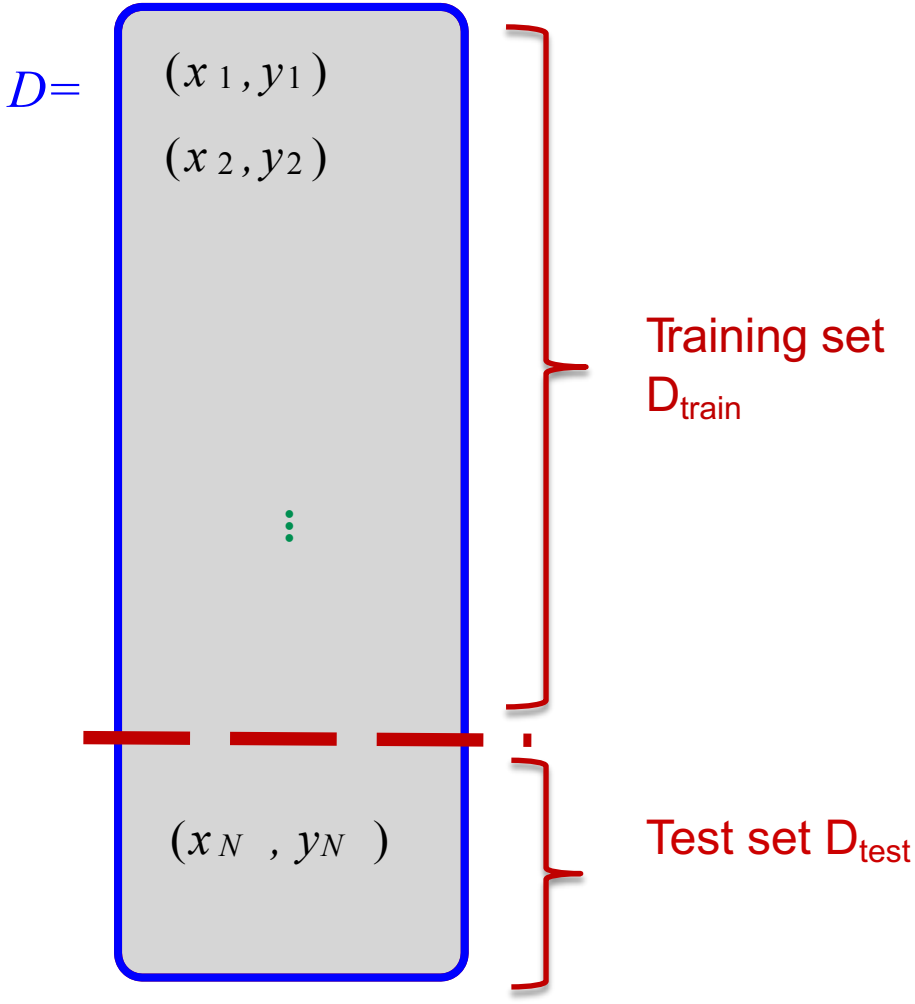
Lab 5

Cross validation

Lab5

- Objectives
  - Cross-validation: evaluating estimator performance
    - Train_test_split
    - Shuffle-Split
    - kFold
  - Hyperparameter search and model search

Simple train-test procedure



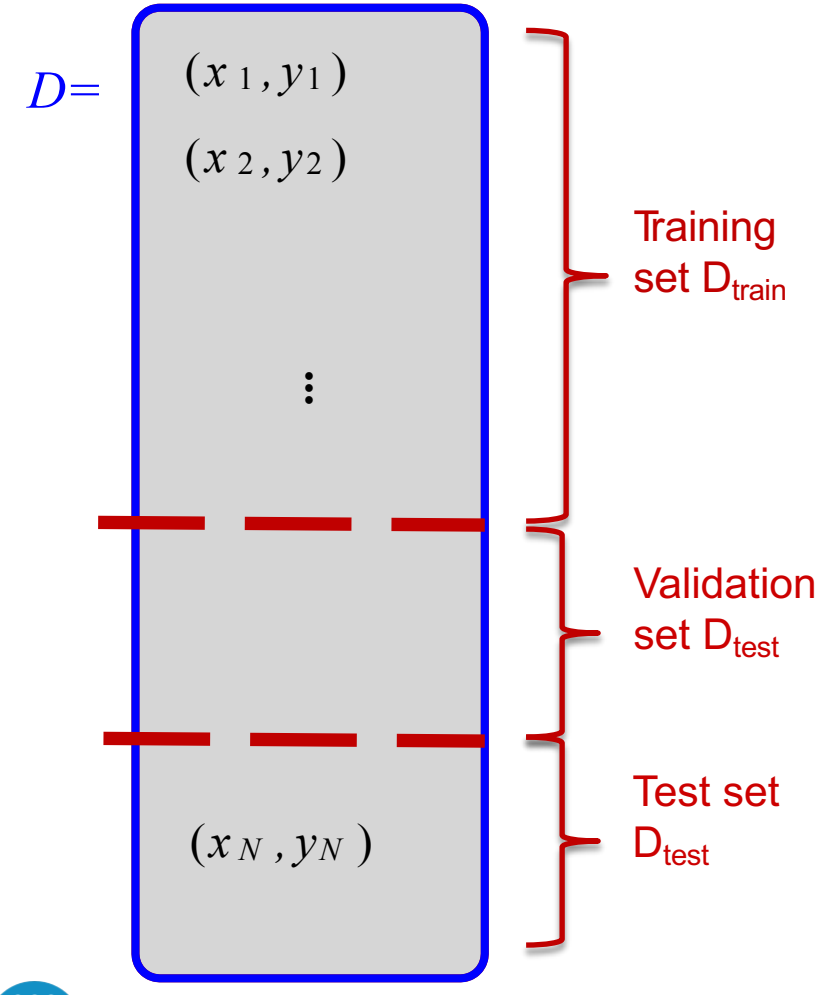
If there are no hyperparameters!!

- Provided large enough dataset D drown from p_{data}
- Arrange samples in random order
- Split dataset in two: D_{train} and D_{test}
- Use D_{train} to find the best predictor f
- Use D_{test} to evaluate the generalization performance of f

Hyperparameter tuning: train-val-test set

Make sure examples are in random order

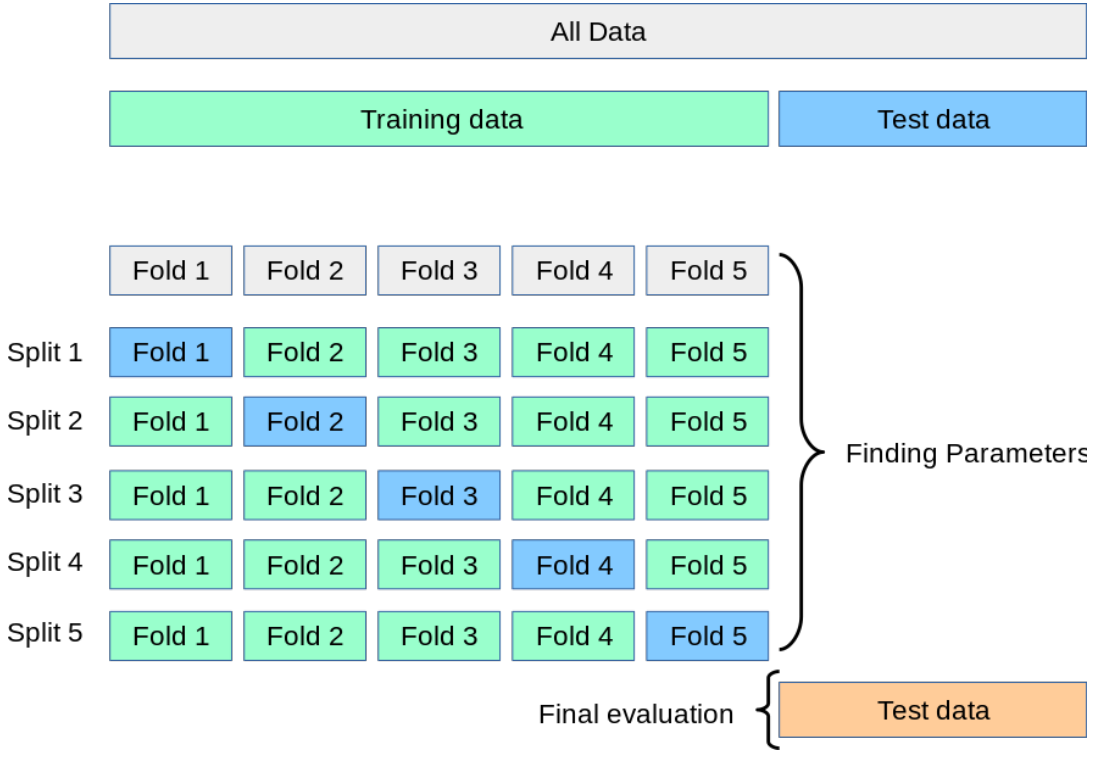
Split data *D* in 3: D_{train} D_{valid} D_{test}



- Data is split into 3 subsets
- The sets must be disjoint
  - Training and Validation set used only to find the right predictor (**optimize hyperparameters**)
    - Repeat training on D_{train} and evaluation on D_{valid} **for each value of hyperparameter**
    - Select the best performing value of hyperp. *
    - Retrain the model using training + validation data using the best hyperparameter
  - A Test set used to report the performance of the algorithm
- **Extension:** do N different splits of the train-val subsets, repeat training on train and average N val results for each hyperparameter

Hyperparemeter tuning with k-fold cross validation

- **For small datasets** (but enough data to keep an independent test set)
- Divide training data into k equal sized folds: train on k-1 folds,and validate on the remainder fold
- Perform k iterations (changing the validation fold).
- Compute average performance across the k iterations



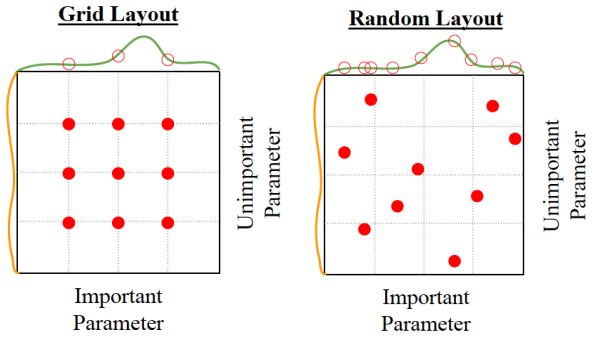
How many folds?

- **Practical rule of thumb:** 5-fold or 10-fold cross validation
- **LOOCV:** leave-one-out cross-validation, K=N (when the dataset is very small)

Hyperparameter search in scikit-learn

A search consists of:

- an estimator (regressor or classifier such as kNNClassifier)
- a parameter space
- a method for searching or sampling candidates
- a cross-validation scheme
- a score function.



`GridSearchCV:` exhaustively considers all parameter combinations

A GridSeachCV object internally iterates over a parameter grid and computes cross-validated scores for each hyper-parameter set.

`RandomizedSearchCV:` can sample a given number of candidates from a parameter space with a specified distribution. Implements a randomized search over parameters, where each setting is sampled from a distribution over possible parameter values. efficiency.

GridSearchCV

Main parameters:

- **estimator**: the model
- **param_grid**: list of parameters and list of values for each one
- **cv**: Cross validation procedure (none (**default 5-fold CV**), int (k in **k-fold** CV), **ShuffleSplit**, etc)

Example:

```
from sklearn.model_selection import GridSearchCV

hidden_layer_sizes = [(100,), (100, 100)]
activation = ["logistic", "relu"]
learning_rate_init = [0.001, 0.01]

grid_search = sklearn.model_selection.GridSearchCV(estimator= model,
param_grid={"clf__hidden_layer_sizes": hidden_layer_sizes,
            "clf__activation": activation,
            "clf__learning_rate_init": learning_rate_init},
cv=sklearn.model_selection.ShuffleSplit(n_splits=1, train_size=0.75, random_state=1)
)

grid_search.fit(X_train,y_train)

grid_search.cv_results_  results of each iteration, can be imported as a DataFrame
grid_search.best_params_  best set of parameters

grid_search.predict(X_train)
grid_search.predict(X_test)
```

Lab5

Hyperparameter search:

- Dataset: with data from breast biopsies, for the purpose of diagnosing breast cancer. For each patient, the data set contains 9 different attributes; we will use only 3

- Use knn and try to find the best k

  1. using the whole training data to train and select k
  2. splitting training-validation set (first manual split, then using train_test_split)
  3. repetitions of random splits and averaging (first manually and then using shuffle-split iterator)
  4. Kfold cross validation (first manually and then using Kfold iterator)

Lab5

Model selection:

- Dataset: pima, prevalence of diabetes in women of Pima Indian heritage, living near Phoenix, Arizona, USA. 7 features.
- Compare LDA, QDA and kNN
- Train and evaluate the model:
  - using the whole training data to train and evaluate
  - Using Kfold cross validation using Kfold iterator