Bachelor project:
Initialisation strategies for dictionary learning

Supervisor: Karin Schnass

Many problems in signal/data processing and analysis can be efficiently solved if the signals in our class are sparse in a dictionary Φ. This means that every signal $y \in R^d$ in the class can be represented/approximated by a linear combination of a small (sparse) number of elements (atoms) of the dictionary, that is for $\Phi = (\phi_1, \ldots, \phi_K) \in \mathbb{R}^{d \times K}$ we have

$$y \approx \sum_{k \in I} \phi_k x_k = \Phi_I x_I \quad \text{with} \quad |I| = S \ll d. \tag{1}$$

In consequence it is desirable to have an automated way to learn such a sparsifying dictionary directly from a few training signals of the data class of interest. Given a set of training signals $Y = (y_1 \ldots y_N)$, the number of desired atoms K and the sparsity level S we want an algorithm that finds a decomposition into a dictionary Φ and a coefficient matrix X with S non zero entries per column,

$$Y \approx \Phi X. \tag{2}$$

To be more precise we have two requirements on a good dictionary learning algorithm, first that it is fast or computationally cheap and second that we have some guarantees that the algorithm will recover an underlying dictionary Φ if the data is known to be sparse in the dictionary. Currently there are mainly two promising directions. On one hand there are graph clustering algorithms and tensor methods, which have global convergence guarantees but are computationally very costly. On the other hand there are (alternating) optimisation schemes, which are computationally efficient, experimentally globally convergent but in the overcomplete case only have local convergence guarantees. One strategy to get an efficient and provably globally convergent algorithms is to first find a good initialisation using a graph clustering algorithm and to then use a fast alternating optimisation algorithm. One such initialisation strategy has been recently proposed in [1].
The goal of the project is to investigate the properties of this new initialisation algorithm in combination with ITKrM (Iterative Thresholding and K residual means), [2], and to compare it to random initialisation.

Tasks:

- Read [1] and implement Algorithm 3.
- Test the performance of Algorithm 3 on synthetic data based on the signal model in [1].
- Adapt Algorithm 3 to handle a unit norm signal model as in [2].
* Test the performance of the adapted algorithm in combination with ITKrM on noisy and not exactly sparse (synthetic) data and compare to random initialisation.
* Test the performance of Algorithm 3 (and ITKrM) on real data, e.g. images patches.

💣 Prove the equivalent of Theorem 19 in [1] for the adapted algorithm.

References

[1] S. Arora, R. Ge, T. Ma, and A. Moitra. Simple, efficient, and neural algorithms for sparse coding. *arXiv:1503.00778*, 2015.

[2] K. Schnass. Convergence radius and sample complexity of ITKM algorithms for dictionary learning. *arXiv:1503.07027*, 2015.