descent is an optimization algorithm used for minimizing the loss function in various machine learning algorithms. It is used for updating the parameters of the learning model.

$$m = m - LD_m$$

$$c = c - LDc$$

m --> slope
c --> intercept
L --> Learning Rate

D_m --> Partial Derivative of loss function with respect to m

D_c --> Partial Derivative of loss function with respect to c

$$D_m = \frac{\partial(Cost\ Function)}{\partial m} = \frac{\partial}{\partial m}\left(\frac{1}{n}\sum_{i=0}^{n}(y_i - y_{i\,pred})^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial m}\left(\sum_{i=0}^{n}(y_i - (mx_i + c))^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial m}\left(\sum_{i=0}^{n}(y_i^2 + m^2x_i^2 + c^2 + 2mx_ic - 2y_imx_i - 2y_ic)\right)$$

$$= \frac{-2}{n}\sum_{i=0}^{n}x_i(y_i - (mx_i + c))$$

$$= \frac{-2}{n}\sum_{i=0}^{n}x_i(y_i - y_{i\,pred})$$

$$D_c = \frac{\partial(Cost\ Function)}{\partial c} = \frac{\partial}{\partial c}\left(\frac{1}{n}\sum_{i=0}^{n}(y_i - y_{i\,pred})^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial c}\left(\sum_{i=0}^{n}(y_i - (mx_i + c))^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial c}\left(\sum_{i=0}^{n}(y_i^2 + m^2x_i^2 + c^2 + 2mx_ic - 2y_imx_i - 2y_ic)\right)$$

$$= \frac{-2}{n}\sum_{i=0}^{n}(y_i - (mx_i + c))$$

$$\frac{-2}{n}\sum_{i=0}^{n}(y_i - y_{i\,pred})$$