

---

# Givens Transform Approach for Efficient Probabilistic Principle Component Analysis for Bayesian Dimensionality Reduction (GT-PPCA)

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

We develop scalable and flexible Probabilistic Principal Component Analysis (PPCA) methods for determining posterior distributions of spanning frames based on a Givens Representation of the PCA which we term (GT-PPCA). This addresses significant challenges that arise with latent variable in a traditional formulation of PPCA. For sampling posterior distributions we develop Hamiltonian Monte-Carlo Methods (HMC) for sampling on the Stiefel Manifold the PCA orthogonal frame sets. We demonstrate our approach on several challenging example problems including tests problems XYZ and problems arising in our recent work on understanding medical patient data associated with coagulopathy (factors influencing blood clotting). We show our methods provides ways to identify when data sets contain a mixture of low dimensional structures that would not be resolved with traditional PCA approaches. We further show how our approach can be used to develop heirarchical models in terms of low dimensional structures learned from the data sets or to develop prior distributions useful in generalizing low dimensional structures to new settings. To facilitate use of our GT-PPCA method we provide a package with the widely-used Stan statistics package.

## 1 Introduction

Principal Component Analysis (PCA) is a widely used dimensionality-reduction tool for exploratory analysis and modeling in both the natural and social sciences. By factorizing an empirical covariance matrix into a product of low rank matrices, PCA effectively finds a low dimensional subspace that describes the dataset in terms of the latent factors. These factors are given by the columns of the low rank factorization. Geometrically, traditional PCA can be interpreted as providing a point estimate of a low dimensional hyper-plane that is closest to a cloud of data points. For binary or integer valued matrices, such as graph adjacency matrices arising in network science, Matrix Factorization (NMF) [9] or Exponential Family PCA (EPCA) [4] are often used to find such low dimensional latent factors.

Probabilistic PCA (PPCA) [15] and Bayesian Exponential Family PCA (BXPCA) [12], posit probabilistic generative models that are equivalent to PCA in the limit of decreasing noise [14, chapt. 12.2]. This probabilistic approach is attractive because it allows a straightforward way, via Bayesian inference, to quantify the uncertainty in our estimates (to prevent overfitting) and conduct hypothesis testing. For example, given high-dimensional fMRI scans of a human brain under two different settings, it would be desirable to find a posterior distribution of low-dimensional subspaces that describe the data, then find the probability given the data that these subspaces, or latent factors are different under the two settings. While uncertainty quantification of the latent factors in PPCA has been explored in the literature [7, 1, 2], there are currently no out-of-the-box solutions available to researchers.

In addition to allowing uncertainty quantification and hypothesis testing, probabilistic models are amenable to expansion and can serve as modules within larger probabilistic graphical models. This is important in real-world settings where true generative models are often complex and where they do not necessarily follow the simple generative process set forth by PPCA. Following the fMRI example, a situation could arise where we believe brain scans from people of different age groups may be described by different low-dimensional representations, but that the respective representations for each group all come from some hierarchical prior distribution. In such a case it would be desirable to build a hierarchical model [6, chapt. 5] of latent factors, but such expanded PPCA models have not been explored much, let alone implemented as a package for general use by researchers and scientists.

**Bayesian inference of orthonormal matrices** Many of the difficulties in conducting full Bayesian inference on PPCA and related models stem from having to infer one or more unknown orthonormal matrices. This is difficult because orthonormal matrices form a rather particular subset of all possible realizable matrices; this yields a probability measure on a sub-manifold within the full space of matrices with a non-trivial geometry. More specifically, the prior and posterior distributions of an orthonormal matrix  $W$  must have support over the set of  $n \times p$  orthonormal matrices. This is known as the Stiefel manifold and denoted  $V_{n,p}$  [13]. When first considering this problem, it may seem that this geometry may rule out a straight-forward use of two of the most prominent techniques for posterior inference: Markov Chain Monte Carlo (MCMC) and Variational Inference (VI). Intuitively, for MCMC, we have no way of guaranteeing that a chain over  $W$  will explore only valid regions of parameter space that satisfy the orthonormality constraints. Similarly for Variational Inference, positing a common variational posterior distribution such as a Gaussian over the elements of  $W$  are sure to lead to posteriors that assign mass to invalid regions of parameter space.

In the case of positively constrained parameters, posterior distributions are in practice rather routinely inferred using both MCMC and VI. This can be accomplished by transforming the constrained variables to an unconstrained space using the one-to-one mapping  $f : \text{supp}(z_{\text{constr}}) \rightarrow \mathbb{R}^D$ , where  $\text{supp}(z_{\text{constr}})$  is the support of the constrained random variable  $z_{\text{constr}}$  [3? ]. To our knowledge, no such transformation has been proposed to map orthonormal matrices to a comparable unconstrained space.

**GT-PPCA** In this paper we draw on techniques from Differential Geometry and Numerical Analysis to introduce a novel and geometrically elegant way to represent orthonormal matrices. In our approach we express orthonormal matrices in terms of a sequence of fundamental rotations through given angles. This gives insight in to the geometry of the Stiefel manifold, and results in a transform we call the Givens Transform, that maps orthonormal matrices to an unconstrained space. We apply the Givens Transform to inference of PPCA-based models, and collectively refer to this as GT-PPCA.

GT-PPCA is straight-forward to implement in stand-alone inference schemes, but is particularly useful in the context of a probabilistic programming languages like Stan [3], where we can use it for uncertainty quantification and hypothesis testing of PPCA models, as well as extending PPCA to more complex probabilistic graphical models, two previously difficult tasks. We provide Stan code for our example models allowing for use by researchers and scientists out-of-the-box, and we show how inference of these models in Stan yields good empirical performance in diverse applications such as hierarchical subspace modeling for fMRI data and sparse PPCA for medical data.

Furthermore, GT-PPCA gives insight in to novel and useful ways to work with and interpret our models. For instance, the elegant geometric representation lets us see how by limiting the range of the parameters in GT-PPCA, we can naturally avoid issues of unidentifiability and multi-modal posteriors that arise in other methods. GT-PPCA also allows us new and creative ways to generate and use prior distributions on orthonormal matrices. In the setting of using the matrix directly this task has previously been rather complicated and rather intractable for even small problem sizes. This is linked to the difficulty of evaluating densities of orthonormal matrix distributions in other representations. As we shall discuss in more detail, our GT representation provides a rather natural way to specify prior distributions comparable to the Matrix Langevin prior [13].

**Related work** To address these issues, there has been some recent work on posterior inference of orthonormal matrices based on modified versions of Hamiltonian Monte Carlo (HMC) with sampling for constrained parameters (cite). These methods use modified numerical integrators to ensure that

exploration only occurs in valid areas of parameter space where the necessary constraints are satisfied (cite). Because these methods use modified integrators for constrained parameters, they require approaches in practice for keeping track of the type of each variable and for which type of integrator to use on each variable. This adds an extra layer of algorithmic complexity, especially for large complex probabilistic graphical models. This further makes it difficult to implement these methods in a scalable fashion and within widely used probabilistic programming languages and packages, such as Stan [3]. Furthermore, we have found while these methods represent some progress on this challenging problems the numerical methods are sometimes not rigorous or robust and at times can suffer from errors and instabilities that jeopardize the integrity of samples. In addition, we have found these methods can at times induce unnecessary multi-modality that are an artifact of the chosen representation and not intrinsic to the model under study. This arises in part from the choice of representations directly in terms of unconstrained matrices in these methods (cite). As a consequence, the matrix in the PPCA model has equal likelihood when making an arbitrary change of sign of one of its columns. Given this equivalent relation this can result in significant multi-modality that must be handled with some care in the final analysis.

**Paper outline** We give a brief overview of probabilistic dimensionality reduction in Section 2. We discuss how orthonormal matrices arise naturally for performing inference and introduce a precise description of the Stiefel manifold in Section 2.2. We also briefly review prior work on Bayesian dimensionality reduction using orthonormal matrices in Section 2.2. We introduce the Givens Transform (GT) and use for reductions to obtain our representations in Section 2.4. We also discuss practical consideration for GT in computing and using the representation in Section 2.4. We then show how our methods can be used in practice for Bayesian inference on a few probabilistic graphical models in Section (ref). This includes illustration of the basic aspects of the technique on XYZ and presenting a more advanced real-world application to patient data for factors affecting coagulopathy in Section (ref). We conclude discussing how our GT-PPCA methods can be utilized on other problems and by discussing our implementation in Stan that can be utilized by others interested in our approach.

## 2 Probabilistic Dimensionality Reduction

**PJA: Discuss general context here... linear dimensionality reductions... motivations briefly etc...**

**PJA: Discuss specific approach... PPCA...**

### 2.1 Probabilistic Principle Component Analysis (PPCA)

In probabilistic principle component analysis (PPCA) one starts by considering a collection of data points in a typically high-dimensional vector space and seeks to find a posterior distribution over a reduced representations of the data using a much lower dimensional subspace. The central postulate is that for a data vector  $\mathbf{x} \in \mathbb{R}^n$  there exists an unknown low-dimensional latent representation  $\mathbf{z} \in \mathbb{R}^p$  where  $p < n$ , (ideally with  $p \ll n$ ). The two representations are related to each other by a single unknown linear transformation  $\mathbf{x} \rightarrow \mathbf{z}$ . Mathematically, we consider a finite collection of sampled data vectors  $\mathbf{x}^i \in \mathbb{R}^n$ ,  $i = 1, \dots, N$  and see and estimate of the subspace. Formally, in PPCA this consists of using the following generative process

$$\begin{aligned} p(\mathbf{z}^i) &\sim \mathcal{N}_p(0, I) \\ p(\mathbf{x}^i | \mathbf{z}^i, W, \Lambda, \sigma^2) &\sim \mathcal{N}_n(W\Lambda\mathbf{z}^i, \sigma^2 I). \end{aligned} \quad (1)$$

The  $W$  is an  $n \times p$  orthonormal matrix and  $\Lambda$  is a  $p \times p$  diagonal matrix with positive elements. For simplicity in our presentation of PPCA, we have assumed here that the data has only zero mean but the more general case can also readily be considered (cite). We also remark that generative model could also be expanded to the case of non-Gaussian data by replacing equation 1 with an exponential family member whose natural parameters are given by  $\text{Expon}(W\Lambda\mathbf{z}^i)$  where  $\text{Expon}(\cdot)$  is an appropriate link function [12].

### 2.2 Importance of the Orthonormality Condition

We mention that the orthonormal constraint on the matrix  $W$  plays an important role in obtaining robust methods for in making inferences in probabilistic PCA. If one were to relax the orthonormal

constraint the likelihood function would assign identical probability to a whole equivalence class of matrices  $W \sim V$  where the span is the same linear subspace  $\text{span}\{W\} = \text{span}\{V\}$ . While this might not seem theoretically too problematic, in practice this presents a number of major challenges. This first is that the matrices in a given equivalence class are not all equally well-conditioned numerically in defining the linear subspace and round-off errors and truncation errors become problematic in practical calculations. Secondly, these issues with the representation further manifest in the log-likelihood objective function where regions arise of particularly large curvature. This causes significant numerical issues for variational inference (VI) in nonlinear optimization methods and in monte-carlo (MC) approaches with samplers having slow mixing times Murphy [14], Mohamed et al. [12], Holbrook et al. [8, chapt. 12.1.3]. These issues render making inferences in PPCA with such non-orthonormal matrices impractical resulting from a statistics stand-point in effective unidentifiability.

## 2.3 Inference using the Stiefel Manifold

To develop robust approaches for PPCA we restrict inferences to be over matrices that are constrained to be orthonormal Murphy [14, chapt. 12.1.3]. In the space of  $n \times p$  orthonormal matrices defines the Stiefel Manifold [13]

$$V_{n,p} := \{Y \in \mathbb{R}^{n \times p} : YY^T = I\}. \quad (2)$$

The Stiefel Manifold has an intrinsic dimension of  $\dim V_{n,p} = np - p(p+1)/2$ . This arises from the constraints on the columns of the matrix that impose orthonormality. This dimensionality can be seen by observing, that the first column of  $Y \in V_{n,p}$  must have norm one and hence has one constraint placed on it. The second column must also have norm one and also must be orthogonal to the first column hence with two constraints placed on it. Continuing to the third column through the  $n^{\text{th}}$  one arrives at the conclusion that each point of the Stiefel Manifold has only  $np - p(p+1)/2$  degrees of freedom. We shall discuss how to develop coordinate-charts for representing points within the Stiefel Manifold using Givens transformations in Section (ref).

## 2.4 Givens Transform (GT) approach to PPCA (GT-PPCA)

**PJA:** I would give a streamlined discussion of the GT-PPCA approach giving the basics of the GT and how the representation is used.

**PJA:** I will move much of the material below to an appendix for now and move pieces or re-writes up into this section as needed to make clear. Can always refer reader to the Appendix for the more technical parts. A streamlined discussion GT should be given enough to develop the PPCA, but not too much technicalities on differential geometry and all the rest that an expert could piece together. It is an art at what level to cover things and I can certainly help with that.

# 3 Examples

## 3.1 Synthetic Data

We generated a synthetic, three-dimensional dataset that lies on a two-dimensional plane with  $N = 15$  observations according the generative process of PPCA 1. We chose  $\Lambda = (1, 1)$ ,  $\sigma^2 = 1$ , and  $W$  to be  $I_{3,2}$  which in the Givens representation corresponds to  $\theta_{12} = \theta_{13} = \theta_{23} = 0$ . A standard classical PCA analysis yields the singular values  $\hat{\lambda} = (1.52 \ 1.27 \ 0.77)$ , suggesting that our data lie close to some two-dimensional plane. Figure ?? illustrates a point estimate of the subspace found by PCA. Meanwhile, Figure ?? shows posterior intervals of angles and singular values obtained from HMC in Stan.

## 3.2 A hierarchical PPCA model for fMRI data

## 3.3 School Network

Mainly here to show we can do exponential families easy too. Possibly make this a change point detection?

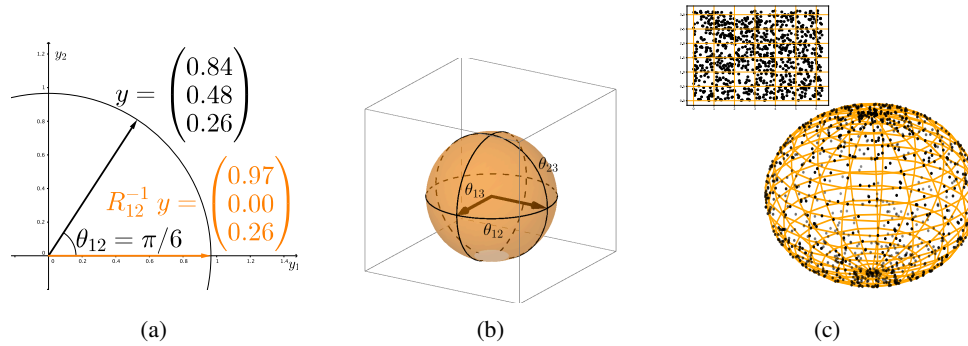


Figure 1: Visualizing the Givens Transform. (a) How the Givens Reduction “zeros out” a column vector. (b) A geometric view of the Stiefel manifold, two-frame in three dimensions. (c) Sampling without a proper measure adjustment. **PJA: Good start on figures. Please remember to use vector format .PDF final output. Can use Inkscape or Adobe Illustrator to process PDF and SVG files from plotters like Python Matplotlib or Matlab or Stan, etc... Please be sure to save both .SVGZ (source file) and .PDF files in Git repo. This allows for easy edits later to figures as needed. PJA: Fonts are a little too small on Fig. a. Color in Fig. b would be better to use light blue or something more neutral. Use Red for labels... remember color balance. Point sizes a little bigger in in-set in Fig. C. Save both sphere dots and square separately and use Inkscape / Illustrator to combine.**

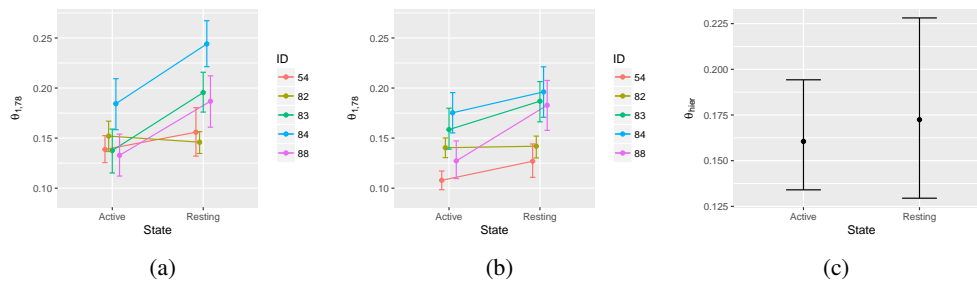


Figure 2: Inferring a hierarchical subspace model for fMRI data. (a) Individual posterior estimates of Givens Transform angle  $\theta_{1,78}$ . (b) Hierarchical estimates. (c) Posterior mean of hierarchical mean parameter.

### 3.4 Coagulopathy using hierarchical subspace models

Don’t forget to mention sparse PCA, via Cauchy priors.

## 4 Discussion

### Acknowledgments

Use unnumbered third level headings for the acknowledgments. All acknowledgments go at the end of the paper. Do not include acknowledgments in the anonymized submission, only in the final paper.

## References

- [1] Marcus Brubaker, Mathieu Salzmann, and Raquel Urtasun. A family of mcmc methods on implicitly defined manifolds. In *Artificial Intelligence and Statistics*, pages 161–172, 2012.
- [2] Simon Byrne and Mark Girolami. Geodesic monte carlo on embedded manifolds. *Scandinavian Journal of Statistics*, 40(4):825–845, 2013.
- [3] Bob Carpenter, Andrew Gelman, Matt Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Michael A Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software*, 20, 2016.

- 197 [4] Michael Collins, Sanjoy Dasgupta, and Robert E Schapire. A generalization of principal  
198 component analysis to the exponential family. In *Nips*, volume 13, page 23, 2001.
- 199 [5] Alan Edelman. 18.325: Finite random matrix theory volumes and integration. 2005.
- 200 [6] Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian data analysis*,  
201 volume 2. Chapman & Hall/CRC Boca Raton, FL, USA, 2014.
- 202 [7] Peter D Hoff. Simulation of the matrix bingham–von mises–fisher distribution, with applications  
203 to multivariate and relational data. *Journal of Computational and Graphical Statistics*, 18(2):  
204 438–456, 2009.
- 205 [8] Andrew Holbrook, Alexander Vandenberg-Rodes, and Babak Shahbaba. Bayesian inference on  
206 matrix manifolds for linear dimensionality reduction. *arXiv preprint arXiv:1606.04478*, 2016.
- 207 [9] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In  
208 *Advances in neural information processing systems*, pages 556–562, 2001.
- 209 [10] Benedict Leimkuhler and Sebastian Reich. *Simulating hamiltonian dynamics*, volume 14.  
210 Cambridge University Press, 2004.
- 211 [11] Carl D Meyer. *Matrix analysis and applied linear algebra*, volume 2. Siam, 2000.
- 212 [12] Shakir Mohamed, Zoubin Ghahramani, and Katherine A Heller. Bayesian exponential family  
213 pca. In *Advances in neural information processing systems*, pages 1089–1096, 2009.
- 214 [13] Robb J Muirhead. *Aspects of multivariate statistical theory*, volume 197. John Wiley & Sons,  
215 2009.
- 216 [14] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- 217 [15] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis.  
218 *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622,  
219 1999.

## 220 Appendix

### 221 4.1 Givens Reductions of Matrices

222 PJA: Give a basic description of the background on Givens reductions of a matrix that we can refer to  
223 with additional details than we need in the main text.

### 224 4.2 Differential Geometry of the Steifel Manifold

225 PJA: Discuss the basic differential geometry of the Steifel Manifold and how we handle these various  
226 issues. Degenerate regions and metric factors (Jacobians). Discuss how we use "multi-coordinate  
227 charts" when necessary to avoid being close to regions with bad metric factors and degeneracy, etc...

$$R_{ij} := \begin{pmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & \cos \theta_{ij} & & & & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & \sin \theta_{ij} & & & & \cos \theta_{ij} & & \\ & & & & & & & 1 & \\ & & & & & & & & \ddots & \\ & & & & & & & & & 1 \end{pmatrix} \begin{matrix} i \\ j \end{matrix}$$

$$R_{12}^{-1}Y = R_{12}^{-1} \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1p} \\ y_{21} & y_{22} & \cdots & y_{2p} \\ y_{31} & y_{32} & \cdots & y_{3p} \\ \vdots & \vdots & \vdots & \vdots \\ y_{p1} & y_{p2} & \cdots & y_{pp} \\ y_{p+1,1} & y_{p+1,2} & \cdots & y_{p+1,p} \\ \vdots & \vdots & \vdots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{np} \end{pmatrix} = \begin{pmatrix} * & * & \cdots & * \\ 0 & * & \cdots & * \\ y_{31} & y_{32} & \cdots & y_{3p} \\ \vdots & \vdots & \vdots & \vdots \\ y_{p1} & y_{p2} & \cdots & y_{pp} \\ y_{p+1,1} & y_{p+1,2} & \cdots & y_{p+1,p} \\ \vdots & \vdots & \vdots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{np} \end{pmatrix}. \quad (3)$$

$$(R_{1n}^{-1} \cdots R_{13}^{-1} R_{12}^{-1})Y = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & * & \cdots & * \\ 0 & * & \cdots & * \\ \vdots & \vdots & \vdots & \vdots \\ 0 & * & \cdots & * \\ 0 & * & \cdots & * \\ \vdots & \vdots & \vdots & \vdots \\ 0 & * & \cdots & * \end{pmatrix}. \quad (4)$$

$$(R_{2n}^{-1} \cdots R_{23}^{-1})(R_{1n}^{-1} \cdots R_{12}^{-1})Y = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & * \\ 0 & 0 & \cdots & * \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & * \\ 0 & 0 & \cdots & * \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & * \end{pmatrix}. \quad (5)$$

229 Continuing in this fashion yields

$$(R_{pn}^{-1} \cdots R_{p,p+1}^{-1}) \cdots (R_{2n}^{-1} \cdots R_{23}^{-1})(R_{1n}^{-1} \cdots R_{12}^{-1})Y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} = I_{np}. \quad (6)$$

## 230 5 Misc LaTeX

### 231 5.1 Related works using HMC

232 **PJA: I'd move this material in some form to the introduction and give a brief overview of how out**  
 233 **methods are different. Then maybe mention briefly during discussions prior work when our approach**  
 234 **differs and explain why (as you do below, but in a sentence or two in the particular context).**

235 Before discussing details of our approach in more detail, we briefly discuss related prior works.  
 236 Recently, two approaches to posterior inference of orthonormal matrices and other constrained  
 237 parameters have been proposed. These methods work by modifying the leap-frog integrator used to  
 238 simulate Hamiltonian dynamics in HMC to ensure that any trajectory in parameter space remains on  
 239 the Stiefel manifold. Brubaker et al. [1] uses the SHAKE integrator [10] to ensure trajectories through  
 240 parameter space continuously satisfy the required constraints. The integrator works by repeatedly  
 241 taking a step forward that may be off the manifold using ordinary leap frog, then projecting back  
 242 down to the nearest point on the manifold. This projection is done via Newton iterations, which may  
 243 not converge in practice, possibly jeopardizing the ergodicity of a Markov chain.

244 Byrne and Girolami [2] took a different approach, exploiting the fact that closed form solutions are  
 245 known for the geodesic equations over the Stiefel manifold in the embedded coordinates,  $W$ . While  
 246 this method is completely explicit, requiring no Newton iterations, in practice we found that for larger  
 247 step sizes, the integrator steps off the Stiefel manifold, due to the numerical imprecision of the matrix  
 248 exponential function.

249 Both methods rely on applying different integrators to constrained parameters. This adds imple-  
 250 mentation difficulty in practice, as one must keep track of the type of each variable (unconstrained  
 251 or constrained, and type of constraint), and apply the appropriate integrator. Additionally, we note  
 252 that even using orthonormal matrices, the PPCA likelihood is equivalent for a matrix  $W$  and any  
 253 permutation of the columns of  $W$  being negative. As such, these methods lead to multi-modal  
 254 posteriors, that can be avoided in a straight-forward way, as we show in the next section, using the  
 255 Givens transform.

### 256 5.2 Givens Reductions

257 **PJA: I've temporarily moved this to the misc text section, to be added back into the main exposition**  
 258 **in a more streamlined way.** We provide a brief exposition on Givens Reductions, which motivate the  
 259 Givens Transform, then describe the Givens Transform along with relevant practical considerations.

260 Define  $R_{ij}$  to be the  $n \times n$  rotation matrix that performs a counter-clockwise rotation in the  $(i, j)$ -  
 261 plane of  $\mathbb{R}^n$ , where  $j > i$ . In  $\mathbb{R}^3$ , there are three such matrices,  $R_{12}$ ,  $R_{13}$ , and  $R_{23}$ . They perform  
 262 counter-clockwise rotation of angle  $\theta_{ij}$  in the  $(x, y)$ ,  $(x, z)$  and  $(y, z)$  planes respectively. Rotation  
 263 matrices have the following key properties:

- 264 1. They preserve length and angles between vectors, i.e. for two vectors  $u, v \in \mathbb{R}^n$ ,  $R_{ij}u, R_{ij}v$   
 265 are the same length as  $u$  and  $v$  respectively, and if  $u$  and  $v$  are orthogonal then so are  $R_{ij}u$   
 266 and  $R_{ij}v$ .



267 2. They are invertible and their inverse is their transpose  $R_{ij}^{-1} = R_{ij}^T$ . Their inverse corresponds  
 268 to a clockwise rotation in the  $(i, j)$ -plane.

269 Now we consider an  $n \times p$  matrix  $Y$ , with orthonormal columns. In general, the first column is a  
 270 vector in  $\mathbb{R}^n$  with a non-zero second element. However, we can apply an invertible clockwise rotation  
 271 in the  $(1, 2)$ -plane,  $R_{12}^{-1}$ , to “zero out” the second element of the first column. Figure 2a depicts this  
 272 visually for a 3D vector projected on to the  $(1, 2)$ -plane.

273 Similarly, we can apply consecutive rotations  $R_{13}^{-1}, R_{14}^{-1}, \dots, R_{1n}^{-1}$  to this result so that all entries  
 274 of the first column besides the first element are zero. The first column of the resulting matrix,  
 275  $R_{1n}^{-1} \dots R_{13}^{-1} R_{12}^{-1} Y$ , will be the length one vector  $(1 \ 0 \ \dots \ 0)^T$  which lies entirely along the first axis.  
 276 If one takes the perspective that these rotations are applied to the columns of  $Y$ , then with the two  
 277 properties of rotation matrices we mentioned earlier, it is evident that columns 2 through  $n$  must have  
 278 zero in their first element because they will be orthogonal to the first column,  $(1 \ 0 \ \dots \ 0)^T$ .

279 To “zero out” the elements of the second column, one can similarly apply rotations  $R_{23}^{-1}, \dots, R_{2n}^{-1}$ .  
 280 These rotations will leave the first column and first row unaffected, as the first column no lies entirely  
 281 on the 1st axis, which these rotations do not involve. Continuing in this fashion yields

$$(R_{pn}^{-1} \dots R_{p,p+2}^{-1} R_{p,p+1}^{-1}) \dots (R_{2n}^{-1} \dots R_{24}^{-1} R_{23}^{-1})(R_{1n}^{-1} \dots R_{13}^{-1} R_{12}^{-1})Y = I_{n,p}, \quad (7)$$

282 where  $I_{n,p}$  consists of the first  $p$  columns of the  $n \times n$  identity matrix. This process of applying  
 283 consecutive rotation matrices to a matrix is known in numerical analysis as the Givens reduction  
 284 [11], and is applied more generally to square matrices for matrix-vector solves. In total we will have  
 285 applied  $(n-1) + (n-2) + \dots + (n-p) = np - p(p+1)/2$  rotations matrices. We note that  
 286 because rotation matrices are very sparse in high dimensions, multiplication by a rotation matrix is  
 287 computationally much less intensive than matrix multiplication otherwise would be.

### 288 5.3 Givens Transform

289 Because rotation matrices are invertible, equation 7 implies that we can rewrite the  $n \times p$  orthonormal  
 290 matrix  $Y$  as the product of counter-clockwise rotation matrices and  $I_{n,p}$ :

$$Y = (R_{12} \dots R_{1n}) \dots (R_{23} \dots R_{2n})(R_{p+1,n} \dots R_{pn})I_{n,p}. \quad (8)$$

291 Recall that each of the  $np - p(p+1)/2$  rotation matrices have an associated angle  
 292  $(\theta_{12} \dots \theta_{1n}) \dots (\theta_{23} \dots \theta_{2n})(\theta_{p+1,n} \dots \theta_{pn})$ , that we collectively refer to as  $\Theta$ . In this way we  
 293 have reparameterized all  $n \times p$  orthonormal matrices<sup>1</sup>, a constrained space, in terms of unconstrained  
 294 angles<sup>2</sup>, using a transform  $\Theta : V_{n,p} \rightarrow \mathbb{R}^{np-p(p+1)/2}$ . We refer to 8 as the Givens representation or  
 295 Givens transform.

#### 296 5.3.1 A geometric perspective

297 The Givens transform can be visualized in the simple case when  $n = 3$  and  $p = 2$ . This is the case  
 298 where data is three-dimensional and we seek a two-dimensional subspace to represent the data. In  
 299 this case, we can imagine a two dimensional bases, visualized as two perpendicular vectors that rotate  
 300 rigidly together along three angles of rotations. This is referred to as a 2-frame in differential geometry  
 301 [13] and is depicted in Figure 2b. Recall  $\theta_{12}$  controls the amount of rotation in the  $(1, 2)$ -plane,  $\theta_{13}$   
 302 in the  $(1, 3)$ -plane, and  $\theta_{23}$  controls a rotation of the second basis vector about the first. All rotations  
 303 will preserve the length and orthogonality of the basis vectors. Given a flat, pancake-like cloud of  
 304 points in 3D, the two dimensional bases will move according to the appropriate angles to be aligned  
 305 with the cloud of points, if one were conducting a (Maximum A-Posteriori) MAP estimation over the  
 306 Stiefel manifold.

<sup>1</sup>other than a set of measure zero we explain in the next subsection

<sup>2</sup>the angles are themselves constrained to lie in certain intervals e.g.  $[0, \pi)$  but these sorts of constraints are routine to deal with using a one-to-one diffeomorphism between intervals and the real line e.g. the sigmoid transform

### 5.3.2 Practical considerations of topology and angles

Topologically,  $V_{n,p}$  is locally equivalent to Euclidean space, but not globally equivalent, meaning it is impossible to find a one-to-one map between the Stiefel manifold and Euclidean space. Technically speaking, the Givens transform can map angles to all of  $V_{n,p}$  except for a subset  $S \subset V_{n,p}$ , that in the  $n = 3, p = 2$  case corresponds to a sliver when  $\theta_{12} \in (-\pi, \pi)$ ,  $\theta_{13} \in (-\pi/2, \pi/2)$ , and  $\theta_{23} \in (-\pi/2, \pi/2)$ . Luckily this set is of measure zero (under the proper measure for the Stiefel manifold, see section 5.3.4), and thus, with probability one, the orthonormal matrix that describes the true subspace our data lie in will not be in that set.

In practice, we actually limit the angle  $\theta_{12}$  to an interval of length  $\pi$  rather than an interval of length  $2\pi$ , that traverses the entire Stiefel manifold. Examining the angles of the Givens transform makes it evident that in the latter case, two equivalent bases that are the negation of each other can be reached, resulting in a multi-modal posterior that makes sampling and VI more difficult and harder to interpret. To avoid this multi-modality using the modified integrator methods would require a mechanism to avoid boundaries, which are not as intuitively defined in the default embedded coordinates as in the Givens transform.

Lastly, we note that if the true bases lies near a pole, i.e.  $\theta_{ij}$  is close to  $-\pi/2$  or  $\pi/2$ , then posteriors will tend to be multi-modal as the region in parameter space close to the boundaries will be close to equally valid, while the region near zero, will not be valid and thus contain little probability mass. In these cases, one can simply change the chart so that  $\theta_{ij} \in (0, \pi)$ , creating a uni-modal posterior in the new coordinate system, and alleviating numerical issues. In Stan this is straight-forward, as one simply has to change the lower and upper bound of the angle parameter.

### 5.3.3 Jacobian under a change of variables

The Givens transform 8 allows us to represent orthonormal matrices as angles  $Y(\Theta)$ . This in turn allows us to write probability densities  $p_Y(Y)$  in terms of angles, so that we can conduct inference in an unconstrained space. It is well known in probability theory that the transform of a random variable is in general not the density of the transform, i.e.  $p_\Theta(\Theta) \neq p_Y(Y(\Theta))$  [14, chapt. 2.6]. To be more precise, densities are measured against volumes and integrated to get actual probabilities (otherwise known as probability mass). Under a transformation, densities are unaffected, but volumes (or rather the way in which volumes are measured) may change. This is important in the context of posterior inference, as not including the Jacobian adjustment would result in different priors than we intend.

Figure 2c depicts how samples that are uniform in the angle space are not uniform on the sphere. Samples congregate at the poles of the sphere because a patch of area in the angle space that is near the top corresponds to a very tiny patch of the sphere near the pole. In practice this would lead to posteriors that bias towards the poles, when what we really intend is a prior that is uniform on the Stiefel manifold.

For a  $K$ -dimensional random vector  $Y$  and a transformation  $f : \text{supp}(Y) \rightarrow \mathbb{R}^K$  the proper way to measure probability under a transform is by multiplying by the determinant of the Jacobian of the the inverse transform:

$$\int p_\Theta(\Theta) d\Theta = \int p_Y(Y(\Theta)) |\det J_{f^{-1}}(\Theta)| d\Theta. \quad (9)$$

In our case this poses a problem however, because an  $n \times p$  orthonormal matrix is  $np$ -dimensional, but the Givens transform,  $\Theta(Y)$ , maps this set to a  $np - p(p+1)2$ -dimensional set. In this more general scenario, one can not simply take the determinant of the Jacobian as the volume morphing factor, because the Jacobian is not even square and hence the determinant is undefined. To obtain the correct factor one must appeal to the calculus of differential forms.

### 5.3.4 Differential forms

We offer an intuitive high-level overview of differential forms. For a thorough account we recommend Muirhead [13], Edelman [5], or any standard text in differential geometry. The simplest non-trivial example of differential-forms between spaces of different sizes arises when trying to measure probability using spherical coordinates. Spherical coordinates give us a map from  $\mathbb{R}^2$  to the sphere,  $(\theta, \varphi) \mapsto (x(\theta, \varphi), y(\theta, \varphi), z(\theta, \varphi))$ . Although a sphere lies in 3-D space, if we have a density

356  $p_{\text{Euc}}(x, y, z)$  on the sphere, the “natural” way to measure the probability of a spherical random  
 357 variable falling within some area on the surface of that sphere is by first covering that area with tiny  
 358 rectangles tangent to the sphere, taking the average density of each rectangle, multiplying that density  
 359 by the area of that rectangle, and summing over the resulting products. The probability of the random  
 360 variable falling within that area is defined to be the limit of that result as the size of the rectangles  
 361 go to zero. Area forms, or two-forms can be thought of as these infinitesimal rectangles. They are  
 362 written as the wedge product of two differentials, e.g.  $dx \wedge dy$ , and they are objects that are inserted  
 363 into integrals to obtain measures. Differential forms form (no pun intended) a vector space useful  
 364 for measuring areas and high-dimensional volumes. For example, the area forms on a sphere can be  
 365 written as a linear combination of the standard-basis two forms

$$a(x, y, z) dx \wedge dy + b(x, y, z) dx \wedge dz + c(x, y, z) dy \wedge dz. \quad (10)$$

366 One can then integrate this over a sub-area of the sphere to obtain the measure of that sub-area.  
 367 The beauty of differential forms is that a different coordinate system such as polar coordinates,  
 368 simply correspond to using a different basis for representing forms, which again are vectors. In fact  
 369 writing a form in a different bases simply involves taking partial derivatives, e.g.  $dx = (\partial x / \partial \theta) d\theta +$   
 370  $(\partial x / \partial \varphi) d\varphi$ . If we substitute the forms in the angle bases in to 10 and simplify using the well-defined  
 371 anti-symmetric and distributive properties of wedge products and differential forms (see [13]), we  
 372 obtain a proper area-form 10 in spherical coordinates,  $S(x(\theta, \varphi), y(\theta, \varphi), z(\theta, \varphi)) d\theta \wedge d\varphi$ , that can  
 373 then be integrated using a double integral in polar coordinates. The absolute value of this area-form  
 374 evaluated at some point specified in angle coordinates,  $(\theta_0, \varphi_0)$ , intuitively measures how much the  
 375 area of tiny square in angle space gets shrunk or stretched when the area is mapped to an area on the  
 376 sphere.

377 Analogously, we can measure volumes on the Stiefel manifold. For  $n \times p$  orthonormal matrices, there  
 378 are only  $np - p(p + 1)/2$  free parameters and so the proper form to measure sets of orthonormal  
 379 matrices is in fact a  $np - p(p + 1)/2$ -form. For an orthonormal,  $n \times p$  matrix,  $Y$ , we can find an  
 380 orthonormal  $n \times n$  matrix  $G$  such that  $G^T Y = I_{n,p}$ . In fact  $G$  just comes from the product of the  
 381 appropriate rotation matrices that comes from the Givens Reduction 7. Muirhead [13] shows that  
 382 the correct form comes from wedging the elements of the  $n \times p$  matrix  $G^T dY$  that lie below the  
 383 diagonal i.e.

$$\bigwedge_{i=1}^p \bigwedge_{j=i+1}^n G_j^T dY_i \quad (11)$$

384 where  $G_j$  is the  $j$ th column of  $G$  and  $Y_i$  is the  $i$ th column of  $Y$ . To obtain the form in angle  
 385 coordinates we simply obtain  $dY$  in angle coordinates.  $dY_i$  can be obtained in terms of the angle  
 386 coordinates by the following relationship,  $dY_i = J_{Y_i}(\Theta) d\Theta$ , where  $J_{Y_i}$  is the Jacobian of  $Y_i$  with  
 387 respect to the angle coordinates. Once we obtain the form 11 in terms of the angle coordinates, the  
 388 result is a wedge product of  $np - p(p + 1)/2$  vectors that are  $np - p(p + 1)/2$  dimensional, which  
 389 reduces to the determinant of these vectors aligned side by side as a  $np - p(p + 1)/2 \times np - p(p + 1)/2$   
 390 matrix. This determinant is analogous to and serves the same purpose as Jacobian adjustment that  
 391 comes from transforming random variables. We can insert it in to the log-probability of a model to  
 392 avoid the sort of unintended sampling behavior depicted in Figure 2c. We incorporate the form 11 in  
 393 to the log-probability of all of our Stan examples.