1️⃣ The TOP-LEVEL HIERARCHY (Start Here)

All ML models first split into **three big families** based on *learning signal*:

```
Machine Learning
│
├── Supervised Learning
├── Unsupervised Learning
└── Semi-supervised / Reinforcement Learning
```

We'll focus mainly on the **first two**, because that's where most business ML lives.

2️⃣ SUPERVISED LEARNING (Most Business Problems)

Definition

👉 You have **input (X)** and **known output (y)**.

Examples:

- Churn (Yes/No)
- Fraud (Yes/No)
- Price prediction
- Demand forecasting

2.1 Supervised Learning → By Output Type

```
Supervised Learning
│
├── Classification (Discrete output)
└── Regression (Continuous output)
```

🔷 Classification (Yes / No / Categories)

Examples:

- Churn prediction
- Spam detection
- Disease detection

🔷 Regression (Numbers)

Examples:

- Revenue prediction
- Price prediction
- Time-to-failure

3️⃣ SUPERVISED MODELS → BY *HOW THEY LEARN*

This is the hierarchy you're really asking for 👇

3.1 Rule-Based / Tree-Based Models (Interpretable)

Core idea

Learn **if–else rules** from data.

```
Tree-Based Models
│
├── Decision Tree
├── Random Forest
└── Gradient Boosting (XGBoost, LightGBM, CatBoost)
```

Characteristics

- Pre-hoc interpretability
- Handles non-linearities
- No scaling required
- Works very well on tabular data

Classification / Regression

Model	Classification	Regression
Decision Tree	✅	✅
Random Forest	✅	✅
Gradient Boosting	✅	✅

📌 **Your churn project lives here**.

3.2 Distance-Based Models (Geometry-Driven)

Core idea

Similar points → similar outputs.

```
Distance-Based Models
│
├── K-Nearest Neighbors (KNN)
└── Support Vector Machines (SVM)
```

Characteristics

- Require feature scaling
- Sensitive to distance metric
- Decision boundaries are geometric
- Harder to explain

Classification / Regression

Model	Classification	Regression
KNN	✅	✅
SVM	✅	✅ (SVR)

📌 **Your SVM churn project lives here**.

3.3 Linear / Parametric Models (Equation-Driven)

Core idea

Learn a weighted sum of features.

```
Linear Models
│
├── Linear Regression
├── Logistic Regression
├── Ridge / Lasso / ElasticNet
```

Characteristics

- Simple
- Fast
- Strong assumptions
- Easy to explain

Classification / Regression

Model	Classification	Regression
Logistic Regression	✅	❌
Linear Regression	❌	✅
Ridge / Lasso	❌	✅

📌 Often used as **baseline models**.

3.4 Probabilistic / Bayesian Models

Core idea

Model uncertainty explicitly.

```
Probabilistic Models
│
├── Naive Bayes
├── Bayesian Regression
└── Hidden Markov Models
```

Characteristics

- Fast
- Strong assumptions
- Good with small data
- Outputs probabilities naturally

Classification / Regression

Model	Classification	Regression
Naive Bayes	✅	❌
Bayesian Regression	❌	✅

3.5 Neural Networks (Black-Box Models)

Core idea

Learn complex non-linear functions via layers.

```
Neural Networks
│
├── Feedforward NN
├── CNN
├── RNN / LSTM
└── Transformers
```

Characteristics

- Very powerful
- Data-hungry
- Hard to interpret
- SHAP often needed

Classification / Regression

Model	Classification	Regression
Neural Networks	✅	✅

📌 Usually **not the first choice** for tabular business data.

4️⃣ UNSUPERVISED LEARNING (No Labels)

Definition

👉 You only have **X**, no **y**.

Used for:

- Exploration
- Segmentation
- Anomaly detection
- Feature learning

4.1 Clustering Models

```
Clustering
│
├── K-Means
├── Hierarchical Clustering
└── DBSCAN
```

Use cases:

- Customer segmentation
- Market segmentation

📌 **No classification or regression here** — no labels.

4.2 Dimensionality Reduction

```
Dimensionality Reduction
│
├── PCA
├── t-SNE
└── UMAP
```

Use cases:

- Visualization
- Noise reduction
- Feature compression

4.3 Anomaly Detection

```
Anomaly Detection
│
├── Isolation Forest
├── One-Class SVM
└── Autoencoders
```

Used for:

- Fraud
- Network intrusion
- Rare events

---

5 HYBRID / ADVANCED CATEGORIES

Semi-Supervised Learning

- Few labels + many unlabeled points

Reinforcement Learning

- Sequential decisions
- Rewards
- Not typical for churn

6️⃣ Where Your Current Knowledge Fits (Very Important)

You now understand:

Model Family	You Understand
Tree-based	✅ Deep
Distance-based	✅ Deep
Cost-based decisioning	✅
Interpretability	✅ Advanced
SHAP	✅ Advanced

That's **already strong senior-level coverage** for tabular ML.

7️⃣ Final Mental Map (One Glance)

```
Models
│
├── Interpretable (Pre-hoc)
│    ├── Decision Trees
│    ├── Random Forest
│    └── Linear Models
│
├── Black-Box
│    ├── SVM
│    ├── Gradient Boosting
│    └── Neural Networks
│
├── Distance-Based
│    ├── KNN
│    └── SVM
│
└── Unsupervised
     ├── Clustering
     ├── PCA
     └── Anomaly Detection
```

8️⃣ Interview-Ready Summary (Memorize This)

> "I choose models based on interpretability, data type, and business constraints. For tabular churn problems, tree-based models offer the best balance of performance and explainability, while distance-based and black-box models are considered when performance gains justify the complexity."

That's a **very strong answer**.

If you want next, we can:

- Build a **model-selection decision tree**
- Map **each model to real-world use cases**
- Prepare **interview Q&A on model choice**
- Compare **tree vs SVM vs NN on tabular data**

Just tell me 👍