Deep learning

2.4. Proper evaluation protocols

François Fleuret

https://fleuret.org/dlc/

Learning algorithms, in particular deep-learning ones, require the tuning of many meta-parameters.

These parameters have a strong impact on the performance, resulting in a "meta" over-fitting through experiments.

We must be extra careful with performance estimation.

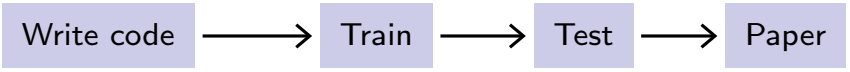
Running 100 times the MNIST experiment, with randomized weights, we get:

Worst	Median	Best
1.3%	1.0%	0.82%

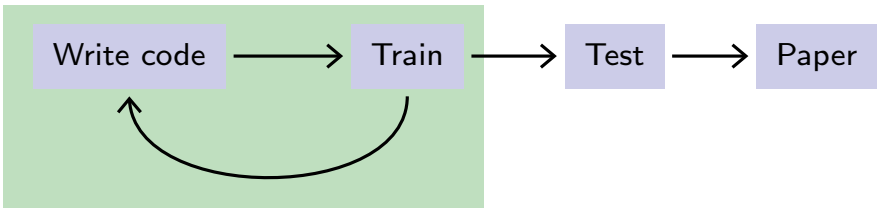
Notes

The meta-parameters can be related to the structure of the model (e.g. degree D of our polynomials) and the optimization process (e.g. the penalty coefficient ρ). There are many such meta-parameters to choose in any deep-learning setup.
We have seen previously the concept of over-fitting when a model has a high performance on the training set but a low performance on a test set, which shows that it does not generalize well on unseen data.
In the same spirit, we can have a "meta" over-fitting when one spends months in tuning the meta-parameters of the training itself until obtaining good results.
To illustrate the variability in the performance we get, we trained 100 times a classifier on the MNIST data set with the piece of code shown in the deck 1.2. "Current applications and success". The learning procedure is exactly the same, the meta-parameters are the same; the only difference is how the parameters of the networks are randomly initialized before training.
On the 100 runs, although the code to generate the results is exactly the same, the worst error rate is 1.3%, and the best rate is 0.82%, which is close to 20% better than the median rate. This can be problematic when only the best value is reported in a publication, or when a researcher stop investigating when "something good happens."

The ideal development cycle is

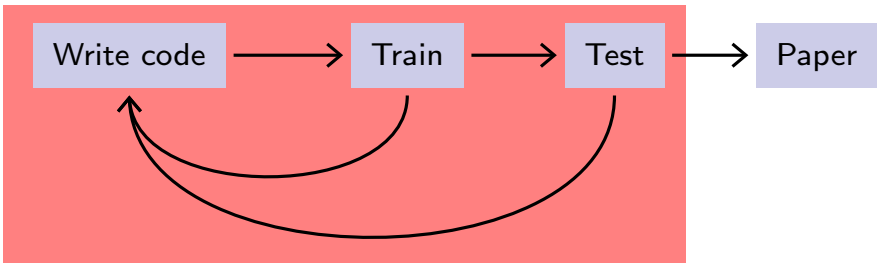


or in practice something like

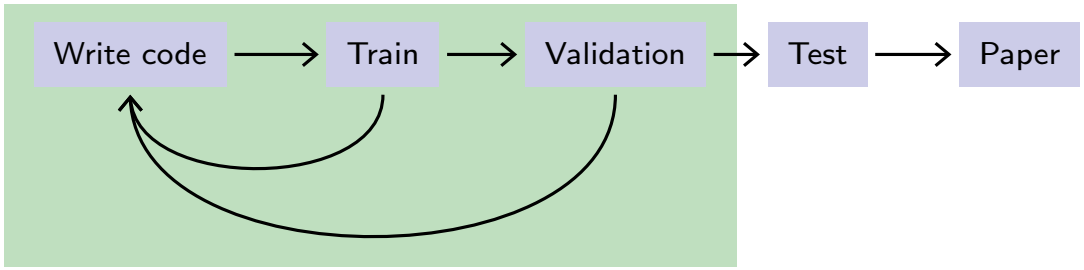


There may be over-fitting, but it does not bias the final performance evaluation.

Unfortunately, it often looks like



This should be avoided at all costs. The standard strategy is to have a separate validation set for the tuning.



Notes

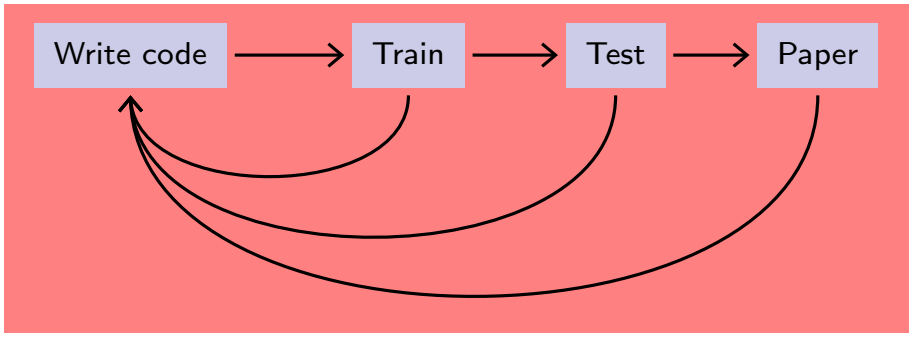
The validation set is also called the development set.

When data is scarce, one can use cross-validation: average through multiple random splits of the data in a train and a validation sets.

There is no unbiased estimator of the variance of cross-validation valid under all distributions (Bengio and Grandvalet, 2004).

Some data-sets (MNIST!) have been used by thousands of researchers, over millions of experiments, in hundreds of papers.

The global overall process looks more like



"Cheating" in machine learning, from bad to "are you kidding?":

- "Early evaluation stopping",
- meta-parameter (over-)tuning,
- data-set selection,
- algorithm data-set specific clauses,
- seed selection.

Top-tier conferences are demanding regarding experiments, and are biased against "complicated" pipelines.

The community pushes toward accessible implementations, reference data-sets, leader boards, and constant upgrades of benchmarks.

Notes

- "Early evaluation stopping": as soon as the model achieves good performance, stop investigating how and why it works, and publish the results;
- meta-parameter (over-)tuning: tuning again and again the meta-parameters and the architecture until reaching a good performance on the test set;
- data-set selection: cherry picking the [rare] data sets on which the method works without mentioning that it fails on other ones;
- algorithm data-set specific clauses: when the parameters are hard coded for a given data set;
- seed selection: report only the result for the particular random seed that yields a good performance (i.e. in our MNIST example, the seed which leads to 0.82% error rate).

A leader board is a web platform on which one submits one's results on a test set, the label of which are not given to the participant. The final evaluation of the performance is done by the leader.
Data set over-fitting over time can be avoided by constant updates of the data set across the years.

References

Y. Bengio and Y. Grandvalet. **No unbiased estimator of the variance of k-fold cross-validation**. Journal of Machine Learning Research (JMLR), 5:1089–1105, 2004.