1. Requirements

- **Offline Metric**: interpretable eval metric for model select, e.g. 1. classify = AUC, F1, Precision, Recall 2. regress = RMSE, MAE, Huber, MAPE, R^2
- **Online Metric**: product specific, monitor in production, e.g. CTR, churn rate, avg watch time, avg cart size

2. Data

- **Choose Target**: 1. explicit (e.g. likes) 2. implicit (must operationalize, e.g. dislikes = reported/negative sentiment comment)
- **Training Data**: 1. sources (e.g. users, products, queries) 2. feats per source (e.g. users = demographics (age), sessions (logins), history (likes))
- **Feats**: 1. preprocess (num → cat encode/scale/outliers/missing, text → TFIDF/embeds) 2. feat engineer 3. feat select (feat importance, regularize)
- **Data Imbalance**: new metric, class weights, undersample/oversample **Data Split**: Train/Val/Test or Train/Test + k-fold val

3. Model

- **Offline Training**: 1. lin/logreg (fast to train, linear) 2. XGB (better performance, nonlin, little preprocessing) 3. DNN (SOTA, nonlin, lots of tuning)
- **Tune**: 1. hyperparam tune (grid search) 2. early stopping (train/val loss diverge) 3. loss func 4. offline metric for model selection

4. Deploy

- **Online Eval**: A/B test new model on user subset using online metric
- **Monitor**: 1. dashboard each query (error rate, metric score, latency time) 2. seasonality 3. retrain when new data or feat distributions change