

# An introduction to optimization on smooth manifolds

Nicolas Boumal

This is a pre-publication version, compiled with pdf annotations  
correcting an error and offering clarifications up to September 15, 2023.

Published version available from Cambridge University Press:  
<https://cambridge.org/9781009166157>

Author's book webpage, with video lectures, exercises and more:  
<https://www.nicolasboumal.net/book>



*To my family and mentors.*



# Contents

	<i>Preface</i>	<i>page</i> ix
	<i>Notation</i>	xiv
<b>1</b>	<b>Introduction</b>	1
<b>2</b>	<b>Simple examples</b>	4
2.1	Sensor network localization from directions: an affine subspace	4
2.2	Single extreme eigenvalue or singular value: spheres	5
2.3	Dictionary learning: products of spheres	6
2.4	Principal component analysis: Stiefel and Grassmann	8
2.5	Synchronization of rotations: special orthogonal group	11
2.6	Low-rank matrix completion: fixed-rank manifold	12
2.7	Gaussian mixture models: positive definite matrices	14
2.8	Smooth semidefinite programs	14
<b>3</b>	<b>Embedded geometry: first order</b>	17
3.1	Reminders of Euclidean space	21
3.2	Embedded submanifolds of a linear space	25
3.3	Smooth maps on embedded submanifolds	33
3.4	The differential of a smooth map	34
3.5	Vector fields and the tangent bundle	37
3.6	Moving on a manifold: retractions	39
3.7	Riemannian manifolds and submanifolds	41
3.8	Riemannian gradients	42
3.9	Local frames*	46
3.10	Notes and references	50
<b>4</b>	<b>First-order optimization algorithms</b>	53
4.1	A first-order Taylor expansion on curves	54
4.2	First-order optimality conditions	55
4.3	Riemannian gradient descent	56
4.4	Regularity conditions and iteration complexity	59
4.5	Backtracking line-search	61
4.6	Local convergence*	64

---

4.7	Computing gradients*	71
4.8	Numerically checking a gradient*	79
4.9	Notes and references	80
<b>5</b>	<b>Embedded geometry: second order</b>	<b>83</b>
5.1	The case for another derivative of vector fields	85
5.2	Another look at differentials of vector fields in linear spaces	85
5.3	Differentiating vector fields on manifolds: connections	86
5.4	Riemannian connections	89
5.5	Riemannian Hessians	94
5.6	Connections as pointwise derivatives*	97
5.7	Differentiating vector fields on curves	100
5.8	Acceleration and geodesics	105
5.9	A second-order Taylor expansion on curves	107
5.10	Second-order retractions	108
5.11	Special case: Riemannian submanifolds*	110
5.12	Special case: metric projection retractions*	114
5.13	Notes and references	116
<b>6</b>	<b>Second-order optimization algorithms</b>	<b>119</b>
6.1	Second-order optimality conditions	119
6.2	Riemannian Newton's method	121
6.3	Computing Newton steps: conjugate gradients	124
6.4	Riemannian trust regions	131
6.5	The trust-region subproblem: truncated CG	144
6.6	Local convergence of RTR with tCG*	147
6.7	Simplified assumptions for RTR with tCG*	148
6.8	Numerically checking a Hessian*	150
6.9	Notes and references	151
<b>7</b>	<b>Embedded submanifolds: examples</b>	<b>154</b>
7.1	Euclidean spaces as manifolds	154
7.2	The unit sphere in a Euclidean space	157
7.3	The Stiefel manifold: orthonormal matrices	159
7.4	The orthogonal group and rotation matrices	163
7.5	Fixed-rank matrices	165
7.6	The hyperboloid model	173
7.7	Manifolds defined by $h(x) = 0$	177
7.8	Notes and references	180

<b>8</b>	<b>General manifolds</b>	182
8.1	A permissive definition	182
8.2	The atlas topology, and a final definition	188
8.3	Embedded submanifolds are manifolds	192
8.4	Tangent vectors and tangent spaces	194
8.5	Differentials of smooth maps	196
8.6	Tangent bundles and vector fields	197
8.7	Retractions and velocity of a curve	199
8.8	Coordinate vector fields as local frames	200
8.9	Riemannian metrics and gradients	201
8.10	Lie brackets as vector fields	201
8.11	Riemannian connections and Hessians	203
8.12	Covariant derivatives and geodesics	205
8.13	Taylor expansions and second-order retractions	205
8.14	Submanifolds embedded in manifolds	206
8.15	Notes and references	210
<b>9</b>	<b>Quotient manifolds</b>	212
9.1	A definition and a few facts	216
9.2	Quotient manifolds through group actions	219
9.3	Smooth maps to and from quotient manifolds	223
9.4	Tangent, vertical and horizontal spaces	224
9.5	Vector fields	226
9.6	Retractions	231
9.7	Riemannian quotient manifolds	232
9.8	Gradients	234
9.9	A word about Riemannian gradient descent	236
9.10	Connections	238
9.11	Hessians	239
9.12	A word about Riemannian Newton's method	240
9.13	Total space embedded in a linear space	243
9.14	Horizontal curves and covariant derivatives	246
9.15	Acceleration, geodesics and second-order retractions	247
9.16	Grassmann manifold: summary*	250
9.17	Notes and references	254
<b>10</b>	<b>Additional tools</b>	260
10.1	Distance, geodesics and completeness	260
10.2	Exponential and logarithmic maps	264
10.3	Parallel transport	270
10.4	Lipschitz conditions and Taylor expansions	274
10.5	Transporters	285
10.6	Finite difference approximation of the Hessian	292
10.7	Tensor fields and their covariant differentiation	294
10.8	Notes and references	302

<b>11</b>	<b>Geodesic convexity</b>	307
11.1	Convex sets and functions in linear spaces	308
11.2	Geodesically convex sets and functions	310
11.3	Alternative definitions of geodesically convex sets*	314
11.4	Differentiable geodesically convex functions	316
11.5	Geodesic strong convexity and Lipschitz continuous gradients	319
11.6	Example: positive reals and geometric programming	323
11.7	Example: positive definite matrices	326
11.8	Notes and references	329
	<i>References</i>	331
	<i>Index</i>	347



# Preface

Optimization problems on smooth manifolds arise in science and engineering as a result of natural geometry (e.g., the set of orientations of physical objects in space is a manifold), latent data simplicity (e.g., high-dimensional data points lie close to a low-dimensional linear subspace, leading to low-rank data matrices), symmetry (e.g., observations are invariant under rotation, translation or other group actions, leading to quotients) and positivity (e.g., covariance matrices and diffusion tensors are positive definite). This has led to successful applications notably in machine learning, computer vision, robotics, scientific computing, dynamical systems and signal processing.

Accordingly, optimization on manifolds has garnered increasing interest from researchers and engineers alike. Building on fifty years of research efforts that have recently intensified, it is now recognized as a wide, beautiful and effective generalization of unconstrained optimization on linear spaces.

Yet, engineering programs seldom include training in differential geometry: the field of mathematics concerned with smooth manifolds. Moreover, existing textbooks on this topic usually align with the interests of mathematicians more than with the needs of engineers and applied mathematicians. This creates a significant but avoidable barrier to entry for optimizers.

One of my goals in writing this book is to offer a different, if at times unorthodox, introduction to differential geometry. Definitions and tools are introduced in a need-based order for optimization. We start with a restricted setting—that of embedded submanifolds of linear spaces—which allows us to define all necessary concepts in direct reference to their usual counterparts from linear spaces. This covers a wealth of applications.

In what is perhaps the clearest departure from standard exposition, charts and atlases are not introduced until quite late. The reason for doing so is twofold: pedagogically, charts and atlases are more abstract than what is needed to work on embedded submanifolds; and pragmatically, charts are seldom if ever useful in practice. It would be unfortunate to give them center stage.

Of course, charts and atlases are the right tool to provide a unified treatment of all smooth manifolds in an intrinsic way. They are introduced eventually, at which point it becomes possible to discuss quotient manifolds: a powerful language to understand symmetry in optimization. Perhaps this abstraction is necessary to fully appreciate the depth of optimization on manifolds as more

than just a fancy tool for constrained optimization in linear spaces, and truly a mathematically natural setting for *unconstrained* optimization in a wider sense.

Time-tested optimization algorithms are introduced immediately after the early chapters about embedded geometry. Crucially, the design and analysis of these methods remain unchanged whether we are optimizing on a manifold which is embedded in a linear space or not. This makes it possible to get to algorithms early on, without sacrificing generality. It also underlines the conceptual point that the algorithms truly operate on the manifolds intrinsically.

The last two chapters visit more advanced topics that are not typically necessary for simple applications. The first one delves deeper into geometric tools. The second one introduces the basics of *geodesic* convexity: a broad generalization of convexity, which is one of the most fruitful structures in classical optimization.

### Intended audience

This book is intended for students and researchers alike. The material has proved popular with applied mathematicians and mathematically inclined engineering and computer science students at the graduate and advanced undergraduate levels.

Readers are assumed to be comfortable with linear algebra and multivariable calculus. Central to the *raison d'être* of this book, there are no prerequisites in differential geometry or optimization. For computational aspects, it is helpful to have notions of numerical linear algebra, for which I recommend the approachable textbook by Trefethen and Bau [TB97].

Building on these expectations, the aim is to give full proofs and intuition for all concepts that are introduced, at least for submanifolds of linear spaces. The hope is to equip readers to pursue research projects in (or using) optimization on manifolds, involving both mathematical analysis and efficient implementation.

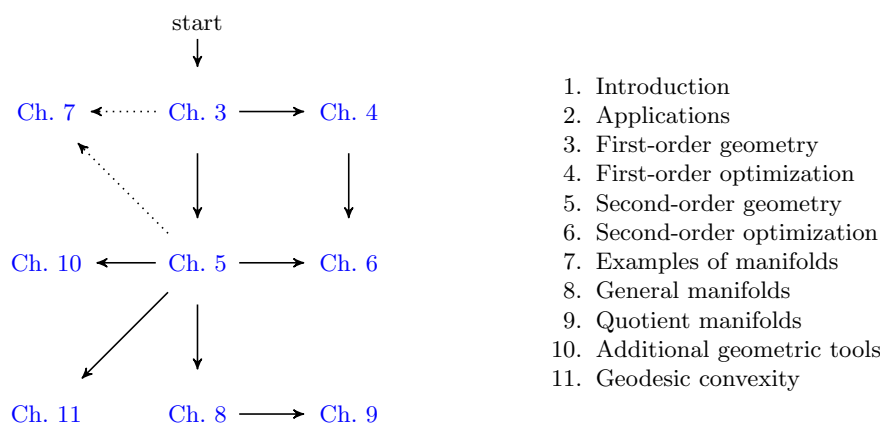
### How to use this book

The book is self-contained and should suit both self-learners and instructors.

Chapters 3 and 5 can serve as a standalone introduction to differential and Riemannian geometry. They focus on embedded submanifolds of linear spaces, with proofs. Chapter 7 details examples of manifolds: it is meant for on-and-off reading in parallel with Chapters 3 and 5. These chapters do not involve charts, and they aim to convey the fact that geometric tools are computational tools.

From there, the expected next step is to work through Chapters 4 and 6 about optimization algorithms. Readers may also choose to embark on Chapter 8 to see how embedded manifolds fit into the general theory of smooth manifolds. That is a useful (though not fully necessary) stepping stone toward Chapter 9 about quotient manifolds. Alternatively, they may decide to learn about further geometric tools in Chapter 10 or about a Riemannian notion of convexity in Chapter 11.

These chapter dependencies are summarized in the diagram below, where an arrow from A to B means it is preferable to read A before B.



In a graduate course at Princeton University in 2019 and 2020 (24 lectures of 80 minutes each), I covered much of Chapters 1–6 and select parts of Chapter 7 before the midterm break, then much of Chapters 8–9 and select parts of Chapters 10–11 after the break. At EPFL in 2021, I discussed mostly Chapters 1–8 in 13 lectures of 90 minutes each supplemented with exercise sessions.

The numerous exercises in the book have wide-ranging difficulty levels. Some are included in part as a way to convey information while skipping technicalities.

Starred sections can be skipped safely for a first encounter with the material. Chapters end with references and notes that many readers may find relevant but which would otherwise break the flow. Did the mark in the margin catch your attention? That is its purpose. You may see a couple of those in the book. ★

## What is new, or different, or hard to find elsewhere

The de facto reference for optimization on manifolds is the landmark 2008 book *Optimization Algorithms on Matrix Manifolds* by Pierre-Antoine Absil, Robert Mahony and Rodolphe Sepulchre [AMS08]. It is an important source for the present book as well, with significant overlap of topics. In the years since, the field has evolved, and with it the need for an entry point catering to a broader audience. In an effort to address these needs, I aim to:

### 1. Provide a different, self-contained introduction to the core concepts.

This includes a “charts last” take on differential geometry with proofs adapted accordingly; a somewhat unusual yet equivalent definition of connections that (I believe) is more intuitive from an optimizer’s point of view; and an account of optimization on quotient manifolds which benefits from years of hindsight. This introduction is informed by the pains I had entering the field.

**2. Discuss new topics that have grown in importance since 2008.**

This includes a replacement of asymptotic convergence results in favor of worst-case, non-asymptotic iteration complexity results; a related take on Lipschitz continuity for Riemannian gradients and Hessians paired with their effect on Taylor expansions on manifolds; an explicit construction of geometric tools necessary for optimization over matrices of fixed rank; a simple study of metric projection retractions; an extrinsic view of the Riemannian Hessian for submanifolds through the Weingarten map and second fundamental form; a discussion of the smooth invertibility of retractions and of the domain of the inverse of the exponential map; transporters as a natural alternative to vector and parallel transports; finite differences of gradients to approximate Hessians; and an introduction to geodesic convexity (not restricted to Hadamard manifolds) with a gradient algorithm for the strongly convex case. Many of these build on research papers referenced in text.

**3. Share tricks of the trade that are seldom, if ever, spelled out.**

This includes several examples of manifolds worked out in full detail; pragmatic instructions for how to derive expressions for gradients and Hessians of matrix functions, and how to check them numerically; explicit formulas for geometric tools on product manifolds (mostly given as exercises); and a number of comments informed by ten years of software development in the field.

The main differential geometry references I used are the fantastic books by Lee [Lee12, Lee18], O’Neill [O’N83], and Brickell and Clark [BC70]. Definitions of geometric concepts in this book, though at times stated differently, are fully compatible with Absil et al.’s book. This is also compatible with Lee’s textbooks with one exception: Riemannian *sub*manifolds to us are understood to be embedded submanifolds, whereas Lee also allows them to be merely immersed submanifolds. Moreover, we use the word “manifold” to mean “smooth manifold,” that is,  $C^\infty$ . Most results extend to manifolds and functions of class  $C^k$ .

There is much to say about the impact of curvature on optimization. This is an active research topic that has not stabilized yet. Therefore, I chose to omit curvature entirely from this book, save for a few brief comments in the last two chapters. Likewise, optimization on manifolds is proving to be a particularly fertile ground for benign non-convexity and related phenomena. There are only a few hints to that effect throughout the book: the research continues.

## Software and online resources

Little to no space is devoted to existing software packages for optimization on manifolds, or to numerical experiments. Yet, such packages significantly speed up research and development in the field. The reader may want to experiment with Manopt (Matlab), PyManopt (Python) or Manopt.jl (Julia), all available from [manopt.org](https://manopt.org).

In particular, the Matlab implementations of most manifolds discussed in this book are listed in Table 7.1 on p155. Gradient descent (Algorithm 4.1) with backtracking line-search (Algorithm 4.2) is available as `steepestdescent`. The trust-region method (Algorithm 6.3) with the truncated conjugate gradient sub-problem solver (Algorithm 6.4) is available as `trustregions`. These implementations include a wealth of tweaks and tricks that are important in practice: many are explained here, some are only documented in the code. The Python and Julia versions offer similar features.

The following webpage collects further resources related to this book:

[nicolasboumal.net/book](https://nicolasboumal.net/book)

In particular, teaching and learning material will be listed there, as well as errata.

## Thanks

A number of people offered decisive comments. I thank Pierre-Antoine Absil (who taught me much of this) and Rodolphe Sepulchre for their input at the early stages of planning for this book, as well as (in no particular order) Eitan Levin, Chris Criscitiello, Quentin Rejock, Razvan-Octavian Radu, Joe Kileel, Bart Vandereycken, Coralia Cartis, Bamdev Mishra, Suvrit Sra, Stephen McKeown, John M. Lee and Sándor Z. Németh for numerous conversations that led to direct improvements. Likewise, reviewers offered welcome advice and suggestions for additions (that I was only able to implement partially). Another big thank you to the people involved with the Manopt toolboxes: these efforts are led with Bamdev for the Matlab version; by Jamie Townsend, Niklas Koep and Sebastian Weichwald for the Python version; and by Ronny Bergmann for the Julia version.

I am also indebted to the mathematics departments at Princeton University and EPFL for supporting me while I was writing. Finally, I thank Katie Leach at Cambridge University Press for her enthusiasm and candid advice that helped shape this project into its final form.

Nicolas Boumal  
[nicolas.boumal@epfl.ch](mailto:nicolas.boumal@epfl.ch)

Lausanne, Switzerland

# Notation

The following lists typical uses of symbols. Local exceptions are documented in place. For example,  $c$  typically denotes a curve, but sometimes denotes a real constant. Symbols defined and used locally only are omitted.

$\mathbb{R}, \mathbb{C}$	Real and complex numbers
$\mathbb{R}_+$	Positive reals ( $x > 0$ )
$\mathbb{R}^{m \times n}$	Real matrices of size $m \times n$
$\mathbb{R}_r^{m \times n}$	Real matrices of size $m \times n$ and rank $r$
$\text{Sym}(n), \text{Skew}(n)$	Symmetric and skew-symmetric real matrices of size $n$
$\text{sym}(M), \text{skew}(M)$	Symmetric and skew-symmetric parts of a matrix $M$
$\text{Sym}(n)^+$	Symmetric positive definite real matrices of size $n$
$\text{Tr}(M), \det(M)$	Trace, determinant of a square matrix $M$
$\text{diag}(M)$	Vector of diagonal entries of a matrix $M$
$\text{diag}(u_1, \dots, u_n)$	Diagonal matrix of size $n$ with given diagonal entries
$M^\dagger$	Moore–Penrose pseudoinverse of matrix $M$
$I_d$	Identity matrix of size $d$
$I$	Subset of $\mathbb{R}$ (often open with $0 \in I$ ) or identity matrix
$\text{Id}$	Identity operator
$ a $	Modulus of $a \in \mathbb{C}$ (absolute value if $a \in \mathbb{R}$ )
$ A $	Cardinality of a set $A$
$\mathcal{E}, \mathcal{E}', \mathcal{F}$	Linear spaces, often with a Euclidean structure
$\mathcal{M}, \mathcal{M}', \overline{\mathcal{M}}, \mathcal{N}$	Smooth manifolds, often with a Riemannian structure
$S^{d-1}$	Unit sphere, in a Euclidean space of dimension $d$
$\text{OB}(d, n)$	Oblique manifold (product of $S^{d-1}$ copied $n$ times)
$\text{O}(d), \text{SO}(d)$	Orthogonal and special orthogonal groups in $\mathbb{R}^{d \times d}$
$\text{St}(n, p)$	Stiefel manifold embedded in $\mathbb{R}^{n \times p}$
$\text{Gr}(n, p)$	Grassmann manifold as the quotient $\text{St}(n, p)/\text{O}(p)$
$\text{GL}(n)$	General linear group (invertible matrices in $\mathbb{R}^{n \times n}$ )
$\text{H}^n$	Hyperbolic space as hyperboloid embedded in $\mathbb{R}^{n+1}$
$\dim \mathcal{M}$	Dimension of $\mathcal{M}$
$x, y, z$	Points on a manifold
$u, v, w, s, \xi, \zeta$	Tangent vectors
$p, q$	Integers or polynomials or points on a manifold
$\text{T}\mathcal{M}$	Tangent bundle of $\mathcal{M}$

$T_x\mathcal{M}$	Tangent space to $\mathcal{M}$ at $x \in \mathcal{M}$
$N_x\mathcal{M}$	Normal space (orthogonal complement of $T_x\mathcal{M}$ )
$\text{Proj}_x, \text{Proj}_x^\perp$	Orthogonal projector to $T_x\mathcal{M}, N_x\mathcal{M}$
$H_x, V_x$	Horizontal and vertical space at $x$ for a quotient manifold
$\text{Proj}_x^H, \text{Proj}_x^V$	Orthogonal projectors to $H_x, V_x$
$\text{lift}_x$	Horizontal lift operator for quotient manifolds
$R_x(v)$	Retraction $R$ evaluated at $(x, v) \in T\mathcal{M}$
$\text{Exp}_x(v)$	Exponential map $\text{Exp}$ evaluated at $(x, v) \in T\mathcal{M}$
$\text{Log}_x(y)$	Vector $v$ such that $\text{Exp}_x(v) = y$ (see Definition 10.20)
$\exp, \log$	Scalar or matrix exponential and logarithm
$\mathcal{O}, \mathcal{O}_x$	Domain of $\text{Exp}$ (subset of $T\mathcal{M}$ ), $\text{Exp}_x$ (subset of $T_x\mathcal{M}$ ); Can also denote these domains for a non-global retraction.
$\text{inj}(\mathcal{M}), \text{inj}(x)$	Injectivity radius of a manifold, at a point
$\langle \cdot, \cdot \rangle_x$	Riemannian inner product on $T_x\mathcal{M}$
$\langle \cdot, \cdot \rangle$	Euclidean inner product; Sometimes denotes $\langle \cdot, \cdot \rangle_x$ with subscript omitted.
$\  \cdot \ , \  \cdot \ _x$	Norms associated to $\langle \cdot, \cdot \rangle$ and $\langle \cdot, \cdot \rangle_x$
$\  \cdot \ $	Also denotes operator norm for linear maps
$[\cdot, \cdot]$	Lie bracket
$\nabla$	Affine connection (often Riemannian) on a manifold
$\frac{d}{dt}$	Classical derivative with respect to $t$
$\frac{D}{dt}$	Covariant derivative induced by a connection $\nabla$
$\frac{\partial}{\partial x_i}$	Partial derivative with respect to real variable $x_i$
$x_i$	Often the $i$ th coordinate of a vector $x \in \mathbb{R}^n$
$x_k$	Often the $k$ th element of a sequence $x_0, x_1, x_2, \dots \in \mathcal{M}$
$f, g$	Real-valued functions
$f_{\text{low}}$	A real number such that $f(x) \geq f_{\text{low}}$ for all $x$
$h$	Often a local defining function with values in $\mathbb{R}^k$
$\text{grad}f, \text{Hess}f$	Riemannian gradient and Hessian of $f$ ; Euclidean gradient, Hessian if domain of $f$ is Euclidean.
$\nabla f, \nabla^2 f$	First and second covariant derivatives of $f$ as tensor fields
$c, \gamma$	Curves
$c', \gamma', \dot{c}, \dot{\gamma}$	Velocity vector fields of curves $c, \gamma$
$c'', \gamma''$	Intrinsic acceleration vector fields of $c, \gamma$
$\ddot{c}, \ddot{\gamma}$	Extrinsic acceleration vector fields of $c, \gamma$
$L, L_g, L_H$	Lipschitz constants (nonnegative reals)
$L(c)$	Length of a curve $c$
$\text{dist}(x, y)$	Distance (often Riemannian) between two points $x, y$
$B(x, r)$	Open ball $\{v \in T_x\mathcal{M} : \ v\ _x < r\}$ or $\{y \in \mathcal{E} : \ y - x\  < r\}$
$\bar{B}(x, r)$	Closed ball as above
$\mathcal{A}, \mathcal{L}$	Linear maps
$\mathcal{A}, \mathcal{A}^+$	Atlas, maximal atlas
$\text{im } \mathcal{L}, \ker \mathcal{L}$	Range space (image) and null space (kernel) of $\mathcal{L}$

---

$\text{rank}(M), \text{rank}(\mathcal{L})$	Rank of a matrix or linear map
$M^\top, M^*$	Transpose or Hermitian conjugate-transpose of matrix $M$
$\mathcal{L}^*$	Adjoint of a linear map $\mathcal{L}$ between Euclidean spaces
$\mathcal{A} \succeq 0, \mathcal{A} \succ 0$	States $\mathcal{A} = \mathcal{A}^*$ is positive semidefinite or positive definite
$\text{span}(u_1, \dots, u_m)$	Linear subspace spanned by vectors $u_1, \dots, u_m$
$F, G, H$	Maps, usually to and from linear spaces or manifolds
$F: A \rightarrow B$	A map defined on the whole domain $A$
$F _U$	Restriction of the map $F$ to the domain $U$
$F(\cdot, y)$	For a map $(x, y) \mapsto F(x, y)$ , this is the map $x \mapsto F(x, y)$
$F \circ G$	Composition of maps: $(F \circ G)(x) = F(G(x))$
$DF(x)[v]$	Differential of $F$ at $x$ along $v$
$U, V, W, X, Y, Z$	Vector fields on a manifold, or Matrices which could be tangent vectors or points on $\mathcal{M}$ ;
$Y, Z$	Can also be vector fields along a curve.
$U, V, W, O$	Can also be open sets, usually in a linear space.
$\bar{f}, \bar{F}, \bar{V}, \dots$	Smooth extensions or lifts of $f, F, V, \dots$
$\bar{u}, \bar{U}$	Can also denote complex conjugation of $u, U$
$T$	Tensor field
$T_s$	Differential of retraction $\text{DR}_x(s)$
$\mathcal{U}, \mathcal{V}$	Open sets in a manifold
$\mathcal{W}$	Weingarten map
$\Pi$	Second fundamental form
$fV$	Vector field $x \mapsto f(x)V(x)$ (with $f$ real valued)
$Vf$	Real function $x \mapsto Df(x)[V(x)]$
$\mathfrak{F}(\mathcal{M}), \mathfrak{F}(\mathcal{E})$	Set of smooth real-valued functions on $\mathcal{M}, \mathcal{E}$
$\mathfrak{X}(\mathcal{M}), \mathfrak{X}(\mathcal{E})$	Set of smooth vector fields on $\mathcal{M}, \mathcal{E}$
$\mathfrak{X}(c)$	Set of smooth vector fields along a curve $c$
$T_{y \leftarrow x}$	Vector transport from $T_x \mathcal{M}$ to $T_y \mathcal{M}$
$\text{PT}_{t_1 \leftarrow t_0}^c$	Parallel transport along curve $c$ from $c(t_0)$ to $c(t_1)$
$P_s$	Parallel transport along $\gamma(t) = \text{Exp}_x(ts)$ from 0 to 1
$\sim$	Equivalence relation
$A/\sim$	Quotient set of $A$ by the relation $\sim$
$[x]$	Equivalence class of $x$ for some equivalence relation
$\pi$	Canonical projection $\pi: T\mathcal{M} \rightarrow \mathcal{M}$ or $\pi: \overline{\mathcal{M}} \rightarrow \overline{\mathcal{M}}/\sim$ ; Occasionally denotes the mathematical constant.
$A \subset B$	$A$ is a proper subset of $B$ (the sets are not equal)
$A \subseteq B$	$A$ is a subset of $B$ (the sets may be equal)
$A \cap B$	Intersection of sets $A, B$
$A \cup B$	Union of sets $A, B$
$\emptyset$	Empty set.



# 1 Introduction

---

Optimization is a staple of mathematical modeling. In this rich framework, we consider a set  $S$  called the *search space*—it contains all possible answers to our problem, good and bad—and a *cost function*  $f: S \rightarrow \mathbb{R}$  which associates a cost  $f(x)$  to each element  $x$  of  $S$ . The goal is to find  $x \in S$  such that  $f(x)$  is as small as possible, that is, a best answer. We write

$$\min_{x \in S} f(x)$$

to represent both the optimization problem and the minimal cost (if it exists). Occasionally, we wish to denote specifically the subset of  $S$  for which the minimal cost is attained; the standard notation is

$$\arg \min_{x \in S} f(x),$$

bearing in mind that this set might be empty. We will discuss a few simple applications which can be modeled in this form.

Rarely, optimization problems admit an analytical solution. Typically, we need numerical algorithms to (try to) solve them. Often, the best algorithms exploit mathematical structure in  $S$  and  $f$ .

An important special case arises when  $S$  is a linear space such as  $\mathbb{R}^n$ . Minimizing a function  $f$  in  $\mathbb{R}^n$  is called *unconstrained optimization* because the variable  $x$  is free to move around  $\mathbb{R}^n$ , unrestricted.

If  $f$  is sufficiently differentiable and  $\mathbb{R}^n$  is endowed with an inner product (that is, if we make it into a Euclidean space), then we have a notion of gradient and perhaps even a notion of Hessian for  $f$ . These objects give us a firm understanding of how  $f$  behaves locally around any given point. Famous algorithms such as gradient descent and Newton’s method exploit these objects to move around  $\mathbb{R}^n$  efficiently in search of a solution.

Notice, however, that the Euclidean structure of  $\mathbb{R}^n$  and the smoothness of  $f$  are irrelevant to the definition of the optimization problem itself: they are merely structures that we may (and as experience shows, we should) use algorithmically to our advantage.

Subsuming linearity, we focus on *smoothness* as the key structure to exploit: we assume the set  $S$  is a *smooth manifold* and the function  $f$  is smooth on  $S$ . This calls for precise definitions, constructed first in Chapter 3. For a first intuition,

one can think of smooth manifolds as surfaces in  $\mathbb{R}^n$  that do not have kinks or boundaries, such as a plane, a sphere, a torus, or a hyperboloid.

We could think of optimization over such surfaces as *constrained*, in the sense that  $x$  is not allowed to move freely in  $\mathbb{R}^n$ : it is constrained to remain on the surface. Alternatively, and this is the viewpoint favored here, we can think of this as unconstrained optimization, in a world where the smooth surface is the only thing that exists: like an ant walking on a large ball might feel unrestricted in its movements, aware only of the sphere it lives on; or like the two-dimensional inhabitants of Flatland [Abb84] who find it hard to imagine that there exists such a thing as a third dimension, feeling thoroughly free in their own subspace.

A natural question then is: can we generalize the Euclidean algorithms from unconstrained optimization to handle the broader class of optimization over smooth manifolds? The answer is essentially yes, going back to the 70s [Lue72, Lic79], the 80s [Gab82] and the 90s [Udr94, Smi94, HM96, Rap97, EAS98], and sparking a significant amount of research in the past two decades.

To generalize algorithms such as gradient descent and Newton’s method, we need a proper notion of gradient and Hessian on smooth manifolds. In the linear case, this required the introduction of an inner product: a Euclidean structure. In our more general setting, we leverage the fact that smooth manifolds can be linearized locally around every point. The linearization at  $x$  is called the *tangent space* at  $x$ . By endowing each tangent space with its own inner product (varying smoothly with  $x$ , in a sense to be made precise), we construct what is called a *Riemannian structure* on the manifold: it becomes a *Riemannian manifold*.

A Riemannian structure is sufficient to define gradients and Hessians on the manifold, paving the way for optimization. There exist several Riemannian structures on each manifold: our choice may impact algorithmic performance. In that sense, identifying a useful structure is part of the algorithm design—as opposed to being part of the problem formulation, which ended with the definition of the search space (as a crude set) and the cost function.

Chapter 2 covers a few simple applications, mostly to give a sense of how manifolds come up. We then go on to define smooth manifolds in a restricted<sup>1</sup> setting in Chapter 3, where manifolds are *embedded* in a linear space, much like the unit sphere in three-dimensional space. In this context, we define notions of smooth functions, smooth vector fields, gradients and *retractions* (a means to move around on a manifold). These tools are sufficient to design and analyze a first optimization algorithm in Chapter 4: Riemannian gradient descent. As readers progress through these chapters, it is the intention that they also read bits of Chapter 7 from time to time: useful embedded manifolds are studied there in detail. Chapter 5 provides more advanced geometric tools for embedded manifolds, including the notions of Riemannian *connections* and Hessians. These

<sup>1</sup> Some readers may know Whitney’s celebrated embedding theorems, which state that any smooth manifold can be embedded in a linear space [BC70, p82]. The mere existence of an embedding, however, is of little use for computation.

are put to good use in Chapter 6 to design and analyze Riemannian versions of Newton’s method and the trust-region method.

The linear *embedding space* is useful for intuition, to simplify definitions, and to design tools. Notwithstanding, all the tools and concepts we define in the restricted setting are *intrinsic*, in the sense that they are well defined regardless of the embedding space. We make this precise much later, in Chapter 8, where all the tools from Chapters 3 and 5 are redefined in the full generality of standard treatments of differential geometry. This is also the time to discuss topological issues to some extent. Generality notably makes it possible to discuss a more abstract class of manifolds called *quotient manifolds* in Chapter 9. They offer a beautiful way to harness symmetry, so common in applications.

In closing, Chapter 10 offers a limited treatment of more advanced geometric tools such as the Riemannian distance, geodesics, the exponential map and its inverse, parallel transports and transporters, notions of Lipschitz continuity, finite differences, and covariant differentiation of tensor fields. Then, Chapter 11 covers elementary notions of convexity on Riemannian manifolds with simple implications for optimization. This topic has been around since the 90s, and has been gaining traction in research lately.

More than 150 years ago, Riemann invented a new kind of geometry for the abstract purpose of understanding curvature in high-dimensional spaces. Today, this geometry plays a central role in the development of efficient algorithms to tackle technological applications Riemann himself—arguably—could have never envisioned. Through this book, I invite you to enjoy this singularly satisfying success of mathematics, with an eye to turning geometry into algorithms.

## 2 Simple examples

---

Before formally defining what manifolds are, and before introducing any particular algorithms, this chapter surveys simple problems that are naturally modeled as optimization on manifolds. These problems are motivated by applications in various scientific and technological domains. We introduce them chiefly to illustrate how manifolds arise and to motivate the mathematical abstractions in subsequent chapters.

The first example leads to optimization on an affine subspace: it falls within the scope of optimization on manifolds, but one can also handle it with classical tools. Subsequently, we encounter optimization on spheres, products of spheres, orthonormal matrices, the set of all linear subspaces, rotation matrices, fixed-rank matrices, positive definite matrices and certain quadratic surfaces. Through those, we get a glimpse of the wide reach of optimization on manifolds.

Below, we use a few standard concepts from linear algebra and calculus that are revisited in Section 3.1.

### 2.1 Sensor network localization from directions: an affine subspace

Consider  $n$  sensors located at unknown positions  $t_1, \dots, t_n$  in  $\mathbb{R}^d$ . We aim to locate the sensors, that is, estimate the positions  $t_i$ , based on some directional measurements. Specifically, for each pair of sensors  $(i, j)$  corresponding to an edge of a graph  $G$ , we receive a noisy measurement of the direction from  $t_j$  to  $t_i$ :

$$v_{ij} \approx \frac{t_i - t_j}{\|t_i - t_j\|},$$

where  $\|x\| = \sqrt{x_1^2 + \dots + x_d^2}$  is the Euclidean norm on  $\mathbb{R}^d$  induced by the inner product  $\langle u, v \rangle = u^\top v = u_1 v_1 + \dots + u_d v_d$ .

There are two fundamental ambiguities in this task. First, directional measurements reveal nothing about the global location of the sensors: translating the sensors as a whole does not affect pairwise directions. Thus, we may assume without loss of generality that the sensors are centered:

$$t_1 + \dots + t_n = 0.$$

Second, the measurements reveal nothing about the global scale of the sensor

arrangement. Specifically, scaling all positions  $t_i$  by a scalar  $\alpha > 0$  as  $\alpha t_i$  has no effect on the directions separating the sensors, so that the true scale cannot be recovered from the measurements. It is thus legitimate to fix the scale arbitrarily, to break symmetry. One fruitful way is to assume the following [HLV18]:

$$\sum_{(i,j) \in G} \langle t_i - t_j, v_{ij} \rangle = 1.$$

Indeed, if this constraint holds for some set of locations  $t_1, \dots, t_n$ , then it does not hold for locations  $\alpha t_1, \dots, \alpha t_n$  unless  $\alpha = 1$ .

Given a tentative estimator  $\hat{t}_1, \dots, \hat{t}_n \in \mathbb{R}^d$  for the locations, we may assess its compatibility with the measurement  $v_{ij}$  by computing

$$\|(\hat{t}_i - \hat{t}_j) - \langle \hat{t}_i - \hat{t}_j, v_{ij} \rangle v_{ij}\|.$$

Indeed, if  $\hat{t}_i - \hat{t}_j$  and  $v_{ij}$  are aligned in the same direction, this evaluates to zero. Otherwise, it evaluates to a positive number, growing as alignment degrades. Combined with the symmetry-breaking conditions, this suggests the following formulation for sensor network localization from direction measurements:

$$\begin{aligned} \min_{\hat{t}_1, \dots, \hat{t}_n \in \mathbb{R}^d} \quad & \sum_{(i,j) \in G} \|(\hat{t}_i - \hat{t}_j) - \langle \hat{t}_i - \hat{t}_j, v_{ij} \rangle v_{ij}\|^2 \\ \text{subject to} \quad & \hat{t}_1 + \dots + \hat{t}_n = 0 \quad \text{and} \quad \sum_{(i,j) \in G} \langle \hat{t}_i - \hat{t}_j, v_{ij} \rangle = 1. \end{aligned}$$

The role of the second constraint is clear: it excludes  $\hat{t}_1 = \dots = \hat{t}_n = 0$ , which would otherwise be optimal.

Grouping the variables as the columns of a matrix, we find that the search space for this problem is an affine subspace of  $\mathbb{R}^{d \times n}$ : this is a *linear manifold*. It is also an *embedded submanifold* of  $\mathbb{R}^{d \times n}$ . Hence, it falls within our framework.

With the simple cost function as above, this problem is in fact a convex quadratic minimization problem on an affine subspace. As such, it admits an explicit solution which merely requires solving a linear system. Optimization algorithms can be used to solve this system implicitly. More importantly, the power of optimization algorithms lies in the flexibility that they offer: alternative cost functions may be used to improve robustness against specific noise models for example, and those require more general algorithms [HLV18].

## 2.2 Single extreme eigenvalue or singular value: spheres

Let  $A \in \mathbb{R}^{n \times n}$  be a symmetric matrix:  $A = A^\top$ . By the spectral theorem,  $A$  admits  $n$  real eigenvalues  $\lambda_1 \leq \dots \leq \lambda_n$  and corresponding real, orthonormal eigenvectors  $v_1, \dots, v_n \in \mathbb{R}^n$ , where orthonormality is assessed with respect to the standard inner product over  $\mathbb{R}^n$ :  $\langle u, v \rangle = u^\top v$ .

For now, we focus on computing one extreme eigenpair of  $A$ :  $(\lambda_1, v_1)$  or  $(\lambda_n, v_n)$

will do. Let  $\mathbb{R}_*^n$  denote the set of nonzero vectors in  $\mathbb{R}^n$ . It is well known that the *Rayleigh quotient*,

$$r: \mathbb{R}_*^n \rightarrow \mathbb{R}: x \mapsto r(x) = \frac{\langle x, Ax \rangle}{\langle x, x \rangle},$$

attains its extreme values when  $x$  is aligned with  $\pm v_1$  or  $\pm v_n$ , and that the corresponding value of the quotient is  $\lambda_1$  or  $\lambda_n$ . We will rediscover such properties through the prism of optimization on manifolds as a running example in this book. One can gain some insight by checking that  $r(v_i) = \lambda_i$ .

Say we are interested in the smallest eigenvalue,  $\lambda_1$ . Then, we must solve the following optimization problem:

$$\min_{x \in \mathbb{R}_*^n} \frac{\langle x, Ax \rangle}{\langle x, x \rangle}.$$

The set  $\mathbb{R}_*^n$  is open in  $\mathbb{R}^n$ : it is an *open submanifold* of  $\mathbb{R}^n$ . Optimization over an open set has its challenges (more on this later). Fortunately, we can easily circumvent these issues in this instance.

Since the Rayleigh quotient is invariant to scaling, that is, since  $r(\alpha x) = r(x)$  for all nonzero real  $\alpha$ , we may fix the scale arbitrarily. Given the denominator of  $r$ , one particularly convenient way is to restrict our attention to unit-norm vectors:  $\|x\|^2 = \langle x, x \rangle = 1$ . The set of such vectors is the *unit sphere* in  $\mathbb{R}^n$ :

$$S^{n-1} = \{x \in \mathbb{R}^n : \|x\| = 1\}.$$

This is an *embedded submanifold* of  $\mathbb{R}^n$ . Our problem becomes:

$$\min_{x \in S^{n-1}} \langle x, Ax \rangle. \quad (2.1)$$

This is perhaps the simplest non-trivial instance of an optimization problem on a manifold: we use it recurrently to illustrate concepts as they occur.

Similarly to the above, we may compute the largest singular value of a matrix  $M \in \mathbb{R}^{m \times n}$  together with associated left- and right-singular vectors by solving

$$\max_{x \in S^{m-1}, y \in S^{n-1}} \langle x, My \rangle. \quad (2.2)$$

This is the basis of *principal component analysis*: see also Section 2.4. The search space is a Cartesian product of two spheres. This too is a manifold; specifically, an embedded submanifold of  $\mathbb{R}^m \times \mathbb{R}^n$ . In general:

*Products of manifolds are manifolds.*

This is an immensely useful property.

## 2.3 Dictionary learning: products of spheres

JPEG and its more recent version JPEG 2000 are some of the most commonly used compression standards for photographs. At their core, these algorithms rely

on basis expansions: discrete cosine transforms for JPEG, and wavelet transforms for JPEG 2000. That is, an image (or rather, each patch of the image) is written as a linear combination of a fixed collection of basis images. To fix notation, say an image is represented as a vector  $y \in \mathbb{R}^d$  (its pixels rearranged into a single column vector) and the basis images are  $b_1, \dots, b_d \in \mathbb{R}^d$  (each of unit norm). There exists a unique set of coordinates  $c \in \mathbb{R}^d$  such that:

$$y = c_1 b_1 + \dots + c_d b_d.$$

Since the basis images are fixed (and known to anyone creating or reading image files in this format), it is equivalent to store  $y$  or  $c$ .

The basis is designed carefully with two goals in mind. First, the transform between  $y$  and  $c$  should be fast to compute (one good starting point to that effect is orthogonality). Second, images encountered in practice should lead to many of the coefficients  $c_i$  being zero, or close to zero. Indeed, to recover  $y$ , it is only necessary to record the nonzero coefficients. To compress further, we may also decide not to store the small coefficients: if so,  $y$  can still be reconstructed approximately. Beyond compression, another benefit of sparse expansions is that they can reveal structural information about the contents of the image.

In *dictionary learning*, we focus on the second goal. As a key departure from the above, the idea here is not to design a basis by hand, but rather to learn a good basis from data automatically. This way, we may exploit structural properties of images that come up in a particular application. For example, it may be the case that photographs of faces can be expressed more sparsely in a dedicated basis as compared to a standard wavelet basis. Pushing this idea further, we relax the requirement of identifying a basis, instead allowing ourselves to pick more than  $d$  images for our expansions. The collection of images  $b_1, \dots, b_n \in \mathbb{R}^d$  forms a *dictionary*. Its elements are called *atoms*, and they normally span  $\mathbb{R}^d$  in an overcomplete way, meaning any image  $y$  can be expanded into a linear combination of atoms in more than one way. The aim is that at least one of these expansions should be sparse, or have many small coefficients. For the magnitudes of coefficients to be meaningful, we further require all atoms to have the same norm:  $\|b_i\| = 1$  for all  $i$ .

Thus, given a collection of  $k$  images  $y_1, \dots, y_k \in \mathbb{R}^d$ , the task in dictionary learning is to find atoms  $b_1, \dots, b_n \in \mathbb{R}^d$  such that (as much as possible) each image  $y_i$  is a sparse linear combination of the atoms. Collect the input images as the columns of a data matrix  $Y \in \mathbb{R}^{d \times k}$ , and the atoms into a matrix  $D \in \mathbb{R}^{d \times n}$  (to be determined). Expansion coefficients for the images in this dictionary form the columns of a matrix  $C \in \mathbb{R}^{n \times k}$  so that

$$Y = DC.$$

Typically, many choices of  $C$  are possible. We aim to pick  $D$  such that there exists a valid (or approximately valid) choice of  $C$  with numerous zeros. Let  $\|C\|_0$  denote the number of entries of  $C$  different from zero. Then, one possible formulation of dictionary learning balances both aims with a parameter  $\lambda > 0$

as (with  $b_1, \dots, b_n$  the columns of the dictionary matrix  $D$ ):

$$\begin{aligned} \min_{D \in \mathbb{R}^{d \times n}, C \in \mathbb{R}^{n \times k}} \|Y - DC\|^2 + \lambda \|C\|_0 \\ \text{subject to } \|b_1\| = \dots = \|b_n\| = 1. \end{aligned} \quad (2.3)$$

The matrix norm  $\|\cdot\|$  is the Frobenius norm, induced by the standard inner product  $\langle U, V \rangle = \text{Tr}(U^\top V)$ .

Evidently, allowing the dictionary to be overcomplete ( $n > d$ ) helps sparsity. An extreme case is to set  $n = k$ , in which case an optimal solution consists in letting  $D$  be  $Y$  with normalized columns. Then, each image can be expressed with a single nonzero coefficient ( $C$  is diagonal). This is useless of course, if only because both parties of the communication must have access to the (possibly huge) dictionary, and because this choice may generalize poorly when presented with new images. Interesting scenarios involve  $n$  much smaller than  $k$ .

The search space for  $D$  is a product of several spheres, which is an embedded submanifold of  $\mathbb{R}^{d \times n}$  called the *oblique manifold*:

$$\text{OB}(d, n) = (\text{S}^{d-1})^n = \{X \in \mathbb{R}^{d \times n} : \text{diag}(X^\top X) = \mathbf{1}\},$$

where  $\mathbf{1} \in \mathbb{R}^n$  is the all-ones vector and  $\text{diag}: \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^n$  extracts the diagonal entries of a matrix. The search space in  $C$  is the linear manifold  $\mathbb{R}^{n \times k}$ . Overall, the search space of the dictionary learning optimization problem is

$$\text{OB}(d, n) \times \mathbb{R}^{n \times k},$$

which is an embedded submanifold of  $\mathbb{R}^{d \times n} \times \mathbb{R}^{n \times k}$ .

We note in closing that the cost function in (2.3) is discontinuous because of the term  $\|C\|_0$ , making it hard to optimize. A standard reformulation replaces the culprit with  $\|C\|_1$ : the sum of absolute values of the entries of  $C$ . This is continuous but nonsmooth. A possible further step then is to smooth the cost function, for example exploiting that  $|x| \approx \sqrt{x^2 + \varepsilon^2}$  or  $|x| \approx \varepsilon \log(e^{x/\varepsilon} + e^{-x/\varepsilon})$  for small  $\varepsilon > 0$ : these are standard tricks.

Regardless of changes to the cost function, the manifold  $\text{OB}(d, n)$  is non-convex, so that finding a global optimum for dictionary learning as stated above is challenging: see work by Sun et al. [SQW17] for some guarantees.

## 2.4 Principal component analysis: Stiefel and Grassmann

Let  $x_1, \dots, x_n \in \mathbb{R}^d$  represent a large collection of centered data points in a  $d$ -dimensional linear space. We may think of it as a cloud of points. It may be the case that this cloud lies on or near a low-dimensional subspace of  $\mathbb{R}^d$ , and it may be distributed anisotropically in that subspace, meaning it shows more variation along some directions than others. One of the pillars of data analysis is to determine the main directions of variation of the data. This goes by the name of principal component analysis (PCA), which we encountered in Section 2.2.



## 2.4 Principal component analysis: Stiefel and Grassmann

9

One way to think of a main direction of variation, called a *principal component*, is as a vector  $u \in \mathbb{S}^{d-1}$  such that projecting the data points to the one-dimensional subspace spanned by  $u$  ‘preserves most of the variance.’ Specifically, let  $X \in \mathbb{R}^{d \times n}$  be the matrix whose columns are the data points and let  $uu^\top$  be the orthogonal projector from  $\mathbb{R}^d$  to the span of  $u$ . We wish to maximize the following for  $u \in \mathbb{S}^{d-1}$ :

$$\sum_{i=1}^n \|uu^\top x_i\|^2 = \|uu^\top X\|^2 = \langle X^\top uu^\top, X^\top uu^\top \rangle = \langle XX^\top u, u \rangle.$$

We recognize the Rayleigh quotient of the matrix  $XX^\top$  to be maximized for  $u$  over  $\mathbb{S}^{d-1}$  (Section 2.2). An optimal solution is given by a dominant eigenvector of  $XX^\top$ , or equivalently by a dominant left singular vector of  $X$ .

Let  $u_1 \in \mathbb{S}^{d-1}$  be a principal component. We would like to find a second one. That is, we aim to find  $u_2 \in \mathbb{S}^{d-1}$ , *orthogonal to  $u_1$* , such that projecting the data to the subspace spanned by  $u_1$  and  $u_2$  preserves the most variance. The orthogonal projector to that subspace is  $u_1 u_1^\top + u_2 u_2^\top$ . We maximize

$$\|(u_1 u_1^\top + u_2 u_2^\top)X\|^2 = \langle XX^\top u_1, u_1 \rangle + \langle XX^\top u_2, u_2 \rangle$$

over  $u_2 \in \mathbb{S}^{d-1}$  with  $u_2^\top u_1 = 0$ . The search space for  $u_2$  is an embedded submanifold of  $\mathbb{R}^d$ : it is a unit sphere in the subspace orthogonal to  $u_1$ .

It is often more convenient to optimize for  $u_1$  and  $u_2$  simultaneously rather than sequentially. Then, since the above cost function is symmetric in  $u_1$  and  $u_2$ , as is the constraint  $u_2^\top u_1 = 0$ , we add weights to the two terms to ensure  $u_1$  captures a principal component and  $u_2$  captures a second principal component:

$$\max_{u_1, u_2 \in \mathbb{S}^{d-1}, u_2^\top u_1 = 0} \alpha_1 \langle XX^\top u_1, u_1 \rangle + \alpha_2 \langle XX^\top u_2, u_2 \rangle,$$

with  $\alpha_1 > \alpha_2 > 0$  arbitrary.

More generally, aiming for  $k$  principal components, we look for a matrix  $U \in \mathbb{R}^{d \times k}$  with  $k$  orthonormal columns  $u_1, \dots, u_k \in \mathbb{R}^d$ . The set of such matrices is called the *Stiefel manifold*:

$$\text{St}(d, k) = \{U \in \mathbb{R}^{d \times k} : U^\top U = I_k\},$$

where  $I_k$  is the identity matrix of size  $k$ . It is an embedded submanifold of  $\mathbb{R}^{d \times k}$ . The orthogonal projector to the subspace spanned by the columns of  $U$  is  $UU^\top$ . Hence, PCA amounts to solving the problem:

$$\max_{U \in \text{St}(d, k)} \sum_{i=1}^k \alpha_i \langle XX^\top u_i, u_i \rangle = \max_{U \in \text{St}(d, k)} \langle XX^\top U, U D \rangle, \quad (2.4)$$

where  $D \in \mathbb{R}^{k \times k}$  is diagonal with diagonal entries  $\alpha_1 > \dots > \alpha_k > 0$ .

It is well known that collecting  $k$  top eigenvectors of  $XX^\top$  (or, equivalently,  $k$  top left singular vectors of  $X$ ) yields a global optimum of (2.4), meaning this optimization problem can be solved efficiently using tools from numerical linear

algebra. Still, the optimization perspective offers significant flexibility that standard linear algebra algorithms cannot match. Specifically, within an optimization framework, it is possible to revisit the variance criterion by changing the cost function. This allows one to promote sparsity or robustness against outliers, for example to develop variants such as sparse PCA [dBEG08, JNRS10] and robust PCA [MT11, GZAL14, MZL19, NNSS20]. There may also be computational advantages, for example in tracking and online models where the dataset changes or grows with time: it may be cheaper to update a previously computed good estimator using few optimization steps than to run a complete eigenvalue or singular value decomposition anew.

If the top  $k$  principal components are of interest but their ordering is not, then we do not need the weight matrix  $D$ . In this scenario, we are seeking an orthonormal basis  $U$  for a  $k$  dimensional subspace of  $\mathbb{R}^d$  such that projecting the data to that subspace preserves as much of the variance as possible. This description makes it clear that the particular basis is irrelevant: only the selected subspace matters. This is apparent in the cost function,

$$f(U) = \langle XX^\top U, U \rangle,$$

which is invariant under orthogonal transformations. Specifically, for all  $Q$  in the orthogonal group

$$O(k) = \{Q \in \mathbb{R}^{k \times k} : Q^\top Q = I_k\},$$

we have  $f(UQ) = f(U)$ . This induces an *equivalence relation*<sup>1</sup>  $\sim$  on the Stiefel manifold:

$$U \sim V \iff V = UQ \text{ for some } Q \in O(k).$$

This equivalence relation partitions  $\text{St}(d, k)$  into *equivalence classes*:

$$[U] = \{V \in \text{St}(d, k) : U \sim V\} = \{UQ : Q \in O(k)\}.$$

The set of equivalence classes is called the *quotient set*:

$$\text{St}(d, k)/\sim = \text{St}(d, k)/O(k) = \{[U] : U \in \text{St}(d, k)\}.$$

Importantly,  $U, V \in \text{St}(d, k)$  are equivalent if and only if their columns span the same subspace of  $\mathbb{R}^d$ . In other words: the quotient set is in one-to-one correspondence with the set of subspaces of dimension  $k$  in  $\mathbb{R}^d$ . With the right geometry, the latter is called the *Grassmann manifold*:

$$\text{Gr}(d, k) = \{ \text{subspaces of dimension } k \text{ in } \mathbb{R}^d \} \equiv \text{St}(d, k)/O(k),$$

where the symbol  $\equiv$  reads “is equivalent to” (context indicates in what sense). As defined here, the Grassmann manifold is a *quotient manifold*. This type of

<sup>1</sup> Recall that an equivalence relation  $\sim$  on a set  $M$  is a reflexive ( $a \sim a$ ), symmetric ( $a \sim b \iff b \sim a$ ) and transitive ( $a \sim b$  and  $b \sim c \implies a \sim c$ ) binary relation. The equivalence class  $[a]$  is the set of elements of  $M$  that are equivalent to  $a$ . Each element of  $M$  belongs to exactly one equivalence class.

manifold is more abstract than embedded submanifolds, but we can still develop numerically efficient tools to work with them.

Within our framework, computing the dominant eigenspace of dimension  $k$  of the matrix  $XX^\top$  can be written as:

$$\max_{[U] \in \text{Gr}(d,k)} \langle XX^\top U, U \rangle.$$

The cost function is well defined over  $\text{Gr}(d,k)$  since it depends only on the equivalence class of  $U$ , not on  $U$  itself.

Going back to (2.4), we note in passing that  $k$  top left and right singular vectors of a matrix  $M \in \mathbb{R}^{m \times n}$  can be computed by solving the following problem on a product of Stiefel manifolds (this and (2.4) are sometimes called *Brockett cost functions*):

$$\max_{U \in \text{St}(m,k), V \in \text{St}(n,k)} \langle MV, UD \rangle,$$

where  $D = \text{diag}(\alpha_1, \dots, \alpha_k)$  with arbitrary  $\alpha_1 > \dots > \alpha_k > 0$  as above.

A book by Trendafilov and Gallo provides more in-depth discussion of applications of optimization on manifolds to data analysis [TG21].

## 2.5 Synchronization of rotations: special orthogonal group

In structure from motion (SfM), the 3D structure of an object is to be reconstructed from several 2D images of it. For example, in the paper *Building Rome in a Day* [ASS<sup>+</sup>09], the authors automatically construct a model of the Colosseum from over 2000 photographs freely available on the Internet. Because the pictures are acquired from an unstructured source, one of the steps in the reconstruction pipeline is to estimate camera locations and *pose*. The pose of a camera is its orientation in space.

In single particle reconstruction through cryo electron microscopy, an electron microscope is used to produce 2D tomographic projections of biological objects such as proteins and viruses. Because shape is a determining factor of function, the goal is to estimate the 3D structure of the object from these projections. Contrary to X-ray crystallography (another fundamental tool of structural biology), the orientations of the objects in the individual projections are unknown. In order to estimate the structure, a useful step is to estimate the individual orientations (though high noise levels do not always allow it, in which case alternative statistical techniques must be used.)

Mathematically, orientations correspond to rotations of  $\mathbb{R}^3$ . Rotations in  $\mathbb{R}^d$  can be represented with orthogonal matrices:

$$\text{SO}(d) = \{R \in \mathbb{R}^{d \times d} : R^\top R = I_d \text{ and } \det(R) = +1\}.$$

The determinant condition excludes reflections of  $\mathbb{R}^d$ . The set  $\text{SO}(d)$  is the *special*

*orthogonal group*: it is both a group (in the mathematical sense of the term) and a manifold (an embedded submanifold of  $\mathbb{R}^{d \times d}$ )—it is a *Lie group*.

In both applications described above, similar images or projections can be compared to estimate relative orientations. *Synchronization of rotations* is a mathematical abstraction of the ensuing task. It consists in estimating  $n$  individual rotation matrices,

$$R_1, \dots, R_n \in \text{SO}(d),$$

from pairwise relative rotation measurements: for some pairs  $(i, j)$  corresponding to the edges of a graph  $G$ , we observe a noisy version of  $R_i R_j^{-1}$ . Let  $H_{ij} \in \text{SO}(d)$  denote such a measurement. Then, one possible formulation of synchronization of rotations is:

$$\min_{\hat{R}_1, \dots, \hat{R}_n \in \text{SO}(d)} \sum_{(i,j) \in G} \|\hat{R}_i - H_{ij} \hat{R}_j\|^2.$$

This is an optimization problem over  $\text{SO}(d)^n$ , which is a manifold.

This also comes up in *simultaneous localization and mapping* (SLAM), whereby a robot must simultaneously map its environment and locate itself in it as it moves around [RDTEL21]. An important aspect of SLAM is to keep track of the robot's orientation accurately, by integrating all previously acquired information to correct estimator drift.

## 2.6 Low-rank matrix completion: fixed-rank manifold

Let  $M \in \mathbb{R}^{m \times n}$  be a large matrix of interest. Given some of its entries, our task is to estimate the whole matrix. A commonly cited application for this setup is that of recommender systems, where row  $i$  corresponds to a user, column  $j$  corresponds to an item (a movie, a book...) and entry  $M_{ij}$  indicates how much user  $i$  appreciates item  $j$ : positive values indicate appreciation, zero is neutral, and negative values indicate dislike. The known entries may be collected from user interactions. Typically, most entries are unobserved. Predicting the missing values may be helpful to automate personalized recommendations.

Of course, without further knowledge about how the entries of the matrix are related, the completion task is ill-posed. Hope comes from the fact that certain users share similar traits, so that what one user likes may be informative about what another, similar user may like. In the same spirit, certain items may be similar enough that whole groups of users may feel similarly about them. One mathematically convenient way to capture this idea is to assume  $M$  has (approximately) low rank. The rationale is as follows: if  $M$  has rank  $r$ , then it can be factored as

$$M = LR^\top,$$

where  $L \in \mathbb{R}^{m \times r}$  and  $R \in \mathbb{R}^{n \times r}$  are full-rank factor matrices. Row  $i$  of  $L$ ,  $\ell_i$ ,

attributes  $r$  numbers to user  $i$ , while the  $j$ th row of  $R$ ,  $r_j$ , attributes  $r$  numbers to item  $j$ . Under the low-rank model, the rating of user  $i$  for item  $j$  is  $M_{ij} = \langle \ell_i, r_j \rangle$ . One interpretation is that there are  $r$  latent features (these could be movie genres for example): a user has some positive or negative appreciation for each feature, and an item has traits aligned with or in opposition to these features; the rating is obtained as the inner product of the two feature vectors.

Under this model, predicting the user ratings for all items amounts to *low-rank matrix completion*. Let  $\Omega$  denote the set of pairs  $(i, j)$  such that  $M_{ij}$  is observed. Allowing for noise in the observations and inaccuracies in the model, we aim to solve

$$\begin{aligned} \min_{X \in \mathbb{R}^{m \times n}} \sum_{(i,j) \in \Omega} (X_{ij} - M_{ij})^2 \\ \text{subject to } \text{rank}(X) = r. \end{aligned}$$

The search space for this optimization problem is the set of matrices of a given size and rank:

$$\mathbb{R}_r^{m \times n} = \{X \in \mathbb{R}^{m \times n} : \text{rank}(X) = r\}.$$

This set is an embedded submanifold of  $\mathbb{R}^{m \times n}$  which is frequently useful in machine learning applications.

Another use for this manifold is solving high-dimensional matrix equations that may come up in systems and control applications: aiming for a low-rank solution may be warranted in certain settings, and exploiting this can lower the computational burden substantially. Yet another context where optimization over low-rank matrices occurs is in completing and denoising approximately separable bivariate functions based on sampled values [Van10, Van13, MV13].

The same set can also be endowed with other geometries, that is, it can be made into a manifold in other ways. For example, exploiting the factored form more directly, note that any matrix in  $\mathbb{R}_r^{m \times n}$  admits a factorization as  $LR^\top$  with both  $L$  and  $R$  of full rank  $r$ . This correspondence is not one-to-one however, since the pairs  $(L, R)$  and  $(LJ^{-1}, RJ^\top)$  map to the same matrix in  $\mathbb{R}_r^{m \times n}$  for all invertible matrices  $J$ : they are equivalent. Similarly to the Grassmann manifold, this leads to a definition of  $\mathbb{R}_r^{m \times n}$  as a quotient manifold instead of an embedded submanifold. Many variations on this theme are possible, some of them more useful than others depending on the application [Mey11, Mis14].

The set  $\mathbb{R}_r^{m \times n}$  is not closed in  $\mathbb{R}^{m \times n}$ , which may create difficulties for optimization. The closure of the set corresponds to all matrices of rank at most  $r$  (rather than exactly equal to  $r$ ). That set is not a manifold, but it can be smoothly parameterized by a manifold in several ways [LKB22b]. One particularly simple way is through the map  $(L, R) \mapsto LR^\top$  where  $L$  and  $R$  are allowed to be rank deficient.

## 2.7 Gaussian mixture models: positive definite matrices

A common model in machine learning assumes data  $x_1, \dots, x_n \in \mathbb{R}^d$  are sampled independently from a *mixture of  $K$  Gaussians*, that is, each data point is sampled from a probability distribution with density of the form

$$f(x) = \sum_{k=1}^K w_k \frac{1}{\sqrt{2\pi \det(\Sigma_k)}} e^{-\frac{(x-\mu_k)^\top \Sigma_k^{-1} (x-\mu_k)}{2}},$$

where the centers  $\mu_1, \dots, \mu_K \in \mathbb{R}^d$ , covariances  $\Sigma_1, \dots, \Sigma_K \in \text{Sym}(d)^+$  and mixing probabilities  $(w_1, \dots, w_K) \in \Delta_+^{K-1}$  are to be determined. We use the following notation:

$$\text{Sym}(d)^+ = \{X \in \mathbb{R}^{d \times d} : X = X^\top \text{ and } X \succ 0\}$$

for symmetric, positive definite matrices of size  $d$ , and

$$\Delta_+^{K-1} = \{w \in \mathbb{R}^K : w_1, \dots, w_K > 0 \text{ and } w_1 + \dots + w_K = 1\}$$

for the positive part of the simplex, that is, the set of non-vanishing discrete probability distributions over  $K$  objects. In this model, with probability  $w_k$ , a point  $x$  is sampled from the  $k$ th Gaussian, with mean  $\mu_k$  and covariance  $\Sigma_k$ . The aim is only to estimate the parameters, not to estimate which Gaussian each point  $x_i$  was sampled from.

For a given set of observations  $x_1, \dots, x_n$ , a maximum likelihood estimator solves:

$$\max_{\substack{\hat{\mu}_1, \dots, \hat{\mu}_K \in \mathbb{R}^d, \\ \hat{\Sigma}_1, \dots, \hat{\Sigma}_K \in \text{Sym}(d)^+, \\ w \in \Delta_+^{K-1}}} \sum_{i=1}^n \log \left( \sum_{k=1}^K w_k \frac{1}{\sqrt{2\pi \det(\Sigma_k)}} e^{-\frac{(x_i - \mu_k)^\top \Sigma_k^{-1} (x_i - \mu_k)}{2}} \right). \quad (2.5)$$

This is an optimization problem over  $\mathbb{R}^{d \times K} \times (\text{Sym}(d)^+)^K \times \Delta_+^{K-1}$ , which can be made into a manifold because  $\text{Sym}(d)^+$  and  $\Delta_+^{K-1}$  can be given a manifold structure.

The direct formulation of maximum likelihood estimation for Gaussian mixture models in (2.5) is, however, not computationally favorable. See [HS15] for a beneficial reformulation, still on a manifold.

## 2.8 Smooth semidefinite programs

*Semidefinite programs* (SDPs) are optimization problems of the form

$$\min_{X \in \text{Sym}(n)} \langle C, X \rangle \quad \text{subject to} \quad \mathcal{A}(X) = b \quad \text{and} \quad X \succeq 0, \quad (2.6)$$

where  $\text{Sym}(n)$  is the space of real, symmetric matrices of size  $n \times n$ ,  $\langle A, B \rangle = \text{Tr}(A^\top B)$ ,  $\mathcal{A}: \text{Sym}(n) \rightarrow \mathbb{R}^m$  is a linear map defined by  $m$  symmetric matrices  $A_1, \dots, A_m$  as  $\mathcal{A}(X)_i = \langle A_i, X \rangle$ , and  $X \succeq 0$  means  $X$  is positive semidefinite.

SDPs are convex and they can be solved to global optimality in polynomial time using interior point methods [Nes18, §5.4.4]. Still, handling the positive semidefiniteness constraint  $X \succeq 0$  and the dimensionality of the problem (namely, the  $\frac{n(n+1)}{2}$  variables required to define  $X$ ) both pose significant computational challenges for large  $n$ .

A popular way to address both issues is the Burer–Monteiro approach [BM05], which consists in factorizing  $X$  as  $X = YY^\top$  with  $Y \in \mathbb{R}^{n \times p}$ : the number  $p$  of columns of  $Y$  is a parameter. Notice that  $X$  is now automatically positive semidefinite. If  $p \geq n$ , the SDP can be rewritten equivalently as

$$\min_{Y \in \mathbb{R}^{n \times p}} \langle CY, Y \rangle \quad \text{subject to} \quad \mathcal{A}(YY^\top) = b. \quad (2.7)$$

If  $p < n$ , this corresponds to the SDP with the additional constraint  $\text{rank}(X) \leq p$ . There is a computational advantage to taking  $p$  as small as possible. Interestingly, if the set of matrices  $X$  that are feasible for the SDP is compact, then the *Pataki–Barvinok bound* [Pat98, Bar95] provides that at least one of the global optimizers of the SDP has rank  $r$  such that  $\frac{r(r+1)}{2} \leq m$ . In other words: assuming compactness, the Burer–Monteiro formulation (2.7) is *equivalent* to the original SDP so long as  $p$  satisfies  $\frac{p(p+1)}{2} \geq m$ . This is already the case for  $p = O(\sqrt{m})$ , which may be significantly smaller than  $n$ .

The positive semidefiniteness constraint disappeared, and the dimensionality of the problem went from  $O(n^2)$  to  $np$ —a potentially appreciable gain. Yet, we lost something important along the way: the Burer–Monteiro problem is not convex. It is not immediately clear how to solve it.

The search space of the Burer–Monteiro problem is the set of feasible  $Y$ :

$$\mathcal{M} = \{Y \in \mathbb{R}^{n \times p} : \mathcal{A}(YY^\top) = b\}. \quad (2.8)$$

Assume the map  $Y \mapsto \mathcal{A}(YY^\top)$  has the property that its differential at all  $Y$  in  $\mathcal{M}$  has rank  $m$ . Then,  $\mathcal{M}$  is an embedded submanifold of  $\mathbb{R}^{n \times p}$ . In this special case, we may try to solve the Burer–Monteiro problem through optimization over that manifold. It turns out that non-convexity is mostly benign in that scenario, in a precise sense [BVB19]:

*If  $\mathcal{M}$  is compact and  $\frac{p(p+1)}{2} > m$ , then, for a generic cost matrix  $C$ , the smooth optimization problem  $\min_{Y \in \mathcal{M}} \langle CY, Y \rangle$  has no spurious local minima, in the sense that any point  $Y$  which satisfies first- and second-order necessary optimality conditions is a global optimum.*

(Necessary optimality conditions are detailed in Sections 4.2 and 6.1.) Additionally, these global optima map to global optima of the SDP through  $X = YY^\top$ . This suggests that smooth-and-compact SDPs may be solved to global optimality via optimization on manifolds. The requirement that  $\mathcal{M}$  be a regularly defined smooth manifold is not innocuous, but it is satisfied in a number of interesting applications.

There has been a lot of work on this front in recent years, including the early

work by Burer and Monteiro [BM03, BM05], the first manifold-inspired perspective by Journée et al. [JBAS10], qualifications of the benign non-convexity at the Pataki–Barvinok threshold [BVB16, BVB19] and below in special cases [BBV16], a proof that  $p$  cannot be set much lower than that threshold in general [WW20], smoothed analyses to assess whether points which satisfy necessary optimality conditions approximately are also approximately optimal [BBJN18, PJB18, CM19] and extensions to accommodate scenarios where  $\mathcal{M}$  is not a smooth manifold but, more generally, a real algebraic variety [BBJN18, Cif21]. See all these references for applications, including Max-Cut, community detection, the trust-region subproblem, synchronization of rotations and more.



## 3 Embedded geometry: first order

---

Our goal is to develop optimization algorithms to solve problems of the form

$$\min_{x \in \mathcal{M}} f(x), \quad (3.1)$$

where  $\mathcal{M}$  is a smooth and possibly nonlinear space, and  $f: \mathcal{M} \rightarrow \mathbb{R}$  is a smooth cost function. In order to do so, our first task is to clarify what we mean by a “smooth space,” and a “smooth function” on such a space. Then, we need to develop any tools required to construct optimization algorithms in this setting. Let us start with a bird’s-eye view of what this entails, and formalize later on.

For smoothness of  $\mathcal{M}$ , our model space is the unit sphere in  $\mathbb{R}^d$ :

$$\mathbb{S}^{d-1} = \{x \in \mathbb{R}^d : x^\top x = 1\}. \quad (3.2)$$

Intuitively, we think of  $\mathbb{S}^{d-1}$  as a smooth nonlinear space in  $\mathbb{R}^d$ . Our definitions below are compatible with this intuition: we call  $\mathbb{S}^{d-1}$  an *embedded submanifold* of  $\mathbb{R}^d$ .

An important element in these definitions is to capture the idea that  $\mathbb{S}^{d-1}$  can be locally approximated by a linear space around any point  $x$ : we call these *tangent spaces*, denoted by  $T_x \mathbb{S}^{d-1}$ . This is as opposed to a cube for which no good linearization exists at the edges. More specifically for our example,  $\mathbb{S}^{d-1}$  is defined by the constraint  $x^\top x = 1$ . We may expect that differentiating this constraint should yield a suitable linearization, and indeed it does:

$$T_x \mathbb{S}^{d-1} = \{v \in \mathbb{R}^d : v^\top x + x^\top v = 0\} = \{v \in \mathbb{R}^d : x^\top v = 0\}. \quad (3.3)$$

In the same spirit, it stands to reason that linear spaces and open subsets of linear spaces should also be considered smooth, as they are locally linear too.

We say a function from  $\mathbb{R}^d$  to  $\mathbb{R}$  is smooth if it is infinitely differentiable. We may expect that a function  $f: \mathbb{S}^{d-1} \rightarrow \mathbb{R}$  obtained by restriction to  $\mathbb{S}^{d-1}$  of a smooth function on  $\mathbb{R}^d$  ought to be considered smooth. We adopt (essentially) this as our definition of smooth functions on  $\mathbb{S}^{d-1}$ .

In this early chapter, we give a restricted definition of smoothness, focusing on embedded submanifolds. This allows us to build our initial toolbox more rapidly, and is sufficient to handle many applications. We extend our perspective to the general framework later on, in Chapter 8.

To get started with a list of required tools, it is useful to review briefly the

main ingredients of optimization on a *linear* space  $\mathcal{E}$ :

$$\min_{x \in \mathcal{E}} f(x). \quad (3.4)$$

For example,  $\mathcal{E} = \mathbb{R}^d$  or  $\mathcal{E} = \mathbb{R}^{n \times p}$ . Perhaps the most fundamental algorithm to address this class of problems is *gradient descent*, also known as *steepest descent*. Given an initial guess  $x_0 \in \mathcal{E}$ , this algorithm produces *iterates* on  $\mathcal{E}$  (a sequence of points on  $\mathcal{E}$ ) as follows:<sup>1</sup>

$$x_{k+1} = x_k - \alpha_k \text{grad} f(x_k), \quad k = 0, 1, 2, \dots \quad (3.5)$$

where the  $\alpha_k > 0$  are aptly chosen step-sizes and  $\text{grad} f: \mathcal{E} \rightarrow \mathcal{E}$  is the gradient of  $f$ . Under mild assumptions, the accumulation points of the sequence  $x_0, x_1, x_2, \dots$  have relevant properties for the optimization problem (3.4). We study these later, in Chapter 4.

From this discussion, we can identify a list of desiderata for a geometric toolbox, meant to solve

$$\min_{x \in S^{d-1}} f(x) \quad (3.6)$$

with some smooth function  $f$  on the sphere. The most pressing point is to find an alternative for the implicit use of linearity in (3.5). Indeed, above, both  $x_k$  and  $\text{grad} f(x_k)$  are elements of  $\mathcal{E}$ . Since  $\mathcal{E}$  is a linear space, they can be combined with linear operations. Putting aside for now the issue of defining a proper notion of gradient for a function  $f$  on  $S^{d-1}$ , we must still contend with the issue that  $S^{d-1}$  is *not* a linear space: we have no notion of linear combination here.

Alternatively, we can reinterpret iteration (3.5) and say:

*To produce  $x_{k+1} \in S^{d-1}$ , move away from  $x_k$  along the direction  $-\text{grad} f(x_k)$  over some distance, while remaining on  $S^{d-1}$ .*

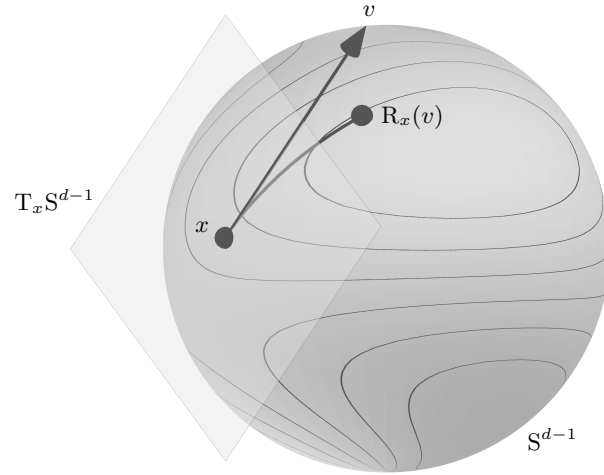
Surely, if the purpose is to remain on  $S^{d-1}$ , it would make little sense to move radially away from the sphere. Rather, using the notion that smooth spaces can be linearized around  $x$  by a tangent space  $T_x S^{d-1}$ , we only consider moving along directions in  $T_x S^{d-1}$ . To this end, we introduce the concept of *retraction* at  $x$ : a map  $R_x: T_x S^{d-1} \rightarrow S^{d-1}$  which allows us to move away from  $x$  smoothly along a prescribed tangent direction while remaining on the sphere. One suggestion might be as follows, with  $\|u\| = \sqrt{u^T u}$  (see Figure 3.1):

$$R_x(v) = \frac{x + v}{\|x + v\|}. \quad (3.7)$$

In this chapter, we give definitions that allow for this natural proposal.

It remains to make sense of the notion of gradient for a function on a smooth, nonlinear space. Once more, we take inspiration from the linear case. For a smooth function  $f: \mathcal{E} \rightarrow \mathbb{R}$ , the gradient is defined with respect to an *inner*

<sup>1</sup> Here,  $x_k$  designates an element in a sequence  $x_0, x_1, \dots$ . Sometimes, we also use subscript notation such as  $x_i$  to select the  $i$ th entry of a column vector  $x$ . Context tells which is meant.



**Figure 3.1** Retraction  $R_x(v) = \frac{x+v}{\|x+v\|}$  on the sphere.

product  $\langle \cdot, \cdot \rangle : \mathcal{E} \times \mathcal{E} \rightarrow \mathbb{R}$  (see Definition 3.1 below for a reminder):  $\text{grad} f(x)$  is the unique element of  $\mathcal{E}$  such that, for all  $v \in \mathcal{E}$ ,

$$Df(x)[v] = \langle v, \text{grad} f(x) \rangle, \quad (3.8)$$

where  $Df(x) : \mathcal{E} \rightarrow \mathbb{R}$  is the differential of  $f$  at  $x$ , that is, it is the linear map defined by:

$$Df(x)[v] = \lim_{t \rightarrow 0} \frac{f(x + tv) - f(x)}{t}. \quad (3.9)$$

Crucially, the gradient of  $f$  depends on a choice of inner product (while the differential of  $f$  does not).

For example, on  $\mathcal{E} = \mathbb{R}^d$  equipped with the standard inner product

$$\langle u, v \rangle = u^\top v \quad (3.10)$$

and the canonical basis  $e_1, \dots, e_d \in \mathbb{R}^d$  (the columns of the identity matrix), the  $i$ th coordinate of  $\text{grad} f(x) \in \mathbb{R}^d$  is given by

$$\begin{aligned} \text{grad} f(x)_i &= \langle e_i, \text{grad} f(x) \rangle = Df(x)[e_i] \\ &= \lim_{t \rightarrow 0} \frac{f(x + te_i) - f(x)}{t} \triangleq \frac{\partial f}{\partial x_i}(x), \end{aligned} \quad (3.11)$$

that is, the  $i$ th partial derivative of  $f$  as a function of  $x_1, \dots, x_d \in \mathbb{R}$ . This covers a case so common that it is sometimes presented as the definition of the gradient:

$$\text{grad} f = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \cdots & \frac{\partial f}{\partial x_d} \end{bmatrix}^\top.$$

Turning to our nonlinear example again, in order to define a proper notion of gradient for  $f : S^{d-1} \rightarrow \mathbb{R}$ , we find that we need to (a) provide a meaningful

notion of differential  $Df(x): T_x S^{d-1} \rightarrow \mathbb{R}$ , and (b) introduce inner products on the tangent spaces of  $S^{d-1}$ . Let us focus on the latter for this outline.

Since  $T_x S^{d-1}$  is a different linear subspace for various  $x \in S^{d-1}$ , we need a different inner product for each point:  $\langle \cdot, \cdot \rangle_x$  denotes our choice of inner product on  $T_x S^{d-1}$ . If this choice of inner products varies smoothly with  $x$  (in a sense we make precise below), then we call it a *Riemannian metric*, and  $S^{d-1}$  equipped with this metric is called a *Riemannian manifold*. This allows us to define the *Riemannian gradient* of  $f$  on  $S^{d-1}$ :  $\text{grad} f(x)$  is the unique tangent vector at  $x$  such that, for all  $v \in T_x S^{d-1}$ ,

$$Df(x)[v] = \langle v, \text{grad} f(x) \rangle_x.$$

Thus, first we choose a Riemannian metric, then a notion of gradient ensues.

One arguably natural way of endowing  $S^{d-1}$  with a metric is to exploit the fact that each tangent space  $T_x S^{d-1}$  is a linear subspace of  $\mathbb{R}^d$ , so that we may define  $\langle \cdot, \cdot \rangle_x$  by restricting the inner product of  $\mathbb{R}^d$  (3.10) to  $T_x S^{d-1}$ : for all  $u, v \in T_x S^{d-1}$ ,  $\langle u, v \rangle_x = \langle u, v \rangle$ . This is indeed a Riemannian metric, and  $S^{d-1}$  endowed with this metric is called a *Riemannian submanifold* of  $\mathbb{R}^d$ .

For Riemannian submanifolds, the Riemannian gradient is particularly simple to compute. As per our definitions,  $f: S^{d-1} \rightarrow \mathbb{R}$  is smooth if and only if there exists a function  $\bar{f}: \mathbb{R}^d \rightarrow \mathbb{R}$ , smooth in the usual sense, such that  $f$  and  $\bar{f}$  coincide on  $S^{d-1}$ . We will argue that

$$\text{grad} f(x) = \text{Proj}_x(\text{grad} \bar{f}(x)), \quad \text{with} \quad \text{Proj}_x(v) = (I_d - xx^\top)v,$$

where  $\text{Proj}_x: \mathbb{R}^d \rightarrow T_x S^{d-1}$  is the orthogonal projector from  $\mathbb{R}^d$  to  $T_x S^{d-1}$  (orthogonal with respect to the inner product on  $\mathbb{R}^d$ .) The functions  $f$  and  $\bar{f}$  often have the same analytical expression. For example,  $f(x) = x^\top A x$  (for some matrix  $A \in \mathbb{R}^{d \times d}$ ) is smooth on  $S^{d-1}$  because  $\bar{f}(x) = x^\top A x$  is smooth on  $\mathbb{R}^d$  and they coincide on  $S^{d-1}$ . To summarize:

*For Riemannian submanifolds, the Riemannian gradient is the orthogonal projection of the “classical” gradient to the tangent spaces.*

With these tools in place, we can justify the following algorithm, an instance of *Riemannian gradient descent* on  $S^{d-1}$ . Given  $x_0 \in S^{d-1}$ , let

$$\begin{aligned} x_{k+1} &= R_{x_k}(-\alpha_k \text{grad} f(x_k)), & \text{with} & \quad \text{grad} f(x) = (I_d - xx^\top) \text{grad} \bar{f}(x), \\ & & \text{and} & \quad R_x(v) = \frac{x + v}{\|x + v\|}, \end{aligned}$$

where  $\bar{f}$  is a smooth extension of  $f$  to  $\mathbb{R}^d$ . More importantly, these tools give us a formal framework to design and analyze such algorithms on a large class of smooth, nonlinear spaces.

We now proceed to construct precise definitions, starting with a few reminders of linear algebra and multivariate calculus in linear spaces.

### 3.1 Reminders of Euclidean space

The letter  $\mathcal{E}$  always denotes a *linear space* (or *vector space*) over the reals, that is, a set equipped with (and closed under) vector addition and scalar multiplication by real numbers. Frequent examples include  $\mathbb{R}^d$  (column vectors of length  $d$ ),  $\mathbb{R}^{n \times p}$  (matrices of size  $n \times p$ ),  $\text{Sym}(n)$  (real, symmetric matrices of size  $n$ ),  $\text{Skew}(n)$  (real, skew-symmetric matrices of size  $n$ ), and their (linear) subspaces.

We write  $\text{span}(u_1, \dots, u_m)$  to denote the subspace of  $\mathcal{E}$  *spanned* by vectors  $u_1, \dots, u_m \in \mathcal{E}$ . By extension,  $\text{span}(X)$  for a matrix  $X \in \mathbb{R}^{n \times m}$  denotes the subspace of  $\mathbb{R}^n$  spanned by the columns of  $X$ .

Given two linear spaces  $\mathcal{E}$  and  $\mathcal{F}$ , a *linear map* or *linear operator* is a map  $\mathcal{L}: \mathcal{E} \rightarrow \mathcal{F}$  such that  $\mathcal{L}(au + bv) = a\mathcal{L}(u) + b\mathcal{L}(v)$  for all  $u, v \in \mathcal{E}$  and  $a, b \in \mathbb{R}$ . We let  $\text{im } \mathcal{L}$  denote the *image* (or the *range*) of  $\mathcal{L}$ , and we let  $\text{ker } \mathcal{L}$  denote the *kernel* (or *null space*) of  $\mathcal{L}$ .

A *basis* for  $\mathcal{E}$  is a set of vectors (elements of  $\mathcal{E}$ )  $e_1, \dots, e_d$  such that each vector  $x \in \mathcal{E}$  can be expressed uniquely as a linear combination  $x = a_1e_1 + \dots + a_de_d$  with real coefficients  $a_1, \dots, a_d$ . All bases have the same number of elements, called the *dimension* of  $\mathcal{E}$  ( $\dim \mathcal{E} = d$ ): it is always finite in our treatment. Each basis induces a one-to-one linear map identifying  $\mathcal{E}$  and  $\mathbb{R}^d$  to each other: we write  $\mathcal{E} \equiv \mathbb{R}^d$ .

#### *Topology.*

Recall that a *topology* on a set is a collection of subsets called *open* such that (a) the whole set and the empty set are open, (b) any union of opens is open, and (c) the intersection of a finite number of opens is open—more on this in Section 8.2. A subset is *closed* if its complement is open. A subset may be open, closed, both, or neither. We always equip  $\mathbb{R}^d$  with its usual topology [Lee12, Ex. A.6]. Each linear space  $\mathcal{E}$  of dimension  $d$  inherits the topology of  $\mathbb{R}^d$  through their identification as above. A *neighborhood* of  $x$  in  $\mathcal{E}$  is an open subset of  $\mathcal{E}$  which contains  $x$ . Some authors call such sets *open neighborhoods*. All our neighborhoods are open, hence we omit the qualifier.

#### *Euclidean structure.*

It is useful to endow  $\mathcal{E}$  with more structure.

**Definition 3.1.** An inner product on  $\mathcal{E}$  is a function  $\langle \cdot, \cdot \rangle : \mathcal{E} \times \mathcal{E} \rightarrow \mathbb{R}$  with the following properties. For all  $u, v, w \in \mathcal{E}$  and  $a, b \in \mathbb{R}$ , we have:

1. *Symmetry:*  $\langle u, v \rangle = \langle v, u \rangle$ ;
2. *Linearity:*  $\langle au + bv, w \rangle = a \langle u, w \rangle + b \langle v, w \rangle$ ; and
3. *Positive definiteness:*  $\langle u, u \rangle \geq 0$  and  $\langle u, u \rangle = 0 \iff u = 0$ .

Two vectors  $u, v$  are *orthogonal* if  $\langle u, v \rangle = 0$ .

**Definition 3.2.** A linear space  $\mathcal{E}$  with an inner product  $\langle \cdot, \cdot \rangle$  is a Euclidean space. An inner product induces a norm on  $\mathcal{E}$  called the Euclidean norm:

$$\|u\| = \sqrt{\langle u, u \rangle}. \quad (3.12)$$

**Definition 3.3.** A basis  $u_1, \dots, u_d$  of a Euclidean space  $\mathcal{E}$  is orthonormal if

$$\forall 1 \leq i, j \leq d, \quad \langle u_i, u_j \rangle = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

The standard inner product on  $\mathbb{R}^d$  and the associated norm are:

$$\langle u, v \rangle = u^\top v = \sum_i u_i v_i, \quad \|u\| = \sqrt{\sum_i u_i^2}. \quad (3.13)$$

Similarly, the standard inner product on linear spaces of matrices such as  $\mathbb{R}^{n \times p}$  and  $\text{Sym}(n)$  is the so-called *Frobenius inner product*, with its associated *Frobenius norm*:

$$\langle U, V \rangle = \text{Tr}(U^\top V) = \sum_{ij} U_{ij} V_{ij}, \quad \|U\| = \sqrt{\sum_{ij} U_{ij}^2}, \quad (3.14)$$

where  $\text{Tr}(M) = \sum_i M_{ii}$  is the *trace* of a matrix. Summations are over all entries. When we do not specify it, we mean to use the standard inner product and norm.

We often use the following properties of the above inner product, with matrices  $U, V, W, A, B$  of compatible sizes:

$$\begin{aligned} \langle U, V \rangle &= \langle U^\top, V^\top \rangle, & \langle UA, V \rangle &= \langle U, VA^\top \rangle, \\ \langle BU, V \rangle &= \langle U, B^\top V \rangle, & \langle U \odot W, V \rangle &= \langle U, V \odot W \rangle, \end{aligned} \quad (3.15)$$

where  $\odot$  denotes entrywise multiplication (*Hadamard product*).

Although we only consider linear spaces over the reals, we can still handle complex matrices. For example,  $\mathbb{C}^n$  is a real linear space of dimension  $2n$ . The standard basis for it is  $e_1, \dots, e_n, ie_1, \dots, ie_n$ , where  $e_1, \dots, e_n$  form the standard basis of  $\mathbb{R}^n$  (the columns of the identity matrix of size  $n$ ), and  $i$  is the imaginary unit. Indeed, any vector in  $\mathbb{C}^n$  can be written uniquely as a linear combination of those basis vectors using real coefficients. The standard inner product and norm on  $\mathbb{C}^n$  as a real linear space are:

$$\langle u, v \rangle = \Re\{u^* v\} = \Re\left\{\sum_k \bar{u}_k v_k\right\}, \quad \|u\| = \sqrt{\sum_k |u_k|^2}, \quad (3.16)$$

where  $u^*$  is the Hermitian conjugate-transpose of  $u$ ,  $\bar{u}_k$  is the complex conjugate of  $u_k$ ,  $|u_k|$  is its magnitude and  $\Re\{a\}$  is the real part of  $a$ . This perspective is equivalent to identifying  $\mathbb{C}^n$  with  $\mathbb{R}^{2n}$ , where real and imaginary parts are considered as two vectors in  $\mathbb{R}^n$ . Likewise, the set of complex matrices  $\mathbb{C}^{n \times p}$  is a real linear space of dimension  $2np$ , with the following standard inner product

and norm:

$$\langle U, V \rangle = \Re\{\text{Tr}(U^*V)\} = \Re\left\{\sum_{k\ell} \bar{U}_{k\ell} V_{k\ell}\right\}, \quad \|U\| = \sqrt{\sum_{k\ell} |U_{k\ell}|^2}. \quad (3.17)$$

The analog of (3.15) in the complex case is:

$$\begin{aligned} \langle U, V \rangle &= \langle U^*, V^* \rangle, & \langle UA, V \rangle &= \langle U, VA^* \rangle, \\ \langle BU, V \rangle &= \langle U, B^*V \rangle, & \langle U \odot W, V \rangle &= \langle U, V \odot \bar{W} \rangle. \end{aligned} \quad (3.18)$$

A linear map between two Euclidean spaces has a unique *adjoint*, which we now define. These are often useful in deriving gradients of functions—more on this in Section 4.7.

**Definition 3.4.** Let  $\mathcal{E}$  and  $\mathcal{F}$  be two Euclidean spaces, with inner products  $\langle \cdot, \cdot \rangle_{\mathcal{E}}$  and  $\langle \cdot, \cdot \rangle_{\mathcal{F}}$ , respectively. Let  $\mathcal{L}: \mathcal{E} \rightarrow \mathcal{F}$  be a linear map. The adjoint of  $\mathcal{L}$  is the linear map  $\mathcal{L}^*: \mathcal{F} \rightarrow \mathcal{E}$  defined by the following property:

$$\forall u \in \mathcal{E}, v \in \mathcal{F}, \quad \langle \mathcal{L}(u), v \rangle_{\mathcal{F}} = \langle u, \mathcal{L}^*(v) \rangle_{\mathcal{E}}.$$

**Definition 3.5.** Let  $\mathcal{E}$  be a Euclidean space with inner product  $\langle \cdot, \cdot \rangle$ . If the linear map  $\mathcal{A}: \mathcal{E} \rightarrow \mathcal{E}$  satisfies

$$\forall u, v \in \mathcal{E}, \quad \langle \mathcal{A}(u), v \rangle = \langle u, \mathcal{A}(v) \rangle,$$

that is, if  $\mathcal{A} = \mathcal{A}^*$ , we say  $\mathcal{A}$  is self-adjoint or symmetric.

As we can see from (3.15) and (3.18), adjoints and matrix transposes are intimately related: it is an exercise to make this precise.

Self-adjoint linear maps have important spectral properties.

**Theorem 3.6** (Spectral theorem). A self-adjoint map  $\mathcal{A}$  on a Euclidean space  $\mathcal{E}$  admits an orthonormal basis of eigenvectors  $v_1, \dots, v_d \in \mathcal{E}$  associated to real eigenvalues  $\lambda_1, \dots, \lambda_d$  so that  $\mathcal{A}(v_i) = \lambda_i v_i$  for  $i = 1, \dots, d$  with  $d = \dim \mathcal{E}$ .

**Definition 3.7.** A self-adjoint map  $\mathcal{A}$  on  $\mathcal{E}$  is positive semidefinite if, for all  $u \in \mathcal{E}$ , we have  $\langle u, \mathcal{A}(u) \rangle \geq 0$ ; we write  $\mathcal{A} \succeq 0$ . Owing to the spectral theorem, this is equivalent to all eigenvalues of  $\mathcal{A}$  being nonnegative. Similarly,  $\mathcal{A}$  is positive definite if  $\langle u, \mathcal{A}(u) \rangle > 0$  for all nonzero  $u \in \mathcal{E}$ ; we write  $\mathcal{A} \succ 0$ . This is equivalent to all eigenvalues of  $\mathcal{A}$  being positive.

Norms on vector spaces induce norms for linear maps.

**Definition 3.8.** The operator norm of  $\mathcal{L}: \mathcal{E} \rightarrow \mathcal{F}$  is defined as

$$\|\mathcal{L}\| = \max_{u \in \mathcal{E}, u \neq 0} \frac{\|\mathcal{L}(u)\|_{\mathcal{F}}}{\|u\|_{\mathcal{E}}},$$

where  $\|\cdot\|_{\mathcal{E}}$  and  $\|\cdot\|_{\mathcal{F}}$  denote the norms on the Euclidean spaces  $\mathcal{E}$  and  $\mathcal{F}$ , respectively.

Equivalently,  $\|\mathcal{L}\|$  is the smallest real such that  $\|\mathcal{L}(u)\|_{\mathcal{F}} \leq \|\mathcal{L}\| \|u\|_{\mathcal{E}}$  for all  $u$  in  $\mathcal{E}$ . The *singular values* of  $\mathcal{L}$  are the nonnegative square roots of the eigenvalues of  $\mathcal{L}^* \circ \mathcal{L}$ , and  $\|\mathcal{L}\|$  is the largest singular value of  $\mathcal{L}$ . For a self-adjoint map  $\mathcal{A}$  with eigenvalues  $\lambda_1, \dots, \lambda_d$ , it follows that  $\|\mathcal{A}\| = \max_{1 \leq i \leq d} |\lambda_i|$ .

### Calculus.

We write  $F: A \rightarrow B$  to designate a map  $F$  whose domain is all of  $A$ . If  $C$  is a subset of  $A$ , we write  $F|_C: C \rightarrow B$  to designate the *restriction* of  $F$  to the domain  $C$ , so that  $F|_C(x) = F(x)$  for all  $x \in C$ .

Let  $U, V$  be open sets in two linear spaces  $\mathcal{E}, \mathcal{F}$ . A map  $F: U \rightarrow V$  is *smooth* if it is infinitely differentiable (class  $C^\infty$ ) on its domain. We also say that  $F$  is *smooth at a point*  $x \in U$  if there exists a neighborhood  $U'$  of  $x$  such that  $F|_{U'}$  is smooth. Accordingly,  $F$  is smooth if it is smooth at all points in its domain.

If  $F: U \rightarrow V$  is smooth at  $x$ , the *differential* of  $F$  at  $x$  is the linear map  $DF(x): \mathcal{E} \rightarrow \mathcal{F}$  defined by

$$DF(x)[u] = \left. \frac{d}{dt} F(x + tu) \right|_{t=0} = \lim_{t \rightarrow 0} \frac{F(x + tu) - F(x)}{t}. \quad (3.19)$$

For a curve  $c: \mathbb{R} \rightarrow \mathcal{E}$ , we write  $c'$  to denote its velocity:  $c'(t) = \frac{d}{dt} c(t)$ .

For a smooth function  $f: \mathcal{E} \rightarrow \mathbb{R}$  defined on a Euclidean space  $\mathcal{E}$ , the (Euclidean) *gradient* of  $f$  is the map  $\text{grad} f: \mathcal{E} \rightarrow \mathcal{E}$  defined by the following property:

$$\forall x, v \in \mathcal{E}, \quad \langle \text{grad} f(x), v \rangle = Df(x)[v].$$

The (Euclidean) *Hessian* of  $f$  at  $x$  is the linear map  $\text{Hess} f(x): \mathcal{E} \rightarrow \mathcal{E}$  defined by

$$\text{Hess} f(x)[v] = D(\text{grad} f)(x)[v] = \lim_{t \rightarrow 0} \frac{\text{grad} f(x + tv) - \text{grad} f(x)}{t}.$$

The *Clairaut–Schwarz theorem* implies that  $\text{Hess} f(x)$  is self-adjoint with respect to the inner product of  $\mathcal{E}$ .

**Exercise 3.9** (Adjoint and transpose). *Let  $u_1, \dots, u_n$  form an orthonormal basis of  $\mathcal{E}$ . Likewise, let  $v_1, \dots, v_m$  form an orthonormal basis of  $\mathcal{F}$ . Consider a linear map  $\mathcal{L}: \mathcal{E} \rightarrow \mathcal{F}$ . For each  $1 \leq i \leq n$ , the vector  $\mathcal{L}(u_i)$  is an element of  $\mathcal{F}$ ; therefore, it expands uniquely in the basis  $v_1, \dots, v_m$  as follows:*

$$\mathcal{L}(u_i) = \sum_{j=1}^m M_{ji} v_j,$$

where we collect the coefficients into a matrix  $M \in \mathbb{R}^{m \times n}$ . This matrix represents  $\mathcal{L}$  with respect to the chosen bases. Show that the matrix which represents  $\mathcal{L}^*$  with respect to those same bases is  $M^\top$ : the transpose of  $M$ . In particular, a linear map  $\mathcal{A}: \mathcal{E} \rightarrow \mathcal{E}$  is self-adjoint if and only if the matrix associated to it with respect to the basis  $u_1, \dots, u_n$  is symmetric.



## 3.2 Embedded submanifolds of a linear space

We set out to define what it means for a subset  $\mathcal{M}$  of a linear space  $\mathcal{E}$  to be smooth. Our main angle is to capture the idea that a smooth set can be linearized in some meaningful way around each point. To make sense of what that might mean, consider the sphere  $S^{d-1}$ . This is the set of vectors  $x \in \mathbb{R}^d$  satisfying

$$h(x) = x^\top x - 1 = 0.$$

As we discussed in the introduction, it can be adequately linearized around each point by the set (3.3). The perspective we used to obtain this linearization is that of differentiating the defining equation. More precisely, consider a truncated Taylor expansion of  $h$ :

$$h(x + tv) = h(x) + t Dh(x)[v] + O(t^2).$$

If  $x$  is in  $S^{d-1}$  and  $v$  is in  $\ker Dh(x)$  (so that  $h(x) = 0$  and  $Dh(x)[v] = 0$ ), then  $h(x + tv) = O(t^2)$ , indicating that  $x + tv$  nearly satisfies the defining equation of  $S^{d-1}$  for small  $t$ . This motivates us to consider the subspace  $\ker Dh(x)$  as a linearization of  $S^{d-1}$  around  $x$ . Since

$$Dh(x)[v] = \lim_{t \rightarrow 0} \frac{h(x + tv) - h(x)}{t} = x^\top v + v^\top x = 2x^\top v,$$

the kernel of  $Dh(x)$  is the subspace orthogonal to  $x$  in  $\mathbb{R}^d$  (with respect to the usual inner product). This coincides with (3.3), arguably in line with intuition.

At first, one might think that if a set is defined by an equation of the form  $h(x) = 0$  with some smooth function  $h$ , then that set is smooth and can be linearized by the kernels of  $Dh$ . However, this is not the case. Indeed, consider the following example in  $\mathbb{R}^2$  (see Figure 3.2):

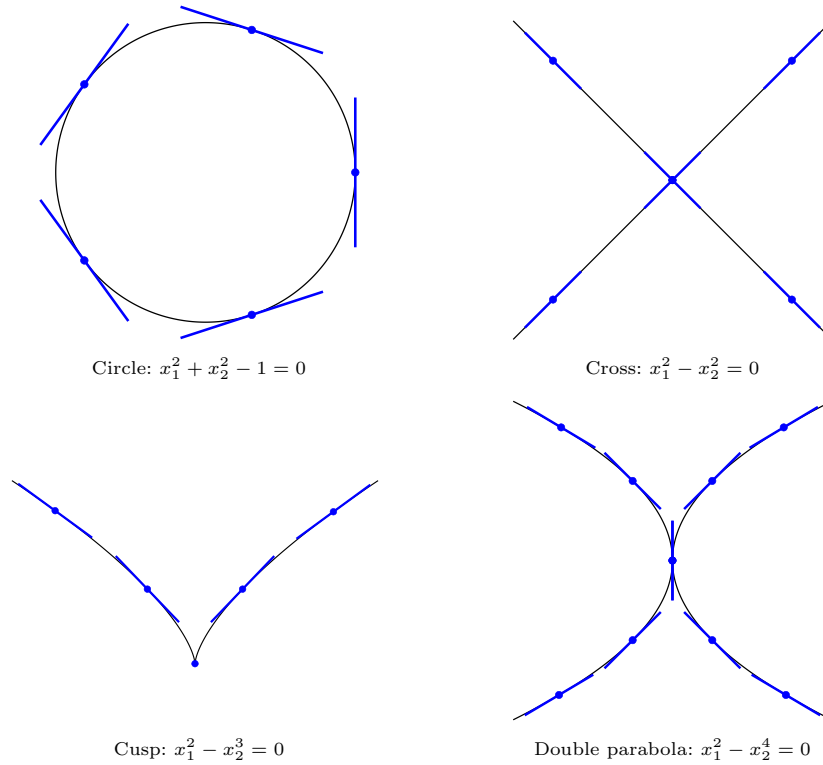
$$\mathcal{X} = \{x \in \mathbb{R}^2 : h(x) = x_1^2 - x_2^2 = 0\}.$$

The defining function  $h$  is smooth, yet the set  $\mathcal{X}$  is a cross in the plane formed by the union of the lines  $x_1 = x_2$  and  $x_1 = -x_2$ . We want to exclude such sets because of the kink at the origin. What went wrong with it? If we blindly use the kernel of the differential to linearize  $\mathcal{X}$ , we first determine

$$Dh(x) = \left[ \frac{\partial h}{\partial x_1}(x), \frac{\partial h}{\partial x_2}(x) \right] = [2x_1, -2x_2].$$

At  $x = 0$ ,  $Dh(0) = [0, 0]$ , whose kernel is all of  $\mathbb{R}^2$ : that does not constitute a reasonable linearization of  $\mathcal{X}$  around the origin.

We can gain further insight into the issue at hand by considering additional examples. The zero-sets of the functions  $h(x) = x_1^2 - x_2^3$  and  $h(x) = x_1^2 - x_2^4$  from  $\mathbb{R}^2$  to  $\mathbb{R}$ , respectively, define a cusp and a double parabola, both of which fail our intuitive test of smoothness at the origin. What the cross, cusp and double parabola have in common is that the rank of  $Dh(x)$  suddenly drops from one to zero at the origin, whereas for the sphere that rank is constant (and maximal) on the whole set.



**Figure 3.2** Four different sets  $\mathcal{S}$  defined as the zero-sets of a smooth function from  $\mathbb{R}^2$  to  $\mathbb{R}$ . For each, the sets  $T_x\mathcal{S}$  (Definition 3.14) are drawn at a few different points. Only the circle (top left) is an embedded submanifold of  $\mathbb{R}^2$ .

These observations motivate the definition below. Since smoothness is a local notion, the definition is phrased in terms of what the set  $\mathcal{M}$  looks like around each point. Since a set  $\mathcal{M}$  may be equivalently defined by many different functions  $h$ , and since it may not be practical (or even possible, see Section 3.10) to define all of  $\mathcal{M}$  with a single function  $h$ , the definition allows for a different one to be used around each point.

**Definition 3.10.** Let  $\mathcal{E}$  be a linear space of dimension  $d$ . A non-empty subset  $\mathcal{M}$  of  $\mathcal{E}$  is a (smooth) embedded submanifold of  $\mathcal{E}$  of dimension  $n$  if either

1.  $n = d$  and  $\mathcal{M}$  is open in  $\mathcal{E}$ —we also call this an open submanifold; or
2.  $n = d - k$  for some  $k \geq 1$  and, for each  $x \in \mathcal{M}$ , there exists a neighborhood  $U$  of  $x$  in  $\mathcal{E}$  and a smooth function  $h: U \rightarrow \mathbb{R}^k$  such that
  - (a) If  $y$  is in  $U$ , then  $h(y) = 0$  if and only if  $y \in \mathcal{M}$ ; and
  - (b)  $\text{rank } Dh(x) = k$ .

Such a function  $h$  is called a local defining function for  $\mathcal{M}$  at  $x$ .

If  $\mathcal{M}$  is a linear (sub)space, we also call it a linear manifold.

Condition 2(a) in the previous definition can be stated equivalently as:

$$\mathcal{M} \cap U = h^{-1}(0) \triangleq \{y \in U : h(y) = 0\}.$$

It is an exercise to verify that Definition 3.10 excludes various pathological sets such as the cross ( $x_1^2 = x_2^2$ ), cusp ( $x_1^2 = x_2^3$ ) and double parabola ( $x_1^2 = x_2^4$ ).

Differential geometry defines a broader class of smooth sets called (*smooth*) *manifolds*. We typically omit the word ‘smooth’ as all of our manifolds are smooth, though bear in mind that in the literature there exist different kinds of manifolds, not all of which are smooth. Embedded submanifolds are manifolds. When the statements we make hold true for smooth manifolds in general, we use the word manifold (rather than embedded submanifold) to signal it. This is common throughout Chapters 3 and 5. ★

The hope is that limiting our initial treatment of manifolds to embedded submanifolds provides a more intuitive entry point to build all the tools we need for optimization. This is all the more relevant considering that many of the manifolds we encounter in applications are in fact embedded submanifolds, presented to us as zero-sets of their local defining functions. All of our optimization algorithms work on general manifolds. The general theory is in Chapter 8.

To build additional support for our definition of embedded submanifolds, we further argue that small patches of  $\mathcal{M}$  can be deformed into linear subspaces in a smooth and smoothly invertible way. This captures an important feature of smoothness, namely: upon zooming close to a point of  $\mathcal{M}$ , what we see can hardly be distinguished from what we would have seen had  $\mathcal{M}$  been a linear subspace of  $\mathcal{E}$ .

**Definition 3.11.** A diffeomorphism is a bijective map  $F: U \rightarrow V$ , where  $U, V$  are open sets and such that both  $F$  and  $F^{-1}$  are smooth.

**Theorem 3.12.** Let  $\mathcal{E}$  be a linear space of dimension  $d$ . A subset  $\mathcal{M}$  of  $\mathcal{E}$  is an embedded submanifold of  $\mathcal{E}$  of dimension  $n = d - k$  if and only if for each  $x \in \mathcal{M}$  there exists a neighborhood  $U$  of  $x$  in  $\mathcal{E}$ , an open set  $V$  in  $\mathbb{R}^d$  and a diffeomorphism  $F: U \rightarrow V$  such that  $F(\mathcal{M} \cap U) = E \cap V$ , where  $E = \{y \in \mathbb{R}^d : y_{n+1} = \cdots = y_d = 0\}$  is a linear subspace of  $\mathbb{R}^d$ .

The main tool we need to prove Theorem 3.12 is the standard *inverse function theorem*, stated here without proof [Lee12, Thm. C.34].

**Theorem 3.13** (Inverse function theorem). Suppose  $U \subseteq \mathcal{E}$  and  $V \subseteq \mathcal{F}$  are open subsets of linear spaces of the same dimension, and  $F: U \rightarrow V$  is smooth. If  $DF(x)$  is invertible at some point  $x \in U$ , then there exist neighborhoods  $U' \subseteq U$  of  $x$  and  $V' \subseteq V$  of  $F(x)$  such that  $F|_{U'}: U' \rightarrow V'$  (the restriction of  $F$  to  $U'$  and  $V'$ ) is a diffeomorphism.

Equipped with this tool, we proceed to prove Theorem 3.12.

*Proof of Theorem 3.12.* We prove one direction of the theorem, namely: we assume  $\mathcal{M}$  is an embedded submanifold and construct diffeomorphisms  $F$ . The

other direction is left as an exercise. For the latter, it is helpful to note that if  $F$  is a diffeomorphism with inverse  $F^{-1}$ , then  $DF(x)$  is invertible and

$$(DF(x))^{-1} = DF^{-1}(F(x)). \quad (3.20)$$

To see this, apply the chain rule to differentiate  $F^{-1} \circ F$ , noting that this is nothing but the identity map.

If  $n = d$  (that is,  $\mathcal{M}$  is open in  $\mathcal{E}$ ), the claim is clear: simply let  $F$  be any invertible linear map from  $\mathcal{E}$  to  $\mathbb{R}^d$  (for example, using a basis of  $\mathcal{E}$ ), and restrict its domain and codomain to  $U = \mathcal{M}$  and  $V = F(U)$ .

We now consider the more interesting case where  $n = d - k$  with  $k \geq 1$ . Let  $h: U \rightarrow \mathbb{R}^k$  be any local defining function for  $\mathcal{M}$  at  $x$ . We work in coordinates on  $\mathcal{E}$ , which is thus identified with  $\mathbb{R}^d$ . Then, we can think of  $Dh(x)$  as a matrix of size  $k \times d$ . By assumption,  $Dh(x)$  has rank  $k$ . This means that it is possible to pick  $k$  columns of that matrix which form a  $k \times k$  invertible matrix. If needed, permute the chosen coordinates so that the last  $k$  columns have that property (this is without loss of generality). Then, we can write  $Dh(x)$  in block form so that

$$Dh(x) = \begin{bmatrix} A & B \end{bmatrix},$$

where  $B \in \mathbb{R}^{k \times k}$  is invertible and  $A$  is in  $\mathbb{R}^{k \times (d-k)}$ . Now consider the function  $F: U \rightarrow \mathbb{R}^d$  (recall  $U \subseteq \mathcal{E}$  is the domain of  $h$ ) defined by

$$F(y) = (y_1, \dots, y_{d-k}, h_1(y), \dots, h_k(y))^T, \quad (3.21)$$

where  $y_1, \dots, y_d$  denote the coordinates of  $y \in \mathcal{E}$ . In order to apply the inverse function theorem to  $F$  at  $x$ , we must verify that  $F$  is smooth—this is clear—and that the differential of  $F$  at  $x$  is invertible. Working this out one row at a time, we get the following expression for that differential:

$$DF(x) = \begin{bmatrix} I_{d-k} & 0 \\ A & B \end{bmatrix},$$

where  $I_{d-k}$  is the identity matrix of size  $d-k$ , and  $0$  here denotes a zero matrix of size  $(d-k) \times k$ . The matrix  $DF(x)$  is invertible, as demonstrated by the following expression for its inverse:

$$(DF(x))^{-1} = \begin{bmatrix} I_{d-k} & 0 \\ -B^{-1}A & B^{-1} \end{bmatrix}. \quad (3.22)$$

(Indeed, their product is  $I_d$ .) Hence, the inverse function theorem asserts that we may reduce  $U$  to a possibly smaller neighborhood of  $x$  so that  $F$  (now restricted to that new neighborhood) is a diffeomorphism from  $U$  to  $V = F(U)$ . The property  $F(\mathcal{M} \cap U) = E \cap V$  follows by construction of  $F$  from the property  $\mathcal{M} \cap U = h^{-1}(0)$ .  $\square$

In order to understand the local geometry of a set around a point, we aim to describe acceptable directions of movement through that point. This is close in spirit to the tools we look to develop for optimization, as they involve moving

away from a point while remaining on the set. Specifically, for a subset  $\mathcal{M}$  of a linear space  $\mathcal{E}$ , consider all the smooth curves of  $\mathcal{E}$  which lie entirely on  $\mathcal{M}$  and pass through a given point  $x$ . Collect their velocities as they do so in a set  $T_x\mathcal{M}$  defined below. In that definition,  $c$  is smooth in the usual sense as a map from (an open subset of)  $\mathbb{R}$  to  $\mathcal{E}$ —two linear spaces.

**Definition 3.14.** Let  $\mathcal{M}$  be a subset of  $\mathcal{E}$ . For all  $x \in \mathcal{M}$ , define:

$$T_x\mathcal{M} = \{c'(0) \mid c: I \rightarrow \mathcal{M} \text{ is smooth and } c(0) = x\}, \quad (3.23)$$

where  $I$  is any open interval containing  $t = 0$ . That is,  $v$  is in  $T_x\mathcal{M}$  if and only if there exists a smooth curve on  $\mathcal{M}$  passing through  $x$  with velocity  $v$ .

Note that  $T_x\mathcal{M}$  is a subset of  $\mathcal{E}$ . For the sphere, it is easy to convince oneself that  $T_x\mathcal{M}$  coincides with the subspace in (3.3). We show in the next theorem that this is always the case for embedded submanifolds.

**Theorem 3.15.** Let  $\mathcal{M}$  be an embedded submanifold of  $\mathcal{E}$ . Consider  $x \in \mathcal{M}$  and the set  $T_x\mathcal{M}$  (3.23). If  $\mathcal{M}$  is an open submanifold, then  $T_x\mathcal{M} = \mathcal{E}$ . Otherwise,  $T_x\mathcal{M} = \ker Dh(x)$  with  $h$  any local defining function at  $x$ .

*Proof.* For open submanifolds, the claim is clear. By definition,  $T_x\mathcal{M}$  is included in  $\mathcal{E}$ . The other way around, for any  $v \in \mathcal{E}$ , consider  $c(t) = x + tv$ : this is a smooth curve from some non-empty interval around 0 to  $\mathcal{M}$  such that  $c(0) = x$ , hence  $v = c'(0)$  is in  $T_x\mathcal{M}$ . This shows  $\mathcal{E}$  is included in  $T_x\mathcal{M}$ , so that the two coincide.

Now consider the case of  $\mathcal{M}$  an embedded submanifold of dimension  $n = d - k$  with  $k \geq 1$ . Let  $h: U \rightarrow \mathbb{R}^k$  be a local defining function of  $\mathcal{M}$  around  $x$ . The proof is in two steps. First, we show that  $T_x\mathcal{M}$  is included in  $\ker Dh(x)$ . Then, we show that  $T_x\mathcal{M}$  contains a linear subspace of the same dimension as  $\ker Dh(x)$ . These two facts combined indeed confirm that  $T_x\mathcal{M} = \ker Dh(x)$ .

**Step 1.** If  $v$  is in  $T_x\mathcal{M}$ , there exists  $c: I \rightarrow \mathcal{M}$ , smooth, such that  $c(0) = x$  and  $c'(0) = v$ . Since  $c(t)$  is in  $\mathcal{M}$ , we have  $h(c(t)) = 0$  for all  $t \in I$  (if need be, restrict the interval  $I$  to ensure  $c(t)$  remains in the domain of  $h$ ). Thus, the derivative of  $h \circ c$  vanishes at all times:

$$0 = \frac{d}{dt}h(c(t)) = Dh(c(t))[c'(t)].$$

In particular, at  $t = 0$  this implies  $Dh(x)[v] = 0$ , that is,  $v \in \ker Dh(x)$ . This confirms  $T_x\mathcal{M} \subseteq \ker Dh(x)$ .

**Step 2.** To show that  $T_x\mathcal{M}$  contains a subspace of the same dimension as  $\ker Dh(x)$  (namely, of dimension  $n = d - k$ ), we must construct smooth curves on  $\mathcal{M}$  that pass through  $x$  with various velocities. To do so, we call upon Theorem 3.12. The latter provides us with a diffeomorphism  $F: U \rightarrow V$  (where  $U$  is now a possibly smaller neighborhood of  $x$  than the original domain of  $h$ .) We use  $F^{-1}$  to construct smooth curves on  $\mathcal{M}$  that pass through  $x$ . Specifically, pick an

arbitrary  $u \in \mathbb{R}^{d-k}$  and let

$$\gamma(t) = F(x) + t \begin{bmatrix} u \\ 0 \end{bmatrix}.$$

(Here, 0 denotes a zero vector of size  $k$ .) Notice that  $\gamma$  remains in  $E \cap V$  for  $t$  close to zero, where  $E$  is the subspace of  $\mathbb{R}^d$  consisting of all vectors whose last  $k$  entries are zero. Since  $F^{-1}(E \cap V) = \mathcal{M} \cap U$ , it follows that

$$c(t) = F^{-1}(\gamma(t)) \quad (3.24)$$

resides in  $\mathcal{M}$  for  $t$  close to zero. Moreover,  $c(0) = x$  and  $c$  is smooth since  $F^{-1}$  is smooth. It follows that  $c$  is indeed a smooth curve on  $\mathcal{M}$  passing through  $x$ . What is the velocity of this curve at  $x$ ? Applying the chain rule to (3.24), we get

$$c'(t) = DF^{-1}(\gamma(t)) [\gamma'(t)].$$

In particular, at  $t = 0$  we have

$$c'(0) = DF^{-1}(F(x)) \begin{bmatrix} u \\ 0 \end{bmatrix}.$$

Since  $F$  is a diffeomorphism, we know from (3.20) that  $DF^{-1}(F(x))$  is an invertible linear map, equal to  $(DF(x))^{-1}$ . The specific form of  $c'(0)$  is unimportant. What matters is that each  $c'(0)$  of the form above certainly belongs to  $T_x\mathcal{M}$  (3.23). Since  $DF^{-1}(F(x))$  is invertible and  $u \in \mathbb{R}^{d-k}$  is arbitrary, this means that  $T_x\mathcal{M}$  contains a subspace of dimension  $d-k$ . But we know from the previous step that  $T_x\mathcal{M}$  is included in a subspace of dimension  $d-k$ , namely,  $\ker Dh(x)$ . It follows that  $T_x\mathcal{M} = \ker Dh(x)$ . Since this holds for all  $x \in \mathcal{M}$ , the proof is complete.  $\square$

Thus, for an embedded submanifold  $\mathcal{M}$  of dimension  $n = d - k$ , for each  $x$  in  $\mathcal{M}$ , the set  $T_x\mathcal{M}$  is a linear subspace of  $\mathcal{E}$  of dimension  $n$ . These subspaces are the linearizations of the smooth set  $\mathcal{M}$ .

**Definition 3.16.** We call  $T_x\mathcal{M}$  the tangent space to  $\mathcal{M}$  at  $x$ . Vectors in  $T_x\mathcal{M}$  are called tangent vectors to  $\mathcal{M}$  at  $x$ . The dimension of  $T_x\mathcal{M}$  (which is independent of  $x$ ) coincides with the dimension of  $\mathcal{M}$ , denoted by  $\dim \mathcal{M}$ .

We consider three brief examples of embedded submanifolds: two obvious by now, and one arguably less obvious. It is good to keep all three in mind when assessing whether a certain proposition concerning embedded submanifolds is likely to be true. Chapter 7 details further examples.

**Example 3.17.** The set  $\mathbb{R}^d$  is a linear manifold of dimension  $d$  with tangent spaces  $T_x\mathcal{M} = \mathbb{R}^d$  for all  $x \in \mathbb{R}^d$ . The affine space  $\{x \in \mathbb{R}^d : Ax = b\}$  defined by a matrix  $A \in \mathbb{R}^{k \times d}$  of rank  $k$  and arbitrary vector  $b \in \mathbb{R}^k$  is a manifold of dimension  $n = d - k$  embedded in  $\mathbb{R}^d$ .

### 3.2 Embedded submanifolds of a linear space

31

**Example 3.18.** The sphere  $S^{d-1} = \{x \in \mathbb{R}^d : x^\top x = 1\}$  is the zero level set of  $h(x) = x^\top x - 1$ , smooth from  $\mathbb{R}^d$  to  $\mathbb{R}$ . Since  $Dh(x)[v] = 2x^\top v$ , it is clear that  $\text{rank } Dh(x) = 1$  for all  $x \in S^{d-1}$ . As a result,  $S^{d-1}$  is an embedded submanifold of  $\mathbb{R}^d$  of dimension  $n = d - 1$ . Furthermore, its tangent spaces are given by  $T_x S^{d-1} = \ker Dh(x) = \{v \in \mathbb{R}^d : x^\top v = 0\}$ .

**Example 3.19.** Let  $\text{Sym}(2)_1$  denote the set of symmetric matrices of size two and rank one, that is,

$$\text{Sym}(2)_1 = \left\{ X = \begin{bmatrix} x_1 & x_2 \\ x_2 & x_3 \end{bmatrix} : \text{rank } X = 1 \right\}.$$

This is a subset of  $\text{Sym}(2)$ , the linear space of symmetric matrices of size two. The rank function is not a smooth map (it is not even continuous), hence we cannot use it as a local defining function. Nevertheless, we can construct local defining functions for  $\text{Sym}(2)_1$ . Indeed, a matrix of size  $2 \times 2$  has rank one if and only if it is nonzero and its determinant is zero, hence:

$$\text{Sym}(2)_1 = \left\{ X = \begin{bmatrix} x_1 & x_2 \\ x_2 & x_3 \end{bmatrix} : x_1 x_3 - x_2^2 = 0 \text{ and } X \neq 0 \right\}.$$

Let  $U = \text{Sym}(2) \setminus \{0\}$  be the open subset of  $\text{Sym}(2)$  obtained by removing the zero matrix. Consider  $h: U \rightarrow \mathbb{R}$  defined by  $h(X) = x_1 x_3 - x_2^2$ . Clearly,  $h$  is smooth and  $h^{-1}(0) = \text{Sym}(2)_1 \cap U = \text{Sym}(2)_1$ . Furthermore,

$$Dh(X)[\dot{X}] = \dot{x}_1 x_3 + x_1 \dot{x}_3 - 2x_2 \dot{x}_2 = \begin{bmatrix} x_3 & -2x_2 & x_1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix},$$

where  $\dot{X}$  is a matrix in  $\text{Sym}(2)$ : the dot is a visual indication that we should think of  $\dot{X}$  as a perturbation of  $X$ . This linear map has rank one provided  $X \neq 0$ , which is always the case in the domain of  $h$ . Hence,  $h$  is a defining function for  $\text{Sym}(2)_1$  around any  $X$  in  $\text{Sym}(2)_1$ . This confirms that the latter is an embedded submanifold of  $\text{Sym}(2)$  of dimension  $\dim \text{Sym}(2) - 1 = 3 - 1 = 2$ . Its tangent space at  $X$  is given by  $\ker Dh(X)$ :

$$T_X \text{Sym}(2)_1 = \left\{ \dot{X} = \begin{bmatrix} \dot{x}_1 & \dot{x}_2 \\ \dot{x}_2 & \dot{x}_3 \end{bmatrix} : \dot{x}_1 x_3 + x_1 \dot{x}_3 - 2x_2 \dot{x}_2 = 0 \right\}.$$

Contrary to the two previous examples, this manifold is neither open nor closed in its embedding space. It is also not connected. Visualized in  $\mathbb{R}^3$ , it corresponds to a double, infinite elliptic cone. Indeed,  $X \neq 0$  is in  $\text{Sym}(2)_1$  if and only if

$$(x_1 + x_3)^2 = (2x_2)^2 + (x_1 - x_3)^2.$$

After the linear change of variables  $z_1 = x_1 - x_3$ ,  $z_2 = 2x_2$  and  $z_3 = x_1 + x_3$ , the defining equation becomes  $z_1^2 + z_2^2 = z_3^2$ , omitting the origin.

We can combine manifolds to form new ones. For instance, it is an exercise to show that Cartesian products of manifolds are manifolds.<sup>2</sup>

**Proposition 3.20.** *Let  $\mathcal{M}, \mathcal{M}'$  be embedded submanifolds of  $\mathcal{E}, \mathcal{E}'$  (respectively). Then,  $\mathcal{M} \times \mathcal{M}'$  is an embedded submanifold of  $\mathcal{E} \times \mathcal{E}'$  of dimension  $\dim \mathcal{M} + \dim \mathcal{M}'$  with tangent spaces given by:*

$$T_{(x,x')}(\mathcal{M} \times \mathcal{M}') = T_x \mathcal{M} \times T_{x'} \mathcal{M}'.$$

For example, after showing that the sphere  $S^{d-1}$  is an embedded submanifold of  $\mathbb{R}^d$ , it follows that  $S^{d-1} \times \cdots \times S^{d-1} = (S^{d-1})^k$  is an embedded submanifold of  $(\mathbb{R}^d)^k \equiv \mathbb{R}^{d \times k}$ : it is called the *oblique manifold*  $\text{OB}(d, k)$ .

In closing this section, we equip embedded submanifolds of  $\mathcal{E}$  with the topology induced by  $\mathcal{E}$ .<sup>3</sup> Having a topology notably allows us to define notions such as local optima and convergence of sequences on  $\mathcal{M}$ . Both are useful when studying iterative optimization algorithms.

**Definition 3.21.** *A subset  $\mathcal{U}$  of  $\mathcal{M}$  is open (resp., closed) in  $\mathcal{M}$  if  $\mathcal{U}$  is the intersection of  $\mathcal{M}$  with an open (resp., closed) subset of  $\mathcal{E}$ . This is called the subspace topology.*

Echoing the conventions laid out in Section 3.1, our neighborhoods are open.

**Definition 3.22.** *A neighborhood of  $x$  in  $\mathcal{M}$  is an open subset of  $\mathcal{M}$  which contains  $x$ . By extension, a neighborhood of a subset of  $\mathcal{M}$  is an open set of  $\mathcal{M}$  which contains that subset.*

It is an exercise to show that open subsets of a manifold  $\mathcal{M}$  are manifolds; we call them *open submanifolds* of  $\mathcal{M}$ .

**Proposition 3.23.** *Let  $\mathcal{M}$  be an embedded submanifold of  $\mathcal{E}$ . Any open subset of  $\mathcal{M}$  is also an embedded (but not necessarily open) submanifold of  $\mathcal{E}$ , with same dimension and tangent spaces as  $\mathcal{M}$ .*

**Exercise 3.24.** *Complete the proof of Theorem 3.12.*

**Exercise 3.25.** *Give a proof of Proposition 3.20.*

**Exercise 3.26.** *Give a proof of Proposition 3.23. In particular, deduce that the relative interior of the simplex, that is,*

$$\Delta_+^{d-1} = \{x \in \mathbb{R}^d : x_1 + \cdots + x_d = 1 \text{ and } x_1, \dots, x_d > 0\}, \quad (3.25)$$

*is an embedded submanifold of  $\mathbb{R}^d$ . This is useful to represent non-vanishing discrete probability distributions. See also Exercise 3.41.*

<sup>2</sup> We collect facts about product manifolds along the way: see Table 7.2 for a summary.

<sup>3</sup> About terminology: the general definition of submanifolds allows for other topologies. The qualifier ‘embedded’ (some say ‘regular’) indicates we use the induced topology. More on this in Section 8.14.



**Exercise 3.27.** Show that the cross  $\mathcal{X} = \{x \in \mathbb{R}^2 : x_1^2 = x_2^2\}$  is not an embedded submanifold. It is not sufficient to show that  $x \mapsto x_1^2 - x_2^2$  is not a local defining function at the origin: it is necessary to show that no local defining function exists at that point. Hint: proceeding by contradiction, assume there exists a local defining function around the origin and show that its kernel is too large.

**Exercise 3.28.** Show that the cusp  $\mathcal{C} = \{x \in \mathbb{R}^2 : x_1^2 = x_2^3\}$  is not an embedded submanifold. Hint: argue that  $T_0\mathcal{C}$  (as defined by (3.23)) is too low-dimensional.

**Exercise 3.29.** Show that the double parabola  $\mathcal{P} = \{x \in \mathbb{R}^2 : x_1^2 = x_2^4\}$  is not an embedded submanifold, yet  $T_x\mathcal{P}$  as defined by (3.23) is a linear subspace of dimension one in  $\mathbb{R}^2$  for all  $x \in \mathcal{P}$ . This example shows that Definition 3.10 is more restrictive than just requiring all sets  $T_x\mathcal{P}$  to be subspaces of the same dimension. Hint: proceeding by contradiction, assume  $\mathcal{P}$  is an embedded submanifold and call upon Theorem 3.12 to construct a special diffeomorphism  $F$ ; then, derive a contradiction from the fact that  $\mathcal{P}$  around the origin does not look like a one-dimensional curve. Specifically, you may want to use the fact that it is impossible to have three or more disjoint open intervals of the real line sharing a common accumulation point.

### 3.3 Smooth maps on embedded submanifolds

Now that we have a notion of smooth sets, we can introduce the all important notion of smooth maps between smooth sets. It relies heavily on the classical notion of smooth maps between (open subsets of) linear spaces. In optimization, two examples of maps between manifolds are cost functions ( $\mathcal{M} \rightarrow \mathbb{R}$ ) and iteration maps ( $\mathcal{M} \rightarrow \mathcal{M}$ ); more will come up.

**Definition 3.30.** Let  $\mathcal{M}$  and  $\mathcal{M}'$  be embedded submanifolds of  $\mathcal{E}$  and  $\mathcal{E}'$  (respectively). A map  $F: \mathcal{M} \rightarrow \mathcal{M}'$  is smooth at  $x \in \mathcal{M}$  if there exists a function  $\bar{F}: U \rightarrow \mathcal{E}'$  which is smooth on a neighborhood  $U$  of  $x$  in  $\mathcal{E}$  and such that  $F$  and  $\bar{F}$  coincide on  $\mathcal{M} \cap U$ , that is,  $F(y) = \bar{F}(y)$  for all  $y \in \mathcal{M} \cap U$ . We call  $\bar{F}$  a (local) smooth extension of  $F$  around  $x$ . The map  $F$  is smooth if it is smooth at all  $x \in \mathcal{M}$ .

In the above definition,  $\bar{F}$  is a map between open subsets of linear spaces: for it to be smooth means it is infinitely differentiable on its domain, in the usual sense. By definition, if  $\bar{F}$  is any smooth map on  $\mathcal{E}$ , and  $\mathcal{M}$  is an embedded submanifold of  $\mathcal{E}$ , then the restriction  $F = \bar{F}|_{\mathcal{M}}$  is smooth on  $\mathcal{M}$ . This still holds if  $\bar{F}$  is only defined on a neighborhood of  $\mathcal{M}$  in  $\mathcal{E}$ , that is, on an open subset of  $\mathcal{E}$  which contains  $\mathcal{M}$ .

Conversely, the following proposition states that a smooth map on  $\mathcal{M}$  admits a smooth extension to a neighborhood of  $\mathcal{M}$ : there is no need to pick a different local smooth extension around each point. While this fact is not needed to establish results hereafter, it is convenient to shorten exposition; so much so that

we typically think of it as the definition of a smooth map. See Section 3.10 for a discussion.

**Proposition 3.31.** *Let  $\mathcal{M}$  and  $\mathcal{M}'$  be embedded submanifolds of  $\mathcal{E}$  and  $\mathcal{E}'$ . A map  $F: \mathcal{M} \rightarrow \mathcal{M}'$  is smooth if and only if  $F = \bar{F}|_{\mathcal{M}}$  where  $\bar{F}$  is some smooth map from a neighborhood of  $\mathcal{M}$  in  $\mathcal{E}$  to  $\mathcal{E}'$ .*

In particular, a real-valued function  $f: \mathcal{M} \rightarrow \mathbb{R}$  is smooth if and only if there exists a smooth extension  $\bar{f}: U \rightarrow \mathbb{R}$  defined on a neighborhood  $U$  of  $\mathcal{M}$  in  $\mathcal{E}$  and which coincides with  $f$  on  $\mathcal{M}$ .

**Definition 3.32.** *A scalar field on a manifold  $\mathcal{M}$  is a function  $f: \mathcal{M} \rightarrow \mathbb{R}$ . If  $f$  is a smooth function, we say it is a smooth scalar field. The set of smooth scalar fields on  $\mathcal{M}$  is denoted by  $\mathfrak{F}(\mathcal{M})$ .*

Smoothness is preserved under composition, and also under linear combinations and products of maps when those are defined: see the exercises in the next section.

**Exercise 3.33.** *Give an example of an embedded submanifold  $\mathcal{M}$  in a linear space  $\mathcal{E}$  and a smooth function  $f: \mathcal{M} \rightarrow \mathbb{R}$  for which there does not exist a smooth extension  $\bar{f}: \mathcal{E} \rightarrow \mathbb{R}$  smooth on all of  $\mathcal{E}$ . Hint: use Example 3.19, or use Proposition 3.23 and consider removing a point from a simple manifold.*

## 3.4 The differential of a smooth map

Let  $\bar{F}: U \subseteq \mathcal{E} \rightarrow \mathcal{E}'$  be a smooth function between two linear spaces, possibly restricted to an open set  $U$ . The differential of  $\bar{F}$  at  $x \in U$  is a linear map  $D\bar{F}(x): \mathcal{E} \rightarrow \mathcal{E}'$  defined by:

$$D\bar{F}(x)[v] = \lim_{t \rightarrow 0} \frac{\bar{F}(x + tv) - \bar{F}(x)}{t}. \quad (3.26)$$

This tells us how  $\bar{F}(x)$  changes when we push  $x$  along  $v$ . Applying this definition to a smooth map  $F: \mathcal{M} \rightarrow \mathcal{M}'$  between two embedded submanifolds is problematic because  $x + tv$  generally does not belong to  $\mathcal{M}$ , even for tiny nonzero values of  $t$ :  $F$  may not be defined there.

We can propose to resolve this issue in at least two ways:

1. Rely on Definition 3.14:  $t \mapsto x + tv$  is nothing but a curve in  $\mathcal{E}$  which passes through  $x$  with velocity  $v$ ; we can use curves on  $\mathcal{M}$  instead.
2. Rely on Definition 3.30: we can smoothly extend  $F$  and differentiate the extension instead.

As it turns out, these two approaches are equivalent. We start with the first one because it is more geometric: it gives the right picture of how things work on general manifolds. The second one is convenient for computation.

For any tangent vector  $v \in T_x\mathcal{M}$ , there exists a smooth curve  $c$  on  $\mathcal{M}$  passing