Extrinsic Kernel Ridge Regression Classifier for Kendall's Planar Shape Space

Hwiyoung Lee

Department of Statistics, Florida State University

Overview

1. Extrinsic Analysis
2. Introduction of the Planar Shape Space
3. Regression Classifier
4. Kernel Ridge Regression Classifier on Σ_2^k
5. Real Data Analysis

Defining a Mean on $\mathcal{M}$

Let $\mathbf{X}$ be a random variable on $\mathbb{R}^p$

$$\boldsymbol{\mu} = \mathbb{E}(\mathbf{X}) = \int \mathbf{x}\mathcal{Q}(d\mathbf{x}) \tag{1}$$

Q. What if $\mathbf{X}$ is a random object on $(\mathcal{M}, \rho)$?

- The expectation in (1) can not generalize mean on $\mathcal{M}$
- A new definition of mean is needed

Idea. Consider a function $\sigma^2(\cdot) : \mathbb{R}^p \to \mathbb{R}$

$$\sigma^2(\mathbf{y}) = \int \|\mathbf{x} - \mathbf{y}\|^2 \mathcal{Q}(d\mathbf{x})$$

- $\sigma^2(\cdot)$ is minimized at $\mathbf{y} = \mathbb{E}(\mathbf{X})$

Fréchet Mean

Let $\mathbf{X} \in \mathcal{M}$ be an random object in a metric space $(\mathcal{M}, \rho)$

- For $\mathbf{q} \in \mathcal{M}$ consider the Fréchet function (Fréchet, 1948)

$$\mathcal{F}(\mathbf{q}) = \int_{\mathcal{M}} \rho^2(\mathbf{x}, \mathbf{q}) \mathcal{Q}(d\mathbf{x}),$$

where $P(\cdot)$ denotes the probability distribution of $\mathbf{X}$.

- **Fréchet mean**

$$\boldsymbol{\mu} = \underset{\mathbf{q} \in \mathcal{M}}{\operatorname{argmin}} \int_{\mathcal{M}} \rho^2(\mathbf{x}, \mathbf{q}) \mathcal{Q}(d\mathbf{x}), \tag{2}$$

where ρ denotes generic metric on $\mathcal{M}$.

Extrinsic vs Intrinsic Data Analysis

Intrinsic Data Analysis

- Geodesic distance
- Intrinsic Mean :

$$\boldsymbol{\mu}_I = \underset{\mathbf{q}\in\mathcal{M}}{\operatorname{argmin}} \int_{\mathcal{M}} \rho_R^2(\mathbf{x},\mathbf{q})\mathcal{Q}(d\mathbf{x})$$

- Riemannian optimization algorithm

Extrinsic Data Analysis

- Euclidean distance induced by the embedding $J : \mathcal{M} \to E^d$
- Extrinsic Mean :

$$\boldsymbol{\mu}_E = \underset{\mathbf{q}\in\mathcal{M}}{\operatorname{argmin}} \int_{\mathcal{M}} \|J(\mathbf{x}) - J(\mathbf{q})\|^2\mathcal{Q}(d\mathbf{x})$$

- J is not unique in general
- Not all choices of embedding J lead to a good estimation result.

Sample Extrinsic Mean

Given data $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^n \in \mathcal{M}$ consisting i.i.d copies of $\mathbf{X}$

$$\begin{aligned}\overline{\mathbf{X}}_E &= \operatorname*{argmin}_{\mathbf{q}\in\mathcal{M}} \sum_{i=1}^n \|J(\mathbf{x}_i) - J(\mathbf{q})\|^2 \\ &= J^{-1}\Bigg(\mathcal{P}\Bigg(\underbrace{\operatorname*{argmin}_{\boldsymbol{m}\in E^d} \sum_{i=1}^n \|J(\mathbf{x}_i) - \boldsymbol{m}\|^2}_{\widetilde{\boldsymbol{m}}}\Bigg)\Bigg),\end{aligned}$$

where

- $\tilde{\boldsymbol{m}} = \sum_{i=1}^n J(\mathbf{x}_i)/n$
- $\mathcal{P} : E^d \to J(\mathcal{M})$

See Bhattacharya and Patrangenaru (2003, 2005)

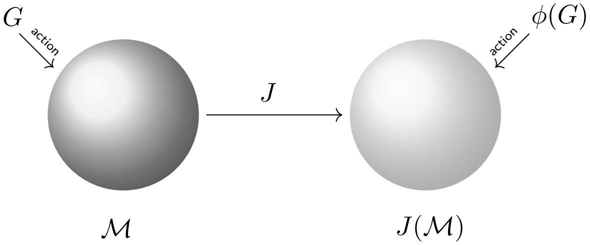
Equivariant Embedding

Let G be a Lie group, for any $g \in G$, and $\mathbf{p} \in \mathcal{M}$, if there exist a Lie group homomorphism $\phi : G \to GL(d, \mathbb{R})$ satisfying

$$J(g\mathbf{p}) = \phi(g)J(\mathbf{p}), \qquad (3)$$

then the above J is called the G equivariant embedding

- The image of $\mathcal{M}$ under the group action of the Lie group G is preserved by the group action of $\phi(G)$ on $J(\mathcal{M})$



- Equivariant embedding preserves a substantial amount of geometry

Kendall's Planar Shape Space of k-ads : Preshape Space

- k-ads on the plane can be represented as a set of k complex numbers

$$\mathbf{z} = (z_1, \cdots, z_k), \text{ where } z_j = x_j + iy_j$$

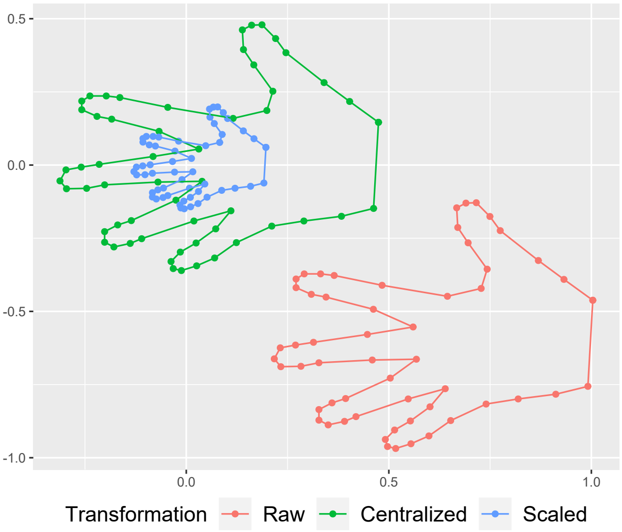
- These configurations can be mapped to the pre-shape space by removing the effect of translation and scaling

$$\mathbf{u} = \frac{\mathbf{z} - \langle \mathbf{z} \rangle}{\|\mathbf{z} - \langle \mathbf{z} \rangle\|} \in \mathbb{C}S^{k-1}, \text{ where } \langle \mathbf{z} \rangle = (\bar{\mathbf{z}}, \cdots, \bar{\mathbf{z}}), \ \bar{\mathbf{z}} = \frac{1}{k} \sum_{i=1}^{k} z_i$$

- The pre-shape space $\mathbb{C}S^{k-1}$ can be identified as a complex hypersphere

$$\mathbb{C}S^{k-1} = \left\{ \mathbf{u} \in \mathbb{C}^k \mid \sum_{j=1}^{k} u_j = 0, \|\mathbf{u}\| = 1 \right\}.$$

Graphical Illustration of Preshape



Kendall's Planar Shape Space of k-ads : Σ_2^k

- Shape is defined as an orbit of $\mathbf{u} \in \mathbb{C}S^{k-1}$ by filtering out rotation

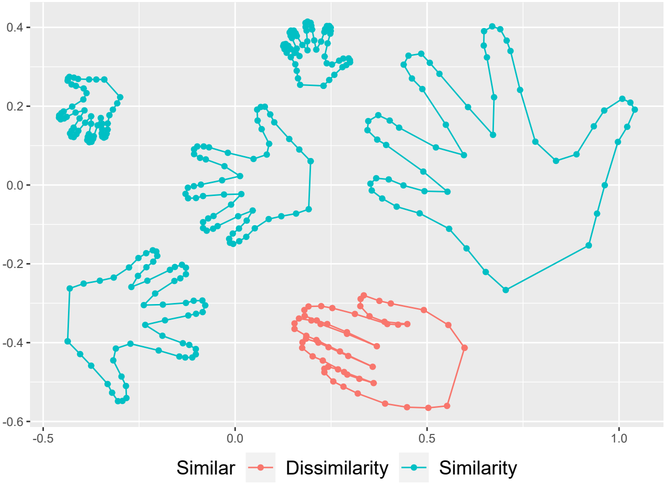
$$[\mathbf{z}] = \left\{e^{i\theta}\mathbf{u} : -\pi < \theta \leqslant \pi\right\} \in \mathbb{C}S^{k-1}/SO(2) = \Sigma_2^k$$

Alternatively, the k-ad can be represented as

$$\{\lambda(\mathbf{z} - \langle\mathbf{z}\rangle) : \lambda \in \mathbb{C}\backslash\{0\}\}\ ,\ \text{where } \lambda = re^{i\theta}$$

- The set of all complex lines through the origin in $k-1$ dimensional complex hyperplane, $\mathbb{H}_{k-1} = \{\mathbf{w} \in \mathbb{C}_k\backslash\{0\} : \sum_{i=1}^{k} w_i = 0\}$
- Thus, $\Sigma_2^k \simeq \mathbb{C}P^{k-2}$: the complex projective space

Graphical Illustration of Σ_2^k



Extrinsic Analysis on Σ_2^k

Veronese–Whitney embedding is typically used (Kent, 1992)

$$J : \Sigma_2^k \to \mathcal{S}(k, \mathbb{C})\ ;\ J([\mathbf{z}]) \mapsto \mathbf{u}\mathbf{u}^*, \tag{4}$$

where $\mathbf{u}^*$ denotes the complex conjugate transpose of $\mathbf{u} \in \mathbb{C}S^{k-1}$, and $\mathcal{S}(k, \mathbb{C})$ is a space of $k \times k$ Hermitian matrices ($\mathbf{A} = \mathbf{A}^*$)

- The image of Σ_2^k under VW embedding :

$$J(\Sigma_2^k) = \{\mathbf{A} \in \mathcal{S}^+(k, \mathbb{C}) : \mathrm{rank}(\mathbf{A}) = 1, \mathrm{Trace}(\mathbf{A}) = 1, \mathbf{A}\mathbf{1}_k = \mathbf{0}\}$$

- Alternatively, Using preshape u with $(k-1)$ Helmertized coordinates embed the shape into $\mathcal{S}(k-1, \mathbb{C})$

$$J(\Sigma_2^k) = \{\mathbf{A} \in \mathcal{S}^+(k-1, \mathbb{C}) : \mathrm{rank}(\mathbf{A}) = 1, \mathrm{Trace}(\mathbf{A}) = 1\}$$

See Bhattacharya and Bhattacharya (2012)

Extrinsic Distance

Squared extrinsic distance :

$$\rho_E^2([\mathbf{z}_1],[\mathbf{z}_2]) = \|J([\mathbf{z}_1]) - J([\mathbf{z}_2])\|_F^2, \text{ where}$$

$\|\cdot\|_F$ is the Frobenius norm, i.e., for any $J([\mathbf{z}_1]), J([\mathbf{z}_2]) \in \mathcal{S}(k,\mathbb{C})$,

$$\begin{aligned}\|J([\mathbf{z}_1]) - J([\mathbf{z}_2])\|_F^2 &= \text{Trace}\Big(\Big(J([\mathbf{z}_1]) - J([\mathbf{z}_2])\Big)^* \Big(J([\mathbf{z}_1]) - J([\mathbf{z}_2])\Big)\Big) \\ &= \sum_{j=1}^{k}\sum_{i=1}^{k} \left|\Big(J([\mathbf{z}_1]) - J([\mathbf{z}_2])\Big)_{i,j}\right|^2\end{aligned}$$

- The squared Euclidean distance between $J([\mathbf{z}_1])$ and $J([\mathbf{z}_2])$
- Also known as the full Procrustes distance

$$\rho_E^2([\mathbf{z}_1],[\mathbf{z}_2]) = 2(1 - |\mathbf{u}_1^*\mathbf{u}_2|^2)\ , \text{ where}$$

$\mathbf{u}_1, \mathbf{u}_2$ are the preshapes of $[\mathbf{z}_1]$ and $[\mathbf{z}_2]$ respectively

Equivariant Embedding

VW embedding is an equivariant embedding w.r.t the special unitary group

$$\mathrm{SU}(k) = \{\mathbf{A} \in \mathrm{GL}(k, \mathbb{C}) \mid \mathbf{A}\mathbf{A}^* = \mathbf{I}, \det(\mathbf{A}) = 1\}$$

Since $\mathbf{A}[\mathbf{z}] = [\mathbf{A}\mathbf{z}]$ holds for any $\mathbf{A} \in \mathrm{SU}(k)$,
We have

$$J(\mathbf{A}[\mathbf{z}]) = \mathbf{A}\mathbf{u}\mathbf{u}^*\mathbf{A}^*$$

Taking a Lie group homomorphism $\phi : \mathrm{SU}(k) \to \mathrm{GL}(k, \mathbb{C})$, such that

$$\phi(\mathbf{A})\mathbf{B} = \mathbf{A}\mathbf{B}\mathbf{A}^*$$

Then $J(\mathbf{A}[\mathbf{z}]) = \mathbf{A}\mathbf{u}\mathbf{u}^*\mathbf{A}^* = \phi(\mathbf{A})\mathbf{u}\mathbf{u}^* = \phi(\mathbf{A})J([\mathbf{z}])$.

(Sample) Extrinsic Mean on Σ_2^k

Given data $\mathcal{D} = \{[\mathbf{z}_i]\}_{i=1}^n \in \Sigma_2^k$

$$\overline{\mathbf{X}}_E = J^{-1}\left(\mathcal{P}\left(\underset{\tilde{\boldsymbol{\mu}} \in E^d}{\operatorname{argmin}} \sum_{i=1}^n \|J([\mathbf{z}_i] - \tilde{\boldsymbol{\mu}}\|^2\right)\right)$$

1. Find $\tilde{\boldsymbol{\mu}} \in \mathrm{E}^d = \mathcal{S}(k, \mathbb{C})$

$$\tilde{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^n J([\mathbf{z}_i]) = \frac{1}{n}\sum_{i=1}^n \mathbf{u}_i\mathbf{u}_i^*$$

2. $\mathcal{P} : \mathcal{S}(k, \mathbb{C}) \to J(\Sigma_2^k)$

$$\mathcal{P}(\tilde{\boldsymbol{\mu}}) = \boldsymbol{\gamma}\boldsymbol{\gamma}^* \text{ , where}$$

$\boldsymbol{\gamma}$: unit eigenvector of $\tilde{\boldsymbol{\mu}}$ corresponding to the largest eigenvalue

3. $\overline{\mathbf{X}}_E = J^{-1}(\boldsymbol{\gamma}\boldsymbol{\gamma}^*) = [\gamma] = \left\{e^{i\theta}\frac{\boldsymbol{\gamma} - \langle\boldsymbol{\gamma}\rangle}{\|\boldsymbol{\gamma} - \langle\boldsymbol{\gamma}\rangle\|}\right\}$

Regression Classifier

Regression Classifier

- $\mathcal{D} : \{\mathbf{x}_i, y_i\}_{i=1}^n \in (\mathbb{R}^d, \mathcal{Y} = \{1, \cdots, C\})$
- Data matrix of the i th class

$$\mathbf{X}_{(i)} = [\mathbf{x}_{(i)}^1, \mathbf{x}_{(i)}^2, \cdots, \mathbf{x}_{(i)}^{n_i}] \in \mathbb{R}^{d \times n_i}$$

- Assumption : Data from a specific class lie on a linear subspace

$$\boldsymbol{x}_{(i)} = \mathbf{X}_{(i)} \boldsymbol{\beta}_{(i)},$$

where $\boldsymbol{\beta}_{(i)} \in \mathbb{R}^{n_i}$ is the class specific regression coefficient vector

- Popular in face recognition (Naseem et al., 2010)
- Drawback : require $d \geqslant n_i$

Ridge Regression Classifier (He et al., 2014)

- For any new test sample $\boldsymbol{x}$, consider a linear regression model

$$\boldsymbol{x} = \mathbf{X}_{(i)}\boldsymbol{\beta}_{(i)} + \boldsymbol{\varepsilon}_{(i)} \ , \ \text{where } i = 1, \cdots, C$$

- Ridge regression fit

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{(i)} &= \underset{\boldsymbol{\beta}_{(i)}}{\operatorname{argmin}} \|\boldsymbol{x} - \mathbf{X}_{(i)}\boldsymbol{\beta}_{(i)}\|_2^2 + \lambda\|\boldsymbol{\beta}_{(i)}\|_2^2 \\ &= \left(\mathbf{X}_{(i)}^\top \mathbf{X}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{X}_{(i)}^\top \boldsymbol{x}\end{aligned}$$

- The projection of $\boldsymbol{x}$ onto the subspace of the i th class

$$\hat{\boldsymbol{x}}_{(i)} = \mathbf{X}_{(i)}\hat{\boldsymbol{\beta}}_{(i)} = \mathbf{X}_{(i)}(\mathbf{X}_{(i)}^\top \mathbf{X}_{(i)} + \lambda \mathbf{I})^{-1}\mathbf{X}_{(i)}^\top \boldsymbol{x}$$

- The estimated class

$$\hat{y} = \underset{i \in \{1, \cdots, C\}}{\operatorname{argmin}} \|\hat{\boldsymbol{x}}_{(i)} - \boldsymbol{x}\|_2^2$$

Ridge Regression Classifier

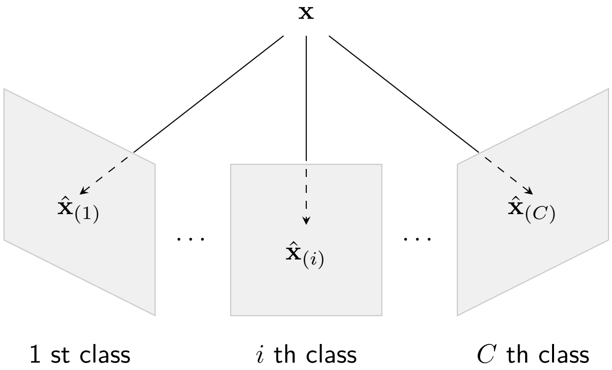


Figure: Graphical interpretation of regression classifiers : $\mathbf{x}$ is the new data point, and gray shaded regions display the subspaces of the corresponding classes, $\hat{\mathbf{x}}_{(i)}$ is the projected value onto the ith subspace.

Naive RRC for Σ_2^k

- The training data $\mathcal{D} = \{[\mathbf{z}_i], y_i\}_{i=1}^{n} \in (\Sigma_2^k, \mathcal{Y})$
- If one treated shapes as a k dimensional complex random vectors, then for the i th class, class specific data matrix

$$\mathbf{U}_{(i)} = \left[\mathbf{u}_{(i)}^1, \mathbf{u}_{(i)}^2, \cdots, \mathbf{u}_{(i)}^{n_i}\right] \in \mathbb{C}^{k \times n_i}$$

- For each class and the new data $\mathbf{u}$, class specific regression model

$$\mathbf{u} = \mathbf{U}_{(i)}\boldsymbol{\beta}_{(i)} + \boldsymbol{\varepsilon}_{(i)} \tag{5}$$

where, $\boldsymbol{\beta}_{(i)} \in \mathbb{C}^{n_i}$ is regression coefficients, and $\boldsymbol{\varepsilon}$ is a $\mathbb{C}^k$ valued random error.

Naive RRC for Σ_2^k (Estimation)

The complex regularized least square problem.

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{(i)} &= \underset{\beta_{(i)}}{\operatorname{argmin}} \|\mathbf{u} - \mathbf{U}_{(i)}\boldsymbol{\beta}_{(i)}\|_2^2 + \lambda\|\boldsymbol{\beta}_{(i)}\|_2^2 \\ &= \left(\mathbf{U}_{(i)}^* \mathbf{U}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{U}_{(i)}^* \mathbf{u}\end{aligned} \tag{6}$$

The projected value by

$$\hat{\mathbf{u}}_{(i)} = \mathbf{U}_{(i)}\hat{\boldsymbol{\beta}}_{(i)} = \mathbf{U}_{(i)}\left(\mathbf{U}_{(i)}^* \mathbf{U}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{U}_{(i)}^* \mathbf{u}$$

Prediction of the class of the given shape

$$\hat{y}_{\mathbf{u}} = \underset{i \in \{1, \cdots, C\}}{\operatorname{argmin}} \|\hat{\mathbf{u}}_{(i)} - \mathbf{u}\|_2^2 \tag{7}$$

Naive RRC for Σ_2^k

Drawbacks :

- The shape space Σ_2^k is a nonlinear manifold
- The geometry of the shape space is ignored

The RRC can not be immediately applicable for shape classification

Kernel Ridge Regression Classifier on Σ_2^k

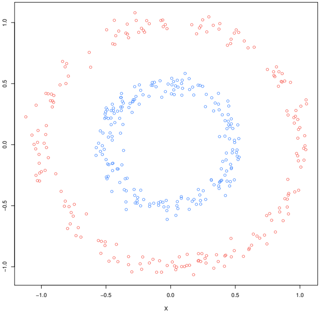
Kernel Methods

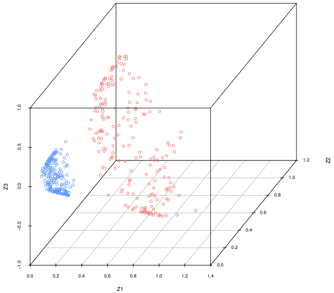
- Map the original samples into a higher dimensional Hilbert space $\mathcal{F}$,

$$\Phi : \mathcal{X} \rightarrow \mathcal{F}$$

- Learning problem is linear in $\mathcal{F}$, not in $\mathcal{X}$

- Toy example : $\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3; (x_1, x_2) \mapsto (x_1^2, x_2^2, \sqrt{2}x_1x_2)$





Kernel Methods (Kernel Trick)

If the learning algorithms only depend on inner products

- The form of nonlinear map $\Phi(\mathbf{x})$ is not necessarily known explicitly
- It could be **determined by a kernel function** $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$

$$\text{s.t. } \kappa(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle_{\mathcal{F}}$$

- The kernel computes inner products in $\mathcal{F}$ directly from the inputs
- Kernel function has to be **positive definite function**

Two benefits :

1. Nonlinear methods via Linear algorithm
2. Arbitrary Space

Kernel Ridge Regression Classifier

The transformed data matrix

$$\mathbf{\Phi}_{(i)} = \begin{pmatrix} | & | & & | \\ \Phi\left(\mathbf{u}_{(i)}^{1}\right) & \Phi\left(\mathbf{u}_{(i)}^{2}\right) & \cdots & \Phi\left(\mathbf{u}_{(i)}^{n_i}\right) \\ | & | & & | \end{pmatrix}_{\dim(\mathcal{F})\times n_i}$$

Transformed class specific regression problem

$$\Phi_{(i)}(\mathbf{u}) = \mathbf{\Phi}_{(i)}\boldsymbol{\beta}_{\Phi,(i)} + \boldsymbol{\varepsilon} \tag{8}$$

Kernel Ridge Regression Classifier

Estimated coefficient

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\Phi,(i)} &= \underset{\boldsymbol{\beta}_{(i)}}{\operatorname{argmin}} \|\Phi(\mathbf{u}) - \boldsymbol{\Phi}_{(i)}\boldsymbol{\beta}_{(i)}\|_2^2 + \lambda\|\boldsymbol{\beta}_{(i)}\|_2^2 \\ &= \left(\boldsymbol{\Phi}_{(i)}^\top \boldsymbol{\Phi}_{(i)} + \lambda \mathbf{I}\right)^{-1} \boldsymbol{\Phi}_{(i)}^\top \Phi(\mathbf{u}) \end{aligned} \tag{9}$$

The projected value of $\mathbf{u}$ onto the subspace of the i th class

$$\begin{aligned}\hat{\Phi}_{(i)}(\mathbf{u}) &= \boldsymbol{\Phi}_{(i)}\hat{\boldsymbol{\beta}}_{\Phi,(i)} \\ &= \boldsymbol{\Phi}_{(i)} \left(\boldsymbol{\Phi}_{(i)}^\top \boldsymbol{\Phi}_{(i)} + \lambda \mathbf{I}\right)^{-1} \boldsymbol{\Phi}_{(i)}^\top \Phi(\mathbf{u}) \end{aligned} \tag{10}$$

The estimated class

$$\begin{aligned}\hat{y}_{\mathbf{u}} &= \underset{i=\{1,\cdots,C\}}{\operatorname{argmin}} \|\hat{\Phi}_{(i)}(\mathbf{u}) - \Phi(\mathbf{u})\|_2^2 \\ &= \underset{i=\{1,\cdots,C\}}{\operatorname{argmin}} \hat{\Phi}_{(i)}(\mathbf{u})^\top \hat{\Phi}_{(i)}(\mathbf{u}) - 2\hat{\Phi}_{(i)}(\mathbf{u})^\top \Phi(\mathbf{u}) + \Phi^\top(\mathbf{u})\Phi(\mathbf{u}) \end{aligned} \tag{11}$$

Kernel Ridge Regression Classifier

Define some notations

$$\mathbf{\Phi}_{(i)}^{\top}\Phi(\mathbf{u}) = \left(\langle\Phi(\mathbf{u}_{(i)}^{1}), \Phi(\mathbf{u})\rangle, \cdots, \langle\Phi(\mathbf{u}_{(i)}^{n_i}), \Phi(\mathbf{u})\rangle\right)^{\top}$$

$$= \left(\kappa\left(\Phi(\mathbf{u}_{(i)}^{1}), \Phi(\mathbf{u})\right), \cdots, \kappa\left(\Phi(\mathbf{u}_{(i)}^{n_i}), \Phi(\mathbf{u})\right)\right)^{\top} = \mathbf{k}_{(i)}$$

$$\mathbf{\Phi}_{(i)}^{\top}\mathbf{\Phi}_{(i)} = \begin{pmatrix} \langle\Phi(\mathbf{u}_{(i)}^{1}), \Phi(\mathbf{u}_{(i)}^{1})\rangle & \cdots & \langle\Phi(\mathbf{u}_{(i)}^{1}), \Phi(\mathbf{u}_{(i)}^{n_i})\rangle \\ \vdots & \vdots & \vdots \\ \langle\Phi(\mathbf{u}_{(i)}^{n_i}), \Phi(\mathbf{u}_{(i)}^{1})\rangle & \cdots & \langle\Phi(\mathbf{u}_{(i)}^{n_i}), \Phi(\mathbf{u}_{(i)}^{n_i})\rangle \end{pmatrix}$$

$$= \begin{pmatrix} \kappa\left(\mathbf{u}_{(i)}^{1}, \mathbf{u}_{(i)}^{1}\right) & \cdots & \kappa\left(\mathbf{u}_{(i)}^{n_i}, \mathbf{u}_{(i)}^{1}\right) \\ \vdots & \vdots & \vdots \\ \kappa\left(\mathbf{u}_{(i)}^{n_i}, \mathbf{u}_{(i)}^{1}\right) & \cdots & \kappa\left(\mathbf{u}_{(i)}^{n_i}, \mathbf{u}_{(i)}^{n_i}\right) \end{pmatrix} = \mathbf{K}_{(i)}$$

Estimation

Then (9), (10) can be written as

$$\begin{aligned}\hat{\beta}_{\Phi,(i)} &= \left(\boldsymbol{\Phi}_{(i)}^\top \boldsymbol{\Phi}_{(i)} + \lambda \mathbf{I}\right)^{-1} \boldsymbol{\Phi}_{(i)}^\top \Phi(\mathbf{u}) \\ &= \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{k}_{(i)}\end{aligned}$$

$$\begin{aligned}\hat{\Phi}_{(i)}(\mathbf{u}) &= \boldsymbol{\Phi}_{(i)} \left(\boldsymbol{\Phi}_{(i)}^\top \boldsymbol{\Phi}_{(i)} + \lambda \mathbf{I}\right)^{-1} \boldsymbol{\Phi}_{(i)}^\top \Phi(\mathbf{u}) \\ &= \boldsymbol{\Phi}_{(i)} \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{k}_{(i)}\end{aligned}$$

Estimation

By using

$$\hat{\Phi}_{(i)}(\mathbf{u})^\top \hat{\Phi}_{(i)}(\mathbf{u}) = \mathbf{k}_{(i)}(\mathbf{u})^\top \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{K}_{(i)} \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{k}_{(i)}(\mathbf{u})$$

$$\hat{\Phi}_{(i)}(\mathbf{u})^\top \Phi(\mathbf{u}) = \mathbf{k}_{(i)}(\mathbf{u})^\top \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{k}_{(i)}(\mathbf{u})$$

Class estimation step in (11) also can be written as

$$\hat{y}_{\mathbf{u}} = \underset{i=\{1,\cdots,C\}}{\operatorname{argmin}} \mathbf{k}_{(i)}(\mathbf{u})^\top \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{K}_{(i)} \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{k}_{(i)}(\mathbf{u})$$
$$- 2\mathbf{k}_{(i)}(\mathbf{u})^\top \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right) \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \mathbf{k}_{(i)}(\mathbf{u})$$

$$= \underset{i\{1,\cdots,C\}}{\operatorname{argmin}} \mathbf{k}_{(i)}(\mathbf{u})^\top \left(\mathbf{K}_{(i)} + \lambda \mathbf{I}\right)^{-1} \left(-\mathbf{K}_{(i)} - 2\lambda \mathbf{I}\right) (\mathbf{K}_{(i)} + \lambda \mathbf{I})^{-1} \mathbf{k}_{(i)}(\mathbf{u})$$

- **Doesn't depend on Φ**

Choice of Kernels

- Gaussian RBF kernel on $\mathbb{R}^d$

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{\sigma}\right),$$

where $\sigma > 0$ is a tuning parameter

Q. How can we choose a kernel on Σ_2^k ?

- Gaussian RBF kernel with the Riemannian geodesic distance on Σ_2^k

$$\kappa([\mathbf{z}_i], [\mathbf{z}_j]) = \exp\left(-\frac{\arccos(|\langle[\mathbf{z}_1], [\mathbf{z}_2]\rangle|)^2}{\sigma}\right)$$

- Not Positive definite kernel

Extrinsic Veronese Whitney Gaussian kernel

Extrinsic Veronese Whitney Gaussian kernel

$$\kappa([\mathbf{z}_1],[\mathbf{z}_2]) = \exp\left(-\frac{\rho_E^2\left(J([\mathbf{z}_1]),J([\mathbf{z}_2])\right)}{\sigma^2}\right),$$

where ρ_E^2 denotes the Squared extrinsic distance

$$\rho_E^2(\mathbf{z}_1,\mathbf{z}_2) = \mathrm{Trace}\left(\Big(J([\mathbf{z}_1]) - J([\mathbf{z}_2])\Big)^*\Big(J([\mathbf{z}_1]) - J([\mathbf{z}_2])\Big)\right)$$

Q. Positive Definite ?

Positive definiteness of the Extrinsic VW Gaussian kernel

Proposition Berg et al. (1984)

The kernels having the form of $\exp(-tf(\mathbf{x}_i, \mathbf{x}_j))$ is positive definite for all $t > 0$, if and only if f is (conditionally) negative definite function

(Conditionally) Negative definite function : Let $\mathcal{X}$ be a nonempty set. A function $f : (\mathcal{X} \times \mathcal{X}) \rightarrow \mathbb{R}$ is negative definite if and only if f is symmetric, and

$$\sum_{i,j=1}^{n} \alpha_i \alpha_j f(\mathbf{x}_i, \mathbf{x}_j) \leqslant 0$$

for $\forall n \in \mathbb{N}, \{\mathbf{x}_1, \cdots, \mathbf{x}_n\} \subseteq \mathcal{X}$, and $\{\alpha_1, \cdots, \alpha_n\} \subseteq \mathbb{R}$ with $\sum_{i=1}^{n} \alpha_i = 0$

Note : See also Schoenberg (1938)

Positive definiteness of the Extrinsic VW Gaussian kernel

Theorem

The squared extrinsic Euclidean distance function $\rho_E^2 : (\Sigma_2^k \times \Sigma_2^k) \to \mathbb{R}$, induced by the Veronese Whitney ebedding $J : \Sigma_2^k \to S(k, \mathbb{C})$

$$\begin{aligned} \rho_E^2([\mathbf{z}_i], [\mathbf{z}_j]) :&= \|J[\mathbf{z}_i] - J[\mathbf{z}_j]\|_F^2 \\ &= \mathrm{Trace}\left\{\Big(J[\mathbf{z}_i] - J[\mathbf{z}_j]\Big)\Big(J[\mathbf{z}_i] - J[\mathbf{z}_j]\Big)^*\right\} \end{aligned}$$

is **negative definite.**

Claim : The Extrinsic VW Gaussian kernel is positive definite

Proof

$$
\begin{aligned}
\sum_{i,j=1}^{n} \alpha_i \alpha_j \rho_E^2(J([\mathbf{z}_i]), J([\mathbf{z}_j]))& \\
&= \sum_{i,j=1}^{n} \alpha_i \alpha_j \| J([\mathbf{z}_i]) - J([\mathbf{z}_j]) \|_F^2 \\
&= \sum_{i,j=1}^{n} \alpha_i \alpha_j \langle J([\mathbf{z}_i]) - J([\mathbf{z}_j]), J([\mathbf{z}_i]) - J([\mathbf{z}_j]) \rangle_F \\
&= \sum_{j=1}^{n} \alpha_j \sum_{i=1}^{n} \alpha_i \langle J([\mathbf{z}_i]), J([\mathbf{z}_i]) \rangle - 2 \sum_{i,j=1}^{n} \alpha_i \alpha_j \langle J([\mathbf{z}_i]), J([\mathbf{z}_j]) \rangle_F \\
&\quad + \sum_{i=1}^{n} \alpha_i \sum_{j=1}^{n} \alpha_i \langle J([\mathbf{z}_j]), J([\mathbf{z}_j]) \rangle \\
&= -2 \sum_{i,j=1}^{n} \alpha_i \alpha_j \langle J([\mathbf{z}_i]), J([\mathbf{z}_j]) \rangle_F = -2 \| \sum_{i=1}^{n} \alpha_i J([\mathbf{z}_i]) \|_F^2 \\
&\leqslant 0
\end{aligned}
$$

Extension : Symmetric Positive Definite Matrices

Diffusion tensor imaging (DTI)

- Embedding (Lin et al., 2019):

$$\log : \mathsf{SPD}(3) \to \mathsf{Sym}(3);$$
$$\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{-1} \in \mathsf{SPD}(3) \mapsto \mathsf{log}(\mathbf{A}) = \mathbf{U}\log(\boldsymbol{\Lambda})\mathbf{U}^{-1}$$

- Extrinsic distance :

$$\rho_E(\mathbf{A}_1, \mathbf{A}_2) = \|\log(\mathbf{A}_1) - \log(\mathbf{A}_2)\|_F$$

- Gaussian Kernel (Jayasumana et al., 2013) :

$$\kappa(\mathbf{A}_1, \mathbf{A}_2) = \exp\left(-\frac{\rho_E^2(\mathbf{A}_1, \mathbf{A}_2)}{\sigma^2}\right)$$

is positive definite

Real Data Analysis

Real Data Anlysis

- `PassifloraLeaves` data (Chitwood and Otoni, 2016, 2017)
  `https://github.com/DanChitwood/PassifloraLeaves`

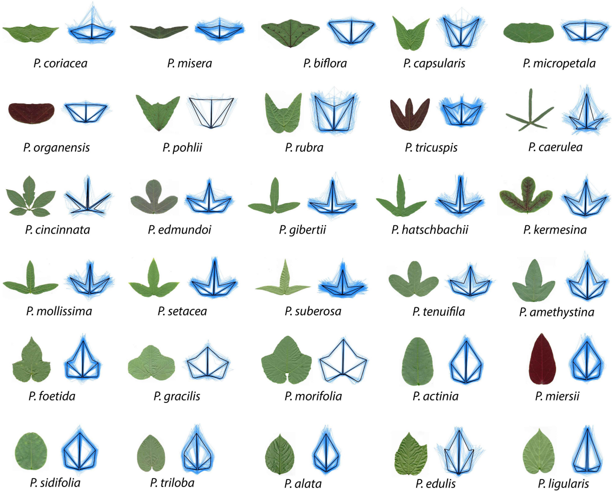


Figure: Parts of `PassifloraLeaves` data (Chitwood and Otoni, 2017)

Real Data Anlysis

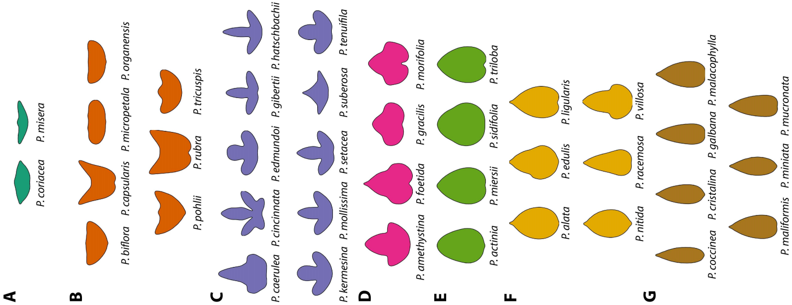


Figure: The categorization of `PassifloraLeaves` data (Chitwood and Otoni, 2017)

Real Data Anlysis

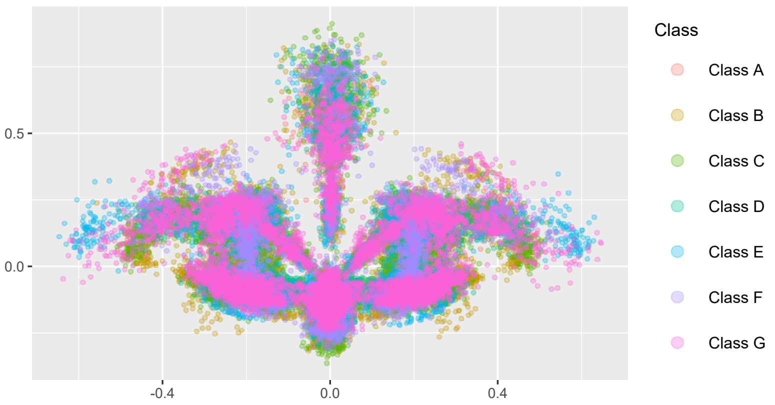


Figure: The landmarks of leaf data

Real data anlysis

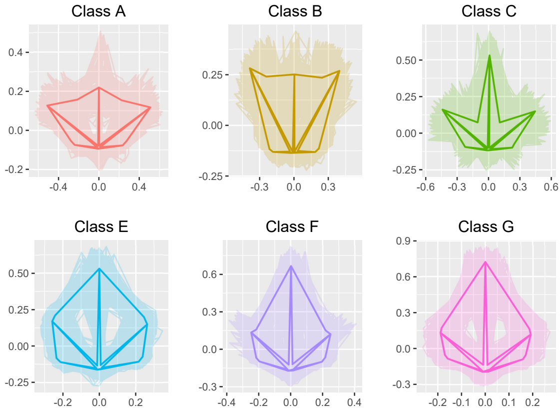


Figure: The shapes of leaves with different classes : The set of transparenct lines represents observations for each class, and the bold solid lines are their VW means.

Scheme

For each run of simulations

- Training set (60 % of each class)
- Test set (40 % of each class)

Subspace size

- Drawing subsamples of equal sizes from the training set
- $n_A = n_B = \cdots = n_G$
- Small subspace size 10, and large 100 are considered

Evaluation Metrics

- Averaged Precision

$$\text{prec} = \sum_{i=1}^{G} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i} / G$$

- Averaged Recall

$$\text{rec} = \sum_{i=1}^{G} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i} / G$$

- F_1 scores

$$F_1 = 2 \times \frac{\text{prec} \cdot \text{rec}}{\text{prec} + \text{rec}}$$

- Averaged Accuracy

$$\text{Avg. Accuracy} = \sum_{i=1}^{G} \frac{\text{TP}_i + \text{TN}_i}{\text{TP}_i + \text{FN}_i + \text{FP}_i + \text{TN}_i} / G$$

Note : Confusion Matrix

Predicted Value

Actual Value		$\widehat{\mathbf{P}}$	$\widehat{\mathbf{N}}$
	P	True Positive **TP**	False Negative **FN**
	N	False Positive **FP**	True Negative **TN**

Competing Methods

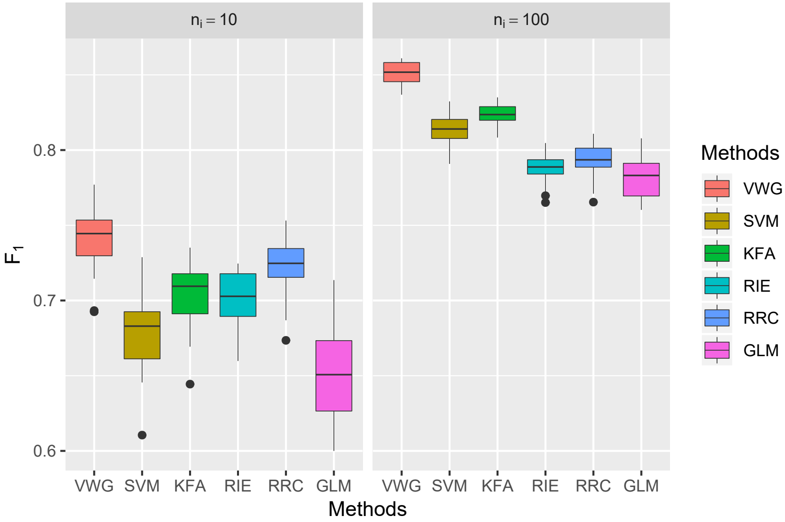
1. KRRC with Extrinsic VW Gaussian Kernel (VWG)
2. Support Vector Machine with Gaussian kernel (SVM)
3. Kernel Fisher Discriminant Analysis (KFA)
4. KRRC with intrinsic Gaussian Kernel (RIE)
5. Naive RRC (RRC)
6. Multiclass GLM with the ridge penalty (GLM)

Results

	$n_i = 10$			
	Precision	Recall	F_1	Accuracy
VWG	**0.7450**	**0.7490**	**0.7389**	**0.9297**
SVM	0.6940	0.6843	0.6779	0.9110
KFA	0.7102	0.7185	0.7092	0.9196
RIE	0.7098	0.7139	0.7006	0.9195
RRC	0.7403	0.7282	0.7215	0.9246
GLM	0.6589	0.6675	0.6515	0.9091
	$n_i = 100$			
	Precision	Recall	F_1	Accuracy
Extrinsic VWG	**0.8509**	**0.8597**	**0.8506**	**0.9609**
Extrinsic CG	0.8145	0.8242	0.8137	0.9500
Full PCG	0.8259	0.8345	0.8234	0.9527
Riemannian	0.7875	0.7866	0.7867	0.9411
RRC	0.8074	0.7968	0.7935	0.9411
GLM	0.7769	0.7930	0.7820	0.9419

Table: Results of the classification. The best result for each category is in bold

Results



$n_i = 10$
$n_i = 100$
0.8
0.7
0.6
F_1
VWG
SVM
KFA
RIE
RRC
GLM
Methods
Methods
VWG
SVM
KFA
RIE
RRC
GLM

Results

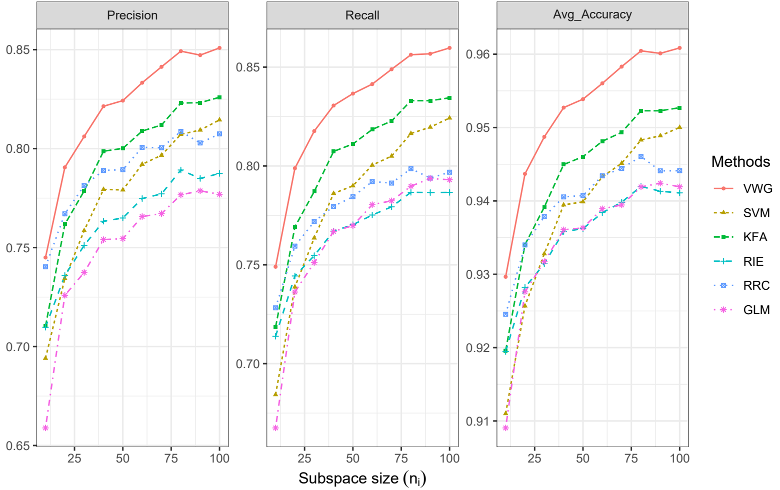


Figure: The performance of KRRCs with three different positive definite kernels. The extrinsic KRRC with VW Gaussian appears to perform better than the others

Conclusion

- Riemannian kernel has a poor performance
- Extrinsic kernel has the best performance
- Extrinsic kernel approach can be applied to other manifolds

- **Note** : The R package for shape KRRC can be available at https://github.com/hwiyoungstat/ShapeKRRC

References I

Berg, C., Christensen, J. P. R., and Ressel, P. (1984). *Harmonic Analysis on Semigroups*. Springer.

Bhattacharya, A. and Bhattacharya, R. (2012). *Nonparametric Inference on Manifolds : With Applications to Shape Spaces. IMS Monograph #2*. Cambridge University Press.

Bhattacharya, R. N. and Patrangenaru, V. (2003). Large sample theory of intrinsic and extrinsic sample means on manifolds-part i. *Annals of Statistics*, 31:1–29.

Bhattacharya, R. N. and Patrangenaru, V. (2005). Large sample theory of intrinsic and extrinsic sample means on manifolds- part ii. *Annals of Statistics*, 33:1211–1245.

Chitwood, D. H. and Otoni, W. C. (2016). Morphometric analysis of passiflora leaves: the relationship between landmarks of the vasculature and elliptical fourier descriptors of the blade. *GigaScience*, 6:1–13.

References II

Chitwood, D. H. and Otoni, W. C. (2017). Divergent leaf shapes among passiflora species arise from a shared juvenile morphology. *Plant Direct*, 1:1–15.

Fréchet, M. (1948). Les élements aléatoires de nature quelconque dans un espace distancié. *Ann. Inst. H. Poincaré*, 10:215–310.

He, J., Ding, L., Jiang, L., and Ma, L. (2014). Kernel ridge regression classification. *International Joint Conference on Neural Networks*, pages 2263–2267.

Jayasumana, S., Hartley, R., Salzmann, M., Li, H., and Harandi, M. (2013). Kernel methods on the riemannian manifold of symmetric positive definite matrices. *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 73–80.

Kent, J. T. (1992). *New directions in shape analysis. In: The Art of Statistical Science*. John Wiley & Sons, Ltd, Chichester.

References III

Lin, L., Mu, N., Cheung, P., and Dunson, D. (2019). Extrinsic gaussian processes for regression and classification on manifolds. *Bayesian Anal.*, 14(3):887–906.

Naseem, I., Togneri, R., and Bennamoun, M. (2010). Linear regression for face recognition. *IEEE transactions on pattern analysis and machine Intelligence*, 32:2106–2112.

Schoenberg, I. J. (1938). Metric spaces and positive definite functions. *Transactions of the American Mathematical Society*, 44(3):522–536.