Spherical-Homoscedastic Distributions: The Equivalency of Spherical and Normal Distributions in Classification

Onur C. Hamsici hamsicio@ece.osu.edu

Aleix M. Martinez aleix@ece.osu.edu

Department of Electrical and Computer Engineering
The Ohio State University
Columbus, OH 43210, USA

Editor: Greg Ridgeway

Abstract

Many feature representations, as in genomics, describe directional data where all feature vectors share a common norm. In other cases, as in computer vision, a norm or variance normalization step, where all feature vectors are normalized to a common length, is generally used. These representations and pre-processing step map the original data from $\mathbb{R}^p$ to the surface of a hypersphere S^{p-1}. Such representations should then be modelled using spherical distributions. However, the difficulty associated with such spherical representations has prompted researchers to model their spherical data using Gaussian distributions instead – as if the data were represented in $\mathbb{R}^p$ rather than S^{p-1}. This opens the question to whether the classification results calculated with the Gaussian approximation are the same as those obtained when using the original spherical distributions. In this paper, we show that in some particular cases (which we named spherical-homoscedastic) the answer to this question is positive. In the more general case however, the answer is negative. For this reason, we further investigate the additional error added by the Gaussian modelling. We conclude that the more the data deviates from spherical-homoscedastic, the less advisable it is to employ the Gaussian approximation. We then show how our derivations can be used to define optimal classifiers for spherical-homoscedastic distributions. By using a kernel which maps the original space into one where the data adapts to the spherical-homoscedastic model, we can derive non-linear classifiers with potential applications in a large number of problems. We conclude this paper by demonstrating the uses of spherical-homoscedasticity in the classification of images of objects, gene expression sequences, and text data.

Keywords: Directional Data, Spherical Distributions, Normal Distributions, Norm Normalization, Linear and Non-linear Classifiers, Computer Vision.

1. Introduction

Many problems in science and engineering involve spherical representations or directional data, where the sample vectors lie on the surface of a hypersphere. This is typical, for example, of some genome sequence representations (Janssen et al., 2001; Audit and Ouzounis, 2003), in text analysis and clustering (Dhillon and Modha, 2001; Banerjee et al., 2005), and in morphometrics (Slice, 2005). Moreover, the use of some kernels (e.g., radial basis function) in machine learning algorithms, will reshape all sample feature vectors to have

a common norm. That is, the original data is mapped into the surface of a hypersphere. Another area where spherical representations are common is in computer vision, where spherical representations emerge after the common norm-normalization step is incorporated. This pre-processing step guarantees that all vectors have a common norm and it is used in systems where the representation is based on the shading properties of the object to make the algorithm invariant to changes of the illumination intensity, and when the representation is shape-based to provide scale and rotation invariance. Typical examples are in object and face recognition (Murase and Nayar, 1995; Belhumeur and Kriegman, 1998), pose estimation (Javed et al., 2004), shape analysis (Dryden and Mardia, 1998) and gait recognition (Wang et al., 2003; Veeraraghavan et al., 2005).

Figure 1 provides two simple computer vision examples. On the left hand side of the figure the two p-dimensional feature vectors $\hat{\mathbf{x}}_i$, $i = \{1, 2\}$, correspond to the same face illuminated from the same angle but with different intensities. Here, $\hat{\mathbf{x}}_1 = \alpha\hat{\mathbf{x}}_2$, $\alpha \in \mathbb{R}$, but normalizing these vectors to a common norm results in the same representation $\mathbf{x}$; i.e., the resulting representation is invariant to the intensity of the light source. On the right hand side of Figure 1, we show a classical application to shape analysis. In this case, each of the p elements in $\hat{\mathbf{y}}_i$ represents the Euclidean distances from the centroid of the 2D shape to a set of p equally separated points on the shape. Normalizing each vector with respect to its norm, guarantees our representation is scale invariant.

The most common normalization imposes that all vectors have a unit norm, that is,

$$\mathbf{x} = \frac{\hat{\mathbf{x}}}{\|\hat{\mathbf{x}}\|},$$

where $\hat{\mathbf{x}} \in \mathbb{R}^p$ is the original feature vector, and $\|\mathbf{x}\|$ is the magnitude (2-norm length) of the vector $\mathbf{x}$. When the feature vectors have zero mean, it is common to normalize these with respect to their variances instead,

$$\mathbf{x} = \frac{\hat{\mathbf{x}}}{\sqrt{\frac{1}{p-1}\sum_{i=1}^{p}\hat{\mathbf{x}}^2}} = \sqrt{p-1}\frac{\hat{\mathbf{x}}}{\|\hat{\mathbf{x}}\|},$$

which generates vectors with norms equal to $\sqrt{p-1}$. This second option is usually referred to as variance normalization.

It is important to note that these normalizations enforce all feature vectors $\mathbf{x}$ to be at a common distance from the origin; i.e., the original feature space is mapped to a spherical representation (see Figure 1). This means that the data now lays on the surface of the $(p-1)$-dimensional unit sphere S^{p-1}. [1]

Our description above implies that the data would now need to be interpreted as spherical. For example, while the illumination subspace of a (Lambertian) convex object illuminated by a single point source at infinity is known to be 3-dimensional (Belhumeur and Kriegman, 1998), this corresponds to the 2-dimensional sphere S^2 after normalization. The third dimension (not shown in the spherical representation) corresponds to the intensity of the source. Similarly, if we use norm-normalized images to define the illumination cone,

1. Since all spherical representations are invariant to the radius (i.e., there is an isomorphism connecting any two representations of distinct radius), selecting a specific value for the radius is not going to effect the end result. In this paper, we always impose this radius to be equal to one.

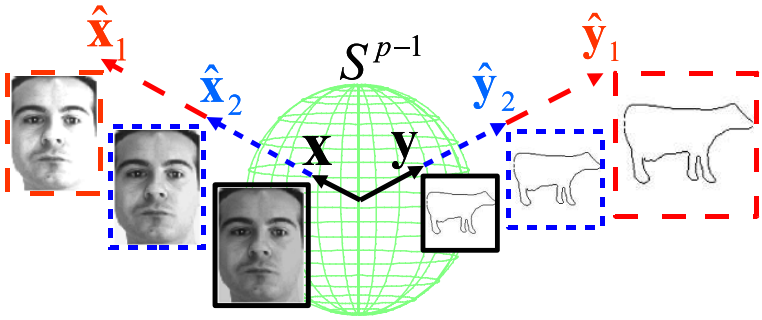

Figure 1: On the left hand side of this figure, we show two feature vectors corresponding to the same face illuminated from the same position but with different intensities. This means that $\hat{\mathbf{x}}_1 = \alpha\hat{\mathbf{x}}_2$, $\alpha \in \mathbb{R}$. Normalizing these two vectors with respect to their norm yields a common solution, $\mathbf{x} = \frac{\hat{\mathbf{x}}_1}{\|\hat{\mathbf{x}}_1\|} = \frac{\hat{\mathbf{x}}_2}{\|\hat{\mathbf{x}}_2\|}$. The norm-normalized vector $\mathbf{x}$ is on S^{p-1}, whereas $\hat{\mathbf{x}}_i \in \mathbb{R}^p$. On the right hand side of this figure we show a shape example where the elements of the feature vectors $\hat{\mathbf{y}}_i$ represent the Euclidean distance between the centroid of the 2D shape and p points on the shape contour. As above, $\hat{\mathbf{y}}_1 = \beta\hat{\mathbf{y}}_2$ (where $\beta \in \mathbb{R}$), and normalizing them with respect to their norm yields $\mathbf{y}$.

the extreme rays that define the cone will be the extreme points on the corresponding hypersphere.

An important point here is that data would now need to be modeled using spherical distributions. However, the computation of the parameters that define spherical models is usually complex, very costly and, in many cases, impossible to obtain (see Section 2 for a review). This leaves us with an unsolved problem: To make a system invariant to some parameters, we want to use spherical representations (as in genomics) or normalize the original feature vectors to such a representation (as in computer vision). But, in such cases, the parameter estimation of our distribution is impossible or very difficult. This means, we are left to approximate our spherical distribution with a model that is well-understood and easy to work with. Typically, the most convenient choice is the Gaussian (Normal) distribution.

The question arises: *how accurate are the classification results obtained when approximating spherical distributions with Gaussian distributions?*

Note that if the Bayes decision boundary obtained with Gaussians is very distinct to that found by the spherical distributions, our results will not generally be useful in practice. This would be catastrophic, because it would mean that by using spherical representations to solve one problem, we have created another problem that is even worse.

In this paper, we show that in almost all cases where the Bayes classifier is linear (which is the case when the data is what we will refer to as *spherical-homoscedastic* – a rotation-invariant extension of homoscedasticity) the classification results obtained on the true underlying spherical distributions and on those Gaussians that best approximate them are identical. We then show that for the general case (which we refer to as *spherical-heteroscedastic*) these classification results can vary substantially. In general, the more the

data deviates from our spherical-homoscedastic definition, the more the classification results diverge from each other. This provides a mechanism to test when it makes sense to use the Gaussian approximation and when it does not.

Our definition of spherical-homoscedasticity will also allow us to define simple classification algorithms that provide the minimal Bayes classification error for two spherical homoscedastic distributions. This result can then be extended to the more general spherical-heteroscedastic case by incorporating the idea of the kernel trick. Here, we will employ a kernel to (intrinsically) map the data to a space where the spherical-homoscedastic model provides a good fit.

The rest of this paper is organized as follows. Section 2 presents several of the commonly used spherical distributions and describes some of the difficulties associated to their parameter estimation. In Section 3, we introduce the concept of spherical-homoscedasticity and show that whenever two spherical distributions comply with this model, the Gaussian approximation works well. Section 4 illustrates the problems we will encounter when the data deviates from our spherical-homoscedastic model. In particular, we study the classification error added when we model spherical-heteroscedastic distributions with the Gaussian model. Section 5 presents the linear and (kernel) non-linear classifiers for spherical-homoscedastic and -heteroscedastic distributions, respectively. Our experimental results are in Section 6. Conclusions are in Section 7. A summary of our notation is in Appendix A.

2. Spherical Data

2.1 Spherical Distributions

Spherical data can be modelled using a large variety of data distributions (Mardia and Jupp, 2000), most of which are analogous to distributions defined for the Cartesian representation. For example, the von Mises-Fisher (vMF) distribution is the spherical counterpart of those Gaussian distributions that can be represented with a covariance matrix of the form $\tau^2\mathbf{I}$; where $\mathbf{I}$ is the $p \times p$ identity matrix and $\tau > 0$. More formally, the probability density function (pdf) of the p-dimensional vMF model $M(\mu, \kappa)$ is defined as

$$f(\mathbf{x}|\mu, \kappa) = c_{MF}(p, \kappa) \exp\{\kappa \mu^T \mathbf{x}\}, \tag{1}$$

where $c_{MF}(p, \kappa)$ is a normalizing constant which guarantees that the integral of our density over the $(p-1)$-dimensional sphere S^{p-1} is one, $\kappa \geq 0$ is the concentration parameter, and μ is the mean direction vector (i.e. $\|\mu\| = 1$). Here, the concentration parameter κ is used to represent distinct types of data distributions – from uniformly distributed (for which κ is small) to very localized (for which κ is large). This means that when $\kappa = 0$ the data will be uniformly distributed over the sphere, and when $\kappa \to \infty$ the distribution will approach a point.

As mentioned above, Eq. (1) can only be used to model circularly symmetric distributions around the mean direction. When the data does not conform to such a distribution type, one needs to use more flexible pdfs such as the Bingham distribution (Bingham, 1974). The pdf for the p-dimensional Bingham $B(\mathbf{A})$ is an antipodally symmetric function (i.e., $f(-\mathbf{x}) = f(\mathbf{x})$, $\forall \mathbf{x} \in S^{p-1}$) given by

$$f(\pm\mathbf{x}|\mathbf{A}) = c_B(p, \mathbf{A}) \exp\{\mathbf{x}^T \mathbf{A} \mathbf{x}\}, \tag{2}$$

where $c_B(p, \mathbf{A})$ is the normalizing constant and $\mathbf{A}$ is a $p \times p$ symmetric matrix defining the parameters of the distribution. Note that since the feature vectors have been mapped onto the unit sphere, $\mathbf{x}^T\mathbf{x} = 1$. This means that substituting $\mathbf{A}$ for $\mathbf{A} + c\mathbf{I}$ with any $c \in \mathbb{R}$ would result in the same pdf as that shown in (2). To eliminate this redundancy, we need to favor a solution with an additional constraint. One such constraint is $\lambda_{MAX}(\mathbf{A}) = 0$, where $\lambda_{MAX}(\mathbf{A})$ is the largest eigenvalue of $\mathbf{A}$.

For many applications the assumption of antipodally symmetric is inconvenient. In such cases, we can use the Fisher-Bingham distribution (Mardia and Jupp, 2000) which combines the idea of von Mises-Fisher with that of Bingham, yielding the following p-dimensional Fisher-Bingham $FB(\mu, \kappa, \mathbf{A})$ pdf

$$f(\mathbf{x}|\mu, \kappa, \mathbf{A}) = c_{FB}(\kappa, \mathbf{A}) \exp\{\kappa\mu^T\mathbf{x} + \mathbf{x}^T\mathbf{A}\mathbf{x}\}, \tag{3}$$

where $c_{FB}(\kappa, \mathbf{A})$ is the normalizing constant and $\mathbf{A}$ is a symmetric $p \times p$ matrix, with the constraint $tr(\mathbf{A}) = 0$. Note that the combination of the two components in the exponential function shown above, provides enough flexibility to represent a large variety of ellipsoidal distributions (same as Bingham) but without the antipodally symmetric constraint (same as in von Mises-Fisher).

2.2 Parameter Estimation

To use each of these distributions, we need to first estimate their parameters from a training data-set, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$; $\mathbf{X}$ a $p \times n$ matrix, with $\mathbf{x}_i \in S^{p-1}$ the sample vectors.

If one assumes that the samples in $\mathbf{X}$ arise from a von Mises-Fisher distribution, we will need to estimate the concentration parameter κ and the (unit) mean direction μ. The most common way to estimate these parameters is to use the maximum likelihood estimates (m.l.e.). The sample mean direction $\hat{\mu}$ is given by

$$\hat{\mu} = \frac{\bar{\mathbf{x}}}{\|\bar{\mathbf{x}}\|}, \tag{4}$$

where $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$ is the average feature vector. It can be shown that

$$c_{MF} = \left(\frac{\kappa}{2}\right)^{p/2-1} \frac{1}{(2\pi)^{p/2} I_{p/2-1}(\kappa)}$$

in (1), where $I_v(.)$ denotes the modified Bessel function of the first kind and of order v (Banerjee et al., 2005).[2] By substituting this normalization constant in (1) and calculating the expected value of $\mathbf{x}$ (by integrating the pdf over the surface of S^{p-1}), we obtain $\|\bar{\mathbf{x}}\| = \frac{I_{p/2}(\hat{\kappa})}{I_{p/2-1}(\hat{\kappa})}$. Unfortunately, equations defining a ratio of Bessel functions cannot be inverted and, hence, approximation methods need to be defined for $\hat{\kappa}$. Banerjee et al. (2005) have recently proposed one such approximation which can be applied regardless of the dimensionality of the data,

$$\hat{\kappa} = \frac{\|\bar{\mathbf{x}}\|p - \|\bar{\mathbf{x}}\|^3}{1 - \|\bar{\mathbf{x}}\|^2}.$$

2. The modified Bessel function of the first kind is proportional to the contour integral of the exponential function defined in (1) over the $(p-1)$-dimensional sphere S^{p-1}.

This approximation makes the parameter $\hat{\kappa}$ directly dependent on the training data and, hence, can be easily computed.

For the Bingham distribution, the normalizing constant, $c_B^{-1} = \int_{S^{p-1}} \exp\left\{\mathbf{x}^T\mathbf{A}\mathbf{x}\right\}\, d\mathbf{x}$, requires that we estimate the parameters defined in $\mathbf{A}$. Since $\mathbf{A}$ is a symmetric matrix, its spectral decomposition can be written as $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$, where $\mathbf{Q} = (\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_p)$ is a matrix whose columns $\mathbf{q}_i$ correspond to the eigenvectors of $\mathbf{A}$ and $\mathbf{\Lambda} = diag\,(\lambda_1, \ldots, \lambda_p)$ is the $p \times p$ diagonal matrix of corresponding eigenvalues. This allows us to calculate the log-likelihood of the data by adding the log version of (2) over all samples in $\mathbf{X}$, $\mathcal{L}(\mathbf{Q}, \Lambda) = n\, tr(\mathbf{S}\mathbf{Q}\Lambda\mathbf{Q}^T) + n\ln(c_B(p, \Lambda))$; where $\mathbf{S} = n^{-1}\,\mathbf{X}\mathbf{X}^T$ is the sample autocorrelation matrix (sometimes also referred to as scatter matrix). Since the $tr(\mathbf{S}\mathbf{A})$ is maximized when the eigenvectors of $\mathbf{S}$ and $\mathbf{A}$ are the same, the m.l.e. of $\mathbf{Q}$ (denoted $\hat{\mathbf{Q}}$) is given by the eigenvector decomposition of the autocorrelation matrix, $\mathbf{S} = \hat{\mathbf{Q}}\hat{\Lambda}_S\hat{\mathbf{Q}}$; where $\hat{\Lambda}_S$ is the eigenvalue matrix of $\mathbf{S}$. Unfortunately, the same does not apply to the estimation of the eigenvalues Λ, because these depend on $\mathbf{S}$ and c_B. Note that in order to calculate the normalizing constant c_B we need to know Λ, but to compute Λ we need to know c_B. This chicken-and-egg problem needs to be solved using iterative approaches or optimization algorithms. To define such iterative approaches, we need to calculate the derivative of c_B. Since there are no known ways to express Λ as a function of the derivative of $c_B(p, \Lambda)$, approximations for c_B (which permit such a dependency) are necessary. Kume and Wood (2005) have recently proposed a saddlepoint approximation that can be used for this purpose. In their approach, the estimation of the eigenvalues is given by the following optimization problem

$$\arg\max_{\Lambda}\; n\, tr(\hat{\Lambda}_S\Lambda) - n\ln(\hat{c}_B(\Lambda)), \tag{5}$$

where $\hat{c}_B(\Lambda)$ is now the estimated normalizing constant given by the saddlepoint approximation of the density function of the 2-norm of $\mathbf{x}$.

The estimation of the parameters of the Fisher-Bingham distribution comes at an additional cost given by the large number of parameters that need to be estimated. For example, the normalizing constant c_{FB} depends on κ, μ and $\mathbf{A}$, making the problem even more complex than that of the Bingham distribution. Hence, approximation methods are once more required. One such approximation is given by Kent (Kent, 1982), where it is assumed that the data is highly concentrated (i.e., κ is large) or that the data is distributed more or less equally about every dimension (i.e., the distribution is almost circularly symmetric). In this case, the mean direction μ is estimated using (4) and the estimate of the parameter matrix (for the 3-dimensional case) is given by $\mathbf{A} = \beta(\mathbf{q}_1\mathbf{q}_1^T - \mathbf{q}_2\mathbf{q}_2^T)$; where $\mathbf{q}_1$ and $\mathbf{q}_2$ are the two eigenvectors associated to the two largest eigenvalues ($\lambda_1 \geq \lambda_2$) of $\bar{\mathbf{S}}$, $\bar{\mathbf{S}}$ is the scatter matrix calculated on the null space of the mean direction, and β is the parameter that allows us to deviate from the circle defined by κ.[3] Note that the two eigenvectors $\mathbf{q}_1$ and $\mathbf{q}_2$ are constrained to be orthogonal to the unit vector $\hat{\mu}$ describing the mean direction. To estimate the value for κ and β, Kent proposes to further assume the data can be locally represented as Gaussian distributions on S^{p-1} and shows that in the three dimensional case $\hat{\beta} \cong \frac{1}{2}\left((2 - 2\|\bar{\mathbf{x}}\| - r_2)^{-1} + (2 - 2\|\bar{\mathbf{x}}\| + r_2)^{-1}\right)$ and $\hat{\kappa} \cong (2-2\|\bar{\mathbf{x}}\|-r_2)^{-1}+(2-2\|\bar{\mathbf{x}}\|+r_2)^{-1}$, where $r_2 = \lambda_1 - \lambda_2$.

3. Recall that the first part of the definition of the Fisher-Bingham pdf is the same as that of the von Mises-Fisher, $\kappa\mu^T\mathbf{x}$, which defines a small circle on S^{p-1}, while the second component $\mathbf{x}^T\mathbf{A}\mathbf{x}$ allows us to deviate from the circle and represent a large variety of ellipsoidal pdfs.

The Kent distribution is one of the most popular distribution models for the estimation of 3-dimensional spherical data, since it has fewer parameters to be estimated than Fisher-Bingham and can model any ellipsoidally symmetric distribution. A more recent and general approximation for Fisher-Bingham distributions is given in (Kume and Wood, 2005), but this requires the estimate of a larger number of parameters, a cost to be considered.

As summarized above, the actual values of the parameters of spherical distributions can rarely be computed, and approximations are needed. Furthermore, we have seen that most of these approximation algorithms require assumptions that may not be applicable to our problem. In the case of the Kent distribution, these assumptions are quite restrictive. When the assumptions do not hold, we cannot make use of these spherical pdfs.

2.3 Distance Calculation

The probability ("distance") of a new (test) vector $\mathbf{x}$ to belong to a given distribution can be defined as (inversely) proportional to the likelihood or log-likelihood of $\mathbf{x}$. For instance, the pdf of the p-dimensional Gaussian $N(\mathbf{m}, \Sigma)$ is $f(\mathbf{x}|\mathbf{m}, \Sigma) = c_N(\Sigma) \exp\{-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \Sigma^{-1} (\mathbf{x} - \mathbf{m})\}$, where $\mathbf{m}$ is the mean, Σ is the sample covariance matrix of the data and $c_N^{-1}(\Sigma) = (2\pi)^{p/2} |\Sigma|^{1/2}$ is the normalizing constant. When the priors of each class are the same, the optimum "distance" measurement (in the Bayes sense) of a point $\mathbf{x}$ to $f(\mathbf{x}|\mathbf{m}, \Sigma)$ as derived by the Bayes rule is the negative of the log-likelihood

$$d_N^2(\mathbf{x}) = -\ln f(\mathbf{x}|\mathbf{m}, \Sigma) = \frac{1}{2} (\mathbf{x} - \mathbf{m})^T \Sigma^{-1} (\mathbf{x} - \mathbf{m}) - \ln(c_N(\Sigma)). \tag{6}$$

Similarly, we can define the distance of a test sample to each of the spherical distributions defined above (i.e., von Mises-Fisher, Bingham and Fisher-Bingham) as

$$\begin{aligned} d_{MF}^2(\mathbf{x}) &= -\kappa \mu^T \mathbf{x} - \ln(c_{MF}(p, \kappa)), & (7) \\ d_B^2(\mathbf{x}) &= -\mathbf{x}^T \mathbf{A} \mathbf{x} - \ln(c_B(p, \mathbf{A})), & (8) \\ d_{FB}^2(\mathbf{x}) &= -\kappa \mu^T \mathbf{x} - \mathbf{x}^T \mathbf{A} \mathbf{x} - \ln(c_{FB}(\kappa, \mathbf{A})). & (9) \end{aligned}$$

As seen in Section 2.2, the difficulty with the distance measures defined in (7-9) will be given by the estimation of the parameters of our distribution (e.g., μ, κ, and $\mathbf{A}$), because this is usually complex and sometimes impossible. This is the reason why most researchers prefer to use the Gaussian model and its corresponding distance measure defined in (6) instead. The question remains: *are the classification errors obtained using the spherical distributions defined above lower than those obtained when using the Gaussian approximation? And, if so, when?*

The rest of this paper addresses this general question. In particular, we show that when the data distributions conform to a specific relation (which we call spherical-homoscedastic), the classification errors will be the same. However, in the most general case, they need not be.

3. Spherical-Homoscedastic Distributions

If the distributions of each class are known, the optimal classifier is given by the Bayes Theorem. Furthermore, when the class priors are equal, this decision rule simplifies to

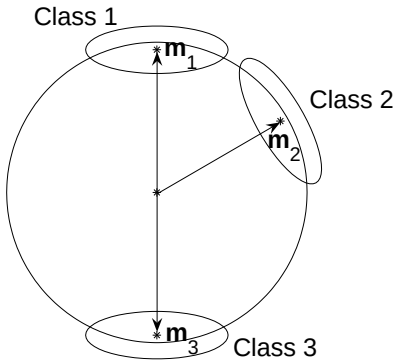

Figure 2: *Assume we model the data of three classes laying on S^{p-1} using three Gaussian distributions. In this case, each set of Gaussian distributions can only be homoscedastic if the mean feature vector of one class is the same as that of the others up to a sign. In this figure, Class 1 and 3 are homoscedastic, but classes 1,2 and 3 are not. Classes 1, 2 and 3 are however spherical-homoscedastic.*

the comparison of the likelihoods (maximum likelihood classification), $p(\mathbf{x}|w_i)$; where w_i specifies the i^{th} class. In the rest of this paper, we will make the assumption of equal priors (i.e., $P(w_i) = P(w_j)\ \forall i, j$). An alternative way to calculate the likelihood of an observation $\mathbf{x}$ to belong to a class, is to measure the log-likelihood-based distance (e.g., d_N^2 in the case of a Gaussian distribution). In the spherical case, the distances defined in Section 2.3 can be used.

In the Gaussian case, we say that a set of r Gaussians, $\{N_1(\mathbf{m}_1, \Sigma_1), \ldots, N_r(\mathbf{m}_r, \Sigma_r)\}$, are *homoscedastic* if their covariance matrices are all the same (i.e. $\Sigma_1 = \cdots = \Sigma_r$). Homoscedastic Gaussian distributions are relevant, because their Bayes decision boundaries are given by hyperplanes.

However, when all feature vectors are restricted to lay on the surfaces of a hypersphere, the definition of homoscedasticity given above becomes too restrictive. For example, if we use Gaussian pdfs to model some spherical data that is ellipsoidally symmetric about its mean, then only those distributions that have the same mean up to a sign can be homoscedastic. This is illustrated in Figure 2. Although the three classes shown in this figure have the same covariance matrix up to a rotation, only the ones that have the same mean up to a sign (i.e., Class 1 and 3) have the exact same covariance matrix. Hence, only classes 1 and 3 are said to be homoscedastic. Nonetheless, the decision boundaries for each pair of classes in Figure 2 are all hyperplanes. Furthermore, we will show that these hyperplanes (given by approximating the original distributions with Gaussians) are generally the same as those obtained using the Bayes Theorem on the true underlying distributions. Therefore, it is important to define a new and more general type of homoscedasticity that is rotational invariant.

Definition 1 *Two distributions (f_1 and f_2) are said to be spherical-homoscedastic if the Bayes decision boundary between f_1 and f_2 is given by one or more hyperplanes and the variances in f_1 are the same as those in f_2.*

Recall that the variance of the vMF distribution is defined using a single parameter, κ. This means that in the vMF case, we will only need to impose that all concentration parameters be the same. For the other cases, these will be defined by κ and the eigenvalues of $\mathbf{A}$. This is what is meant by "the variances" in our definition above.

Further, in this paper, we will work on the case where the two distributions (f_1 and f_2) are of the same form, that is, two Gaussian, vMF, Bingham or Kent distributions.

Our main goal in the rest of this section, is to demonstrate that the linear decision boundaries (given by the Bayes Theorem) of a pair of spherical-homoscedastic von Mises-Fisher, Bingham or Kent, are the same as those obtained when these are assumed to be Gaussian. We start with the study of the Gaussian distribution.

Theorem 2 *Let two Gaussian distributions $N_1(\mathbf{m}, \Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m}, \mathbf{R}^T\Sigma\mathbf{R})$ model the spherical data of two classes on S^{p-1}; where $\mathbf{m}$ is the mean, Σ is the covariance matrix (which is assumed to be full ranked), and $\mathbf{R} \in SO(p)$ is a rotation matrix. Let $\mathbf{R}$ be spanned by two of the eigenvectors of Σ, $\mathbf{v}_1$ and $\mathbf{v}_2$, and let one of these eigenvectors define the same direction as $\mathbf{m}$ (i.e., $\mathbf{v}_i = \mathbf{m}/\|\mathbf{m}\|$ for i equal to 1 or 2). Then, $N_1(\mathbf{m}, \Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m}, \mathbf{R}^T\Sigma\mathbf{R})$ are spherical-homoscedastic.*

Proof We want to prove that the Bayes decision boundaries are hyperplanes. This boundary is given when the ratio of the log-likelihoods of $N_1(\mathbf{m}, \Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m}, \mathbf{R}^T\Sigma\mathbf{R})$ equals one. Formally,

$$\begin{aligned}&\ln(c_N(\Sigma)) - \frac{1}{2}(\mathbf{x}-\mathbf{m})^T\Sigma^{-1}(\mathbf{x}-\mathbf{m}) = \\ &\ln(c_N(\mathbf{R}^T\Sigma\mathbf{R})) - \frac{1}{2}\left(\mathbf{x}-\mathbf{R}^T\mathbf{m}\right)^T(\mathbf{R}^T\Sigma\mathbf{R})^{-1}\left(\mathbf{x}-\mathbf{R}^T\mathbf{m}\right).\end{aligned}$$

Since for any function f we know that $f(|\mathbf{R}^T\Sigma\mathbf{R}|) = f(|\Sigma|)$, the constant parameter $c_N(|\mathbf{R}^T\Sigma\mathbf{R}|) = c_N(|\Sigma|)$; where $|\mathbf{M}|$ is the determinant of $\mathbf{M}$. Furthermore, since the normalizing constant c_N only depends on the determinant of the covariance matrix, we know that $c_N(\mathbf{R}^T\Sigma\mathbf{R}) = c_N(\Sigma)$. This allows us to simplify our previous equation to

$$\begin{aligned}(\mathbf{x}-\mathbf{m})^T\Sigma^{-1}(\mathbf{x}-\mathbf{m}) &= \left(\mathbf{x}-\mathbf{R}^T\mathbf{m}\right)^T\mathbf{R}^T\Sigma^{-1}\mathbf{R}\left(\mathbf{x}-\mathbf{R}^T\mathbf{m}\right) \\ &= (\mathbf{R}\mathbf{x}-\mathbf{m})^T\Sigma^{-1}(\mathbf{R}\mathbf{x}-\mathbf{m}).\end{aligned}$$

Writing this equation in an open form,

$$\begin{aligned}\mathbf{x}^T\Sigma^{-1}\mathbf{x} - 2\mathbf{x}^T\Sigma^{-1}\mathbf{m} + \mathbf{m}^T\Sigma^{-1}\mathbf{m} &= (\mathbf{R}\mathbf{x})^T\Sigma^{-1}(\mathbf{R}\mathbf{x}) - 2(\mathbf{R}\mathbf{x})^T\Sigma^{-1}\mathbf{m} + \mathbf{m}^T\Sigma^{-1}\mathbf{m} \\ \mathbf{x}^T\Sigma^{-1}\mathbf{x} - 2\mathbf{x}^T\Sigma^{-1}\mathbf{m} &= (\mathbf{R}\mathbf{x})^T\Sigma^{-1}(\mathbf{R}\mathbf{x}) - 2(\mathbf{R}\mathbf{x})^T\Sigma^{-1}\mathbf{m}. \qquad (10)\end{aligned}$$

Let the spectral decomposition of Σ be $\mathbf{V}\Lambda\mathbf{V}^T = (\mathbf{v}_1, \cdots, \mathbf{v}_p)\, diag(\lambda_1, \cdots, \lambda_p)\, (\mathbf{v}_1, \cdots, \mathbf{v}_p)^T$.

Now use the assumption that $\mathbf{m}$ is orthogonal to all the eigenvectors of Σ except one (i.e., $\mathbf{v}_j = \mathbf{m}/\|\mathbf{m}\|$ for some j). More formally, $\mathbf{m}^T\mathbf{v}_i = 0$ for all $i \neq j$ and $\mathbf{m}^T\mathbf{v}_j = s\|\mathbf{m}\|$, where $s = \pm 1$. Without loss of generality, let $j = 1$, which yields $\Sigma^{-1}\mathbf{m} = \mathbf{V}\Lambda^{-1}\mathbf{V}^T\mathbf{m} = {\lambda_1}^{-1}\mathbf{m}$. Substituting this in (10) we get

$$\mathbf{x}^T\Sigma^{-1}\mathbf{x} - 2\lambda_1^{-1}\mathbf{x}^T\mathbf{m} = (\mathbf{R}\mathbf{x})^T\Sigma^{-1}(\mathbf{R}\mathbf{x}) - 2\lambda_1^{-1}(\mathbf{R}\mathbf{x})^T\mathbf{m}.$$

Writing Σ^{-1} in an open form,

$$\sum_{i=1}^{p} \lambda_i^{-1} \left(\mathbf{x}^T\mathbf{v}_i\right)^2 - 2\lambda_1^{-1}\mathbf{x}^T\mathbf{m} = \sum_{i=1}^{p} \lambda_i^{-1} \left((\mathbf{R}\mathbf{x})^T\mathbf{v}_i\right)^2 - 2\lambda_1^{-1}(\mathbf{R}\mathbf{x})^T\mathbf{m}$$
$$\sum_{i=1}^{p} \left(\lambda_i^{-1} \left[\left(\mathbf{x}^T\mathbf{v}_i\right)^2 - \left(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_i\right)^2\right]\right) - 2\lambda_1^{-1}\mathbf{x}^T\mathbf{m} + 2\lambda_1^{-1}\mathbf{x}^T\mathbf{R}^T\mathbf{m} = 0.$$

Recall that the rotation is constrained to be in the subspace spanned by two of the eigenvectors of Σ. One of these eigenvectors must be $\mathbf{v}_1$. Let the other eigenvector be $\mathbf{v}_2$. Then, $\mathbf{x}^T\mathbf{R}^T\mathbf{v}_i = \mathbf{x}^T\mathbf{v}_i$ for $i \neq \{1, 2\}$. This simplifies our last equation to

$$\lambda_1^{-1}\left[\left(\mathbf{x}^T\mathbf{v}_1\right)^2 - \left(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1\right)^2\right] + \lambda_2^{-1}\left[\left(\mathbf{x}^T\mathbf{v}_2\right)^2 - \left(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_2\right)^2\right] + 2\lambda_1^{-1}\left(\mathbf{x}^T\mathbf{R}^T\mathbf{m} - \mathbf{x}^T\mathbf{m}\right) = 0. \tag{11}$$

Noting that $\left(\mathbf{x}^T\mathbf{v}_1\right)^2 - \left(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1\right)^2 = (\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{v}_1)(-\mathbf{x}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{R}^T\mathbf{v}_1)$, and that $\mathbf{m} = s\|\mathbf{m}\|\mathbf{v}_1$, allows us to rewrite (11) as

$$\begin{aligned}
&\lambda_1^{-1}(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{v}_1)(2\|\mathbf{m}\|s - \mathbf{x}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{R}^T\mathbf{v}_1) \\
&+\lambda_2^{-1}\left(\left(\mathbf{x}^T\mathbf{v}_2\right)^2 - \left(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_2\right)^2\right) = 0 \\
&\lambda_1^{-1}(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{v}_1)(2\|\mathbf{m}\|s - \mathbf{x}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{R}^T\mathbf{v}_1) \\
&+\lambda_2^{-1}(\mathbf{x}^T(\mathbf{v}_2 - \mathbf{R}^T\mathbf{v}_2))(\mathbf{x}^T(\mathbf{v}_2 + \mathbf{R}^T\mathbf{v}_2)) = 0.
\end{aligned} \tag{12}$$

In addition, we know that the rotation matrix $\mathbf{R}$ defines a rotation in the $(\mathbf{v}_1, \mathbf{v}_2)$-plane. Assume that $\mathbf{R}$ rotates the vector $\mathbf{v}_1$ θ degrees in the clockwise direction yielding $\mathbf{R}^T\mathbf{v}_1$. Similarly, $\mathbf{v}_2$ becomes $\mathbf{R}^T\mathbf{v}_2$. From Figure 3 we see that

$$\begin{aligned}
\mathbf{v}_2 - \mathbf{R}^T\mathbf{v}_2 &= 2\mathbf{u}\cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right), \quad \mathbf{v}_2 + \mathbf{R}^T\mathbf{v}_2 = 2\mathbf{w}\cos\left(\frac{\theta}{2}\right), \\
\mathbf{R}^T\mathbf{v}_1 - \mathbf{v}_1 &= 2\mathbf{w}\cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right), \quad \mathbf{R}^T\mathbf{v}_1 + \mathbf{v}_1 = 2\mathbf{u}\cos\left(\frac{\theta}{2}\right),
\end{aligned}$$

where $\mathbf{u}$ and $\mathbf{w}$ are the unit vectors as shown in Figure 3. Therefore,

$$\mathbf{v}_2 + \mathbf{R}^T\mathbf{v}_2 = (\mathbf{R}^T\mathbf{v}_1 - \mathbf{v}_1)\cot\left(\frac{\theta}{2}\right) \quad \text{and} \quad \mathbf{v}_2 - \mathbf{R}^T\mathbf{v}_2 = (\mathbf{R}^T\mathbf{v}_1 + \mathbf{v}_1)\tan\left(\frac{\theta}{2}\right).$$

If we use these results in (12), we find that

$$\begin{aligned}
&\lambda_1^{-1}(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{v}_1)(2\|\mathbf{m}\|s - \mathbf{x}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{R}^T\mathbf{v}_1) \\
+ \quad &\lambda_2^{-1}(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1 - \mathbf{x}^T\mathbf{v}_1)\cot\left(\frac{\theta}{2}\right)(\mathbf{x}^T\mathbf{R}^T\mathbf{v}_1 + \mathbf{x}^T\mathbf{v}_1)\tan\left(\frac{\theta}{2}\right) = 0,
\end{aligned}$$

which can be reorganized to

$$\left[\mathbf{x}^T(\mathbf{R}^T\mathbf{v}_1 - \mathbf{v}_1)\right]\left[(\lambda_2^{-1} - \lambda_1^{-1})\mathbf{x}^T(\mathbf{R}^T\mathbf{v}_1 + \mathbf{v}_1) + 2\lambda_1^{-1}\|\mathbf{m}\|s\right] = 0. \tag{13}$$

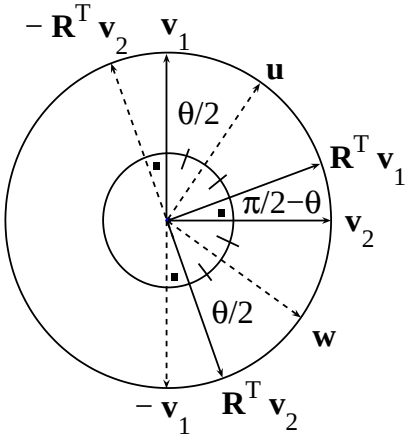

Figure 3: *Shown here are two orthonormal vectors,* $\mathbf{v}_1$ *and* $\mathbf{v}_2$*, and their rotated versions,* $\mathbf{R}^T\mathbf{v}_1$ *and* $\mathbf{R}^T\mathbf{v}_2$*. We see that* $\mathbf{R}^T\mathbf{v}_1+\mathbf{v}_1 = 2\mathbf{u}\cos\left(\frac{\theta}{2}\right)$*,* $\mathbf{R}^T\mathbf{v}_2+\mathbf{v}_2 = 2\mathbf{w}\cos\left(\frac{\theta}{2}\right)$*,* $\mathbf{R}^T\mathbf{v}_1-\mathbf{v}_1 = 2\mathbf{w}\cos\left(\frac{\pi}{2}-\frac{\theta}{2}\right)$ *and* $\mathbf{v}_2-\mathbf{R}^T\mathbf{v}_2 = 2\mathbf{u}\cos\left(\frac{\pi}{2}-\frac{\theta}{2}\right)$*.*

The two possible solutions of this equation provide the two hyperplanes for the Bayes classifier. The first hyperplane,

$$\mathbf{x}^T\frac{(\mathbf{R}^T\mathbf{v}_1-\mathbf{v}_1)}{2\sin\left(\frac{\theta}{2}\right)} = 0, \tag{14}$$

passes through the origin and its normal vector is $(\mathbf{R}^T\mathbf{v}_1-\mathbf{v}_1)/2\sin(\frac{\theta}{2})$. The second hyperplane,

$$\mathbf{x}^T\frac{\mathbf{R}^T\mathbf{v}_1+\mathbf{v}_1}{2\cos\left(\frac{\theta}{2}\right)} + \frac{\lambda_1^{-1}\|\mathbf{m}\|s}{(\lambda_2^{-1}-\lambda_1^{-1})\cos\left(\frac{\theta}{2}\right)} = 0, \tag{15}$$

has a bias equal to

$$\frac{\lambda_1^{-1}\|\mathbf{m}\|s}{(\lambda_2^{-1}-\lambda_1^{-1})\cos\left(\frac{\theta}{2}\right)},$$

and its normal is $\left(\mathbf{R}^T\mathbf{v}_1+\mathbf{v}_1\right)/2\cos\left(\theta/2\right)$. ■

The result above, shows that when the rotation matrix is spanned by two of the eigenvectors of Σ, then N_1 and N_2 are spherical-homoscedastic. The reader may have noted though, that there exist some Σ (e.g. $\tau^2\mathbf{I}$) which are less restrictive on $\mathbf{R}$. The same applies to spherical distributions. We start our study with the case where the true underlying distributions of the data are von Mises-Fisher.

Theorem 3 *Two von Mises-Fisher distributions* $M_1(\mu,\kappa)$ *and* $M_2(\mathbf{R}^T\mu,\kappa)$ *are spherical-homoscedastic if* $\mathbf{R}\in SO(p)$*.*

Proof As above, the Bayes decision boundary is given when the ratio between the log-likelihood of $M_1(\mu,\kappa)$ and that of $M_2(\mathbf{R}^T\mu,\kappa)$ is equal to one. This means that

$$\begin{aligned}\kappa\mu^T\mathbf{x}+\ln\left(c_{MF}(\kappa)\right) &= \kappa(\mathbf{R}^T\mu)^T\mathbf{x}+\ln\left(c_{MF}(\kappa)\right)\\ \kappa(\mathbf{x}^T\mathbf{R}^T\mu-\mathbf{x}^T\mu) &= 0. \end{aligned} \tag{16}$$

We see that (16) defines the decision boundary between $M_1(\mu,\kappa)$ and $M_2(\mathbf{R}^T\mu,\kappa)$ and that this is a hyperplane[4] with normal

$$\frac{\mathbf{R}^T\mu-\mu}{2\cos\left(\omega/2\right)},$$

where ω is the magnitude of the rotation angle.[5] Hence, $M_1(\mu,\kappa)$ and $M_2(\mathbf{R}^T\mu,\kappa)$ are spherical-homoscedastic. ■

We now want to show that the Bayes decision boundary of two spherical-homoscedastic vMF is the same as the classification boundary obtained when these distributions are modelled using Gaussian pdfs. However, Theorem 2 provides two hyperplanes – those given in Eqs. (14-15). We need to show that one of these equations is the same as the hyperplane given in Eq. (16), and that the other equation gives an irrelevant decision boundary. The irrelevant hyperplane is that in (15). To show why this equation is not relevant for classification, we will demonstrate that this hyperplane is always outside S^{p-1} and, hence, cannot divide the spherical data into more than one region.

Proposition 4 *When modelling the data of two spherical-homoscedastic von Mises-Fisher distributions, $M_1(\mu,\kappa)$ and $M_2(\mathbf{R}^T\mu,\kappa)$, using two Gaussian distributions, $N_1(\mathbf{m},\Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m},\mathbf{R}^T\Sigma\mathbf{R})$, the Bayes decision boundary will be given by the two hyperplanes defined in Eqs. (14-15). However, the hyperplane given in (15) does not intersect with the sphere and can be omitted for classification purposes.*

Proof Recall that the bias of the hyperplane given in (15) was

$$b_2=\frac{s\|\mathbf{m}\|}{((\lambda_1/\lambda_2)-1)\cos\left(\theta/2\right)}. \tag{17}$$

We need to show that the absolute value of this bias is greater than one; i.e. $|b_2|>1$.

We know from (Dryden and Mardia, 1998) that if $\mathbf{x}$ is distributed as $M(\mu,\kappa)$, then

$$\mathbf{m}=E(\mathbf{x})=A_p(\kappa)\mu,$$

and the covariance matrix of $\mathbf{x}$ is given by

$$\Sigma=A_p'(\kappa)\mu\mu^T+\frac{A_p(\kappa)}{\kappa}(\mathbf{I}_p-\mu\mu^T),$$

4. Since this hyperplane is constrained with $\|\mathbf{x}\|=1$, the decision boundary will define a great circle on the sphere.
5. Note that since the variance about every direction orthogonal to μ is equal to τ^2, all rotations can be expressed as a planar rotation spanned by μ and any $\mu^\perp$ (where $\mu^\perp$ is a vector orthogonal to μ) .

where $A_p(\kappa) = I_{p/2}(\kappa)/I_{p/2-1}(\kappa)$ and $A'_p(\kappa) = 1 - A_p^2(\kappa) - \frac{p-1}{\kappa}A_p(\kappa)$. Note that the first eigenvector of the matrix defined above is aligned with the mean direction, and that the rest are orthogonal to it. Furthermore, the first eigenvalue of this matrix is

$$\lambda_1 = 1 - A_p^2(\kappa) - \frac{p-1}{\kappa}A_p(\kappa),$$

and the rest are all equal and defined as

$$\lambda_i = \frac{A_p(\kappa)}{\kappa}, \quad \forall i > 1.$$

Substituting the above calculated terms in (17) yields

$$\hat{b}_2(\kappa) = \frac{A_p(\kappa)}{\frac{1-A_p^2(\kappa)-\frac{p-1}{\kappa}A_p(\kappa)}{\frac{A_p(\kappa)}{\kappa}} - 1} = \frac{A_p^2(\kappa)}{\kappa\left(1 - A_p^2(\kappa) - \frac{p}{\kappa}A_p(\kappa)\right)} \tag{18}$$

with $b_2 = \frac{s\,\hat{b}_2(\kappa)}{\cos(\theta/2)}$.

Note that we can rewrite $A_p(\kappa)$ as

$$A_p(\kappa) = \frac{I_\nu(\kappa)}{I_{\nu-1}(\kappa)},$$

where $\nu = p/2$. Moreover, the recurrence relation between modified Bessel functions states that $I_{\nu-1}(\kappa) - I_{\nu+1}(\kappa) = \frac{2\nu}{\kappa}I_\nu(\kappa)$, which is the same as $1 - (I_{\nu+1}(\kappa)/I_{\nu-1}(\kappa)) = (2\nu I_\nu(\kappa))\,/\,(\kappa I_{\nu-1}(\kappa))$. This can be combined with the result shown above to yield

$$\frac{p}{\kappa}A_p(\kappa) = 1 - \frac{I_{\nu+1}(\kappa)}{I_{\nu-1}(\kappa)}.$$

By substituting these terms in (18), one obtains

$$\hat{b}_2(\kappa) = \frac{\left(\frac{I_\nu(\kappa)}{I_{\nu-1}(\kappa)}\right)^2}{\kappa\left(-\left(\frac{I_\nu(\kappa)}{I_{\nu-1}(\kappa)}\right)^2 + \frac{I_{\nu+1}(\kappa)I_{\nu-1}(\kappa)}{I_{\nu-1}^2(\kappa)}\right)} = \frac{I_\nu^2(\kappa)}{\kappa\left(-I_\nu^2(\kappa) + I_{\nu+1}(\kappa)I_{\nu-1}(\kappa)\right)}.$$

Using the bound defined by Joshi (1991) (see Eq. (3.14), page 340), $0 < I_v^2(\kappa) - I_{v-1}(\kappa)I_{v+1}(\kappa) < \frac{I_v^2(\kappa)}{v+\kappa}$ $(\forall\kappa > 0)$, we have

$$0 < I_\nu^2(\kappa) - I_{\nu-1}(\kappa)I_{\nu+1}(\kappa) < \frac{I_\nu^2(\kappa)}{\nu+\kappa}$$

$$\frac{I_\nu^2(\kappa)}{\kappa(I_\nu^2(\kappa) - I_{\nu-1}(\kappa)I_{\nu+1}(\kappa))} > \frac{(\nu+\kappa)}{\kappa}$$

$$\frac{I_\nu^2(\kappa)}{\kappa(-I_\nu^2(\kappa) + I_{\nu-1}(\kappa)I_{\nu+1}(\kappa))} < \frac{(\nu+\kappa)}{-\kappa} < -1.$$

This upper-bound shows that $|b_2| = \left|\frac{\hat{b}_2(\kappa)}{\cos(\theta/2)}\right| > 1$. This means that the second hyperplane will not divide the data into more than one class and therefore can be ignored for classification purposes. ■

From our proof above, we note that Σ is spanned by μ and a set of $p-1$ basis vectors that are orthogonal to μ. Furthermore, since a vMF is circularly symmetric around μ, these basis vectors can be represented by any orthonormal set of vectors orthogonal to μ. Next, note that the mean direction of M_2 can be written as $\mu_2 = \mathbf{R}^T\mu$. This means that $\mathbf{R}$ can be spanned by μ and any unit vector orthogonal to μ (denoted $\mu^\perp$) – such as an eigenvector of Σ. Therefore, N_1 and N_2 are spherical-homoscedastic. We can summarize this results in the following.

Corollary 5 *If we model two spherical-homoscedastic von Mises-Fisher, $M_1(\mu,\kappa)$ and $M_2(\mathbf{R}^T\mu,\kappa)$, with their corresponding Gaussian approximations $N_1(\mathbf{m},\Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m},\mathbf{R}^T\Sigma\mathbf{R})$, then N_1 and N_2 are also spherical-homoscedastic.*

This latest result is important to show that the Bayes decision boundaries of two spherical-homoscedastic vMFs can be calculated exactly using the Gaussian model.

Theorem 6 *The Bayes decision boundary of two spherical-homoscedastic von Mises-Fisher, $M_1(\mu,\kappa)$ and $M_2(\mathbf{R}^T\mu,\kappa)$, is the same as that given in (14), which is obtained when modelling $M_1(\mu,\kappa)$ and $M_2(\mathbf{R}^T\mu,\kappa)$ using the two Gaussian distributions $N_1(\mathbf{m},\Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m},\mathbf{R}^T\Sigma\mathbf{R})$.*

Proof From Corollary 5 we know that $N_1(\mathbf{m},\Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m},\mathbf{R}^T\Sigma\mathbf{R})$ are spherical-homoscedastic. And, from Proposition 4, we know that the hyperplane decision boundary given by Eq. (15) is outside the sphere and can be eliminated. In the proof of Proposition 4 we also showed that $\mathbf{v}_1 = \mu$. This means, (14) can be written as

$$\mathbf{x}^T\frac{(\mathbf{R}^T\mu-\mu)}{2\sin(\theta/2)} = 0. \tag{19}$$

The decision boundary for two spherical-homoscedastic vMF was derived in Theorem 3, where it was shown to be

$$\kappa(\mathbf{x}^T R^T\mu - \mathbf{x}^T\mu) = 0. \tag{20}$$

We note that the normal vectors of Eqs. (19-20) are the same and that both biases are zero. Therefore, the two equations define the same great circle on S^{p-1}; i.e., they yield the same classification results. ■

When the vMF model is not flexible enough to represent our data, we need to use a more general definition such as that given by Bingham. We will now study under which conditions two Bingham distributions are spherical-homoscedastic and, hence, can be efficiently approximated with Gaussians.

Theorem 7 *Two Bingham distributions, $B_1(\mathbf{A})$ and $B_2(\mathbf{R}^T\mathbf{A}\mathbf{R})$, are spherical-homoscedastic if $\mathbf{R} \in SO(p)$ defines a planar rotation in the subspace spanned by any two of the eigenvectors of $\mathbf{A}$, say $\mathbf{q}_1$ and $\mathbf{q}_2$.*

Proof Making the ratio of the log-likelihood equations equal to one yields

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = \mathbf{x}^T\mathbf{R}^T\mathbf{A}\mathbf{R}\mathbf{x}. \tag{21}$$

Since the rotation is defined in the subspace spanned by $\mathbf{q}_1$ and $\mathbf{q}_2$ and $\mathbf{A} = \mathbf{Q}\Lambda\mathbf{Q}^T$, then the above equation can be expressed (in open form) as $\sum_{i=1}^{p} \lambda_i(\mathbf{x}^T\mathbf{q}_i)^2 = \sum_{i=1}^{p} \lambda_i(\mathbf{x}^T\mathbf{R}^T\mathbf{q}_i)^2$. In addition, $\mathbf{R}^T\mathbf{q}_i = \mathbf{q}_i$ for $i > 2$, which simplifies our equation to

$$\sum_{i=1}^{p} \lambda_i(\mathbf{x}^T\mathbf{q}_i)^2 = \sum_{i=1}^{2} \lambda_i(\mathbf{x}^T\mathbf{R}^T\mathbf{q}_i)^2 + \sum_{i=3}^{p} \lambda_i(\mathbf{x}^T\mathbf{q}_i)^2$$

$$\lambda_1\left((\mathbf{x}^T\mathbf{q}_1)^2 - (\mathbf{x}^T\mathbf{R}^T\mathbf{q}_1)^2\right) + \lambda_2\left((\mathbf{x}^T\mathbf{q}_2)^2 - (\mathbf{x}^T\mathbf{R}^T\mathbf{q}_2)^2\right) = 0. \tag{22}$$

From the proof of Theorem 2, we know that $\mathbf{q}_2$ can be expressed as a function of $\mathbf{q}_1$ as $\mathbf{q}_2 + \mathbf{R}^T\mathbf{q}_2 = \left(\mathbf{R}^T\mathbf{q}_1 - \mathbf{q}_1\right)\cot(\theta/2)$ and $\mathbf{q}_2 - \mathbf{R}^T\mathbf{q}_2 = \left(\mathbf{R}^T\mathbf{q}_1 + \mathbf{q}_1\right)\tan(\theta/2)$. This allows us to write the decision boundary given in (22) as

$$\mathbf{x}^T(\mathbf{R}^T\mathbf{q}_1 + \mathbf{q}_1) = 0, \tag{23}$$

and

$$\mathbf{x}^T(\mathbf{R}^T\mathbf{q}_1 - \mathbf{q}_1) = 0. \tag{24}$$

These two hyperplanes are necessary to successfully classify the antipodally symmetric data of two Bingham distributions. ■

Since antipodally symmetric distributions, such as Bingham distributions, have zero mean, the Gaussian distributions fitted to the data sampled from these distributions will also have zero mean. We now study the spherical-homoscedastic Gaussian pdfs when the mean vector is equal to zero.

Lemma 8 *Two zero-mean Gaussian distributions, $N_1(\mathbf{0}, \Sigma)$ and $N_2(\mathbf{0}, \mathbf{R}^T\Sigma\mathbf{R})$, are spherical-homoscedastic if $\mathbf{R} \in SO(p)$ defines a planar rotation in the subspace spanned by any two of the eigenvectors of Σ, say $\mathbf{v}_1$ and $\mathbf{v}_2$.*

Proof The Bayes classification boundary between these distributions can be obtained by making the ratio of the log-likelihood equations equal to one, $\mathbf{x}^T\Sigma\mathbf{x} = \mathbf{x}^T\mathbf{R}^T\Sigma\mathbf{R}\mathbf{x}$. Note that this equation is in the same form as that derived in (21). Furthermore, since the rotation is defined in the subspace spanned by $\mathbf{v}_1$ and $\mathbf{v}_2$ and $\Sigma = \mathbf{V}\Lambda\mathbf{V}^T$, we can follow the proof of Theorem 7 to show

$$\mathbf{x}^T(\mathbf{R}^T\mathbf{v}_1 + \mathbf{v}_1) = 0, \tag{25}$$

and

$$\mathbf{x}^T(\mathbf{R}^T\mathbf{v}_1 - \mathbf{v}_1) = 0. \tag{26}$$

And, therefore, N_1 and N_2 are also spherical-homoscedastic. ■

We are now in a position to prove that the decision boundaries obtained using two Gaussian distributions are the same as those defined by spherical-homoscedastic Bingham distributions.

Theorem 9 *The Bayes decision boundaries of two spherical-homoscedastic Bingham distributions, $B_1(\mathbf{A})$ and $B_2(\mathbf{R}^T\mathbf{A}\mathbf{R})$, are the same as those obtained when modelling $B_1(\mathbf{A})$ and $B_2(\mathbf{R}^T\mathbf{A}\mathbf{R})$ with two Gaussian distributions, $N_1(\mathbf{m},\Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m},\mathbf{R}^T\Sigma\mathbf{R})$, where $\mathbf{m}=\mathbf{0}$ and $\Sigma=\mathbf{S}$.*

Proof Since the data sampled from a Bingham distribution is symmetric with respect to the origin, its mean will be the origin, $\mathbf{m}=\mathbf{0}$. Therefore, the sample covariance matrix will be equal to the sample autocorrelation matrix $\mathbf{S}=n^{-1}\mathbf{X}\mathbf{X}^T$. In short, the estimated Gaussian distribution of $B_1(\mathbf{A})$ will be $N_1(\mathbf{0},\mathbf{S})$. We also know from Section 2.2 that the m.l.e. of the orthonormal matrix $\mathbf{Q}$ (where $\mathbf{A}=\mathbf{Q}\Lambda\mathbf{Q}^T$) is given by the eigenvectors of $\mathbf{S}$. This means that the two Gaussian distributions representing the data sampled from two spherical-homoscedastic Bingham distributions, $B_1(\mathbf{A})$ and $B_2(\mathbf{R}^T\mathbf{A}\mathbf{R})$, are $N_1(\mathbf{0},\mathbf{S})$ and $N_2(\mathbf{0},\mathbf{R}^T\mathbf{S}\mathbf{R})$.

Following Lemma 8, $N_1(\mathbf{0},\mathbf{S})$ and $N_2(\mathbf{0},\mathbf{R}^T\mathbf{S}\mathbf{R})$ are spherical-homoscedastic if $\mathbf{R}$ is spanned by any two eigenvectors of $\mathbf{S}$. Since the eigenvectors of $\mathbf{A}$ and $\mathbf{S}$ are the same ($\mathbf{v}_i=\mathbf{q}_i$ for all i), these two Gaussian distributions representing the two spherical-homoscedastic Bingham distributions will also be spherical-homoscedastic. Furthermore, the hyperplanes of the spherical-homoscedastic Bingham distributions $B_1(\mathbf{A})$ and $B_2(\mathbf{R}^T\mathbf{A}\mathbf{R})$, Eqs. (23 - 24), and the hyperplanes of the spherical-homoscedastic Gaussian distributions $N_1(\mathbf{0},\mathbf{S})$ and $N_2(\mathbf{0},\mathbf{R}^T\mathbf{S}\mathbf{R})$, Eqs. (25 - 26), will be identical. ■

We now turn to study the similarity between the results obtained using the Gaussian distribution and the Kent distribution. We first define when two Kent distributions are spherical-homoscedastic.

Theorem 10 *Two Kent distributions $K_1(\mu,\kappa,\mathbf{A})$ and $K_2(\mathbf{R}^T\mu,\kappa,\mathbf{R}^T\mathbf{A}\mathbf{R})$ are spherical-homoscedastic, if the rotation matrix $\mathbf{R}$ is defined on the plane spanned by the mean direction μ and one of the eigenvectors of $\mathbf{A}$.*

Proof By making the two log-likelihood equations equal, we have $\kappa\mu^T\mathbf{x}+\mathbf{x}^T\mathbf{A}\mathbf{x}=\kappa(\mathbf{R}^T\mu)^T\mathbf{x}+\mathbf{x}^T\mathbf{R}^T\mathbf{A}\mathbf{R}\mathbf{x}$. Let the spectral decomposition of $\mathbf{A}$ be $\mathbf{A}=\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$, then $\kappa(\mu^T\mathbf{x}-\mu^T\mathbf{R}\mathbf{x})+\sum_{i=1}^p\lambda_i(\mathbf{x}^T\mathbf{q}_i)^2-\sum_{i=1}^p\lambda_i(\mathbf{x}^T(\mathbf{R}^T\mathbf{q}_i))^2=0$. Since $\mathbf{R}$ is defined to be in the plane spanned by an eigenvector of $\mathbf{A}$ (say, $\mathbf{q}_1$) and the mean direction μ, one can simplify the above equation to $\kappa(\mu^T\mathbf{x}-\mu^T\mathbf{R}\mathbf{x})+\lambda_1(\mathbf{x}^T\mathbf{q}_1)^2-\lambda_1(\mathbf{x}^T(\mathbf{R}^T\mathbf{q}_1))^2=0$. Since the first term of this equation is a constant, its transpose would yield the same result,

$$\kappa(\mathbf{x}^T(\mu-\mathbf{R}^T\mu))+\lambda_1(\mathbf{x}^T(\mathbf{q}_1-\mathbf{R}^T\mathbf{q}_1))(\mathbf{x}^T(\mathbf{q}_1+\mathbf{R}^T\mathbf{q}_1))=0.$$

Using the relation between $\mathbf{v}_1$ and $\mathbf{v}_2$ (now μ and $\mathbf{q}_1$) given in the proof of Theorem 2,

$$\begin{aligned}(\mathbf{q}_1 + \mathbf{R}^T\mathbf{q}_1) &= (\mathbf{R}^T\mu - \mu)\cot\left(\frac{\theta}{2}\right)\\ (\mathbf{q}_1 - \mathbf{R}^T\mathbf{q}_1) &= (\mathbf{R}^T\mu + \mu)\tan\left(\frac{\theta}{2}\right),\end{aligned}$$

we can write $(\mathbf{x}^T(\mathbf{R}^T\mu - \mu))(\mathbf{x}^T(\mathbf{R}^T\mu + \mu)\lambda_1 - \kappa) = 0$, where θ is the rotation angle defined by $\mathbf{R}$. This equation gives us the hyperplane decision boundary equations,

$$\mathbf{x}^T\frac{(\mathbf{R}^T\mu - \mu)}{\sin\left(\frac{\theta}{2}\right)} = 0 \tag{27}$$

$$\mathbf{x}^T\left(\frac{\mathbf{R}^T\mu + \mu}{\cos\left(\frac{\theta}{2}\right)}\right) - \frac{\kappa}{\cos\left(\frac{\theta}{2}\right)\lambda_1} = 0. \tag{28}$$

■

Finally, we are in a position to define the relation between Kent and Gaussian pdfs.

Theorem 11 *The first hyperplane given by the Bayes decision boundary of two spherical-homoscedastic Kent distributions, $K_1(\mu, \kappa, \mathbf{A})$ and $K_2(\mathbf{R}^T\mu, \kappa, \mathbf{R}^T\mathbf{A}\mathbf{R})$, is equal to the first hyperplane obtained when modelling K_1 and K_2 with the two Gaussian distributions $N_1(\mathbf{m}, \Sigma)$ and $N_2(\mathbf{R}^T\mathbf{m}, \mathbf{R}^T\Sigma\mathbf{R})$ that best approximate them. Furthermore, when $\kappa > \lambda_{1_K}$ and $\|m\| > 1 - \lambda_{1_G}/\lambda_{2_G}$, then the second hyperplanes of the Kent and Gaussian distributions are outside the sphere and can be ignored in classification; where λ_{1_K} is the eigenvalue associated to the eigenvector of $\mathbf{A}$ defining the rotation $\mathbf{R}$, and λ_{1_G} and λ_{2_G} are the two eigenvalues of Σ defining the rotation $\mathbf{R}$.*

Proof If we fit a Gaussian distribution to an ellipsoidally symmetric pdf, then the mean direction of the data is described by one of the eigenvectors of the covariance matrix. Since the Kent distribution assumes the data is either concentrated or distributed more or less equally about every dimension, one can conclude that the eigenvectors of $\bar{\mathbf{S}}$ (the scatter matrix calculated on the null-space of the mean direction) are a good estimate of the orthonormal bases of $\mathbf{A}$ (Kent, 1982). This means that (14) and (27) will define the same hyperplane equation. Furthermore, we see that the bias in (15) and that of (28) will be the same when

$$-\kappa\lambda_{1_K}^{-1} = \frac{\|\mathbf{m}\|s}{\left(\frac{\lambda_{1_G}}{\lambda_{2_G}} - 1\right)},$$

where λ_{1_K} is the eigenvalue associated to the eigenvector defining the rotation plane (as given in Theorem 10), and λ_{1_G} and λ_{2_G} are the eigenvalues associated to the eigenvectors that span the rotation matrix $\mathbf{R}$ (as shown in Theorem 2 – recall that λ_{1_G} is associated to the eigenvector aligned with the mean direction).

Similarly to what happened for the vMF case, the second hyperplane may be outside S^{p-1}. When this is the case, such planes are not relevant and can be eliminated. For this

to happen, the two biases need not be the same, but need be larger than one; i.e.,

$$|\kappa\lambda_{1_K}^{-1}| > 1 \quad \text{and} \quad \left| \frac{s\|\mathbf{m}\|}{\left(\frac{\lambda_{1_G}}{\lambda_{2_G}} - 1\right)} \right| > 1. \tag{29}$$

These two last conditions can be interpreted as follows. The second hyperplane of the Kent distribution will not intersect with the sphere when $\kappa > \lambda_{1_K}$. The second hyperplane of the Gaussian estimate will be outside the sphere when $\|m\| > 1 - \lambda_{1_G}/\lambda_{2_G}$. These two conditions hold, for example, when the data is concentrated, but not when the data is uniformly distributed. ■

Thus far, we have shown where the results obtained by modelling the true underlying spherical distributions with Gaussians do not pose a problem. We now turn to the case where both solutions may differ.

4. Spherical-Heteroscedastic Distributions

When two (or more) distributions are not spherical-homoscedastic, we will refer to them as spherical-heteroscedastic. In such a case, the classifier obtained with the Gaussian approximation needs not be the same as that computed using the original spherical distributions. To study this problem, one may want to compute the classification error that is added to the original Bayes error produced by the Bayes classifier on the two (original) spherical distributions. Following the classical notation in Bayesian theory, we will refer to this as the reducible error. This idea is illustrated in Figure 4. In Figure 4(a) we show the original Bayes error obtained when using the original vMF distributions. Figure 4(b) depicts the classifier obtained when one models these vMF using two Gaussian distributions. And, in Figure 4(c), we illustrate the reducible error added to the original Bayes error when one employs the new classifier in lieu of the original one.

In theory, we could calculate the reducible error by means of the posterior probabilities of the two spherical distributions, $P_S(w_1|\mathbf{x})$ and $P_S(w_2|\mathbf{x})$, and the posteriors of the Gaussians modelling them, $P_G(w_1|\mathbf{x})$ and $P_G(w_2|\mathbf{x})$. This is given by,

$$\begin{aligned} P(\text{reducible error}) &= \int_{\frac{P_G(w_1|\mathbf{x})}{P_G(w_2|\mathbf{x})} \geq 1} P_S(w_2|\mathbf{x})\, p(\mathbf{x})\, dS^{p-1} + \int_{\frac{P_G(w_1|\mathbf{x})}{P_G(w_2|\mathbf{x})} < 1} P_S(w_1|\mathbf{x})\, p(\mathbf{x})\, dS^{p-1} \\ &- \int_{S^{p-1}} \min(P_S(w_1|\mathbf{x}), P_S(w_2|\mathbf{x}))\, p(\mathbf{x})\, dS^{p-1}, \end{aligned}$$

where the first two summing terms calculate the error defined by the classifier obtained using the Gaussian approximation (as for example that shown in Figure 4(b)), and the last term is the Bayes error associated to the original spherical-heteroscedastic distributions (Figure 4(a)).

Unfortunately, in practice, this error cannot be calculated because it is given by the integral of a number of class densities over a nonlinear region on the surface of a sphere. Note that, since we are exclusively interested in knowing how the reducible error increases

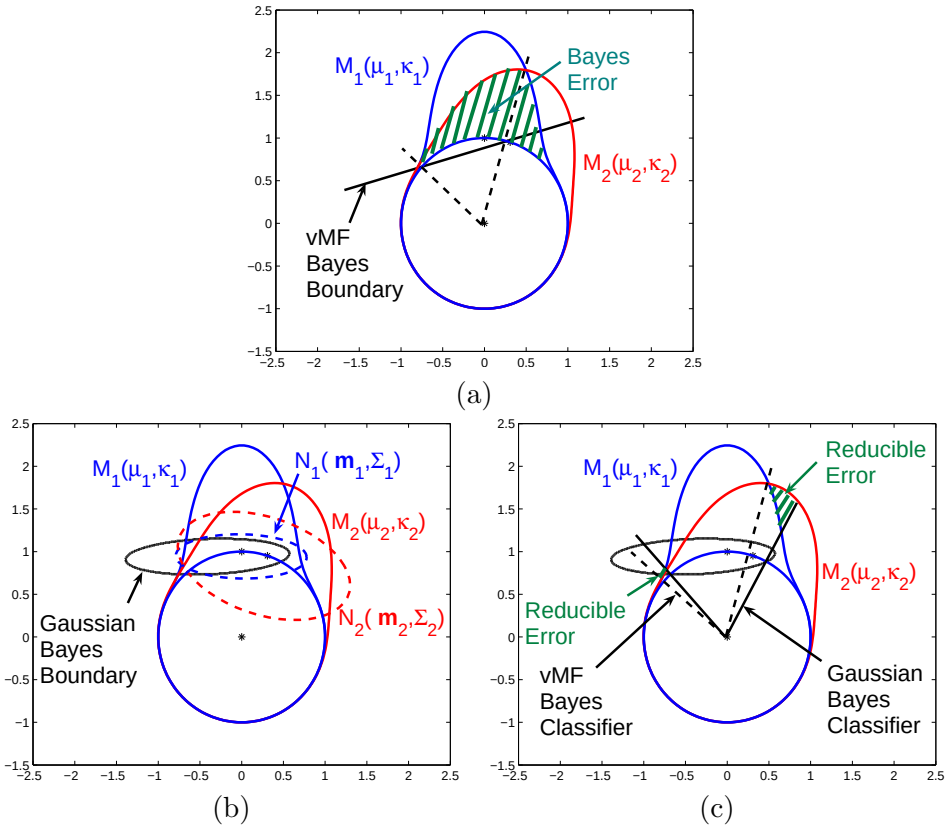

Figure 4: *(a) Shown here are two spherical-heteroscedastic vMF distributions with $\kappa_1 = 10$ and $\kappa_2 = 5$. The solid line is the Bayes decision boundary, and the dashed lines are the corresponding classifiers. This classifier defines the Bayes error, which is represented by the dashed fill. In (b) we show the Bayes decision boundary obtained using the two Gaussian distributions that best model the original vMFs. (c) Provides a comparison between the Bayes classifier derived using the original vMFs (dashed lines) and that calculated from the Gaussian approximation (solid lines). The dashed area shown in (c) corresponds to the error added to the original Bayes error; i.e., the reducible error.*

as the data distributions deviate from spherical-homoscedasticity, the use of error bounds would not help us solve this problem either. We are therefore left to empirically study how the reducible error increases as the original data distributions deviate from spherical-homoscedastic. This we will do next.

4.1 Modelling Spherical-Heteroscedastic vMFs

We start our analysis with the case where the two original spherical distributions are given by the vMF model. Let these two vMFs be $M_1(\mu_1, \kappa_1)$ and $M_2(\mu_2, \kappa_2)$, where $\mu_2 = \mathbf{R}^T\mu_1$ and $\mathbf{R} \in SO(p)$. Recall that $\mathbf{R}$ defines the angle θ which specifies the rotation between μ_1 and μ_2. Furthermore, let $N_1(\mathbf{m}_1, \Sigma_1)$ and $N_2(\mathbf{m}_2, \Sigma_2)$ be the two Gaussian distributions that best model $M_1(\mu_1, \kappa_1)$ and $M_2(\mu_2, \kappa_2)$, respectively. From the proof of Proposition

4 we know that the means and covariance matrices of these Gaussians can be defined in terms of the mean directions and the concentration parameters of the corresponding vMF distributions. Defining the parameters of the Gaussian distributions in terms of κ_i and μ_i ($i = \{1, 2\}$) allows us to estimate the reducible error with respect to different rotation angles θ, concentration parameters κ_1 and κ_2, and dimensionality p.

Since our goal is to test how the reducible error increases as the original data distributions deviate from spherical-homoscedasticity, we will plot the results for distinct values of κ_2/κ_1. Note that when $\kappa_2/\kappa_1 = 1$ the data is spherical-homoscedastic and that the larger the value of κ_2/κ_1 is, the more we deviate from spherical-homoscedasticity. In our experiments, we selected κ_1 and κ_2 so that the value of κ_2/κ_1 varied from a low of 1 to maximum of 10 at 0.5 intervals. To do this we varied κ_1 from 1 to 10 at unit steps and selected κ_2 such that the κ_2/κ_1 ratio is equal to one of the values described above.

The dimensionality of the feature space is also varied from 2 to 100 at 10-step intervals. In addition, we also vary the value of the angle θ from 10^o to 180^o at 10^o increments.

The average of the reducible error over all possible values of κ_1 and over all values of θ from 10^o to 90^o is shown in Figure 5(a). As anticipated by our theory, the reducible error is zero when $\kappa_2/\kappa_1 = 1$ (i.e., when the data is spherical-homoscedastic). We see that as the distributions start to deviate from spherical-homoscedastic, the probability of reducible error increases really fast. Nonetheless, we also see that after a short while, this error starts to decrease. This is because the data of the second distribution (M_2) becomes more concentrated. To see this, note that to make κ_2/κ_1 larger, we need to increase κ_2 (with respect to κ_1). This means that the area of possible overlap between M_1 and M_2 decreases and, hence, the reducible error will generally become smaller. In summary, the probability of reducible error increases as the data deviates from spherical-homoscedasticity and decreases as the data becomes more concentrated. This means that, in general, the more two non-highly concentrated distributions deviate from spherical-homoscedastic, the more sense it makes to take the extra effort to model the spherical data using one of the spherical models introduced in Section 2. Nevertheless, it is important to note that the reducible error remains relatively low ($\sim 3\%$).

We also observe, in Figure 5(a), that the probability of reducible error decreases with the dimensionality (which would be something unexpected had the original pdf been defined in the Cartesian space). This effect is caused by the spherical nature of the von Mises-Fisher distribution. Note that since the volume of the distributions need to remain constant, the probability of the vMF at each given point will be reduced when the dimensionality is made larger. Therefore, as the dimensionality increase, the volume of the reducible error area (i.e., the probability) will become smaller.

In Figure 5(b) we show the average probability of the reducible error over κ_1 and p, for different values of κ_2/κ_1 and θ. Here we also see that when two vMFs are spherical-homoscedastic (i.e. $\kappa_2/\kappa_1 = 1$), the average of the reducible error is zero. As we deviate from spherical-homoscedasticity, the probability of reducible error increases. Furthermore, it is interesting to note that as θ increases (in the spherical-heteroscedastic case), the probability of reducible error decreases. This is because as θ increases, the two distributions M_1 and M_2 fall farther apart and, hence, the area of possible overlap generally reduces.

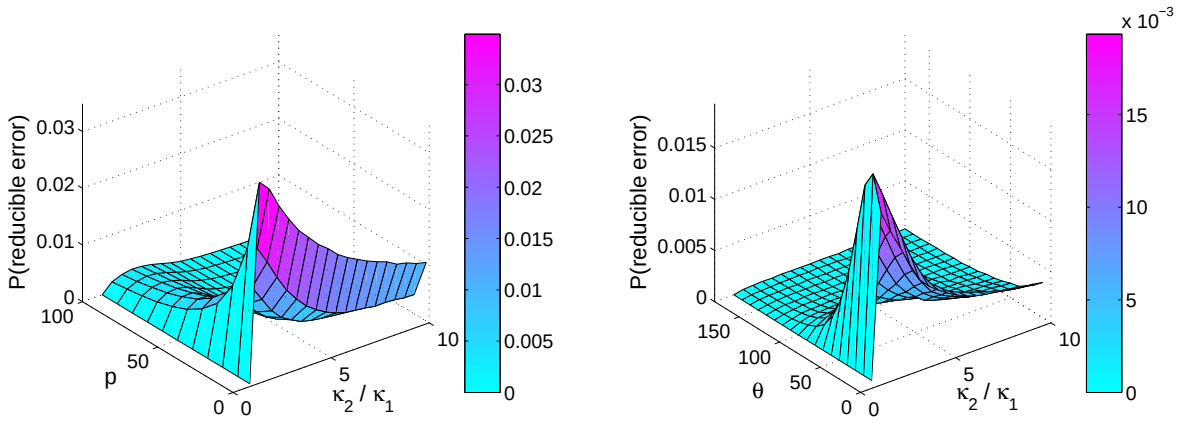

Figure 5: *In (a) we show the average of the probability of reducible error over* κ_1 *and* $\theta = \{10^o, 20^o, \ldots, 90^o\}$ *for different values of* κ_2/κ_1 *and dimensionality* p*. In (b) we show the average probability of reducible error over* p *and* κ_1 *for different values of* κ_2/κ_1 *and* $\theta = \{10^o, 20^o, \ldots, 180^o\}$*.*

4.2 Modelling Spherical-Heteroscedastic Bingham Distributions

As already mentioned earlier, the parameter estimation for Bingham is much more difficult than that of vMF and equations directly linking the parameters of any two Bingham distributions $B_1(\mathbf{A}_1)$ and $B_2(\mathbf{A_2})$ to those of the corresponding Gaussians $N_1(\mathbf{0}, \Sigma_1)$ and $N_2(\mathbf{0}, \Sigma_2)$ are not usually available. Hence, some parameters will need to be estimated from randomly chosen samples from B_i. Recall from Section 2.2 that another difficulty is the calculation of the normalizing constant $c_B(\mathbf{A})$ because this requires us to solve a contour integral on S^{p-1}. This hypergeometric function will be calculated with the method defined by Koev and Edelman (2006).

Moreover, if we want to calculate the reducible error on S^{p-1}, we will need to simulate each of the p variance parameters of B_1 and B_2 and the $p-1$ possible rotations between their means; i.e. $3p-1$. While it would be almost impossible to simulate this for a large number of dimensions p, we can easily restrict the problem to one of our interest that is of a manageable size.

In our simulation, we are interested in testing the particular case where $p = 3$.[6] Furthermore, we constrain our analysis to the case where the parameter matrix $\mathbf{A}_1$ has a diagonal form; i.e., $\mathbf{A}_1 = diag\,(\lambda_1, \lambda_2, \lambda_3)$. The parameter matrix $\mathbf{A}_2$ can then be defined as a rotated and scaled version of $\mathbf{A}_1$ as $\mathbf{A}_2 = \varsigma \mathbf{R}^T \mathbf{A}_1 \mathbf{R}$, where ς is the scale parameter,

$$\mathbf{R} = \mathbf{R}_1 \mathbf{R}_2 \in SO(3), \tag{30}$$

$\mathbf{R}_1$ defines a planar rotation θ in the range space given by the first two eigenvectors of $\mathbf{A}_1$, and $\mathbf{R}_2$ specifies a planar rotation ϕ in the space defined by the first and third eigenvectors

6. We have also simulated the cases where p was equal to 10 and 50 and observed almost identical results to those shown in this paper.

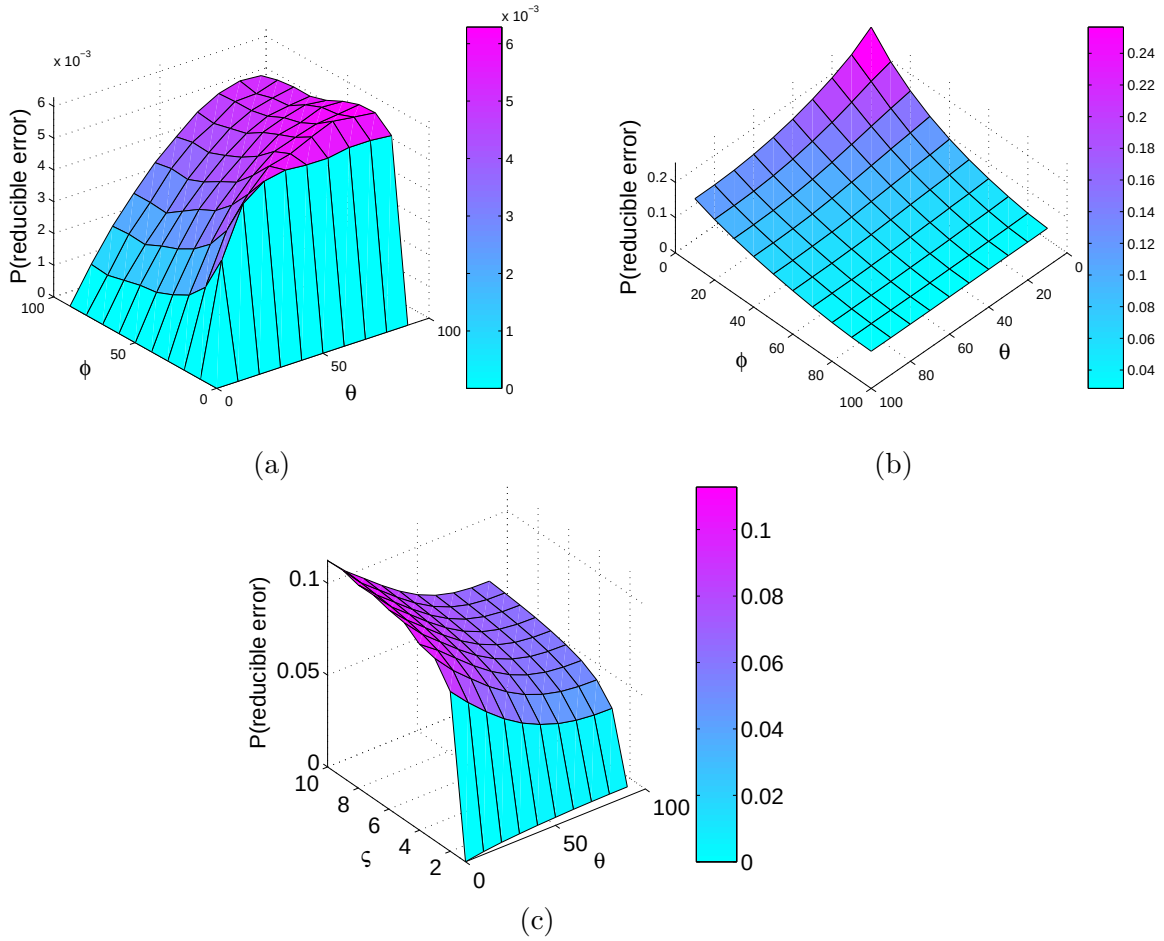

Figure 6: *We show the average of the probability of reducible error over all the possible set of variance parameters when $\varsigma = 1$ in (a) and when $\varsigma \neq 1$ in (b). In (c) we show the increase of reducible error as the data deviates from spherical-homoscedastic (i.e., when ς increases). The more the data deviates from spherical-homoscedastic, the larger the reducible error is. This is independent of the closeness of the two distributions.*

of $\mathbf{A}_1$. Note that B_1 and B_2 can only be spherical-homoscedastic if $\varsigma = 1$ and the rotation is planar ($\phi = 0$ or $\theta = 0$). To generate our results, we used all possible combinations of the values given by $-1/2j$ (with j the odd numbers from 1 to 15 to represent low concentrations and $j = \{30, 60\}$ to model high concentrations) as entries for λ_1, λ_2 and λ_3 constrained to $\lambda_1 < \lambda_2 < \lambda_3$ (i.e., a total of 120 combinations). We also let $\theta = \{0^o, 10^o, \ldots, 90^o\}$, $\phi = \{0^o, 10^o, \ldots, 90^o\}$ and $\varsigma = \{1, 2, \ldots, 10\}$.

In Figure 6(a), we study the case where $\varsigma = 1$. In this case, B_1 and B_2 are spherical-homoscedastic when either ϕ or θ is zero. As seen in the figure, the probability of reducible error increases as the data starts to deviate from spherical-homoscedastic (i.e., when the

rotation $\mathbf{R}$ does not define a planar rotation). Nonetheless, the probability of reducible error is still very small even for large values of ϕ and θ – approximately 0.006.[7]

When the scale parameter ς is not one, the two Bingham distributions B_1 and B_2 can never be spherical-homoscedastic. In this case, the probability of reducible error is generally expected to be larger. This is shown in Figure 6(b) where the probability of reducible error has been averaged over all possible combinations of variance parameters $(\lambda_1, \lambda_2, \lambda_3)$ and scales $\varsigma \neq 1$. Here, it is important to note that as the two original distributions get closer to each other the probability of reducible error increases quite rapidly. In fact, the error can be incremented by more than 20%. As in vMF, this means that if the data largely deviates from spherical-homoscedastic, extra caution needs to be taken with our results.

To further illustrate this point, we can plot the probability of reducible error over ς and θ, Figure 6(c). In this case, the larger ς is, the more different the eigenvalues of the parameter matrices ($\mathbf{A}_1$ and $\mathbf{A}_2$) will be. This means, that the larger the value of ς, the more the distributions deviate from spherical-homoscedastic. Hence, this plot shows the increase in reducible error as the two distributions deviate from spherical-homoscedastic. We note that this is in fact independent of how close the two distributions are, since the slop of the curve increases for every value of θ.

4.3 Modelling Spherical-Heteroscedastic Kent Distributions

Our final analysis involves the study of spherical-heteroscedastic Kent distributions. Here, we want to estimate the probability of reducible error when two Gaussian distributions $N_1(\mathbf{m}_1, \Sigma_1)$ and $N_2(\mathbf{m}_2, \Sigma_2)$ are used to model the data sampled from two Kent distributions $K_1(\mu_1, \kappa_1, \mathbf{A}_1)$ and $K_2(\mu_2, \kappa_2, \mathbf{A}_2)$. Recall that the shape of a Kent distribution is given by two parameters: β_i, which defines the ovalness of K_i, and κ_i, the concentration parameter. From (Kent, 1982) we know that if $2\beta_i/\kappa_i < 1$, then the normalizing constant $c(\kappa_i, \beta_i)$ can be approximated by $2\pi e_i^{\kappa}[(\kappa_i - 2\beta_i)(\kappa_i + 2\beta_i)]^{-1/2}$. Note that $2\beta_i/\kappa_i < 1$ holds regardless of the ovalness of our distribution when the data is highly concentrated, whereas in the case where the data is not concentrated (i.e., κ_i is small) the condition holds when the distribution is almost circular (i.e., β_i is very small).

To be able to use this approximation in our simulation, we have selected two sets of concentration parameters: one for the low concentration case (where $\kappa_i = \{2, 3, \ldots, 10\}$), and another where the data is more concentrated ($\kappa_i = \{15, 20, \ldots, 50\}$). The value of β_i is then given by the following set of equalities $2\beta_i/\kappa_i = \{0.1, 0.3, \ldots, 0.9\}$. As was done in the previous section (for the spherical-heteroscedastic Bingham), we fixed the mean direction μ_1 and then rotate μ_2 using a rotation matrix $\mathbf{R}$; i.e. $\mu_2 = \mathbf{R}^T\mu_1$. To do this, we used the same rotation matrix $\mathbf{R}$ defined in (30). Now, however, $\mathbf{R}_1$ defines a planar rotation in the space spanned by μ_1 and the first eigenvector of $\mathbf{A}_1$, and $\mathbf{R}_2$ is a planar rotation defined in the space of μ_1 and the second eigenvector of $\mathbf{A}_1$. In our simulations we used $\{0^o, 15^o, \ldots, 90^o\}$ as values for the rotations defined by θ and ϕ.

In Figure 7(a), we show the results of our simulation for the special case where the variance parameters of K_1 and K_2 are the same (i.e., $\kappa_1 = \kappa_2$ and $\beta_1 = \beta_2$) and the data is not concentrated. Note that the criteria defined in (29) hold when either $\mathbf{R}_1$ or $\mathbf{R}_2$ is

7. Note that the plot shown in Figure 6(a) is not symmetric. This is due to the constraint given above ($\lambda_1 < \lambda_2 < \lambda_3$) which gives less flexibility to ϕ.

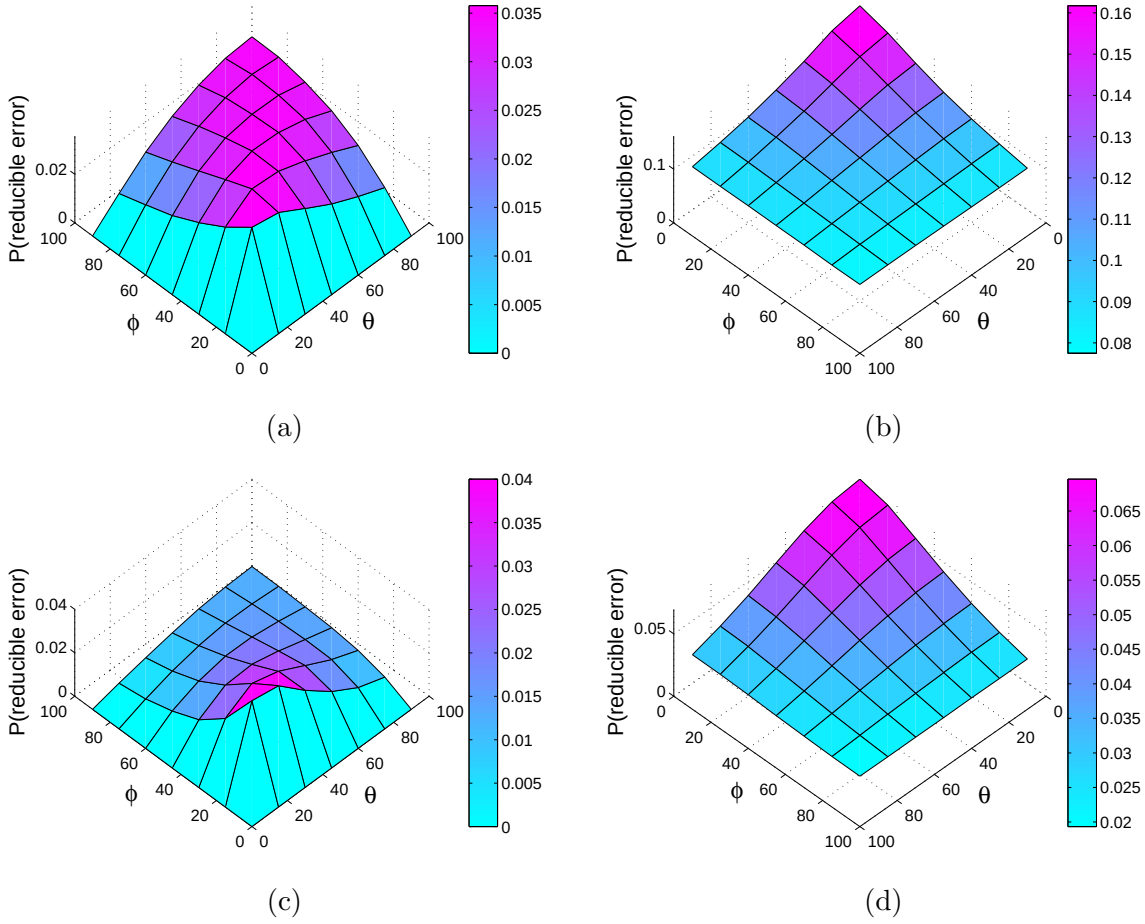

Figure 7: *In (a) we show the average of the probability of reducible error when $\kappa_1 = \kappa_2$ and $\beta_1 = \beta_2$ and the data is not concentrated. (b) Shows the probability of reducible error when the parameters of the pdf are different in each distribution and the values of κ_i are small. (c-d) Do the same as (a) and (b) but for the cases where the concentration parameters are large (i.e., the data is concentrated).*

the identity matrix (or equivalently, in Figure 7(a), when θ or ϕ is zero). As anticipated in Theorem 11, in these cases the probability of reducible error is zero. Then the more these two distributions deviate from spherical-homoscedastic, the larger the probability of reducible error will become. It is worth mentioning, however, that the probability of reducible error is small over all possible values for θ and ϕ (i.e. < 0.035). We conclude (as with the analysis of the Bingham distribution comparison) that when the data only deviates from spherical-homoscedastic by a rotation (but the variances remain the same), the Gaussian approximation is a reasonable one. This means that whenever the parameters of the two distributions (Bingham or Kent) are defined up to a rotation, the results obtained using the Gaussian approximation will generally be acceptable.

The average of the probability of reducible error for all possible values for κ_i and β_i (including those where $\kappa_1 \neq \kappa_2$ and $\beta_1 \neq \beta_2$) is shown in Figure 7(b). In this case, we see that the probability of reducible error is bounded by 0.17 (i.e. 17%). Therefore, in the general case, unless the two original distributions are far away from each other, it is not advisable to model them using Gaussian distributions.

Figure 7(c-d) show exactly the same as (a-b) but for the case where the data is highly-concentrated. As expected, when the data is more concentrated in a small area, the probability of reducible error decreases fast as the two distributions fall far apart from each other. Similarly, since the data is concentrated, the maximum of the probability of reducible error shown in Figure 7(d) is smaller than that observed in (b).

As seen up to now, there are several conditions under which the Gaussian assumption is acceptable. In vMF, this happens when the distributions are highly concentrated, and in Bingham and Kent when the variance parameters of the distributions are the same.

Our next point relates to what can be done when neither of these assumptions hold. A powerful and generally used solution is to employ a kernel to (implicitly) map the original space to a high-dimensional one where the data can be better separated. Our next goal is thus to show that the results defined thus far are also applicable in the kernel space.

This procedure will provide us with a set of new classifiers (in the kernel space) that are based on the idea of spherical-homoscedastic distributions.

5. Kernel Spherical-Homoscedastic Classifiers

To relax the linear constraint stated in Definition 1, we will now employ the idea of the kernel trick, which will permit us to define classifiers that are nonlinear in the original space, but linear in the kernel one. This will be used to tackle the general spherical-heteroscedastic problem as if the distributions were linearly separable spherical-homoscedastic. Our goal is thus to find a kernel space where the classes adapt to this model.

We start our description for the case of vMF distributions. First, we define the sample mean direction of the first distribution $M_1(\mu, \kappa)$ as $\hat{\mu}_1$. The sample means of a set of spherical-homoscedastic distributions can be represented as rotated versions of this first one, i.e., $\hat{\mu}_a = \mathbf{R}_a^T \hat{\mu}_1$, with $\mathbf{R}_a \in SO(p)$ and $a = \{2, \ldots, C\}$, C the number of classes. Following this notation, we can derive the classification boundary between any pair of distributions from (16) as

$$\mathbf{x}^T(\hat{\mu}_a - \hat{\mu}_b) = 0, \quad \forall a \neq b.$$

The equation above directly implies that any new vector $\mathbf{x}$ will be classified to that class a for which the inner product between $\mathbf{x}$ and $\hat{\mu}_a$ is largest, that is,

$$\arg\max_a \mathbf{x}^T \hat{\mu}_a. \tag{31}$$

We are now in a position to provide a similar result in the feature space $\mathcal{F}$ obtained by the function $\phi(\mathbf{x})$, which maps the feature vector $\mathbf{x}$ from our original space S^{p-1} to a new spherical space S^d of d dimensions. In general, this can be described as a kernel $k(\mathbf{x}_i, \mathbf{x}_j)$, defined as the inner product of the two feature vectors in $\mathcal{F}$, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$.

Note that the mappings we are considering here are such that the resulting space $\mathcal{F}$ is also spherical. This is given by *all* those mappings for which the resulting norm of all vectors

is a constant; i.e., $\phi(\mathbf{x})^T\phi(\mathbf{x}) = h$, $h \in \mathbb{R}^+$.[8] In fact, many of the most popular kernels have such a property. This includes kernels such as the Radial Basis Function (RBF), polynomial, and Mahalanobis, when working in S^{p-1}, and several (e.g., RBF and Mahalanobis) when the original space is $\mathbb{R}^p$. This observation makes our results of more general interest yet, because even if the original space is not spherical, the use of some kernels will map the data into S^d. This will again require the use of spherical distributions or their Gaussian equivalences described in this paper.

In this new space $\mathcal{F}$, the sample mean direction of class a is given by

$$\begin{aligned}\hat{\mu}_a^\phi &= \frac{\frac{1}{n_a}\sum_{i=1}^{n_a}\phi(\mathbf{x}_i)}{\sqrt{\frac{1}{n_a}\sum_{i=1}^{n_a}\phi(\mathbf{x}_i)^T\frac{1}{n_a}\sum_{j=1}^{n_a}\phi(\mathbf{x}_j)}} \\ &= \frac{\frac{1}{n_a}\sum_{i=1}^{n_a}\phi(\mathbf{x}_i)}{\sqrt{\mathbf{1}^T\mathbf{K}\mathbf{1}}},\end{aligned}$$

where $\mathbf{K}$ is a symmetric positive semidefinite matrix with elements $\mathbf{K}(i,j) = k(\mathbf{x}_i, \mathbf{x}_j)$, $\mathbf{1}$ is a vector with all elements equal to $1/n_a$, and n_a is the number of samples in class a.

By finding a kernel which transforms the original distributions to spherical-homoscedastic vMF, we can use the classifier defined in (31), which states that the class label of any test feature vector $\mathbf{x}$ is

$$\begin{aligned}\arg\max_a \ \phi(\mathbf{x})^T\hat{\mu}_a^\phi &= \frac{\frac{1}{n_a}\sum_{i=1}^{n_a}\phi(\mathbf{x})^T\phi(\mathbf{x}_i)}{\sqrt{\mathbf{1}^T\mathbf{K}\mathbf{1}}} \\ &= \frac{\frac{1}{n_a}\sum_{i=1}^{n_a}k(\mathbf{x},\mathbf{x}_i)}{\sqrt{\mathbf{1}^T\mathbf{K}\mathbf{1}}}.\end{aligned} \tag{32}$$

Therefore, any classification problem that uses a kernel which converts the data to spherical-homoscedastic vMF distributions, can employ the solution derived in (32).

A similar result can be derived for Bingham distributions. We already know that the decision boundary for two spherical-homoscedastic Bingham distributions defined as $B_1(\mathbf{A})$ and $B_2(\mathbf{R}^T\mathbf{A}\mathbf{R})$, with $\mathbf{R}$ representing a planar rotation given by any two eigenvectors of $\mathbf{A}$, is given by the two hyperplane equations (23) and (24). Since the rotation matrix is defined in a 2-dimensional space (i.e., planar rotation), one of the eigenvectors was described as a function of the other in the solution derived in Theorem 7. For classification purposes, this result will vary depending on which of the two eigenvectors of $\mathbf{A}$ we choose to use. To derive our solution we go back to (22) and rewrite it for classifying a new feature vector $\mathbf{x}$. That is, $\mathbf{x}$ will be classified in the first distribution, B_1, if the following holds

$$\lambda_1\left((\mathbf{x}^T\mathbf{q}_1)^2 - (\mathbf{x}^T\mathbf{R}^T\mathbf{q}_1)^2\right) + \lambda_2\left((\mathbf{x}^T\mathbf{q}_2)^2 - (\mathbf{x}^T\mathbf{R}^T\mathbf{q}_2)^2\right) > 0.$$

Using the result shown in Theorem 7, where we expressed $\mathbf{q}_2$ as a function of $\mathbf{q}_1$, we can simplify the above equation to

$$(\lambda_1 - \lambda_2)\left((\mathbf{x}^T\mathbf{q}_1)^2 - (\mathbf{x}^T\mathbf{R}^T\mathbf{q}_1)^2\right) > 0.$$

8. Further, any kernel $k_1(\mathbf{x}_i, \mathbf{x}_j)$ can be defined to have this property by introducing the following simple normalization step $k(\mathbf{x}_i, \mathbf{x}_j) = k_1(\mathbf{x}_i, \mathbf{x}_j)/\sqrt{k_1(\mathbf{x}_i, \mathbf{x}_i)k_1(\mathbf{x}_j, \mathbf{x}_j)}$.

Then, if $\lambda_1 > \lambda_2$, $\mathbf{x}$ will be in B_1 when

$$(\mathbf{x}^T\mathbf{q}_1)^2 > (\mathbf{x}^T\mathbf{R}^T\mathbf{q}_1)^2,$$

which can be simplified to

$$|\mathbf{x}^T\mathbf{q}_1| > |\mathbf{x}^T\mathbf{R}^T\mathbf{q}_1|. \tag{33}$$

If this condition does not hold, $\mathbf{x}$ is classified as a sample of B_2. Also, if $\lambda_2 > \lambda_1$, the reverse applies. In the following, and without loss of generality, we will always assume $\mathbf{q}_1$ corresponds to the eigenvector of $\mathbf{A}$ defining the rotation plane of $\mathbf{R}$ that is associated to the largest of the two eigenvalues. This means that whenever (33) holds, $\mathbf{x}$ is classified in B_1.

The relevance of (33) is that (as in vMF), a test feature vector $\mathbf{x}$ is classified to that class providing the largest inner product value. We can now readily extend this result to the multi-class problem. For this, let $B_1(\mathbf{A})$ be the distribution of the first class and $B_a(\mathbf{R}_a^T\mathbf{A}\mathbf{R}_a)$ that of the a^{th} class, where now $\mathbf{R}_a$ is defined by two eigenvectors of $\mathbf{A}$, $\mathbf{q}_{a_1}$ and $\mathbf{q}_{a_2}$, with corresponding eigenvalues λ_{a_1} and λ_{a_2} and we have assumed $\lambda_{a_1} > \lambda_{a_2}$. Then, the class of a new test feature vector $\mathbf{x}$ is given by

$$\arg\max_a |\mathbf{x}^T\mathbf{q}_{a_1}|. \tag{34}$$

Following Theorem 7, we note that not all the rotations $\mathbf{R}_a$ should actually be considered, because some may result in two distributions B_a and B_b that are not spherical-homoscedastic. To see this, consider the case with three distributions, $B_1(\mathbf{A})$, $B_2(\mathbf{R}_2^T\mathbf{A}\mathbf{R}_2)$ and $B_3(\mathbf{R}_3^T\mathbf{A}\mathbf{R}_3)$. In this case, B_1 and B_2 will always be spherical-homoscedastic if $\mathbf{R}_2$ is defined in the plane spanned by two of the eigenvectors of $\mathbf{A}$. The same applies to B_1 and B_3. However, even when $\mathbf{R}_2$ and $\mathbf{R}_3$ are defined by two eigenvectors of the parameter matrix, the rotation between B_2 and B_3 may not be planar. Nonetheless, we have shown in Section 4 that if the variances of the two distributions are the same up to an arbitrary rotation, the reducible error is negligible. Therefore, and since imposing additional constraints would make it very difficult to find a kernel that can map the original distributions to spherical-homoscedastic, we will consider all rotations about every $\mathbf{q}_{a_1}$.

We see that (34) still restricts the eigenvectors $\mathbf{q}_{a_1}$ to be rotated versions of one another. This constraint comes from (33), where the eigenvector of the second distribution must be the first eigenvector rotated by the rotation matrix relating the two distributions. Since the rotation is a rigid transformation, all $\mathbf{q}_{a_1}$ will be defined by the same index i in $\hat{\mathbf{q}}_{a_i}$, where $\widehat{\mathbf{Q}}_a = \{\hat{\mathbf{q}}_{a_1}, \ldots, \hat{\mathbf{q}}_{a_p}\}$ are the eigenvectors of the autocorrelation matrix $\mathbf{S}_a$ of the a^{th} class. This equivalency comes from Section 2.2, where we saw that the eigenvectors of $\mathbf{A}_a$ are the same as those of the autocorrelation matrix. Also, since we know the data is spherical-homoscedastic, the eigenvectors of the correlation matrix will be the same as those of the covariance matrix Σ_a of the (zero-mean) Gaussian distribution as seen in Theorem 9.

Our next step is to derive the same classifier in the kernel space. From our discussion above, we require to find the eigenvectors of the covariance matrix. The covariance matrix in $\mathcal{F}$ can be computed as

$$\Sigma_a^\Phi = \Phi(\mathbf{X}_a)\Phi(\mathbf{X}_a)^T,$$

where $\mathbf{X}_a$ is a matrix whose columns are the sample feature vectors, $\mathbf{X}_a = \left(\mathbf{x}_{a_1}, \mathbf{x}_{a_2}, \ldots, \mathbf{x}_{a_{n_a}}\right)$, and $\Phi(\mathbf{X})$ is a function which maps the columns $\mathbf{x}_i$ of $\mathbf{X}$ with $\phi(\mathbf{x}_i)$.

This allows us to obtain the eigenvectors of the covariance matrix from

$$\Sigma_a^\Phi \mathbf{V}_a^\Phi = \mathbf{V}_a^\Phi \Lambda_a^\Phi.$$

Further, these d-dimensional eigenvectors $\mathbf{V}_a^\Phi = \{\mathbf{v}_{a_1}^\Phi, \ldots, \mathbf{v}_{a_d}^\Phi\}$ are not only the same as those of $\mathbf{A}_a^\Phi$, but, as shown in Theobald (1975), are also sorted in the same order.

As pointed out before though, the eigenvalue decomposition equation shown above may be defined in a very high dimensional space. A usual way to simplify the computation is to employ the kernel trick. Here, note that since we only have n_a samples in class a, $rank(\Lambda_a^\Phi) \leq n_a$. This allows us to write $\mathbf{V}_a^\Phi = \Phi(\mathbf{X}_a)\Delta_a$, where Δ_a is a $n_a \times n_a$ coefficient matrix, and thus the above eigenvalue decomposition equation can be stated as

$$\Phi(\mathbf{X}_a)\Phi(\mathbf{X}_a)^T\Phi(\mathbf{X}_a)\Delta_a = \Phi(\mathbf{X}_a)\Delta_a\Lambda_a^\Phi.$$

Multiplying both sides by $\Phi(\mathbf{X}_a)^T$ and cancelling terms, we can simplify this equation to

$$\begin{aligned} \Phi(\mathbf{X}_a)^T\Phi(\mathbf{X}_a)\Delta_a &= \Delta_a\Lambda_a^\Phi \\ \mathbf{K}_a\Delta_a &= \Delta_a\Lambda_a^\Phi, \end{aligned}$$

where $\mathbf{K}_a$ is known as the Gram matrix.

We should now be able to obtain the eigenvectors in $\mathcal{F}$ using the equality

$$\widehat{\mathbf{V}}_a^\Phi = \Phi(\mathbf{X}_a)\Delta_a.$$

However, the norm of the vectors $\widehat{\mathbf{V}}_a^\Phi$ thus obtained is not one, but rather

$$\Lambda_a^\Phi = \Delta_a^T\Phi(\mathbf{X}_a)^T\Phi(\mathbf{X}_a)\Delta_a.$$

To obtain the (unit-norm) eigenvectors, we need to include a normalization coefficient into our result,

$$\mathbf{V}_a^\Phi = \Phi(\mathbf{X}_a)\Delta_a\Lambda_a^{\Phi\,-1/2},$$

where $\mathbf{V}_a^\Phi = \{\mathbf{v}_{a_1}^\phi, \ldots, \mathbf{v}_{a_{n_a}}^\phi\}$, and $\mathbf{v}_{a_i}^\phi \in S^d$.

The classification scheme derived in (34) can now be extended to classify $\phi(\mathbf{x})$ as

$$\arg\max_a \; |\phi(\mathbf{x})^T\mathbf{v}_{a_i}^\phi|,$$

where the index $i = \{1, \ldots, p\}$ defining the eigenvector $\mathbf{v}_{a_i}^\phi$ must be kept constant for all a.

The result derived above, can be written using a kernel as

$$\arg\max_a \left| \sum_{l=1}^{n_a} \frac{k(\mathbf{x}, \mathbf{x}_l)\delta_{a_i}(l)}{\sqrt{\lambda_{a_i}^\phi}} \right|, \tag{35}$$

where $\Delta_a = \{\delta_{a_1}, \ldots, \delta_{a_{n_a}}\}$, $\delta_{a_i}(l)$ is the l^{th} coefficient of the vector δ_{a_i}, and again i takes a value from the set $\{1, \ldots, p\}$ but otherwise kept constant for all a.

We note that there is an important difference between the classifiers derived in this section for vMF in (32) and for Bingham in (35). While in vMF we are only required to optimize the kernel responsible to map the spherical-heteroscedastic data to one that adapts to spherical-homoscedasticity, in Bingham we will also require the optimization of the eigenvector (associated to the largest eigenvalue) $\mathbf{v}_{a_i}^{\phi}$ defining the rotation matrix $\mathbf{R}_a$. This is because there are many different solutions which can convert a set of Bingham distributions into spherical-homoscedastic.

To conclude this section, we turn to the derivations of a classifier for the Kent distribution in the kernel space. From Theorem 10, the 2-class classifier can be written as

$$\begin{aligned} \mathbf{x}^T(\mu_a - \mu_b) &> 0 \\ \mathbf{x}^T(\mu_a + \mu_b) - \frac{\kappa}{\lambda_{a_1}} &> 0. \end{aligned}$$

The first of these equations is the same as that used to derive the vMF classifier, and will therefore lead to the same classification result. Also, as seen in Theorem 11 the second equation can be eliminated when either: *i)* the second hyperplane is identical to the first, or *ii)* the second hyperplane is outside S^{p-1}. Any other case should actually not be considered, since this would not guarantee the equality of the Gaussian model. Therefore, and rather surprisingly, we conclude that the Kent classifier in the kernel space will be the same as that derived by the vMF distribution. The rational behind this is, however, quite simple, and it is due to the assumption of the concentration of the data made in the Kent distribution: Since the parameters of the Kent distributions are estimated in the tangent space of the mean direction, the resulting classifier should only use this information. This is exactly the solution derived in (32).

6. Experimental Results

In this section we show how the results reported in this paper can be used in real applications. In particular, we apply our results to the classification of text data, genomics, and object classification. Before we get to these though, we need to address the problems caused by noise and limited number of samples. We start by looking at the problem of estimating the parameters of the Gaussian fit of a spherical-homoscedastic distribution from a limited number of samples and how this may effect the equivalency results of Theorems 6, 9 and 11.

6.1 Finite Sample Set

Theorems 6, 9 and 11 showed that the classifiers separating two spherical-homoscedastic distributions are the same as those of the corresponding Gaussian fit. This assumes, however, that the parameters of the spherical distributions are known. In the applications to be presented in this section, we will need to estimate the parameters of such distributions from a limited number of sample feature vectors. The question to be addressed here is to what extent this may effect the classification results obtained with the best Gaussian fit. Following the notation used above, we are interested in finding the *reducible error* that will (on average) be added to the classification error when using the Gaussian model.

To study this problem, we ran a set of simulations. We started by drawing samples from two underlying spherical-homoscedastic distributions that are hidden (unknown) to the classifier. Then, we estimated the mean and covariance matrix defining the Gaussian distribution of the samples in each of the two classes. New test feature vectors were randomly drawn from each of the two (underlying) distributions and classified using the log-likelihood equations defined in Section 2.3. The percentage of samples misclassified by the Gaussian model (i.e., using (6)) as opposed to the correct classification given by the corresponding spherical model (i.e. equations (7)-(9)), determines the probability of reducible error.

Our spherical-homoscedastic vMF simulation used the following concentration parameters $\kappa_1 = \kappa_2 = \{1, 2, \ldots, 10\}$, and the following rotations between distributions $\theta = \{10, 20, \ldots, 180\}$. The number of samples randomly drawn from each distribution varied from 10 times the dimensionality to 100 times that value, at increments of ten. The dimensionality was tested for the following values $p = \{2, 10, 20, 30, 40, 50\}$. Our experiment was repeated 100 times for each of the possible parameters, and the average was computed to obtain the probability of reducible error.

Fig. 8(a)-(b) show these results. In (a) the probability of reducible error is shown over the dimensionality of the data p and the scalar γ, which is defined as the number of samples over their dimensionality, $\gamma = n/p$. As already noticed in Section 4, the probability of reducible error decreases with the dimensionality. This is the case since the volume of the distribution is taken over a larger number of dimensions, forcing the overlapping area to shrink. As the scalar γ increases, so does the number of samples n. In those cases, if p is small, the probability of the reducible error decreases. However, when the dimensionality is large, this probability increases at first. In short, we are compensating the loss in volume caused by the increase in dimensionality, by including additional samples. When the dimension to sample ratio is adequate, the probability of reducible error decreases as expected. The decreasing rate for larger dimensions is, however, much slower. This result calls for a reduction of dimensionality from S^{p-1} to a hypersphere where the scalar γ is not too large. This will be addressed in our next section.

We also note that in the worse case scenario, the probability of reducible error is very low and shall not have a dramatic effect in the classification results when working with vMF.

In Fig. 8(b), we see that the closer two vMF distributions get, the larger the reducible error can be. This is because the closer the distributions, the larger the overlap. Once more, however, this error is negligible. With this, we can conclude that the parameters of the vMF can very reliably be estimated with the corresponding Gaussian fit described in this paper even if the number of samples is limited. This is especially true when the dimensionality of the data is relatively low.

To simulate the probability of reducible error in Bingham and Kent, several of the parameters of the distributions were considered. For simplicity, we show the results obtained in the three-dimensional case, $p = 3$. Similar results were observed in larger dimensions (e.g., 10 and 50). The number of samples $n = \gamma p$, was determined as above by varying the values of the scalar, $\gamma = \{10, 20, \ldots, 100\}$. Recall that in both distributions, spherical-homoscedasticity is satisfied when $\theta_1 = 0$ and $\theta_2 = \{10, 20, \ldots, 90\}$, and when $\theta_2 = 0$ and $\theta_1 = \{10, 20, \ldots, 90\}$. The parameter matrix of the second distribution $\mathbf{A}_2$ can be defined as a rotated version of $\mathbf{A}_1$ as $\mathbf{A}_2 = \mathbf{R}^T\mathbf{A}_1\mathbf{R}$, where $\mathbf{R}$ is given by (30). Further, in Kent,

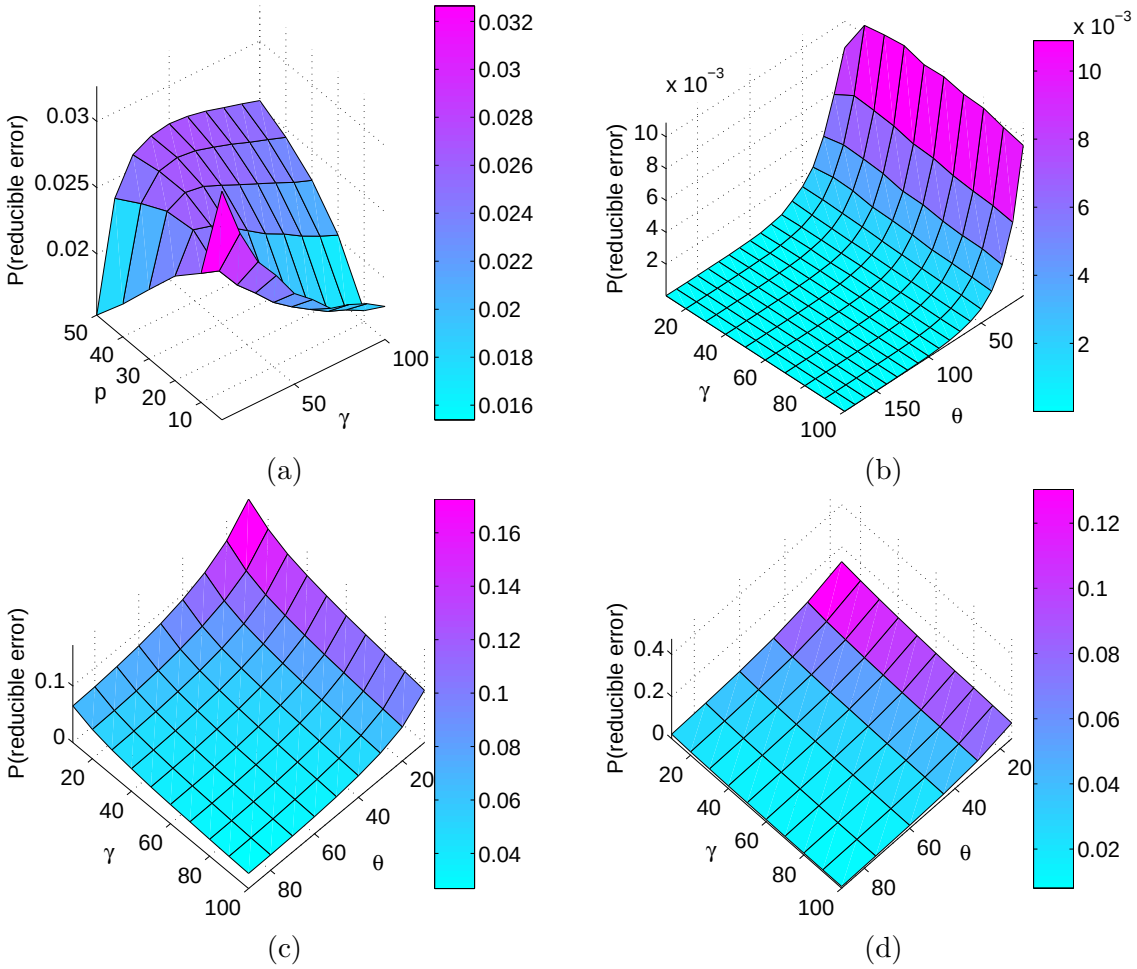

Figure 8: *(a-b) Shown here are the average reducible errors obtained when fitting two Gaussian distributions to the data sampled from two spherical-homoscedastic vMFs with parameters* $\kappa = \{1, 2, \ldots, 10\}$ *and* $\theta = \{10, 20, \ldots, 180\}$*. The number of samples is* $n = \gamma p$*, where* γ *is a constant taking values in* $\{10, 20, \ldots, 100\}$*, and* p *is the dimensionality with values* $p = \{2, 10, 20, \ldots, 50\}$*. (c) Shows the average reducible error when data sampled from spherical-homoscedastic Bingham distributions is fitted with the Gaussians estimates. In this case,* $p = 3$ *and the number of samples as defined above. (d) Summarizes the same simulation for spherical-homoscedastic Kent distributions, with* $p = 3$*.*

we impose the concentration and skewness parameters to be equal in both distributions, i.e., $\kappa_1 = \kappa_2$ and $\beta_1 = \beta_2$.

In Fig. 8(c-d) we show the probability of reducible error as a function of γ and the rotation θ. In both cases, and as expected, when the two distributions get closer, the error increases. In Bingham, when the sample to dimensionality ratio decreases, the error slightly increases (to about 6%). But, in Kent, as in vMF, reducing the number of samples produces a negligible effect. Therefore, as in vMF, we conclude that, in the worst case scenarios, keeping a ratio of 10 samples per dimension, results in good approximations. When the distributions are not in tight proximity, this number can be smaller. These observations bring us to our next topic: how to project the data onto a feature space where the ratio sample-to-dimensionality is adequate.

6.2 Subsphere projection

As demonstrated in Section 6.1, a major concern in fitting a distribution model to a dataset is the limited amount of samples available. This problem is exacerbated when the dimensionality of the data p surpasses the number of samples n. This problem is usually referred to as the curse of dimensionality. In such circumstances, the classifier can easily overfit and lead to poor classification results on an independent set of observations. A typical technique employed to mitigate this problem, is the use of Principal Components Analysis (PCA). By projecting the sample vectors onto the subspace defined by the PCs of largest data variance, we can eliminate data noise and force independent test vectors to be classified in the subspace defined by the training data. This procedure is typically carried out in $\mathbb{R}^p$ by means of an eigendecomposition of the covariance matrix of the training data. In this section, we show how to employ PCA on the correlation matrix to do the same in S^{p-1}.

This problem can be easily stated as follows: We want to find a representation $\widetilde{\mathbf{x}}_i \in S^{r-1}$ of the original feature vectors $\mathbf{x}_i \in S^{p-1}$, $r \leq p$, with minimal loss of information. This can be done with a $p \times r$ orthogonal projection matrix $\mathbf{W}$, i.e., $\mathbf{W}^T\mathbf{W} = \mathbf{I}$. The least-squares solution to this problem is then given by

$$\arg\min_{\mathbf{W}} \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{W}\widetilde{\mathbf{x}}_i)^T(\mathbf{x}_i - \mathbf{W}\widetilde{\mathbf{x}}_i), \tag{36}$$

where $\widetilde{\mathbf{x}}_i = \mathbf{W}^T\mathbf{x}_i / \|\mathbf{W}^T\mathbf{x}_i\|$, that is, the unit-length vector represented in S^{r-1}.

Note that as opposed to PCA in the Euclidean space, our solution requires the subspace vectors to have unit length, since these are also in a hypersphere. To resolve this, we can first search for that projection which minimizes the projection error,

$$\begin{aligned} & \arg\min_{\mathbf{W}} \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{W}\mathbf{W}^T\mathbf{x}_i)^T(\mathbf{x}_i - \mathbf{W}\mathbf{W}^T\mathbf{x}_i) \\ = & \arg\min_{\mathbf{W}} \sum_{i=1}^{n} (\mathbf{x}_i^T\mathbf{x}_i - 2\mathbf{x}_i^T\mathbf{W}\mathbf{W}^T\mathbf{x}_i + \mathbf{x}_i^T\mathbf{W}\mathbf{W}^T\mathbf{W}\mathbf{W}^T\mathbf{x}_i) \\ = & \arg\min_{\mathbf{W}} \sum_{i=1}^{n} (1 - \|\mathbf{W}^T\mathbf{x}_i\|^2) \end{aligned}$$

$$
\begin{aligned}
&= \arg\max_{\mathbf{W}} \sum_{i=1}^{n} \|\mathbf{W}^T\mathbf{x}_i\|^2 \\
&= \arg\max_{\mathbf{W}} \mathbf{W}^T \sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^T\mathbf{W} \\
&= \arg\max_{\mathbf{W}} \mathbf{W}^T\mathbf{S}\mathbf{W},
\end{aligned}
$$

where $\mathbf{S}$ is the sample correlation matrix. Then, the resulting data vectors need to be normalized to unit length to produce the final result.

To justify this solution, note that the direct minimization of (36) would result in

$$
\begin{aligned}
\arg\min_{\mathbf{W}} \sum_{i=1}^{n} \left(\mathbf{x}_i^T\mathbf{x}_i - 2\frac{\mathbf{x}_i^T\mathbf{W}\mathbf{W}^T\mathbf{x}_i}{\|\mathbf{W}^T\mathbf{x}_i\|} + \widetilde{\mathbf{x}}_i^T\widetilde{\mathbf{x}}_i \right) &= \arg\min_{\mathbf{W}} \sum_{i=1}^{n} \left(2 - \frac{2\mathbf{x}_i^T\mathbf{W}\mathbf{W}^T\mathbf{x}_i}{\|\mathbf{W}^T\mathbf{x}_i\|} \right) \\
&= \arg\min_{\mathbf{W}} \sum_{i=1}^{n} \left(1 - \|\mathbf{W}^T\mathbf{x}_i\| \right) \\
&= \arg\max_{\mathbf{W}} \sum_{i=1}^{n} \|\mathbf{W}^T\mathbf{x}_i\|. \qquad (37)
\end{aligned}
$$

Since we are only interested in eliminating those dimensions of S^{p-1} that are close to zero and $\|\mathbf{x}_i\| = 1$, it follows that $\|\mathbf{W}^T\mathbf{x}_i\| \cong \|\mathbf{W}^T\mathbf{x}_i\|^2$. Further, because (37) may not generate unit length vectors $\widetilde{\mathbf{x}}_i$, this will require a normalization step.

The projection to a lower dimensional sphere S^{r-1} minimizing the least-squares error can hence be carried out in two simple steps. First, project the data onto the subspace that keeps most of the variance as defined by the autocorrelation matrix, i.e., $\check{\mathbf{x}}_i = \mathbf{W}^T\mathbf{x}_i$, where $\mathbf{S}\mathbf{W} = \mathbf{W}\Lambda$. Second, normalize the vectors to produce the following final result $\widetilde{\mathbf{x}}_i = \check{\mathbf{x}}_i/\|\check{\mathbf{x}}_i\|$.

6.3 Object categorization

As already mentioned in the Introduction, norm normalization is a typical pre-processing step in many systems, including object recognition and categorization. In the latter problem, images of objects need to be classified according to a set of pre-defined categories, e.g., cows and cars. A commonly used database for testing such systems is the ETH-80 dataset of Leibe and Schiele (2003). This database includes the images of eight categories: apples, cars, cows, cups, dogs, horses, pears and tomatoes. Each of these consists of a set of images representing ten different objects (e.g., ten cars) photographed at a total of 41 orientations. This means that we have a total of 410 images per category.

In an attempt to emulate the primary visual areas of the human visual system, many feature representations of objects are based on the outputs of a set of filters corresponding to the derivatives of the Gaussian distribution at different scales. In our experiments, we consider the first derivative about the x and y axes for a total of three different scales, i.e., $\upsilon = \{1, 2, 4\}$, where υ is the variance of the Gaussian filter. The convolution of each of these filters with the input images produces a different result, which means we will generate

a total of six images. Next, we compute the histogram of each of these resulting images. This histogram representation is simplified to 32 intensity intervals. These histograms are generally assumed to be the distribution of the pixel values in the image and, therefore, the sum of the values over all 32 intervals should be one (because the integral of a density is 1). Hence, these 32 values are first represented in vector form and then normalized to have unit norm. The resulting unit-norm feature vectors are concatenated to form a single feature representation. This generates a feature space of 192 dimensions. We refer to this feature representation as *Gauss*.

As an alternative, we also experimented with the use of the magnitude of the gradient and the Laplacian operator. The latter generally used for its rotation-invariant properties, which is provided by its symmetric property. As above, the gradient and Laplacian are computed using three scales, i.e., $\upsilon = \{1, 2, 4\}$. To reduce the number of features, we use the histogram representation described above, which again produces unit-norm feature vectors of 192 dimensions. This second representation will be referred to as *MagLap*.

These two image representations are tested using the leave-one-object-out strategy, where, at each iteration, we leave all the images of one of the objects out for testing and use the remaining for training. This is repeated (iterated) for each of the possible objects that one can leave out.

We now use the PCA procedure described in Section 6.2 to map the data of each of the two feature spaces described above onto the surface of an 18-dimensional sphere. This dimensionality was chosen such that 99.9% of the data variance was kept. Note that in this subsphere, the sample to dimensionality ratio (which is about 180) is more than adequate for classification purposes as demonstrated in Section 6.1.

To generate our results, we first use the vMF and Bingham approximations introduced in Section 2.2 and then utilize (7) and (8) for classification. We refer to these approaches as *vMF* and *Bingham classifiers*, respectively. These are the classifiers one can construct based on the available approximation defined to estimate the parameters of spherical distributions.

Next, we consider the Gaussian equivalency results shown in this paper, which allow us to represent the data with Gaussian distributions and then use (6) for classification. This algorithm will be labelled *Gaussian classifier* in our tables.

In Section 5, we derived three algorithms for classifying spherical-homoscedastic vMF, Bingham, and Kent. These were obtained from Theorems 3, 7 and 10. We also showed how the vMF and Kent classifiers were identical. These classifiers were given in (31) and (34), and will be labelled *SH-vMF* and *SH-Bingham*, respectively.

Table 1 shows the average of the recognition rates using the leave-one-object out with each of the algorithms just mentioned. In this table, we have also included the results we would obtained with the classical Fisher (1938) Linear Discriminant Analysis algorithm (LDA). This algorithm is typically used in classification problems in computer vision, genomics and other machine learning applications. In LDA, the data of each class is approximated with a Gaussian distribution, but only the average of these, $\bar{\Sigma}$, is subsequently used. This is combined with the scatter-matrix of the class means $\mathbf{S}_B$ to generate the LDA basis vectors $\mathbf{U}$ from $\bar{\Sigma}^{-1}\mathbf{S}_B\mathbf{U} = \mathbf{U}\Lambda_{LDA}$; where $\mathbf{S}_B$ is formally defined as $\mathbf{S}_B = \sum_{i=1}^{C} n_i/n \, (\hat{\mu}_i - \hat{\mu}) \, (\hat{\mu}_i - \hat{\mu})^T$, n_i is the number of samples in class i, $\hat{\mu}_i$ the mean of

Method	*vMF*	*Bingham*	*Gaussian*	*SH-vMF*	*SH-Bingham*	*LDA*
Gauss	13.75	73.11	73.14	45.85	46.16	62.9
MagLap	12.5	74.18	73.75	51.95	52.90	66.25

Table 1: Average classification rates for the ETH database with *Gauss* and *MagLap* feature representations.

these samples, and $\hat{\mu}$ the global mean. In LDA, the nearest mean classifier is generally used in the subspace spanned by the eigenvectors of $\mathbf{U}$ associated to non-zero variance.[9]

It is clear that, in this particular case, the vMF model is too simplistic to successfully represent our data. In comparison, the Bingham model provides a sufficient degree of variability to fit the data.

Next, we turn to the results of the Gaussian fit defined in this paper, Table 1. We see that the results are comparable to those obtained with the Bingham model. A simple data analysis reveals that the data of all classes is highly concentrated. We see this by looking at the eigenvalues of each class scatter matrix $\mathbf{S}_a$, $a = \{1, \ldots, C\}$. The average of the variance ratios between the first eigenvector (defining the mean direction) and the second eigenvector (representing the largest variance of the data on S^{d-1}), which is 66.818. From the results obtained in Section 4, we expected the Gaussian fit to perform similarly to Bingham under such circumstances. This is clearly the case in the classification results shown in Table 1.

While the Gaussian fit has proven adequate to represent the data, the spherical-homoscedastic classifiers derived in Section 5 performed worse. This is because the class distributions are far from spherical-homoscedastic. We can see that by studying the variability of the variance in each of the eigenvectors of the class distributions. To do this, we look at the variance $\hat{\lambda}_{a_i}$ about each eigenvector $\hat{\mathbf{q}}_{a_i}$. If the data were spherical-homoscedastic, the variances about the corresponding eigenvectors should be identical, i.e, $\hat{\lambda}_{a_i} = \hat{\lambda}_{b_i}$, $\forall a, b$ (recall a and b are the labels of any two distributions). Seemingly, the more the class distributions deviate from spherical-homoscedastic, the larger the difference between $\hat{\lambda}_{a_i}$ and $\hat{\lambda}_{b_i}$ will be. The percentage of variability among the variances $\hat{\lambda}_{a_i}$ for different a, can be computed as

$$100\,\frac{stdv\{\hat{\lambda}_{1_i}, \ldots, \hat{\lambda}_{C_i}\}}{\frac{1}{C}\sum_{j=1}^{C}\hat{\lambda}_{j_i}}, \tag{38}$$

where $stdv\{\cdot\}$ is the standard deviation of the values of the specified set. Hence, a 0% variability would correspond to perfect spherical-homoscedastic distributions. The more we deviate from this value, the more the distributions will deviate from the spherical-homoscedastic model.

Applying (38) to the resulting class distributions of the ETH dataset with the *Gauss* representation and then computing the mean of all resulting differences yields 58.04%. The same can be done for the *MagLap* representation, resulting in an average variability of 76.51%. In these two cases, we clearly see that the class distributions deviate considerably from spherical-homoscedastic. As demonstrated in Figure 6(c). The effects of heteroscedas-

9. Note that the rank of $\mathbf{U}$ is upper-bounded by $C-1$, because the scatter-matrix matrix $\mathbf{S}_B$ is defined by the $C-1$ vectors interconnecting the C class means.

Method	*K-SH-vMF*	*K-SH-Bingham*
Gauss	79.24 ($\bar{\varsigma} = 3.2$)	78.84 ($\bar{\varsigma} = 3.96$, $\bar{v} = 1$)
MagLap	77.23 ($\bar{\varsigma} = 5.63$)	77.16 ($\bar{\varsigma} = 6.35$, $\bar{v} = 1$)

Table 2: Average classification rates for the ETH database with *Gauss* and *MagLap* feature representations using the nonlinear spherical-homoscedastic vMF and Bingham classifiers. In these results, we used the Mahalanobis kernel.

ticity are further observed in the low performances of the LDA algorithm, which assumes the data is homoscedastic.

As shown in Section 5, we can improve the results of our spherical-homoscedastic classifiers by first (intrinsically) mapping the data into a space where the underlying class distributions fit to the spherical-homoscedastic model.

Since the original class distributions need to be reshaped to fit the spherical-homoscedastic model, the Mahalanobis kernel is a convenient choice. This kernel is defined as

$$k(\mathbf{x}, \mathbf{y}) = \exp\left(\frac{-(\mathbf{x}-\mathbf{y})^T \bar{\Sigma}^{-1} (\mathbf{x}-\mathbf{y})}{2\varsigma^2}\right)$$

where ς is a scale parameter to be optimized using the leave-one-object-out test (looot) on the training set. This looot is not to be confused with the object left out for testing, which is utilized to compute the classification error. In our experimental results, we tested the following scalar values for the kernel, $\varsigma = \{1, 2, \ldots, 10\}$.

Table 2 shows the recognition rates for the (nonlinear) spherical-homoscedastic vMF and Bingham classifiers, using the Mahalanobis kernel just defined. These classifiers are labelled, *K-SH-vMF* and *K-SH-Bingham*, respectively. The classification rates are shown in percentages. The values in parentheses indicate the average over all scale parameters as determined by looot. This includes the average scalar kernel parameter, denoted $\bar{\varsigma}$, for vMF and Bingham, and the average value of i in $\mathbf{q}_{a_i}$, denoted $\bar{v}$, when optimizing the Bingham classifier defined in (35). We see that the classification results are boosted beyond those obtained with Bingham and the Gaussian fit.

6.4 Gene expression classification

Spherical representations can be applied to gene expression data in a variety of ways. For example, some genome sequences are circular, and the gene co-orientation as well as its expression are relevant (Audit and Ouzounis, 2003; Janssen et al., 2001). In analyzing other genomes or the expression of a set of genes from microarrays, correlation coefficients (e.g. Pearson) provide a common way to determine the similarity between samples (Banerjee et al., 2005). These are generally norm invariant and, hence, whenever density estimations are required, spherical models are necessary.

To demonstrate this application, we have used the gene expression dataset A of Pomeroy et al. (2002). This dataset consists of five different classes of tumors in the central nervous system: medulloblastomas, malignant gliomas, AT/RT (atypical teratoid/rhabdoid

tumors), normal cerebellums and supratentorial PNETs (primitive neuroectodermal tumors). The first three classes each has a total of ten samples. The fourth class includes four samples. And, the fifth class, has a total of 8 samples. Each sample is described with $7,132$ features corresponding to the expression level from a set of genes. The goal is to classify each of these gene expression samples into their correct class. This can be done using the leave-one-out test (loot), where we use all but one of the samples for training and determine whether our algorithm can correctly classify the sample left out.

Before we can use the data of Pomeroy et al. (2002), a set of normalization steps are required. First, it is typical to threshold the expression levels in each microarray to conform to a minimum expression value of 100 and a maximum of $1,600$. Second, the maximum and minimum of each feature, across all samples, is computed. The features with low-variance, i.e., those that have $max/min < 12$ and $max - min < 1200$, do not carry significant class information and are therefore eliminated from consecutive analysis. This results in a feature representation of $3,517$. Finally, to facilitate the use of correlation-based classifiers, the variance of each of the samples is norm-normalized. This maps the feature vectors to S^{3516}.

Since the number of samples is only 42, we require to project the data onto a subsphere (Section 6.2) to a dimensionality where the sample-to-dimension ratio is appropriate. By reducing the dimensionality of our space to that of the range of the data, we get an average sample-to-dimension ratio of 1.18. Note that this ratio could not be reduced further, because the number of samples in gene expression analysis is generally very small (in this particular case 10 or less samples per class). Reducing the dimensionality further would impair our ability to analyze the data efficiently.

The average recognition rates obtained with each of the algorithms described earlier with loot are in Table 3. The small sample-to-dimension ratio, characteristic of this gene expression datasets, makes most of the algorithms perform poorly. For example, Fisher's LDA is about as bad as a random classifier which assigns the test vector to a class randomly. And, in particular, the Bingham approximation could not be completed with accuracy and its classifier is even worse than LDA's. As opposed to the results with the ETH database, when one has such a small number of samples, distributions with few parameters will usually produce better results. This is clearly the case here, with vMF and Gaussian providing a better fit than Bingham. To resolve the issues caused by the small sample-to-dimension ratio, one could regularize the covariance matrix of each class to allow the classifier to better adapt to the data (Friedman, 1989). By doing this, we were able to boost the results of our Gaussian fit to around 80%. At each iteration of the loot, the regularization parameter (also optimized with loot over the training data) selected 90% of the actual covariance matrix and 10% for the regularizing identity matrix term. We can now calculate how close to spherical-homoscedastic these regularized distributions are. As above, we can do this by estimating the average of percentage variance from the median of the eigenvalues obtained from the distribution of each class.[10] This results in $\sim 0.08\%$ deviation, which means the data can be very well represented with spherical-homoscedastic distributions. These results imply that the regularized Gaussian fit will outperform the other estimates, which is indeed the case. Since the classifiers derived in Section 5 are also based on the

10. We restricted the comparison to the first three eigenvalues, because the rest were many time zero due to the singularity of some of the class covariance matrices.

Method	*vMF*	*Bingham*	*Gaussian*	*SH-vMF*	*SH-Bingham*	*LDA*
Dataset A	28.57	14.29	33.33	80.95	85.71	21.43
ALL-AML	41.18	41.18	41.18	94.12	94.12	91.18

Table 3: Average classification rates on the gene expression data of Pomeroy et al. (2002) and Golub et al. (1999).

Method	*K-SH-vMF*	*K-SH-Bingham*
Dataset A	88.10 ($\bar{\varsigma} = 0.1$)	90.48 ($\bar{\varsigma} = 0.2$, $\bar{v} = 1.11$)
ALL-AML	85.29 ($\bar{\varsigma} = 0.7$)	85.29 ($\bar{\varsigma} = 0.7$, $\bar{v} = 1$)

Table 4: Average classification rates on the gene expression data using the Mahalanobis kernel. The values in parentheses correspond to the average parameters of the classifiers. These have been optimized using the leave-one-sample-out strategy on the training data.

assumption of spherical-homoscedasticity, these should also perform well in this dataset. We see in the results of Table 3 that these results were also the best. Furthermore, the spherical-homoscedastic classifiers do not require of any regularization. This means that the computation associated to these is very low, reversing the original problem associated to the (non-linear) estimation of the parameters of the distributions – making the *SH-Bingham* the best fit.

Further investigation of the spherical-homoscedastic classifiers in a higher dimensional nonlinear space by means of the Mahalanobis kernels defined above shows that these results can be further improved. In these experiments, the kernel parameters are optimized with loot over the training set. As shown in Table 4, the performances of the *SH-vMF* and *SH-Bingham* classifiers improved to 88.10% and 90.48%, respectively.

Next, we tested our algorithms on the RNA gene expression dataset of Golub et al. (1999). The training set consists of a total of 38 bone marrow samples, 27 of which correspond to Acute Lymphoblastic Leukemia (ALL) and 11 to Acute Myeloid Leukemia (AML). the testing set includes 20 ALL and 14 AML samples, respectively. Each feature vector is constructed with a total of $7,129$ probes from $6,817$ human genes, where a probe is a labelled subset of the original set of bases of a gene.

The feature vectors in this dataset are norm-normalized to eliminate the variance changes across the samples. We then used the subsphere projection technique to represent the training set on a subsphere of dimensionality equal to the range space of the data, resulting in a sample-to-dimension ratio of 1.

The results obtained using the independent testing set on each of the trained classifiers are shown in Table 3. Once again, we see that the sample-to-dimension ratio is too small to allow good results in the estimate of the parameters of the spherical distributions or the Gaussian fit. As we did above, one can improve the Gaussian fit with the use of a regularization parameter, yielding $\sim 94\%$ classification rate (with a regularization of about 0.2). This is superior to the result of LDA, which in this case is much better than before.

Method	*vMF*	*Gaussian*	*SH-vMF*	*SH-Bingham*	*LDA*
Classic	38.64	21.23	89.67	89.23 ($\bar{v} = 1$)	76.40
CMU	57.33	33.33	69.07	68 ($\bar{v} = 1$)	34.53

Table 5: Average classification rate on text datasets.

The LDA results are also surpassed by those of the spherical-homoscedastic classifiers, which again work well because the data can be estimated with spherical-homoscedastic distributions.

We note that the results obtained with the spherical-homoscedastic classifiers are already very good. Adding another optimization step to select the most appropriate kernel parameter may be too much to ask from such a small number of samples. We see in the results of Table 4 (which the use of the Mahalanobis kernel) this actually resulted in poorer classifications. This can be further studied with the use of the polynomial kernel, defined as

$$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \mathbf{y})^d,$$

where d defines the degree. If we substitute the Mahalanobis kernel for this one in our implementation and optimize d with loot, we find that this is given at $d = 1$. That is, the best polynomial kernel is that which does not change the space, and the classification results will be the same as those shown in Table 3.

7. Text Dataset

In text recognition the documents are usually represented as a bag of words, which can be described in vector form by assigning to each feature the frequency of each of the words in the document. In this representation, the important information is that of the frequency of a word with respect to the rest (i.e., percentage of occurrence). This can be easily obtained by norm-normalizing these feature vectors, which maps the data into a spherical representation.

The first dataset (labelled *Classic*) we will test is composed of three classes.[11] The first class is a collection of documents from MEDLINE (the index of medical related papers), which includes 1,033 documents. The second class consists of 1,460 documents from the CISI database. The third class includes 1,400 documents from the aeronautical system database CRANFIELD. The second dataset (labelled *CMU*) we will use, is a subset of the CMU set.[12] This corresponds to 1,500 randomly selected documents, 500 from each of the following three classes: newsgroups comp.graphics, comp.os.ms-windows.misc and comp.windows.x.

The preprocessing of the data, which is common to the two datasets, includes eliminating high and low frequency words as well as those words that have less than 3 letters (examples are, "a", "and","the"). After this preprocessing, each feature vector in the *Classic* dataset consists of 4,427 dimensions and those in the *CMU* dataset 3,006 dimensions.

11. Available at ftp://ftp.cs.cornell.edu/pub/smart/.
12. Available at http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html.

Method	*K-SH-vMF* Polynomial	*K-SH-Bingham* Polynomial	*K-SH-vMF* RBF	*K-SH-Bingham* RBF
Classic	90.21 ($\bar{d} = 9.8$)	88.92 ($\bar{d} = 2.3$, $\bar{v} = 1$)	90.54 ($\bar{\varsigma} = 0.3$)	88.23 ($\bar{\varsigma} = 0.7$, $\bar{v} = 1$)
CMU	71.13 ($\bar{d} = 7.9$)	68.27 ($\bar{d} = 5.4$, $\bar{v} = 1$)	75.27 ($\bar{\varsigma} = 0.22$)	69.33 ($\bar{\varsigma} = 0.50$, $\bar{v} = 1$)

Table 6: Average classification rate on text datasets using kernel extension. The value of $\bar{d}$ specifies the average degree of the polynomial kernel optimized over the training set.

We have done a 10-fold cross-validation on these datasets using the algorithms described above. This means that we kept 10% of the dataset for testing and fit the distribution models or classifiers to the remaining 90%. This was repeated 10 times and the average classification rates are shown in Table 5.

The resulting sample vectors are sparse – described in a high dimensional feature space with most of the features equal to zero. This makes the estimation problem very difficult. For instance, this sparseness did not permit computation of the normalizing constant of the Bingham distribution with the saddlepoint approximation of Kume and Wood (2005). For this reason Table 5 does not provide recognition rates for Binghams. The sparseness of the data did not allow for a good estimate of the parameters of the vMF or Gaussian modeling either, as is made evident in the poor results shown in Table 5. The Gaussian modeling result, in particular, can be improved in two ways. One, as above, would correspond to using a regularization term. A second option corresponds to calculating the average class covariance matrix of all Gaussians and use this as a common covariance matrix. This is in fact LDA's results, which is much better. Since eliminating the bases with lowest variance can reduce noise, we can further improve this results to about 98% for the Classical dataset and 64% for CMU, by using the spherical projection of Section 6.2.

As we have shown above, these issues are generally less problematic when using the spherical-homoscedastic classifiers derived in this paper. The reason for that is given by the simplicity of these classifiers, which facilitates robustness. Yet, these results can be boosted by using the kernel classifiers derived above. To do this, we first note that we should not use the Mahalanobis kernel on these datasets, because the Gaussian fits will be biased by the sparseness. For this reason, we have used the polynomial kernel given above and the RBF kernel defined as $k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\varsigma^2}\right)$. As demonstrated in Table 6, the results improved to about 90% on the Classic dataset and over 75% for CMU.

8. Conclusions

In this paper, we investigated the effect of modelling von Mises-Fisher, Bingham and Kent distributions using Gaussian distributions. We first introduced the concept of spherical-homoscedasticity and showed that if two spherical distributions comply with this model, their Gaussian approximations are enough to obtain optimal classification performance in the Bayes sense. This was shown to be true for the von Mises-Fisher and Bingham distribution. For the Kent distribution the additional criteria defined in (29) must hold.

We further investigated what happens if we model the data sampled from two spherical-heteroscedastic distributions using their Gaussian approximations. In such a scenario, the Gaussian modelling will result in a decision boundary different to that produced by the original spherical distributions. We have referred to the additional error caused by this approximation as the reducible error. We have then empirically evaluated this and showed that as the two distributions start to deviate from spherical-homoscedastic, the probability of reducible error increases. We have also stated the particular cases where two spherical-heteroscedastic distributions may lead to a small error. For example, for vMFs this happens when these are highly concentrated, and for Bingham and Kent when the variance parameters are the same.

Since spherical-homoscedasticity provides an optimal model for parameter estimation, we were also able to define classifiers based on these. These classifiers are linear, since the Bayes decision boundary for two spherical-homoscedastic distributions in S^{p-1} is a hyperplane (same as for homoscedastic distributions in $\mathbb{R}^p$). When the data is spherical-heteroscedastic, we can first map the data into a space where the projected distributions adapt to the spherical-homoscedastic model. With this, we can use our spherical-homoscedastic classifiers in a large number of datasets. Finally, this can be efficiently implemented using the idea of the kernel trick as shown in Section 5. We have shown how all these results apply to a variety of problems in object recognition, gene expression classification, and text organization.

Acknowledgements

We thank the referees for their insightful comments. Thanks also go to Lee Potter for discussion. This research was partially supported by the National Institutes of Health under grant R01-DC-005241.

Appendix A: Notation

$\mathbf{x}$	feature vector
p	dimensionality of the original feature space
S^{p-1}	$(p-1)$-dimensional unit sphere in $\mathbb{R}^p$
$SO(p)$	p-dimensional Special Orthogonal Group
κ	concentration of our spherical distribution
β	ovalness of the Kent distribution
μ	mean direction vector
$\mathbf{m}$	mean feature vector
$\Gamma(\cdot)$	Gamma function
$I_\nu(\cdot)$	Bessel function of the first kind and order ν
$\mathbf{R}$	rotation matrix
Σ	covariance matrix
$\mathbf{S}$	autocorrelation (scatter) matrix
$\bar{\mathbf{S}}$	scatter matrix calculated on the null space of the mean direction
$\mathbf{A}$	parameter matrix for the Bingham and Fisher-Bingham
$\mathbf{K}$	gram matrix
$\mathbf{v}_i$	i^{th} eigenvector of the covariance matrix
$\mathbf{q}_i$	i^{th} eigenvector of the parameter matrix $\mathbf{A}$
λ_i	i^{th} eigenvalue of a symmetric matrix
$N(\mu, \Sigma)$	Gaussian (Normal) distribution
$M(\mu, \kappa)$	von Mises-Fisher distribution
$B(\mathbf{A})$	Bingham distribution
$K(\mu, \kappa, \mathbf{A})$	Kent distribution
$FB(\mu, \kappa, \mathbf{A})$	Fisher-Bingham distribution
$E(\cdot)$	expected value
ς	scale parameter
θ, ϕ, ω	rotation angles
$\mathbf{k}(.,.)$	mercer kernel
$\phi(.)$	vector mapping function
$\Phi(.)$	matrix mapping function

References

B. Audit and C.A. Ouzounis. From genes to genomes: Universal scale-invariant properties of microbial chromosome organisation. *Journal of Molecular Biology*, 332(3):617–633, 2003.

A. Banerjee, I.S. Dhillon, J. Ghosh, and S. Sra. Clustering on the unit hypersphere using von Mises-Fisher distributions. *Journal of Machine Learning Research*, 6:1345–1382, 2005.

P.N. Belhumeur and D.J. Kriegman. What is the set of images of an object under all possible lighting conditions? *International Journal of Computer Vision*, 28(3):245–260, 1998.

C. Bingham. An antipodally symmetric distribution on the sphere. *Annals of Statistics*, 2 (6):1201–1225, 1974.

I.S. Dhillon and D.S. Modha. Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1):143–175, 2001.

I.L. Dryden and K.V. Mardia. *Statistical Shape Analysis*. John Wiley & Sons, West Sussex, England, 1998.

R.A. Fisher. The statistical utilization of multiple measurements. *Annals of Eugenics*, 8: 376–386, 1938.

J.H. Friedman. Regularized discriminant analysis. *Journal of The American Statistical Association*, 84(405):165–175, 1989.

T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

P.J. Janssen, B. Audit, and C.A. Ouzounis. Strain-specific genes of Helicobacter pylori: distribution, function and dynamics. *Nucleic Acids Research*, 29(21):4395–4404, 2001.

O. Javed, M. Shah, and D. Comaniciu. A probabilistic framework for object recognition in video. In *Proceedings of International Conference on Image Processing*, pages 2713–2716, 2004.

C.M. Joshi. Some inequalities for modified Bessel functions. *Journal of the Australian Mathematics Society, Series A*, 50:333–342, 1991.

J.T. Kent. The Fisher-Bingham distribution on the sphere. *Journal of the Royal Statistical Society, Series B (Methodological)*, 44:71–80, 1982.

P. Koev and A. Edelman. The efficient evaluation of the hypergeometric function of a matrix argument. *Mathematics of Computation*, 75:833–846, 2006.

A. Kume and A. T. A. Wood. Saddlepoint approximations for the Bingham and Fisher-Bingham normalising constants. *Biometrika*, 92:465–476, 2005.

B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

K.V. Mardia and P.E. Jupp. *Directional Statistics*. John Wiley & Sons, West Sussex, England, 2000.

H. Murase and S.K. Nayar. Visual learning and recognition of 3D objects from appearance. *International Journal of Computer Vision*, 14:5–24, 1995.

S. L. Pomeroy, P. Tamayo, M. Gaasenbeek, L. M. Sturla, M. Angelo, M. E. McLaughlin, J. Y. H. Kim, L. C. Goumnerova, P. M. Black, C. Lau, J. C. Allen, D. Zagzag, J. M. Olson, T. Curran, C. Wetmore, J. A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D. N. Louis, J. P. Mesirov, E. S. Lander, and T. R. Golub. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, pages 436–442, 2002.

D.E. Slice, editor. *Modern Morphometrics in Physical Anthropology*. Kluwer Academics, New York, NY, 2005.

C. M. Theobald. An inequality for the trace of the product of two symmetric matrices. *Proc. Cambridge Phil. Soc.*, 77:265–267, 1975.

A. Veeraraghavan, R.K. Roy-Chowdhury, and R. Chellappa. Matching shape sequences in video with applications in human movement analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1896–1909, 2005.

L. Wang, T. Tan, W. Hu, and H. Ning. Automatic gait recognition based on statistical shape analysis. *IEEE Transactions on Image Processing*, 12(9):1120–1131, 2003.