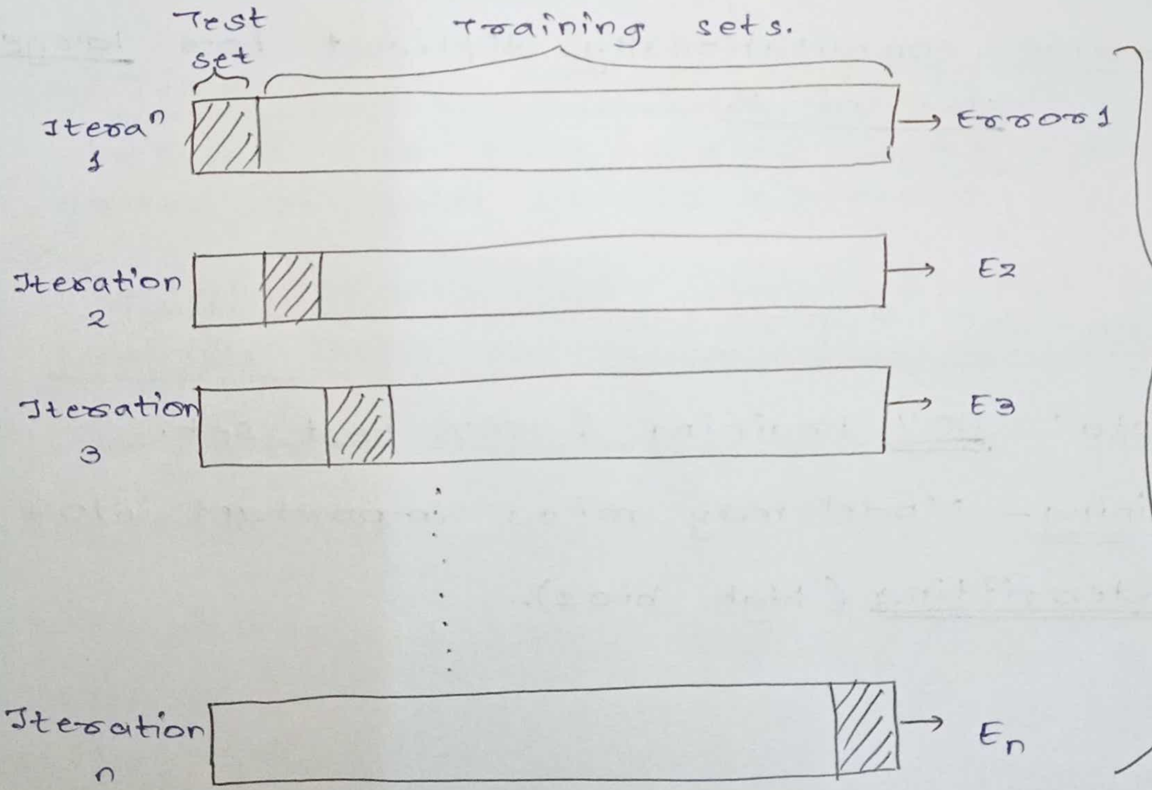
* Cross-Validation *

↳ Randomly Sampling Data:-

- Select random records for train & test.
- May result in overfitting (Model train well on train set, but perform poor on test set).

↳ Cross-Validation:-

- Divide data multiple sets.
- 1 set reserve for test & use others to train.
- Repeat n-times, each time diff test set.
- All results averaged ⇒ performance.
- Prevent overfitting.
- Ensure model robust & generalize well on new data (for testing).



$$\boxed{E = \frac{1}{n} \sum_{i=1}^{n} Error_i}$$

Types of cross-Validation Techniques :-

1. Leave-One-Out cross-Validation (LOOCV) —

↳ Take only 1 data point for testing & user other for training.

↳ Advantage – Bcz train on all data ⇒ low bias.

↳ Disadvantage – • Repeat for len(dataset), require high computational time.
• If test on outlier → High variation in result.

⇒

2. Leave-P-Out cross-Validation (LPOCV) —

↳ Leave P records for test & others for training.

↳ Can control test set size.

↳ Repeat for all samples & average error.

↳ Disadvantage – computationally difficult for large size of P.

3. Validation-Set Approach —

↳ Divide data : 50% training & 50% test set.

↳ 50% training – Model may miss important info & lead to underfitting (high bias).

④ K-Fold Cross-Validation (KFCV) —

↳ Divide data in K-equal sized subsets (folds).

↳ Train & test model K-times, with different test set each time.

↳ Average the errors for performance evaluation.

↳ Easy to understand & output less biased than other techniques.

⑤ Stratified K-Fold Cross-Validation —

↳ Similar to KFCV, but works on stratification concept.

↳ Stratification: Rearrange data to ensure each fold/group represent all classes from complete dataset.

↳ Useful for imbalanced dataset.

⑥ Time Series Cross-Validation —

↳ Used for time-series data, when temporary order of data points essential.

↳ Sequentially split train & test sets, with training data preceding testing data.

↳ Useful — Time-series forecast, stock market prediction, weather forecast.

• Purpose of cross validation:

① Estimate how well model perform on the unseen (testing) data.

② Detect & prevent overfitting (occur when model train well on train set & perform poor on test set.

• Advantages of cross validation:

① Robust model - More robust than single train-test split. Bcz averages evaluation.

② Maximize Data Utilization - Use each data for both training & testing, imp for limited data.

③ Bias Reduction - Reduce bias from single split.

④ Overfitting Detection - when model perform well on train set, but poor on test set.

⑤ Hyperparam Tuning - Help explore different hyper-params & choose one with best result.

• Limitations of cross validation:

① computational cost - n iterations, large datasets.

② Not suitable for time-series data - order is imp. should consider time-based data splits.

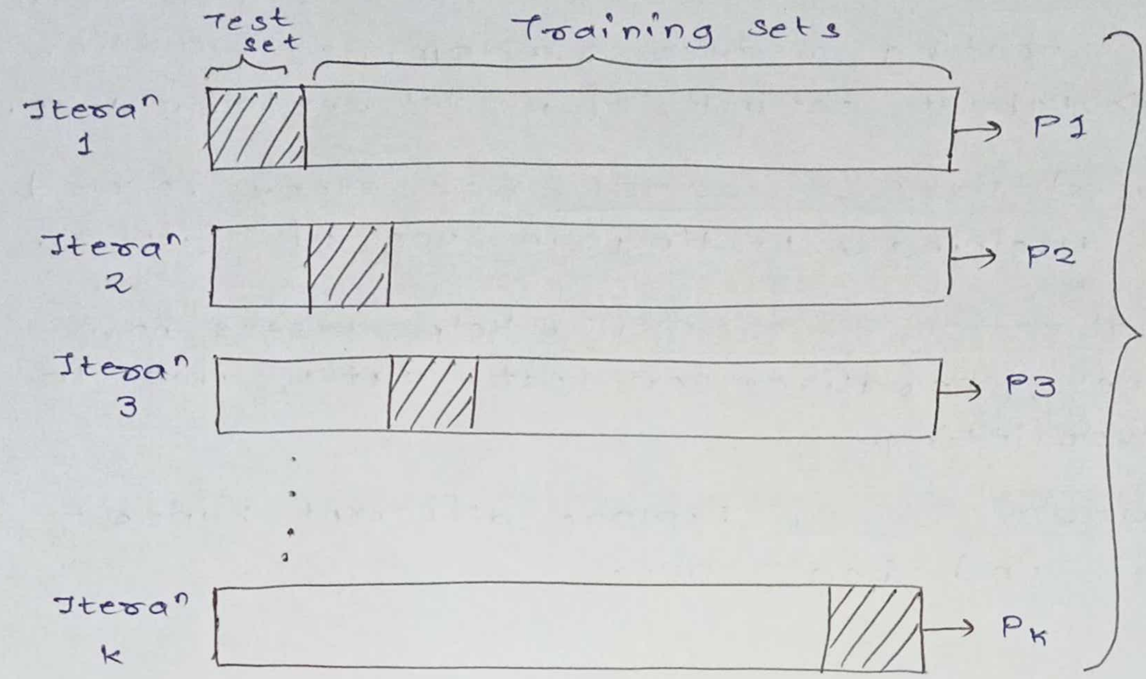
③ Randomness - Effectiveness depend on randomness.

④ Imbalance Datasets - For highly imbalanced data (1 class dominate all others) ⇒ should use stratified KFCV.

⑤ Data Leakage - when not handle properly lead data leakages.

1) K-Fold Cross-Validation (KFCV) :-

↳ Used to evaluate performance of predictive model.



Performance

$$P = \frac{1}{k} \sum_{i=1}^{k} P_i$$

⊙ Steps in KFCV :-

① Dataset Splitting: original dataset divided in K equally-sized, non-overlapping subsets / folds.

② Training & Testing: Model train & evaluate k times. In each itration, use 1 for testing & others trainin
Repeat process k-times.

~~③~~

③ Performance Evaluation: In each iteration, evaluation metrics (accuracy, precision, f1-score) recorded on each test set.

④ Aggregate Metrics - After k iterations, all performance metrics averaged to find single performance estimate.

Advantages of KFCV :-

1. Robustness: since use multiple sets for both training & testing, reduce randomness & outlier impact ⟹ Robust estimat of model performance.

2. Maximized Data Utilization: Bcz every record use for training & testing in one of K iteran.

3. Generalization assessment: Help assess how well model generalizes on test data, imp to avoid overfitting.

4. Hyperparam Tuning: Explore different hyper-params & find best one.

Choosing K value:

↳ Generally between 5 to 10.

↳ Smaller - less time, but more variations.

↳ Larger - Reduce variations, but more time.