Introduction to Data Science

Cross Validation

Gordon Anderson

Testing and Training Data Sets

- One goal of modeling is to be able to predict some outcome on newly observed data.
- We have a data set and expect there to be more data coming in- lots of it!
- In supervised machine learning, we use training and testing data sets. Each set contains an example of the true outcome, a number or a class label, and a set of predictors, or “features” for each row.
- The model “learns” how to relate the features to the true outcomes.
- The test data is “held out” of the learning to be used for testing how well the model has learned from the training set.

Cross Validation

- It is always a good idea to repeat an analysis (this *is* a scientific process after all) many times, especially when a random process is involved.
- When we have a fixed set of data to work with, we can take random samples of the data for repeated training and testing “runs”.
- We record how the model performed on each run and compare the results.
- This provides us with a view of how stable and accurate the model is.
- In other words, we can *validate* the model *across* many trials.

LOOCV

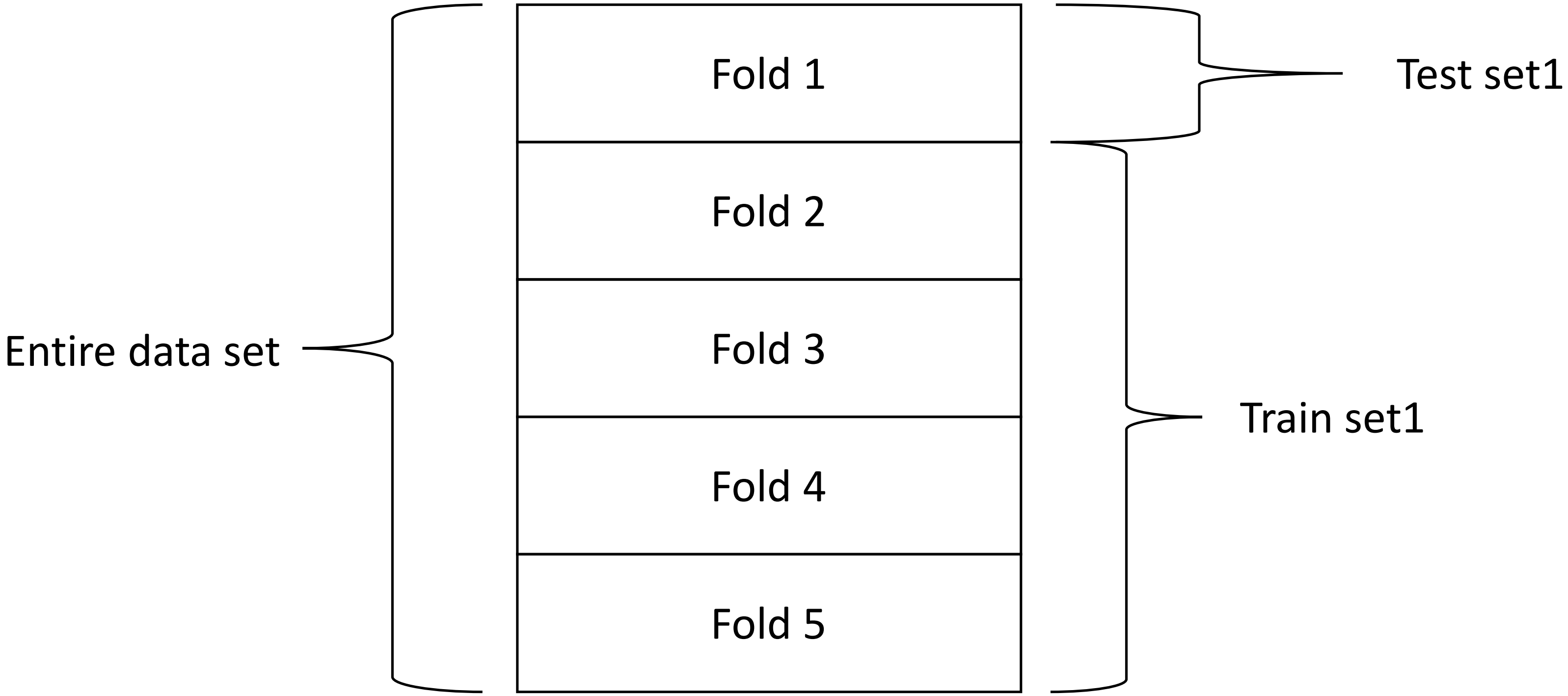
- LOOCV: “leave one out cross validation”.
- The idea is to take one row out of the data set for testing, train a model on the remaining data and test on the row left out.
- Do this for all rows of data.
- If the data has 1,250 rows, you would repeat the above 1,250 times. Each time you pick a different row for testing.
- Start with row 1 as the test set, then row 2, etc. until row 1,250.
- This provides a lot of performance data, but can be computationally expensive for large data sets.

K-fold Cross Validation

- Instead of LOOCV, pick more than one row to leave out for testing.
- Call the left out data a “fold”.
- Define the number of folds=K.
- Divide the data into K subsets.
- Use one fold for testing, the remaining folds for training.
- Repeat K times, each time holding out a different fold.
- Note that if K=1, we have LOOCV.

K-Fold C.V. Example

Let K=5 folds
Data is partitioned into 5 subsets, with no overlapping data.



Entire data set

Fold 1

Fold 2

Fold 3

Fold 4

Fold 5

Test set1

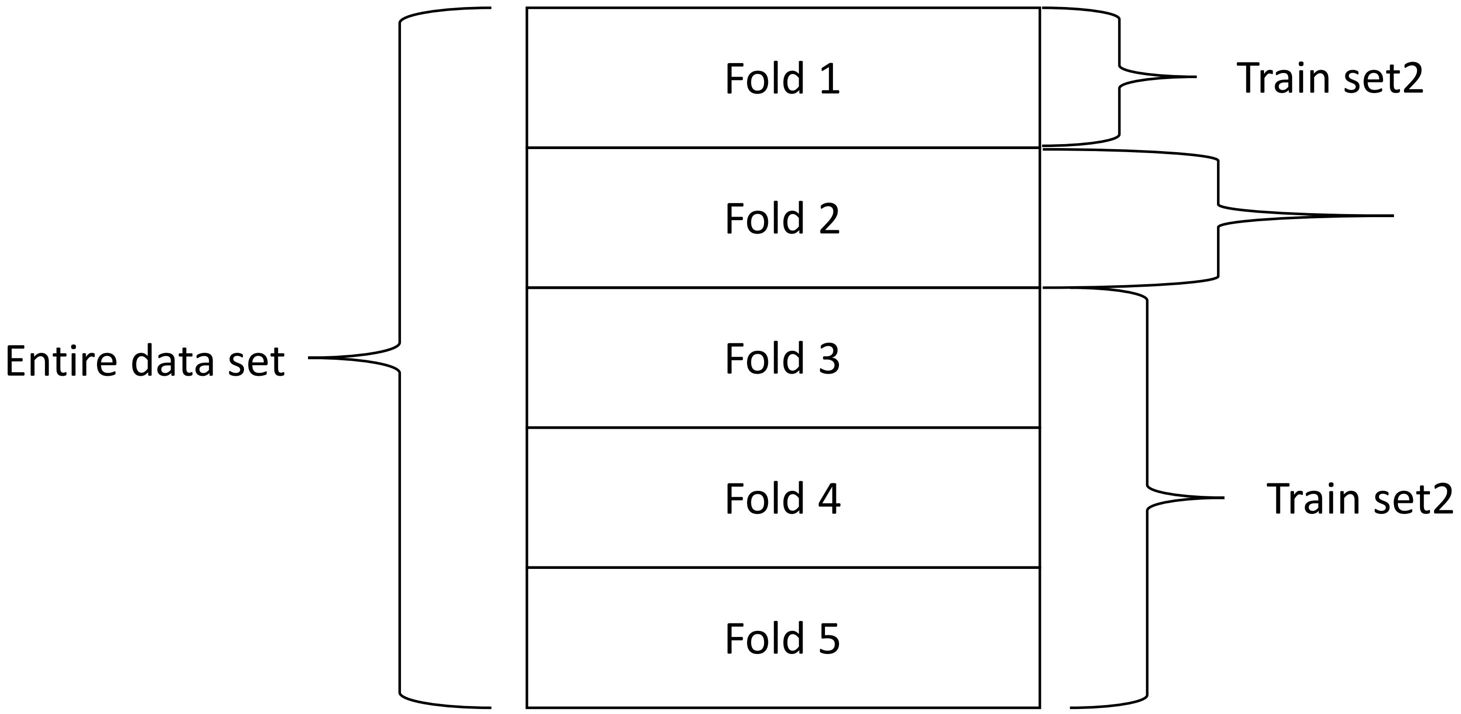
Train set1

Fit model to Train set1
Test model on Test set1
Record performance.

Repeat using a different fold each time.

K-Fold C.V. Example

Second iteration. Each fold gets a chance to be the test set.



Entire data set

Fold 1

Fold 2

Fold 3

Fold 4

Fold 5

Train set2

Test set2

Train set2

K-fold Cross Validation- algorithm

1. Given: data set, number of folds K, model, performance metric.
2. Partition data set into roughly k equal subsets.
3. For each subset *i* from 1 to K
    1. Test set <- data in fold *i*
    2. Train set <- all data excluding fold *i*
    3. Fit model to Train data set.
    4. Predictions <- model predicts outcomes given Test set.
    5. Calculate and save performance metric (predictions vs. true outcomes)
4. Analyze and p.resent performance data