Computational optimization for tensor decompositions

organized by
Rasmus Bro, Michael Friedlander, Tamara G. Kolda, and Stephen Wright

Workshop Summary

In the past decade, there has been an explosion of interest in tensor decompositions as an important mathematical tool in fields such as psychometrics, chemometrics, signal processing, numerical linear algebra, computer vision, numerical analysis, data mining, neuroscience, and graph analysis.

Further advances depend critically on algorithms that are robust, accurate, numerically stable, and fast. Despite widespread interest, the computational tools available to practitioners have not changed dramatically for nearly four decades. State-of-the-art methods are based on simple alternating least squares (ALS); this approach is often slow to converge and there are few guarantees that it will converge to a useful solution. Some researchers have observed that more sophisticated optimization methods provide superior solutions and have presented preliminary computational results to support their claims.

The AIM workshop on tensor decompositions aimed at generating a fruitful collaboration between the tensor algorithm researchers and the optimization community. Thanks to a schedule of full-group talks and demonstrations, focused breakout groups, and a very engaged set of participants, the workshop turned out to be an inspiring and constructive meeting providing many new insights and starting many collaborations that we believe will help accelerate the pace of developments in this field.

Our focus in this workshop was more on the algorithmic and computational aspects of tensor decompositions than on the very rich theory associated with the area, though the participants included several people who are experts in the latter area.

Some of the most important issues that arose in the workshop are described below.

The Alternating Least Squares (ALS) Algorithm for fitting CANDECOMP/PARAFAC

In ALS, each factor in the tensor product are optimized in turn (while the others are held fixed) in order to best match the target tensor according to an element-wise sum-of- squares criterion. Thus, each subproblem is a linear least-squares problem, but one with unusual properties: it is typically highly overdetermined, and the cost of computing the right-hand side in the normal-equations formulation is typically much higher than the cost of forming the coefficient matrix. ALS is attractive because it exploits an obvious structure in current formulations; however, it can be slow to converge and unstable in many situations. Despite these obvious problems, ALS has remained the main underlying engine since the beginning of the field. We anticipate that it will be difficult to completely replace ALS at least in the

foreseeable future. Hence, at the workshop, there were several groups working on improving on this basic method and on improving our understanding of its behavior.

Speeding up ALS by Sampling

The most expensive operation in each ALS subproblem is computation of the right-hand side of the normal equations. One group at the workshop tried to reduce the cost of this operation by random sampling. In computing the matrix-vector product to form the right-hand side, only a subset of rows of the matrix are used. Different sampling schemes were tried (by modifying the Tensor Toolbox code) and the approach was found to be promising, with speedups of about 4 on a number of random problems of various dimensions. This appears to be a promising direction for future research. Another notable aspect of the sampling approach is that the noise it produces in the gradient may help in avoiding local minima (see below).

A somewhat different approach, also involving sampling and randomness, is to use stochastic gradient methods on the full problem (not in the ALS setting). Such methods have proved popular recently in other settings, for example machine learning and stochastic optimization, but their use here would require extension to the nonconvex case. There was not time during the workshop to investigate the approach, but we view it as a promising topic for follow-up work.

Convergence Behavior of ALS

One group tackled the issue of understanding convergence of ALS. Apart from the broader issue of local minima (discussed below), what can be said about the conditions under which ALS converges, and what is its rate of convergence under these assumptions? What is the connection to solving systems of equations? ALS works most of the time" in practice but can we find a better characterization of when this is true, for example conditions analogous to those that ensure that compressed-sensing recovery works? ALS iterates certainly converge to a limiting function value, because the sequence of function values is lower bounded by zero and it is monotonically decreasing. However, iterates often continue to change in large steps. Even worse, the accumulation points in some cases do not appear even to be local minima; rather they iterates appear to be stuck in a cycle. How should poor conditioning and ill conditioning of the least-squares subproblems be handled? What is the connection between the conditioning of the subproblems and poor convergence? What properties of the tensor lead to ill-conditioned subproblems? Can a more judicious choice of rank (i.e., number of columns in each matrix factor) improve performance of ALS, and how can a good rank be identified?

One line of analysis follows by recognizing that ALS is a block coordinate relaxation algorithm (i.e., a Gauss-Seidel method). The connection provides no panacea, but gives us a starting point.

Several possible angles of analysis were discussed during the workshop, particularly on the last day. There have been follow-up investigations after the workshop, but this area remains work in progress.

Modifications of ALS

One approach suggested for modifying ALS was the use of more elaborate line-search approaches. ALS traditionally steps to the minimum of each least-squares problem a greedy approach. However, there may be some benefit to over- or under-relaxing the steps in some systematic way. (Such techniques have proved to be useful in other contexts, e.g., solving sparse linear equations.) Another idea is to combine search directions in each of the factors and take a combined step. A minimizer along the combined search direction could be found by solving a higher-order scalar polynomial. Again, such approaches have proved useful in other optimization contexts. Other possible modifications include the use of damping (i.e., use of a Levenberg-Marquardt parameter) to improve the conditioning of the least-squares subproblems, which are often rank deficient.

Time did not permit serious progress to be made at the workshop on these issues, but we hope they can be addressed in the follow-up.

Local Minima

Being a nonconvex problem, the tensor decomposition problem may yield solutions that only locally minimize the misfit between the target tensor and the factors. Can we say something about the number of local minima? How deep, typically, are the wells containing local minima? If they are deep (as happens in other examples, e.g., energy potentials for protein folding) then it is hard to design algorithms that avoid them.

Can we incorporate techniques into the algorithms that make convergence to non-global minima less likely? For example, accepting some steps that increase the objective (as in simulated annealing) or using non-monotone techniques rather than the standard descent techniques are possibilities.

Some time was devoted to this topic during the meeting but not enough it remains an interesting topic for further work.

General Optimization Techniques

One group asked how much progress could be made by treating the tensor factorization problem as a general nonlinear optimization problem. They thus modeled the problem (and its nonnegative variants) by using AMPL and submitted problems to the NEOS Server for solution, using standard optimization software. This approach provides a baseline for performance of specialized methods. Preliminary indications were that the approach is successful but that algorithms that exploit the structure (which were the main focus of the workshop) are likely to me more efficient, particularly on large problems. The group was able to use the modeling power of AMPL to handle difficult problems without explicitly reformulating them. However, this is just a demonstration of the potential power of using more sophisticated optimization techniques. More work is needed to specialize the useful techniques for tensor model fitting.

Alternative Loss Functions

Most tensor decompositions are designed to minimize sum-squared error: implicit in this formulation is a Gaussian noise model for the observations. Other loss functions could be considered. If they are convex, then the subproblems arising in the alternating directions approach will also be convex (as in ALS, for the sum-of-squares loss function). At the workshop, new algorithms based on L1 fitting were developed and it was shown in practice that such algorithms produced results that were essentially unaffected by shot- noise and spurious data elements. In this sense they were robust, and contrasted quite dramatically with the sum-of-squares loss function which produces results that are seriously distorted by observation errors of this type.

Constraints in Compressed Representations

The technique of compression identifying and working in primary subspaces of a tensor, with smaller total dimension has been known and used for several decades, and is helpful in reducing computation time. What is less well understood is how to impose structure and constraints in the compressed representation. Specifically, it was now known how to impose the requirement of nonnegativity of the factors in the compressed setting. (As a related issue, PARAFAC2 has a hidden mode, but it has not been shown whether it is possible to impose constraints in that mode.) At the workshop, the use of AMPL (mentioned above) showed that showed that it is indeed possible to build nonnegativity into the compressed structure. Further work is needed to make the suggested approach practical but a proof-of-principle was obtained.

Testing new solutions

How should we go about creating data? The field desperately needs real datasets to test new methods on. The data may be from real-world problems or artificially generated. We need different types of problems as well those that are known to have troublesome local minima, degeneracies, etc. We also need large problems that are so big they dont fit on a standard computer. We need both dense and sparse problems. We are motivated by other communities that have compiled useful data sets. (The netlib set for LP and the SPARCO set for compressed sensing come to mind.) For our goal of engaging the optimization community on tensor problems, such data sets would allow all papers to use the same data and comparisons. It is also useful to understand the data by having a detailed explanation of what the data actually is, what it means, etc. To this end, it was decided to make a more elaborate web-site containing data sets with known problems and importantly with known solutions. Rasmus Bro will start this up.

Tensor completion

We considered the tensor analogue to matrix completion how to fill in missing values in your data when it is known to have underlying low-rank structure or something similar (e.g., low-rank plus noise, etc.). We can always matricize a tensor and use matrix completion techniques, but this approach ignores the multilinear structure and may be more expensive and less accurate.

One group also considered the theory of tensor completion. In the matrix case, we can say a lot about the nuclear norm of a matrix. The group considered extensions of the matrix nuclear norm in the tensor case. There are two variations. One is a theoretical construct that considers the minimum sum of weights of a rank-one factorization. The other is to average the nuclear norms of the tensor unfolded along each mode.

Many other subjects were proposed during the workshop but not discussed in detail due to lack of time and hectic activity in the above subjects. The most important of these are mentioned below. We mention in particular some of the questions that remain to be addressed in each category.

Manifold of Solutions

Due to permutation and scaling ambiguities, any CANDECOMP/PARAFAC tensor decomposition actually defines a manifold of possible solutions. A similar situation holds for the Tucker decomposition. This raises several questions: How we can isolate solutions on the manifold? What parametrizations of the manifold are useful?. Which of these parametrizations leads to each local minimum being a unique subspace? It is possible to remove the scaling by introducing constraints or regularizing the problem. In these cases, how are the manifolds represented in a way that's tractable? You have an equivalence class, as long as you have a point in the set. Is the set of solutions a manifold, or a set?

Rank

It is generally difficult to determine the numerical rank of a data tensor. It is well known that the determination of rank is NP hard and that low-rank approximations may not exist. What are practical approaches for determining the numerical rank of a tensor? The current approach is just to try different ranks until one has a good fit without over-fitting (an art more than a science). One idea that has been seen in the literature and was explored at the workshop was the idea of cross-validationi.e., leaving out some data and seeing how well the model fits the left out data to determine the rank.

Are there examples/classes of problems where we cannot compute the rank? What are the properties of such examples? For example, we might fail to fit a rank-5 tensor, but this does not assure that no rank-5 approximation exists the issue may be that the optimization routine simply failed to find it.

Convex relaxation

Can we use convex relaxations (or other norms) to find low-rank tensors? In the matrix case, we can relax the rank to a convex problem, using the nuclear norm. Can we similarly relax the tensor rank?

Sparsity and missing values

How do we handle sparse, missing value, nonnegative constrained problems? How do we do this faster than existing methods? How do we incorporate knowledge of problems into

development of new (better) methods? How do we incorporate knowledge of problems into development of new (better) methods?

Damping

How do we use regularization in the optimization problem? Does regularization keep you away from bad local minima? A damping term, such as that used by the Levenberg- Marquardt method, seems to generally speed up convergence, but how does it affect the solutions obtained?

How do we leverage problem characteristics in computing and/or choosing models?

Can we check/verify that multiway structure actually exists in the nerlying application, so that the tensor framework is appropriate? Sometimes you hit the jackpot, e.g. in the case of fluorescence, there is clear physical support for multiway analysis. In general, however, how do we know that the data contains such a structure? Physical understanding may be be lacking. In the case of social data, all bets are off! There should be rules of thumb for knowing if you are doing a good job; some of this appears in a paper by Richard Harshmann called How can I know its real?"

What kind of structure is of interest in the solution?

How can this be addressed in the objective/constraints? A number of people mentioned sparsity. Can sparsity be imposed on the factors? There is some previous work by Morten Mrup on sparse tucker, using a component-wise L1 norm to enforce sparsity, but more work needs to be done.

Conclusion

The goal of the proposed workshop was to discover novel optimization approaches for solving tensor decomposition problems and to spark new lines of research in this area by bringing tensor experts together with optimization specialists. We believe it was successful. The sponsored participants remained engaged and focused during the week. Unsolicited participants (such as Ben Recht, who spoke on the second day) enlivened the proceedings and contributed significantly to the success of the event. Connections were made and strengthened. We look forward to seeing how well the directions identified at the workshop pan out in the months and years ahead.

The organizers are most grateful to AIM for their support and involvement in the workshop at their excellent facility. Their well-grounded ideas for workshop structure were familiar to only a few participants; for others, the group-focused structure was an adventure, but an exciting one. We believe that the ultimate productivity of the workshop will be much higher as a result of this intensive, hands-on approach.