Model Selection and Evaluation

By: Behzad Asadi

Objective

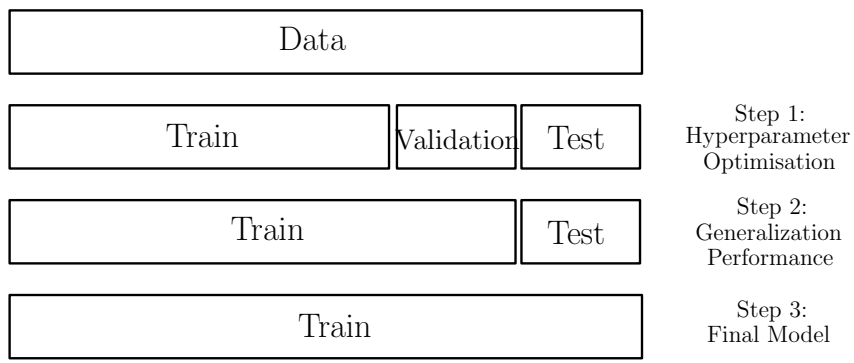
In model selection, our final objective is to find a model which performs the best over unseen (future) data. To do so, we need to accomplish the following three tasks.

- **Model Evaluation:** In this task, we want to estimate the generalization performance of our model.
- **Model Selection:** We here want to find the best values for the hyper-parameters of our model. Hyper-parameters are the internal knobs of the model which are not learned during the training process. This is to achieve the best generalization performance.
- **Algorithm Selection:** We here want to find the best learning algorithm for our problem. This is one step beyond the model selection as we select a family of models. In model selection, we just find the best values for hyper-parameters which means selecting a model from the family of models.

Train-Validation-Test Splitting

In this scheme, we divide our dataset into three non-overlapping parts: training, validation, and test datasets. Dividing the dataset into non-overlapping parts means sub-sampling without replacement. In sub-sampling, we use stratification to keep the distribution of samples in these three parts the same as the original one. This is particularly important for smaller datasets. This scheme consists of the following three steps to output the final model.

- **Step 1:** In the first step, we find the best values for the hyper-parameters of our model which result in the best performance over the validation dataset.
- **Step 2:** In the second step, we estimate the generalization performance of our model using the test dataset.
- **Step 3:** In the last step, we train the selected model using the whole available data, and output the model as the final model.



In Step 2, we can have an interval estimate for the generalization performance rather than just a point estimate. Considering accuracy as the performance metric, we use the proportion confidence interval to derive a confidence interval for the accuracy.

Proportion Confidence Interval

We here review the proportion confidence interval. This is because accuracy is the ratio of successfully labelled samples to the total samples in our test dataset. The number of successes follows the Binomial distribution. The Binomial distribution $\binom{n}{k}p^k(1-p)^{n-k}$ is the sum of n independent Bernoulli trials. The Bernoulli distribution has the mean of p, and the variance of $p(1-p)$. Therefore, the Binomial distribution has the mean of np, and the variance of $np(1-p)$.

The confidence interval for the population proportion is calculated using the mean and standard deviation (standard error) of the sampling distribution of the sample proportion $\hat{p} = \frac{k}{n}$. These two parameters are $\mu = p$ and $\sigma = \sqrt{\frac{p(1-p)}{n}}$.

$\hat{p}$ is considered as the point estimate of p. As we do not know the population proportion p, we use the sample proportion to have an estimated standard error. Considering a normal sampling distribution, the interval

$$\left[\hat{p} - z_{\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + z_{\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right]$$

is a $1-\alpha$ confidence interval for the population proportion p. For a 95% confidence interval, the value of $z_{0.025}$ is 1.96.

Accuracy Confidence Interval

In Step 2, we have a point estimate of the accuracy using

$$ACC = \frac{1}{n}\sum_{i=1}^{n} L(y_i, \hat{y}_i)$$

where y_i and $\hat{y}_i$ represent true and estimate labels respectively. The function $L(y_i, \hat{y}_i)$ is equal to one when $y_i = \hat{y}_i$, and is equal to zero when $y_i \neq \hat{y}_i$.

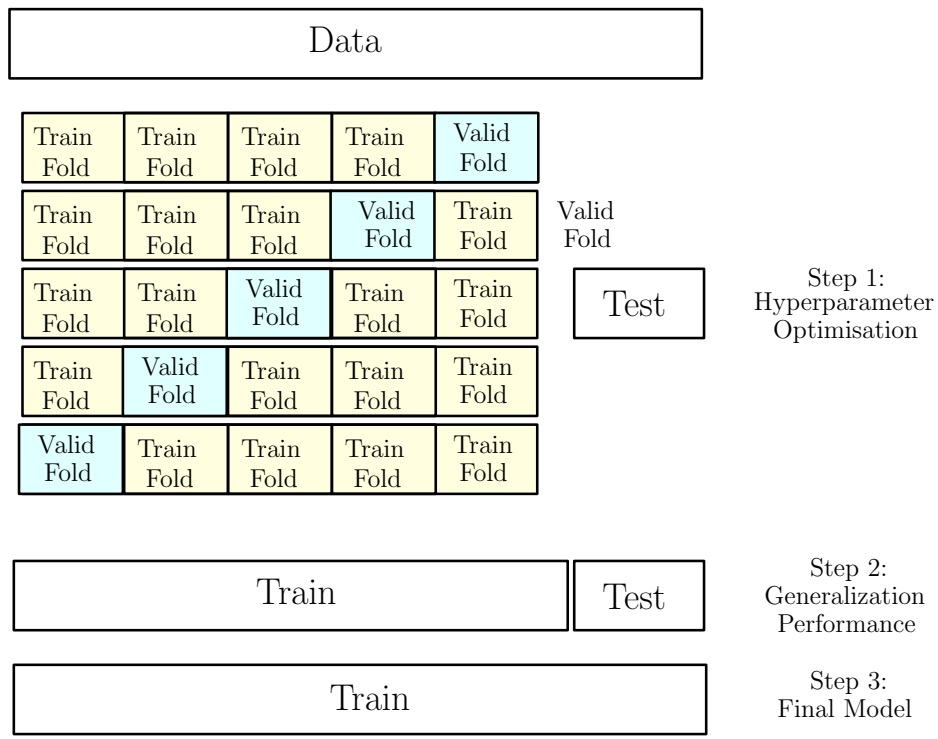
Following the proportion confidence interval, the interval

$$\left[ACC - 1.96\sqrt{\frac{1}{n}ACC(1-ACC)}, ACC + 1.96\sqrt{\frac{1}{n}ACC(1-ACC)}\right]$$

is a 95% confidence interval for the accuracy.

K-Fold Cross Validation

In K-Fold cross validation, we divide our dataset into K independent and equal size parts. Then we use $K-1$ parts for the training purpose and the remaining one part for the validation purpose. This process is repeated K times. The validation performance will be the average of the K obtained validation performances. This is useful when we have a small dataset as we can use all the data except the test part for both training and validation purposes.



Over-fitting and Under-fitting

Two problems in model fitting are under-fitting and over-fitting. Under-fitting happens when the model is too simple to model the data. Over-fitting happens when the model is too complex, and memorizes the data. Over-fitting leads to a high variance in the generalization performance (the model changes significantly for different training data). Under-fitting leads to a high bias in the generalization performance.
We need to detect whether our model suffers from under-fitting or over-fitting. This is done by comparing the performance of the model over the training and the validation data. The validation performance is lower bounded by the training performance. If the performance of the model is poor over both the training and the validation data, we are facing the under-fitting problem. If the performance of the model is good over the training data and poor over the validation data, we are facing the over-fitting problem. After detecting the problem, we need to find a way to solve these problems. Here is a list of possible solutions.

Solutions:

- Bagging
- Boosting
- Drop out (Neural Nets)
- Regularization
- Early Stopping
- More training data

References

1. Sebastian Raschka, Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning, https://arxiv.org/abs/1811.12808v2
2. Kevin P. Murphy, Machine Learning: A Probabilistic Perspective.
3. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, The Elements of Statistical Learning.