DAGM Workshop 2013 on "*Unsolved Problems in Pattern Recognition*", Saarbrücken

Decision making under uncertainty: How informative is your algorithm with noisy inputs and internal computation errors?

Joachim M Buhmann
Department of Computer Science, ETH Zurich
jbuhmann@inf.ethz.ch

Precision versus robustness: Pattern recognition pursues the goal to find structure in given data. Usually, an objective or cost function is introduced to rank structures according to their suitability for solving a data processing task, but more and more often, algorithms are employed to select structures without optimizing any objective. Image segmentation, for example, requires a unique partitioning of pixels into groups, often with a spatial coherence constraint enforced. A variety of spectral clustering methods is at our disposal but also dynamical systems can solve this grouping problem. Principal component analysis for subspace identification demands to cut the eigenvalue spectrum when the signal-to-noise ratio drops below a critical threshold. Conceptually, pattern recognition algorithms optimize empirical risks based on given data and, therefore, they are susceptible to fluctuations. The empirical risk minimizer is most likely not the best choice for equally probable data with the same signal, but different fluctuations. For optimal **prediction**, therefore we should minimize the expected costs that depend on the true distribution function of the data, or at least the true conditional disctribution function of the output variables given the data. Since we are lacking this information, we are forced to minimize a regularized version of the empirical risk. What controls the proper amount of regularization and what regularization penalty should we use?

Many problems in pattern recognition display a combinatorial flavor like spectral clustering, matching or subspace selection and sparse coding. These problem definition share the property that the input space is much larger than the output space. A generative approach is not advisable in such a situation since we are completely satisfied with a sufficiently precise estimate of the posterior distribution even with an highly implausible distribution of the data distribution. Refering to the example of graph cuts with n vertices, the data distribution is defined over a space with dimension $\mathcal{O}(\exp(n^2))$ whereas the set of k-cuts has a cardinality $\mathcal{O}(k^n)$.

Model validation by information theory: To address the validation question for combinatorial optimization problems, we have developed an information theoretic approach to robust optimization [1, 2]. The output space of optimization problems is quantized or coarsened by appropriately placed Gibbs distributions. The set of solutions that a highly probably according to the Gibbs weighting is denoted as a *Generalization Set*. Where the Gibbs distributions are localized, is controlled by a set of randomly chosen transformations in analogy to Shannon's random coding strategy. The ability to identify a specific transformation under the influence of the fluctuations in the data defines a condition how much regularization should be used for inference. Preferable models with highly "peaked" Gibbs distributions can be identified at a higher noise level than fragile models.

The model validation concept does not necessarily depend on a cost function or a Gibbs distribution. Any algorithm that returns a generalitaion set of (still) admissible solutions at an intermediate computation step could be validated in the same way. We have demonstrated this generalization for sorting algorithms [4, 3] and for dominant set clustering. Currently, we explore the sensitivity of generalization sets for approximate spanning trees with the well-known algorithms by Prim and Kruskal and the reverse delete strategy. Preliminary results paint the following qualitative picture: Algorithms that commit too early in the computation to empirically optimal partial solutions turn out to be more fragile than algorithmic strategies with delayed decision making. Ruling out very poor edges in the approximate spanning tree problem yields less fragile generalization sets than committing early to empirically low weight edges.

Why should we be interested in robust algorithms? Primarily two reasons demand for a robustness design of algorithms:

- Many optimization problems have to process inconsistent input data and should be regarded as proxies for prediction problems. The reality test of solutions is performed on future data, not on the training data.
- Modern low energy hardware produces substantially higher soft error rates than current technology. Due to the demand for ever increasing computing power in the Big Data setting, we have to design algorithms that can

tolerate computational errors during most computation steps at the benefit of low energy dissipation.

It is fair to say that algorithmic design concepts of the last eighty years have mostly focussed on time and space resources, i.e., speed and memory consumption. The third dimension, how fragile an algorithmic concept reacts to internal (computation) or external (input) errors was not on the scope of the computer science community. The perspective of energy saving, "sloppy" hardware will fundamentally change the design space of algorithm research and it will move algorithm validation is the foreground.

What are the open problems in information theoretic model validation for robust algorithm design? There exist more open issues than answers so far; a non-exhaustive list is here:

1. The generalization of information theory to algorithm validation so far yields an upper bound for the identifiability of models. The converse of a lower bound is missing.

2. How relates the generalization capacity to the algorithmic complexity, i.e., are information theoretically robust algorithms computationally efficient?

3. How can we robustify well-known algorithmic principles like dynamic programming?

4. What is our ultimately interest in pattern recognition? Are we optimizing or are we localizing solutions?

Progress since the Heidelberg workshop on "Unsolved Problems in Pattern Recognition" In Heidelberg, I discussed the information theoretic framework and described its potential extension to algorithms. Since then, we have explored this avenue for sorting and clustering with Pelillo's Dominant Set Algorithm [5]. It is now also clear how the typicallity of instances have to be defined.

References

[1] J. M. Buhmann. Information theoretic model validation for clustering. In *International Symposium on Information Theory, Austin Texas*, pages 1398 – 1402. IEEE, 2010.

[2] J. M. Buhmann. Context sensitive information: Model validation by information theory. In *Mexican Conference on Pattern Recognition*, pages 12–21, 2011.

[3] L. Busse. *Information in orderings (learning to order)*. PhD thesis, # 20600, ETH Zurich, CH-8092 Zurich, Rämistrasse 6, 2012.

[4] L. M. Busse, M. H. Chehreghani, and J. M. Buhmann. The information content in sorting algorithms. In *International Symposium on Information Theory, Cambridge, MA*, pages 2746 – 2750. IEEE, 2012.

[5] M. Pavan and M. Pelillo. Dominant sets and pairwise clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1):167–172, 2007.