Lecture 7

End-to-End ML Project

From Data Collection to Deployment

Part 2

Suman Samui

Checklist: 8 Main steps

- **Frame the problem & Look at a bigger picture**
- **Get the data**
- **Discover and visualize data to gain insights**
- **Prepare the data for Machine Learning Algos**
- **Explore different models and short-list the best one**
- **Fine-tune your models and combine them a greater solution**
- **Present your Solution**
- **Launch, Monitor and Maintain your system**

Discussion so far (please check part 1…)

Data preparation

- Data cleaning → Removing outliers and drop the missing values
- Feature selection (optional step)
- Feature engineering
  - Feature discretization of continuous feature values
  - Feature decomposition
  - Feature transformation
  - Aggregate features into promising new features
- Feature scaling → standardize or normalize features

Topics to be covered

- **Explore different Machine Learning (ML) models**
- **Fine-tune each model**
- **Short-list the promising models or the best model**
- **Present your solution**

Machine Learning Models

Regression models

- Linear regression
- Polynomial regression
- SVM regression
- Random Forrest regression
- Regularized linear models
  - Ridge
  - Lasso
  - Elastic net

Classification models

- Logistic regression
- Support vector machine
- Decision tree
- Random Forrest
- Extra trees

Artificial NN

- Fully connected network
- CNN
- RNN (LSTM and GRU)

Prediction for both Regression and Classification is possible using NN

Fine-tune any model: Why this is required?

- To stop underfitting and overfitting
- To achieve better generalization
- Bias vs. Variance trade-off

(Error on both train set and test should be small)

Hyperparameter tuning and model selection

Two general options

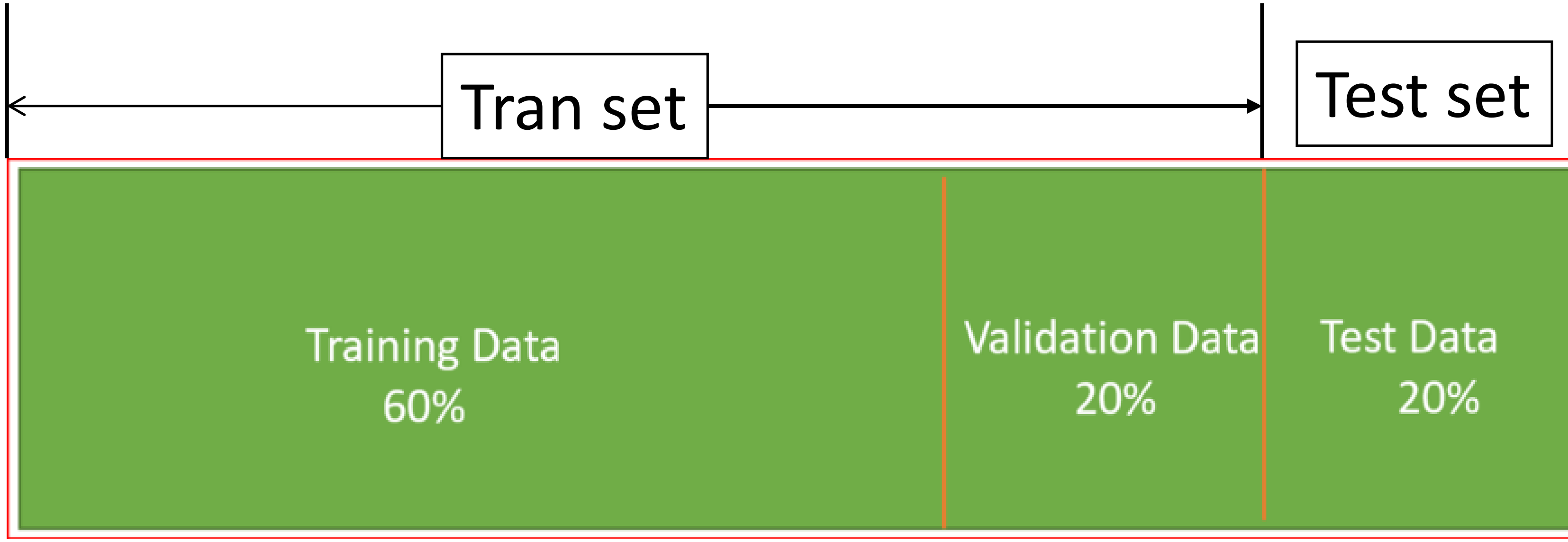
- Holdout validation
- K-fold Cross-validation

Holdout validation

Tip:

This data distribution depends on the size of the dataset

e.g.: if your dataset contains 10 Million instances, then holding out 1% data will be good enough.



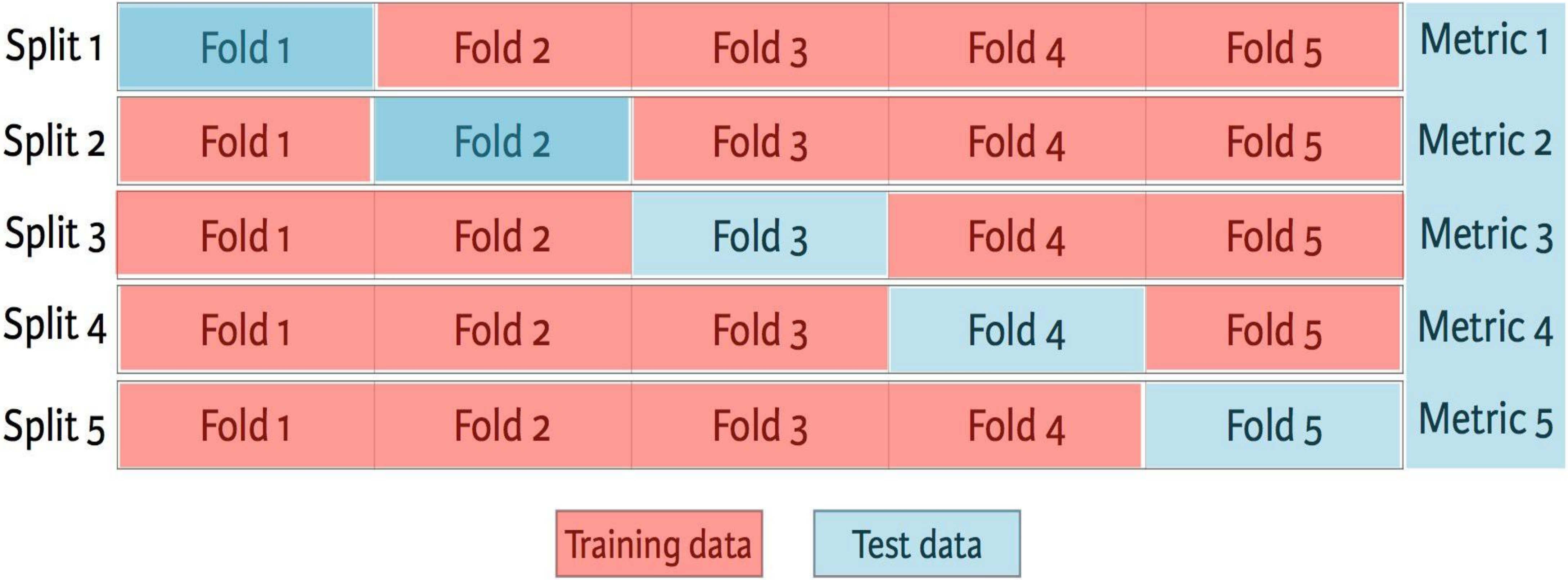
Held out validation set
(Dev set)

Problems of Holdout validation

- **Model evaluations will be imprecise and bad if the validation set is too small or too large.**
- **Due to *sample variability between training and test set*, the model may fail to generalize on test data. This leads to a low training error rate but a high test error rate.**

K fold cross-validation

Option → Scikit-Learn's *GridSearchCV*



Hold-out vs. Cross-validation

- The hold-out method is good to use when you have a very large dataset, you're on a time crunch, or you are starting to build an initial model in your ML project.

- As cross-validation uses multiple train-test splits, it takes more computational power and time to run than using the holdout method.

Evaluation

- **We got your tuned model (with set hyperparameters)**
- **You may try ensemble methods (optional) if you fine-tuned different ML models**
- **You must measure the performance of the model on the test set to estimate the generalization error and make evaluation based on judiciously chosen metrics associated with your ML problem.**

Present your solution

- **Explain why and how your solution achieves the objective**
- **Describe what worked and what didn’t**
- **List your assumptions and your system’s limitations**