Convergence of Gradient Methods on Hierarchical Tensor Varieties

vorgelegt von
Mathematiker (Master of Science)
Benjamin Kutschan

von der Fakultät II - Mathematik und Naturwissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften
Dr. rer. nat.

genehmigte Dissertation

Promotionsausschuss:
Vorsitzender: Prof. Dr. Michael Joswig (TU Berlin)
Gutachter: Prof. Dr. Reinhold Schneider (TU Berlin)
Gutachter: Prof. Dr. Daniel Kressner (EPFL Lausanne)
Gutachter: Dr. André Uschmajew (MPI Leipzig)

Tag der wissenschaftlichen Aussprache: 18. Januar 2019

Berlin 2019

Contents

1 Introduction

This thesis is an exhaustive answer to the last five lines of a paper on the convergence of gradient methods on low-rank matrix varieties. The original paper "Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality" [1] describes an iterative method for non-linear optimization where each iterate lies on the variety of matrices of some bounded rank. The next iterate is computed by going along a gradient-related direction on the variety. [1] proves, that if the series of iterates possesses a cluster point, it is the limit and concludes in the following lines:

> *It would be important and interesting to extend the results to tensor varieties of bounded subspace ranks, e.g., bounded Tucker ranks, hierarchical Tucker ranks or tensor train ranks. As these varieties take the form of intersections of low-rank matrices, the results in this paper can likely be generalized in this direction.*

We confirm the intuition of the authors, that the result is indeed generalizable to all the suggested tensor formats: tensor train, Tucker and hierarchical. For the Tucker and hierarchical format we only sketch the proof ideas. We acknowledge the value of a possible future rigorous formulation of the proof for these two formats.

With great foresight [1] has created an abstract convergence result (see Lemma 6.1 in this thesis), that we can use in the generalization. This abstract convergence result encapsulates everything dealing with Łojasiewicz' inequality saving us from touching it, despite it being an interesting theory (see the beautiful 100 page original work in French [2]).

Of the three missing lemmata, that we are going to provide, the central one is the parametrization of the tangent cone of singular points of the tensor varieties. Chapter 3 will be concerned with getting a general overview of the existing theory and algorithmic methods related to tangent cones. It will enable us to compute the two smallest special cases. Chapter 4 will prove the general case, the proof technique being a rather technical application of the pythagorean theorem.

In Chapter 5 the new parametrization of the tangent cone is used to show, that it is possible to choose a sufficiently gradient-related (satisfying an angle condition) direction. This proof is facilitated by using tensor diagrams and we do not know how notation would be possible without it.

In Chapter 6 everything is being put together, including the last of the three lemmata, which shows the openness of the parametrization.

Wherever we saw the chance for a small detour or corollary, we have taken it. For example we try to visualize the smallest example of a matrix variety. We show that if a tensor variety is the intersection of matrix varieties, then the tangent cone of a tensor variety is the intersection of the tangent cones of the matrix varieties. Every tangent vector of a tensor variety is the first derivative of an analytic curve. We thank the anonymous referee for demanding Chapter 4.2, which inspects the main result from a different angle. The viewpoint as a global optimality condition is covered in Section 4.3. In Chapter 6.1.4 we cover the practical aspects of implementing a low-rank method that takes full advantage of the memory-efficiency of the covered tensor factorizations.

2 Introduction to tensors

In this chapter we introduce the notions of the tensor product of vectors and of vector spaces along with many examples. A matricization is a certain grouping and reordering of the indices of a tensor. The matricization allows to generalize the matrix product to tensors to produce the new notion of contraction. When working with contractions tensor diagrams simplify notation.

2.1 Informal introduction to the tensor product

We want to start with an example. Let $\mathbb{K}$ be one of the fields $\mathbb{C}$ or $\mathbb{R}$. Consider the two vectors

$$u := \begin{pmatrix} a \\ b \\ c \end{pmatrix} \text{ and } v := \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

both from $\mathbb{K}^3$. Then the tensor product of the two vectors is the tensor (or in this case: the matrix) in $\mathbb{K}^{3\times 3}$

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \otimes \begin{pmatrix} x \\ y \\ z \end{pmatrix} := \begin{pmatrix} ax & ay & az \\ bx & by & bz \\ cx & cy & cz \end{pmatrix} \tag{1}$$

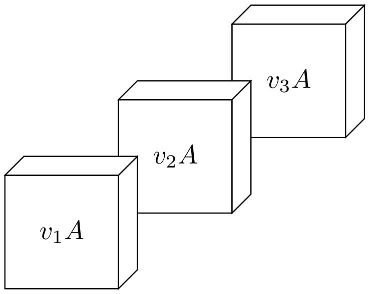
which in this example is equal to the matrix product of u and v^T

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \begin{pmatrix} x & y & z \end{pmatrix} .$$

The entries of the tensor are the products of all possible combinations of an entry from u and an entry from v. We can extend this idea for example to the tensor product of a matrix with a vector. This is shown in Figure 1. The ordering of the result into a cube (or into a matrix as in the previous example) is essential as we will see soon. We can write all the resulting coefficients in one very long vector, thus identifying $\mathbb{K}^{n\times m}$ and $\mathbb{K}^{n\cdot m}$. The number of coefficients of the result is the product of the numbers of the coefficients of the two factors.

Figure 1: tensor product of matrix with vector

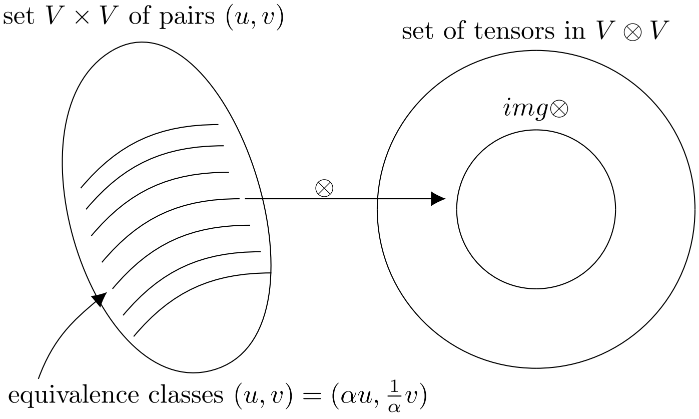
$$A \otimes v = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \otimes \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} =$$



One question arising from looking at equation 1 is: What is the difference between the pair (u, v) and the tensor product $u \otimes v$? Or what is the difference between the set of pairs $\mathbb{K}^3 \times \mathbb{K}^3$ and the set of tensors (in this example: the set of matrices) $\mathbb{K}^{3\times 3}$? First of all, the tensor product

$$\otimes : \mathbb{K}^3 \times \mathbb{K}^3 \to \mathbb{K}^{3\times 3}$$

Figure 2: tensor product defined bijectively


set $V \times V$ of pairs (u, v)
set of tensors in $V \otimes V$
$img\otimes$
$\otimes$
equivalence classes $(u, v) = (\alpha u, \frac{1}{\alpha} v)$

assigns a unique tensor w to every pair (u, v). But not every tensor w is the tensor product of two vectors. A counterexample is

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

In other words, the tensor product $\otimes$ is not surjective.

So are there essentially more tensors than there are pairs? To highlight the significance of this question, consider the example of the tensor product $\otimes : \mathbb{K}^2 \times \mathbb{K}^2 \to \mathbb{K}^{2\times 2}$ where the pairs consist of 4 values and the matrices, too. Here the image of $\otimes$ is also less than the full matrix space. However, is $\otimes$ at least injective, so that we could construct the bijection $\mathbb{K}^2 \times \mathbb{K}^2 \to \operatorname{img} \otimes$? The answer is no, because $u \otimes v = (\alpha u) \otimes (\frac{1}{\alpha} v)$ for all $\alpha \in \mathbb{K}$. Fortunately in this example these are the only pairs mapping to the same tensor. To get an overview we can draw Figure 2. Defining the equivalence relation

$$(u, v) \sim (u', v') \Leftrightarrow \exists \alpha \in \mathbb{K} : u' = \alpha u \text{ and } v = \alpha v'$$

we can define the injective tensor product

$$\otimes : \mathbb{K}^3 \times \mathbb{K}^3/_\sim \to \mathbb{K}^{3\times 3}$$

or the bijection

$$\otimes : \mathbb{K}^3 \times \mathbb{K}^3/_\sim \to \operatorname{img} \otimes$$

But what are all the other tensors, which are not in the image of $\otimes$? As we have quietly defined the codomain of $\otimes$ to be a vector space, every finite sum of tensors from the image of $\otimes$ must be included. More formally, the tensor space must contain the closure of $\otimes$ under addition. In fact we can define a basis of the tensor space that consists only of elements from $\operatorname{img} \otimes$. Given bases of $\mathbb{K}^n$ and $\mathbb{K}^m$ the tensor product of all possible pairs of one basis element each is a basis of $\mathbb{K}^{n\times m}$.

Already in the case of the example above, the image under $\otimes$ can be parametrized by only 6 numbers. Much less than the 9 numbers needed for the whole tensor space. The methods described in this thesis will make extensive use of the fact that $\operatorname{img} \otimes$ is much smaller than $\mathbb{K}^{n\times m}$. The image of $\otimes$ is also called the set of tensors of rank at most 1. A tensor that can be written as a sum of no more than r rank-1 tensors is said to have rank r.

2.2 Formal introduction to tensor spaces

2.2.1 Order 2 tensor spaces and tensor product

As we have already noted in the informal introduction, $\mathbb{K}^{n \cdot m}$ without any additional structure is not sufficient as definition of a tensor space. How can one define

$$\begin{pmatrix} 1 & ... & 1 \\ & & \\ & & \end{pmatrix}$$

to have rank 1 but

$$\begin{pmatrix} 1 & 0 & ... & ... & 0 \\ 0 & 1 & 0 & ... & 0 \\ ... & & & & \\ 0 & ... & ... & 0 & 1 \end{pmatrix}$$

to have rank n without regarding the ordering of the indices? The first matrix has as many ones as the second matrix. It contains the same entries, just permuted. A common way to fix this ambiguity is to include the map

$$\otimes \mathbb{K}^n \times \mathbb{K}^m \to \mathbb{K}^n \otimes \mathbb{K}^m : (x, y) \mapsto z$$

with $z_{ij} := x_i \cdot y_j$ in the definition of the tensor space $\mathbb{K}^n \otimes \mathbb{K}^m$ of $\mathbb{K}^n$ and $\mathbb{K}^m$.

2.2.2 Lexicographic order

Sometimes - for example when implementing an algorithm - it is important to define an order of the multi-indices. One popular way is to order them lexicographically. Having an order on I and on J we can identify I with $\{1, 2, ...n_1\}$ and J with $\{1, 2, ...n_2\}$. Then the multi-indices from $I \times J$ can be lexicographically ordered as

$$(1,1), (1,2), ..., (1, n_2), (2,1), ..., (2, n_2), ..., (n_1, n_2)$$

just as the ordering of words in a non-digital dictionary or the phone book. Compare with Section 3.4.1.

2.2.3 Formal definitions

The tensor product of finite-dimensional vector spaces $\mathbb{K}^n$ can be defined in the following way:

Definition 2.1. Let $\mathbb{K}^{n_1}$, ..., $\mathbb{K}^{n_d}$ be vector spaces over the field $\mathbb{K}$. Defining $I_n := \{1, ..., n\}$, we can define $\mathbb{K}^n := \{I_n \to \mathbb{K}\}$ as a set of functions from the index set I_n to $\mathbb{K}$. For $v \in \mathbb{K}^n$ we introduce the notation $v_i := v(i)$. Then define the tensor product as

$$\mathbb{K}^{n_1 \times ... \times n_d} := \mathbb{K}^{n_1} \otimes ... \otimes \mathbb{K}^{n_d} := \{I_{n_1} \times ... \times I_{n_d} \to \mathbb{K}\}.$$

The notation for accessing elements of the tensor $T \in \mathbb{K}^{n_1 \times ... \times n_d}$ is

$$T_{i_1, ..., i_d} := T(i_1, ..., i_d).$$

Define the map

$$\otimes : \mathbb{K}^{n_1} \times ... \times \mathbb{K}^{n_d} \to \mathbb{K}^{n_1 \times ... \times n_d}$$

by

$$v_1 \otimes ... \otimes v_d : (i_1, ..., i_d) \mapsto v_1(i_1)...v_d(i_d)$$

using the notation $v_1 \otimes ... \otimes v_d := \otimes(v_1, ..., v_d)$.

When considering bilinear maps

$$\varphi : V \times W \to X$$

the tensor product $V \otimes W$ is the *biggest* possible image of any of these maps. This is formalized in the following definition, which works for infinite-dimensional vector spaces as well.

Definition 2.2. Let V and W be any two vector spaces over the same field. Then T is called the tensor product of V and W if there is a bilinear map $\otimes : V \times W \to T$, such that any bilinear map $\varphi : V \times W \to X$ into any vector space X can be written as a composition $\varphi = \psi \circ \otimes$.

Defining the tensor product this way also works for infinite dimensional vector spaces V and W.

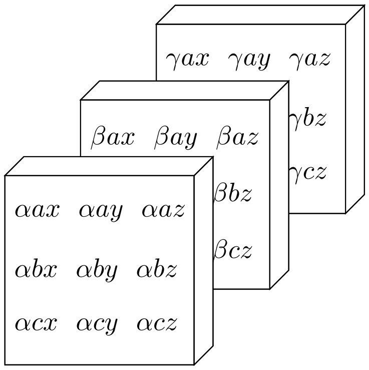
Figure 3: abstract definition of the tensor product

$$\begin{tikzcd}
V \times W \arrow[d, "\otimes"'] \arrow[dd, bend right=60, "\forall \text{ bilinear } \varphi"'] \\
T \arrow[d, "\exists\psi"'] \\
X
\end{tikzcd}$$

2.2.4 Higher order tensor spaces

Figure 4: tensor product of 3 vectors

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \otimes \begin{pmatrix} x \\ y \\ z \end{pmatrix} \otimes \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} ax & ay & az \\ bx & by & bz \\ cx & cy & cz \end{pmatrix} \otimes \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} =$$



γax γay γaz
γbz
γcz
βax βay βaz
βbz
βcz
αax αay αaz
αbx αby αbz
αcx αcy αcz

What we have described so far was the special case of order 2 tensors, also called matrices and well known to the reader. Applying the tensor product iteratively can produce tensor spaces of higher order - the subject of this thesis.

Example 2.1. See Figure 4 for a picture of a $3 \times 3 \times 3$ tensor product.

The tensor product is associative. Not even the (lexicographic) order is affected when multiplying in a different order.

Definition 2.3. We can write

$$\mathbb{K}^{n_1} \otimes ... \otimes \mathbb{K}^{n_d} = \mathbb{K}^{n_1 \times ... \times n_d}$$

for the set of *order* d tensors of *dimensions* n_1,...,n_d.

2.3 Examples of tensors appearing in applications

2.3.1 Polynomials as tensors

In this section, we will show three examples, of how sets of polynomial basis functions can be represented as tensors. To fully use the potential of these examples to parametrize the set of multi-dimensional polynomials, we will need the chapters 2.5 and 2.6 about matricization and contraction of tensors. We will come back to the examples in these chapters.

Example 2.2. The monomial basis of the set of homogeneous polynomials of degree d in the variables x and y can be represented as the tensor

$$\underbrace{\begin{pmatrix} x \\ y \end{pmatrix} \otimes ... \otimes \begin{pmatrix} x \\ y \end{pmatrix}}_{d \text{ times}}.$$

In this example the tensor product multiplies modules over commutative polynomial rings rather than vector spaces. The tensor product of modules is defined in exactly the same way as the tensor product of vector spaces. In the special case of degree 3 polynomials the above product equals

$$\left(\begin{pmatrix} x^3 & x^2y \\ x^2y & xy^2 \end{pmatrix}, \begin{pmatrix} x^2y & xy^2 \\ xy^2 & y^3 \end{pmatrix} \right).$$

Note the redundancy. The terms x^2y and xy^2 appear 3 times each. And in general there are only $d+1$ of these monomials. Thus this tensor product enlarges the number of parameters rather than reducing it.

The next example has the advantage of not containing any redundant terms and including all lower degree polynomials.

Example 2.3. A basis for the set of polynomials of degree bounded by n in each variable can be written as the matrix

$$\begin{pmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^n \end{pmatrix} \otimes \begin{pmatrix} 1 \\ y \\ y^2 \\ \vdots \\ y^n \end{pmatrix}.$$

for maximum degree 2 this reduces to

$$\begin{pmatrix} 1 & y & y^2 \\ x & xy & xy^2 \\ x^2 & x^2y & x^2y^2 \end{pmatrix}.$$

The upper left triangle of this matrix is a basis for the polynomials of total degree bounded by 2.

The next example somewhat combines the two previous ones. It succeeds in including exactly the polynomials of bounded total degree, but for the price of redundancy.

Example 2.4. A basis for the set of polynomials in n variables of total degree at most d is given (redundantly) by the tensor

$$\underbrace{\begin{pmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \otimes \ldots \otimes \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}}_{d \text{ times}}.$$

For polynomials of maximum degree 2 in x and y this reduces to

$$\begin{pmatrix} 1 & x & y \\ x & x^2 & xy \\ y & xy & y^2 \end{pmatrix}.$$

The upper right triangle already contains all the information.

The last example is motivated by the work of Sharir, Shashua and Cohen [3] on arithmetic circuits with applications in artificial intelligence. They map small patches (e.g. patches of 5 × 5 pixels in size) of a photograph to vectors by some function recognizing low-level features such as edges. So to each patch i they assign a vector $(x_{i,1}, ..., x_{i,n})$. Then they construct a monomial basis for a polynomial in all the variables $x_{i,j}$ in the following way.

Example 2.5. The tensor

$$\begin{pmatrix} 1 \\ x_{11} \\ x_{12} \\ \vdots \\ x_{1n} \end{pmatrix} \otimes \begin{pmatrix} 1 \\ x_{21} \\ x_{22} \\ \vdots \\ x_{2n} \end{pmatrix} \otimes \ldots$$

contains the monomials

$$1,\ x_{11},\ x_{21},\ x_{11}x_{21}, \ldots$$

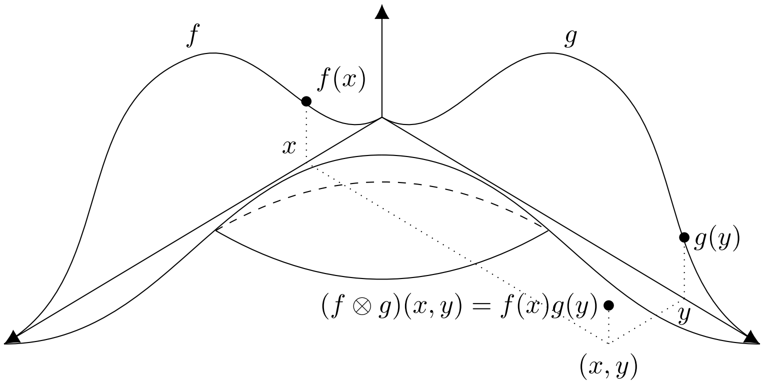
but for example not the monomial

$$x_{11}x_{12}.$$

In [3] small patches of the original image are mapped to the vectors $(1, x_{11}, x_{12}, ...)^T$ in a way that produces sparse vectors. Each of the x_{ij} represents a feature (e.g. a horizontal edge, vertical edge, color, ...). One tries to not identify a horizontal edge and a vertical edge in the same patch, thus the sparsity. The monomials missing in the tensor of the example are the ones that become 0 if the vectors are maximally sparse, that is if they contain only one non-zero element apart from the leading 1. The goal of Shashua and Cohens work is, to find coefficients of this polynomial, such that when the polynomial is evaluated at the transformed patches of a photograph, the value indicates the content of the photograph.

2.3.2 Approximating multivariate functions

Figure 5: tensor product of functions



Considering functions as elements in an infinite-dimensional vector space we can define the tensor product of functions by narrowing Definition 2.2 or by the following construction: Let $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$ be two one-dimensional functions. The tensor product $f \otimes g$ of f and g is the two-dimensional function

$$h : \mathbb{R}^2 \to \mathbb{R} : (x,y) \mapsto f(x)g(y).$$

Prooving the equivalence of the two definitions is rather involved. That this makes sense as a tensor product could also be motivated by discretizing f and g on the grids $x_1, ..., x_n$ and $y_1, ..., y_n$ respectively. See Figure 5 for an illustration. Define the vectors

$$F := \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}, \text{ and } G := \begin{pmatrix} g(y_1) \\ \vdots \\ g(y_n) \end{pmatrix}$$

as the discretizations of f and g. Then the discretization of $h := f \otimes g$ is exactly the (finite dimensional) tensor product $F \otimes G$ of the discretizations.

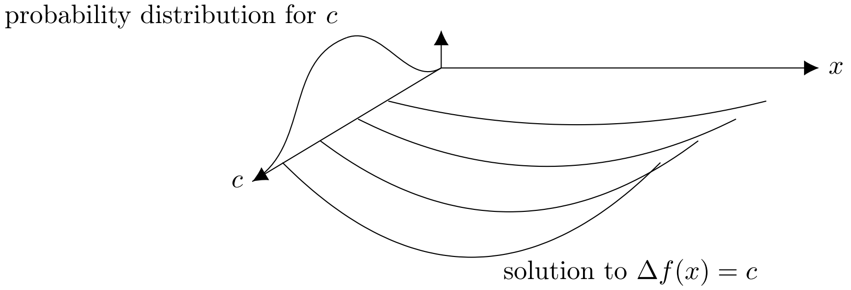
In Example 2.3 of the previous section we have already seen a finite dimensional tensor space of functions. Other prominent tensor function spaces are the space of multivariate Gauss functions or the space of multivariate trigonometric polynomials. The latter is for example used in [4] to show that the solution (function $f : \mathbb{R}^{3N} \to \mathbb{C}$ in $3N$ variables if N particles are involved!) of the electronic Schrödinger equation can be well approximated by a sum of tensor products of 1-dimensional functions. However here we want to give another interesting example, that has recently attracted much attention, because of its importance in constructing fusion reactors. Building a reactor is very expensive (more than one billion Euro for the Wendelstein 7X) and takes a very long time (25 years for the same). Knowing in advance how the plasma is likely to behave is essential. One of the open questions - that if solved would significantly reduce the cost of fusion reactors, according to [5] - is how to avoid turbulence in a plasma.

Example 2.6. (Vlasov Equation) The Vlasov equation models a plasma of charged particles. A plasma differs from a fluid in the way, that the forces acting on the particles are caused by their electric and magnetic fields. Furthermore the Vlasov equation permits non-Boltzmann distributions of speed of the particles. The state space of the plasma is a real valued function $f : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$. The first three dimensions define a position in physical space. The latter three dimensions define a velocity. The value of the function at a certain pair of position and velocity (p, v) defines the amount of particles, that have exactly these parameters.

Before even starting to solve such an equation, we need a way of representing the solution functions in computer memory. The usual approach is to write the solution as a sum of tensor products of one-dimensional functions. If the basis chosen for the one-dimensional space has n elements, then the basis of the tensor product function space has n^d elements if the function to be represented is d-dimensional. Classical approaches optimize over the whole n^d-dimensional function vector space. We will investigate methods that optimize only over a subset of this high-dimensional vector space. Kormann has used tensor decomposition methods in [6] to solve the Vlasov equation.

Example 2.7. (parametric PDEs)

Figure 6: parametric PDE


probability distribution for c
x
c
solution to $\Delta f(x) = c$

Parametric partial differential equations are another application, where high-dimensional functions come into play. Consider for example the Poisson equation

$$\Delta f(x) = c$$

for some parameter c. Now assume that c is not exactly known, but a probability distribution φ is known, that gives the probabilities that c attains a certain value. That means, c is a random variable. Then the question is: For fixed x, what is the probability, that $f(x)$ is inside some interval? Another question might be: What is the average value for $f(x)$? In Figure 6 we have drawn the case, where f is one-dimensional. We have drawn some solutions f_c for different values of c. The solutions to all possible c can be seen as a function F in x and the parameter c. Now for example the average of $f(x)$ would be

$$\int_{-\infty}^{\infty} F(x, c)\varphi(c)\mathrm{d}c.$$

This example is a two-dimensional problem. However if the random variable c is also dependent on the position x, the dimensionality increases. The Karhunen-Loeve expansion also known as infinite dimensional PCA/SVD decomposes c into an infinite sum of independent random variables, each of which is a product of a scalar random variable and a real function of x. Each of the scalar random variables introduces a new variable into the equation, thus increasing the dimensionality of the problem. The Karhunen-Loève theorem allows to approximate by considering only finitely many terms of the decomposition. The number of terms of this decomposition plus the dimensions of x determine the order of the tensor space. See the Phd thesis [7] for details on this topic.

2.4 Frobenius scalar product and norm

To define approximation problems we need to measure distances between tensors. And for many proofs - especially the one of our main result - a notion of orthogonality is essential. Therefore we need to introduce a scalar product.

Definition 2.4. Given two tensors A and B from the same tensor space $\mathbb{R}^{n_1\times\cdots\times n_d}$, we define their scalar product as the standard scalar product on the vector space $\mathbb{R}^{n_1\times\cdots\times n_d}$. This means multiplying element $a_{i_1,\ldots,i_d}$ with element $b_{i_1,\ldots,i_d}$ and summing over all these products

$$\langle A, B\rangle := \sum_{i_1,\ldots,i_d} a_{i_1,\ldots,i_d} b_{i_1,\ldots,i_d}.$$

We will see, why this definition is particularly elegant. It can be written in terms of matricizations or contractions, the topic of the following two chapters.

2.5 Matricization and Tensorization

It is often easier to work with matricizations. Especially because common notation for matrix multiplication can be used. For example we will be able to write

$$\langle A, B\rangle = \operatorname{trace}(A^{(n_1 n_2)\times n_3}\left(B^{(n_1 n_2)\times n_3}\right)^T).$$

The matricization joins several indices into one by ordering the multi-index lexicographically.

Definition 2.5. Let $A \in \mathbb{K}^{n_1\times\cdots\times n_d}$ be a tensor. Then define the *matricization*

$$A^{n_{j_1}\cdot\ldots\cdot n_{j_l}\times n_{j_{l+1}}\cdot\ldots\cdot n_{j_d}}$$

to be a matrix in $\mathbb{K}^{n_{j_1}\cdot\ldots\cdot n_{j_l}\times n_{j_{l+1}}\cdot\ldots\cdot n_{j_d}}$ such that

$$A^{n_{j_1}\cdot\ldots\cdot n_{j_l}\times n_{j_{l+1}}\cdot\ldots\cdot n_{j_d}}(n_{j_l}\cdot\ldots\cdot n_{j_2}\cdot(i_{j_1}-1)+\ldots+i_{j_l}, n_{j_d}\cdot\ldots\cdot n_{j_{l-2}}(i_{j_{l-1}}-1)+\ldots+i_{j_d}) = A(i_1,\ldots,i_d)$$

In the same way we want to define the *tensorization* by

$$\left(A^{n_{j_1}\cdot\ldots\cdot n_{j_l}\times n_{j_{l+1}}\cdot\ldots\cdot n_{j_d}}\right)^{n_1\times\ldots\times n_d} := A$$

This notation is ambiguous for vectorizations of matrices as the superscript can be interpreted as a power. Therefore we recommend using some notation like $A^{(1:\ldots:l\times l+1:\ldots:d)} := A^{n_1\ldots n_l\times n_{l+1}\ldots n_d}$ for any future work. We will give several examples to illustrate the notion of matricization.

Example 2.8. If $A \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ then we can write

$$A^{n_2 n_3 \times n_1}(n_3(i_2 - 1) + i_3, i_1) = A(i_1, i_2, i_3)$$

Example 2.9. Consider a tensor $A \in \mathbb{K}^{n_1 \times n_2 \times n_3}$ with $n_1 = n_2 = n_3 = 3$. This tensor contains the 9 elements

$$a_{111}, a_{112}, a_{113}, a_{121}, ...a_{333}.$$

There are six obvious ways to represent A as a vector by ordering its indices in lexicographic order

$$A^{n_1 n_2 n_3} := \begin{pmatrix} a_{111} \\ a_{112} \\ a_{113} \\ a_{121} \\ \vdots \end{pmatrix}, \; A^{n_1 n_3 n_2} := \begin{pmatrix} a_{111} \\ a_{121} \\ a_{131} \\ a_{112} \\ \vdots \end{pmatrix}, \; A^{n_3 n_2 n_1} := \begin{pmatrix} a_{111} \\ a_{211} \\ a_{311} \\ a_{121} \\ \vdots \end{pmatrix}, \; A^{n_3 n_1 n_2} := \begin{pmatrix} a_{111} \\ a_{121} \\ a_{131} \\ a_{211} \\ \vdots \end{pmatrix}, ...$$

where we use the order of the exponentiated dimensions as an indication of how to sort. We can also rewrite A as a matrix

$$A^{(n_2 n_3) \times n_1} := \begin{pmatrix} a_{111} & a_{211} & a_{311} \\ & & \vdots \\ a_{112} & & \\ a_{113} & & \\ a_{121} & & \\ \vdots & & \end{pmatrix}$$

again with the exponent of A indicating the ordering.

2.6 Contraction

Definition 2.6. The contraction of two tensors $A \in \mathbb{K}^{n_1 \times ... \times n_l \times m}$ and $B \in \mathbb{K}^{m \times n_{l+1} \times ... \times n_d}$ with respect to the $(l+1)$th index of A and the first index of B is defined as the tensor $C \in \mathbb{K}^{n_1 \times ... \times n_d}$ containing the entries

$$C_{i_1,...,i_d} := \sum_{j=1}^{m} A_{i_1,...,j} \cdot B_{j,...,i_d}.$$

We can view the contraction in two ways. The first one needs the notion of fibers. An illustration is shown in Figure 7. A fiber is the vector containing all entries of a tensor when all but one indices are fixed, so for example

$$\begin{pmatrix} A_{i_1,...i_l,1} \\ A_{i_1,...,i_l,2} \\ \vdots \\ A_{i_1,...,i_l,m} \end{pmatrix}.$$

Now the idea is to take all fibers of A with respect to the $(l+1)$th index. These are $n_1 \cdot ... \cdot n_l$ fibers. Also take all $n_{l+1} \cdot ... \cdot n_d$ fibers of B with respect to the first index. Take scalar products of all $n_1 \cdot ... \cdot n_d$ combinations of one fiber from A and one fiber from B and construct the new tensor from the results.

Another way of seeing the contraction is as the sum of m tensor products of hyperslices, depicted in Figure 8. Hyperslices are the dual notion of fibers. They are constructed by fixing only one index.

Figure 7: contraction seen one way

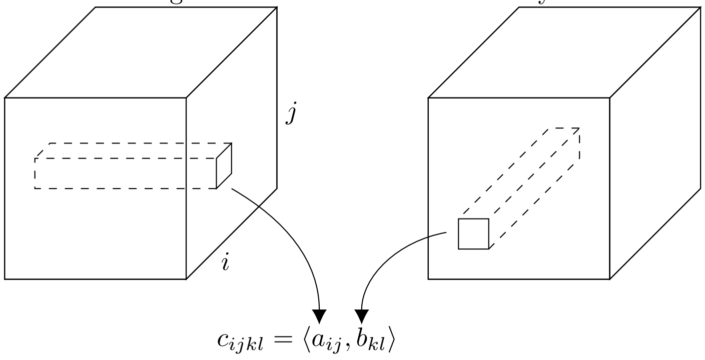
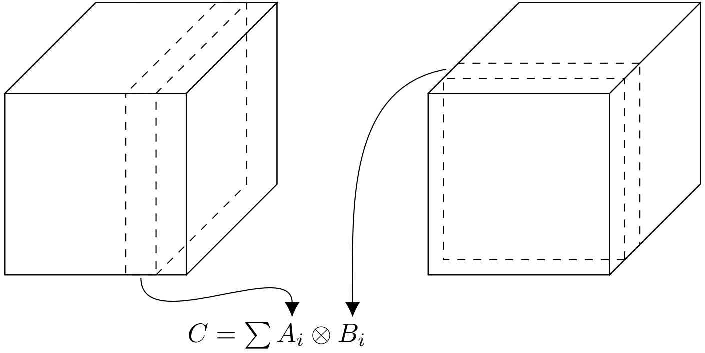


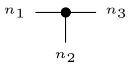
Figure 8: contraction seen a different way



2.7 Tensor diagrams

A very important tool for working with contractions are tensor diagrams. A paper making extensive use of them and including many beautiful diagrams is [8]. In diagrammatic notation every tensor is represented as a dot. Every index (or mode) of the tensor is represented by a line starting at the dot. The positioning of the dot and the direction of the lines can be chosen freely. Thus a tensor of order 3 can look like in Figure 9.

Figure 9: Tensor of order 3



We label the lines with the dimensions of the indices they represent. We can represent a contraction by connecting the lines of the corresponding indices. Figures 7 and 8 would look like in Figure 10 as a tensor diagram.

Figure 11 shows, how we can represent the scalar product by a tensor diagram.

Figure 10: diagram of a contraction

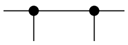
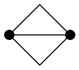
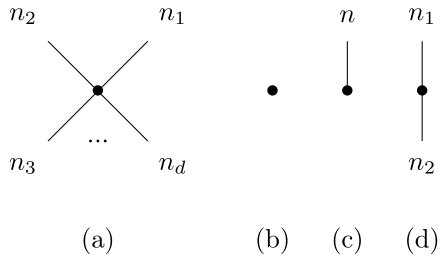


Figure 11: scalar product of two tensors



Further examples are depicted in Figure 12: a tensor of order d as in Definition 2.3 (a); a tensor of order 0 i.e. a scalar (b); of order 1 i.e. a vector (c) and of order 2 i.e. a matrix (d).

Figure 12: tensors in diagram notation



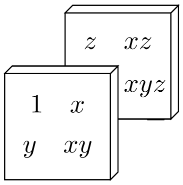
Example 2.10. (evaluation of polynomials)

We want to express the polynomial

$$P : (x, y) \mapsto a + bx + cy + dz + exy + fxz + gyz + hxyz$$

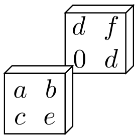
as a tensor. From Example 2.2 we know, that the elements of the monomial basis appear in the tensor

$$\begin{pmatrix}1\\x\end{pmatrix}\otimes\begin{pmatrix}1\\y\end{pmatrix}\otimes\begin{pmatrix}1\\z\end{pmatrix} =$$

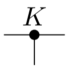


Defining the coefficient tensor for example as

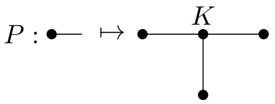
$K =$



$=$



we can write the polynomial as the contraction of three copies of (x, y) with the coefficient tensor K



This is also equivalent to taking the Frobenius scalar product of K and the triple tensor product

$$\begin{pmatrix} 1 \\ x \end{pmatrix} \otimes \begin{pmatrix} 1 \\ y \end{pmatrix} \otimes \begin{pmatrix} 1 \\ z \end{pmatrix}.$$

Remark 2.1. In algebraic statistics the dual graphs are used and also often contractions with more than two factors (being thus able to encode canonical/CP-like decompositions as well). See [9].

3 Introduction to algebraic geometry

Algebraic geometry studies systems of polynomial equations. Interesting for us is the fact that the set of bounded-rank matrices can be defined by polynomial equations. See example 3.4. Also the sets of low-rank tensors defined in Section 3.3.7 are of this kind. As we want to do local optimization on these sets we are interested in the following questions: How can we parametrize the set? Being at some point on the set, how can we find all possible directions that are tangential to the set? And finally, how can we parametrize the set of tangents? About half of this section is an overview over basic algebraic geometry, that is covered in [10]. It is included here to ensure self-containedness. The other half are applications to matrix and tensor varieties.

3.1 Varieties

We have to start with some definitions, which are almost identical copies from [10]. *Monomials* in the variables $x_1, ..., x_d$ are products of the form

$$x_1^{\alpha_1} \cdot ... \cdot x_d^{\alpha_d}$$

with $\alpha_i \in \mathbb{N}_0$ which can also be written in multi-index notation as

$$x^\alpha$$

where $x := (x_1, ..., x_d)$ and $\alpha = (\alpha_1, ..., \alpha_d)$. Monomials are usually viewed as functions

$$(x_1, ..., x_d) \mapsto x_1^{\alpha_1} \cdot ... \cdot x_d^{\alpha_d}.$$

Polynomials are finite linear combinations

$$\sum_\alpha a_\alpha x^\alpha, \quad a_\alpha \in \mathbb{K}$$

of monomials where usually the *coefficients* a_α are elements of some field or ring. An algebraic variety is the set of solutions to a system of polynomial equations.

Definition 3.1. Let $f_1, ..., f_k : \mathbb{K}^n \to \mathbb{K}^m$ be polynomials. Then

$$V(f_1, ..., f_k) := \{x \in \mathbb{K}^n : f_i(x) = 0 \ \forall i\}$$

is the *algebraic variety* defined by $f_1, ..., f_k$.

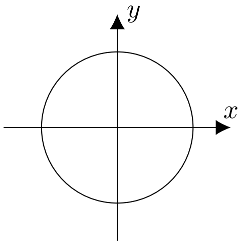
Most of this section can be done over any field. Only when it comes to tangent cones we will have to cite a result that is only known for the field $\mathbb{C}$.

Let us have a look at some examples.

Example 3.1. (unit circle) Let

$$f : \mathbb{R}^2 \to \mathbb{R} : \begin{pmatrix} x \\ y \end{pmatrix} \mapsto x^2 + y^2 - 1.$$

Then $V(f)$ is the unit circle

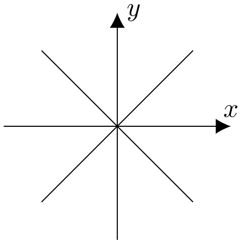


The next example illustrates, why algebraic varieties are not necessarily submanifolds.

Example 3.2. (crossing lines) Let

$$f : \mathbb{R}^2 \to \mathbb{R} : \begin{pmatrix} x \\ y \end{pmatrix} \mapsto (x+y)(x-y).$$

Then $V(f)$ is the union of the two diagonals.

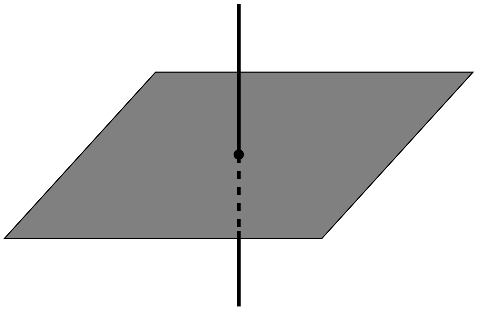


In contrast to manifolds, algebraic varieties are allowed to have *crossings*. All points that have a neighborhood that is a submanifold are called *smooth points*. All other points are called *singular points* and their union the *singular locus*.

Example 3.3. The algebaic variety

$$V(xz, yz) \subset \mathbb{R}^3$$

is the union of the xy-plane and the z-axis:



See [10, p.7] for more examples with three-dimensional pictures of algebraic varieties. The following example is the important one for our work.

Example 3.4. (matrix varieties) The set of $n \times n$-matrices of rank at most $n-1$ is the set of singular matrices. They are precisely those with vanishing determinant

$$\mathcal{M}^{n\times n}_{\le(n-1)} := \left\{ A \in \mathbb{K}^{n\times n} : \det(A) = 0 \right\}.$$

3.2 Ideals

This section is not strictly necessary to understand the main contribution of this thesis. But for the curious reader we want to provide an overview of current algorithmic tools for manipulating algebraic varieties.

Just as varieties, ideals can also be defined by a set of generating polynomials. There are obvious and far less obvious connections to varieties as we will see. For a motivation consider an algebraic variety V defined by the polynomials f and g. This means V is the set of all points where f and g vanish. Now taking any other polynomial h and $x \in V$ we see by

$$(hf)(x) = h(x)f(x) = h(x) \cdot 0 = 0$$

that hf also vanishes on V. Furthermore $f + g$ also vanishes on V. To match this phenomenon the notion of an ideal is defined as:

Definition 3.2. An *ideal* is a subset of polynomials

$$I \subset \mathbb{K}[x_1, ..., x_n]$$

satisfying

$$0 \in I,$$
$$f, g \in I \Rightarrow f + g \in I \text{ and}$$
$$f \in I,\, h \in \mathbb{K}[x_1, ..., x_n] \Rightarrow hf \in I.$$

We can define an ideal as the set of all polynomials that can be obtained from the *generators* f and g by multiplication with another polynomials or by addition.

Definition 3.3. [10] Let $f_1, ..., f_k : \mathbb{K}^n \to \mathbb{K}^m$ be polynomials. Then

$$\langle f_1, ..., f_k \rangle := \left\{ \sum_{i=1}^{k} h_i f_i : \text{ all } h_i \text{ polynomial in } x_1, ..., x_n \right\}$$

is the *ideal generated by* $f_1, ..., f_k$.

Unfortunately the ideal generated by $f_1, ..., f_k$ does not necessarily need to be the set of *all* polynomials vanishing on the variety defined by the same f_i. A counter example is $f : x \mapsto x^2$ in $\mathbb{C}$. The ideal generated by x^2 does not contain x. However x vanishes whenever x^2 vanishes. More on resolving this issue for algebraically closed fields using the notion of *radical ideals* can be found in [10, p.175].

3.3 Tensor varieties

3.3.1 Segre varieties

In this chapter we define the two important building blocks for low-rank tensor varieties of any flavour. The Segre variety is the variety of rank-1 tensors, whereas the Secant variety is the closure of the (Minkowski) sum of several varieties.

We have already encountered the Segre variety in Chapter 2. The setting for defining it is the following: Start with a set of vector spaces $V_1, ..., V_d$.

Definition 3.4. Then the image of the tensor product

$$\otimes : V_1 \times ... \times V_d \to V_1 \otimes ... \otimes V_d$$

is called the *Segre variety*

$$\mathrm{Seg}(V_1, ..., V_d).$$

It is equivalently called the set of *tensors of (canonical) rank at most* 1.

Let us try to visualize the simplest possible Segre variety, the set of rank-1 matrices in $\mathbb{R}^{2\times 2}$.

Example 3.5. $\mathcal{M}^{2\times 2}_{\leq 1} := \mathrm{Seg}(\mathbb{R}^2, \mathbb{R}^2)$ is an algebraic variety. However, removing the origin it becomes a smooth manifold. Furthermore it is projective in the sense that if $A \in \mathrm{Seg}(\mathbb{R}^2, \mathbb{R}^2)$ so is αA for every $\alpha \in \mathbb{R}$. We know thus that $\mathrm{Seg}(\mathbb{R}^2, \mathbb{R}^2)\backslash\{0\}$ is topologically some manifold $\mathcal{M}$ times an open interval $(0, \infty)$. The manifold $\mathcal{M}$ can be defined as $\mathrm{Seg}(\mathbb{R}^2, \mathbb{R}^2) \cap \mathbb{S}^3$, i.e. the set of all rank-1 matrices

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

of unit norm

$$a^2 + b^2 + c^2 + d^2 = 1.$$

Now, we need to visualize this subset of $\mathbb{S}^3 := \{(a, b, c, d) \in \mathbb{R}^4 : a^2 + b^2 + c^2 + d^2 = 1\}$. Just as from the sphere $\mathbb{S}^2$ we can remove one point and flatten the rest to a disk, we can remove one point of $\mathbb{S}^3$ and flatten the rest to a solid sphere $\mathbb{D}^3$. We can think of the boundary of this solid sphere as the point we have removed. So the whole boundary is identified to one point, let us say to

$$\begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}.$$

Then $\mathrm{Seg}(\mathbb{R}^2, \mathbb{R}^2) \cap \mathbb{S}^3$ would contain the matrices - and convex combinations between any two of them - depicted in Figure 13. The lines connecting matrices correspond to linear combinations of the matrices at the endpoints, which are also of rank 1.

The matrices $\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ and $\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$ are connected by a line in the diagram. This line is the set of all matrices of the form

$$\frac{1}{a^2 + b^2} \begin{pmatrix} a & b \\ 0 & 0 \end{pmatrix}$$

with a and b from $[0, 1]$, i.e. the convex combinations between the two matrices.

There is no line between $\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ and $\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$ because the convex combinations would have rank 2 and are thus not contained in $\mathcal{M}^{2\times 2}_{\leq 1}$.

In Figure 13 you can see, that the dashed loop and the fat loop are linked. In Figure 14 we have drawn the torus untwisted to better see the surface and see that it is indeed topologically a torus. Having seen all this, we can write it as

$$\mathcal{M}^{2\times 2}_{\leq 1} \cap \mathbb{S}^3 \cong \mathbb{T}^2$$

Figure 13: Topology of rank-1 matrices ...

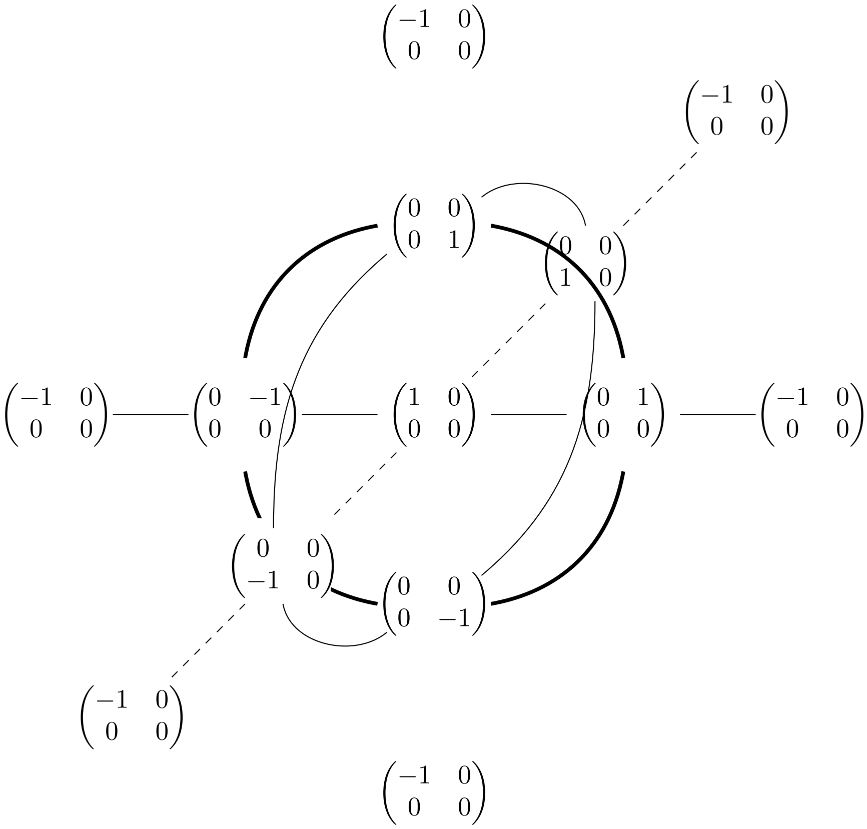
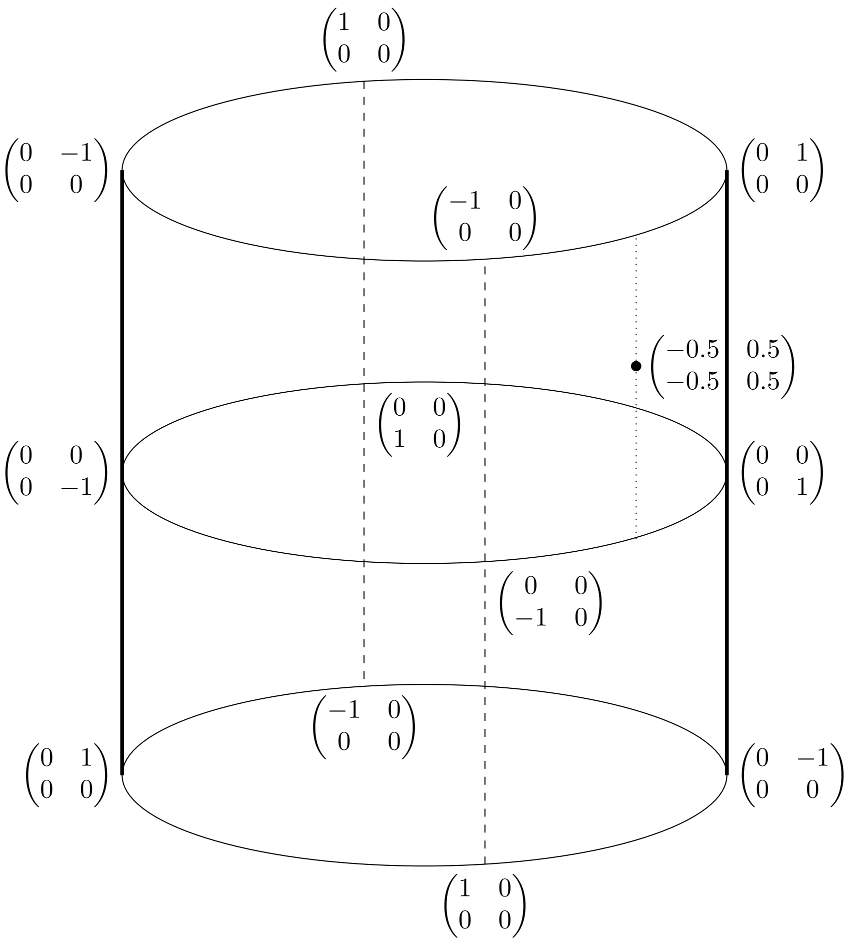

$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & -1\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ -1 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ 0 & -1\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$

Figure 14: ... is a torus


$\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & -1\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}-0.5 & 0.5\\ -0.5 & 0.5\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ 0 & -1\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}$
$\begin{pmatrix}0 & 0\\ -1 & 0\end{pmatrix}$
$\begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}0 & -1\\ 0 & 0\end{pmatrix}$
$\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}$

where $\cong$ shall mean homeomorphic (but not ambient isotopic because of the linked circles). To obtain the projective variety we identify matrix A with $-A$, thus identifying left and right half of the torus in Figure 14 by a point reflection along the vertical central axis. It remains that

$$\mathbb{P}\left(\mathcal{M}_{\leq 1}^{2\times 2}\right) \cong \mathbb{T}^2.$$

Analogously we see that in general the topology of Segre varieties can be characterized by

$$\mathbb{P}\left(\mathcal{M}_{\leq 1}^{n_1\times\ldots\times n_d}\right) \cong \mathcal{M}_{\leq 1}^{n_1\times\ldots\times n_d} \cap \mathbb{S}^{n_1\cdot\ldots\cdot n_d} \cong \mathbb{S}^{n_1-1}\times \ldots \times \mathbb{S}^{n_d-1}.$$

This result can be found in [11]. Note the compatibility of the fibre product and the product topology.

3.3.2 Secant varieties

Any tensor of rank 2 or less is by definition of the rank the sum of at most two tensors of rank 1. The Segre veriety generalizes this construction allowing not only rank-1 tensors to be added but elements from arbitrary varieties. Even though in the current work we are only interested in the Segre case, we want the reader to be aware of the general definition.

Definition 3.5. (Zariski closure and secant variety) Given any subset M of $\mathbb{K}^n$, the *Zariski closure* $\overline{M}$ is defined to be the smallest algebraic variety containing the whole set M.

Let X_1, ..., X_r be r algebraic (sub-)varieties of $\mathbb{K}^s$. Then the Zariski closure of the sum of the X_i

$$\mathrm{Sec}\left(X_1, ..., X_r\right) := \overline{\left\{ z \in \mathbb{K}^s : \exists x_i \in X_i, \alpha_i \in \mathbb{K} : z = \sum_{i=1}^{r} \alpha_i x_i \right\}}$$

is called the Secant variety of $(X_1, ..., X_r)$. Define the shorthand form

$$\mathrm{Sec}_k(X) := \mathrm{Sec}\,\underbrace{(X, ..., X)}_{k \text{ times}}.$$

Remark 3.1. We have to take the Zariski closure as the sum alone might not be an algebraic variety. See the following example.

Example 3.6. Consider $V(y^3 - x^2)$. The union of its secants is $\mathbb{R}^2 \backslash \left\{ \begin{pmatrix} 0 \\ y \end{pmatrix} : y < 0 \right\}$. The Zariski closure is the whole plane $\mathbb{R}^2$.

Now we have the necessary tools to define all tensor subvarieties that are of interest to this work.

3.3.3 Rank one tensors

Tensors of rank 1 play a special role and have the amazing property of being of rank 1 in no matter which of our tensor formats - CP, TT, HT, Tucker (CP and TT will be defined on following page). Furthermore the set of all rank 1 tensors is a variety which cannot be said about CP tensors of higher ranks.

Definition 3.6. (rank 1 tensors) The set of tensors of rank at most 1 (rank 1 tensors and the zero tensor) is

$$\mathcal{M}_{\leq 1}^{n_1\times\ldots\times n_d} := \mathrm{Seg}\left(\mathbb{R}^{n_1}, ..., \mathbb{R}^{n_d}\right) := \mathrm{img}\left(\otimes\right) \subset \mathbb{R}^{n_1\times\ldots\times n_d}.$$

3.3.4 Matrices of bounded rank

The set of matrices of rank bounded by r

$$\mathcal{M}_{\leq r}^{n\times m} := \operatorname{Sec}\underbrace{\left(\operatorname{Seg}\left(\mathbb{R}^n, \mathbb{R}^m\right), ..., \operatorname{Seg}\left(\mathbb{R}^n, \mathbb{R}^m\right)\right)}_{r \text{ times}}$$

is an algebraic variety because it is defined by the determinants of all $(r+1)\times(r+1)$ submatrices.

3.3.5 Canonically decomposed (CP) tensors

The frequently used abbreviation CP now commonly stands for canonical polyadic [12]. It denotes the decomposition in which a tensor is written as a sum of elementary (rank 1) tensors. The minimum number of summands is called the rank r of the tensor. A formal definition is:

Definition 3.7. (CP tensors) The set of tensors of canonical (CP) rank bounded by r is defined as

$$\mathcal{M}_{\leq r}^{n_1\times...\times n_d} := \left\{A \in \mathbb{R}^{n_1\times...\times n_d} : \exists\ A_1, ..., A_r \in \mathbb{R}^{n_1\times...\times n_d} : \operatorname{rank}(A_i) \leq 1 \text{ and } A = A_1 + ... + A_r\right\}.$$

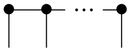
Remark 3.2. The secant variety

$$\operatorname{Sec}\underbrace{\left(\operatorname{Seg}\left(\mathbb{R}^{n_1}, ..., \mathbb{R}^{n_d}\right), ..., \operatorname{Seg}\left(\mathbb{R}^{n_1}, ..., \mathbb{R}^{n_d}\right)\right)}_{r \text{ times}}$$

includes in general also some tensors of rank greater than r because we have taken the Zariski closure in the definition of secant varieties. More on this problem can be found in [11, p. 37, 118]. Optimization on this set is investigated in [13].

3.3.6 Tensor trains

Recall the diagrammatic notation from the introduction to tensors in chapter 2.7. TT tensors can thus be written as:



Definition 3.8. The variety of TT tensors of rank bounded by $\mathbf{k} := (k_1, ..., k_{d-1})$ can be defined as

$$\mathcal{M}_{\leq(k_1,...,k_{d-1})}^{n_1\times...\times n_d} := \bigcap_{i=1}^{d-1} \operatorname{Sec}_{k_i}\left(\operatorname{Seg}\left(\mathbb{R}^{n_1\times...\times n_i}, \mathbb{R}^{n_{i+1}\times...\times n_d}\right)\right).$$

Matricizing the elements of

$$\operatorname{Sec}_{k_i}\left(\operatorname{Seg}\left(\mathbb{R}^{n_1\times...\times n_i}, \mathbb{R}^{n_{i+1}\times...\times n_d}\right)\right)$$

produces the variety of matrices of bounded rank

$$\operatorname{Sec}_{k_i}\left(\operatorname{Seg}\left(\mathbb{R}^{n_1\cdot...\cdot n_i}, \mathbb{R}^{n_{i+1}\cdot...\cdot n_d}\right)\right)$$

which is a subset of

$$\mathbb{R}^{n_1\cdot...\cdot n_i} \otimes \mathbb{R}^{n_{i+1}\cdot...\cdot n_d}.$$

3.3.7 Hierarchical tensors

The variety of hierarchical tensors can be defined analogously to the variety of tensor trains. The notation however neccessitates the use of trees.

Definition 3.9. Let $v_1, ..., v_d$ be the leaves and $w_1, ..., w_D$ be the inner nodes of a tree T. Edges are (unordered) pairs of nodes. Let $e_1, ..., e_{D-1}$ be the edges not connected to a leaf. Removing an edge e_i cuts the tree in two parts, one with leaves $v_{(j_i)_1}, ..., v_{(j_i)_l}$ and the other with leaves $v_{(j_i)_{l+1}}, ..., v_{(j_i)_d}$.

The variety of HT tensors of rank bounded by $\mathbf{k} := (k_1, ..., k_{D-1})$ can then be defined as

$$\mathcal{M}(T)^{n_1 \times ... \times n_d}_{\leq (k_1, ..., k_{D-1})} := \bigcap_{i=1}^{D-1} \mathrm{Sec}_{k_i} \left(\mathrm{Seg} \left(\mathbb{R}^{n_{(j_i)_1} \times ... \times n_{(j_i)_l}}, \mathbb{R}^{n_{(j_i)_{l+1}} \times ... \times n_{(j_i)_d}} \right) \right).$$

See [14] for an alternative definition of an object, that is slightly different, but would work for our purposes equally well.

3.4 Gröbner bases for the parametrization of tangent cones

In subsequent chapters we will see the derivation of a parametrization of the tangent cones to tensor train varieties. This derivation will work entirely without the Gröbner basis machinery. However when we were still lacking a proof and being unsure about the structure of the parametrization, computing small examples gave hope and the right direction. Therefore we want to include in this thesis also a reference to the existing (symbolic) computational methods of algebraic geometry. These methods are able to determine the tangent cone of a given variety at a given point. They are also capable of verifying a guessed parametrization. The only drawback of these methods we know of is their enormous computational complexity making it only feasible to investigate the very smallest examples of TT varieties.

The methods described in this chapter are almost entirely discussed in [10]. Our contribution amounts to applying them to TT varieties.

3.4.1 Lexicographic ordering

For all that follows we need to define an ordering on the monomials. As is important later for implicitizing a parametrization, a good choice for our purposes will be a lexicographic ordering induced by an ordering of the variables. For example

$$x_1 > x_2 > x_3$$

would imply

$$x_1 x_3 > x_1 x_2$$

and

$$x_1 x_3 > x_2 x_1 x_3$$

just like the ordering of words in an English dictionary.

Definition 3.10. [10] Given two multi indices $\alpha = (\alpha_1, ..., \alpha_n)$ and $\beta = (\beta_1, ..., \beta_n)$, we say $x^\alpha > x^\beta$ in lexicographic order if the leftmost nonzero entry of $\alpha - \beta$ is positive.

Having defined an ordering, we can talk of the *leading term* of a polynomial as the one monomial that is greatest with respect to the monomial ordering. When writing down polynomials, we usually order the monomials accordingly.

3.4.2 Division algorithm

The calculation of Gröbner bases is based on the division algorithm for multivariate polynomials. Given a polynomial f and s polynomials $f_1, ..., f_s$ to be divided by, we want to write f as

$$f = a_1 f_1 + ... + a_s f_s + r$$

such that all monomials in r are not divisible by any of the leading terms of the f_i.

The division algorithm works as follows. Take the leading monomial of f. If there is a leading monomial of any of the f_i dividing it, then do a division step. If not, then add it to the remainder. See [10] for details.

Example 3.7. Using the ordering $x_1 > x_2$ we want to divide $f = x_1 + x_2^2$ by $f_1 = x_1^2$ and $f_2 = x_2^2$.

The first term of f is not divisible by any of the leading terms of the f_i so it goes into the remainder. The second term of f is divisible by f_2 and we end up with

$$f = 1 \cdot f_2 + x_1.$$

3.4.3 Buchberger's algorithm

Buchberger's algorithm is for polynomial systems of equations what Gauß elimination is for linear systems. The input is a finite set of polynomials $f_1, ..., f_s$. The output (a Gröbner basis) is another set of polynomials with the same set of solutions and the property that it can (in principle) be solved using backward substitution. Furtheremore the Gröbner basis can be made unique (reduced basis) [10, p. 92] to be able to compare the solution sets of two polynomial systems.

In contrast to linear systems the number of polynomials in the Gröbner basis returned by Buchberger's algorithm is in general rather large. Here we want to copy the definition of Gröbner basis from [10].

Definition 3.11. (Gröbner basis) A finite subset $G = \{g_1, ..., g_t\}$ of an ideal I is said to be a Gröbner basis if

$$\langle \mathrm{LT}(g_1), ..., \mathrm{LT}(g_t) \rangle = \langle \mathrm{LT}(I) \rangle$$

where $\mathrm{LT}(g_i)$ denotes the leading term of g_i and $\mathrm{LT}(I)$ is the ideal generated by the leading terms of *all* polynomials in I.

For the calculation of Gröbner bases we need the so called S-polynomials. Given two polynomials $p_1 = \sum \alpha_i x^{\alpha_i}$ and $p_2 = \sum \beta_i x^{\beta_i}$ (in multi-index notation), the S-polynomial of p_1 and p_2 is a sum

$$h_1 p_1 + h_2 p_2$$

such that the leading terms cancel and are both the least common multiple of $\mathrm{LT}(p_1)$ and $\mathrm{LT}(p_2)$. See the example.

Example 3.8. We use lexicographic order.

$$p_1 := x_1x_2^2 + x_1x_3$$
$$p_2 := x_1^2x_2x_3 + x_5$$

Then we have

$$S(p_1, p_2) = x_1x_3p_1 - x_2p_2 = x_1^2x_2^2x_3 - x_1^2x_2^2x_3 + x_1^2x_3^2 - x_2x_5 = x_1^2x_3^2 - x_2x_5.$$

Though conceptionally seamingly straight forward, Buchberger's algorithm was not described until 1985 in [15] . The starting point is a set of polynomials $S = \{f_1, ..., f_s\}$. They define an ideal (and thus also an algebraic variety). The goal is to find a Gröbner basis defining the same ideal. Buchberger's algorithms works as follows:

Theorem 3.1. *(Buchberger's algorithm) Calculate the S-polynomials of all pairs (f_i, f_j) for $f_i, f_j \in S$ and divide each of them by all $f_i \in S$ (The result will be polynomials that are also part of the ideal, though are hopefully smaller in terms of the monomial order). Add any non-vanishing remainder to S. Do this process while there are still non-vanishing remainders. The result will be a Gröbner basis of $\langle f_1, ..., f_s\rangle$.*

Example 3.9. (Gröbner basis of $\mathcal{M}_{\leq 1}^{3\times 3}$) The set of 3 × 3-matrices of rank at most 1 is defined by the 2 × 2-minors of the matrix. For

$$\mathbb{R}^{3\times 3} \ni A = \begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix}$$

the defining polynomials are

$$f_1 = x_1y_2 - x_2y_1, \quad f_2 = x_1y_3 - x_3y_1, \quad f_3 = x_2y_3 - x_3y_2, ...$$

They already form a Gröbner basis, because all S-polynomials have remainder 0 after division by all f_i. See exemplarily that

$$S(f_1, f_2) = f_1y_3 - f_2y_2 = -x_2y_1y_3 + x_3y_1y_2 = -y_1(x_2y_3 - x_3y_2) = -y_1f_3.$$

There are more advanced algorithms (notably F4 and F5 [16]) for calculating Gröbner bases that enable the solution of problems that would be intractible with Buchberger's algorithms. Faugère's algorithms are famous for breaking the HFE challenge 1 and the *cyclic 10* problem.

3.4.4 Comparing varieties

Given two sets of generators of the same variety, their Gröbner bases are in general different. This can have two causes:

1. The sets of generators must not define the same ideal. For example $\{x\}$ and $\{x^2\}$ definde the same variety but two different ideals. They are both Gröbner bases. The solution to this problem is an algorithm from [17] (implemented in Macaulay 2) that computes a Gröbner basis of the *radical* (in easy words the simplest ideal defining the same variety) of an ideal.

2. Gröbner bases are not unique. However so called *reduced* Gröbner bases are. Since [10, p. 92, Prop. 6] contains a constructive proof for the uniqueness, this problem is also solved.

3.4.5 Parametrization

The essential idea of tensor decomposition methods is to parametrize a low-dimensional subvariety / subset of the tensor space. Thus we are naturally interested in parametrizing whatever subset we get our hands on. In this chapter we will recall the algebraic geometric notion of a parametrization and the corresponding algorithmic possibilities.

Example 3.10. The easiest example of a low-rank decomposition is the set

$$V := \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} : ad - bc = 0 \right\}$$

of 2×2-matrices of rank at most 1. The function

$$f : (x, y, u, v) \mapsto \begin{pmatrix} xu & yu \\ xv & yv \end{pmatrix}$$

defines a parametrization of V. Note that f is not injective, as

$$f\left(\alpha x, \alpha y, \frac{1}{\alpha} u, \frac{1}{\alpha} v\right) = f\left(x, y, u, v\right).$$

The following will be a running example.

Example 3.11. Consider the variety

$$V := \left\{ \begin{pmatrix} x \\ y \end{pmatrix} : x^2 = y^3 \right\}.$$

See Figure 16 for a picture. It is the image of the function

$$f : t \mapsto \begin{pmatrix} t^3 \\ t^2 \end{pmatrix}.$$

We call f a parametrization of V.

In algebraic geometry the definition of a parametrization is taylored to suit a theorem that states how to obtain an implicit description of a variety given a parametrization. All of what follows is also subject of and heavily influenced by [10, Chapter 3].

Definition 3.12. A rational function

$$f : (t_1, ..., t_r) \mapsto \begin{pmatrix} f_1(t_1, ..., t_r) \\ \vdots \\ f_n(t_1, ..., t_r) \end{pmatrix}$$

is called a *parametrization* of a variety

$$V = \{(x_1, ..., x_n) : g_i(x_1, ..., x_n) = 0 \ \forall i = 1, ..., m\}$$

if the image of f is contained in V but is not contained in any proper subvariety of V. Rational means, that the f_i are quotients of polynomials.

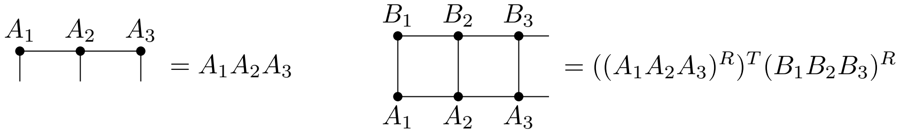
Remark 3.3. For parametrizing TT, HT and Tucker varieties polynomial parametrizations suffice and for everything in this work we do not need rational but only polynomial parametrizations.

3.4.6 Parametrization of TT varieties

As in Definition 2.5, we write $A^{(n_1 \dots n_i) \times (n_{i+1} \dots n_d)}$ for the matricization (i.e. combining several indices into one using lexicographic order) of $A \in \mathbb{R}^{n_1 \times \dots \times n_d}$ and $A^{n_1 \times \dots \times n_d}$ for the tensorization. Define the shorthand $A^L := A^{n_1 \times (n_2 \dots n_d)}$ and $A^R := A^{(n_1 \dots n_{d-1}) \times n_d}$.

In subsequent chapters we will use the *TT product* defined below. In the matrix case it is equivalent to the matrix product and it allows us with little effort to rigorously describe many tensor diagrams. Figure 15 shall serve as a dictionary between tensor diagrams and the TT product notation.

Figure 15: tensor diagrams



Definition 3.13. We define a scalar product on $\mathbb{R}^{n_1 \times \dots \times n_d}$ as the standard scalar product on $\mathbb{R}^{n_1 \dots n_d}$. This induces a norm and the notion of orthogonality. We denote the *TT product* of the two tensors $A \in \mathbb{R}^{n_1 \times \dots \times n_i \times k}$ and $B \in \mathbb{R}^{k \times n_{i+1} \times \dots \times n_d}$ by

$$AB := \left(A^R B^L\right)^{n_1 \times \dots \times n_d} \in \mathbb{R}^{n_1 \times \dots \times n_d}.$$

The entries of this tensor are

$$(AB)(j_1, ..., j_d) := \sum_{m=1}^{k} A(j_1, ..., j_i, m) B(m, j_{i+1}, ..., j_d).$$

The TT product is associative. It is equivalent to the matrix product if A and B are matrices. The definition of the TT variety from Section 3.3.6 can be rewritten as:

Definition 3.14. The variety of TT tensors [18] of order d and dimensions $(n_1, ..., n_d)$ of rank bounded by $\mathbf{k} = (k_1, ..., k_{d-1})$ can also be defined as

$$\mathcal{M}^{n_1 \times \dots \times n_d}_{\leq (k_1, \dots, k_{d-1})} := \{A \in \mathbb{R}^{n_1 \times \dots \times n_d} : \forall i : \operatorname{rank}\left(A^{(n_1 \dots n_i) \times (n_{i+1} \dots n_d)}\right) \leq k_i\}.$$

The equivalence to the original Definition 3.8 follows by matricization. Define the manifold of TT tensors of order d and dimensions $(n_1, ..., n_d)$ of rank exactly $(k_1, ..., k_{d-1})$ as

$$\mathcal{M}^{n_1 \times \dots \times n_d}_{= (k_1, \dots, k_{d-1})} := \{A \in \mathbb{R}^{n_1 \times \dots \times n_d} : \forall i : \operatorname{rank}\left(A^{(n_1 \dots n_i) \times (n_{i+1} \dots n_d)}\right) = k_i\}.$$

A proof for the TT manifold being a smooth quotient manifold can be found in [19].

Any tensor $\mathcal{X}$ from $\mathcal{M}^{n_1 \times \dots \times n_d}_{\leq (k_1, \dots, k_{d-1})}$ can be decomposed as a product $\mathcal{X} = A_1 ... A_d$ with $A_1 \in \mathbb{R}^{n_1 \times k_1}$, $A_i \in \mathbb{R}^{k_{i-1} \times n_i \times k_i}$ $\forall i = 2, ..., d-1$ and $A_d \in \mathbb{R}^{k_{d-1} \times n_d}$. This defines a parametrization $(A_1, ..., A_d) \mapsto A_1 \cdot ... \cdot A_d$ of the TT variety. We will need this fact in Lemma 4.4. The proof is not difficult and can be found for example in [18, Thm. 2.1].

Parametrizing algebraic varieties in general however is difficult and not all varieties admit a parametrization!

For the inverse of parametrizing there is an algorithm, the subject of the following chapter.

3.4.7 Implicitization

Given a parametrization, the process of finding the corresponding variety is called *implicitization*. Let us look at the previous example again.

Example 3.12. (continued) Suppose we have a parametrization

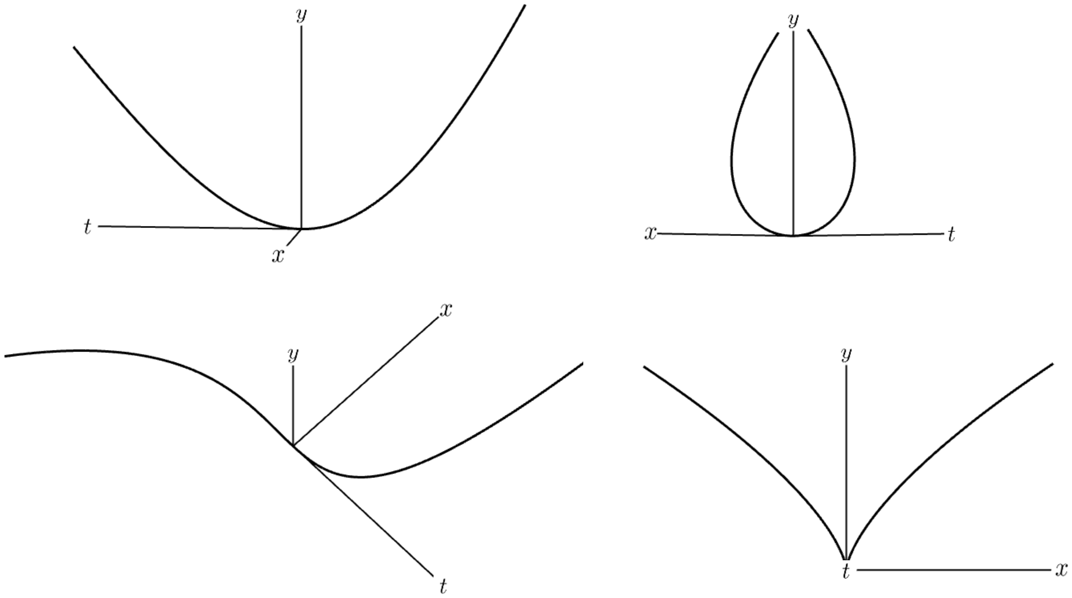
$$f : t \mapsto \begin{pmatrix} t^3 \\ t^2 \end{pmatrix}.$$

This defines a variety in $\mathbb{K}^3$, namely

$$W := \left\{ \begin{pmatrix} t \\ x \\ y \end{pmatrix} : x = t^3, y = t^2 \right\}.$$

See Figure 16.

Figure 16: W seen from different points of view



The projection V of W onto the x, y-plane is the variety that is parametrized by f. The result of this projection is called first elimination ideal because it eliminates the first variable, here t.

Algorithm 1 [10, Ch. 3.3] Input: parametrization of f
Output: implicit description of variety V whose parametrization is f

Calculate a Gröbner basis of $V(x_1 - f_1(t_1, ..., t_r), x_2 - f_2(t_1, ..., t_r), ...)$ with respect to a lexicographic (not graded) order where $t_i > x_j \; \forall i, j$. Prune all polynomials of the Gröbner basis that contain t_i's.

Example 3.13. (continued) In our example we would do the following Gröbner basis calculation

$$g_1 = t^3 - x$$

$$g_2 = t^2 - y$$

$$g_3 = S(g_1, g_2) = t^3 - x - (t^3 - ty) = ty - x$$

$$g_4 = S(g_2, g_3) = t^2y - y^2 - (t^2y - tx) = tx - y^2$$

$$g_5 = S(g_3, g_4) = txy - y^3 - (txy - x^2) = x^2 - y^3$$

We see that $(g_1 = 0 \wedge g_2 = 0)$ implies $g_5 = 0$. So the projection of $V(g_1, g_2)$ onto the x, y-plane is contained in $V(g_5)$. That $V(g_5)$ is indeed the smallest variety containing the projection is ensured by Theorem 1 in [10, p. 130, Chapter 3.3].

3.4.8 Algebraic tangent cones

The notion of the tangent cone is quite intuitive. It is the set of all tangent directions, i.e. the directions in which one can start a curve that lies on the variety. This definition can be formalized as:

Definition 3.15. (e.g. used in [20]) The tangent cone of $\mathcal{M}$ at A is the set of the first non-zero derivatives of all analytic arcs in $\mathcal{M}$ going through A

$$\{v \in \mathbb{R}^N : \exists n \in \mathbb{N}, \gamma : [0,1] \to \mathcal{M} \text{ real analytic} :$$

$$\gamma(0) = A,\ \gamma^{(n)}(0) = v \text{ and } \forall i < n : \gamma^{(i)}(0) = 0\}.$$

Example 3.14. (also published in [21]) The first derivatives do not suffice as our running example can show. Consider the variety

$$\mathcal{M} := \left\{(x, y) \in \mathbb{R}^2 : x^2 = y^3\right\}$$

and an analytic arc γ with values in $\mathcal{M}$ such that $\gamma(0) = (0, 0)$. Then $\dot{\gamma}(0)$ always vanishes. Verify this by plugging the analytic arc $\gamma : t \mapsto (a_1t + a_2t^2 + ..., b_1t + b_2t^2 + ...)$ into the defining equation and compare coefficients. But the tangent cone of $\mathcal{M}$ at $(0, 0)$ is $\{(0, a), a \geq 0\}$ which is more than $\{(0, 0)\}$. This example also works in the complex case. For example $\gamma : t \mapsto (t^{\frac{3}{2}}, t)$ has the desired tangent vector but is not analytic.

For real algebraic varieties (and thus also for complex ones via $\mathbb{C} \cong \mathbb{R}^2$, analytic in real and imaginary part) the previous definition is equivalent to the following via [22].

Definition 3.16. (e.g. used in [1]) The tangent cone of an algebraic variety $\mathcal{M} \in \mathbb{R}^N$ at a point $A \subset \mathcal{M}$ is the set of all vectors that are limits of secants through A:

$$T_A\mathcal{M} := \{\xi \in \mathbb{R}^N : \exists (x_n) \subset \mathcal{M}, (a_n) \subset \mathbb{R}^+ \text{ s.t. } x_n \to A,\ a_n(x_n - A) \to \xi\}.$$

For complex varieties the first two definitions are also equivalent to the algebraic tangent cone [10, p. 500, Chapter 9.7, Theorem 6]. For defining the latter, we need to say what we mean by a smallest homogeneous component. Given a polynomial f its monomials can be divided into sets of monomials of equal total degree. For example

$$f = x^2 + xy + x^2y + xy^2 + x^2y^2$$

has x^2 and xy of total degree 2, x^2y and xy^2 of total degree 3 and x^2y^2 of total degree 4. Take the set with the monomials of smallest total degree and sum them up. The result is called *smallest homogeneous component* f_{min} of f.

We can translate a variety to bring any point p to the origin by redefining its defining polynomials as

$$f_p(x) := f(x + p).$$

Definition 3.17. The tangent cone of a variety V at the origin is defined by the minimal homogeneous components of all the elements in the ideal of V

$$V(f_{p_{min}}, f \in I(V)).$$

Remark 3.4. Determining the tangent cone algorithmically is hindered mainly by the fact, that minimal homogeneous components of all elements of the ideal, not only of some defining polynomials are required. However using Gröbner bases it is possible to compute the algebraic tangent cone as explained in Proposition 4 of [10, p. 497, Chapter 9.7] and implemented in Macaulay 2.

3.4.9 Experimental verification of parametrization candidate for the tensor train tangent cone

We now have all the machinery to do the following:

Given some fixed TT format and a fixed point in this format, determine whether a guess for the parametrization of the tangent cone at this point is correct.

Example 3.15. Let $\mathcal{M}^{3\times3\times3}_{\leq(2,2)} \subset \mathbb{C}^{3\times3\times3}$ be the variety of $3 \times 3 \times 3$-tensors of TT rank at most $(2, 2)$.

$$p = \begin{pmatrix}1\\0\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\\0\end{pmatrix}$$

lies on this variety. In fact by linear transformations in each dimension separately we can transform any rank 1 point to p.

Defining $A_1 = \begin{pmatrix}1 & 0 & 0\end{pmatrix} \in \mathbb{R}^{3\times1\times1}$, $A_2 = \begin{pmatrix}1 & 0 & 0\end{pmatrix} \in \mathbb{R}^{1\times3\times1}$ and $A_3 = \begin{pmatrix}1 & 0 & 0\end{pmatrix} \in \mathbb{R}^{1\times1\times3}$ we can write $p = A_1A_2A_3$ in TT product notation. Our early guess (that turned out to be true in general) for the tangent cone was

$$TC_p\mathcal{M}^{3\times3\times3}_{\leq(2,2)} = \left\{ \begin{pmatrix}A_1 & U_1 & X_1\end{pmatrix} \begin{pmatrix}A_2 & U_2 & X_2\\ 0 & Z_2 & V_2\\ 0 & 0 & A_2\end{pmatrix} \begin{pmatrix}X_3\\ V_3\\ A_3\end{pmatrix} \right\}$$

with dimensions of U_i, V_i, Z_i equal to $3-2=1$. Here block matrix notation is used with respect to the first and last indices of the tensors. Writing this explicitly yields tangent vectors of the form

$$\begin{pmatrix} 1 & t_0 & t_3 \\ 0 & t_1 & t_4 \\ 0 & t_2 & t_5 \end{pmatrix} C \begin{pmatrix} t_{18} & t_{19} & t_{20} \\ t_{21} & t_{22} & t_{23} \\ 1 & 0 & 0 \end{pmatrix}$$

where the three coronal slices (slices with respect to the second index) of C are

$$\begin{pmatrix} 1 & t_6 & t_9 \\ 0 & t_{12} & t_{15} \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & t_7 & t_{10} \\ 0 & t_{13} & t_{16} \\ 0 & 0 & 0 \end{pmatrix} \text{ and } \begin{pmatrix} 0 & t_8 & t_{11} \\ 0 & t_{14} & t_{17} \\ 0 & 0 & 0 \end{pmatrix}.$$

Now using the algorithm for determining the algebriac tangent cone, we can calculate a Gröbner basis of $TC_p\mathcal{M}$.

Using the algorithm for implicitization, we can calculate a Gröbner basis of the variety that is parametrized by our guessed parametrization. Both can be compared if in reduced form. A Macaulay 2 [23] program implementing all this is:

```
R=QQ[t_(0) .. t_(23), x_(0,0,0) .. x_(2,2,2), MonomialOrder=>Eliminate 24];
A=genericMatrix(R,x_(0,0,0),3,9);
B=matrix{{1+x_(0,0,0)-x_(0,0,0), 0, 0,0,0,0,0,0,0},{0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0}};
C=A+B;
I1=minors(3,C);
D=genericMatrix(R,x_(0,0,0),9,3);
E=matrix{{1+x_(0,0,0)-x_(0,0,0), 0, 0},{0,0,0},{0,0,0},{0,0,0},{0,0,0},
{0,0,0},{0,0,0},{0,0,0},{0,0,0}};
F=D+E;
I2=minors(3,F);
I3=I1+I2;
I4 = tangentCone(I3);

G=matrix{{1,t_(0),t_(3)},{0,t_(1),t_(4)},{0,t_(2),t_(5)}};
K=matrix{{t_(18), t_(19), t_(20)},{t_(21),t_(21),t_(23)},{1,0,0}};
H=matrix{{1,t_(6),t_(9)},{0,t_(12),t_(15)},{0,0,1}};
I=matrix{{0,t_(7),t_(10)},{0,t_(13),t_(16)},{0,0,0}};
J=matrix{{0,t_(8),t_(11)},{0,t_(14),t_(17)},{0,0,0}};
L=G*H*K;
M=G*I*K;
N=G*J*K;
P=L||M||N;
Q=P-D;
I5 = ideal(Q);{*parametrized set*}

I6=ideal(selectInSubring(1,gens gb I5));{*Implicitization*}
```

```
{*We’re considering the parametrization

(A1,U1,X1) (A2,U2,X2) (X3)
           (0 ,Z2,V2) (V3)
           (0 ,0 ,A2) (A3)
i
n a TT-format of rank (2,2) in the space of 3x3x3-tensors.
The rank-deficient tensor is any tensor of rank 1.
For computation we use (1,0,0)ox(1,0,0)ox(1,0,0), which can be
translated to any other rank-1 tensor by a rank-preserving
and invertible linear transformation.

Is the parametrization a subset of the tangent cone?*}
isSubset(I4,I6) {*Is the tangent cone a subset of the parametrization?*}
isSubset(I6,I4) {*Known to be true by Lemma*}
```

This program comes to the conclusion that at least for this special case the parametrization of the tangent cone is correct. The output is

```
Macaulay2, version 1.10
with packages: ConwayPolynomials, Elimination, IntegralClosure, InverseSystems,
LLLBases, PrimaryDecomposition, ReesAlgebra, TangentCone

i1 : input "TangentConeTT"
[...]
ii25 : isSubset(I4,I6) {*Is the tangent cone a subset of the parametrization?*}
oo25 = true
ii26 : isSubset(I6,I4) {*Known to be true by Lemma*}
oo26 = true
```

In the following chapter we will prove the general case analytically.

4 Parametrizing tangent cones of tensor train varieties

The content of the current chapter is a nearly one-to-one copy of [21].

After having collected evidence for the structure of the tangent cone in the previous chapter, this one contains a proof for the parametrization of tangent cones to TT varieties of arbitrary dimension (Theorem 4.1). It is the key result of this thesis. It results in the surprising Corollary 4.2, stating that the tangent cone of TT varieties is the intersection of tangent cones of the matrix varieties that are defining the TT format. In general the intersections of tangent cones does not equal the tangent cone of the intersection as Example 4.1 will show. It is open, whether this corollary can be proven in an elegant algebraic way.

However more interestingly for numerical analysis, the knowledge about the structure of the tangent cone enables a Łojasiewicz-based convergence proof for Riemannian line search methods, the topic of Chapters 5 and 6.

The proof of the main result (Theorem 4.1) uses nothing but the orthogonal projection and the corresponding result for the matrix case as proof techniques.

Lemmata 4.1 and 4.2 are trivial but essential for the proof of our main result.

Lemma 4.1.

$$\mathcal{M}^{n_1\times n_2\times n_3}_{\leq(k_1,k_2)} = \mathcal{M}^{n_1\times(n_2n_3)}_{\leq k_1} \cap \mathcal{M}^{(n_1n_2)\times n_3}_{\leq k_2}$$

Proof. by definition. □

On a subset we can only define a subset of the secants and thus a subset of the tangents.

Lemma 4.2. *For every $A \in \mathcal{M}^{n_1\times n_2\times n_3}_{\leq(k_1,k_2)}$ we have*

$$T_A\mathcal{M}^{n_1\times n_2\times n_3}_{\leq(k_1,k_2)} \subset T_A\mathcal{M}^{n_1\times(n_2n_3)}_{\leq k_1}$$

and thus

$$T_A\mathcal{M}^{n_1\times n_2\times n_3}_{\leq(k_1,k_2)} \subset T_A\mathcal{M}^{n_1\times(n_2n_3)}_{\leq k_1} \cap T_A\mathcal{M}^{(n_1n_2)\times n_3}_{\leq k_2}.$$

Proof. by definition. □

Definition 4.1. Define the range of $A \in \mathbb{R}^{n_1\times\ldots\times n_i\times k}$ as

$$\operatorname{range}(A) := \{a \in \mathbb{R}^{n_1\times\ldots\times n_i} : \exists b \in \mathbb{R}^k : a = Ab\}.$$

4.1 Parametrization of the tangent cone

We will recall the matrix case as a guiding example and as a necessary prerequisite. Along the way, we will introduce all ideas for the proof of the general case. In the following we will intentionally allow submatrices to have zero rows or columns. Consider the matrix variety

$$\mathcal{M}^{n\times m}_{\leq k+s}, \quad s \geq 0$$

i.e. the set of $n \times m$ matrices of rank at most $k + s$. Let $k \leq \min(m, n)$. Let further $A \in \mathbb{R}^{n\times k}$ and $B \in \mathbb{R}^{k\times m}$ have full rank. Then AB has rank k and is a singular point of $\mathcal{M}^{n\times m}_{\leq k+s}$. As for example shown in [1] (compare also to [24, p.256]), any tangent vector in the tangent cone at AB can be decomposed as

$$\mathcal{X} = AY + XB + UV = \begin{pmatrix} A & U & X \end{pmatrix} \begin{pmatrix} Y \\ V \\ B \end{pmatrix}$$

with $U \in \mathbb{R}^{n\times s}$ and $V \in \mathbb{R}^{s\times m}$. The converse is true by the following: The analytic arc

$$\gamma : t \mapsto \begin{pmatrix} A + tX & tU \end{pmatrix} \begin{pmatrix} B + tY \\ V \end{pmatrix}$$

lies in $\mathcal{M}^{n\times m}_{\leq k+s}$ and its derivative is $\dot\gamma(0) = AY + XB + UV$. Use $\left(\gamma\left(\frac{1}{\ell}\right)\right)_{\mathbb{N}\ni\ell\geq N}$ to see, that $\dot\gamma(0)$ lies in the tangent cone.

We can assume $A^T U = 0$ (i.e. the columns of U are orthogonal to the columns of A), $VB^T = 0$ and either $A^T X = 0$ or $YB^T = 0$ by the following argument. $P_A := AA^\dagger$ is the orthogonal projector onto range(A), where $A^\dagger$ denotes the Moore-Penrose Pseudoinverse. Defining $\dot U := A^\dagger U$ and $\hat U := (I - P_A)U$ we can decompose

$$U = P_A U + (I - P_A)U = AA^\dagger U + \hat U = A\dot U + \hat U \tag{2}$$

where $\hat U$ is orthogonal to A, i.e. $A^T \hat U = 0$. Decomposing V and X in the same way, we can write $\mathcal{X} = AY + (A\dot X + \hat X)B + (A\dot U + \hat U)(\hat V + \dot V B) = A(Y + \dot X B + \dot U \hat V + \dot U \dot V B) + \hat X B + \hat U \hat V$. We can furthermore assume U and V to have full rank by choosing them from $\mathbb{R}^{n\times\tilde s}$ and $\mathbb{R}^{\tilde s\times m}$ respectively with $\tilde s$ minimal. We introduce a definition for this, because we will need it in the tensor case.

Definition 4.2. Let $A \in \mathbb{R}^{n\times m}$ be a matrix of rank k and $A_1 \in \mathbb{R}^{n\times k}$, $A_2 \in \mathbb{R}^{k\times m}$ be such that $A = A_1 A_2$. Call for the purpose of this paper

$$\mathcal{X} = A_1 Y + X A_2 + UV$$

an *s-decomposition (with respect to A_1 and A_2)* of the matrix $\mathcal{X} \in \mathbb{R}^{n\times m}$ if $U \in \mathbb{R}^{n\times s}$, $V \in \mathbb{R}^{s\times m}$, $A_1^T X = 0$, $A_1^T U = 0$, $VA_2^T = 0$ and U and V have full rank s.

Remark 4.1. The s-decomposition depends on A_1 and A_2. Whether $\mathcal{X}$ admits one depends only on A (and s).

Lemma 4.3. *Let $\mathcal{X} \in \mathbb{R}^{n\times m}$. It admits an s-decomposition with respect to $A_1 \in \mathbb{R}^{n\times k}$ and $A_2 \in \mathbb{R}^{k\times m}$ if and only if it lies in the tangent cone of $\mathcal{M}^{n\times m}_{k+s}$ at $A := A_1 A_2$ but not in the tangent cone of $\mathcal{M}^{n\times m}_{k+s-1}$ at A. In other words, for every matrix in the tangent cone of $\mathcal{M}^{n\times m}_{k+S}$ at A there is a unique integer $0 \leq s \leq S$ such that it admits an s-decomposition with respect to A.*

Proof. See above. □

Remark 4.2. In the following we will allow matrices and tensors that have no entries because one or more of the dimensions is zero.

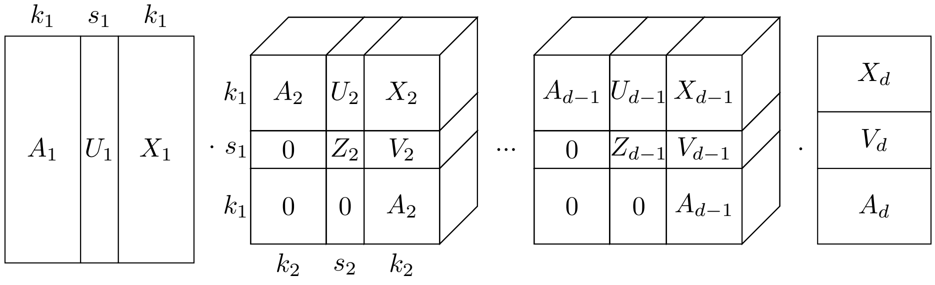
As a first step, we will prove the converse (Lemma 4.4) of our main result (Theorem 4.1) as the proof is completely analogous to the matrix case.

Lemma 4.4. *Assume $A \in \mathcal{M}^{n_1\times\ldots\times n_d}_{=(k_1,\ldots,k_{d-1})}$, i.e. there are $A_1 \in \mathbb{R}^{n_1\times k_1}$, $A_i \in \mathbb{R}^{k_{i-1}\times n_i \times k_i}$ $\forall i = 2,\ldots,d-1$ and $A_d \in \mathbb{R}^{k_{d-1}\times n_d}$ such that $A = A_1 \ldots A_d$. If a tensor $\mathcal{X}$ can be factorized as*

$$\mathcal{X} = \begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} \ldots \begin{pmatrix} A_{d-1} & U_{d-1} & X_{d-1} \\ 0 & Z_{d-1} & V_{d-1} \\ 0 & 0 & A_{d-1} \end{pmatrix} \begin{pmatrix} X_d \\ V_d \\ A_d \end{pmatrix}$$

with block matrix dimensions $(k_i + s_i + k_i) \times (k_{i+1} + s_{i+1} + k_{i+1})$ then it is in the tangent cone of $\mathcal{M}^{n_1\times\ldots\times n_d}_{\leq(k_1+s_1,\ldots,k_{d-1}+s_{d-1})}$ at $A_1 \ldots A_d$.

Figure 17: Decomposition of a tangent vector



Proof. The curve

$$\gamma : (-\varepsilon, \varepsilon) \to \mathcal{M}^{n_1\times\ldots\times n_d}_{\leq(k_1+s_1,\ldots,k_{d-1}+s_{d-1})} : t \mapsto$$

$$\begin{pmatrix} A_1 + tX_1 & U_1 \end{pmatrix} \begin{pmatrix} A_2 + tX_2 & U_2 \\ tV_2 & Z_2 \end{pmatrix} \ldots \begin{pmatrix} A_{d-1} + tX_{d-1} & U_{d-1} \\ tV_{d-1} & Z_{d-1} \end{pmatrix} \begin{pmatrix} A_d + tX_d \\ tV_d \end{pmatrix}$$

is analytic and has $\mathcal{X}$ as its first derivative. See this by differentiating γ in $t = 0$ using the product rule. Use Definition 3.16 for the tangent cone with the sequence $\left(\gamma\left(\frac{1}{\ell}\right)\right)_{\mathbb{N}\ni\ell\geq N}$. □

What follows is a technical lemma that facilitates proving both, the case for order 3 TT varieties as well as the inductive step for arbitrary order. Its first two assumptions (equations (3) and (4)) arrive from applying the matrix version to the two matricizations with respect to index 1 and 3. The idea of the proof is the following: Represent an arbitrary tangent vector as the tangent vector of the matricizations using Lemma 4.2. Then decompose using the result on matrix tangent cones above. Orthogonalizing with respect to A_1 and A_3 allows us to decompose the tangent vector into an orthogonal sum and compare the orthogonal components separately. Recall the notation $A^L := A^{n_1\times(n_2\ldots n_d)}$ and $A^R := A^{(n_1\ldots n_{d-1})\times n_d}$.

Lemma 4.5. *Let $A \in \mathcal{M}^{n_1\times n_2\times n_3}_{=(k_1,k_2)}$ be a (possibly singular) point in $\mathcal{M}^{n_1\times n_2\times n_3}_{\leq(k_1+s_1,k_2+s_2)}$ $(s_1, s_2 \geq 0)$. Consider further a decmposition $A = A_1A_2A_3$ into the three tensors $A_1 \in \mathbb{R}^{n_1\times k_1}$, $A_2 \in \mathbb{R}^{k_1\times n_2\times k_2}$ and $A_3 \in \mathbb{R}^{k_2\times n_3}$. Assume the orthogonality of A_1 and A_2, $A_1^T A_1 = I$, $\left(A_2^R\right)^T A_2^R = I$. Let $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$ be a tensor whose matricizations admit the $\tilde{s}_1$-decomposition*

$$\mathcal{X}^{n_1\times n_2n_3} = A_1\mathbf{Y}^{k_1\times n_2n_3} + X(A_2A_3)^{k_1\times n_2n_3} + U\mathbf{V}^{\tilde{s}_1\times n_2n_3} \tag{3}$$

and the $\tilde{s}_2$-decomposition

$$\mathcal{X}^{n_1n_2\times n_3} = (A_1A_2)^{n_1n_2\times k_2}T + \mathbf{S}^{n_1n_2\times k_2}A_3 + \mathbf{O}^{n_1n_2\times \tilde{s}_2}P \tag{4}$$

with $\tilde{s}_1 \le s_1$ and $\tilde{s}_2 \le s_2$. Then $\mathcal{X}$ is decomposable as

$$\mathcal{X} = \begin{pmatrix} A_1 & U & X \end{pmatrix} \begin{pmatrix} A_2 & \dot{O} & \dot{S} \\ 0 & Z_2 & \dot{V} \\ 0 & 0 & A_2 \end{pmatrix} \begin{pmatrix} T \\ P \\ A_3 \end{pmatrix} \tag{5}$$

with $\dot{O} = A_1^\dagger \mathbf{O}$, $\dot{S} = A_1^\dagger \mathbf{S}$, $\dot{V} = \mathbf{V}A_3^\dagger$ and $Z_2 = U^\dagger(\mathbf{O} - A_1\dot{O})$. In particular we have the orthogonality statements $\left(A_2^R\right)^T \dot{O}^R = 0$, $\left(A_2^R\right)^T \dot{S}^R = 0$, $\left(\dot{V}A_3\right)^L \left((A_2A_3)^L\right)^T = 0$. We also find that $Z_2P + \dot{V}A_3$ and $A_1\dot{O} + UZ_2$ have full rank and the equality

$$\begin{pmatrix} A_1 & U & X \end{pmatrix} \begin{pmatrix} A_2 & \dot{O} & \dot{S} \\ 0 & Z_2 & \dot{V} \\ 0 & 0 & A_2 \end{pmatrix} = \begin{pmatrix} A_1A_2 & \mathbf{S} & \mathbf{O} \end{pmatrix}$$

holds.

Remark 4.3. If one or two of the s_i is zero, then $U, \mathbf{V}$ or $\mathbf{O}, P$ and the corresponding submatrices in Equation 5 need to be removed. The decompositions (3) and (4) are not symmetric versions of one another in the sense that $\mathbf{Y}$ and T are both not orthogonalized but X and $\mathbf{S}$ are (see Definition 4.2). Therefore $\mathbf{Y}$ cannot play the same role as $\mathbf{S}$. Note also that $\mathbf{Y}$ does not appear in equation (5).

Proof. Define $\dot{Y} := \mathbf{Y}A_3^\dagger$, $\dot{V} := \mathbf{V}A_3^\dagger$, $\dot{T} := TA_3^\dagger$, $\dot{S} := A_1^\dagger\mathbf{S}$ and $\dot{O} := A_1^\dagger\mathbf{O}$, where $A^\dagger$ denotes the Moore-Penrose Pseudoinverse and can be replaced by A^T for orthogonal matrices and by $A^T(AA^T)^{-1}$ for full rank matrices with more columns than rows. Then we can decompose $\mathbf{Y}$, $\mathbf{V}$, $\mathbf{S}$, $\mathbf{O}$ and T into

$$\mathbf{Y} = \hat{Y} + \dot{Y}A_3,\ \mathbf{V} = \hat{V} + \dot{V}A_3,\ \mathbf{O} = \hat{O} + A_1\dot{O},\ \mathbf{S} = \hat{S} + A_1\dot{S} \text{ and } T = \hat{T} + \dot{T}A_3.$$

The hat-wearing variables are orthogonal to A_1 or A_3 respectively:

$$\hat{Y}A_3^T = 0,\ \hat{V}A_3^T = 0,\ A_1^T\hat{O} = 0,\ A_1^T\hat{S} = 0,\ \hat{T}A_3^T = 0.$$

Then we can write the tangent vector as an orthogonal sum (w.r.t. the scalar product on $\mathbb{R}^{n_1n_2n_3}$) in the four spaces

$$\begin{gathered} \operatorname{range}(A_1) \otimes \mathbb{R}^{n_2} \otimes \operatorname{range}(A_3^T), \\ \operatorname{range}(A_1) \otimes \mathbb{R}^{n_2} \otimes \operatorname{range}(A_3^T)^\perp, \\ \operatorname{range}(A_1)^\perp \otimes \mathbb{R}^{n_2} \otimes \operatorname{range}(A_3^T), \\ \operatorname{range}(A_1)^\perp \otimes \mathbb{R}^{n_2} \otimes \operatorname{range}(A_3^T)^\perp. \end{gathered}$$

Rewriting equations (3) and (4) yields

$$\mathcal{X} = A_1\dot{Y}A_3 + A_1\hat{Y} + (XA_2 + U\dot{V})A_3 + U\hat{V} \tag{6}$$

and

$$\mathcal{X} = A_1(A_2\dot{T} + \dot{S})A_3 + A_1(A_2\hat{T} + \dot{O}P) + \hat{S}A_3 + \hat{O}P \tag{7}$$

respectively. Both representations need to be equal. Because they are orthogonal sums in the same four spaces, each summand has to be equal to the corresponding summand in the other sum. In particular we have

$$\hat{O}P = U\hat{V}.$$

By defining $Z_2 := U^\dagger \hat{O}$, we can write

$$U\hat{V} = \hat{O}P = UZ_2P \tag{8}$$

and see that $Z_2 = \hat{V}P^\dagger$ (by multiplying equation (8) by the full rank matrices $U^\dagger$ and $P^\dagger$). Using the first and second summand of equation (7), the third summand of equation (6) and equation (8) we assemble the desired representation from equation (5)

$$\mathcal{X} = A_1\dot{S}A_3 + A_1A_2T + A_1\dot{O}P + XA_2A_3 + U\dot{V}A_3 + UZ_2P$$

with all the desired properties. See this in the following way: $A_1\dot{O} + UZ_2 = A_1\dot{O} + \hat{O} = \mathbf{O}$ is orthogonal to A_1A_2, therefore $0 = \left((A_1A_2)^R\right)^T \mathbf{O}^R = \left((A_1A_2)^R\right)^T \left(A_1\dot{O} + UZ_2\right)^R = \left((A_1A_2)^R\right)^T \left(A_1\dot{O}\right)^R = \left(A_2^R\right)^T \dot{O}^R$. And analogously for $Z_2P + \dot{V}A_3 = \mathbf{V}$ and $A_1\dot{S} + U\dot{V} + XA_2 = \mathbf{S}$ (by $XA_2 + U\dot{V} = \hat{S}$ from equations (20) and (7)). □

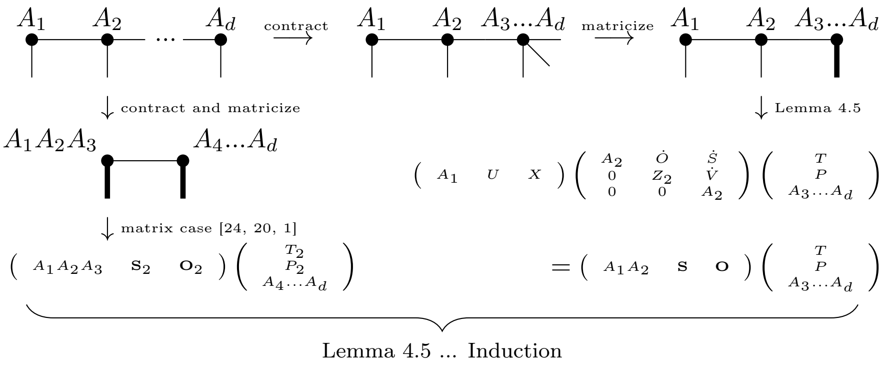
We can now state our main result for arbitrary TT varieties. Note that the condition $\left(A_i^R\right)^T U_i^R = 0$ in the theorem is equivalent to $\left((A_1...A_i)^R\right)^T (A_1...A_{i-1}U_i)^R = 0$ because we assumed the A_i to be orthogonalized. Orthogonalizing the A_i is not strictly necessary for the statement. It makes the inductive step in the proof of the theorem easier and is also neccessary to prove Lemma 4.6, which results in the convergence proof. Where $s_i = 0$ the corresponding submatrices are meant to be removed.

Theorem 4.1. *Let $A \in \mathcal{M}^{n_1\times...\times n_d}_{=(k_1,...,k_{d-1})}$ be a (possibly singular) point in $\mathcal{M}^{n_1\times...\times n_d}_{\leq(k_1+s_1,...,k_{d-1}+s_{d-1})}$ ($s_i \geq 0$) and let $A_1 \in \mathbb{R}^{n_1\times k_1}$, $A_2 \in \mathbb{R}^{k_1\times n_2\times k_2}$,... and $A_d \in \mathbb{R}^{k_{d-1}\times n_d}$ be tensors such that $A_1...A_d = A$ and $A_1^TA_1 = I$, $\left(A_i^R\right)^T A_i^R = I \; \forall i = 2, ..., d-1$. Then any vector in the tangent cone of $\mathcal{M}^{n_1\times...\times n_d}_{\leq(k_1+s_1,...,k_{d-1}+s_{d-1})}$ at the point $A_1...A_d$ can be written as the TT tensor*

$$\begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} ... \begin{pmatrix} A_{d-1} & U_{d-1} & X_{d-1} \\ 0 & Z_{d-1} & V_{d-1} \\ 0 & 0 & A_{d-1} \end{pmatrix} \begin{pmatrix} X_d \\ V_d \\ A_d \end{pmatrix} \tag{9}$$

with block matrix dimensions $(k_i + s_i + k_i) \times (k_{i+1} + s_{i+1} + k_{i+1})$. It is possible to enforce the orthogonality conditions $\left(A_i^R\right)^T U_i^R = 0 \; \forall i$, $\left(A_i^R\right)^T X_i^R = 0 \; \forall i \neq d$ and $(V_iA_{i+1}...A_d)^L \left((A_i...A_d)^L\right)^T = 0 \; \forall i$.

Figure 18: proof of the theorem



A_1 A_2 ... A_d
contract
A_1 A_2 $A_3 ... A_d$
matricize
A_1 A_2 $A_3 ... A_d$
contract and matricize
Lemma 4.5
$A_1 A_2 A_3$ $A_4 ... A_d$
$\begin{pmatrix} A_1 & U & X \end{pmatrix} \begin{pmatrix} A_2 & \dot{O} & \dot{S} \\ 0 & Z_2 & \dot{V} \\ 0 & 0 & A_2 \end{pmatrix} \begin{pmatrix} T \\ P \\ A_3 ... A_d \end{pmatrix}$
matrix case [24, 20, 1]
$\begin{pmatrix} A_1 A_2 A_3 & \mathbf{S}_2 & \mathbf{O}_2 \end{pmatrix} \begin{pmatrix} T_2 \\ P_2 \\ A_4 ... A_d \end{pmatrix}$
$= \begin{pmatrix} A_1 A_2 & \mathbf{S} & \mathbf{O} \end{pmatrix} \begin{pmatrix} T \\ P \\ A_3 ... A_d \end{pmatrix}$
Lemma 4.5 ... Induction

Proof. The idea of the proof is illustrated in Figure 18. Applying the matrix version of this theorem [20, 1, Thm 3.2] or equivalently Lemma 4.3 to the matricizations from $\mathcal{M}^{n_1 \times (n_2 ... n_d)}_{\leq (k_1 + s_1)}$ and to $\mathcal{M}^{(n_1 n_2) \times (n_3 ... n_{d-2})}_{\leq (k_2 + s_2)}$ gives us the two s_i-decompositions needed to apply Lemma 4.5 and can decompose the tangent vector in the form

$$\mathcal{X} = \begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} \begin{pmatrix} T_3 \\ P_3 \\ A_3 ... A_d \end{pmatrix}$$

with U_1 and X_1 orthogonal to A_1, the two matrices U_2 and X_2 orthogonal to A_1, $(V_2 A_3)^L$ orthogonal to $(A_2 A_3)^L$ from the left and right respectively and $A_1 U_2 + U_1 Z_2$ having full rank. Using this as inductive basis we continue by proving the inductive step: Assume that $\mathcal{X}$ has the decomposition

$$\mathcal{X} = \begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} ... \begin{pmatrix} A_i & U_i & X_i \\ 0 & Z_i & V_i \\ 0 & 0 & A_i \end{pmatrix} \begin{pmatrix} T_{i+1} \\ P_{i+1} \\ A_{i+1} ... A_d \end{pmatrix}$$

with $\left(A_i^R\right)^T U_i^R = 0 \; \forall i$, $\left(A_i^R\right)^T X_i^R = 0 \; \forall i$ and $(V_i A_{i+1} ... A_d)^L \left((A_i ... A_d)^L\right)^T = 0 \; \forall i$. Then we see that in the contraction

$$\begin{pmatrix} A_1 ... A_i & B & C \end{pmatrix} := \begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} ... \begin{pmatrix} A_i & U_i & X_i \\ 0 & Z_i & V_i \\ 0 & 0 & A_i \end{pmatrix} \tag{10}$$

B^R and C^R are both orthogonal to $(A_1 ... A_i)^R$ from the left, i.e. $\left((A_1 ... A_i)^R\right)^T B^R = 0$ and $\left((A_1 ... A_i)^R\right)^T C^R = 0$. We thus have the first assumption (equation (3)) of Lemma 4.5 for the variety $\mathcal{M}^{(n_1 ... n_i) \times n_{i+1} \times (n_{i+2} ... n_d)}_{\leq (k_i + s_i, k_{i+1} + s_{i+1})}$. The second assumption follows by the matrix version from [1].

Thus we can apply Lemma 4.5 to achieve the decomposition

$$\mathcal{X} = \begin{pmatrix} A_1 ... A_i & B & C \end{pmatrix} \begin{pmatrix} A_{i+1} & U_{i+1} & X_{i+1} \\ 0 & Z_{i+1} & V_{i+1} \\ 0 & 0 & A_{i+1} \end{pmatrix} \begin{pmatrix} T_{i+2} \\ P_{i+2} \\ A_{i+2} ... A_d \end{pmatrix}.$$

Combining this with equation (10) completes the inductive step and the proof of Theorem 4.1. □

Remark 4.4. For parametrizing the tangent cone, we use the same number of parameters as in the parametrizations of the TT variety. Each block $\begin{pmatrix} U_i & X_i \\ Z_i & V_i \end{pmatrix}$ is of size $(k_{i-1} + s_{i-1}) \times (k_i + s_i)$.

The tangent cone is a local approximation of the variety and thus has the same dimension (dimension is defined as the dimension of the smooth part of the variety). We will see in Section 4.2 that even after factoring redundancy the dimensions still match.

Lemma 4.6. *Evaluating the expression*

$$\begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} ... \begin{pmatrix} A_{d-1} & U_{d-1} & X_{d-1} \\ 0 & Z_{d-1} & V_{d-1} \\ 0 & 0 & A_{d-1} \end{pmatrix} \begin{pmatrix} X_d \\ V_d \\ A_d \end{pmatrix}$$

for the tangent cone parametrization yields

$$\begin{gathered} A_1 ... A_{d-1} X_d + A_1 ... A_{d-2} X_{d-1} A_d + ... + X_1 A_2 ... A_d \\ + A_1 ... A_{d-2} U_{d-1} V_d + A_1 ... A_{d-3} U_{d-2} V_{d-1} A_d + ... + U_1 V_2 A_3 ... A_d \\ + A_1 ... A_{d-3} U_{d-2} Z_{d-1} V_d + ... + U_1 Z_2 V_3 A_4 ... A_d \\ \vdots \\ + U_1 Z_2 ... Z_{d-1} V_d \end{gathered} \tag{11}$$

where all summands are pairwise orthogonal in the standard scalar product on $\mathbb{R}^{n_1 \cdots n_d}$.

Remark 4.5. An ALS algorithm only uses directions from the first line of this decomposition. The DMRG algorithm additionally uses directions from the second line. See [25] for a study of both, ALS and DMRG.

We can deduce, that in the case of TT varieties the intersection of the tangent cones is the tangent cone of the intersection.

Corollary 4.2.

$$\bigcap_{i=1,...,d-1} T_A \mathcal{M}_{\leq k_i}^{(n_1 ... n_i) \times (n_{i+1} ... n_d)} \subset T_A \mathcal{M}_{\leq (k_1, ..., k_{d-1})}^{n_1 \times ... \times n_d}$$

and thus

$$T_A \mathcal{M}_{\leq (k_1, ..., k_{d-1})}^{n_1 \times ... \times n_d} = \bigcap_{i=1,...,d-1} T_A \mathcal{M}_{\leq k_i}^{(n_1 ... n_i) \times (n_{i+1} ... n_d)}.$$

Proof. If

$$\mathcal{X} \in \bigcap_{i=1,...,d-1} T_A \mathcal{M}_{\leq k_i}^{(n_1 ... n_i) \times (n_{i+1} ... n_d)}$$

then by Lemma 4.5 we can find coefficient tensors such that we can write $\mathcal{X}$ in our parametrization. But then by Lemma 4.4

$$\mathcal{X} \in T_A \mathcal{M}^{n_1 \times \ldots \times n_d}_{\leq (k_1, \ldots, k_{d-1})}.$$

□

This corollary was unexpected because of the following example.

Example 4.1. The tangent cone of the intersection is not always equal to the intersection of the tangent cones. Consider the plane $\mathcal{M} := \{(x, y, z) \in \mathbb{R}^3 : x = 0\}$ and the cylinder $\mathcal{N} := \{(x, y, z) \in \mathbb{R}^3 : (x - 1)^2 + y^2 = 1\}$ and the point $(0, 0, 0) \in \mathcal{N} \cap \mathcal{M}$. Being the line where both varieties touch, the tangent cone $T_A\mathcal{M}$ of $\mathcal{M}$ at A is the same as the tangent cone of $\mathcal{N}$ at A, namely the y-z-plane. However the tangent cone of $\mathcal{M} \cap \mathcal{N} = \{(x, y, z) \in \mathbb{R}^3 : x = y = 0\}$ at A is only the z-axis.

We can show that the issue raised in example 3.14 is unimportant for TT varieties. Namely:

Corollary 4.3. *The tangent cone to a TT variety is equal to the set of all first derivatives to analytic arcs.*

Proof. By Theorem 4.1 every tangent vector can be written in the form

$$\begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} \ldots \begin{pmatrix} A_{d-1} & U_{d-1} & X_{d-1} \\ 0 & Z_{d-1} & V_{d-1} \\ 0 & 0 & A_{d-1} \end{pmatrix} \begin{pmatrix} X_d \\ V_d \\ A_d \end{pmatrix}$$

and by Lemma 4.4 this is the first derivative of the analytic curve

$$\gamma : t \mapsto$$

$$\begin{pmatrix} A_1 + tX_1 & U_1 \end{pmatrix} \begin{pmatrix} A_2 + tX_2 & U_2 \\ tV_2 & Z_2 \end{pmatrix} \ldots \begin{pmatrix} A_{d-1} + tX_{d-1} & U_{d-1} \\ tV_{d-1} & Z_{d-1} \end{pmatrix} \begin{pmatrix} A_d + tX_d \\ tV_d \end{pmatrix}.$$

The converse is trivial by using the sequence $\left(\eta\left(\frac{1}{m}\right)\right)_{\mathbb{N} \ni m \geq N}$ for an analytic curve η. □

4.2 Uniqueness and dimension

In which sense is the representation given in Theorem 4.1 unique? To answer this question, we revisit the matrix case. Consider a rank-k matrix A, that can be decomposed into two matrices of full rank $A = A_1 A_2$. As we have seen in the previous section, any tangent vector at A in $\mathcal{M}_{\leq k+s}$ can be written in the form

$$\mathcal{X} = A_1 Y + X A_2 + UV$$

with X and U orthogonal to A_1 and with V orthogonal to A_2. We furthermore imposed the restriction, that the dimension $\tilde{s}$ of U and V shall be as small as possible. This is equivalent to demanding U and V to have full rank.

In this representation X and Y are uniquely defined by

$$Y := A_1^\dagger \mathcal{X}$$

and

$$X := (\mathcal{X} - A_1 Y) A_2^\dagger. \tag{12}$$

U and V however are not uniquely defined. The product UV is equal to the remainder

$$UV = \mathcal{X} - A_1 Y - X A_2.$$

How this remainder is factorized however is subject to some redundancy. We can introduce a big identity between the two. If G is an invertible $\tilde{s} \times \tilde{s}$ matrix, then

$$(UG)(G^{-1}V) = UV$$

where UG and $G^{-1}V$ admit the same orthogonality restrictions as are imposed on U and V above respectively. This is a well-known technique, that has previously been used in [19] in the setting of quotient manifolds through Lie group actions.

These considerations generalize in a straight-forward way to the TT case. For example in the order 3 case, an element of the Lie group consists of the two matrices $G \in \mathbb{R}^{\tilde{s_1} \times \tilde{s_1}}$ and $F \in \mathbb{R}^{\tilde{s_2} \times \tilde{s_2}}$. The result of its action is

$$\left(\begin{array}{ccc} A_1 & U_1 G & X_1 \end{array} \right) \left(\begin{array}{ccc} A_2 & U_2 F & X_2 \\ 0 & G Z_2 F & G V_2 \\ 0 & 0 & A_2 \end{array} \right) \left(\begin{array}{c} X_3 \\ F V_3 \\ A_3 \end{array} \right).$$

The action does not affect any of the orthogonality properties mentioned in Theorem 4.1. Note how Z_2 is not a matrix but a tensor. Therefore the actions from left and right do not permit to simplify the representation in the seemingly possible way (forcing it to be "diagonal").

We want to count the dimensions of the TT variety

$$\mathcal{M}^{n_1 \times \ldots \times n_d}_{\leq (k_1 + s_1, \ldots, k_{d-1} + s_{d-1})} \tag{13}$$

and its tangent cone

$$T_{A_1 \ldots A_d} \mathcal{M}^{n_1 \times \ldots \times n_d}_{\leq (k_1 + s_1, \ldots, k_{d-1} + s_{d-1})} \tag{14}$$

at an arbitrary, possibly singular point

$$A_1 \ldots A_d \in \mathcal{M}^{n_1 \times \ldots \times n_d}_{= (k_1, \ldots, k_{d-1})}.$$

A parametrized variety is irreducible [10, p.199] and thus its smooth part has constant dimension [26, p.101f]. Therefore the dimension of the variety (13) is the dimension of its smooth part, that is, the dimension of the manifold

$$\mathcal{M}^{n_1 \times \ldots \times n_d}_{= (k_1 + s_1, \ldots, k_{d-1} + s_{d-1})}$$

which is known to be

$$n_1 r_1 + r_1 n_2 r_2 + \ldots + r_{d-1} n_d - r_1^2 - \ldots - r_{d-1}^2. \tag{15}$$

Here the positive terms count the degrees of freedom in the parametrization and the negative terms count the dimension of the Lie group action that keeps the tensor invariant. See [19] for the rigorous proof.

For counting the dimensions of the tangent cone (14) we first find a smooth point on it. Consider the parametrization (9). The U_i are orthogonal to the A_i from the left in the standard scalar product and V_i are orthogonal to the A_i from the right in the scalar product defined by $A_{i+1} \ldots A_d$. Choosing all U_i to have full rank s_i and all V_i to have full rank s_{i-1} - combined with the triangular block structure of (9) - implies that the component tensors of the parametrization of the tangent cone all have full rank. Equation (11) is a decomposition of the tangent vector into components from orthogonal spaces. These orthogonal spaces can be defined by the ranges and the orthogonal complement of the ranges of the A_i. Thus we can count the dimensions seperately.

For all terms

$$U_i V_{i+1}$$

in (11) the U_i and V_i define Grassmannians in the orthogonal complement of the columns of $A_1 ... A_{i-1}$ and $A_{j+1} ... A_d$ respectively. Their combined dimension is

$$k_{i-1} n_i s_i + s_i n_{i+1} k_{i+1} - s_i^2 - s_i k_i - s_i k_i \tag{16}$$

where we define $k_0 := 1$ and $k_d := 1$. The $s_i k_i$-terms come from the orthogonality conditions and the s_i^2 from the Lie group action of putting a big identity in between U_i and V_{i+1}. The full rank A_i in front and after $U_i V_{i+1}$ do not change the dimension. From the terms with only one Z_i we can see that the degrees of freedom of the Z_i are directly mapped to dimensions of the manifold parametrized by

$$A_1 ... A_{i-2} U_{i-1} Z_i V_{i+1} A_{i+2} ... A_d$$

because the A_i, U_i and V_i all have full rank. The resulting dimension is

$$s_{i-1} n_i s_i$$

with $s_0 := 1$ and $s_d := 1$. The dimensions of the linear part - i.e. the first row in (11) - are known from [19] as the dimension of the linear tangent space:

$$n_1 k_1 + k_1 n_2 k_2 + ... + k_{d-1} n_d - k_1^2 - ... - k_{d-1}^2.$$

In particular the X_i are uniquely defined by projections. To see this, examine the definitions of X, $\dot{S}$ and T in (5).

All other terms with more than one Z_i are dependent on the former terms because they do not contain any new variables and thus do not contribute any new dimensions. The overall dimension count of the tangent cone is thus the same as for the variety, which is in accordance to the basic theory of Algebraic Geometry.

4.3 A remarkable a priori statement or a global optimality condition

In [1, Cor. 3.4] it has been noted for the matrix case that the tangent cone has the following rather remarkable property if it is the tangent cone to a singular point: It spans the whole matrix space. This is due to the following observation. If in a singular point $A = A_1 A_2$ a tangent vector takes the form

$$\mathcal{X} = A_1 Y + X A_2 + UV.$$

This representation is able to reproduce every rank-1 matrix $\mathcal{Y}$. Given some rank-1 matrix, then Y and X and the product UV are uniquely determined by the formulas given in the previous section. These formulas represent orthogonal projections and therefore do not increase the rank and their results fit into X, Y and UV. Therefore every rank-1 matrix can be represented as a tangent vector with UV having rank 1 or vanishing.

There is an even simpler way to see that all rank-1 matrices are contained in the tangent cone of a singular point on a matrix variety. The rank of the matrix A is strictly smaller than k. Thus every sum

$$A + \varepsilon \mathcal{Y}$$

is in $\mathcal{M}_{\leq k}$. As you can see, this is a whole line when varying ε, that lies in the variety. A line through A that lies completely in the variety must obviously be a tangent line.

In the TT case, singularity of a point does not imply that its tangent cone spans the whole tensor space. A point on the TT variety is singular if at least one of the TT ranks is deficient. A tensor A having a deficient TT rank at position i in the variety $\mathcal{M}^{n_1\times\ldots\times n_d}_{\leq(k_1,\ldots,k_d)}$ means that it lies in some variety $\mathcal{M}^{n_1\times\ldots\times n_d}_{\leq(\tilde{k}_1,\ldots,\tilde{k}_d)}$ with $\tilde{k}_i < k_i$. However all ranks have to be deficient to ensure that the tangent cone includes rank-1 tensors and thus spans the whole tensor space.

Take for example the order 3 case with $s_1 = 1$ and $s_2 = 0$ (using the notation from Theorem 4.1). Then a tangent vector admits the decomposition

$$\begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & X_2 \\ 0 & V_2 \\ 0 & A_2 \end{pmatrix} \begin{pmatrix} X_3 \\ A_3 \end{pmatrix}.$$

Rewriting this as

$$A_1 A_2 X_3 + (\ldots) A_3$$

shows that the tangent cone is part of the linear subspace

$$T\mathcal{M}^{(n_1 n_2)\times n_3}_{=k_2}.$$

This linear subspace is the tangent space of a matrix manifold and in general a proper subspace of the whole tensor space.

Showing that the tangent cone spans the whole space if all ranks are deficient is a matter of generalizing the above reasoning from the matrix to the tensor case in a straight-forward way.

In [27] this topic is discussed for canonical tensor decompositions and neural networks. However, in contrast to [27, Thm 15], we and [1] do not assume a regularization term and arrive at a similar global optimality condition for the special case of tensor train decompositions: A local minimizer of a convex differentiable function

$$f : \mathbb{R}^{n_1\times\ldots\times n_d} \to \mathbb{R}$$

on the TT variety, which is rank deficient in each TT rank, must be a global minimizer. To relate rank deficiency and the zero slices assumed in [27], see that the following TT decomposition containing zero slices

$$\begin{pmatrix} A_1 & x_1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} A_2 & x_2 \\ 0 & 0 \end{pmatrix} \cdots \begin{pmatrix} A_d \\ 0 \end{pmatrix} = \begin{pmatrix} A_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} A_2 & 0 \\ 0 & 0 \end{pmatrix} \cdots \begin{pmatrix} A_d \\ 0 \end{pmatrix}$$

is rank deficient in each TT rank.

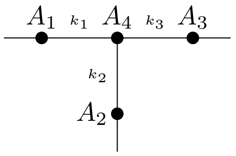
4.4 The hierarchical format

All of the above generalizes in a straight-forward way to the hierarchical and Tucker format. However the notation is difficult. We will show only the proof ideas and check the facts that we expect to be the key elements of a future more detailed and formal proof.

See Definition 3.9 or [14, 28] for a detailed study of the hierarchical tensor format. We will only give the equivalent of the technical Lemma 4.5 for the Tucker format with order 3. This will allow us to use the same inductive step as in Theorem 4.1 to prove the parametrization for any binary tree. In further generalizing the technical lemma to arbitrary Tucker formats, one could prove the theorem for arbitrary tree formats.

Let $A_1 \in \mathbb{R}^{n_1\times k_1}$, $A_2 \in \mathbb{R}^{k_2\times n_2}$, $A_3 \in \mathbb{R}^{n_3\times k_3}$ and $A_4 \in \mathbb{R}^{k_1\times k_2\times k_3}$ with A_1, A_2 and A_3 having full rank. For writing simple tensor tree diagrams, we can use the Kronecker product. Sorting the indices k_1 and k_3 lexicographically, we can identify the tree diagram and the term depicted in Figure 19.

Figure 19: Kronecker product notation for tensor trees



$$= \left((A_1 \otimes A_3)A_4^{(k_1k_3)\times k_2} A_2\right)^{n_1\times n_2\times n_3}$$

We can write this in the following three ways:

$$\begin{aligned} &\left((A_1 \otimes A_3)A_4^{(k_1k_3)\times k_2}\right) \cdot A_2 \\ &= A_1 \cdot \left(A_4^{k_1\times(k_3k_2)}(A_3 \otimes A_2)\right) \\ &= A_3 \cdot \left(A_4^{k_3\times(k_1k_2)}(A_1 \otimes A_2)\right). \end{aligned}$$

Now any tangent vector from a tucker variety $\mathcal{M}^{n_1\times n_2\times n_3}_{\leq(k_1+s_1,k_2+s_2,k_3+s_3)}$ (we use the obvious generalization of the symbols defined for the TT varieties) parametrized by A_1, A_2, A_3 and A_4 can be decomposed in the $\tilde{s}_2$-decomposition

$$\left((A_1 \otimes A_3)A_4^{(k_1k_3)\times k_2}\right) Y_2 + \mathbf{X}_2 A_2 + \mathbf{U}_2 V_2,$$

in the $\tilde{s}_1$-decomposition

$$A_1\mathbf{Y}_1 + X_1 \left(A_4^{k_1\times(k_2k_3)}(A_2 \otimes A_3)\right) + U_1\mathbf{V_1}$$

and the $\tilde{s}_3$-decomposition

$$A_3\mathbf{Y}_3 + X_3 \left(A_4^{k_3\times(k_1k_2)}(A_1 \otimes A_2)\right) + U_3\mathbf{V}_3$$

with $\tilde{s}_1 \leq s_1$, $\tilde{s}_2 \leq s_2$ and $\tilde{s}_3 \leq s_3$. We can further decompose each of the three into the 8 orthogonal subspaces

$$\begin{aligned} \operatorname{range}(A_1) \otimes \operatorname{range}(A_2^T) \otimes \operatorname{range}(A_3), &\quad \operatorname{range}(A_1) \otimes \operatorname{range}(A_2^T) \otimes \operatorname{range}(A_3)^\perp, \\ \operatorname{range}(A_1)^\perp \otimes \operatorname{range}(A_2^T) \otimes \operatorname{range}(A_3), &\quad \operatorname{range}(A_1)^\perp \otimes \operatorname{range}(A_2^T) \otimes \operatorname{range}(A_3)^\perp, \\ \operatorname{range}(A_1) \otimes \operatorname{range}(A_2^T)^\perp \otimes \operatorname{range}(A_3), &\quad \operatorname{range}(A_1) \otimes \operatorname{range}(A_2^T)^\perp \otimes \operatorname{range}(A_3)^\perp, \end{aligned}$$

$$\operatorname{range}(A_1)^\perp \otimes \operatorname{range}(A_2^T)^\perp \otimes \operatorname{range}(A_3), \quad \operatorname{range}(A_1)^\perp \otimes \operatorname{range}(A_2^T)^\perp \otimes \operatorname{range}(A_3)^\perp.$$

Exemplarily we further decompose the $\tilde{s}_1$-decomposition. For this purpose we need to write $\mathbf{Y}_1$ as the orthogonal sum

$$\mathbf{Y}_1^{k_1\times(n_2n_3)} = (I\otimes A_3)\dot{Y}_1A_2 + Y_1^3A_2 + (I\otimes A_3)Y_1^2 + Y_1^{2,3}$$

such that $(I\otimes A_3)^TY_1^3 = 0$, $Y_1^2A_2^T = 0$, $(I\otimes A_3)^TY_1^{2,3} = 0$ and $Y_1^{2,3}A_2^T = 0$ (use pseudo inverses for this purpose as in equation (2)). Analogously we rewrite $\mathbf{V}_1$ as

$$\mathbf{V}_1^{k_1\times(n_2n_3)} = (I\otimes A_3)\dot{V}_1A_2 + V_1^3A_2 + (I\otimes A_3)V_1^2 + V_1^{2,3}$$

such that the $\tilde{s}_1$-decomposition can be rewritten as the orthogonal sum

$$\begin{gathered}(A_1\otimes A_3)\dot{Y}_1A_2\\ +(A_1\otimes I)Y_1^3A_2\\ +(A_1\otimes A_3)Y_1^2\\ +(A_1\otimes I)Y_1^{2,3}\\ +((U_1\otimes A_3)\dot{V}_1 + (X_1\otimes A_3)A_4)A_2\\ +U_1V_1^3A_2\\ +(U_1\otimes A_3)V_1^2\\ +U_1V_1^{2,3}.\end{gathered}$$

Comparing coefficients with the orthogonal decompositions of the $\tilde{s}_2$- and $\tilde{s}_3$-decompositions, we arrive at the representation

$$\mathcal{X} = \Big(\begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \otimes \begin{pmatrix} A_3 & U_3 & X_3 \end{pmatrix}\Big)\,\mathbf{C}\begin{pmatrix} Y_2\\ V_2\\ A_2\end{pmatrix}$$

with $\mathbf{C}\in\mathbb{R}^{(k_1+\tilde{s}_1+k_1)(k_3+\tilde{s}_3+k_3)\times(k_2+\tilde{s}_2+k_2)}$ having the form depicted in Figure 20.

The coefficients of the block tensor $\mathbf{C}$ are

$$X_4 = (A_1^\dagger\otimes A_3^\dagger)\mathbf{X}_2, \quad Z_4 = (U_1^\dagger\otimes U_3^\dagger)U_2^{1,3}, \quad U_4 = \dot{V}_1,$$

$$W_4 = \dot{U}_2, \quad V_4 = \dot{V}_3, \quad \bar{V}_4 = V_1^2V_2^\dagger = (U_1^T\otimes I)U_2^1,$$

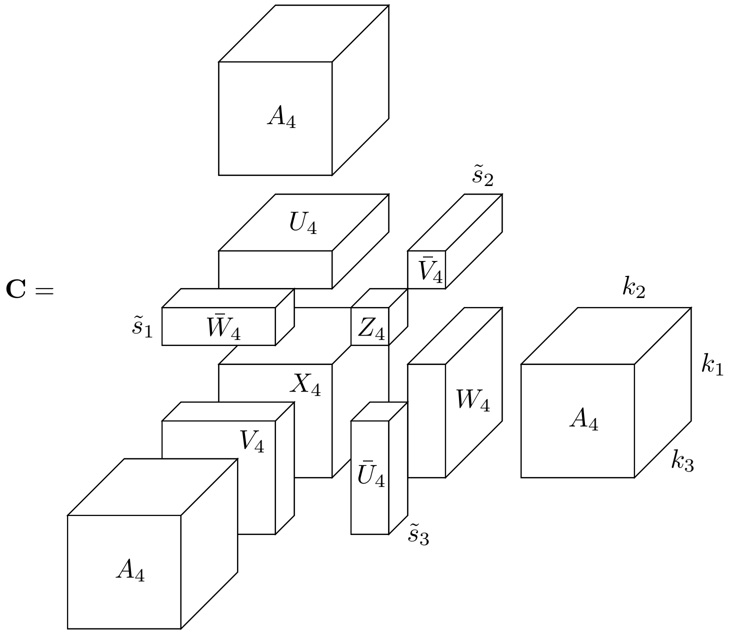
$$\bar{W}_4 = (U_1^\dagger\otimes U_3^\dagger)X_2^{1,3} \quad\text{and}\quad \bar{U}_4 = (I\otimes U_3^\dagger)U_2^3.$$

The inductive step works because by

$$\Big((A_1\otimes A_3)A_4^{(k_1k_3)\times k_2},\, \mathbf{U}_4,\, \mathbf{X}_4\Big) = \Big(\begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \otimes \begin{pmatrix} A_3 & U_3 & X_3 \end{pmatrix}\Big)\,\mathbf{C}$$

we can reduce the parametrization to the matrix case and reproduce $\mathbf{U}_4$ and $\mathbf{X}_4$.

Figure 20: Central coefficient tensor of tangent cone parametrization for order 3 Tucker



4.5 Implicit description of the tangent cone

The tangent cone for the matrix case can be implicitely defined as the variety

$$\left\{\mathcal{X} \in \mathbb{R}^{n\times m} : \operatorname{rank}\left((I - A_1A_1^{\dagger})\mathcal{X}(I - A_2^{\dagger}A_2)\right) \leq s_1\right\}$$

where the rank can be bounded by a set of determinants of minors. Since we have shown in Corollary 4.2 that the tangent cone of a tensor variety is the intersection of tangent cones of matrix varieties, the set of defining equations of the tensor variety is the union of defining equations of matrix varieties of the appropriate matricizations.

5 Projected gradients and the angle condition

5.1 Projected gradient

In local optimization, instead of searching for the minimum of a function $f : \mathbb{R}^n \to \mathbb{R}$ directly, one looks for a point, at which the first derivative in every direction vanishes. Such a point x^* where

$$f'(x^*) \cdot v = \frac{\partial f}{\partial v}(x^*) = 0 \text{ for every direction } v$$

is called a *critical point*. On an algebraic variety $\mathcal{M}$ the set of directions is the tangent cone. Therefore we call a point $x^* \in \mathcal{M}$ on the variety a *critical point* if the derivative in every tangent direction $v \in T_{x^*}\mathcal{M}$ vanishes, that is if

$$f'(x^*) \cdot v = 0 \;\forall v \in T_{x^*}\mathcal{M}.$$

This is equivalent to asking that the best approximation

$$P_{T_{x^*}\mathcal{M}}\left(\nabla f(x^*)\right)$$

of the gradient of f at x^* vanishes. The latter criterion is the more useful one for us. We will even introduce the name

$$f^{'}_{\mathcal{M}}(x) := P_{T_x\mathcal{M}}\left(-\nabla f(x)\right)$$

for the set of *best approximations of the* (or synonymously *projected) antigradient*, the points on $T_x\mathcal{M}$ that are closest to $-\nabla f(x)$. This definition is also used in for example [1] or [29]. See Figure 21 for an illustration.

Remark 5.1. On a Riemannian manifold the best approximation of the gradient is equivalent to the intrinsic (Riemannian) gradient of f when restricted to the manifold, i.e. the vector in the tangent space that, when multiplied by any other tangent vector X produces the directional derivative of f in direction X. However for tangent cones this relationship appears to be less obvious.

Just as for linear subspaces, in a cone the projected gradient also has the smallest possible angle to the vector being approximated.

Lemma 5.1. *Let $C \subset \mathbb{R}^n$ be a cone, which means that $x \in C$ implies $\alpha x \in C$ for all non-negative real α. Let further $v \in \mathbb{R}^n$ be any vector and p be one of its best approximations in C, that is $||p - v|| \leq ||x - v|| \;\forall x \in C$. Then if $p \neq 0$,*

$$\sphericalangle(v, p) \leq \sphericalangle(v, x) \;\forall x \in C$$

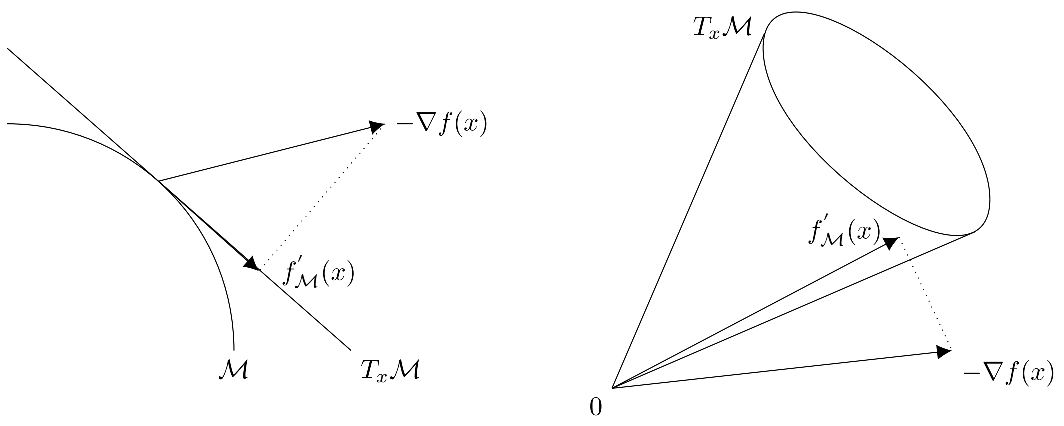
or equivalently

$$\langle v, \frac{p}{||p||}\rangle \geq \langle v, \frac{x}{||x||}\rangle \;\forall x \in C.$$

Conversely if $p \in C$ is such that $\sphericalangle(p, v) \leq \sphericalangle(x, v) \;\forall x \in C$ then there is a non-negative α such that αp is a best approximation of v in C.

Proof. Angles between linear subspaces are contained in $[0, \pi]$. Let p be a best approximation of v as in the formulation of the lemma and α the angle between x and p. $\alpha \in [0, \pi)$ because the best approximation p does not vanish. Suppose there is a direction x with an angle between span(x)

Figure 21: Best approximation of the gradient



and v that is smaller than α. Then $\langle v, \frac{p}{||p||}\rangle \le \langle v, \frac{x}{||x||}\rangle$. Then on span$(x)$ we find a point that is closer to v than p is, namely the orthogonal projection of v onto span(x):

$$||\frac{\langle v,x\rangle}{\langle x,x\rangle}x - v|| = \langle v,v\rangle - \langle v, \frac{x}{||x||}\rangle^2 < \langle v,v\rangle - \langle v, \frac{p}{||p||}\rangle^2 = ||\frac{\langle v,p\rangle}{\langle p,p\rangle}p - v|| \le ||p - v||.$$

We have used the monotonicity of the Cosine on $[0,\pi]$ and of $y \mapsto y^2$ on the positive real axis as well as the equivalence of orthogonal projection and best approximation for a linear subspace. The converse is proven with $\alpha := \frac{\langle v,p\rangle}{\langle p,p\rangle}$ by the inequality

$$||\alpha p - v|| = ||\frac{\langle v,p\rangle}{\langle p,p\rangle}p - v|| = \langle v,v\rangle - \langle v, \frac{p}{||p||}\rangle^2 \le \langle v,v\rangle - \langle v, \frac{x}{||x||}\rangle^2 = ||\frac{\langle v,x\rangle}{\langle x,x\rangle}x - v||$$

which shows that the orthogonal projection onto span(p) is the best approximation in C under the assumption $\sphericalangle(v,p) \le \sphericalangle(v,x)\ \forall x \in C$. $\square$

We can also formalize the mentioned orthogonality property in the following Lemma.

Lemma 5.2. *If p is a best approximation of v in C, then the distance $(p - v)$ is orthogonal to p.*

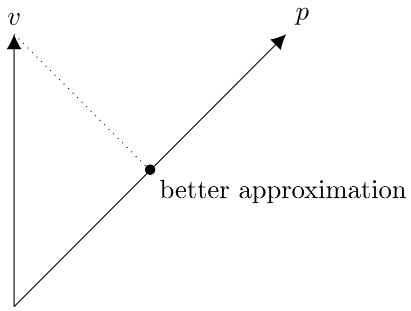
Proof. Let $p \in C$ be a best approximation of v. If $\langle v - p, p\rangle$ was not 0, then project v onto the one-dimensional subspace spanned by p to produce a better approximation. See Figure 22. $\square$

5.2 Tangent spaces at singular points

For every tangent cone $T_x\mathcal{M}$ there is a linear subspace of the ambient space

$$\mathcal{T}_x\mathcal{M}$$

Figure 22: approximation and orthogonality



spanned by all tangent vectors. This is not the same as the tangent cone. In the literature (for example [24] or [10]) this is also called the *tangent space.* In regular points the tangent space coincides with the tangent cone. In singular points the tangent space can have significantly higher dimension than the tangent cone. That is for example the case in the singular points of low-rank matrix and tensor varieties. What follows is a different viewpoint on Section 4.3. In a singular point $x_{sing} \in \mathcal{M}_s$ of the low-rank matrix variety $\mathcal{M}_{\leq k}$, $s < k$ the tangent cone contains the set of rank-1 matrices. The set of rank-1 matrices contains a basis for the set of all matrices, for example those with only one entry. The important implication is thus, that we cannot find a vector that is orthogonal to the whole tangent space (except for the 0-vector). Or in more optimistic terms: If the gradient does not vanish, then it has always a projection of positive length on the tangent cone $T_{x_{sing}}\mathcal{M}_{\leq k}$. In terms of [1] if a singular point is critical on $\mathcal{M}_{\leq k}$ then it is automatically critical on the whole space $\mathbb{R}^{n\times m}$.

5.3 Angle condition

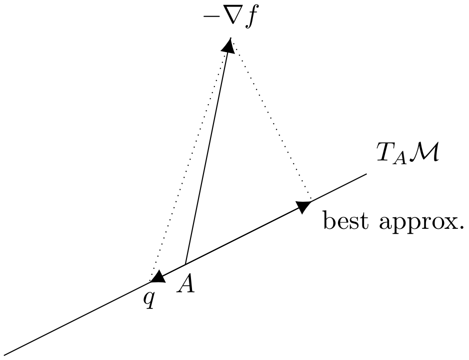
For the matrix variety $\mathcal{M}_{\leq k}$ it is possible to calculate the exact projection onto the tangent cone even in singular points using the singular value decomposition. For low-rank tensor varieties however such an algorithm for an exact projection onto $T_x\mathcal{M}_{\leq(k_1,\dots k_{d-1})}$ is not known to us. The tangent cone contains sheared low-rank tensor varieties. Thus one essential difficulty lies in the exact projection onto the set of low-rank tensors $\mathcal{M}_{\leq(r_1,\dots,r_l)}$.

Remark 5.2. Contrary to intuition, the quasi-best approximation does not satisfy an angle condition, i.e. the angle between a vector v and its quasi-best approximation cannot be bounded by a constant multiple of the angle between v and its best approximation. In Figure 23 we have illustrated a counterexample. The distance of $-\nabla f$ to q is less than twice as much as the distance of $-\nabla f$ to its best approximation (projected antigradient). Thus q is a quasi-best approximation with factor 2. But the best approximation and q point in opposite directions, which makes q as unrelated to the projected antigradient as possible.

The idea is to replace the exact projection with an approximation that still suits our needs:

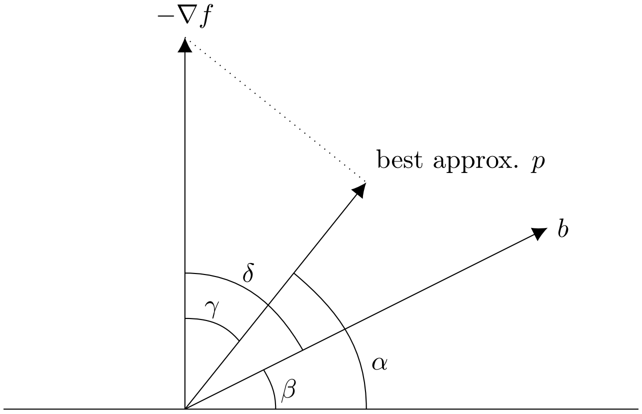
- When the exact projection is 0 the approximated projection should also vanish.

Figure 23: quasi-best approximation does not ensure angle condition



- The angle between the approximated projection and $-\nabla f$ should not be much bigger (and certainly not bigger than 90 degrees) than that between the exact projection and $-\nabla f$.

Figure 24: angle condition



As in [1] we formalize the second necessitiy in a way that will include the first one as a special case.

Definition 5.1. (angle condition) Let ∇f be the gradient of some function $f : \mathbb{R}^N \to \mathbb{R}$ at x and p a best approximation of $-\nabla f$ on the tangent cone $T_x\mathcal{M}$. Let $b \in T_x\mathcal{M}$ be some vector in the cone. Let $\gamma := \sphericalangle(-\nabla f, p)$ and $\delta := \sphericalangle(-\nabla f, b)$ be the angles between the antigradient and p and b respectively. Let $\alpha := \frac{\pi}{2} - \gamma$ and $\beta := \frac{\pi}{2} - \delta$ be as in Figure 24. Then b is said to satisfy an ω-angle condition if

$$\beta \geq \omega\alpha$$

for some real positive $\omega \in (0, 1]$.

The ω-angle condition

$$\beta \geq \omega\alpha$$

is equivalent to

$$\sin(\beta) \geq \omega \sin(\alpha)$$

and to

$$\cos(\delta) \geq \cos(\gamma)$$

and by the definition of an angle to

$$\langle -\frac{\nabla f}{||\nabla f||}, \frac{b}{||b||}\rangle \geq \omega \langle -\frac{\nabla f}{||\nabla f||}, \frac{p}{||p||}\rangle.$$

Multiplying by $||\nabla f||$ we get

$$\langle -\nabla f, \frac{b}{||b||}\rangle \geq \omega \langle -\nabla f, \frac{p}{||p||}\rangle$$

and using Lemma 5.2 (and the orthogonality of the best approximation $\langle -\nabla f - p, p\rangle = 0 \Leftrightarrow \langle -\nabla f, p\rangle = \langle p, p\rangle$) this reduces to

$$\langle -\nabla f, \frac{b}{||b||}\rangle \geq \omega ||p||.$$

Multiplying by $-||b||$ we arrive at the version

$$\langle \nabla f, b\rangle \leq -\omega ||p|| ||b||$$

that has been used in [1].

5.4 Angle condition for the TT variety

We would like to have an angle condition for the tangent cone of a TT variety. That is, given the gradient of some function at a point on the TT variety, we search for a tensor from the tangent cone whose angle with the gradient is not much worse than that of the best approximation on the tangent cone. At the same time, this tensor should be computable within reasonable complexity. As we have seen in Lemma 4.6, the TT tangent cone can be written as the sum of orthogonal components. One of these components is the tangent space to a TT manifold of smaller rank

$$X_1A_2...A_d + A_1X_2A_3...A_d + ... + A_1...A_{d-1}X_d. \tag{17}$$

The projection onto this part is easy and well known. See for example [30]. We assume again

$$\left((A_1...A_i)^R\right)^T (A_1...A_i)^R = I \ \forall i \text{ and } \left(A_i^R\right)^T X_i^R = 0 \ \forall i \neq d.$$

From this we see that all the summands in Term 17 are orthogonal. Thus it suffices to project onto each summand individually. For example the projection of Y onto the last summand is

$$(A_1...A_{d-1})^R (A_1...A_{d-1})^T Y,$$

the projection yielding the second summand is

$$A_1 \left(I - A_2^R (A_2^R)^T\right) A_1^T Y \left((A_3...A_d)^L\right)^T (A_3...A_d)^L$$

and the projection yielding the first summand is

$$\left(I - A_1^R (A_1^R)^T\right) Y \left((A_2...A_d)^L\right)^T (A_2...A_d)^L.$$

5.4.1 Nonlinear terms in the TT tangent cone

Of the orthogonal components of the TT tangent cone identified in Lemma 4.6 the only one linear is Equation 17. All other components are linearly transformed projections of TT varieties. As an example consider

$$A_1 U_2 Z_3 V_4 A_5.$$

The term $U_2 Z_3 V_4$ parametrizes a TT variety of bounded rank. The orthogonality conditions

$$\left(A_2^R\right)^T A_2^R U_2 = 0 \text{ and } (V_4 A_5)^L \left((A_4 A_5)^L\right)^T = 0$$

on U_2 and V_4 can be seen as a projection of this variety onto a linear subspace. We will be able to reduce everything to projecting onto a variety of bounded TT rank. Therefore we investigate this topic first.

5.4.2 Plan

Let $\mathcal{M} := \mathcal{M}^{n_1 \times \ldots \times n_d}_{\leq (k_1, \ldots, k_d)}$ be a TT variety of bounded rank. Given some tensor $v \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ and its best approximation p on $\mathcal{M}$, the aim is to find a tensor $b \in \mathcal{M}$ such that for some a priori known constant ω we have

$$\langle \frac{v}{||v||}, \frac{b}{||b||} \rangle \geq \omega \langle \frac{v}{||v||}, \frac{p}{||p||} \rangle.$$

As

$$\langle \frac{v}{||v||}, \frac{p}{||p||} \rangle \leq 1$$

by definition, it suffices to show that there is a tensor b of length 1 satisfying

$$\langle \frac{v}{||v||}, b \rangle \geq \omega$$

or equivalently

$$\langle v, b \rangle^2 \geq \omega^2 \langle v, v \rangle.$$

In words: Take any tensor $v \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ of length 1. We can always find a tensor b of length 1 in the TT variety that has a positive scalar product with v that is greater than an a priori known constant ω. In fact it suffices to restrict ourselves to TT rank $(1, ..., 1)$.

5.4.3 Angle condition for rank-1 tensor varieties

The TT variety of rank $(1, ..., 1)$ is the same set as the Tucker variety of rank $(1, ..., 1)$ and the same as the canonical variety of rank 1. Therefore we can just call it the rank-1 variety.

Lemma 5.3. *Let $Y \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ be some tensor. Then there is a tensor B of rank 1 such that the angle condition*

$$||B||_F = 1 \text{ and } \langle Y, B \rangle_F^2 \geq \omega \langle Y, Y \rangle_F$$

holds for

$$\omega = \frac{1}{n_1 \ldots n_d}.$$

Proof. Find the position $(i_1, ..., i_d)$ of the entry in Y of greatest absolute value. At the same position B has its only non-zero entry $B_{i_1,...,i_d} = 1$. Then we have

$$\langle Y, B\rangle_F^2 = \left(Y_{i_1,...,i_d}\right)^2 = \frac{1}{n_1...n_d}\left(\left(n_1...n_d\right)\left(Y_{i_1,...,i_d}\right)^2\right) \geq \omega \sum_{j_1,...,j_d}^{n_1,...,n_d} \left(Y_{j_1,...,j_d}\right)^2 = \omega\langle Y, Y\rangle_F.$$

□

Now the problem is, that finding the largest entry of a tensor is believed to be a hard problem. Instead we can find an entry that fulfils the same angle condition and is easily computable. Let $Y \in \mathbb{R}^{n_1\times...\times n_d}$ be a tensor. The j-th hyperslice with respect to index d is the tensor $W^j \in \mathbb{R}^{n_1\times...\times n_{d-1}}$ with entries

$$\left(W^j\right)_{l_1,...,l_{d-1}} := Y_{l_1,...,l_{d-1},j}.$$

Let W^j be the hyperslice with respect to index d that has the largest Frobenius norm. Then

$$\langle W^j, W^j\rangle = \frac{1}{n_d} n_d \langle W^j, W^j\rangle \geq \frac{1}{n_d}\sum_{k=1}^{n_d}\langle W^k, W^k\rangle = \frac{1}{n_d}\langle Y, Y\rangle.$$

We can write the hyperslice as

$$W^j = Y \cdot (0, ..., 0, 1, 0, ..., 0)^T$$

where the j-th entry is 1. The size of the n_d hyperslices can be compared pairwisely to determine the largest. If Y is given in some low-rank format, then the complexity of calculating the norm of a hyperslice is bounded by the cost of calculating the scalar product. If the scalar product is in $O(g(d))$, then finding the desired hyperslice of greatest norm is in $O\left(d \cdot g(d)\right)$. For the TT format for example the calculation of the scalar product is in $O(dnr^3)$ if the ranks are all less than r and the dimensions less than n. This can be seen by contracting the edges from one side of the format to the other. This results in a total cost for finding the direction of $O(d^2nr^3)$ which is better than $O(n^d)$ for searching for the largest entry of the tensor directly. Recursively finding the biggest hyperslice, i.e. finding the greatest entry of the greatest fiber of the greatest slice of the ... of the greatest hyperslice, we can find the desired entry $Y_{i_1,...,i_d}$ which might not be the largest of the tensor, but fulfils the same angle condition

$$\left(Y_{i_1,...,i_d}\right)^2 \geq \omega\langle Y, Y\rangle \text{ with } \omega = \frac{1}{n_1...n_d}.$$

The techniques from [31] and [32] can probably be used to attain slightly better bounds by taking the slice with the biggest projection to any of the slices. The constant ω would improve to

$$\frac{1}{r_1, ..., r_{d-1}}$$

where r_i are the Tucker ranks.

5.4.4 Angle condition for the TT cone

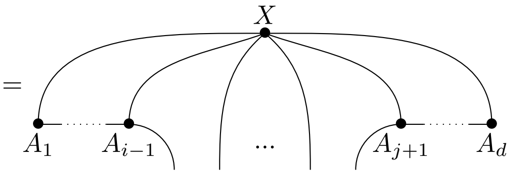
We have decomposed a tangent vector from the TT tangent cone as a sum of orthogonal components in Lemma 4.6. To each component corresponds a space in the way that the component is the orthogonal projection of the tangent vector onto this space. To define these spaces, we will define the orthogonal projection onto them.

Define for $i < j$

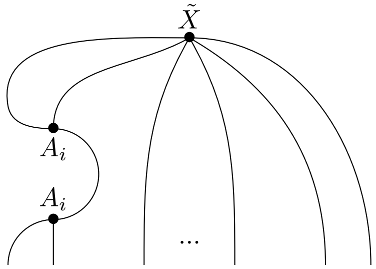
$$P_{ij} : \mathbb{R}^{n_1 \times \ldots \times n_d} \to \mathbb{R}^{n_1 \times \ldots \times n_d} : X \mapsto A_1..A_{i-1}\dot{X}A_{j+1}...A_d$$

in the following way.

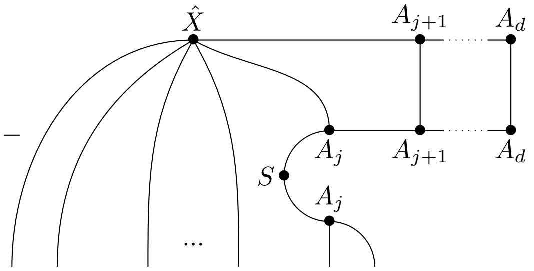
$\tilde{X} :=$



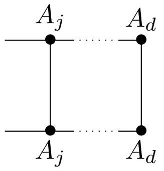
$\hat{X} := \tilde{X} -$



$\dot{X} := \hat{X} -$

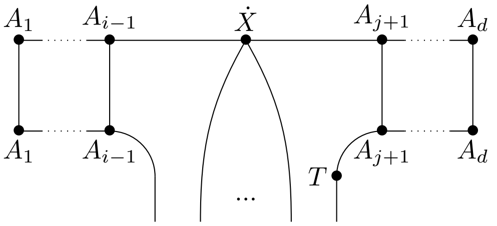


with $S^{-1} :=$

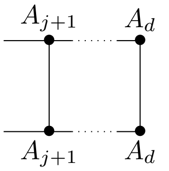


P_{ij} is an orthogonal projection onto its image. Define

$Y :=$



with $T^{-2} =$



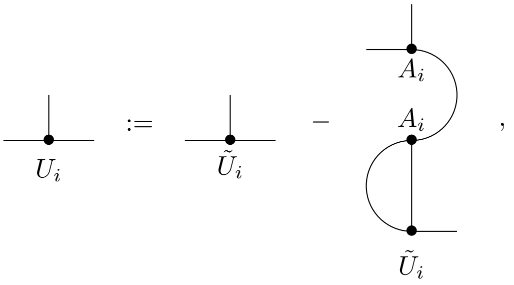
.

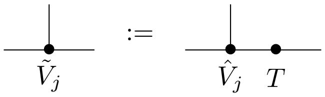
We know from Lemma 5.3 that there are $\tilde{U}_i, Z_{i+1}, ..., Z_{j-1}, \hat{V}_j$ such that

$$\langle Y, \tilde{U}_i Z_{i+1}...Z_{j-1}\hat{V}_j\rangle^2 \geq \frac{1}{n_i...n_j}\langle Y, Y\rangle \tag{18}$$

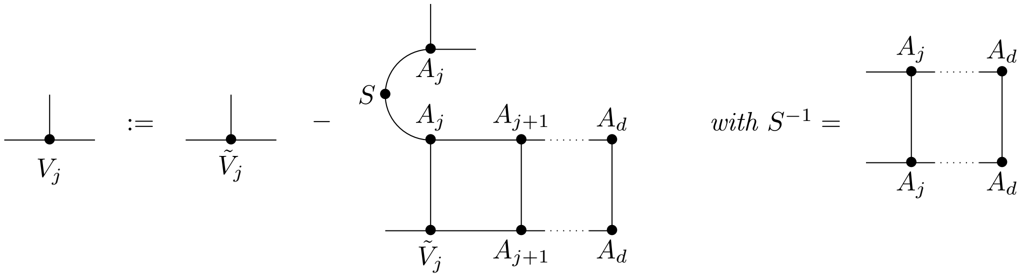
Note that T^{-2} is symmetric positive definite and therefore T exists.

Lemma 5.4. *Defining*





and



the tensor

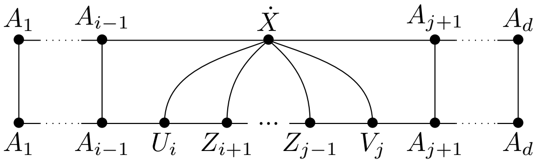
$$W := A_1...A_{i-1}U_i Z_{i+1}...Z_{j-1}V_j A_{j+1}...A_d$$

fulfils the angle condition

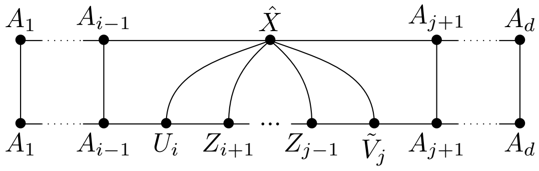
$$\langle P_{ij}X, \frac{W}{||W||}\rangle^2 \geq \frac{1}{n_i...n_j}\langle P_{ij}X, P_{ij}X\rangle.$$

Proof. We calculate

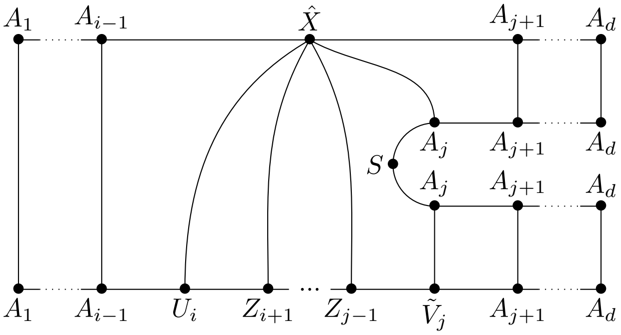
$\langle P_{ij}X, W\rangle =$



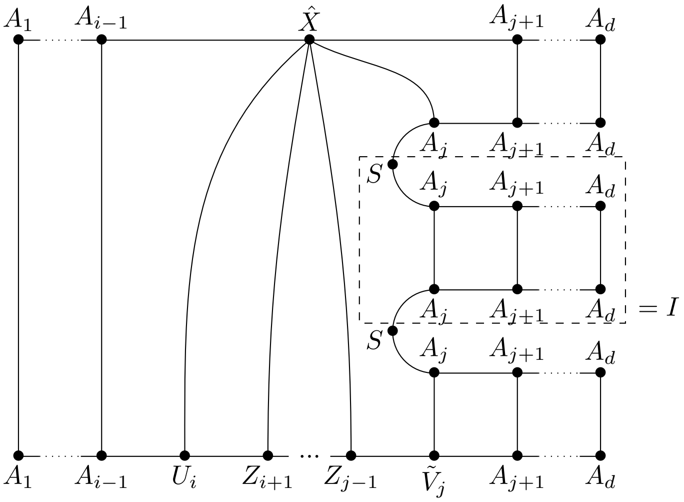
$\overset{\text{Def } \dot{X},\ V_j}{=}$



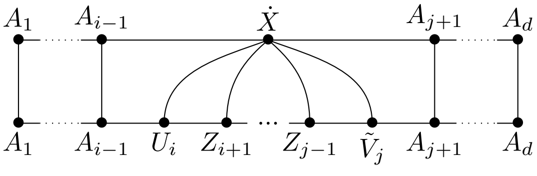
$-\ 2$



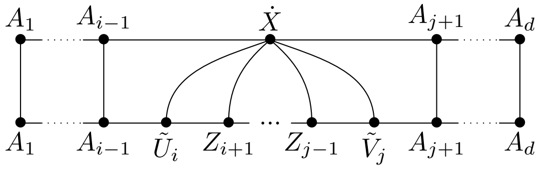
$+$

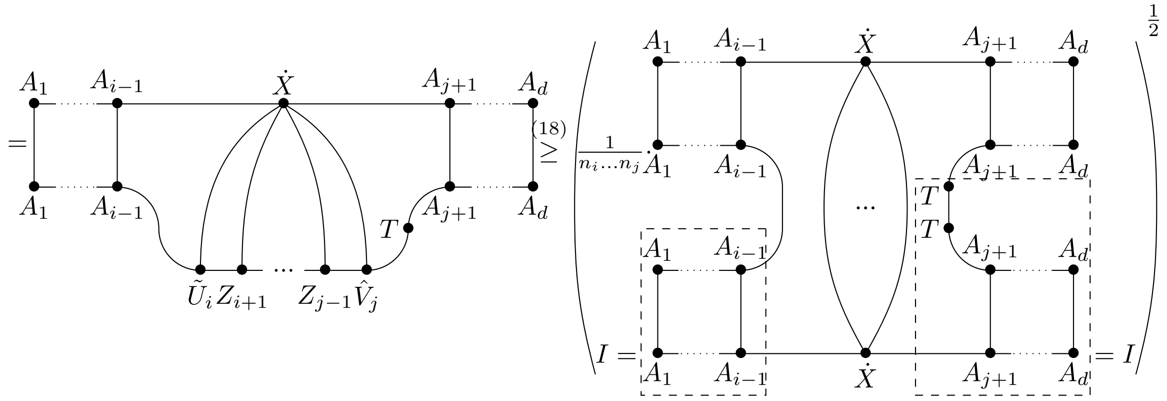


$\overset{\text{Def } \dot{X}}{=}$



$\overset{\text{Def } \dot{X},\, \hat{X},\, U_i}{=}$

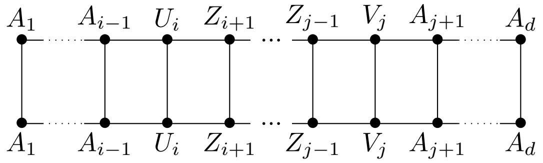




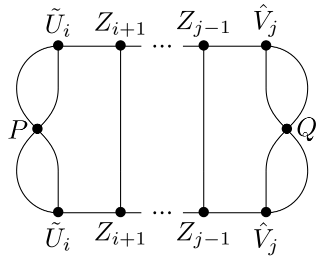
$$= \frac{1}{\sqrt{n_i \dots n_j}} \langle P_{ij} X, P_{ij} X \rangle^{\frac{1}{2}}.$$

It remains to check that the norm of W is bounded by 1. We can do the following calculation.

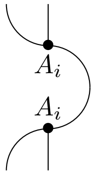
$\langle W, W \rangle =$



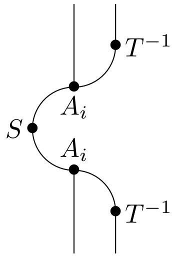
$=$



with $P = I -$

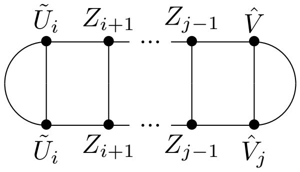


and $Q = I -$

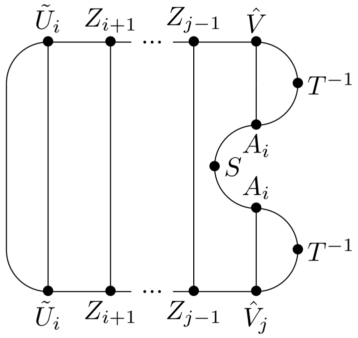


The operator P is an orthogonal projection and in particular $P = PP^T$ such that we can view it as orthogonal projection applied to both factors in the scalar product. An orthogonal projection can only decrease the norm. Q is the identity minus some positive (semi-)definite operator. Thus we can say

$\langle W, W \rangle \leq$



$-$



$$= 1 - ||S^{-1}|| \leq 1.$$

In the last inequality it is of course only of importance, that what we subtract from 1 is positive because it is a square. □

Remark. Any improvement of the constant in Lemma 5.3 would carry over to Lemma 5.4 and thus improve the angle condition for the TT tangent cone.

Lemma 5.5. *(angle condition for TT cone) We can conclude an overall angle condition for the TT tangent cone with a constant*

$$\omega = \frac{1}{\begin{pmatrix} d+1 \\ 2 \end{pmatrix} \sqrt{n_1 ... n_d}}.$$

Proof. The challenge of this proof lies the fact that the U_i, V_i and Z_i each appear in more than one of the orthogonal terms of the decomposition of Equation 11. Therefore we cannot freely add the angle condition fulfilling terms from each of the summands.

Determine

$$(k,l) := \underset{(i,j)}{\operatorname{argmax}} \langle P_{ij} X, W_{ij} \rangle$$

with the notation from Lemma 5.4. Choose all U_i, V_j, Z_k, that do not appear in the decomposition of W_{kl} equal to 0. Let

$$N := \begin{pmatrix} d+1 \\ 2 \end{pmatrix}$$

be the number of possible projections P_{ij}. This includes the linear part of the tangent cone. The binomial coefficient is the number of possible ways to choose two out of $d+1$. The first of the choices shall determine where the U_i is positioned and the second of the choices is one position behind the V_i. If the first and the second choice are next to each other, replace U by X.

Then we have

$$N^2 \langle PX, W_{kl} \rangle^2 = N^2 \langle P_{kl} X, W_{kl} \rangle^2 \geq \left(\sum_{i<j} \langle P_{ij} X, W_{ij} \rangle \right)^2$$

$$\geq \left(\sum_{i<j} \sqrt{\omega} \sqrt{\langle P_{ij} X, P_{ij} X \rangle} \right)^2 \geq \omega \sum_{i<j} \langle P_{ij} X, P_{ij} X \rangle = \omega \langle PX, PX \rangle$$

The first equation is true due to orthogonality. The inequality employ the definition of (k, l), the individual angle conditions of the W_{ij} and the positivity of the terms $\langle P_{ij} X, P_{ij} X \rangle$. For the last equality we define P to be the sum of all P_{ij} - a projection onto a superset of the tangent cone. □

Remark 5.3. The constant might be really bad. It is subject to the curse of dimensionality. However once the order of the tensor d is fixed, convergence of a gradient method can be ensured. This lemma provides, that the projection cannot become arbitrarily small.

6 Riemannian optimization

In this work, we are only considering submanifolds and subvarieties, even though the concepts can and are being applied to abstract manifolds as well. The theory for local optimization on (abstract) manifolds is the topic of [29], including second order algorithms like Newton or trust region methods. Here however we generalize, where possible to tensor varieties as done in [1] for low-rank matrix varieties. Furthermore we focus more on examples and practical application and try to illustrate the theory by examples and figures. The setting starts with a cost function

$$f : \mathbb{R}^N \to \mathbb{R}$$

on some finite-dimensional vector space and a submanifold or subvariety

$$\mathcal{M} \subset \mathbb{R}^N.$$

The ultimate aim is to find a minimum of f on $\mathcal{M}$. Local optimization is only concerned with finding a local minimum or even only a critical point.

6.1 Riemannian gradient method / steepest descent

The basic idea of the gradient method (or more intuitively called the steepest descent method) is to search for a point with smaller cost in the direction, where f decreases fastest. This is the direction v that produces the smallest directional derivative. In the vector space setting with the aim

$$\min_{x \in \mathbb{R}^N} f(x)$$

we begin with some starting point $x_0 \in \mathbb{R}^N$ and determine the direction of smallest directional derivative

$$\operatorname*{argmin}_{v \in \mathbb{R}^N} \frac{\mathrm{d}f}{\mathrm{d}v}.$$

In this setting

$$v = -\nabla f$$

is this direction.

6.1.1 Projected gradient

In the manifold/variety setting the set of possible directions is the tangent cone $T_{x_0}\mathcal{M}$ and we can also define

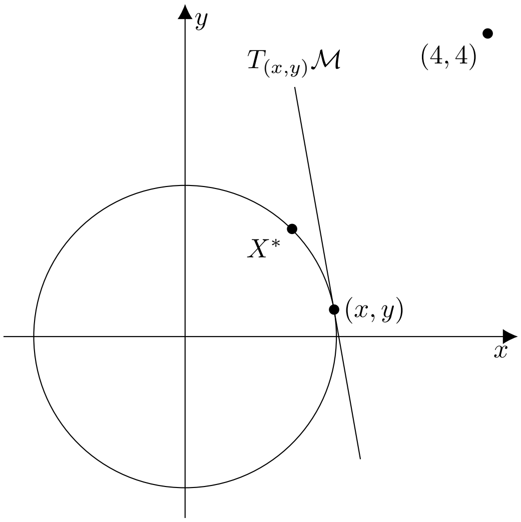
$$\operatorname*{argmin}_{v \in T_{x_0}\mathcal{M},\, ||v||=1} \frac{\mathrm{d}f}{\mathrm{d}v}.$$

This happens to be exactly the best approximation of the gradient of f on the tangent cone (assuming that f is continuously differentiable). We will consider a running example.

Example 6.1. (best approximation on the circle) Consider as manifold or variety the unit circle

$$\mathcal{M} := \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2 : x^2 + y^2 = 1 \right\}$$

in the plane and the task to find the point on this manifold that is closest to the point $\begin{pmatrix} 4 \\ 4 \end{pmatrix}$.



The obvious solution is

$$X^* = \begin{pmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{pmatrix}.$$

Assume that you want to use a gradient method to find X^*. Assume further that you start with some point $X_0 := \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$ on the circle. The tangent cone at X_0 is

$$\mathbb{R}\begin{pmatrix} -y_0 \\ x_0 \end{pmatrix} = \left\{ \begin{pmatrix} u \\ v \end{pmatrix} : \exists \alpha \in \mathbb{R} : \begin{pmatrix} u \\ v \end{pmatrix} = \alpha \begin{pmatrix} -y_0 \\ x_0 \end{pmatrix} \right\}.$$

We want to minimize the square of the distance

$$f : \begin{pmatrix} x \\ y \end{pmatrix} \mapsto \frac{1}{2}\left((x-4)^2 + (y-4)^2\right)$$

to the point $\begin{pmatrix} 4 \\ 4 \end{pmatrix}$. The gradient of f at $\begin{pmatrix} x \\ y \end{pmatrix}$ is

$$\nabla_{(x,y)} f = \begin{pmatrix} x-4 \\ y-4 \end{pmatrix}$$

which is by chance exactly the distance vector

$$\begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} 4 \\ 4 \end{pmatrix}.$$

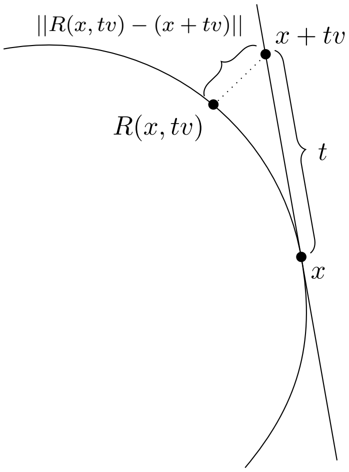
The best approximation of the gradient on the tangent cone is in this case the orthogonal projection

$$\operatorname{grad}_{(x,y)} f = P_{T_{(x,y)}\mathcal{M}} \begin{pmatrix} x-4 \\ y-4 \end{pmatrix} = \begin{pmatrix} -y \\ x \end{pmatrix} \begin{pmatrix} -y & x \end{pmatrix} \begin{pmatrix} x-4 \\ y-4 \end{pmatrix} = 4(y-x) \begin{pmatrix} -y \\ x \end{pmatrix}.$$

In X^* the projected gradient vanishes as it should. The next step is to determine a step size α and "retract" the tangent vector $\alpha \operatorname{grad}_{(x,y)} f$ onto the manifold. This needs to be done in the right order.

6.1.2 Retraction

Figure 25: retraction



Given a point $x \in \mathcal{M}$ on the variety and a direction $v \in T_x\mathcal{M}$ from the tangent cone at x, a retraction defines a point on the variety $\mathcal{M}$ that is "close" to $x + tv$. As we work along the lines of [1] we do not need the *smooth* version of a retraction from [29] but the following:

Definition 6.1. Formally a *retraction* [1] R is a map from the tangent bundle to the variety

$$R : \bigcup_{x \in \mathcal{M}} \{x\} \times T_x\mathcal{M} \to \mathcal{M}$$

that satisfies for every fixed $x \in \mathcal{M}$ and $v \in T_x\mathcal{M}$

$$\lim_{t \to 0} \frac{R(x,tv) - (x+tv)}{t} = 0$$

or equivalently

$$\lim_{t \to 0} \frac{||R(x,tv) - (x+tv)||}{t} = 0.$$

Not usually, but here part of the definition of retraction shall be the condition, that there is a constant M independent of x and v such that

$$||R(x,v) - x|| \leq M||v||.$$

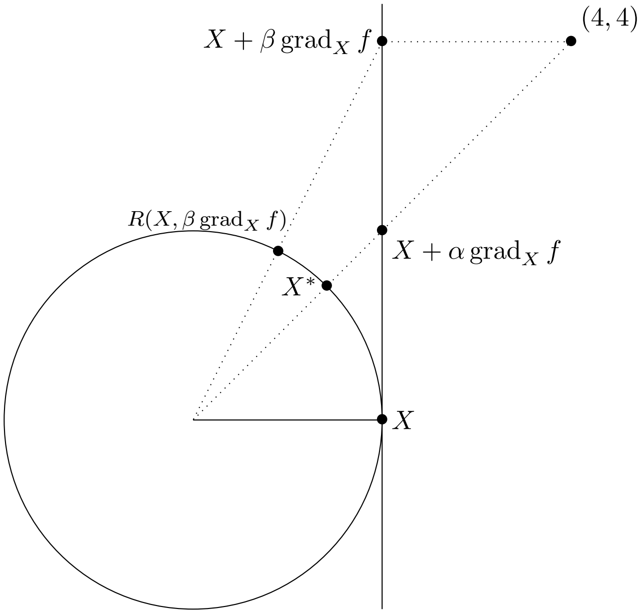
The boundedness in v can be enforced by cutting off the retraction outside of some ball. In Figure 25 we have chosen without loss of generality $||v|| = 1$ and labeled the distances $||(x+tv) - x|| = t$ and $||R(x,tv) - (x+tv)||$. As t converges to 0, the ratio of the two will converge to 0, too. This means that the retraction does not only approach the tangent cone as t approaches 0 but also converges in the direction parallel to the tangent line $x + \mathbb{R}v$.

Example 6.2. (continued) As a retraction one obvious choice for the circle would be the central projection

$$\begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} x \\ y \end{pmatrix} \frac{1}{\sqrt{x^2+y^2}}.$$

Consider the situation in Figure 26

Figure 26: retraction for the circle


(4, 4)
$X + \beta\,\mathrm{grad}_X\, f$
$R(X, \beta\,\mathrm{grad}_X\, f)$
$X + \alpha\,\mathrm{grad}_X\, f$
X^*
X

A first (and wrong) idea for determining the step size is to find the minimum of f on the tangent line that points in the search direction:

$$\beta := \underset{\gamma}{\mathrm{argmin}} f(X) + \gamma\,\mathrm{grad}_X\, f \qquad \text{(wrong step size!)}$$

As you can see in the picture by using the intercept theorem this will not converge to the critical point X^*. For determining a good step size we have to apply the retraction first, i.e. use

$$\alpha := \underset{\gamma}{\mathrm{argmin}}\, R(f(X), \gamma\,\mathrm{grad}_X\, f) \qquad \text{(optimal step size)}$$

In our example step size α would even yield the exact critical point in one step. Finding this optimal step size is often not possible. Instead a sufficient approximation, the Armijo step size can be used.

As a retraction for the TT variety one can use the curve from Lemma 4.4.

Lemma 6.1. *The function*

$$R : \mathcal{X} = \begin{pmatrix} A_1 & U_1 & X_1 \end{pmatrix} \begin{pmatrix} A_2 & U_2 & X_2 \\ 0 & Z_2 & V_2 \\ 0 & 0 & A_2 \end{pmatrix} \dots \begin{pmatrix} A_{d-1} & U_{d-1} & X_{d-1} \\ 0 & Z_{d-1} & V_{d-1} \\ 0 & 0 & A_{d-1} \end{pmatrix} \begin{pmatrix} X_d \\ V_d \\ A_d \end{pmatrix}$$

$$\mapsto \begin{pmatrix} A_1+X_1 & U_1 \end{pmatrix} \begin{pmatrix} A_2+X_2 & U_2 \\ V_2 & Z_2 \end{pmatrix} ... \begin{pmatrix} A_{d-1}+X_{d-1} & U_{d-1} \\ V_{d-1} & Z_{d-1} \end{pmatrix} \begin{pmatrix} A_d+X_d \\ V_d \end{pmatrix}$$

defines a retraction onto the TT variety in the sense of the definition above using the notation from Chapter 4.

Proof. The image under R of the tangent vector multiplied by t, $R(t\mathcal{X})$ is

$$\begin{pmatrix} A_1+tX_1 & U_1 \end{pmatrix} \begin{pmatrix} A_2+tX_2 & U_2 \\ tV_2 & Z_2 \end{pmatrix} ... \begin{pmatrix} A_{d-1}+tX_{d-1} & U_{d-1} \\ tV_{d-1} & Z_{d-1} \end{pmatrix} \begin{pmatrix} A_d+tX_d \\ tV_d \end{pmatrix}.$$

We calculate

$$\lim_{t\searrow 0} \frac{R(x,tv)-x-tv}{t} = \lim_{t\searrow 0} \frac{t^2(\text{polynomial in } t)}{t} = \lim_{t\searrow 0} t(\text{polynomial in } t) = 0.$$

The retraction is well-defined. When the A_i all have full ranks, the decomposition $A = A_1...A_d$ is unique up to a Lie group action. This means, that for any other decomposition $A = \tilde{A_1}...\tilde{A_d}$ we have $\tilde{A_1} = A_1G_1$, $\tilde{A_i} = G_{i-1}^{-1}A_iG_i$ for $i = 2,...,d-1$ and $\tilde{A_d} = G_{d-1}^{-1}A_d$ with invertible G_i. See [19] for details. As we have discussed in Section 4.2 the tangent vectors are also unique up to a Lie group action. The proposed retraction R is independent of those two Lie group actions. When using a different decomposition for both A and the tangent vector the image under R will be

$$\begin{pmatrix} (A_1+X_1)G_1 & U_1g_1 \end{pmatrix} \begin{pmatrix} G_1^{-1}(A_2+X_2)G_2 & G_1^{-1}U_2g_2 \\ g_1^{-1}V_2G_2 & g_1^{-1}Z_2g_2 \end{pmatrix} ... \begin{pmatrix} G_{d-1}^{-1}(A_d+X_d) \\ g_{d-1}^{-1}V_d \end{pmatrix}$$

where we can cancel all G_i and g_i to see the equality. That the same G_i are applied to the A_i and the X_i can be seen by plugging the definition of $\tilde{A_i}$ into the definition of X_i along Equation 12. □

This retraction is particularly easy to calculate if the tangent vectors are given in the described format.

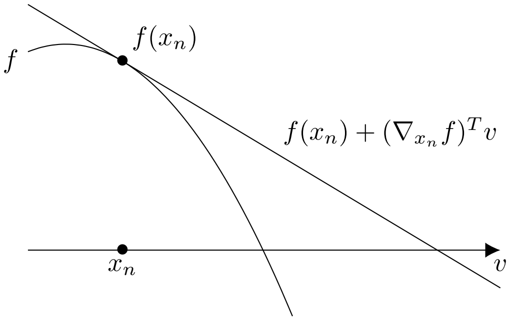
6.1.3 Armijo point

As step size the so-called Armijo point has proven to be a good choice - good enough to guarantee convergence and easy enough to calculate. Consider some continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ and consider that for some iterate x_n you have evaluated $f(x_n)$. The goal is of course to find x_{n+1} such that $f(x_{n+1})$ is smaller than $f(x_n)$ and not just smaller but more-or-less as small as possible along the current search direction. Assume we know the derivative of f. Then we know the directional derivative

$$(\nabla_{x_n} f)^T v$$

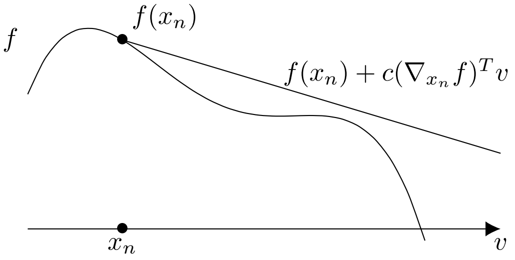
in the search direction v. Then there are several possible cases. The first is, that in the search direction f stays below its linearization

$$f(x_n + tv) \leq f(x_n) + (\nabla_{x_n} f)^T v.$$



In this case there is no minimum. Also if f stays below a flatter version of the linearization

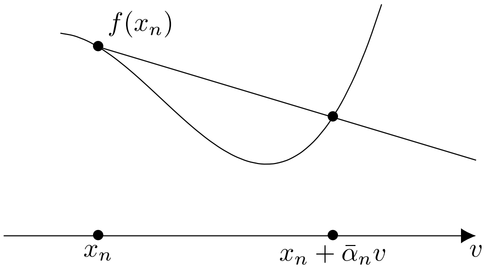
$$f(x_n + tv) \le f(x_n) + c\,(\nabla_{x_n} f)^T\, v \text{ for some } c \in (0,1)$$



there would be no minimum. But for every f that has a minimum on $\mathbb{R}^n$ there is at least one point in the search direction, where the graph of f crosses the "flattened" linearization

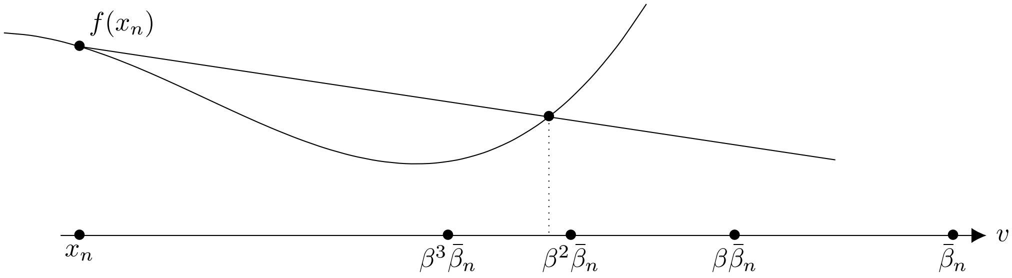
$$f(x_n) + c\,(\nabla_{x_n} f)^T\, v.$$

The crossing that is closest to x_n in the direction $-\nabla_{x_n} f$ is called $x_n - \bar{\alpha}_n \nabla_{x_n} f$ in [1].



As we cannot exactly determine $\bar{\alpha}_n$ we content ourselves with a step size that is only a factor $\beta \in (0,1)$ off $\bar{\alpha}_n$. β can be fixed in advance. The step size can then be found using backtracking.

Figure 27: backtracking for finding Armijo step size



In Figure 27 we have labeled the points on the horizontal axis by the step sizes that produce them. Starting with step size $\bar{\beta}_n$ backtracking yields the Armijo step size $\alpha_n = \beta^3\bar{\beta}_n$. In general on varieties, as used in [1] we define

$$\bar{\alpha}_n := \min\left\{\alpha > 0 : f\left(R(x_n, \alpha v)\right) = f(x_n) + c\alpha \left(\nabla_{x_n} f\right)^T v\right\}$$

and

$$\alpha_n := \max\left\{\beta^m\bar{\beta}_n : m \in \mathbb{N}\cup\{0\},\ f\left(R(x_n, \beta^m\bar{\beta}v)\right) \leq f(x_n) + c\beta^m\bar{\beta}\left(\nabla_{x_n} f\right)^T v\right\}$$

We then have the properties

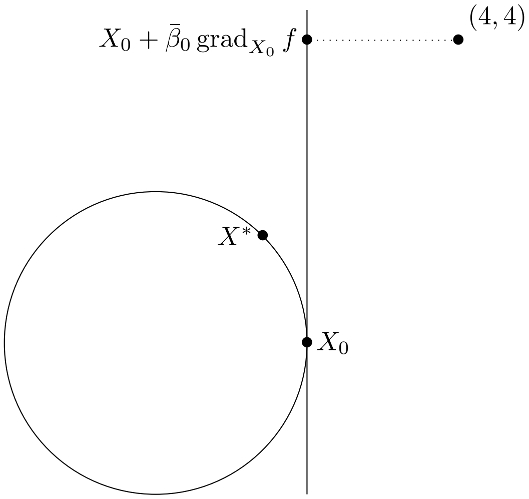
$$f(x_n + \alpha_n v) \leq f(x_n) + c\alpha_n \left(\nabla_{x_n} f\right)^T v$$

and by definition

$$\alpha_n \geq \beta\bar{\alpha}_n.$$

We can now conclude by completing our running example.

Example 6.3. (continued)



We go in detail through one whole iteration step. Let us start with the point

$$X_0 := \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

on the circle. The tangent space at this point is then the line

$$T_{X_0}\mathcal{M} = \mathbb{R}\begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Choose the initial guess for the step size $\bar{\beta}_i$ as the length of the projected gradient

$$\bar{\beta}_0 := ||\operatorname{grad}_{X_0} f|| = 4.$$

Choose

$$\beta = c = \frac{1}{2}.$$

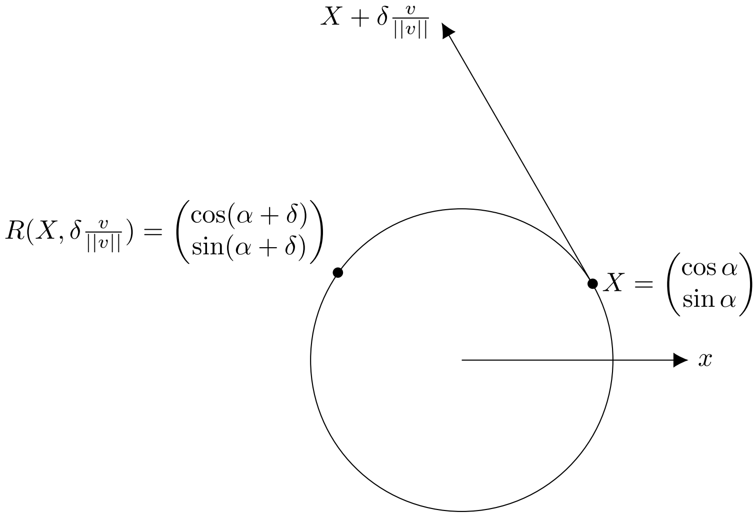
Now we need to do back-tracking. To make this more interesting we will choose as retraction not the central projection but one that wraps the tangent vector around the circle. So if

$$X = \begin{pmatrix} \cos\delta \\ \sin\delta \end{pmatrix}$$

then the retraction shall be

$$R\left(X, \gamma\frac{v}{||v||}\right) = \begin{pmatrix} \cos(\delta+\gamma) \\ \sin(\delta+\gamma) \end{pmatrix}$$

if v points anticlockwise.


$X + \delta\frac{v}{||v||}$
$R(X, \delta\frac{v}{||v||}) = \begin{pmatrix} \cos(\alpha+\delta) \\ \sin(\alpha+\delta) \end{pmatrix}$
$X = \begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix}$
x

Then

$$R\left(X_0, 4\begin{pmatrix} 0 \\ 1 \end{pmatrix}\right) \approx \begin{pmatrix} -0.65 \\ -0.76 \end{pmatrix}$$

which is further from $(4,4)$ than X_0. Trying to bisect the step size

$$R\left(X_0, 2\begin{pmatrix}0\\1\end{pmatrix}\right) \approx \begin{pmatrix}-0.42\\0.91\end{pmatrix}$$

still yields a point further from $(4,4)$. Bisecting a second time yields the point

$$R\left(X_0, 1\begin{pmatrix}0\\1\end{pmatrix}\right) \approx \begin{pmatrix}0.54\\0.84\end{pmatrix}$$

which is closer to $(4,4)$ than the previous iterate. But is it an Armijo point? Recall that our example-f was defined as

$$f : \begin{pmatrix}x\\y\end{pmatrix} \mapsto \frac{1}{2}\left((x-4)^2 + (y-4)^2\right)$$

and with $c = \frac{1}{2}$ and $(\nabla_{X_0} f)^T v = 4$ the Armijo step size has to fulfil

$$f(X_1) \leq f(X_0) - \frac{1}{2}\alpha_0 4 = \frac{25}{2} - 2\alpha_0.$$

In our example this is not fulfilled by the step size 1. Thus we need to halve once more to find that

$$f\left(R\left(X_0, \frac{1}{2}\begin{pmatrix}0\\1\end{pmatrix}\right)\right) = f\begin{pmatrix}0.87...\\0.47...\end{pmatrix} = 11.07... \leq 11.5 = \frac{25}{2} - 1 = f\left(X_0\right) - c\frac{1}{2}\left(\nabla_{X_0} f\right)^T \begin{pmatrix}0\\1\end{pmatrix}$$

and

$$X_1 = \begin{pmatrix}0.87...\\0.47...\end{pmatrix}$$

is thus the next iterate.

6.1.4 Example: rank 1 approximation of a large matrix

We can also formulate the Riemannian gradient method for the best approximation problem in the variety of rank-1 matrices. The variety would then be

$$\mathcal{M} := \mathcal{M}_{\leq 1}^{n\times n} := \left\{A \in \mathbb{R}^{n\times n} : \operatorname{rank}(A) \leq 1\right\}.$$

We want to approximate an arbitrary matrix $B \in \mathbb{R}^{n\times n}$ and do this by minimizing the functional

$$f(A) := \frac{1}{2}||A - B||_F^2$$

on $\mathcal{M}$. We can write $A \in \mathcal{M}$ as a tensor product of two vectors $A = ab$ with $a \in \mathbb{R}^{n\times 1}$ and $b \in \mathbb{R}^{1\times n}$. Where A has rank exactly 1, we can write the tangent space at A as

$$T_A\mathcal{M} = \left\{X \in \mathbb{R}^{n\times n} : \exists v \in \mathbb{R}^{n\times 1}, w \in \mathbb{R}^{1\times n} : X = vb + aw\right\}.$$

The orthogonal projection of some matrix D onto this tangent space can be written as

$$P_{T_A\mathcal{M}}\left(D\right) = v\left(v^T v\right)^{-1} v^T D + \left(\left(I - v\left(v^T v\right)^{-1} v^T\right) D\right) w^T \left(ww^T\right)^{-1} w$$

or

$$P_{T_A\mathcal{M}}\left(D\right) = Dw^T \left(ww^T\right)^{-1} w + v\left(v^T v\right)^{-1} v^T \left(D\left(I - w^T\left(ww^T\right)^{-1} w\right)\right).$$

We implemented this in Octave/Matlab. The function `init` creates a random rank-1 or rank-2 matrix B and a starting iterate X_0.

```
function [B,v,w]=init(n,m)
   B=rand(n,1)*rand(1,m);%+rand(n,1)*rand(1,m);
   v=zeros(n,1);
   w=zeros(1,m);
   v(1,1)=1;
   w(1,1)=1;
endfunction
```

The function step calculates the next iterate.

```
function [B,v,w]=step(B, v, w)

   %calculation of gradient of distance function
   dist=B-v*w;

   %projection onto the tangent space
   y=(v’*v)\v’*dist;
   x=((w*w’)’\w*(dist-v*y)’)’;

   %searching for armijo point
   while norm((v+x)*(w+y)-B) > norm(v*w-B)-0.5*norm(x*w+v*y)*norm(x*w+v*y)
      x=x/2;
      y=y/2;
   endwhile

   %retraction
   v=v+x;
   w=w+y;
endfunction
```

We can plot the convergence behaviour with the function test:

```
function test(m)
   [B,a,b]=init(10,10);
   [U,S,V]=svd(A);
   y=zeros(m,1);
   x=zeros(m,1);
   for i=1:m
      [B,a,b]=step(B,a,b);
      x(i)=i;
      y(i)=norm(a*b-U(:,1:1)*S(1:1,1:1)*V(:,1:1)’);
   endfor
   semilogy(x,y)
endfunction
```

The asymptotic complexity of a single iteration lies in $O(n^2)$. The convergence appears to be quadratic, if B has rank 1 but linear and sometimes very slow if B has rank 2. Possible convergence behaviour can be seen in Figures 28 and 29.

Figure 28: rank-1 approximation of a rank-1 matrix, error against number of iterations

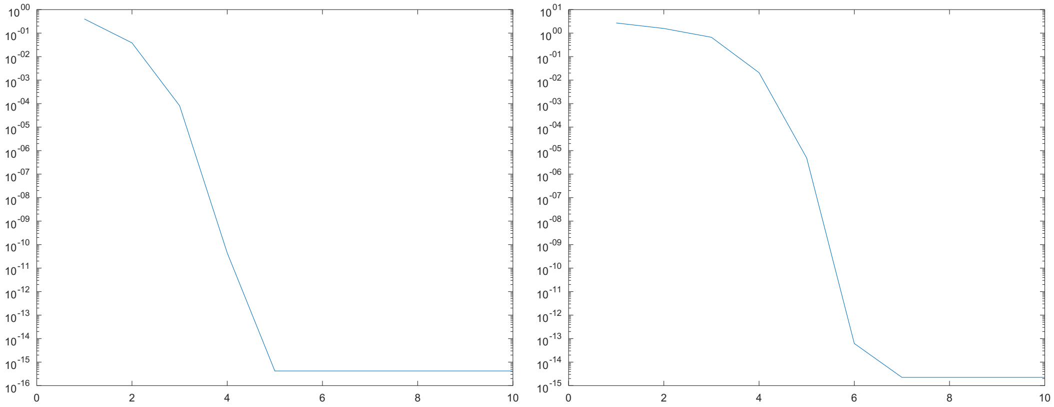
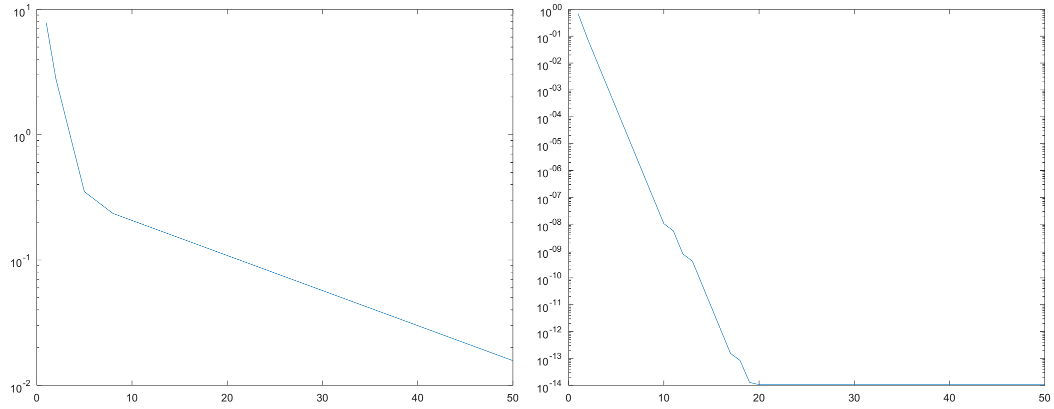


Figure 29: rank-1 approximation of a rank-2 matrix, error against number of iterations



In the case that we want to approximate by a rank-1 matrix and the matrix we want to approximate is already given in a low-rank format, we can do some optimization. Suppose we want to approximate $B = ab$ where $a \in \mathbb{R}^{n\times 2}$ and $b \in \mathbb{R}^{2\times n}$ by a rank-1 matrix $A = vw$ with $v \in \mathbb{R}^{n\times 1}$ and $w \in \mathbb{R}^{1\times n}$. The distance between A and B can be written as

$$\begin{pmatrix} a & v \end{pmatrix} \begin{pmatrix} b \\ -w \end{pmatrix}.$$

The projection of this distance (which is the gradient of f) onto the tangent space at A can be written as

$$P_{T_A\mathcal{M}}\left(B-A\right) = v\left(\left(\frac{v^Ta}{v^Tv}\right)b - w\right) + \frac{1}{ww^T}\left(\left(a - v\frac{v^Ta}{v^Tv}\right)\left(bw^T\right)\right)w$$

which has better asymptotic complexity than the general formula by a factor of n. Modifying the rest of the code as well yields a program, that runs in $O(n)$ for every iteration. This is the fundamental strength of optimizing on low-rank tensor varieties.

The Octave/Matlab code becomes

```
function [a,b,v,w]=initOpt(n,m)
   a=rand(n,1);
   b=rand(1,m);
   v=zeros(n,1);
   w=zeros(1,m);
   v(1,1)=1;
   w(1,1)=1;
endfunction
```

and

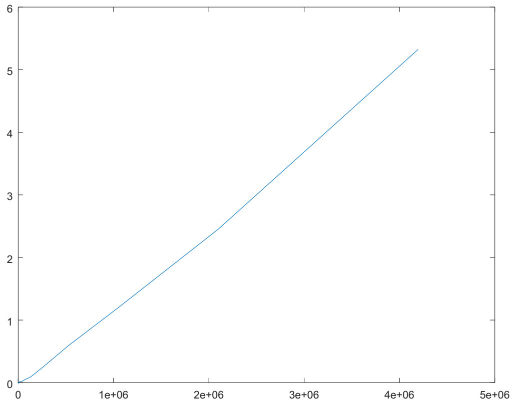
```
function [a,b,v,w]=stepOpt(a, b, v, w)

   %projection onto the tangent space
   y=((v’*a)/(v’*v))*b-w;
   x=1/(w*w’)*(a*(b*w’)-v*(w*w’+y*w’));

   %searching for armijo point
   while sqrt(((v+x)’*(v+x))*((w+y)*(w+y)’)+trace((a’*a)*(b*b’))-2*trace((a’*(v+x))*((w+y)*b’)))
#> sqrt(((v)’*(v))*((w)*(w)’)+trace((a’*a)*(b*b’))-2*trace((a’*(v))*((w)*b’)))
#-0.5*((x’*x)*(w*w’)+(v’*v)*(y*y’)-2*(v’*x)*(w*y’))
      x=x/2;
      y=y/2;
   endwhile
   %retraction
   v=v+x;
   w=w+y;
endfunction
```

For a 1000000 × 1000000 matrix one iteration takes about 2 seconds on a 2-core 3 GHz laptop.

Figure 30: seconds against n, time to evaluate one iteration for $n \times n$ matrix



This graph has been produced using the function

```
function testPerf(n)
   %3 lines for tricking frequency scaling into setting max cpu frequency
   [a,b,v,w]=initOpt(1000000,1000000);
   [a,b,v,w]=stepOpt(a,b,v,w);
   [a,b,v,w]=stepOpt(a,b,v,w);

   for i=1:n
       [a,b,v,w]=stepOpt(a,b,v,w);
       x(i)=2^i;
       [a,b,v,w]=initOpt(2^i,2^i);
       tic;
       [a,b,v,w]=stepOpt(a,b,v,w);
       y(i)=toc;
   endfor
   plot(x,y)
```

6.1.5 Łojasiewicz inequality

We would like to show the convergence of an optimization algorithm for a very wide range of functions. In [33] it is shown, how for functions satisfying Łojasiewicz' inequality a converging algorithm can be constructed. The setting of [33] are functions $\mathbb{R}^n \to \mathbb{R}$. Łojasiewicz' inequality is satisfied for all analytic functions. The type of convergence is the following: If the algorithms possesses a cluster point, it is the limit. [1] combines [33] with the techniques of [29] to generalize the convergence result to matrix varieties. For a list of work about convergence analysis via Łojasiewicz' inequality, see [1] from where we also copy the following definition.

Definition 6.2. Let $\alpha(y)$ be the length of the best approximation of the gradient at y onto the tangent cone. We say that $x \in \mathcal{M}$ satisfies a Łojasiewicz inequality for the projected gradient, if there exists $\delta > 0$, $\Lambda > 0$, and $\theta \in (0, \frac{1}{2}]$ such that for all $y \in \mathcal{M}$ with $||y - x|| < \delta$

$$|f(y) - f(x)|^{1-\theta} \leq \Lambda\alpha(y) \tag{19}$$

holds.

In this chapter we want to prove, that the Łojasiewicz gradient inequality holds on the set of TT tensors. In order to do this we cite the following Proposition.

Proposition 6.1. *(2.2 in [1]) Let f be real-analytic and defined on the open subset $D \subseteq \mathbb{R}^N$. Let further $\mathcal{M} \subseteq D$ be contained in the domain of f. Assume there exists an open set of parameters $\mathcal{N} \subseteq \mathbb{R}^M$ and a preimage $t_0 \in \mathcal{N}$ of $x \in \mathcal{M}$ of the real-analytic map $\tau : \mathcal{N} \to \mathbb{R}^N$ such that*

1. *$\tau(\mathcal{N}) \subseteq \mathcal{M}$ and*
2. *the image of every open neighborhood of t_0 under τ contains an open neighborhood of x in $\mathcal{M}$ (in the induced topology)*

Then the Łojasiewicz gradient inequality holds at x.

Because we are in metric spaces, Condition 2 can be rephrased as

$$\forall t_0 \forall \delta \exists \varepsilon : (y \in B_\varepsilon(x) \cap \mathcal{M} \implies (\exists s \in B_\delta(t_0) : y = \tau(s)))\,.$$

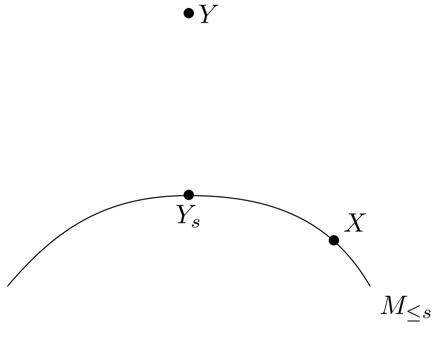
Using the proposition, we illustrate the proof of the following result about the matrix case.

Theorem 6.1. *(proven in [1, Theorem 3.8]) Let the real-analytic $f : D \to \mathbb{R}$ be defined on the open superset D of the set of matrices of bounded rank $\mathcal{M}^{n \times m}_{\leq r}$. Then the Łojasiewicz inequality holds at any point of $\mathcal{M}^{n \times m}_{\leq r}$.*

Proof. Let $X \in \mathcal{M}^{n_1 \times n_2}_{\leq k+s}$ be a matrix of rank s. Then there are full rank matrices U_0, V_0 such that $X = U_0 V_0^T$. Now define the map

$$\tau : \mathbb{R}^{n_1 \times s} \times \mathbb{R}^{n_1 \times k} \times \mathbb{R}^{n_2 \times s} \times \mathbb{R}^{n_2 \times k} \to \mathbb{R}^{n_1 \times n_2} :$$
$$\left(U, \tilde{U}, V, \tilde{V}\right) \mapsto UV^T + \tilde{U}\tilde{V}^T.$$

The image of τ is contained in $\mathcal{M}^{n_1 \times n_2}_{\leq k+s}$. Thus property 1 of Proposition 6.1 holds. We need to show property 2. It suffices to show that for every ε there is a δ such that every $Y \in B_\delta(X) \cap \mathcal{M}^{n_1 \times n_2}_{\leq k+s}$ has a preimage under τ contained in $B_\varepsilon\left((U_0, 0, V_0, 0)\right)$. Let $Y \in B_\delta(X)$, i.e. $||Y - X||_F < \delta$. Let Y_s be the best approximation of Y in $\mathcal{M}_{\leq s}$ in the Frobenius norm and $Y_k = Y - Y_s$.



Then, because Y_s is the best approximation, we have

$$||Y_k||_F = ||Y - Y_s||_F \leq ||Y - X||_F < \delta. \tag{20}$$

If $\sigma_1 > ... > \sigma_{s+k}$ are the singular values of Y, then $\sigma_{s+1} > ... > \sigma_{s+k}$ are the singular values of Y_k and inequality 20 implies $\sigma_{s+1}^2 + ... + \sigma_{s+k}^2 < \delta^2$. In particular every singular value of Y_k is smaller than δ. If $U_k \Sigma V_k^T$ is the singular value decomposition of Y_k, then $\hat{U}_k := U_k\sqrt{\Sigma}$ and $\hat{V}_k := V_k\sqrt{\Sigma}$ have distance at most $\sqrt{k\delta^2} = \sqrt{k}\delta$ from zero. Furthermore by the triangle inequality and the best approximation we have

$$||Y_s - X||_F \leq ||Y_s - Y||_F + ||Y - X||_F \leq 2||Y - X||_F \leq 2\delta.$$

Thus Y_s is contained in a small neighborhood of X on the smooth manifold $\mathcal{M}_{=s}^{n_1 \times n_2}$. Using [Ddé. 16.7.5] (which is a direct consequence of the famous "théorème du rang" 10.3.1) we can decompose

$$Y_s = (U + U_s)(V + V_s)^T$$

where U_s and V_s can be made arbitrarily small by decreasing δ. And we conclude that

$$\tau\left(U + U_s, U_k\sqrt{\Sigma}, V + V_s, V_k\sqrt{\Sigma}\right) = Y$$

and for δ small enough, we can force

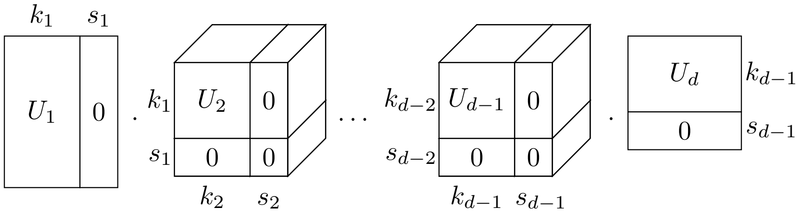
$$||\left(U + U_s, U_k\sqrt{\Sigma}, V + V_s, V_k\sqrt{\Sigma}\right) - (U, 0, V, 0)\,||_F \leq \varepsilon.$$

□

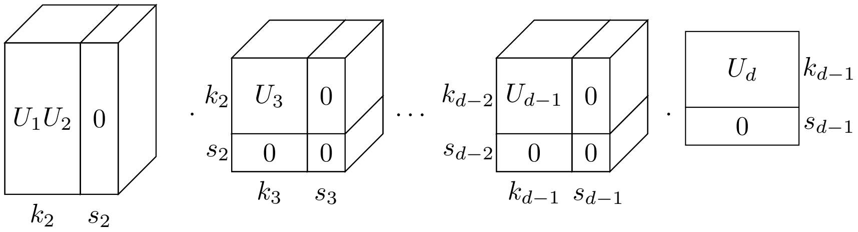
Property 1 of Proposition 6.1 generalizes in the intuitive way to the TT format. It remains to check Property 2 for the TT format and do this by enhancing the proof of the matrix case using induction.

Lemma 6.2. *(generalization of [1, Theorem 3.8]) Let the real-analytic $f : D \to \mathbb{R}$ be defined on the open superset D of the set of TT tensors of bounded rank $\mathcal{M}_{\leq(r_1,...,r_{d-1})}^{n_1\times...\times n_d}$. Then the Łojasiewicz inequality holds at any point of $\mathcal{M}_{\leq(r_1,...,r_{d-1})}^{n_1\times...\times n_d}$.*

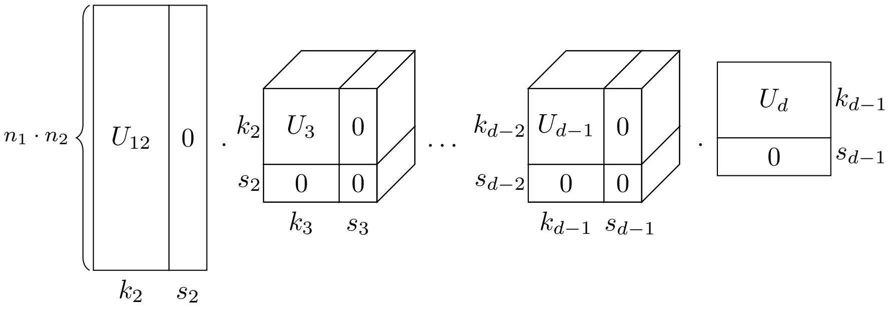
Proof. Assume that Condition 2 has been proven for TT tensors of order $d-1$. Let $X = U_1...U_d$ be a full rank TT decomposition of $X \in \mathcal{M}_{\leq(k_1,...,k_{d-1})}^{n_1\times...\times n_d}$. Setting $s_i := r_i - k_i$, visualizing τ amounts to writing X as



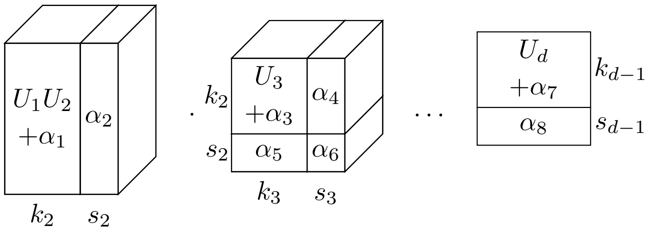
which is equal to



by evaluating the first multiplication. Now joining the first two indices n_1 and n_2 yields

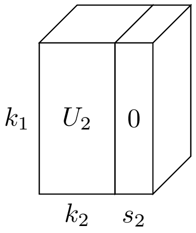


where U_{12} is the matricization of U_1U_2. Applying the inductive hypothesis, for every $\varepsilon > 0$ there is a $\delta > 0$ such that every $Y \in B_\delta(X) \cap \mathcal{M}^{n_1 n_2 \times n_3 \times \ldots \times n_d}_{\leq (r_2, \ldots, r_{d-1})}$ (and thus every $Y \in B_\delta(X) \cap \mathcal{M}^{n_1 \times n_2 \times n_3 \times \ldots \times n_d}_{\leq (r_1, r_2, \ldots, r_{d-1})}$ because it is a subset) can be written as

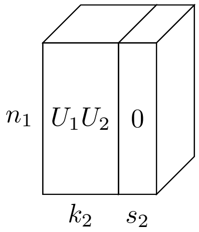


with the α_i each smaller than ε.

As U_2 has full rank k_1 in the first index, the tensor



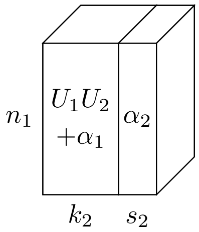
has full rank in the first index as well (horizontal slices stay linearly independent) and its matricization $\dot{U}_2 \in \mathbb{R}^{k_1 \times ((k_2+s_2)\cdot n_2)}$ has full rank k_1. Let $\dot{U}_{12}$ be the matricization of



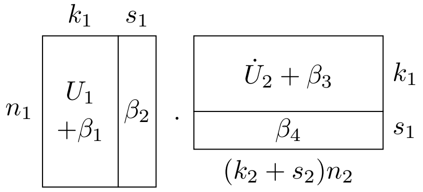
in $\mathbb{R}^{n_1\times((k_2+s_2)\cdot n_2)}$. We have

$$\dot{U}_{12} = U_1 \cdot \dot{U}_2.$$

The matricization of



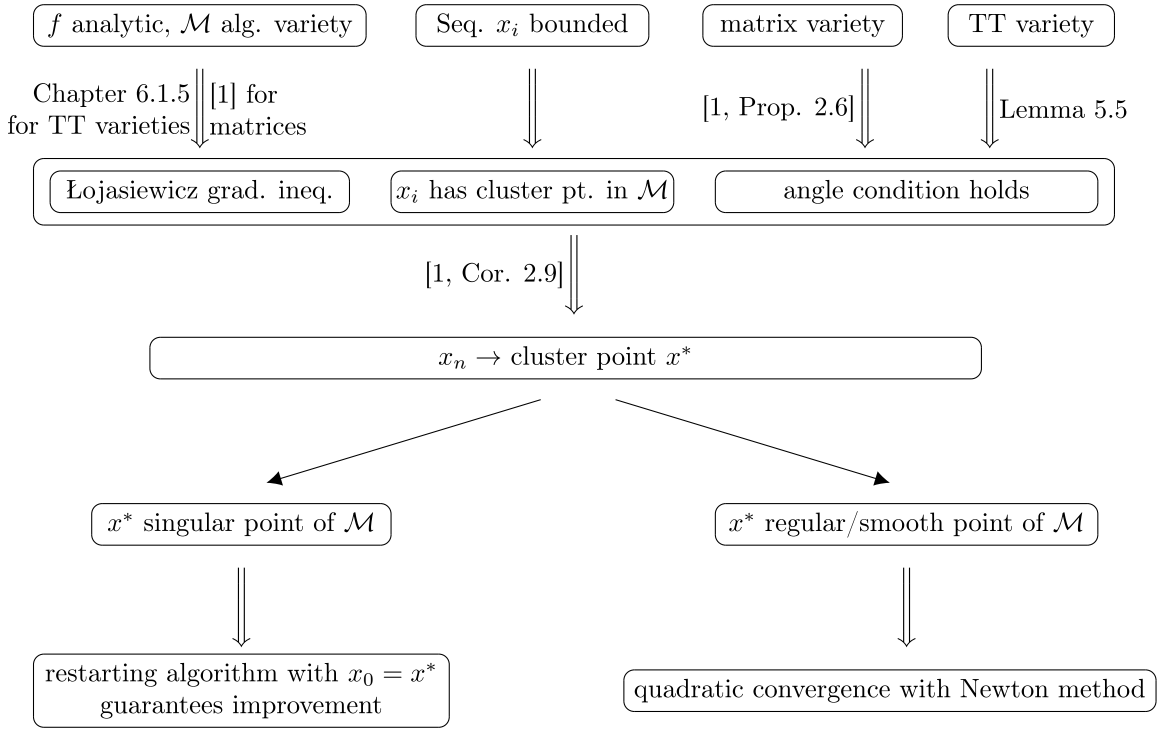
in $\mathbb{R}^{n_1\times((k_2+s_2)\cdot n_2)}$ has rank $\leq r_1$ in the first index (because Y has rank $\leq r_1$ in the first index). Thus it has distance $\sqrt{||\alpha_1||_F^2 + ||\alpha_2||_F^2} \leq \sqrt{2}\varepsilon$ from $\dot{U}_{12}$ and using the matrix version of this lemma, we can write it as



again with arbitrarily small β_i. Tensorization of the second factor yields the desired result. □

6.1.6 Global convergence analysis

We now have all the necessary ingredients to generalize the convergence analysis from [1] to TT tensor varieties. The following diagram shows the individual steps in the argument.


f analytic, $\mathcal{M}$ alg. variety
Seq. x_i bounded
matrix variety
TT variety
Chapter 6.1.5 for TT varieties
[1] for matrices
[1, Prop. 2.6]
Lemma 5.5
Łojasiewicz grad. ineq.
x_i has cluster pt. in $\mathcal{M}$
angle condition holds
[1, Cor. 2.9]
$x_n \to$ cluster point x^*
x^* singular point of $\mathcal{M}$
x^* regular/smooth point of $\mathcal{M}$
restarting algorithm with $x_0 = x^*$ guarantees improvement
quadratic convergence with Newton method

The central part in the argument is Corollary 2.9 from [1]. It states that, if f satisfies a Łojasiewicz gradient inequality and the line search algorithm satisfies an angle condition, then it converges to a cluster point of the sequence of iterates, if one exists. We cite:

Lemma 6.3. *(Corollary 2.9 in [1]) Assume that f is continuously differentiable and bounded below. Assume that*

$$x_{n+1} = R(x_n, \alpha_n v_n)$$

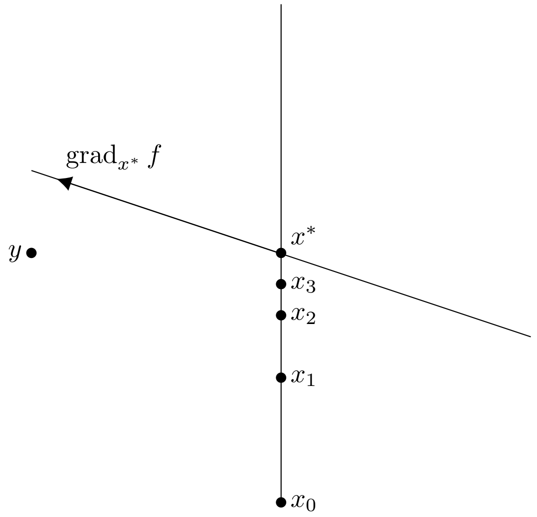
where v_n satisfies an angle condition, α_n is an Armijo step size and R a retraction in the sense of Definition 6.1. Then, if a cluster point x^* of $(x_i)_{i\in\mathbb{N}}$ exists and satisfies the Łojasiewicz inequality, it is the limit

$$x^* = \lim_{i\to\infty} x_i.$$

So, why is this result not stronger? Why can we not prove that the sequence converges to a local minimum or at least to a critical point, where the projected gradient vanishes? Consider the following counter example.

It depicts an algebraic variety (two crossing lines) and the problem of approximating the point y by a point on the variety. x^* is the cluster point and limit of the sequence $(x_i)_{i\in\mathbb{N}}$ that could have been produced by some gradient descent algorithm as in the lemma. The length of the projected gradients of the iterates converges to 0. However the projected gradient in the limit point x^* does not vanish! Fortunately this can only happen, when the limit point is a singular point of $\mathcal{M}$. And in this case we could restart the algorithm with the starting point $x_0 = x^*$ set to the old singular limit point. This would give us either an improvement - i.e. an iterate with decreased value of f -

Figure 31: possible convergence to a singular but non-critical point



or certainty that we have found a critical point in $\mathbb{R}^N$. If the limit is a smooth point, we can either use the convergence rates from [1] or use the Newton method to gain a quadratic convergence rate.

6.2 Riemannian Newton method

The *Riemannian Newton method* is the generalization of Newtons method to Riemannian manifolds. The aim of this chapter is to illustrate this method, described for example in [29], with a running example.

6.2.1 An Example: Finding the best appxoximation on the circle

Example 6.4. Let

$$\mathcal{M} := \left\{ \begin{pmatrix} x \\ y \end{pmatrix} : x^2 + y^2 = 1 \right\}$$

be the unit circle and

$$f : \begin{pmatrix} x \\ y \end{pmatrix} \mapsto \frac{1}{2}(x-4)^2 + \frac{1}{2}(y-4)^2$$

be the function that we aim to minimize.

Define the projected gradient as in the previous chapter as

$$\mathrm{grad}_{\begin{pmatrix} x \\ y \end{pmatrix}} f := P_{T_{\begin{pmatrix} x \\ y \end{pmatrix}}\mathcal{M}} \nabla f.$$

In the example this amounts to

$$\operatorname{grad}_{\begin{pmatrix}x\\y\end{pmatrix}} f = 4(y-x)\begin{pmatrix}-y\\x\end{pmatrix}.$$

The idea of the Newton algorithm is to find the point where the gradient vanishes - in other words: find a zero of the derivative of f. In our example, the gradient ∇f of f does not vanish at any point on the circle $\mathcal{M}$.

We can instead reformulate the problem statement to: Finde the point, where the projected gradient $\operatorname{grad}_{\begin{pmatrix}x\\y\end{pmatrix}} f$ vanishes. In the example, the points

$$\begin{pmatrix}\frac{\sqrt{2}}{2}\\\frac{\sqrt{2}}{2}\end{pmatrix} \text{ and } \begin{pmatrix}-\frac{\sqrt{2}}{2}\\-\frac{\sqrt{2}}{2}\end{pmatrix}$$

would be a solution. But how does the algorithm work to find it?

We have to start out with some first iterate on the variety

$$\begin{pmatrix}x_0\\y_0\end{pmatrix} \in \mathcal{M}.$$

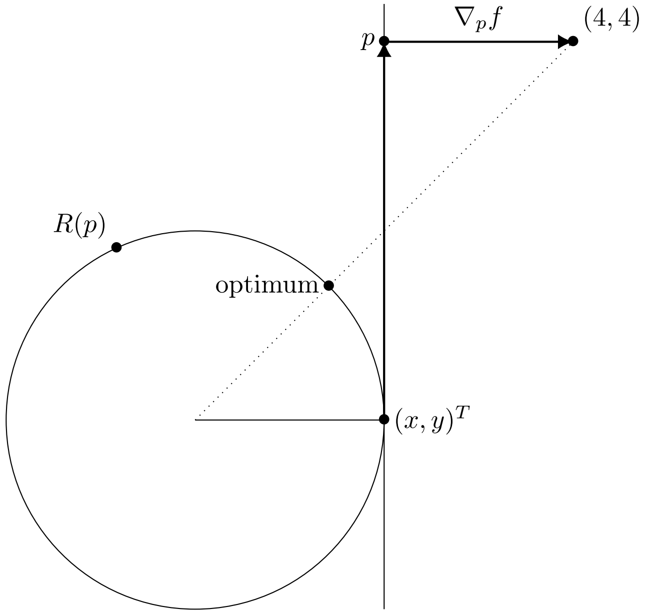
There is a naive but misleading idea: Search for some point on the current tangent space

$$\begin{pmatrix}x\\y\end{pmatrix} \in T_{\begin{pmatrix}x_0\\y_0\end{pmatrix}}\mathcal{M}$$

such that the projected gradient

$$\operatorname{grad}_{\begin{pmatrix}x\\y\end{pmatrix}} f$$

vanishes. The point in the tangent space that would be generated by this method is illustrated in the following picture:



This method, which can be expressed as

$$p \quad \text{such that} \quad P_{T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}} \nabla_p f = 0$$

yields a far too large step if $\mathcal{M}$ has large curvature compared to the gradient. The correct generalization of the Newton method to manifolds is the following: Because the projected gradient $\operatorname{grad}_{\begin{pmatrix}x\\y\end{pmatrix}} f$ depends on the current tangent space $T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}$, its derivative

$$\left(\operatorname{grad}\begin{pmatrix}x\\y\end{pmatrix} f\right)'$$

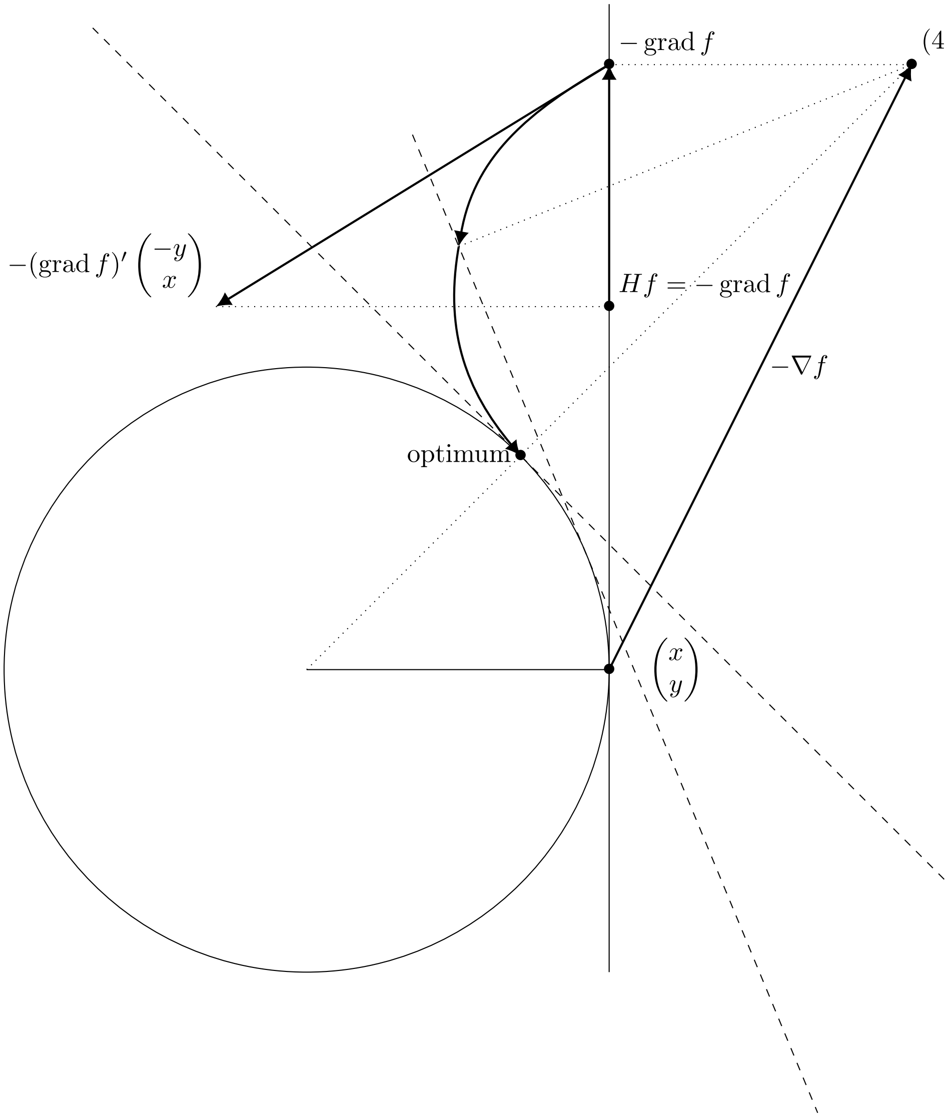
depends on the curvature of the manifold. In our example

$$\operatorname{grad} f = P_{T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}} \begin{pmatrix}x-4\\y-4\end{pmatrix}$$

depends on $T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}$ and

$$(\operatorname{grad} f)' = 4\begin{pmatrix} y & x-2y \\ y-2x & x \end{pmatrix}.$$

What is the meaning of $(\operatorname{grad} f)'$? If $v \in T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}$ is a tangent vector, then $(\operatorname{grad} f)'\, v$ tells how the projection changes when going in direction v. In the following figure you can see that the derivative of $\operatorname{grad}_{\begin{pmatrix}x\\y\end{pmatrix}} f$ does in general not lie in the tangent space.


$-\operatorname{grad} f$
$(4,4)$
$-(\operatorname{grad} f)'\begin{pmatrix}-y\\x\end{pmatrix}$
$Hf = -\operatorname{grad} f$
$-\nabla f$
optimum
$\begin{pmatrix}x\\y\end{pmatrix}$

Projecting it back to $T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}$, i.e.

$$P_{T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}} (\operatorname{grad} f)' \, v$$

yields the change of $\operatorname{grad} f$. The Newton equation $Hf = -\operatorname{grad} f$ is thus explicitely

$$P_{T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}} \left(P_{T_{\begin{pmatrix}x\\y\end{pmatrix}}\mathcal{M}} \nabla f \right)' v = -\operatorname{grad}_{\begin{pmatrix}x\\y\end{pmatrix}} f.$$

In our example this reads

$$-4(y-x)\begin{pmatrix}-y\\x\end{pmatrix} = \begin{pmatrix}-y\\x\end{pmatrix} 4\begin{pmatrix}-y & x\end{pmatrix}\begin{pmatrix}y & x-2y\\ y-2x & x\end{pmatrix} v.$$

As v is in the tangent space, we can write it as

$$\alpha\begin{pmatrix}-y\\x\end{pmatrix}$$

and the Newton equation becomes

$$-4(y-x) = 4\begin{pmatrix}-y & x\end{pmatrix}\begin{pmatrix}y & x-2y\\ y-2x & x\end{pmatrix}\begin{pmatrix}-y\\x\end{pmatrix}\alpha$$

and α can be determined as

$$\alpha = \frac{x-y}{x^3+y^3+xy^2+x^2y}$$

and with the defining equation of the manifold $x^2+y^2=1$ we can simplify to

$$\alpha = \frac{x-y}{x-y}.$$

Thus in the example we get

$$\begin{pmatrix}x_1\\y_1\end{pmatrix} = \frac{1}{x_0+y_0}\begin{pmatrix}1\\1\end{pmatrix}$$

which after an orthogonal projection as a retraction yields the optimal solution in just one single step. Even though in general the optimum is not attained after one step, it can be shown, that the convergence rate is quadratic. See for example [29, p.114, Chapter 6, Theorem 6.3.2].

A proof of the local quadratic convergence of the Newton method on manifolds can be found in [29, Theorem 6.3.2].

6.3 Rank adaptive algorithm

As we have shown above, the proposed gradient method on the TT variety converges either to a critical point on the smooth part or to a singular point. In the case of convergence to a singular point, we cannot ensure that it is locally optimal. See Figure 31 for an illustration of this case.

A possible solution to this problem is the following: When the algorithm detects, that it is converging to a singular point, it could "jump into this point" and restart the Riemannian gradient search from there. [27] use a very similar approach to construct globally convergent gradient methods for neural networks.

Another possible solution has been proposed in [34]: Start with the rank-1 variety, find a critical point and increase the allowed rank by one, and so on. The authors demonstrate very good convergence behaviour for this method in the matrix case.

References

[1] R. Schneider, A. Uschmajew, Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality, SIAM J. Optim. 25 (1) (2015) 622–646.

[2] S. Łojasiewicz, Ensembles semi-analytiques, Note des cours, Institut des Hautes Etudes Scientifiques, 1965.

[3] N. Cohen, O. Sharir, A. Shashua, On the expressive power of deep learning: A tensor analysis, in: V. Feldman, A. Rakhlin, O. Shamir (Eds.), 29th Annual Conference on Learning Theory, Vol. 49 of Proceedings of Machine Learning Research, PMLR, Columbia University, New York, New York, USA, 2016, pp. 698–728.

[4] H. Yserentant, Regularity and Approximability of Electronic Wave Functions, Springer, 2010.

[5] F. von Leitner, T. Klinger, A. Stechow, F. Rieger, Alternativlos.org 36.

[6] K. Kormann, A semi-Lagrangian Vlasov solver in tensor train format, arxiv:1408.7006.

[7] M. Pfeffer, Tensor methods for the numerical solution of high-dimensional parametric partial differential equations, Ph.D. thesis (2018).
URL http://dx.doi.org/10.14279/depositonce-7325

[8] R. Orús, Exploring corner transfer matrices and corner tensors for the classical simulation of quantum lattice systems, arXiv:1112.4101v2.

[9] A. Seigal, E. Robeva, Duality of graphical models and tensor networks, arXiv:1710.01437.

[10] D. Cox, J. Little, D. O'Shea, Ideals, Varieties and Algorithms, 3rd Edition, Springer, 2006.

[11] J. M. Landsberg, Tensors: Geometry and Applications, American Mathematical Society, 2012.

[12] T. G. Kolda, B. W. Bader, Tensor decompositions and applications, SIAM Review 51 (3) (2009) 455–500.

[13] P. Breiding, Numerical and statistical aspects of tensor decompositions, Ph.D. thesis (2017).

[14] W. Hackbusch, Tensor Spaces and Numerical Tensor Calculus, 3rd Edition, Springer, 2012.

[15] B. Buchberger, An algorithmic method in polynomial ideal theory, Reidel Publishing Company, Kluwer Academic Publisher, 1985, Ch. 6, pp. 184–232.

[16] J.-C. Faugère, A new efficient algorithm for computing gröbner bases without reduction to zero.

[17] D. Eisenbud, C. Huneke, W. Vasconcelos, Direct methods for primary decomposition, Inventiones mathematicae 110 (1992) 207–235.

[18] I. V. Oseledets, Tensor-train decomposition, SIAM J. Sci. Comput. 33 (5) (2011) 2295–2317.

[19] A. Uschmajew, B. Vandereycken, The geometry of algorithms using hierarchical tensors, Lin. Alg. Appl. 439 (1) (2013) 133–166.

[20] T. P. Cason, P.-A. Absil, P. Van Dooren, Iterative methods for low rank approximation of graph similarity matrices, Lin. Alg. Appl. 438 (4) (2013) 1863–1882.

[21] B. Kutschan, Tangent cones to tensor train varieties, Lin. Alg. Appl. 544 (2018) 370–390.

[22] D. O'Shea, L. Wilson, Limits of tangent spaces to real surfaces, Amer. J. Math. 126 (2004) 951–980.

[23] D. R. Grayson, M. E. Stillman, Macaulay2, a software system for research in algebraic geometry, Available at http://www.math.uiuc.edu/Macaulay2/.

[24] J. Harris, Algebraic Geometry - A First Course, 3rd Edition, Springer, 1992.

[25] S. Holtz, T. Rohwedder, R. Schneider, The alternating linear scheme for tensor optimization in the tensor train format, SIAM J. Sci. Comput. 43 (2) (2012) A683–A713.

[26] K. Hulek, Elementary Algebraic Geometry, Vol. 20, AMS Student Mathematical Library, 2003.

[27] B. D. Haeffele, R. Vidal, Global optimality in tensor factorization, deep learning, and beyond, arxiv:1506.07540v1.

[28] W. Hackbusch, S. Kühn, A new scheme for the tensor representation, J. Fourier Anal. Appl. 15 (2009) 706–722.

[29] P.-A. Absil, R. Mahony, R. Sepulchre, Optimization Algorithms on Matrix Manifolds, Princeton University Press, 2008.

[30] A. Uschmajew, Zur Theorie der Niedrigrangapproximation in Tensorprodukten von Hilberträumen, Ph.D. thesis (2013).

[31] Z. Li, Y. Nakatsukasa, T. Soma, A. Uschmajew, On orthogonal tensors and best rank-one approximation ratio, SIAM J. Matrix. Anal. Appl. 39 (1) (2018) 400–425.

[32] L. Qi, The best rank-one approximation ratio of a tensor space, SIAM J. Matrix Anal. Appl. 32 (2) (2011) 430–442.

[33] P.-A. Absil, R. Mahony, B. Andrews, Convergence of the iterates of descent methods for analytic cost functions, SIAM J. Optim. 16 (2005) 531–547.

[34] A. Uschmajew, B. Vandereycken, Line-search methods and rank increase on low-rank matrix varieties, 2014 International Symposium on Nonlinear Theory and its Applications.