FROM APPLICATION TO THEORY: SECTION III

PRINCIPAL COMPONENT ANALYSIS

PCA DIMENSION REDUCTION

Input (high dimensional)

$x_1, x_2, \ldots.x_n$ points in R^p

Output (low dimensional)

$y_1, y_2, \ldots.y_n$ points in R^q $(q<<P)$

Projections

1. Assume inputs are centered: $\sum_i^n x_i = 0$ (x_i *is a vector)*
2. Given a unit vector u and a point x, the projection of x onto u is given by $X^T u$
3. Maximize projected variance:

$$\mathrm{var}(y) = \frac{1}{n}\sum_i (x_i^T u)^2 = \frac{1}{n}\sum_i u^T x_i x_i^T u$$

$$= u^T (\frac{1}{n}\sum_i x_i x_i^T) u = u^T C u$$

1. Assume inputs are centered: $\sum_i x_i = 0$
2. Given a unit vector u and a point x, the length of the projection of x onto u is given *by* $\boldsymbol{X}^T u$
3. Maximize projected variance: $$\mathrm{var}(y) = \frac{1}{n}\sum_i (x_i^T u)^2 = \frac{1}{n}\sum_i u^T x_i x_i^T u$$ $$= u^T (\frac{1}{n}\sum_i x_i x_i^T) u = u^T C u$$
4. Minimize the sum of squared distances between all (x_i , y_i)
5. If to a 1D subspace
   - Maximizing $u^T C u$ subject to $\|u\| = 1$, where we have $C = \frac{1}{n}\sum_i x_i x_i^T$ C is the empirical covariance matrix of the data, this gives the principal eigenvector of C

6. If to a q-dimensional subspace
   - We need a collection of $u_1, \ldots u_q$ that are top q eigenvectors of C.
   - $u_1, \ldots u_q$ now form a new, orthogonal basis for the data.
   - We have a low-dimensional representation of **X**, by

$$y_i = \begin{bmatrix} u_1^T x_i \\ u_2^T x_i \\ \ldots \\ u_q^T x_i \end{bmatrix} \in \Re^q$$

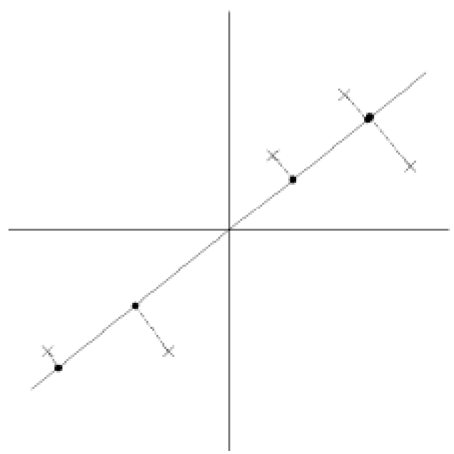
PRACTICAL PART

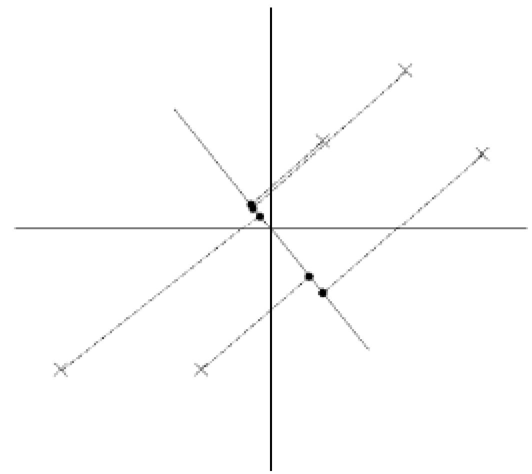
PCA in R

Two R-implementations

- prcomp() and princomp()
- prcomp is numerically more stable
- princomp has more options

GOOD OR POOR



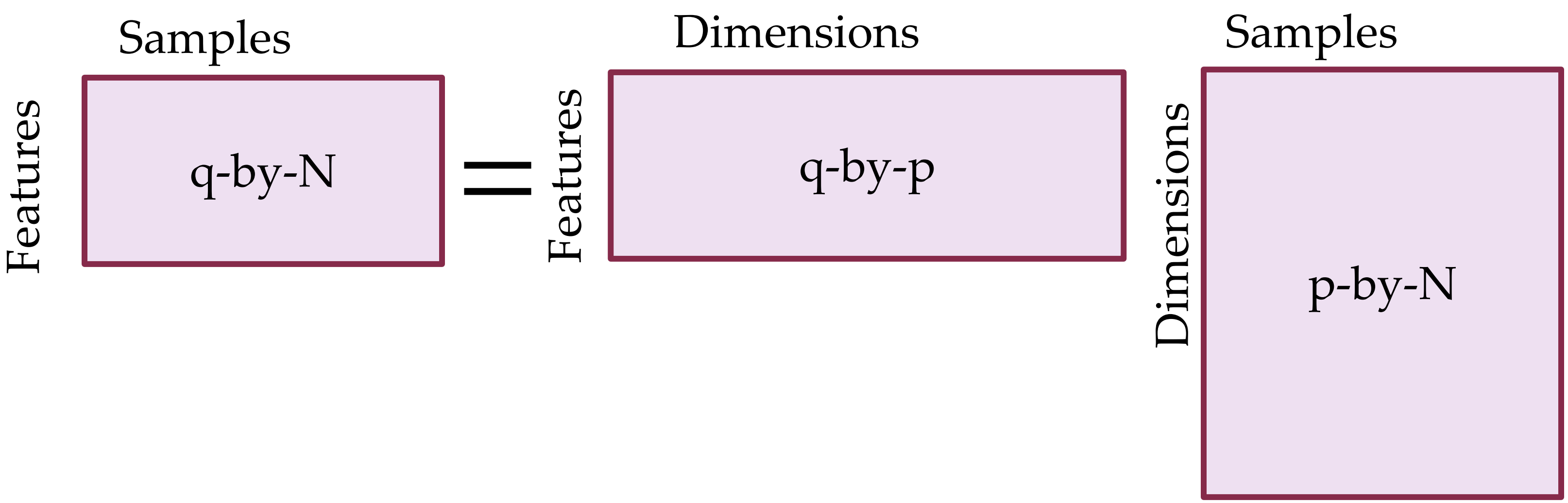


+ is my original data points
. is the projected data.

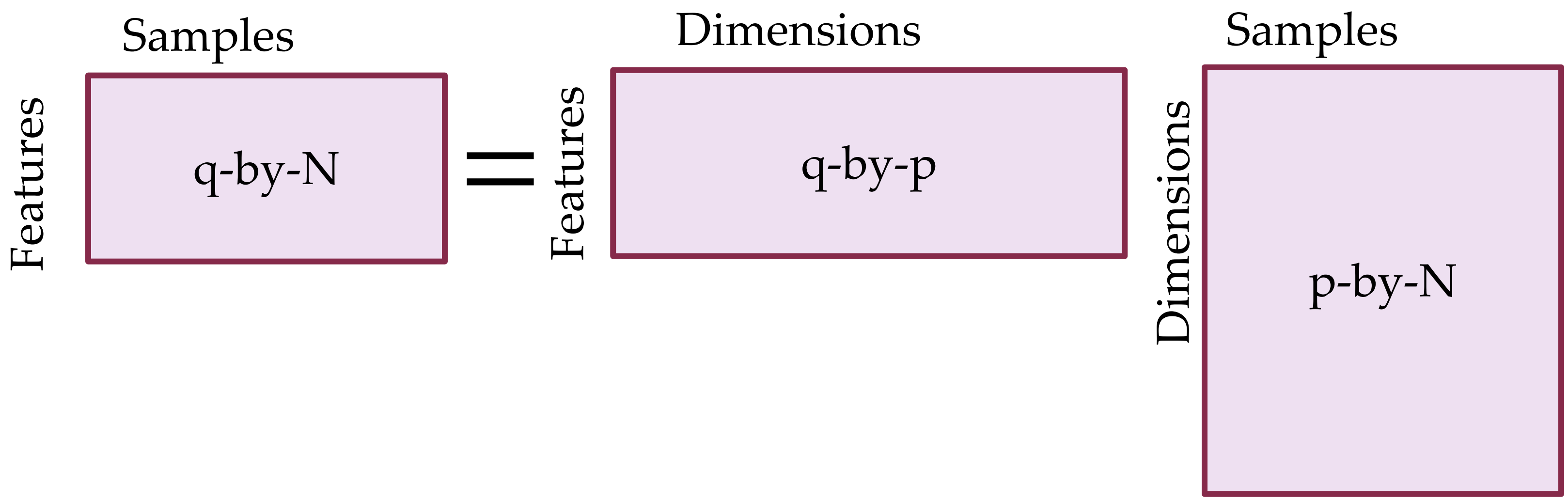
- Seek most accurate data representation in a lower dimensional space
- Good direction/subspace to use for projection lies in the direction of largest variance.

PCA FEATURE EXTRACTION

$$\mathbf{Z} = \mathbf{W}\mathbf{X}$$



Weight Matrix **Feature Matrix** **Input Matrix**



$$\mathbf{Z} = \mathbf{W}\mathbf{X} =$$

$$= \begin{bmatrix} \mathbf{z}_1^T \\ \mathbf{z}_2^T \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \end{bmatrix} \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \end{bmatrix}$$

THE LIST OF GOALS

To find desirable features

- With simple weights
  - Minimize relation of the dimensions
  - De-correlate the transformation: $\mathbf{z}_1^T \mathbf{z}_2 = 0$
- Avoid similarity in the features
  - Minimize relation of the features
  - De-correlate the features: $\mathbf{w}_1^T \mathbf{w}_2 = 0$

HOW TO SOLVE?

$$\mathbf{Z} = \mathbf{W}\mathbf{X} =$$

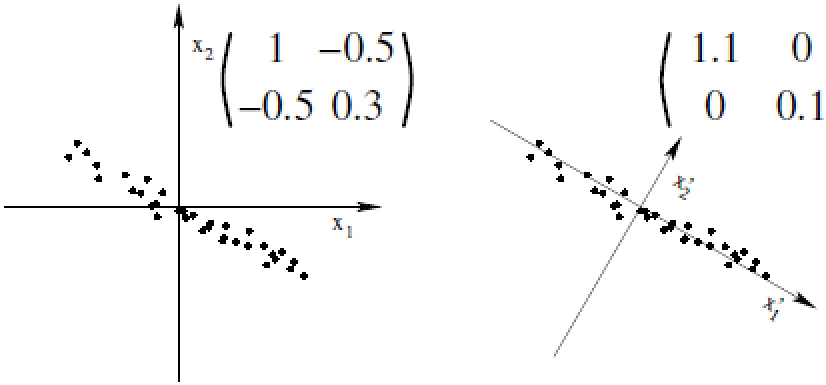
$$= \begin{bmatrix} \mathbf{z}_1^T \\ \mathbf{z}_2^T \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \end{bmatrix} \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \end{bmatrix}$$

- For a given input data X
- We need to find a feature matrix W
- So that the weights de-correlate

$$\boxed{(\mathbf{W}\mathbf{X})(\mathbf{W}\mathbf{X})^T = N\mathbf{I}} \Rightarrow \mathbf{Z}\mathbf{Z}^T = N\mathbf{I}$$

How to solve this?

DIAGONALIZING THE COVARIANCE MATRIX



Diagonalizing the covariance suppresses cross-dimensional co-activity

$$\mathrm{Cov}\left(\mathbf{z}_1, \mathbf{z}_2\right) = \left[\begin{array}{cc} \mathbf{z}_1^T\mathbf{z}_1 & \mathbf{z}_1^T\mathbf{z}_2 \\ \mathbf{z}_2^T\mathbf{z}_1 & \mathbf{z}_2^T\mathbf{z}_2 \end{array}\right] / N = \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array}\right] = \mathbf{I}$$

COVARIANCE MATRIX

Covariance matrix:

$$C_{ij} = < x_i x_j >$$

$$C_x = < \mathbf{x}\mathbf{x}^T > = \frac{1}{n} \sum_i x_i x_i^T$$

$$C_x^T = C_x$$

Eigenvalue equation:

$$C_x \mathbf{u}_i = \mathbf{u}_i \lambda_i$$

- Eigenvalues are ordered $\lambda_i \geq \lambda_i + 1$
- Eigenvectors are <u>orthonormal</u> $\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}$

$$U := (\mathbf{u}_1, \ldots, \mathbf{u}_P)$$

$$\Lambda := diag(\lambda_1, \ldots, \lambda_P)$$

$$\boxed{C_x \mathbf{u}_i = \mathbf{u}_i \lambda_i} \tag{1}$$

$$\lambda_i \geq \lambda_i + 1 \tag{2}$$

(4) & (5)

$$\boxed{\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}} \tag{3}$$

(4)

$$U := (\mathbf{u}_1, \ldots, \mathbf{u}_P) \tag{4}$$

$$\Lambda := diag(\lambda_1, \ldots, \lambda_P) \tag{5}$$

$$\boxed{U^T U = I \Leftrightarrow U U^T = I} \tag{6}$$

$$\boxed{C_x U = U \Lambda} \tag{7}$$

$$U^T C_x U = \Lambda$$

$$C_x = U \Lambda U^T$$

$$< x^T x > = \sum_i \lambda_i$$

Hint: the total variance of some multi-dimensional data equals the trace of its covariance matrix

Proof: HW2. Q2

DIAGONALIZING THE COVARIANCE MATRIX

Define $x' := U^T x$ with new covariance matrix C'_x

Then we have: $x' := U^T x$

$$C'_x := \langle x' x'^T \rangle = \langle (U^T x)(U^T x)^T \rangle$$

$$= \boxed{U^T \langle x x^T \rangle U = U^T C_x U = \Lambda}$$

Solution:

$$\boxed{\begin{aligned} &\mathbf{W}\,\mathrm{Cov}(\mathbf{X})\,\mathbf{W}^T = \mathbf{I} \Rightarrow \\ &\Rightarrow \mathbf{W} = \begin{bmatrix} \sqrt{\lambda_1} & 0 \\ 0 & \sqrt{\lambda_2} \end{bmatrix}^{-1} \mathbf{U}^T \end{aligned}}$$

$$(\mathbf{W}\mathbf{X})(\mathbf{W}\mathbf{X})^T = N\mathbf{I} \Rightarrow$$

$$\Rightarrow \mathbf{W}\mathbf{X}\mathbf{X}^T\mathbf{W}^T = N\mathbf{I} \Rightarrow$$

$$\Rightarrow \boxed{\mathbf{W}\,\mathrm{Cov}(\mathbf{X})\,\mathbf{W}^T = \mathbf{I}}$$

WHITENING OR SPHERING

- Transform a data set to have variance one in all directions
- Stretch and compress the data distribution along the axes of the principal components
- Technically, rotates the data into a coordinate system where the covariance matrix is diagonal → performs the stretching along the axes → rotates the data back into the original coordinate system

PCA

- **Sphering is achieved by multiply the data with a sphering matrix.**

Sphering is achieved by multiply the data with a sphering matrix.

$$Z := U diag(\frac{1}{\sqrt{\lambda_1}}, \frac{1}{\sqrt{\lambda_2}}, ..., \frac{1}{\sqrt{\lambda_P}}) U^T$$

$$\hat{x} := Zx$$

$$C_{\hat{x}} := < \hat{x}\hat{x}^T > = < (Zx)(Zx)^T > = Z < xx^T > Z^T$$

$$= U diag(\frac{1}{\sqrt{\lambda_i}}) U^T C_x U diag(\frac{1}{\sqrt{\lambda_i}}) U^T$$

$$= U diag(\frac{1}{\sqrt{\lambda_i}}) \Lambda diag(\frac{1}{\sqrt{\lambda_i}}) U^T$$

$$= U 1 U^T = 1$$

$$< (n^T \hat{x})^2 > = n < \hat{x}\hat{x}^T > n = n^T n = 1$$

SINGULAR VALUE DECOMPOSITION

When more dimensions and fewer observations

x^i, $i = 1,\ldots,M$ be I-dimensional, (M<I) $\quad \mathbf{X} := (x^1, \ldots, x^M)$

The second-moment matrix: $\quad \mathbf{C}_1 := \mathbf{X}\mathbf{X}^T / M$

Eigenvalue Decomposition: $\quad \mathbf{C}_1\mathbf{U}_1 = \mathbf{U}_1\mathbf{\Lambda}_1 \Leftrightarrow \mathbf{C}_1 = \mathbf{U}_1\mathbf{\Lambda}_1\mathbf{U}_1^T$

Transposed data in eigenvectors: $\quad \mathbf{Y}_1 := \mathbf{U}_1^T\mathbf{X}$

Old way

SINGULAR VALUE DECOMPOSITION

When more dimensions and fewer observations

Now consider $\mathbf{X}^{\mathrm{T}}$ $\mathbf{X}^T := (x^1, \ldots, x^I)$

The second-moment matrix: $\mathbf{C}_2 := \mathbf{X}^T\mathbf{X} / I$

Eigenvalue Decomposition: $\mathbf{C}_2\mathbf{U}_2 = \mathbf{U}_2\mathbf{\Lambda}_2 \Leftrightarrow \mathbf{C}_2 = \mathbf{U}_2\mathbf{\Lambda}_2\mathbf{U}_2^T$

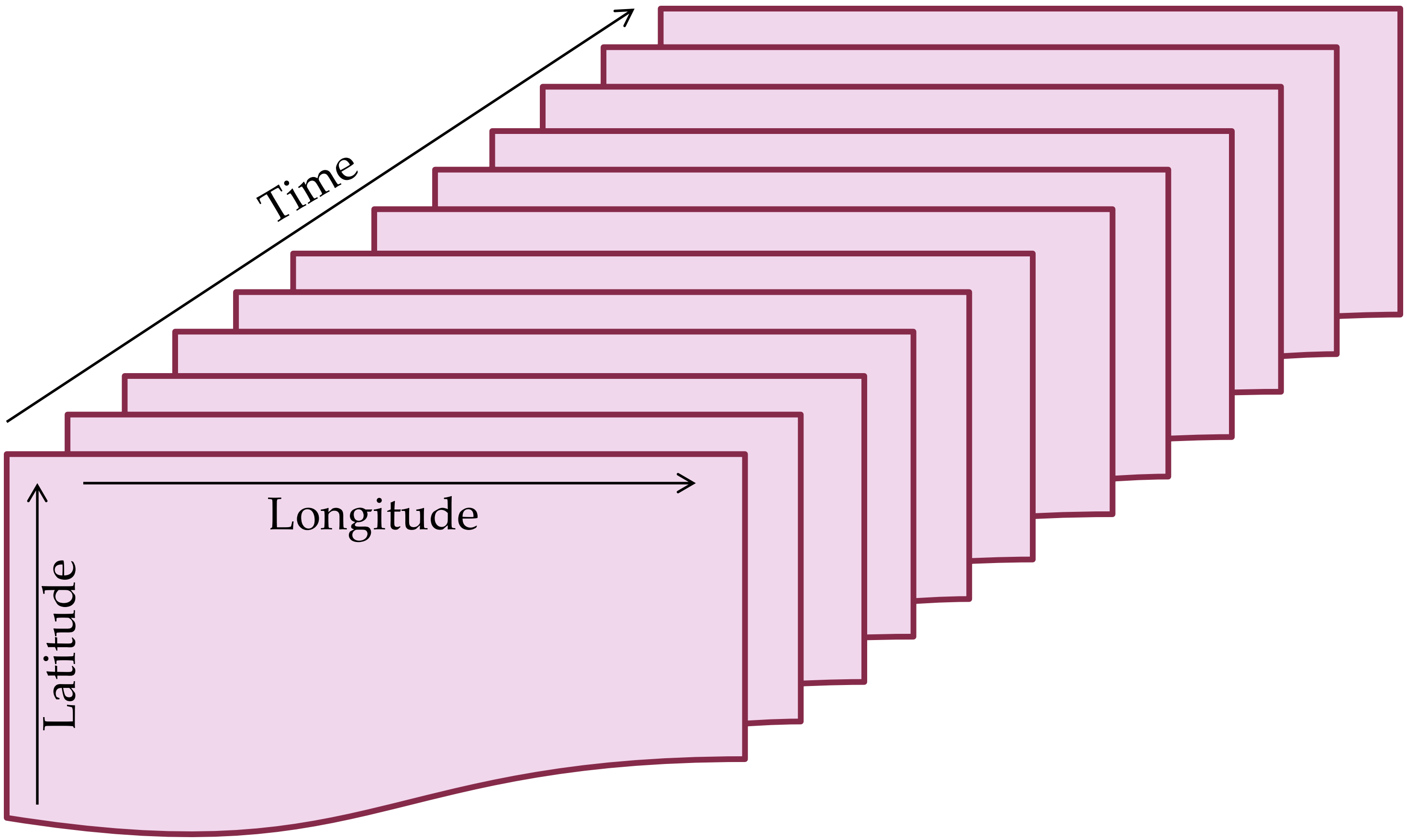
Transposed data in eigenvectors: $\mathbf{Y}_2 := \mathbf{U}_2^T\mathbf{X}^T$

New way

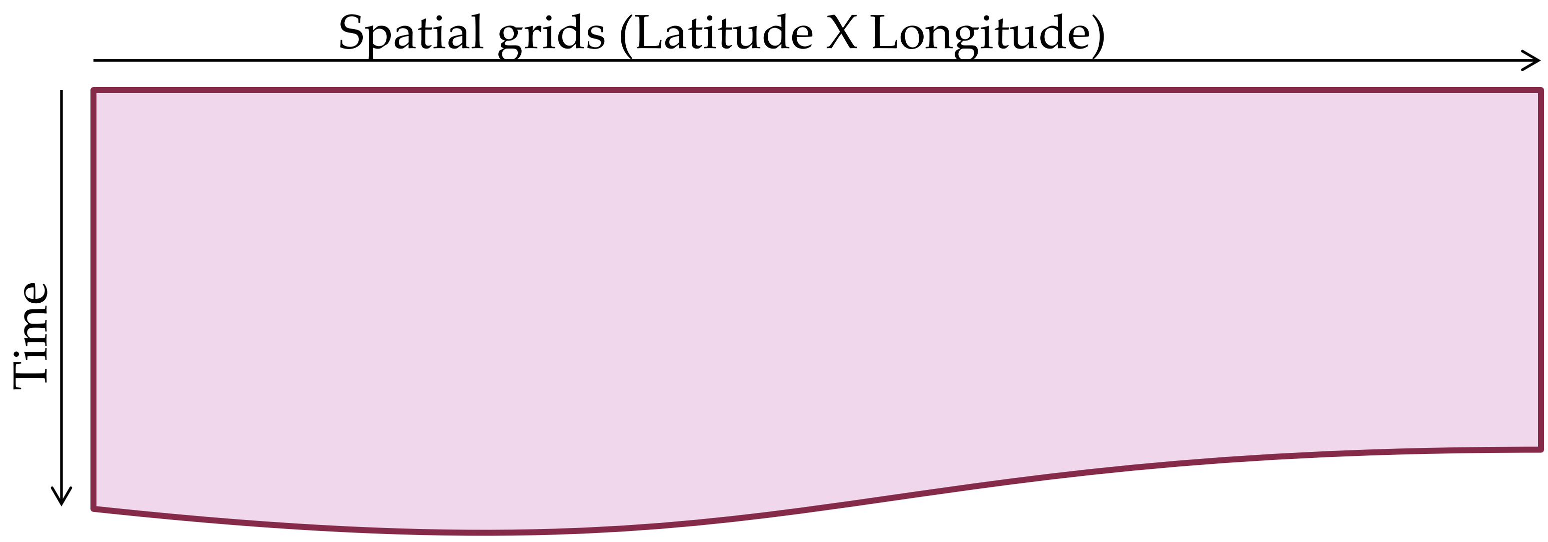
Property: Rows of $\mathbf{Y}_2$ are eigenvectors of $\mathbf{C}_1$. The corresponding eigenvalues of $\mathbf{C}_2$ scaled by I/M.

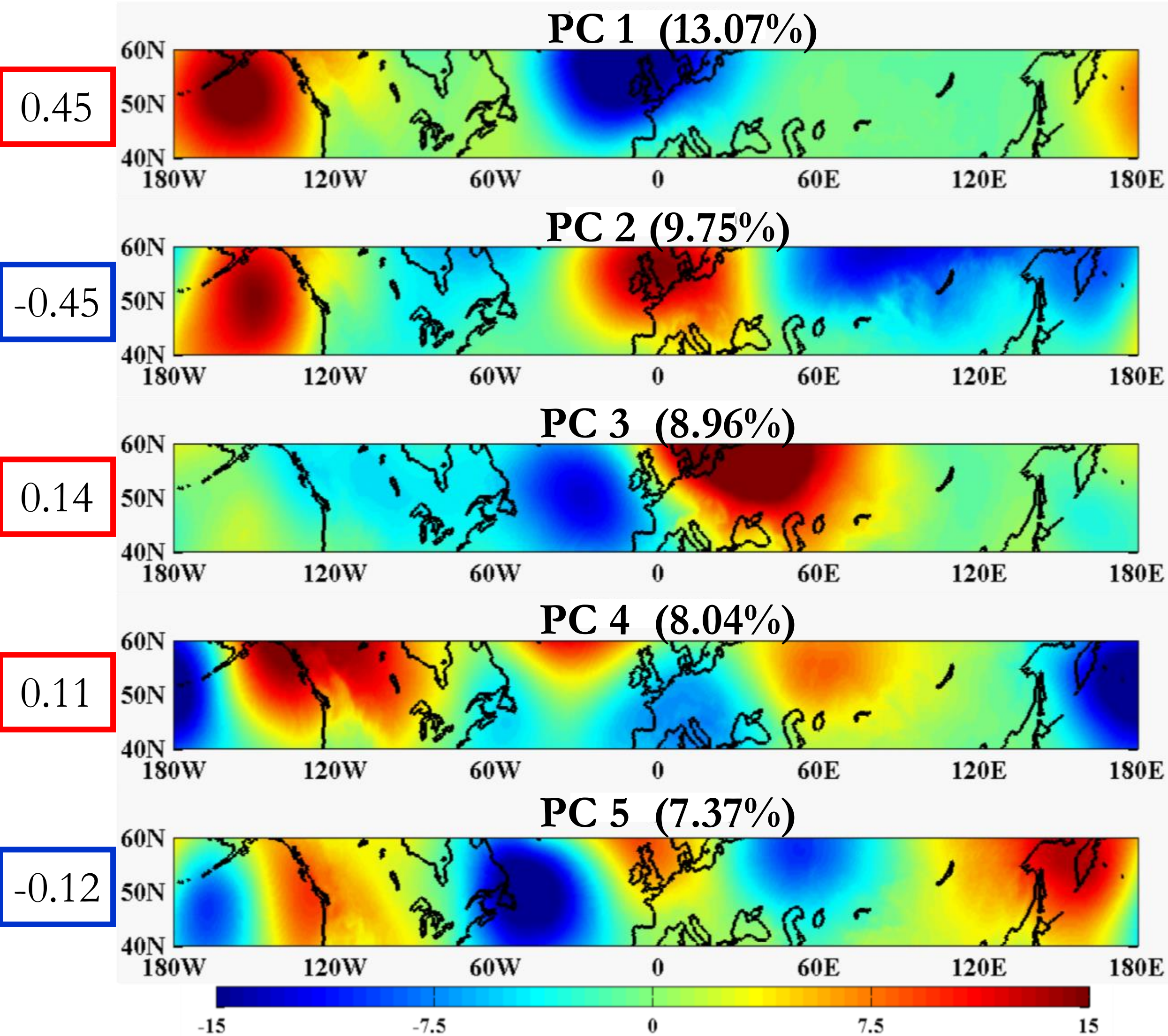
$$\mathbf{C}_1\mathbf{Y}_2^T = \mathbf{Y}_2^T\mathbf{\Lambda}_2 I / M$$

A CASE STUDY



A CASE STUDY

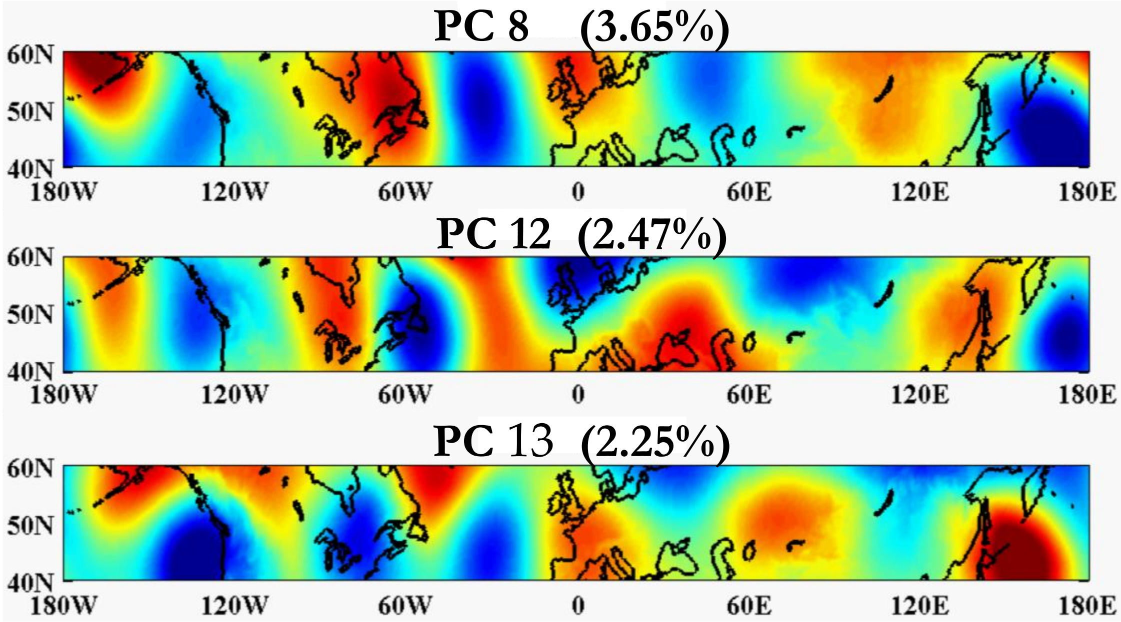




PCA extracts spatial structures and their temporal variations and changes

- Rossby wave structure with different wave numbers
- PC1-5 wavenumber 1-3, potential coupling structures

(Lu et al., 2013)

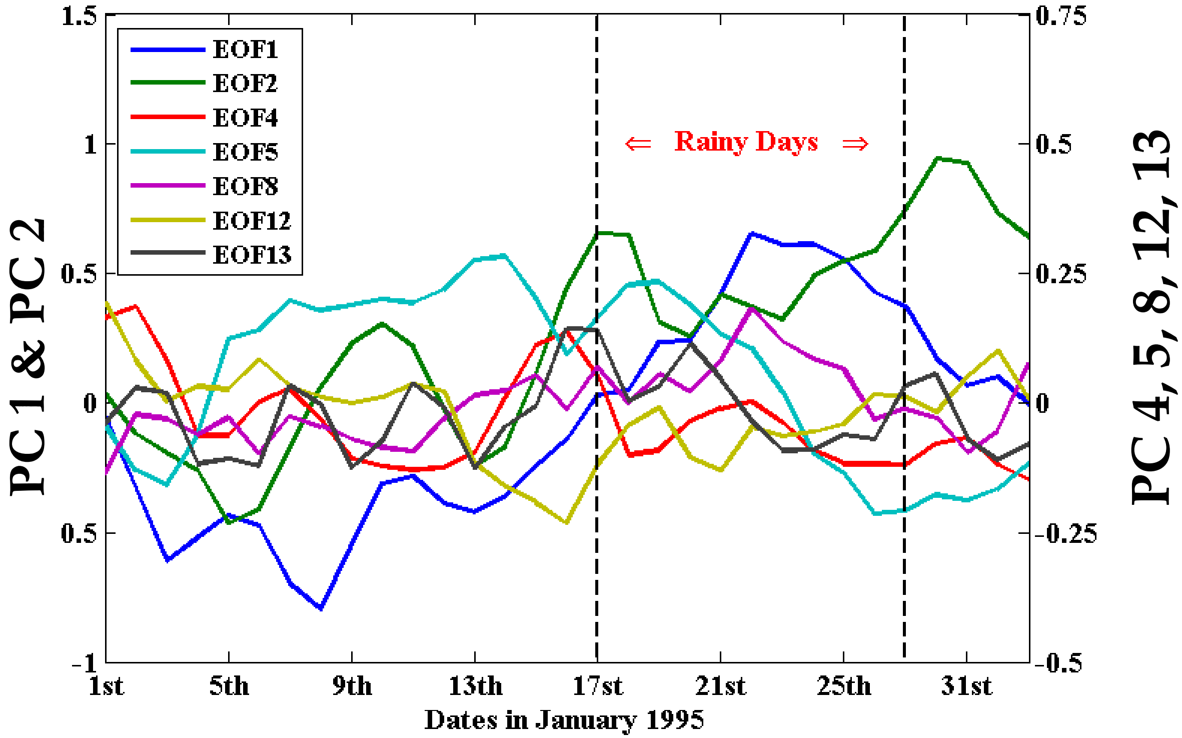


-0.12

0.19

-0.10

- PC8, 12, 13 transient structures



(Lu et al., 2013)

RECONSTRUCTION

The propagation of the composite wave structure over the extreme rainy days