Intro to Machine Learning

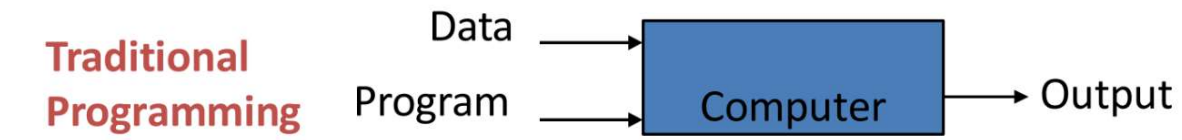
Machine Learning (ML) is a subfield of Artificial Intelligence (AI) that focuses on building systems that can learn from data without being explicitly programmed. Instead of following static, hard-coded instructions, ML algorithms use statistical techniques to identify patterns in data and make predictions or decisions.

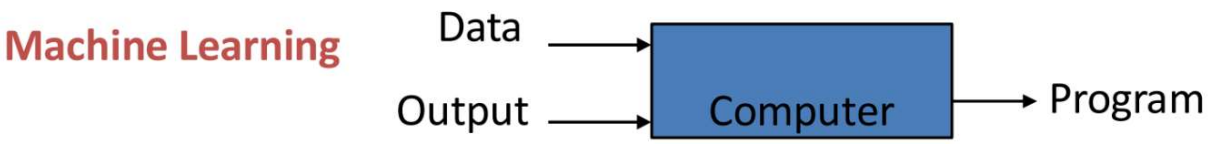
🧠 Core Concept: Learning from Experience

The process of Machine Learning is analogous to how humans learn:

1. **Experience (Data):** The algorithm is fed a large dataset (the "experience").
2. **Tasks (Goal):** The algorithm is designed to perform a specific task (e.g., classifying images, predicting house prices).
3. **Performance (Evaluation):** The algorithm uses the data to learn how to improve its performance on that task over time.

The goal is for the machine to **generalize** from the training data to be able to accurately process and respond to **new, unseen data**.





In Machine Learning, the computer's role is reversed: the goal is to **discover the program itself** (the model, or rules) from the examples.

- **Inputs:** You provide the **Data** (features) and the desired **Output** (labels or answers).
- **Process:** The ML algorithm uses statistical techniques to examine the relationships between the Data and the Output. It attempts to learn a function that accurately maps the inputs to the given outputs.
- **Output:** The **Program** (the learned model, or set of rules).

  Machine Learning: Data + Output $\longrightarrow$ Program

- **Example:** Classifying emails as spam. You input millions of emails (**Data**) and their correct classifications (**Output** - "Spam" or "Not Spam"). The computer learns the hidden rules (the **Program**) that determine which emails are spam, allowing it to classify new emails correctly.

Association Analysis

Association analysis (often called **Association Rule Mining**) is an unsupervised learning technique used to discover relationships between variables in large databases, typically looking for items that occur together frequently.

- **Goal:** Find strong rules and frequent itemsets that connect data points.
- **Example:** In retail, discovering that customers who buy **diapers** often also buy **beer** (market basket analysis).
- **Technique:** The most common algorithm is the **Apriori algorithm**.

- Basket analysis:

  $P(Y \mid X)$ probability that somebody who buys X also buys Y where X and Y are products/services.

Supervised Learning

Supervised learning is defined by the use of **labeled data**—the algorithm is trained on input features paired with their correct outputs (labels). The goal is to learn a mapping function from the input data (X) to the target output (Y).

SUPERVISED LEARNING: USES

Example: decision trees tools that create rules

- Prediction of future cases: Use the rule to predict the output for future inputs
- Knowledge extraction: The rule is easy to understand
- Compression: The rule is simpler than the data it explains
- Outlier detection: Exceptions that are not covered by the rule, e.g., fraud

Regression/Prediction

Regression tasks involve predicting a **continuous numerical value**.

- **Goal:** Predict a number within a range.
- **Output:** A real number (e.g., 15.5, $120,000$).
- **Examples:** Predicting house prices, stock prices, or tomorrow's temperature.

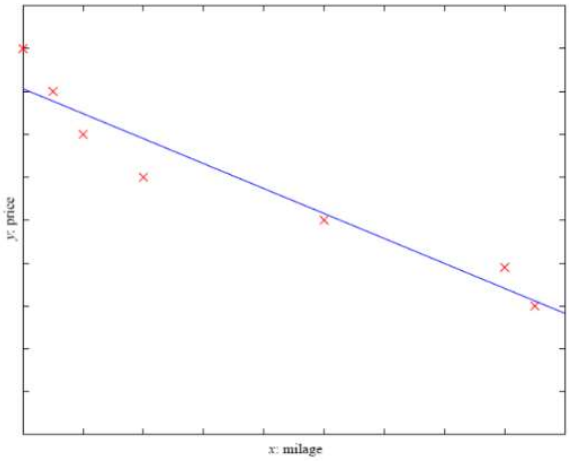
- Example: Price of a used car
- x : car attributes

  y : price

  $y = g(x \mid \theta)$

  $g(\,)$ model,

  θ parameters



$y = wx + w_0$

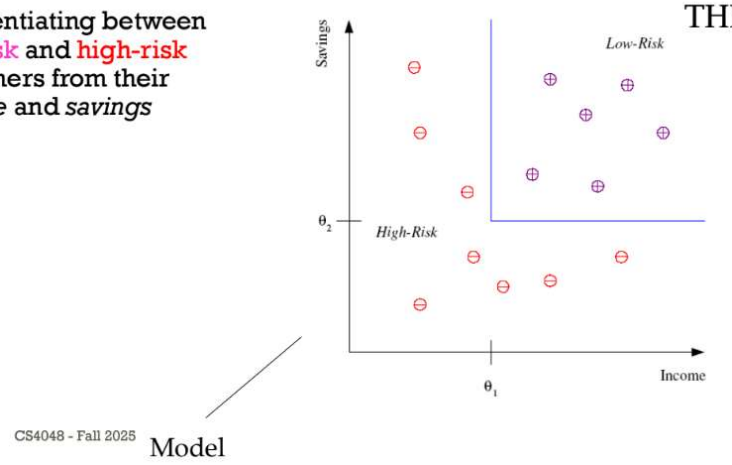
Classification

Classification tasks involve predicting a **discrete category or label**.

- **Goal:** Predict a class or group.
- **Output:** A category (e.g., 'Spam' or 'Not Spam', 'Dog' or 'Cat').
- **Examples:** Image recognition, email filtering, medical diagnosis (e.g., tumor is malignant or benign).

- Example: Credit scoring
- Differentiating between low-risk and high-risk customers from their *income* and *savings*

Discriminant: IF $income > \theta_1$ AND $savings > \theta_2$ THEN low-risk ELSE high-risk



Model

CLASSIFICATION: APPLICATIONS

- A.k.a **Pattern recognition**
- Face recognition: Pose, lighting, occlusion (glasses, beard), make-up, hair style
- Character recognition: Different handwriting styles.
- Speech recognition: Temporal dependency.
  - Use of a dictionary or the syntax of the language.
  - Sensor fusion: Combine multiple modalities; eg, visual (facial image) and acoustic for speech
- Medical diagnosis: From symptoms to illnesses
- Web Advertizing: Predict if a user clicks on an ad on the Internet.

Unsupervised Learning

Unsupervised learning uses **unlabeled data**—the algorithm is only given input data (X) and must discover inherent structures or relationships within it without guidance.

- **Goal:** Find hidden patterns or structure in the data.
- **Data:** Input features only.
- **Primary Tasks:**
  - **Clustering:** Grouping similar data points together (e.g., customer segmentation).
  - **Dimensionality Reduction:** Reducing the number of variables while retaining essential information.

- Learning "what normally happens"
- No output
- Clustering: Grouping similar instances
- Other applications: Summarization, Association Analysis
- Example applications
  - Customer segmentation
  - Image compression: Color quantization
  - Bioinformatics: Learning motifs

Reinforcement Learning (RL)

Reinforcement Learning involves an **agent** that learns an optimal strategy by interacting with an environment through **trial and error**.

- **Goal:** Learn a sequence of actions to maximize cumulative reward over time.
- **Mechanism:** The agent receives a **reward** for desirable actions and a **penalty** for undesirable actions.
- **Applications:** Training robotic control systems, optimizing complex scheduling tasks, and developing game AI.

- Topics:
  - Policies: what actions should an agent take in a particular situation
  - Utility estimation: how good is a state (→used by policy)
- No supervised output but delayed reward

- Applications:
  - Game playing
  - Robot in a maze
  - Multiple agents, partial observability, ...

Model

A model is an **idealized representation** of a system.

Example:
We model the fall of an object on Earth as subject to a constant acceleration of 9.81 m/s^2 due to gravity.

- While this describes the behavior of our system, it is merely an approximation.
- It doesn't account for the effects of air resistance, local variations in gravity, etc.
- But in practice, it's accurate enough to be useful!

Reason 1:

To explain **complex phenomena** occurring in the world we live in.

- How are the parents' average heights related to the children's average heights?
- How do an object's velocity and acceleration impact how far it travels?

Often times, we care about creating models that are **simple and interpretable**, allowing us to understand what the relationships between our variables are.

Reason 2:

To make **accurate predictions** about unseen data.

- Can we predict if an email is spam or not?
- Can we generate a one-sentence summary of this 10-page long article?

Other times, we care more about making extremely accurate predictions, at the cost of having an uninterpretable model. These are sometimes called **black-box models**, and are common in fields like deep learning.

Reason 3:

To make **causal inferences** about if one thing causes another thing.

- Can we conclude that smoking *causes* lung cancer?
- Does a job training program cause increases in employment and wage?

Much harder question because most statistical tools are designed to infer association not causation

This won't be the focus of this course.

CS4048 - Fall 2025

Most of the time, we want to strike a balance between **interpretability** and **accuracy**.

21

A model is fundamentally a **mathematical function** that maps input features (X) to an output (Y):

$$Y = f(X)$$

- X represents the **input data** (the features, like house size, location, and number of bedrooms).
- Y represents the desired **output** (the prediction, like the house price).
- f represents the **model** itself, which contains all the learned weights, coefficients, and rules.

The purpose of the model is to **generalize** the patterns it learned during training to be able to make accurate predictions or decisions on **new, unseen data**.

- For example, once a classification model learns the difference between cats and dogs, it should be able to correctly classify a new photo it has never processed before.

📊 Types of Models

The structure of the model depends on the task it is performing:

Task Type	Model Example	What the Model Represents
Regression (Supervised)	**Linear Regression**	A straight line defined by coefficients (slopes and intercepts) that best fits the data.
Classification (Supervised)	**Decision Tree**	A series of nested *if/then* rules that lead to a specific category prediction.
Clustering (Unsupervised)	**K-Means**	The coordinates of the cluster centers and the assignments of all data points to those centers.

Simple Linear Regression (SLR)

The regression line is the unique straight line that mathematically summarizes the linear relationship between two continuous variables. It is also known as the **Line of Best Fit**.

The entire goal is to predict the value of the **dependent variable (Y)** using one or more **independent variables (X)**.

Notation used in calculus $\hat{y} = a + bx$ → Notation used in ML $\hat{y} = \theta_0 + \theta_1 x$

1. Goal: Minimizing Error (Least Squares)

The placement of the regression line is determined by the **least-squares method**.

- The line is the **unique straight line that minimizes the mean squared error of estimation** among all possible straight lines.
- The **error** for any single data point is the vertical distance between the actual observed value (y) and the value predicted by the line ($\hat{y}$). This distance is called the **residual**.
- **Residual Formula:** $\text{residual} = \text{observed } y - \text{regression estimate } \hat{y}$.
- By minimizing the sum of these squared residuals, the line provides the most accurate possible linear prediction.

2. Components and Formulas

The line is defined by its slope (b_1) and its intercept (b_0).

Component	Formula/Definition	Description
Slope (b_1)	$\text{slope} = r \times \frac{\text{SD of } y}{\text{SD of } x}$	Represents the change in the predicted Y ($\hat{y}$) for a one-unit change in X. It ties directly to the **correlation coefficient (r)**.
Y-Intercept (b_0)	$\text{intercept} = \text{average of } y - \text{slope} \times \text{average of } x$	The point where the regression line crosses the Y-axis (the predicted Y value when $X = 0$). The line always passes through the point $(\bar{x}, \bar{y})$.
Regression Estimate ($\hat{Y}$)	$\text{regression estimate} = \text{intercept} + \text{slope} \times x$	The final equation used to make predictions.

THE REGRESSION LINE

The **regression line** is the unique straight line that minimizes the **mean squared error** of estimation among all straight lines.

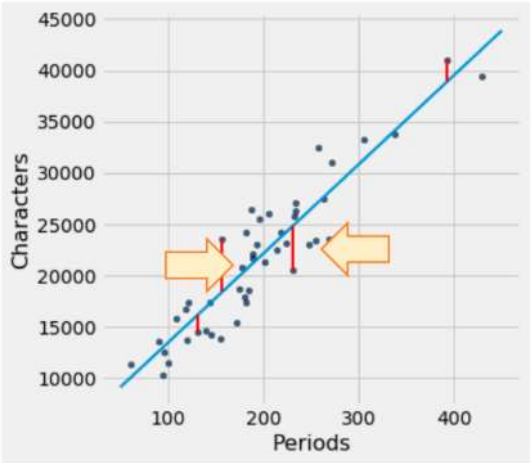
$$\text{slope} = r \times \frac{\text{SD of } y}{\text{SD of } x}$$

$$\text{intercept} = \text{average of } y - \text{slope} \times \text{average of } x$$

$$\text{regression estimate} = \text{intercept} + \text{slope} \times x$$

$$\text{residual} = \text{observed } y - \text{regression estimate}$$

CS4048 - Fall 2025



For every chapter of the novel *Little Women*, Estimate the **# of characters** $\hat{y}$ based on the **number of periods** x in that chapter.

$$\text{Slope} = r\frac{\sigma_y}{\sigma_x}$$

This derivation reveals that the slope in the original units must account for both the **strength of the linear relationship (r)** and the **relative spread** of the two variables (the ratio of σ_y to σ_x).

- σ_y (SD of y): The standard deviation of the dependent variable Y.
- σ_x (SD of x): The standard deviation of the independent variable X.

It can be interpreted as: **For every one standard deviation increase in X (SD of x), the predicted value of Y ($\hat{Y}$) increases by r standard deviations of Y ($r \cdot$ SD of y).**

Covariance

$$Cov(A, B) = E(A \cdot B) - \bar{A}\bar{B}.$$

$$Cov(AllElectroncis, HighTech) = \frac{6 \times 20 + 5 \times 10 + 4 \times 14 + 3 \times 5 + 2 \times 5}{5} - 4 \times 10.80$$

$$= 50.2 - 43.2 = 7.$$

Correlation

Correlation measures the **strength & direction** of a linear association between two variables. Moreover, it makes the variables **unitless** and **limits** it within a range (i.e. [-1, 1]).

correlation

$$\text{slope} = r \times \frac{\text{SD of } y}{\text{SD of } x}$$

$$\text{corr}(\mathbf{x}, \mathbf{y}) = \frac{\text{covariance}(\mathbf{x}, \mathbf{y})}{\text{standard\_deviation}(\mathbf{x}) * \text{standard\_deviation}(\mathbf{y})} = \frac{s_{xy}}{s_x \, s_y}, \quad (2.11$$

where we are using the following standard statistical notation and definitions

$$\text{covariance}(\mathbf{x}, \mathbf{y}) = s_{xy} = \frac{1}{n-1} \sum_{k=1}^{n} (x_k - \overline{x})(y_k - \overline{y}) \quad (2.12$$

$$\text{standard\_deviation}(\mathbf{x}) = s_x = \sqrt{\frac{1}{n-1} \sum_{k=1}^{n} (x_k - \overline{x})^2}$$

$$\text{standard\_deviation}(\mathbf{y}) = s_y = \sqrt{\frac{1}{n-1} \sum_{k=1}^{n} (y_k - \overline{y})^2}$$

$$\overline{x} = \frac{1}{n} \sum_{k=1}^{n} x_k \text{ is the mean of } \mathbf{x}$$

$$\overline{y} = \frac{1}{n} \sum_{k=1}^{n} y_k \text{ is the mean of } \mathbf{y}$$

Side note: **covariance** is $\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = r\sigma_x\sigma_y$

- Correlation measures the strength of a linear association between two variables.
- It ranges **between -1 and 1.**
  - r = 1 indicates perfect linear association; r = -1 perfect negative association.
  - The closer r is to 0, the weaker the linear association is.
- It says nothing about **causation** or **nonlinear association.**
  - Correlation does not imply causation.
  - When r = 0, the two variables are uncorrelated. However, they could still be related through some non-linear relationship.

The correlation r is the average of the product of x and y, both measured in standard units.

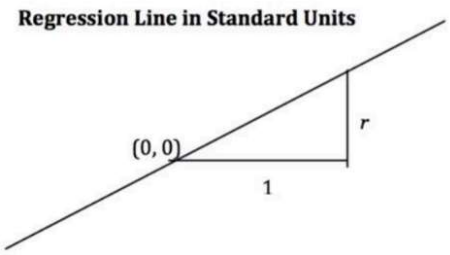
Derivation of Regression Line Expression

When the variables x and y are measured in standard units, the regression line for predicting based on y has slope x passes through r the origin and the equation will be:

- When the variables x and y are measured in standard units, the regression line for predicting based on y has slope x passes through the origin and the equation will be:

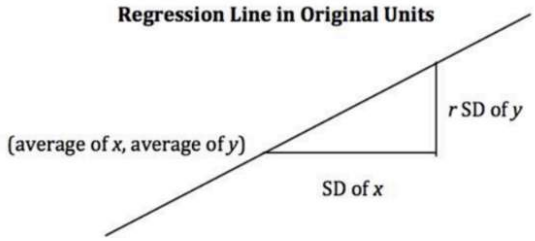
$$\hat{y} = r \times x$$

(both measured in standard units)



- In the original units of the data, this becomes:

$$\frac{\hat{y} - \bar{y}}{\sigma_y} = r \times \frac{x - \bar{x}}{\sigma_x}$$



1. Regression in Standard Units (Standardized Data)

When both the independent variable (X) and the dependent variable (Y) are converted into **standard units** (Z-scores), the regression line simplifies significantly:

- **Standard Units:** A value x converted to standard units is $\frac{x-\bar{x}}{\sigma_x}$, where $\bar{x}$ is the mean and σ_x is the standard deviation.
- **Equation in Standard Units:** When variables are measured in standard units, the regression line passes through the **origin** $(0, 0)$.

$$\hat{y}_{\text{standard}} = r \times x_{\text{standard}}$$

  - Here, $\hat{y}$ is the predicted Y in standard units, and r is the **correlation coefficient**.
  - The slope of the line in standard units is simply r.

2. Deriving the Equation in Original Units

The fundamental relationship from the standard unit equation is then converted back into the original units of the data:

- **Initial Conversion:** Replacing the standard unit variables with their definitions:

$$\frac{\hat{y} - \bar{y}}{\sigma_y} = r \times \frac{x - \bar{x}}{\sigma_x}$$

- **Solving for $\hat{y}$:** By multiplying both sides by σ_y and adding $\bar{y}$, the equation is rearranged to solve for the predicted value, $\hat{y}$, in the original units:

$$\hat{y} = \sigma_y \times r \times \frac{x - \bar{x}}{\sigma_x} + \bar{y}$$

- **Final Form:** The terms are then grouped to isolate the slope and intercept, defining the regression line in the standard $\hat{y} = \hat{a} + \hat{b}x$ format.

$$\hat{y} = \left(\frac{r\sigma_y}{\sigma_x}\right) x + \left(\bar{y} - \frac{r\sigma_y}{\sigma_x}\bar{x}\right)$$

Passage Through the Means: The line in original units always passes through the point defined by the **average of X and the average of Y** $(\bar{x}, \bar{y})$.

4. Error in Regression

The difference between the actual observed value and the predicted value for any data point i is called the **residual** or **error**:

$$e_i = y_i - \hat{y}_i$$

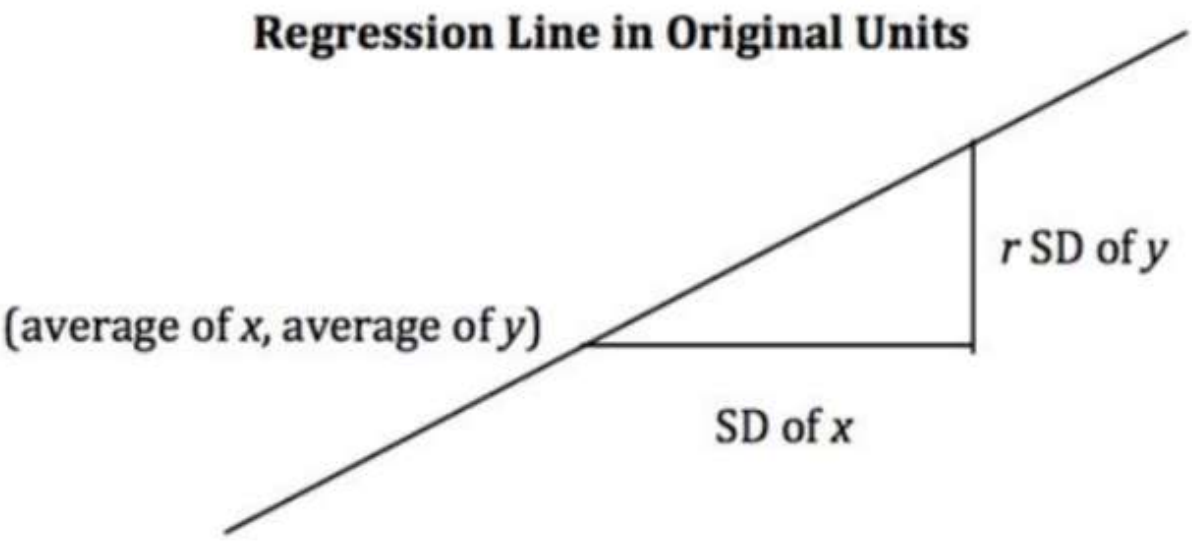
Regression Line Graph

📐 Regression Line in Original Units

In the original units of the data, the regression line is defined by the formula: $\hat{Y} = b_0 + b_1 X$.

Center of the Line

- The line is guaranteed to pass through the point representing the **mean of X and the mean of Y** $(\bar{x}, \bar{y})$.
- The **Y-intercept (b_0)** is the predicted value of Y when X is zero, which is $\bar{y} - b_1\bar{x}$. Unless both $\bar{x}$ and $\bar{y}$ are zero, the line will **not** pass through the origin $(0, 0)$.



$$\text{Slope} = \frac{\text{Rise}}{\text{Run}} = \frac{r \cdot \text{SD of } y}{\text{SD of } x}$$

📏 Regression Line in Standard Units

When both variables X and Y are converted to standard units, you are performing two key transformations:

1. Scaling (Standard Deviation)

Every data point is divided by its respective standard deviation (σ_x or σ_y). This scales the data so that the standard deviation of the standardized variables is 1.

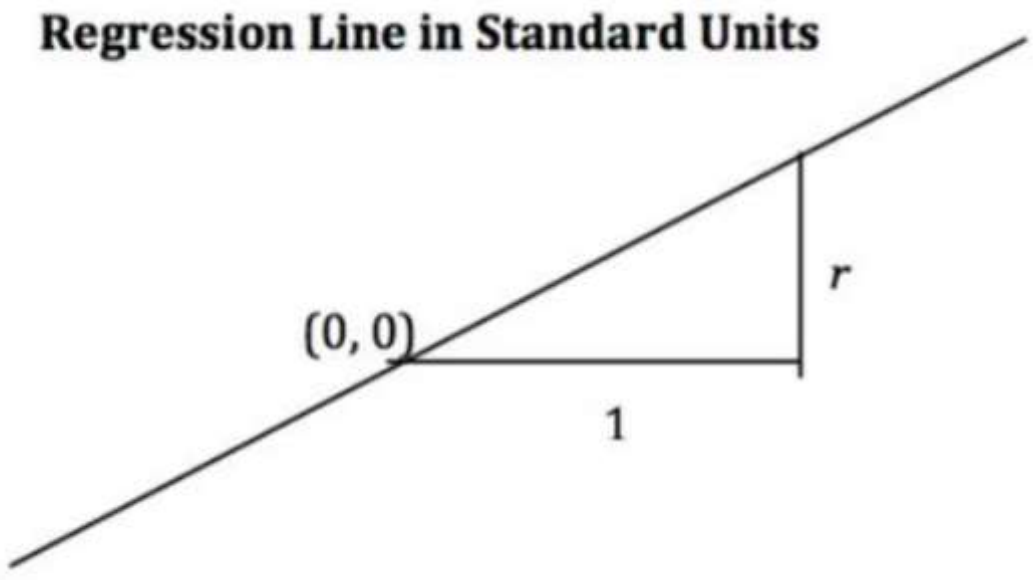
2. Centering (Mean)

Every data point is shifted by subtracting its mean ($\bar{x}$ or $\bar{y}$). This centers the data so that the mean of the standardized variables is 0.

Position of the Line

- Since the mean of X and the mean of Y are both zero in standard units, the center point $(\bar{x}, \bar{y})$ becomes the **origin** $(0, 0)$.
- Therefore, the regression line in standard units **always passes through the origin** $(0, 0)$, simplifying the equation to $\hat{y}_{\text{standard}} = r \times x_{\text{standard}}$.

The graphs in the slide clearly show this difference: the line in original units passes through $(\bar{x}, \bar{y})$, while the line in standard units passes through $(0, 0)$.



Modeling Process

MODELS IN ML

We'll treat a model as some mathematical rule or function to describe the relationships between variables.

Dataset

x	y
x_1	y_1
x_2	y_2
$\vdots$	$\vdots$
x_n	y_n

Observation (x_i, y_i)

- Independent variable
- **Input**
- **Feature**
- **Attribute**

- Dependent variable
- **Target**
- **Output**
- **Outcome**
- **Response**

Prediction

If we use x to predict y, the predictions are denoted as $\hat{y} = \{\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n\}$

Models

Some models we will see in the next few lectures:

$$\hat{y}_i = \theta_0 + \theta_1 x_i$$
$$\hat{y}_i = \theta_0$$
$$\hat{y}_i = x_i^\top \theta$$

→ **Parametric models**

CS4048 - Fall 2025

Parametric models are described by a few **parameters** $(\theta_0, \theta_1, \text{etc.})$

- No one tells us the parameters: the data informs us about them.
- The x, y values are **not** parameters because we directly observe them.
- Sample-based **estimate** of parameter θ is written as $\hat{\theta}$.
  - The "hat" here is different from the "hat" in $\hat{y}$: one means estimate and one means prediction.
- Usually, we pick the parameters that appear "**best**" according to some criterion we choose.

θ Model parameter(s) } $\hat{y} = \theta_0 + \theta_1 x$ Any linear model with parameters $\theta = [\theta_0, \theta_1]$

$\hat{\theta}$ Estimated parameter(s), "best" fit to data in some sense } $\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x$ The "best" fitting linear model with parameters $\hat{\theta} = [\hat{\theta}_0, \hat{\theta}_1]$

Note: Not all statistical models have parameters!

KDEs, k-Nearest Neighbor classifiers are non-parametric models.

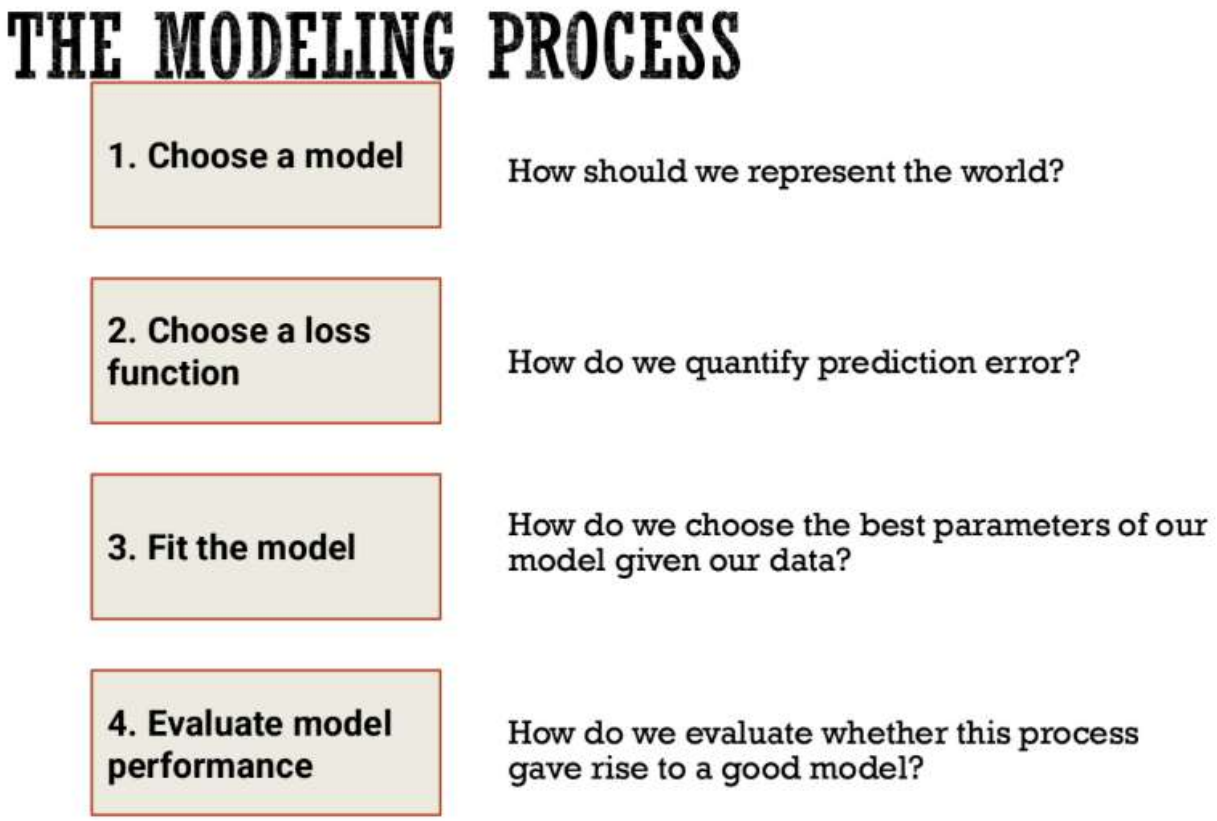
2. Estimated Parameters ($\hat{\theta}$)

The second part of the slide defines the output of the Machine Learning process—the model that has actually been "learned" or fitted to the data:

- **Estimated Parameters ($\hat{\theta}$):** These are the specific numerical values for the coefficients that minimize the error on the training data. They are the model's actual intercept and slope, denoted as $\hat{\theta} = [\hat{\theta}_0, \hat{\theta}_1]$.
- **"Best" Fitting Model:** The resulting equation is the **"best" fitting linear model** to the data in some sense. In standard regression (least-squares), "best fit" means minimizing the **Mean Squared Error (MSE)**, where the difference between the actual Y and predicted $\hat{Y}$ is minimized.

$$\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x$$

The difference between θ and $\hat{\theta}$ highlights the transition from a **hypothetical model structure** to a **learned, actionable model** that can make predictions.



$$\hat{y} = \theta_0 + \theta_1 x$$ SLR model

$$L(y, \hat{y}) = (y - \hat{y})^2$$ Squared loss

$$\hat{R}(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 x))^2$$ MSE for SLR

er $\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x$ $\begin{cases} \hat{\theta}_0 = \bar{y} - \hat{\theta}_1 \bar{x} \\ \hat{\theta}_1 = r\dfrac{\sigma_y}{\sigma_x} \end{cases}$

1.Choose a Model

Simple Linear Regression Model (SLR)

Notation used in calculus $\hat{y} = a + bx$ → Notation used in ML $\hat{y} = \theta_0 + \theta_1 x$

SLR is a parametric model, meaning we choose the "best" parameters for slope and intercept based on data.

- We often express θ as a single parameter vector.
- x is **not** a parameter! It is input to our model.

$x \rightarrow$ SLR $\theta = [\theta_0, \theta_1] \rightarrow \hat{y}$

- Note that the true relationship between x and y is usually non-linear. This is why $\hat{y}$ (and not y) appears in our **estimated linear model** expression.

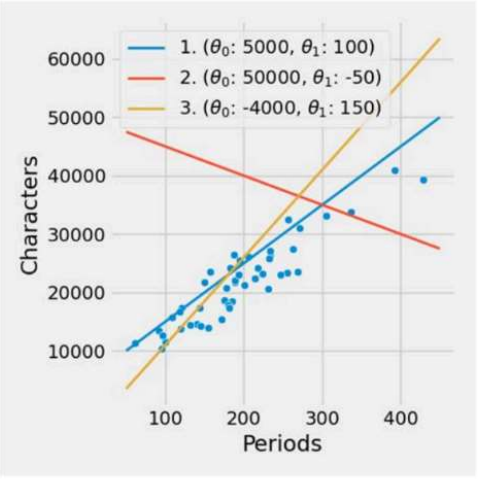
2.Choose a Lose Function

WHICH θ IS BEST?

Based on your interpretation of the data, which are the "optimal parameters" for this linear model?

$$\hat{y} = \theta_0 + \theta_1 x$$

$$\hat{\theta}_0 =? \quad \hat{\theta}_1 =?$$



We only had 3 values to choose from to find the optimal parameter. In practice, our parameter domain is all reals, i.e., $\theta = [\theta_0, \theta_1] \in \mathbb{R}^2$

For every chapter of the novel *Little Women*, Estimate the **# of characters** $\hat{y}$ based on the **number of periods** x in that chapter.

1. **Data Trend:** The blue dots (data points) clearly show a **strong positive linear relationship**; as the number of periods (x) increases, the number of characters ($\hat{y}$) also increases.
2. **Optimal Slope ($\hat{\theta}_1$):** Therefore, the optimal slope must be **positive and relatively steep**.
3. **Optimal Intercept ($\hat{\theta}_0$):** The line should start at a positive y-intercept and then rise.

Model	Parameters (θ_0, θ_1)	Visual Fit	Conclusion
1. Blue Line	$\theta_0 = 5000, \theta_1 = 100$	The line follows the positive trend perfectly and is the closest to the majority of data points.	**BEST FIT**
2. Red Line	$\theta_0 = 50000, \theta_1 = -50$	The slope is **negative** ($\theta_1 = -50$), directly contradicting the positive trend of the data.	Poor Fit
3. Yellow Line	$\theta_0 = -4000, \theta_1 = 150$	The slope is positive, but the intercept ($\theta_0 = -4000$) is too low, causing the line to miss the data cloud significantly at lower x values.	Poor Fit

💡 Conclusion

Based on visual inspection, the **blue line (Model 1)** is the **optimal linear model** because it minimizes the vertical distance (residual) to the data points, correctly modeling the strong positive association between periods and characters.

$$\hat{\theta}_0 \approx 5000$$

$$\hat{\theta}_1 \approx 100$$

In real-world ML, these parameters would be calculated analytically using the least-squares formula, not just guessed.

Loss functions are used to measure the **difference (discrepancy)** between the **predicted** value of a model and the **true** value.

We need some metric of how "good" or "bad" our predictions are.

A **loss function** characterizes the **cost**, error, or fit resulting from a particular choice of model or model parameters.

- Loss quantifies how **bad** a prediction is for a **single** observation.
- If our prediction $\hat{y}$ is **close** to the actual value y, we want **low loss**.
- If our prediction $\hat{y}$ is **far** from the actual value y, we want **high loss**.

$$L(y, \hat{y})$$

There are many definitions of loss functions!

The choice of loss function:

- Affects the accuracy and computational cost of estimation.
- Depends on the estimation task:
  - Are outputs quantitative or qualitative?
  - Do we care about outliers?
  - Are all errors equally costly? (e.g., false negative on cancer test)

Squared Loss

$$L(y, \hat{y}) = (y - \hat{y})^2$$

- Widely used.
- Also called "L2 loss".
- Reasonable:
  - $\hat{y} = y \rightarrow$ good prediction $\rightarrow$ good fit $\rightarrow$ no loss
  - $\hat{y}$ far from $y \rightarrow$ bad prediction $\rightarrow$ bad fit $\rightarrow$ ***lots of loss***

Absolute Loss

$$L(y, \hat{y}) = |y - \hat{y}|$$

- Sounds worse than it is.
- Also called "L1 loss".
- Reasonable:
  - $\hat{y} = y \rightarrow$ good prediction $\rightarrow$ good fit $\rightarrow$ no loss
  - $\hat{y}$ far from $y \rightarrow$ bad prediction $\rightarrow$ bad fit $\rightarrow$ ***some loss***

For our SLR model $\hat{y} = \theta_0 + \theta_1 x$

$$L(y, \hat{y}) = (y - (\theta_0 + \theta_1 x))^2$$

For our SLR model $\hat{y} = \theta_0 + \theta_1 x$

$$L(y, \hat{y}) = |y - (\theta_0 + \theta_1 x)|$$

🛑 Why Simple Residual Error Doesn't Work as a Loss Function

The Loss Function in Machine Learning is the objective that the model seeks to minimize during training. It measures how "bad" the model's predictions are. While the **residual** is the fundamental unit of error, using its raw value directly causes critical problems.

1. The Problem of Cancellation (The Core Issue)

The primary reason we don't use the simple residual (e) is that **positive and negative errors cancel each other out**.

- **The Formula:** The residual is defined as $e = (y - \hat{y})$, where y is the observed value and $\hat{y}$ is the predicted value.
- **The Flaw:** If the model makes two equal-sized, but opposite-direction errors (e.g., one residual is $+10$ and another is -10), the sum of these errors is $10 + (-10) = 0$.
- **Misleading Result:** The model would calculate a total error (loss) of zero, incorrectly suggesting it is perfect, even though the predictions were very far off. **"Big negative residuals shouldn't cancel out big positive residuals!"**.

2. The Solution: Minimizing Magnitude

To solve the cancellation problem, the loss function must ensure that **all errors contribute positively** to the total loss, regardless of the direction of the prediction error.

The two main ways to convert errors into positive values are:

- **Squaring the Error (Squared Residuals):** This leads to the **Mean Squared Error (MSE)**, the standard loss function used in linear regression.

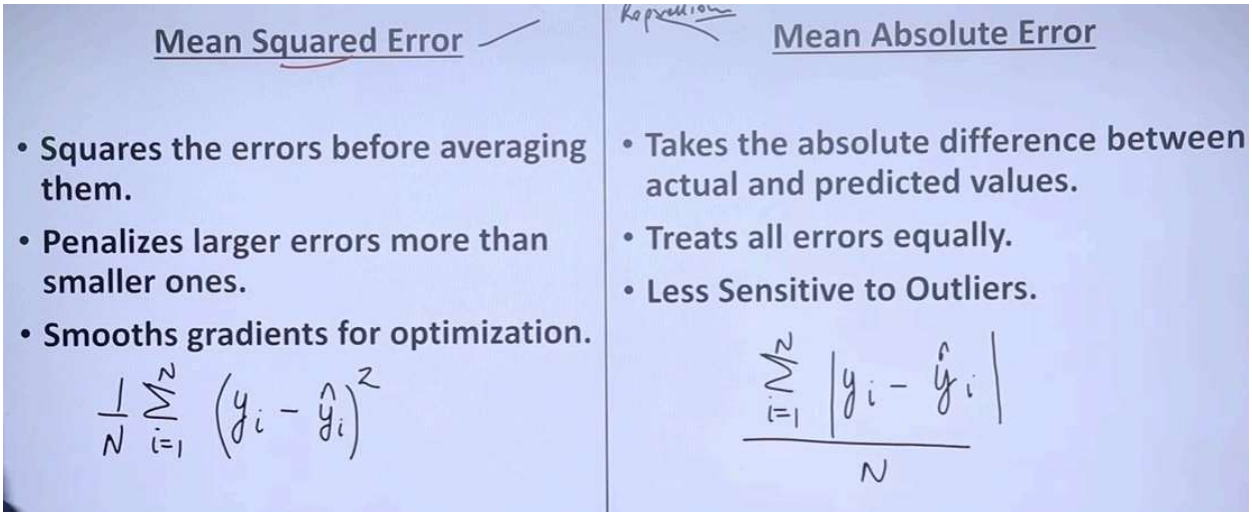
$$\text{Loss} = (y - \hat{y})^2$$

- **Taking the Absolute Value (Absolute Residuals):** This leads to the **Mean Absolute Error (MAE)**, used when outliers are a major concern.

$$\text{Loss} = |y - \hat{y}|$$

By squaring or taking the absolute value, the algorithm is forced to minimize the **magnitude** of all errors, leading to the "best fit" line.

Mean Squared Error



Actual	Predicted	Error	Square Error
10	7	3	9
16	14	2	4
13	17	-4	16
19	20	-1	1
7	4	3	9
		13	39

Empirical Risk $\hat{R}(\theta)$

Empirical Risk is Average Loss Over Data

It is defined as the average loss across all data points in the training sample. It is the measure of how well a Machine Learning model performs on a specific dataset. We care about how bad our model's predictions are for our entire data set, not just for one point.

Minimizing the Empirical Risk is mathematically equivalent to finding the line of best fit.

1. Defining Average Loss (Empirical Risk)

In ML, we don't just care about how bad the prediction is for a single data point; we care about the average error across the **entire dataset**. This average error is called the **Average Loss** or **Empirical Risk**.

- **Goal of Training:** The goal of training an ML model is to find the set of model parameters (θ) that minimizes this Empirical Risk, $\hat{R}(\theta)$.
- **Formula:** The mathematical formula for Empirical Risk is the mean of the loss calculated for every single data point in the dataset, $\mathcal{D}$:

$$\hat{R}(\theta) = \frac{1}{n} \sum_{i} L(y_i, \hat{y}_i)$$

  - $L(y_i, \hat{y}_i)$: This is the **Loss Function** (e.g., Mean Squared Error or MSE) applied to the i-th data point. It measures the error between the true value (y_i) and the predicted value ($\hat{y}_i$).
  - $\sum_i L(\dots)$: This is the sum of the loss across all n data points.
  - $\frac{1}{n}$: Dividing by n gives the **average loss**.

2. Dependence on Parameters (θ)

The Empirical Risk $\hat{R}(\theta)$ is a function of the **model parameters (θ)**.

- The data is held fixed (it doesn't change).
- The model parameters (θ), such as the slope and intercept, determine the predicted value ($\hat{y}$).
- Therefore, changing θ changes $\hat{y}$, which changes the loss, and ultimately changes the Empirical Risk. ML training is the process of iteratively adjusting θ to find the minimum $\hat{R}(\theta)$.

3. Sample vs. Population Risk

The slide emphasizes a crucial distinction:

- The **average loss on the sample** (Empirical Risk) tells us how well the model fits the training data.
- It **does not** tell us how well the model fits the entire **population** (the theoretical true distribution of data, often called **Expected Risk** or **True Risk**).

The hope in ML is that the Empirical Risk (the error on our sample) is **close** to the True Risk (the error on the population). If they are very different, the model has likely **overfitted** the training data.

$$\hat{R}(\theta) = \frac{1}{n}\sum_i L(y_i, \hat{y}_i)$$

The colloquial term for average loss depends on which loss function we choose.

L2 loss → **Mean Squared Error (MSE)** $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

L1 loss → **Mean Absolute Error (MAE)** $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$

49

$$\hat{R}(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 x))^2$$

MSE for SLR

The combination of model + loss that we focus on today is known as **least squares regression**.

Actual	Predicted	Error	Square Error
10	7	3	9
16	14	2	4
13	17	-4	16
19	20	-1	1
7	4	3	9
		13	39

MSE	=(39/5)

MSE	7.8

The **Emperical Risk** has calculated the average of Mean Squared Error:

1. Sum the MSE for each data point.
2. Divide by the number of data points.
3. The result is the Average Loss over data.

The average loss on the sample tells us how well the model fits the data (not the population).

Mean Squared Error (MSE) is the **specific loss function (L)** chosen to measure the error for regression. When you substitute the MSE formula into the general Empirical Risk equation, you get the objective function used for SLR:

- **L2 Loss:** MSE is also known as L2 Loss.
- **Formula:** It calculates the average of the squared residuals:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Therefore, for SLR, the **Empirical Risk is defined as the Mean Squared Error (MSE)**. Minimizing MSE is the mathematical method used by **Ordinary Least Squares (OLS) regression** to find the "line of best fit".

3.Fit the Model

Model Parameters (θ)

🧠 What are Model Parameters?

Model parameters (θ) are the **variables internal to the model** whose values are estimated or "learned" from the training data. They are the coefficients and weights that define the specific function the model uses to map inputs to outputs.

- **Definition:** They are the elements that allow the model to make predictions.
- **Example (Linear Regression):** In the equation $\hat{y} = \theta_0 + \theta_1 x$, the parameters are the **intercept (θ_0)** and the **slope (θ_1)**. These two numbers define the position and orientation of the entire regression line.
- **Goal:** The training process aims to find the specific **estimated parameters ($\hat{\theta}$)** that provide the "best fit" to the training data in some sense (i.e., minimizing the error).

🛠️ How Do Parameters Change? (Training Process)

Parameters change through an iterative optimization process designed to minimize the model's error (or **Loss**).

1. The Goal: Minimizing Loss

The primary objective during training is to find the set of parameters ($\hat{\theta}$) that minimizes the **Empirical Risk** (the **Average Loss** across the entire training dataset).

- **Average Loss Formula:** $\hat{R}(\theta) = \frac{1}{n} \sum_i L(y_i, \hat{y}_i)$.
- The Average Loss is a function of the parameters (θ) because θ determines the predicted output ($\hat{y}$).

2. The Method: Gradient Descent

The most common method for iteratively changing parameters is **Gradient Descent**.

1. **Calculate Loss:** The algorithm uses the current parameters (θ) to make predictions ($\hat{y}$) and calculates the total loss ($\hat{R}(\theta)$).
2. **Calculate Gradient:** The algorithm calculates the **gradient** (the partial derivative) of the loss function with respect to *each parameter* ($\frac{\partial \hat{R}}{\partial \theta_i}$). The gradient points in the direction of the **steepest increase** in loss.
3. **Adjust Parameters:** The algorithm updates the parameters by moving in the **opposite direction** of the gradient (the direction of steepest decrease in loss). The size of this step is controlled by the **learning rate**.

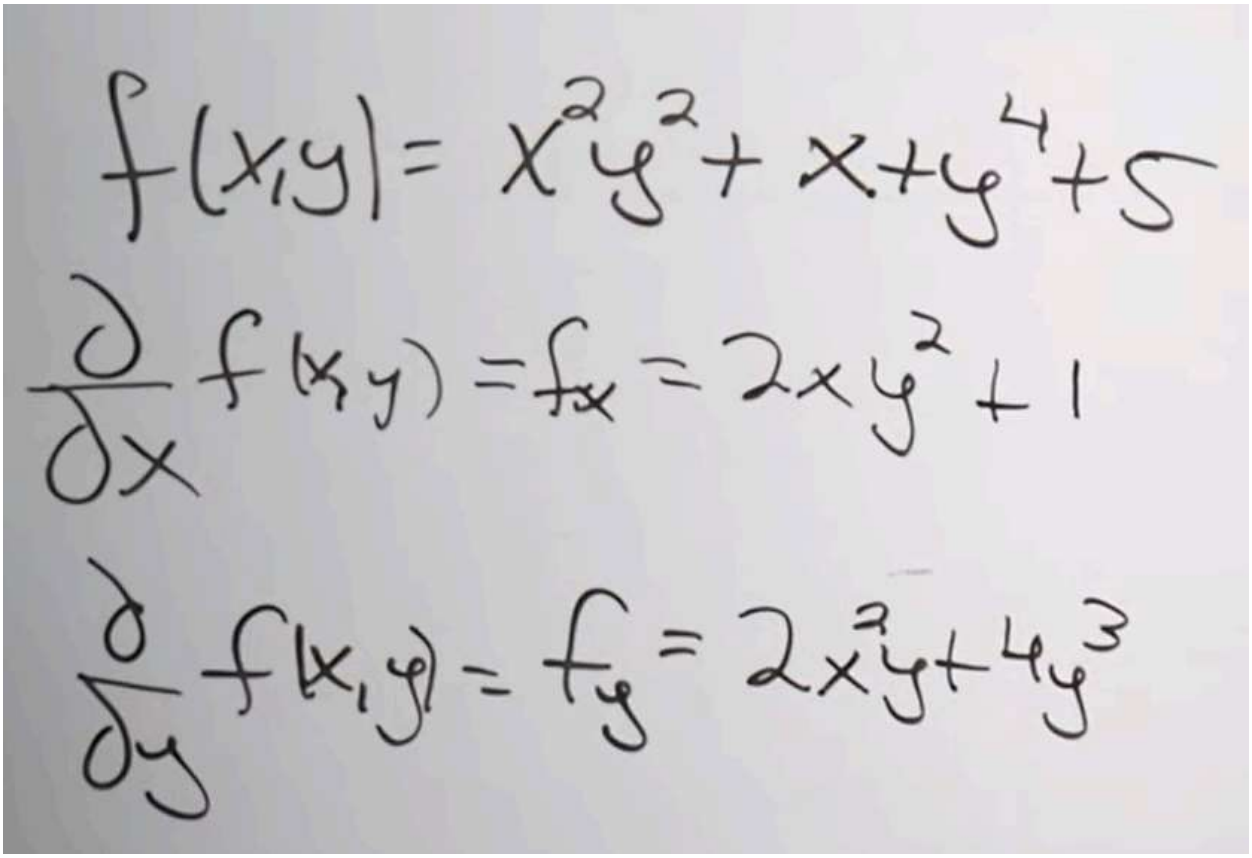
$$\text{New } \theta_i = \text{Old } \theta_i - (\text{Learning Rate} \times \frac{\partial \hat{R}}{\partial \theta_i})$$

4. **Repeat:** This process is repeated across many iterations (or "epochs") until the parameters converge on a value that minimizes the Empirical Risk. The resulting converged values are the estimated parameters, $\hat{\theta}$.

$$\hat{R}(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 x))^2$$

We want to find $\hat{\theta}_0, \hat{\theta}_1$ that minimize this **objective function**.

Partial Derivative



Estimating Parameters ($\hat{\theta}$)

Estimating equations are the constraints that the Least Squares method enforces on the residuals to ensure the regression line is positioned correctly.

To minimize the (sample) Mean Squared Error: $$MSE(\theta_0, \theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 x_i))^2$$

To find the best values, we set derivatives equal to zero to obtain the optimality conditions:

$$\frac{\partial}{\partial \theta_0} MSE = 0$$

$$\frac{\partial}{\partial \theta_1} MSE = 0$$

PARTIAL DERIVATIVE OF MSE WITH RESPECT TO θ_0, θ_1

$$\frac{\partial}{\partial \theta_0} MSE = \frac{\partial}{\partial \theta_0}\frac{1}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)^2$$

Derivative of sum is sum of derivatives $$= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial \theta_0}(y_i - \theta_0 - \theta_1 x_i)^2$$

Chain rule $$= \frac{1}{n}\sum_{i=1}^{n} 2(y_i - \theta_0 - \theta_1 x_i)(-1)$$

Simplify constants $$= \boxed{-\frac{2}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)}$$

$$\frac{\partial}{\partial \theta_1} MSE = \frac{\partial}{\partial \theta_1}\frac{1}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)^2$$

Derivative of sum is sum of derivatives $$= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial \theta_1}(y_i - \theta_0 - \theta_1 x_i)^2$$

Chain rule $$= \frac{1}{n}\sum_{i=1}^{n} 2(y_i - \theta_0 - \theta_1 x_i)(-x_i)$$

Simplify constants $$= \boxed{-\frac{2}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)x_i}$$

ESTIMATING EQUATIONS

To find the best values, we set derivatives equal to zero to obtain the optimality conditions:

$$0 = \frac{\partial}{\partial \theta_0} MSE = -\frac{2}{n}\sum_{i=1}^{n}\left(y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i\right) \iff \frac{1}{n}\sum_i y_i - \hat{y}_i = 0$$

"Equivalent"

$$0 = \frac{\partial}{\partial \theta_1} MSE = -\frac{2}{n}\sum_{i=1}^{n}\left(y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i\right)x_i \iff \frac{1}{n}\sum_i (y_i - \hat{y}_i)x_i = 0$$

Estimating equations

To find the best θ_0, θ_1, we need to solve the **estimating equations** on the right.

Estimating equations are the equations that the model fit has to solve. They help us:

- Derive the estimates.
- Understand what our model is paying attention to.

$$\frac{1}{n}\sum_i y_i - \hat{y}_i = 0$$

$$\frac{1}{n}\sum_i (y_i - \hat{y}_i)x_i = 0$$

For SLR:

- The residuals should **average to zero** (otherwise we should adjust the intercept!)
- The residuals should be **orthogonal to the predictor variable** (or we should adjust the slope!)

1. Condition for the Intercept (Centering the Line)

This equation ensures the line passes through the center of the data.

$$\frac{1}{n}\sum_i y_i - \hat{y}_i = 0$$

- **Simple Meaning:** The **average of the residuals** ($y_i - \hat{y}_i$) must be zero.
- **What it Implies:** The positive errors (over-predictions) must perfectly balance the negative errors (under-predictions).
- **Action:** If this condition isn't met, it means the entire line is systematically too high or too low, and the **intercept must be adjusted**. This is satisfied by forcing the regression line to pass through the point $(\bar{x}, \bar{y})$.

2. Condition for the Slope (Orthogonality)

This equation ensures that the predictor variable (X) has explained all the linear relationship it possibly can.

$$\frac{1}{n}\sum_i (y_i - \hat{y}_i)x_i = 0$$

- **Simple Meaning:** The **covariance between the predictor variable (X) and the residuals (e) must be zero.**
- **Geometric Term:** This is described as the residuals being **orthogonal to the predictor variable.**
- **What it Implies:** There should be no linear pattern remaining in the errors that is related to the value of X. For instance, if large positive errors only occurred at large X values, the slope would need adjustment.
- **Action:** If this condition isn't met, the **slope must be adjusted.** This constraint forces the slope to take the unique value that minimizes the overall Mean Squared Error (MSE).

These two conditions are the **algebraic definitions** of the "best fit" line in Ordinary Least Squares (OLS) regression.

4.Evaluating Models

EVALUATING MODELS

What are some ways to determine if our model was a good fit to our data?

1.Visualize data, compute statistics:

Plot original data.
Compute column means, standard deviation.
If we want to fit a linear model, compute correlation r .

2. Performance metrics:

Root Mean Square Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- It is the square root of MSE, which is the average loss that we've been minimizing to determine optimal model parameters.
- RMSE is in the **same units** as y .
- A lower RMSE indicates more "accurate" predictions (lower "average loss" across data)

3. Visualization:

Look at a residual plot of $e_i = y_i - \hat{y}_i$ to visualize the difference between actual and predicted values.

1. Initial Checks (Visualize Data, Compute Statistics)

Before even running the model, you should understand the raw data:

- **Visualize the Data:** Plotting the original data (e.g., a scatter plot) immediately reveals the relationship's **direction** (positive or negative) and **form** (linear or curved).
- **Compute Statistics:** Calculate the mean, standard deviation, and most importantly, the **correlation coefficient (r)**. A strong r (close to $+1$ or -1) suggests a linear model is appropriate and has a good chance of fitting well.

2. Performance Metrics (Root Mean Square Error - RMSE)

Once the model is trained, you need a single number to quantify its accuracy.

- **RMSE is the Standard Metric:** The **Root Mean Square Error (RMSE)** is the square root of the Mean Squared Error (MSE), which is the **average loss** minimized during training.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- **Key Advantage:** RMSE is in the **same units as y** (the dependent variable). For example, if you are predicting house prices in dollars, the RMSE will also be in dollars, making it easily interpretable.
- **Interpretation:** A **lower RMSE** indicates more accurate predictions and a better fit, as it means the average size of the prediction errors across the dataset is smaller.

3. Visualization (Residual Analysis)

While RMSE gives you one number, a visualization of the errors tells you *how* the model is wrong.

- **Residual:** The residual (e_i) is the difference between the actual observed value (y_i) and the model's predicted value ($\hat{y}_i$).

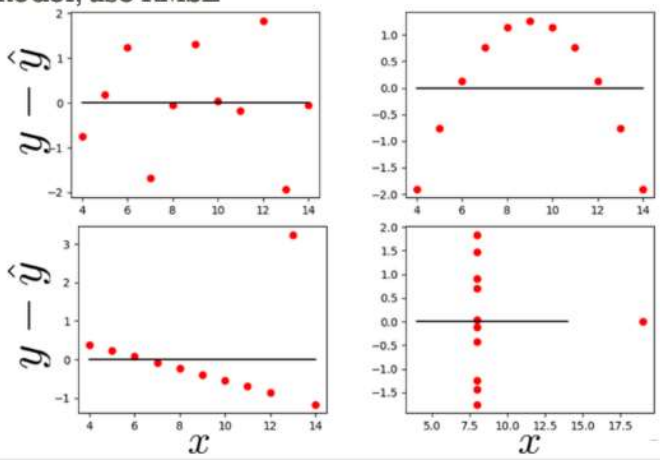
$$e_i = y_i - \hat{y}_i$$

- **Residual Plot:** A residual plot graphs the residuals (e_i) against the predictor variable (x).
- **Goal:** For a good linear model, the residual plot should look like **random scatter** with no discernible pattern. If a pattern remains (e.g., a curve or a cone shape), it means the model is missing key information, suggesting the linear assumption is wrong or that the errors are heteroscedastic.

Ideal model evaluation steps, in order:

1. **Visualize original data, Compute Statistics**
2. **Performance Metrics**
   For our simple linear least square model, use RMSE
3. **Residual Visualization**

The residual plot of a good regression shows **no pattern**.



Terminology of Regression Line

"Least-squares regression line," "ordinary least squares (OLS) regression," and "line of best fit" all refer to the **exact same line** in statistics and Machine Learning. They are interchangeable terms describing the unique straight line that best summarizes the linear relationship between two variables.

The difference in terminology comes from emphasizing either the **goal** (best fit), the **mathematical method** (least squares), or the **full name of the method** (OLS).

1. Line of Best Fit (The Goal)

- **Meaning:** This is the most descriptive, non-technical term. It refers to the straight line that seems to visually fit the scattered data points the closest.
- **Goal:** To provide the single best linear summary of the data, allowing for prediction and interpretation of the relationship.

2. Least-Squares Regression Line (The Method)

- **Meaning:** This term specifically highlights the **mathematical principle** used to determine the line's exact location. The term "regression" indicates that the line is used for prediction (i.e., predicting the dependent variable Y based on the independent variable X).
- **The Principle:** The line is positioned such that the **sum of the squared vertical distances (residuals)** from the data points to the line is minimized.

3. Ordinary Least Squares (OLS) Regression (The Full Procedure)

- **Meaning:** This is the formal statistical name for the entire method used to estimate the parameters (slope and intercept) of the regression line.
- **"Ordinary":** The term "Ordinary" is used to distinguish it from more advanced forms of least squares, such as Weighted Least Squares (WLS) or Generalized Least Squares (GLS), which are used when certain assumptions about the data's error structure are violated.