🎯 What Does It Mean to "Frame a ML Problem"?

Framing a machine learning problem means **translating a real-world business challenge** into a **solvable ML task** — one that includes clearly defined objectives, available data, and measurable outcomes.
Without correct framing, even powerful models will fail to solve the right problem.

---

7 Key Steps to Frame a Machine Learning Problem

Step 1: Understand the Business Problem

- Start with **clarity on the business goal**:
    1. What are we trying to achieve?
    2. What's the value if this problem is solved?

➢ **Case Study (Netflix)**

i. Predicting **user churn** (who will stop subscribing).
ii. ML Task: Build a system that identifies users likely to cancel subscriptions.

Step 2: Convert Business Problem into ML Problem

Ask:

- What **output** should the model predict?
- What is the **input** or feature data?
- Is this **classification**, **regression**, **clustering**, etc.?

Example:
"Will a user churn?" → Classification (Yes/No)
"How many days before churn?" → Regression

Step 3: Identify the Problem Type

Understand what kind of ML problem you're working with:

ML Problem Type	Examples
Classification	Spam or not spam, churn or not churn
Regression	Predicting prices, time, quantity

Clustering	Customer segmentation
Recommendation	Suggesting products, movies
Anomaly Detection	Fraud detection, network intrusion

Step 4: Know the Current Solution (Baseline)

- Is there already a manual or rule-based system?
- What is the **current accuracy or success rate**?
- Helps in **benchmarking** your ML model.

ML must **outperform the existing method** to be useful.

Step 5: Getting and Understanding the Data

Key Questions:

- Do you have enough data?
- Is the data labeled?
- Are the features relevant?
- How recent or representative is the data?

"Bad data = bad model". Data is 80% of ML work.

Step 6: Define Success Metrics

Choose the **right performance metric** based on the problem type:

Problem Type	**Good Metrics**
Classification	Accuracy, F1-score, ROC-AUC, Precision/Recall
Regression	MSE, RMSE, MAE, R^2 Score
Ranking	MAP, NDCG, Hit Rate

Don't default to "accuracy", choose based on your real-world goal.

Step 7: Choose Online or Batch Learning

Ask:

- Do predictions need to happen **in real time** (online)?
- Or is it okay to **process data in chunks** (batch)?

Ex: Fraud detection = online
Product demand forecasting = batch

Bonus Step: Check Assumptions

Make sure your framing is realistic:

- Is enough data available?
- Are labels correct and unbiased?
- Can the ML model solve this problem, or is it too vague?

Final Checklist: Framing an ML Problem

Step	**Description**
1. Business Understanding	What's the real-world goal or pain point?
2. ML Problem Framing	What are the inputs, outputs, and task type?
3. Identify ML Type	Classification, regression, clustering, etc.
4. Current Baseline	How is this done currently?
5. Data Collection	What data do we have or need?
6. Metrics	How do we measure success?
7. Learning Type	Is it real-time (online) or offline (batch)?
8. Assumptions & Limits	Are we missing anything crucial or misframing the task?

Final Thought

Framing the problem is **the most critical step** in any ML or data science project.
A well-framed problem = half-solved problem.
A poorly-framed problem = wasted time, effort, and resources.
