Data partitioning and Evaluation of regression models

Yang Wang

College of Charleston

Prediction accuracy measures

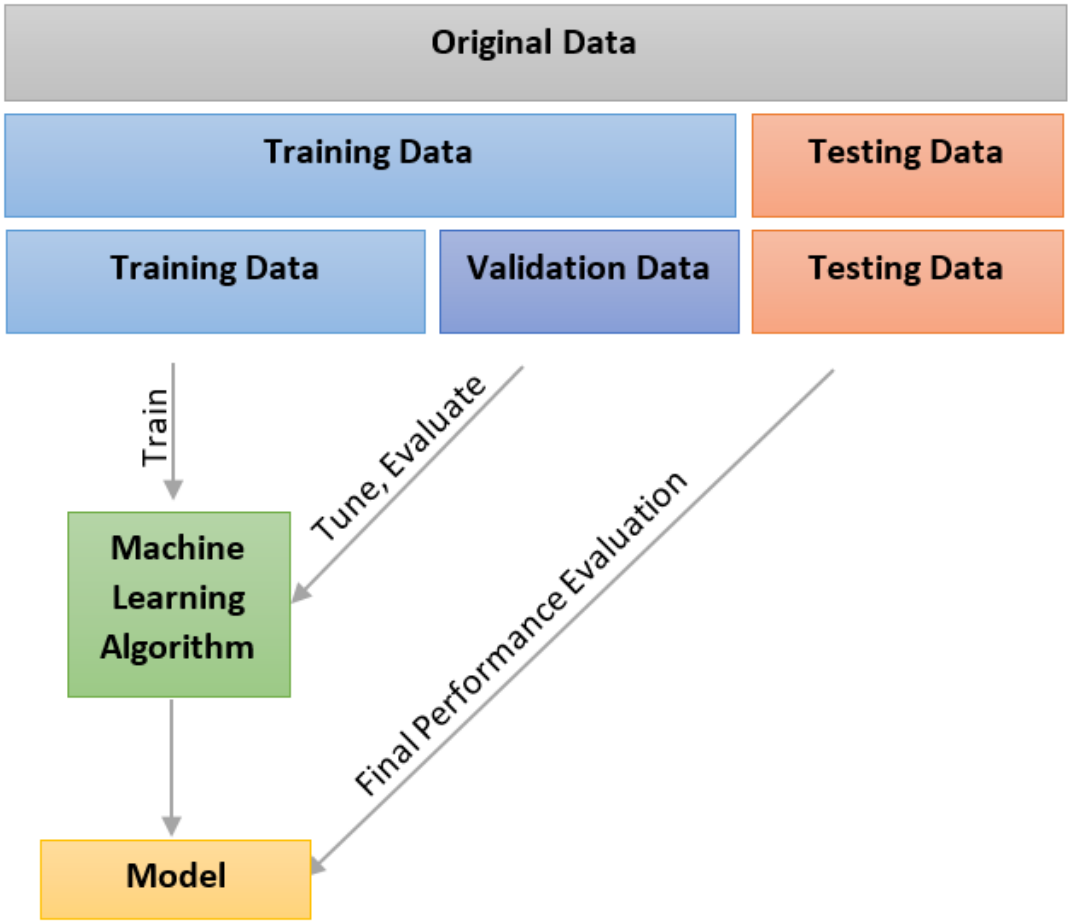
- y_i : response variable value of the i^{th} observation
- $\hat{y}_i$: predicted response value for the i^{th} observation
- $e_i = y_i - \hat{y}_i$: error for the i^{th} observation

Prediction accuracy measures

- **Mean Error** = $\frac{1}{n}\sum_{i=1}^{n} e_i$. This the average error. The disadvantage of this measure is that the negative errors cancel out the positive errors of the same magnitude. ($e_i = y_i - \hat{y}_i$)
- **MAE** (Mean Absolute Error) = $\frac{1}{n}\sum_{i=1}^{n} |e_i|$. This gives the magnitude of the average absolute error.
- **MPE** (Mean Percentage Error) = $100 \times \frac{1}{n}\sum_{i=1}^{n} e_i/y_i$. This gives the percentage score of how predictions deviate from the actual values (on average) taking into account the direction of error.
- **MAPE** (Mean Absolute Percentage Error) = $100 \times \frac{1}{n}\sum_{i=1}^{n} |e_i/y_i|$. This gives the percentage score of how predictions deviate from the actual values (on average).
- **RMSE** (Root Mean Square Error) = $\sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2}$. This is the square root of the error sum of squares. Note that, this is similar to the standard error of estimate in linear regression. The square root ensures that it has the same unit as the response variable.

Data partition

- Typically a model building process involves partitioning the data into three parts: training, testing and validation

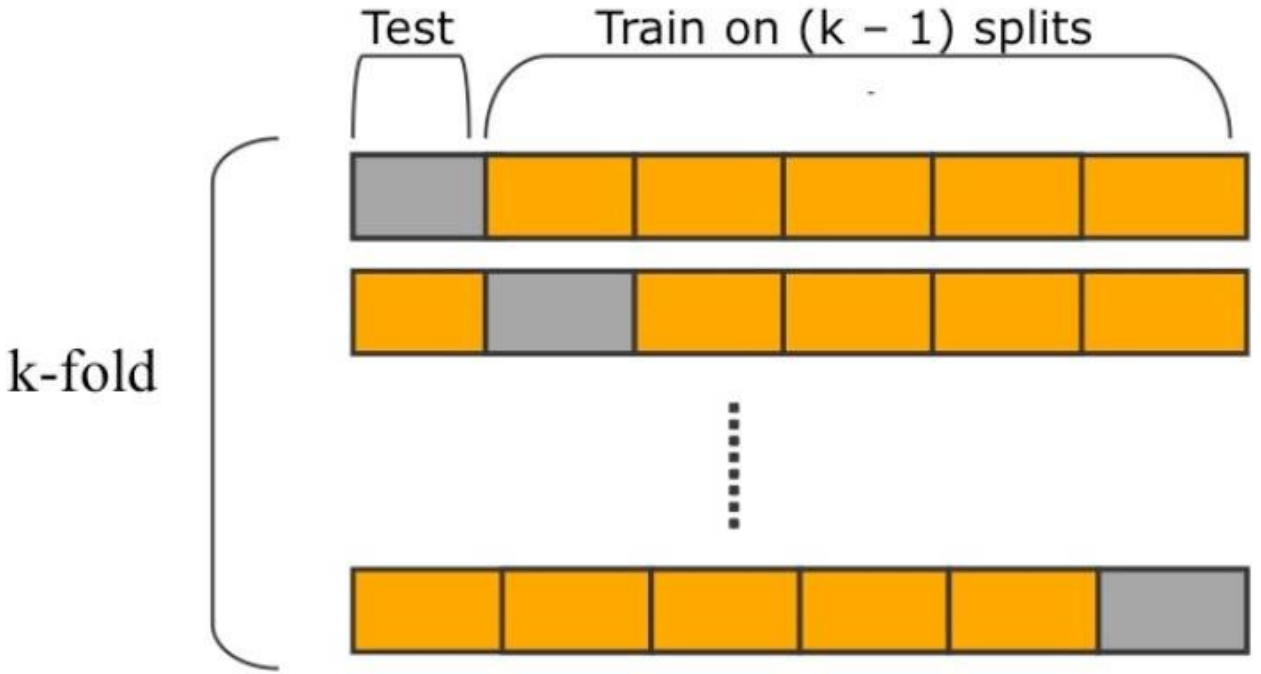


Data partition

- Training data: this is typically the largest partition containing the data used for building the various models that will be examined. Typically the same training data will be used to develop multiple models.
- Validation data: this is used for comparing the predictive performance of models and choosing the best one. In some algorithms this set will be used in an automated fashion to tune and improve the model.
- Test data: this is used for assessing the performance of the chosen model with new data. This is also called holdout or evaluation partition.

Data partition: Cross-validation

- Problem: different partitions will generate different models.
- Step 1: Partition the data into k "folds" or non-overlapping sub-samples.
- Step 2: Fit the model k times. Each time one of the folds is used as test (validation) data and the remaining $k - 1$ folds are used as training data.
- Step 3: Combine the model's prediction from each fold (when they were used as test (validation) data) for evaluating the performance of the model.



Data partition: Cross-validation

- Let the K parts be $C_1, C_2, ..., C_K$, where C_k denotes the indices of the observations in part k. There are n_k observations in part k; if N is a multiple of K, then $n_k = n/K$
- Compute Cross Validation MSE

$$\text{CV}_{(K)} = \sum_{k=1}^{K} \frac{n_k}{n} \text{MSE}_k$$

where $\text{MSE}_k = \sum_{i \in C_k} (y_i - \hat{y}_i)^2 / n_k$ and $\hat{y}_i$ is the fit for the observation i obtained from the data with fold k removed.

- Setting $K = n$ yields n-fold or leave-one-out cross-validation (LOOCV).

Data partition: Leave-One-Out Cross-validation (LOOCV)

- For each i in $1, 2, ..., n$ do the following:
  - Step 1: Create the training data set with observations $1, \ldots, i-1, i+1, \ldots, n$.
  - Step 2: Fit the model on the training data and test it on the i^{th} observation.
- Combine the model's prediction from each step for evaluating the performance of the model.

Data partition: Leave-One-Out Cross-validation (LOOCV)

- With linear or polynomial regression, an amazing shortcut makes the cost of LOOCV the same as that of a single model fit! The following formula holds:

$$\text{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)^2$$

where $\hat{y}_i$ is the ith fitted value from the original least squares fit, and h_i is the leverage (diagonal of the "hat" matrix; see book for details.)

Data partition

According to Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. ”The elements of statistical learning.” (2009), p.61)

- Combine the training and validation data to build the model using k-fold cross-validation or LOOCV.
- Then test the model on test data.

Relative advantage and disadvantages of different approaches related to data partition

- Basic approach: random partitioning of the entire data set into two (training and test) or three (training, validation and test) groups followed by fitting the model using traing data (or training and validation data) and testing it on the test data:
  - Advantage: simplicity
  - Disadvantage: heavily dependent on specific partition
- Cross Validation:
  - Advantage: all observations in training and validation are used for model building
  - Disadvantage: LOOCV is computationally intensive
  - Disadvantage: K-fold: K must be chosen carefully, lower value of K leads to a biased model and a higher value of K can lead to variability in performance metrics of the model

Practice assignment

- Suppose the response variable is y and regressor is x. A polynomial regression model of degree q is given by: $y_i = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_q x^q + \epsilon_i$.
- Polynomials are extremely useful as they allow for more flexible models.
- Consider the autompg data set. The objective is to forecast mpg using hp. Create a 60:30:10 partition of the autompg data set.
- Fit polynomial regression models of 2 and higher degree.
- Compare the MSE for each model in validation data.
- What degree seems to be an optimal fit?
- Calculate the MSE for the optimal model on the test data set.

Practice assignment

Note that, for this algorithm the validation set is used as a part of the training process (to find the degree of the polynomial). As a result, a test set should be used for evaluating model performance. Thus, the algorithm has the following steps,

- partition data into training, validation and test set.
- use training data for making prediction on validation data.
- use validation data for choosing the degree of the polynomial.
- use test for evaluating model performance with the chosen degree of the polynomial.