Principal Components Analysis (PCA)

SYS 4021/6021

Laura Barnes and Julianne Quinn

Organization of lecture

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

1. What is principal components analysis (PCA) and why would we want to do it?
2. Simple illustration of PCA for two variables
3. Extension of PCA to multiple variables
4. Next lecture: Mathematics behind PCA

Motivation for Principal Components Analysis (PCA)

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

Visualizing and modeling data in more than 2 dimensions is conceptually difficult.

Can we collapse our data into a lower number of dimensions (principal components) that still contain most of the information?

What might be some of the benefits of doing this?

1. Reveal otherwise "hidden" patterns in the data.
2. Compress our data into a smaller set of transformed variables without losing much information.

Motivation for PCA

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

If we want to transform our data into a smaller number of variables with almost as much as information, we need a measure of "information" we want to maximize.

One measure of information we will use throughout this class is variability. If we want to predict stock prices, we need to know how they will vary – when will they be high vs. low?

The goal of PCA is to find linear combinations of our original data variables that maximize the variance of the data. This can only be applied to quantitative data variables.

Example in two dimensions

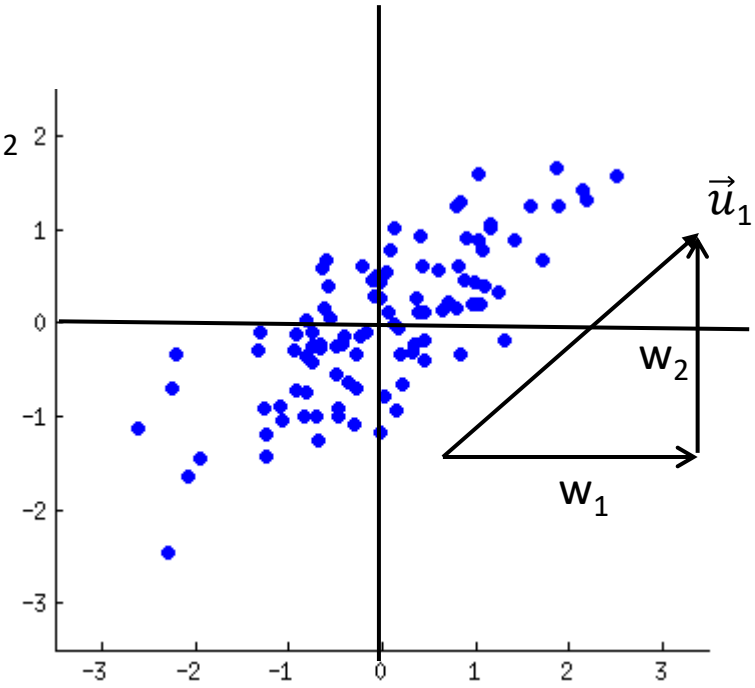
Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

First we transform our data by "centering" it:

$$X_1' = X_1 - \overline{X_1},\ X_2' = X_2 - \overline{X_2}$$

X'_2 = Rate of O_2 consumption anomaly



$\vec{u}_1 = w_1X'_1 + w_2X'_2$

What is the direction of greatest variability?

X'_1 = Heart rate anomaly

https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

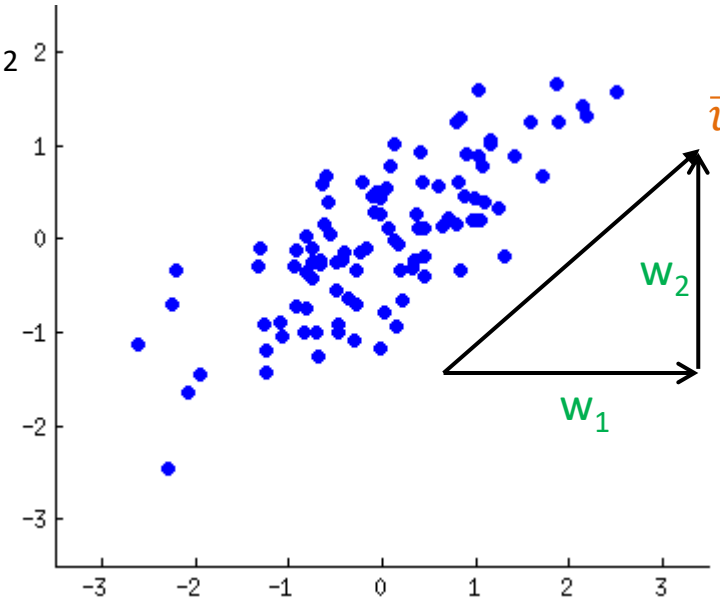
Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

We want to find w_1 and w_2 that maximizes the variance in the direction u_1.

X'_2 = Rate of O_2 consumption anomaly



$\vec{u}_1 = w_1X'_1 + w_2X'_2$

What is the direction of greatest variability?

X'_1 = Heart rate anomaly

https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

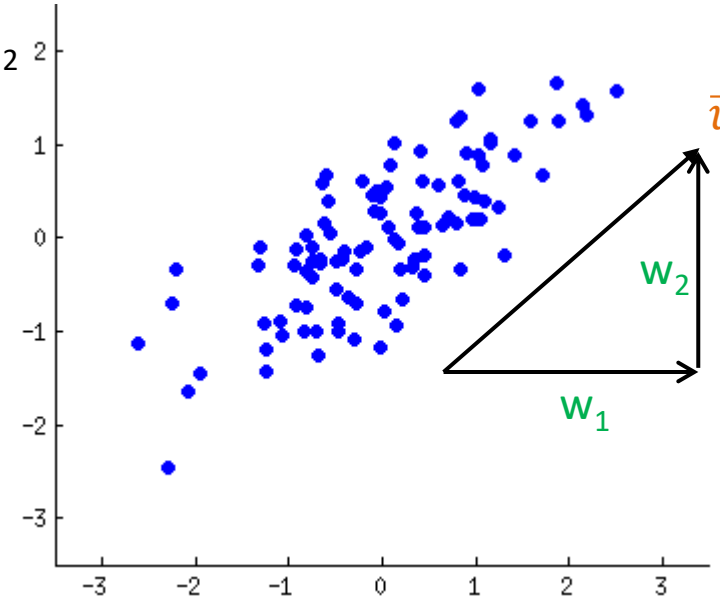
Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

We want to find w_1 and w_2 that maximizes the variance in the direction u_1. This is the variance of the blue points along the axis u_1 when they are projected onto it.

X'_2 = Rate of O_2 consumption anomaly



$\vec{u}_1 = w_1 X'_1 + w_2 X'_2$

What is the direction of greatest variability?

X'_1 = Heart rate anomaly

https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

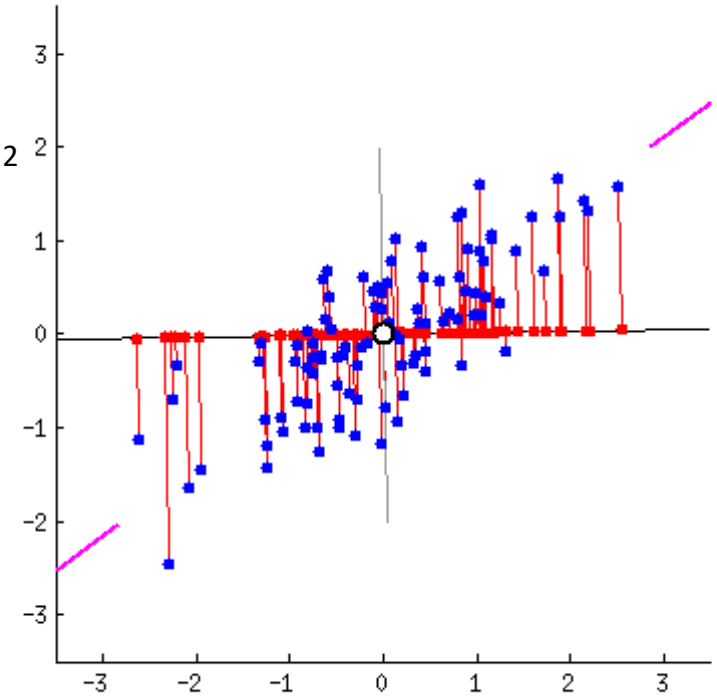
Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

We want to find w_1 and w_2 that maximizes the variance in the direction u_1

X'_2 = Rate of O_2 consumption anomaly



X'_1 = Heart rate anomaly

https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

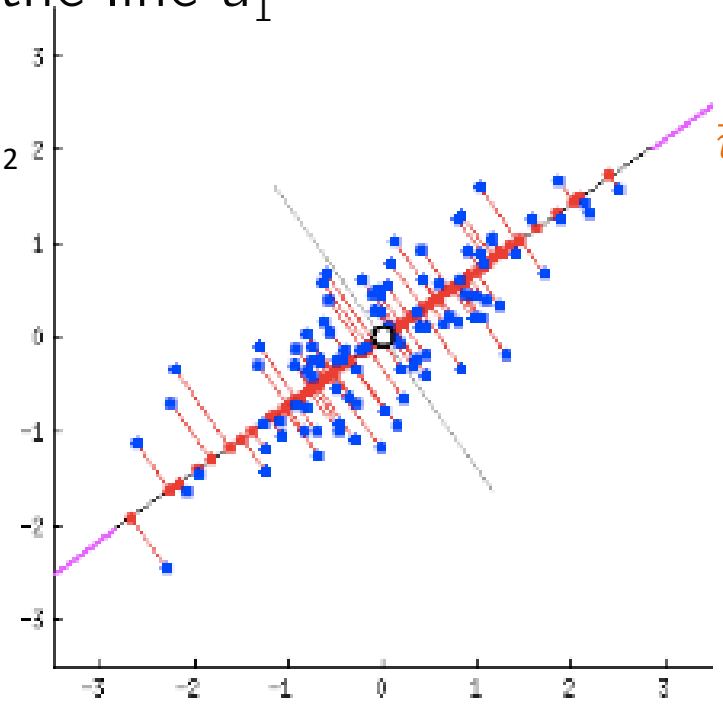
Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

This also minimizes the sum of the squared distances between each original variable (blue point) and the line u_1

X'_2 = Rate of O_2 consumption anomaly



$\vec{u}_1 = w_1 X'_1 + w_2 X'_2$

X'_1 = Heart rate anomaly

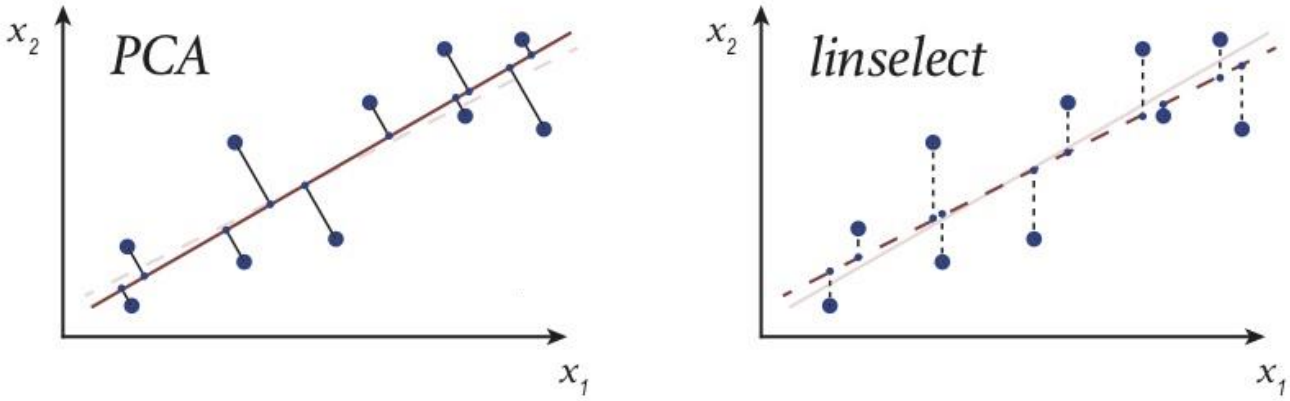
https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

Note, this is likely not the same as the line you would find in a regression. That line minimizes the sum of squared errors, which are perpendicular to x_1, not u_1.



With PCA, we're not trying to predict a particular variable, we're trying to summarize all our data variables as a whole.

https://sunjackson.github.io/2018/08/11/f3e62855f7613e1b98f4f550ee51ad9a/

Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

We now have a new variable u_1 that is a weighted sum (i.e. linear combination) of the original variables:

$$u_1 = w_1 x'_1 + w_2 x'_2$$

u_1 is called the 1st principal component (PC).

The 2nd PC, u_2, is the direction of second greatest variability, conditional on being orthogonal to u_1.

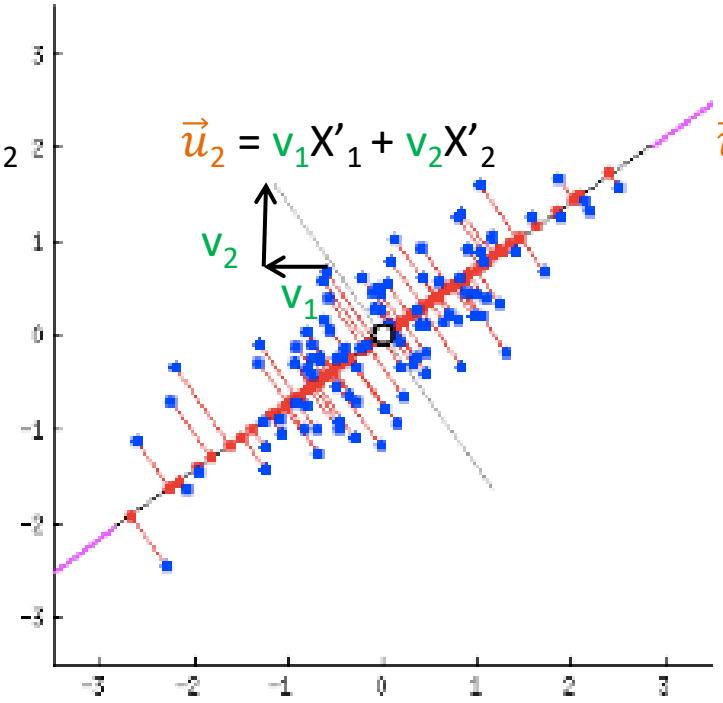
Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

If we have only two dimensions, there is only one option for the second PC.

X'_2 = Rate of O_2 consumption anomaly



X'_1 = Heart rate anomaly

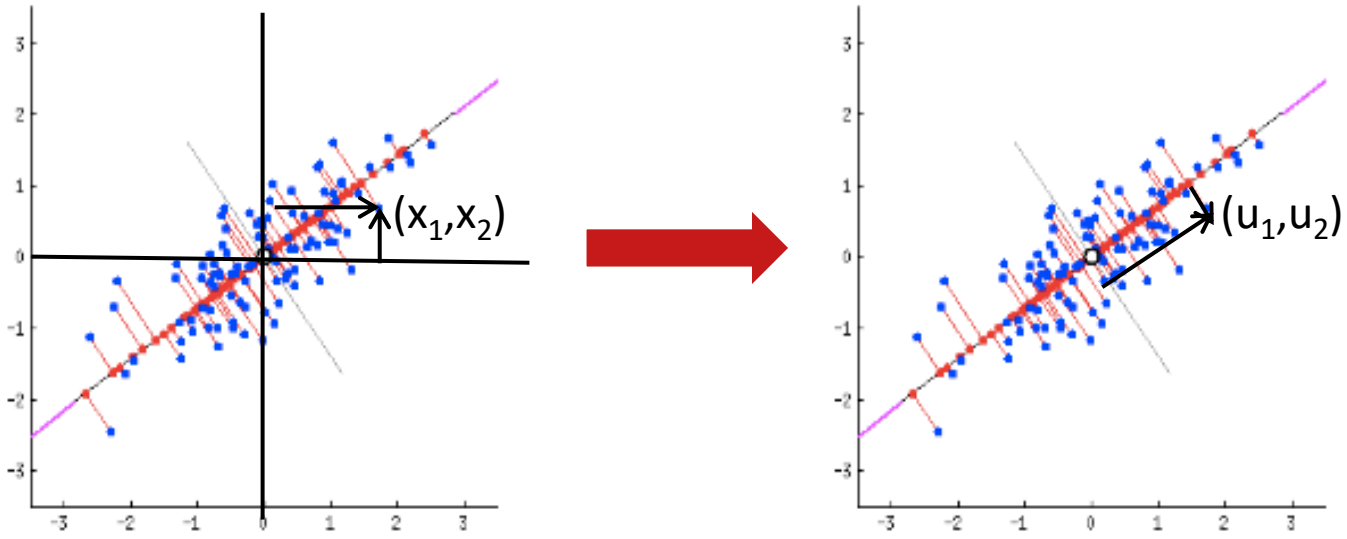
https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

With PCA, the centered data variables located at (x_1,x_2) in the original coordinate system are given new coordinates (u_1,u_2) in a rotated coordinate system



As we'll see in the next lecture, plotting our data in this (u_1,u_2) space can be an informative visualization.

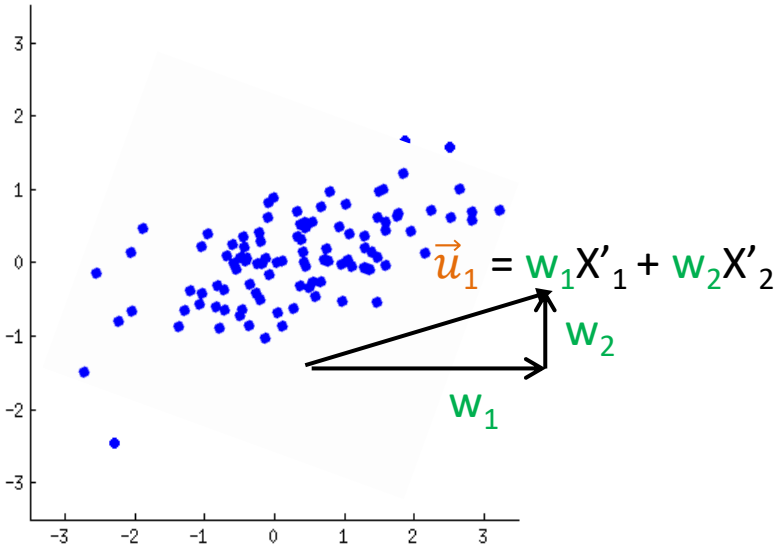
What else do we learn from PCA?

Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

Recall u_1 is a weighted sum of the original variables. Those weights, called "loadings" tell us the relative importance of the original data variables and their relationship.

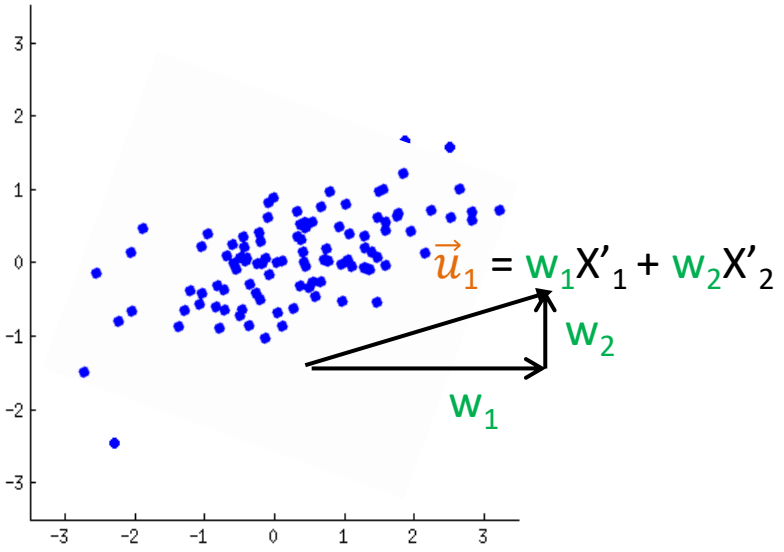


Example in two dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

If $w_1 > w_2$, x_1 explains more of the dataset's variability than x_2 and vice versa.



Example in two dimensions

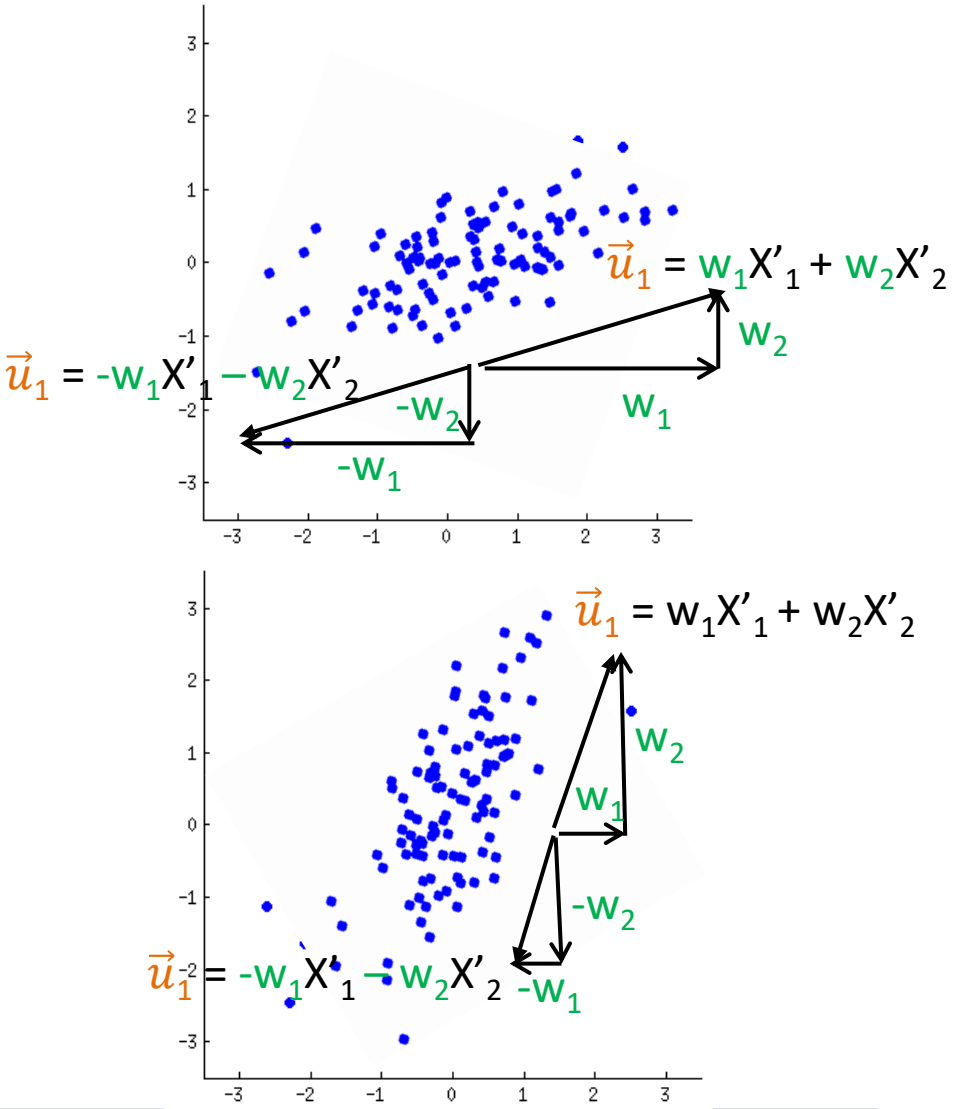
Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

If $w_1 > w_2$, x_1 explains more of the dataset's variability than x_2 and vice versa.

If w_1 and w_2 are both positive, or both negative, x_1 and x_2 are positively correlated.

If one is positive and the other is negative, those variables are negatively correlated.



Example in two dimensions

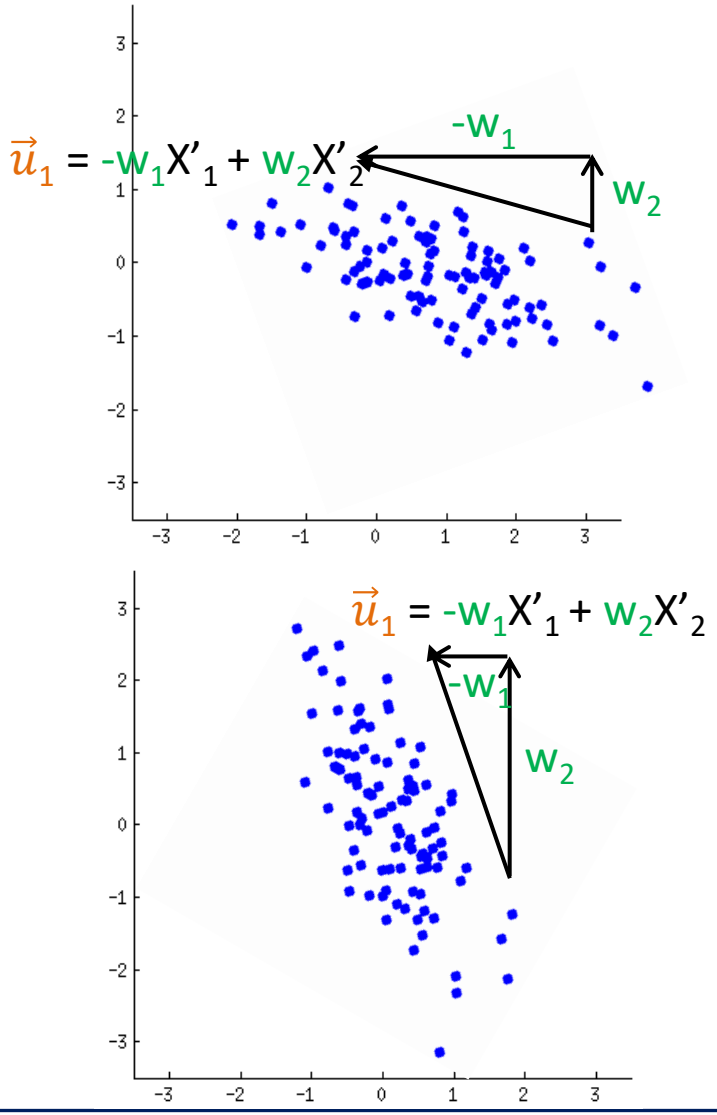
Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

If $w_1 > w_2$, x_1 explains more of the dataset's variability than x_2 and vice versa.

If w_1 and w_2 are both positive, or both negative, x_1 and x_2 are positively correlated.

If one is positive and the other is negative, those variables are negatively correlated.

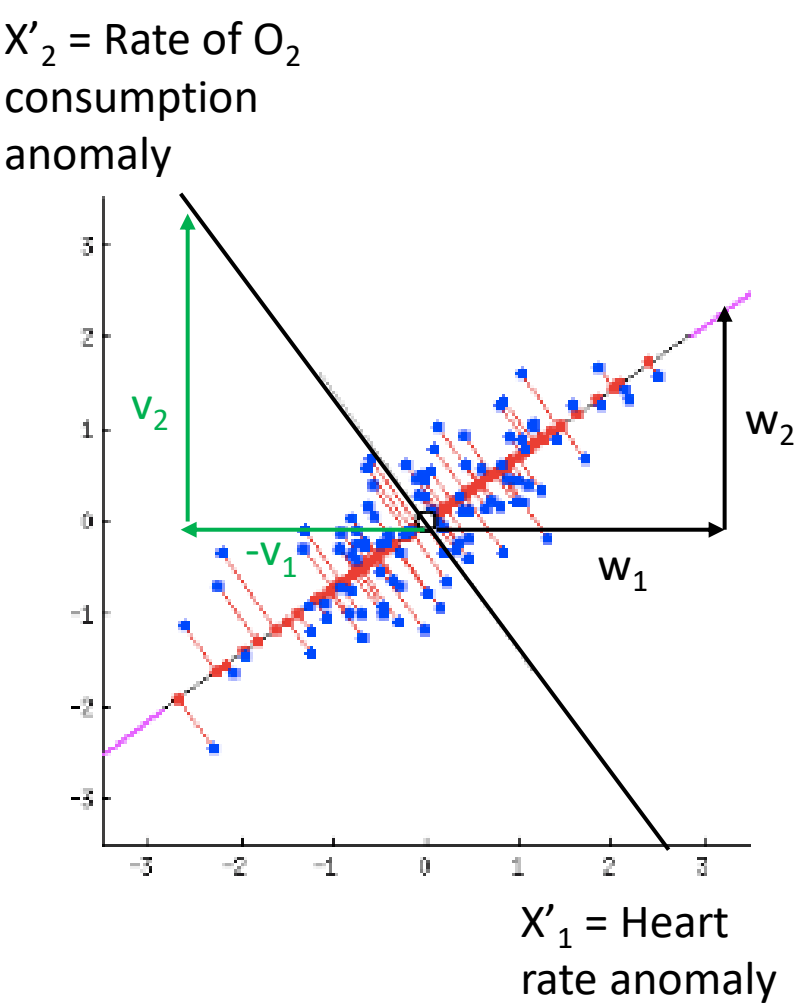


Second Principal Component

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

The weights on each variable in the 2nd PC show their relative importance (magnitude) and relationship (sign), but in the direction of 2nd greatest variability (i.e. after we've removed their primary relationship in the direction of greatest variability).



Extending this to multiple dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

Usually, we have far more than 2 variables. If we can reduce that to a smaller subset that contains most of the information, we can compress our dataset.

Imagine we have n observations of m centered variables, X:

$$X = \begin{array}{c} \begin{matrix} \text{Speed} & \text{Weight} & \text{\#Cars} & \dots & \text{Damages} \end{matrix} \\ \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1m} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2m} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & x_{n3} & \cdots & x_{nm} \end{bmatrix} \end{array} \begin{matrix} \\ \text{Accident 1} \\ \text{Accident 2} \\ \text{Accident 3} \\ \dots \\ \text{Accident } n \end{matrix}$$

How can we perform PCA on this dataset for data compression?

Extending this to multiple dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

As we just showed, PCA finds linear combinations (i.e. weighted sums), U, of the centered data variables, X, that "frontload" the information content in X:

$$[U] = [X][W]$$

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1m} \\ u_{21} & u_{22} & \cdots & u_{2m} \\ u_{31} & u_{32} & \cdots & u_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \cdots & u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ x_{31} & x_{32} & \cdots & x_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ w_{31} & w_{32} & \cdots & w_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mm} \end{bmatrix}$$

These transformed variables, U, are called the principal components.

Extending this to multiple dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

$$[U] = [X][W]$$

$$\begin{bmatrix} \overrightarrow{u_1} \\ u_{11} & u_{12} & \dots & u_{1m} \\ u_{21} & u_{22} & \dots & u_{2m} \\ u_{31} & u_{32} & \dots & u_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ x_{31} & x_{32} & \dots & x_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \begin{bmatrix} \overrightarrow{w_1} \\ w_{11} & w_{12} & \dots & w_{1m} \\ w_{21} & w_{22} & \dots & w_{2m} \\ w_{31} & w_{32} & \dots & w_{3m} \\ \dots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \dots & w_{mm} \end{bmatrix}$$

The information is "frontloaded" by first finding the weighted sum of the centered data variables ($[X]\overrightarrow{w_1}$) that represents the direction of greatest variability of the data. This is the first PC, $\overrightarrow{u_1}$.

This is determined by the weights $\overrightarrow{w_1}$ that maximize $\text{Var}(\overrightarrow{u_1}) = \sum_{i=1}^{n}(u_{i1} - \overline{u_1})^2$

Extending this to multiple dimensions

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

$$[U] = [X][W]$$

$$\begin{bmatrix} u_{11} & \overrightarrow{u_2} & \cdots & u_{1m} \\ u_{21} & u_{22} & \cdots & u_{2m} \\ u_{31} & u_{32} & \cdots & u_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \cdots & u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ x_{31} & x_{32} & \cdots & x_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \begin{bmatrix} w_{11} & \overrightarrow{w_2} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ w_{31} & w_{32} & \cdots & w_{3m} \\ \cdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mm} \end{bmatrix}$$

The second PC, $\overrightarrow{u_2}$, is the direction of next greatest variability, conditional on being orthogonal to the first PC.

The third PC, $\overrightarrow{u_3}$, is the direction of third greatest variability, conditional on being orthogonal to both the first PC and the second. And so on for up to the m^{th} PC.

What do we get out of this?

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

We said PCA could achieve two things:

1. Reveal otherwise "hidden" patterns in the data.
2. Compress our data into a small set of (transformed) variables without losing much information.

The loadings or weights can help reveal "hidden patterns" in our data, viewing the relationships between multiple variables simultaneously in different directions/"modes" of variability

And since the information/variance of our original data is "frontloaded" in the PCs, we can keep just the first K PCs (K of M columns of [U]) and still retain most of the information

What do we get out of this?

Agenda

- What is PCA?
- Illustration for 2 variables
- Extension to multiple variables
- Next: Mathematical derivation

So how do we find these weights?

And how do we decide on how many PCs to retain?

We'll answer these questions in our next lecture

Principal Components Analysis 2 (PCA)

Laura Barnes and Julianne Quinn

Organization of Lecture

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

1. Recap: What is PCA?
2. How do we find the principal components mathematically?
3. Interpreting the outputs of PCA
4. Summary

Recap: What is PCA?

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

PCA collapses our data into a lower number of dimensions (principal components) that still contain most of the information.

This has two main benefits:

1. Can reveal otherwise "hidden" patterns in the data.
2. Can be used to compress our data into a smaller set of transformed variables without losing much information.

Recap: What is PCA?

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

PCA finds weighted sums, [U], of our original data, [X]:

$$[U] = [X][W]$$

$$\begin{bmatrix} \overrightarrow{u_1} & \overrightarrow{u_2} & \\ u_{11} & u_{12} & \dots u_{1m} \\ u_{21} & u_{22} & \dots u_{2m} \\ u_{31} & u_{32} & \dots u_{3m} \\ \vdots & \vdots & \ddots \ \vdots \\ u_{n1} & u_{n2} & \dots u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots x_{1m} \\ x_{21} & x_{22} & \dots x_{2m} \\ x_{31} & x_{32} & \dots x_{3m} \\ \vdots & \vdots & \ddots \ \vdots \\ x_{n1} & x_{n2} & \dots x_{nm} \end{bmatrix} \begin{bmatrix} \overrightarrow{w_1} & \overrightarrow{w_2} & \\ w_{11} & w_{12} & \dots w_{1m} \\ w_{21} & w_{22} & \dots w_{2m} \\ w_{31} & w_{32} & \dots w_{3m} \\ \dots & \vdots & \ddots \ \vdots \\ w_{m1} & w_{m2} & \dots w_{mm} \end{bmatrix}$$

The first PC, $\overrightarrow{u_1}$, is the direction of greatest variability. The weights on the original data variables are $\overrightarrow{w_1}$.

The second PC, $\overrightarrow{u_2}$, is the direction of next greatest variability, conditional on being orthogonal to the first PC. The weights on the original data variables are $\overrightarrow{w_2}$.

The third PC, $\overrightarrow{u_3}$, is the direction of third greatest variability, conditional on being orthogonal to both the first PC and the second. And so on for up to the m^{th} PC.

Recap: What is PCA?

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

So how do we find these weights?

And how do we decide on how many PCs to retain?

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

The 1st weight vector, $\overrightarrow{w_1}$, is that which maximizes the variance of the 1st PC, $\overrightarrow{u_1}$, i.e.:

$$\max_{\overrightarrow{w_1}} Var(\overrightarrow{u_1}) = Var([X]\overrightarrow{w_1})$$

$\overrightarrow{w_1}$ is a vector of constant weights, while [X] is a matrix of random variables.

The variance of the product of a random variable, Y, and a constant c is $Var(cY) = c^2Var(Y)$. In matrix notation:

$$Var([X]\overrightarrow{w_1}) = w_1^T \mathrm{Var}([\mathrm{X}])w_1$$

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

The 1st weight vector, $\overrightarrow{w_1}$, is that which maximizes the variance of the 1st PC, $\overrightarrow{u_1}$, i.e.:

$$\max_{\overrightarrow{w_1}} Var(\overrightarrow{u_1}) = Var([X]\overrightarrow{w_1}) = w_1^T \mathrm{Var}([\mathrm{X}])w_1$$

What is Var([X])? This is the variance-covariance matrix of the variables in X, [S]:

$$[S] = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1m} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2m} \\ \sigma_{31} & \sigma_{32} & \cdots & \sigma_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{m1} & \sigma_{m2} & \cdots & \sigma_m^2 \end{bmatrix}$$

$$\sigma_i^2 = Var(X_i)$$

$$\sigma_{ij} = \sigma_{ji} = Cov(X_i, X_j)$$

Solving for PCA weight matrix

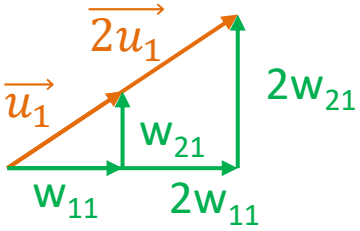
Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

The 1st weight vector, $\overrightarrow{w_1}$, is that which maximizes the variance of the 1st PC, $\overrightarrow{u_1}$, i.e.:

$$\max_{\overrightarrow{w_1}} Var(\overrightarrow{u_1}) = Var([X]\overrightarrow{w_1}) = w_1^T \mathrm{Var}([\mathrm{X}])w_1 = w_1^T [\mathrm{S}]w_1$$

Well, if we just made the weights in $\overrightarrow{w_1}$ infinitely large, this would maximize this equation! But linear multipliers of a weight vector all represent the same direction.



So we need to constrain this optimization problem.

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

The 1st weight vector, $\overrightarrow{w_1}$, is that which maximizes the variance of the 1st PC, $\overrightarrow{u_1}$, i.e.:

$$\max_{\overrightarrow{w_1}} Var(\overrightarrow{u_1}) = Var([X]\overrightarrow{w_1}) = w_1^T \mathrm{Var}([\mathrm{X}])w_1 = w_1^T [\mathrm{S}] w_1$$

Let's constrain the length of $\overrightarrow{w_1}$ to 1, i.e. $\|\overrightarrow{w_1}\| = \sqrt{w_1^T w_1} = 1$

Recall from calculus, that to constrain an optimization problem, add the product of a Lagrange multiplier, λ, and the value of the constraint equal to 0, here $1 - w_1^T w_1$:

$$\max_{\overrightarrow{w_1}} w_1^T [\mathrm{S}] w_1 + \lambda_1 (1 - w_1^T w_1)$$

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

How do we find $\overrightarrow{w_1}$ that maximizes the equation below?

$$\max_{\overrightarrow{w_1}} w_1^T [S] w_1 + \lambda_1 (1 - w_1^T w_1)$$

Take the derivative with respect to $\overrightarrow{w_1}$, set it equal to 0, and solve for $\overrightarrow{w_1}$.

$$\frac{\partial}{\partial w_1} \left[w_1^T [S] w_1 + \lambda_1 (1 - w_1^T w_1) \right] = 0$$

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

$$\frac{\partial}{\partial w_1}\left[w_1^T[S]w_1 + \lambda_1(1 - w_1^T w_1)\right] = 0$$

$w_1^T w_1$ is matrix notation for the vector squared, so the above derivative is:

$$2[S]\overrightarrow{w_1} - 2\lambda_1\overrightarrow{w_1} = 0$$
$$[S]\overrightarrow{w_1} = \lambda_1\overrightarrow{w_1}$$

[S] is a square symmetric matrix, λ_1 is a constant and $\overrightarrow{w_1}$ is a vector. $\overrightarrow{w_1}$ therefore must be an eigenvector of [S].

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

$$[S]\overrightarrow{w_1} = \lambda_1 \overrightarrow{w_1}$$
$$[S]\overrightarrow{e_1} = \lambda_1 \overrightarrow{e_1}$$

where $\overrightarrow{e_1}$ is the 1st eigenvector of [S] and λ_1 is the first eigenvalue of [S]

So the 1st weight vector is the 1st eigenvector of [S], the variance-covariance matrix of the data [X].

What about the 2nd weight vector? This determines the direction of next greatest variability, conditional on being perpendicular to the 1st PC.

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

To find $\overrightarrow{w_2}$ we need to add another constraint to the expression below that $\overrightarrow{w_1} \perp \overrightarrow{w_2}$:

$$\max_{\overrightarrow{w_2}} w_2^T [S] w_2 + \lambda_2 (1 - w_2^T w_2)$$

If $\overrightarrow{w_1} \perp \overrightarrow{w_2}$, then their dot product is 0: $w_1^T w_2 = 0$, so we add another Lagrange multiplier, ϕ, to this expression:

$$\max_{\overrightarrow{w_2}} w_2^T [S] w_2 + \lambda_2 \left(1 - w_2^T w_2\right) + \phi w_1^T w_2$$

Once again, we find $\overrightarrow{w_2}$ by taking the derivative with respect to $\overrightarrow{w_2}$, setting it to 0, and solving for $\overrightarrow{w_2}$.

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

$$\frac{\partial}{\partial w_2}\left[w_2^T[S]w_2 + \lambda_2\left(1 - w_2^T w_2\right) + \phi w_1^T w_2\right] = 0$$

$w_2^T w_2$ is matrix notation for the vector squared, so the above derivative is:

$$2[S]\overrightarrow{w_2} - 2\lambda_2\overrightarrow{w_2} + \phi\overrightarrow{w_1} = 0$$

To solve the above equation, we need to know the value of ϕ, which we can find by multiplying both sides by $\overrightarrow{w_1}$:

$$2w_1^T[S]w_2 - 2\lambda_2 w_1^T w_2 - \phi w_1^T w_1 = 0$$

($w_1^T[S]w_2$ and $w_1^T w_2$ are each labeled 0)

$w_1^T w_2 = 0$ because $\overrightarrow{w_1} \perp \overrightarrow{w_2}$

$w_1^T w_1 = 1$ by our constraint, so ϕ=0

Solving for PCA weight matrix

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

$$\frac{\partial}{\partial w_2}\left[w_2^T[S]w_2 + \lambda_2\left(1 - w_2^T w_2\right) + \phi w_1^T w_2\right] = 0$$

$w_2^T w_2$ is matrix notation for the vector squared, so the above derivative is:

$$2[S]\overrightarrow{w_2} - 2\lambda_2\overrightarrow{w_2} + \phi\overrightarrow{w_1} = 0$$

Therefore the above equation simplifies to:

$$[S]\overrightarrow{w_2} = \lambda_2\overrightarrow{w_2}$$

and $\overrightarrow{w_2} = \overrightarrow{e_2}$, the 2nd eigenvector while λ_2 is the 2nd eigenvalue

Extending this to multiple dimensions

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We have shown the 1st weight vector is the 1st eigenvector of [S], where [S] = Cov([X]), and the 2nd weight vector is the 2nd eigenvector.

If we kept going, we would see the weight matrix is the eigenvector matrix of [S], i.e.:

$$[U] = [X][W] = [X][E]$$

$$\begin{matrix} \overrightarrow{u_1} & \overrightarrow{u_2} & \dots & \overrightarrow{u_m} \end{matrix}$$

$$\begin{bmatrix} u_{11} & u_{12} & \dots & u_{1m} \\ u_{21} & u_{22} & \dots & u_{2m} \\ u_{31} & u_{32} & \dots & u_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ x_{31} & x_{32} & \dots & x_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1m} \\ e_{21} & e_{22} & \dots & e_{2m} \\ e_{31} & e_{32} & \dots & e_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m2} & \dots & e_{mm} \end{bmatrix}$$

$$\begin{matrix} \overrightarrow{e_1} & \overrightarrow{e_2} & \dots & \overrightarrow{e_m} \end{matrix}$$

Understanding the eigenvector elements

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

$$\begin{bmatrix} u_{11} & u_{12} & \dots & u_{1m} \\ u_{21} & u_{22} & \dots & u_{2m} \\ u_{31} & u_{32} & \dots & u_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ x_{31} & x_{32} & \dots & x_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1m} \\ e_{21} & e_{22} & \dots & e_{2m} \\ e_{31} & e_{32} & \dots & e_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m2} & \dots & e_{mm} \end{bmatrix}$$

The elements of each eigenvector are the loadings on each variable in that PC:

e_{11} = weight on X_1 (variable 1) in 1st PC
e_{21} = weight on X_2 (variable 2) in 1st PC
e_{31} = weight on X_3 (variable 3) in 1st PC
…
e_{m1} = weight on X_m (variable m) in 1st PC

How do we get these in R?

Performing PCA in R

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We can perform PCA in R by passing our data frame (of quantitative variables only!) to the function `princomp()`:

```
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG","TOTKLD","TOTINJ")])
> names(xdmgnd.pca)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
```

$[E]$

```
> xdmgnd.pca$loadings

Loadings:
        Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6
CARSDMG                              1.000
EQPDMG   0.506  0.824  0.254
TRKDMG         -0.345  0.934
ACCDMG   0.857 -0.450 -0.251
TOTKLD                                     1.000
TOTINJ                       1.000
```

This only prints significant loadings

Performing PCA in R

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We can perform PCA in R by passing our data frame (of quantitative variables only!) to the function `princomp()`:

```
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG","TOTKLD","TOTINJ")])
> names(xdmgnd.pca)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
```

$[E]$

Specify the columns to get all values

```
> xdmgnd.pca$loadings[,1:6]
              Comp.1       Comp.2       Comp.3       Comp.4       Comp.5       Comp.6
CARSDMG 1.543819e-07  3.610677e-07  3.423203e-07  5.671653e-03  9.999614e-01  6.710669e-03
EQPDMG  5.063181e-01  8.241067e-01  2.539490e-01 -4.758192e-06 -4.347891e-07 -1.337292e-07
TRKDMG  9.235641e-02 -3.446137e-01  9.341904e-01  4.302466e-07 -2.124766e-07  6.058414e-08
ACCDMG  8.573869e-01 -4.495437e-01 -2.505957e-01  5.019561e-08  1.156743e-07 -3.276351e-08
TOTKLD  1.002705e-07  1.334579e-07 -2.924470e-08  4.735083e-03 -6.737557e-03  9.999661e-01
TOTINJ  2.325099e-06  4.089524e-06  8.172040e-07  9.999727e-01 -5.639686e-03 -4.773113e-03
```

Performing PCA in R

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We can perform PCA in R by passing our data frame (of quantitative variables only!) to the function `princomp()`:

```
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG","TOTKLD","TOTINJ")])
> names(xdmgnd.pca)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
```

$[E]$ Convert them into a data frame to plot them

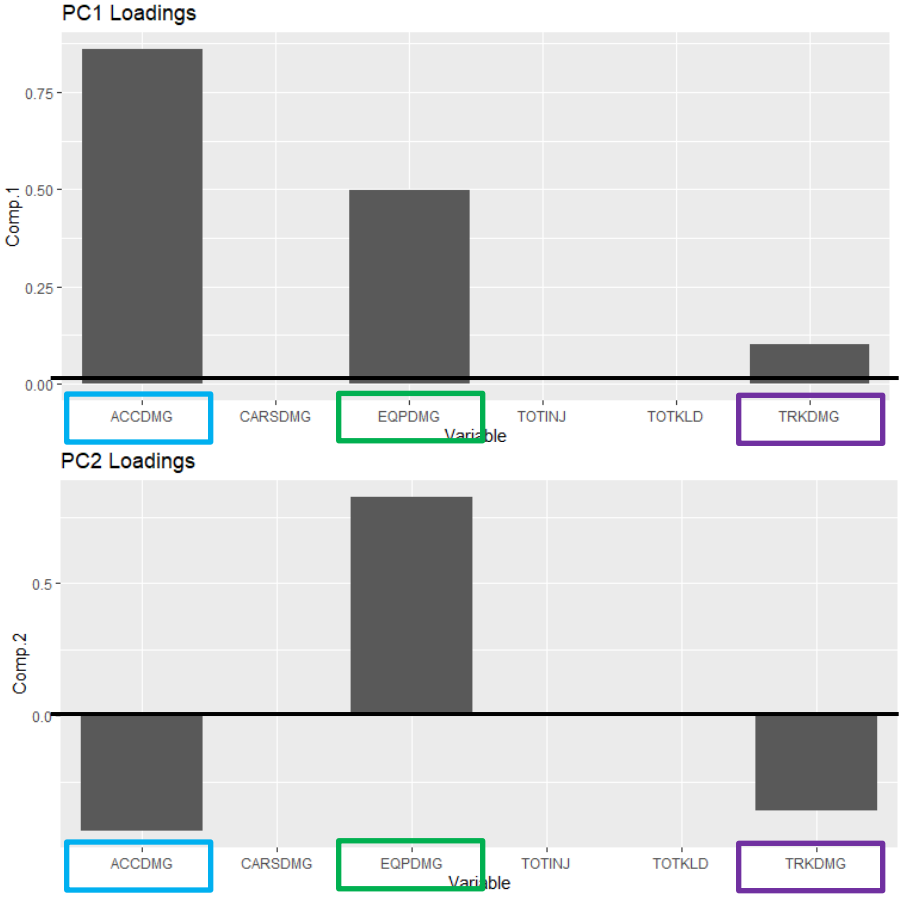
```
> df = as.data.frame(xdmgnd.pca$loadings[,1:6]) # get first 6 columns which are the 6 PCs
> setDT(df, keep.rownames=TRUE)[]
         rn       Comp.1       Comp.2       Comp.3       Comp.4       Comp.5       Comp.6
1: CARSDMG 1.543819e-07  3.610677e-07  3.423203e-07  5.671653e-03  9.999614e-01  6.710669e-03
2:  EQPDMG 5.063181e-01  8.241067e-01  2.539490e-01 -4.758192e-06 -4.347891e-07 -1.337292e-07
3:  TRKDMG 9.235641e-02 -3.446137e-01  9.341904e-01  4.302466e-07 -2.124766e-07  6.058414e-08
4:  ACCDMG 8.573869e-01 -4.495437e-01 -2.505957e-01  5.019561e-08  1.156743e-07 -3.276351e-08
5:  TOTKLD 1.002705e-07  1.334579e-07 -2.924470e-08  4.735083e-03 -6.737557e-03  9.999661e-01
6:  TOTINJ 2.325099e-06  4.089524e-06  8.172040e-07  9.999727e-01 -5.639686e-03 -4.773113e-03
> names(df)[names(df) == "rn"] <- "Variable"
> ggplot(data=df, aes(x=Variable,y=Comp.1)) + geom_bar(stat="identity") + ggtitle("PC1 Loadings")
> ggplot(data=df, aes(x=Variable,y=Comp.2)) + geom_bar(stat="identity") + ggtitle("PC2 Loadings")
```

PCA for data visualization through the loadings

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

Viewing the loadings allows us to see relationships between multiple variables at once in different "modes" (PCs) of variability



In the 1st PC, ACCDMG > EQPDMG > TRKDMG > CARSDMG, TOTINJ and TOTKLD and they all vary in the same direction

In the 2nd PC, EQPDMG > ACCDMG > TRKDMG > CARSDMG, TOTINJ and TOTKLD.

EQPDMG varies opposite ACCDMG and TRKDMG

What are the eigenvalues?

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

So the eigenvector elements represent the weights on each variables, but what do the eigenvalues represent?

Recall each of our eigenvectors/eigenvalues were obtained from the equation:

$$[S]\overrightarrow{e_i} = \lambda_i \overrightarrow{e_i}$$

If we left multiply both sides by e_i^T, we get:

$$e_i^T [S] e_i = e_i^T \lambda_i e_i$$
$$e_i^T Var([X]) e_i = \lambda_i$$
$$Var([X]\overrightarrow{e_i}) = \lambda_i$$
$$Var(\overrightarrow{u_i}) = \lambda_i$$

Because λ_i is a constant and $e_i^T e_i = 1$

So the eigenvalues represent the variance in each PC

Performing PCA in R

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We can perform PCA in R by passing our data frame (of quantitative variables only!) to the function `princomp()`:

```
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG","TOTKLD","TOTINJ")])
> names(xdmgnd.pca)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
```

$\sqrt{\lambda_i}$

```
> xdmgnd.pca$sdev
     Comp.1      Comp.2      Comp.3      Comp.4      Comp.5      Comp.6
1.382290e+06 4.771678e+05 3.018959e+05 1.452932e+01 2.628997e+00 3.702536e-01
```

How are the eigenvalues informative?

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

Since we've "frontloaded" the information content of our data, we can retain some subset *K* of the *M* PCs that retain most of the information in our dataset.

K=2 PCs

$$\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ u_{31} & u_{32} \\ \vdots & \vdots \\ u_{n1} & u_{n2} \end{bmatrix} \begin{matrix} \dots u_{1m} \\ \dots u_{2m} \\ \dots u_{3m} \\ \ddots \quad \vdots \\ \dots u_{nm} \end{matrix} = \begin{bmatrix} x_{11} & x_{12} & \dots x_{1m} \\ x_{21} & x_{22} & \dots x_{2m} \\ x_{31} & x_{32} & \dots x_{3m} \\ \vdots & \vdots & \ddots \quad \vdots \\ x_{n1} & x_{n2} & \dots x_{nm} \end{bmatrix} \begin{bmatrix} e_{11} & e_{12} & \dots e_{1m} \\ e_{21} & e_{22} & \dots e_{2m} \\ e_{31} & e_{32} & \dots e_{3m} \\ \vdots & \vdots & \ddots \quad \vdots \\ e_{m1} & e_{m2} & \dots e_{mm} \end{bmatrix}$$

We can choose *K* based on how much variance is explained by the PCs, determined by λ_i for the i^{th} PC.

Scree plot

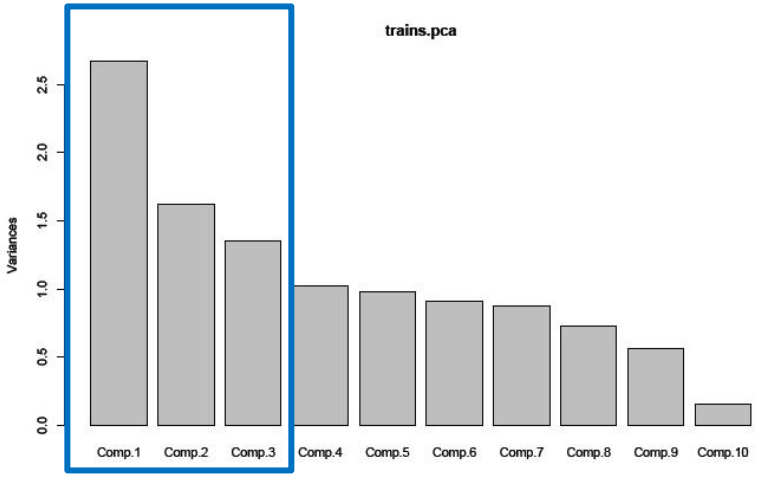
Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

One common approach is to plot the variance in each component on a scree plot and choose as many PCs before a "kink" in the plot. This is somewhat arbitrary, and there may not always be a kink.

```
screeplot(xdmgnd.pca, main = "Variance of each PC")

# If you've installed ggbiplot
ggscreeplot(xdmgnd.pca)
```

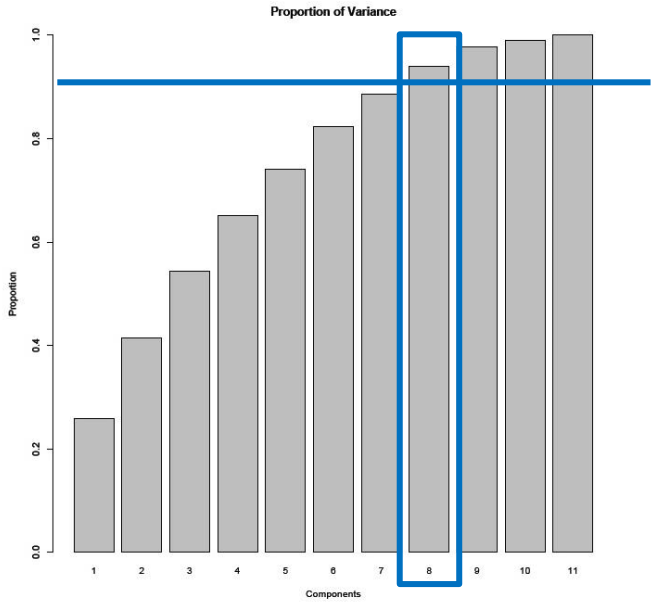


Another approach is to choose as many PCs as needed to retain y% of the variance, where the user chooses y.

```
source("PCAplots.R")

cumsum <- cumplot(xdmgnd.pca, col = "blue")

ggplot(data=cumsum,
       aes(x=Component,y=Proportion)) +
  geom_bar(stat="identity") +
  ggtitle("Cumulative Variance in the PCs")
```



PCA for data compression

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

For regression, we might choose K < M PCs representing p% of the variability and use those as predictors in a model. This is convenient because they are orthogonal, and many regression methods require independent predictors.

But what if we just wanted to represent the data in a lower dimension? This can be done using PCA synthesis in which we invert [U]=[X][E] to reproduce [X] from [U]:

$$[U] = [X][E]$$
$$[U][E]^T = [X][E][E]^T$$
$$[U][E]^T = [X][I] = [X]$$

PCA for data compression

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

$$[X] = [U][E]^T$$

[X] is an (N x M) matrix (M variables with N observations)
[U] is an (N x M) matrix (M PCs with N observations)
[E] is an (M x M) matrix (M eigenvectors with M variable weights)

What if we used only K columns of [U], the first K PCs?

We would need to similarly just use the first K eigenvectors of [E] (rows of $[E]^T$). This would approximate [X].

Recap

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

PCA finds weighted sums of the original data variables that represent the orthogonal directions of greatest sequential variability

The weights are determined by the eigenvectors ([E])

The variances in each direction are the eigenvalues (λ)

But what are the PCs? ([U])

Performing PCA in R

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We can perform PCA in R by passing our data frame (of quantitative variables only!) to the function `princomp()`:

```
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG","TOTKLD","TOTINJ")])
> names(xdmgnd.pca)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
```

$[U]$

`head()` will print only the first few rows

```
> head(xdmgnd.pca$scores)
        Comp.1     Comp.2      Comp.3      Comp.4     Comp.5      Comp.6
[1,] 3688609.5 1122400.24 -156269.430 35.04000018 -1.8534389 0.267196058
[2,] -453250.9 -156410.55  100457.291  0.71761031  3.3561117 0.058289821
[3,] -552251.1   86482.43  -74188.528  0.07432821 -0.6526807 0.006727952
[4,] -649548.2  -63802.45    5359.265  0.85051003 -0.6152211 0.035445119
[5,] 1524464.8   49193.96  323323.430 -4.91286760  1.9334952 -0.143869720
[6,] -618330.2  -77787.42   20003.397  0.82879100  0.3801175 0.041418735
```

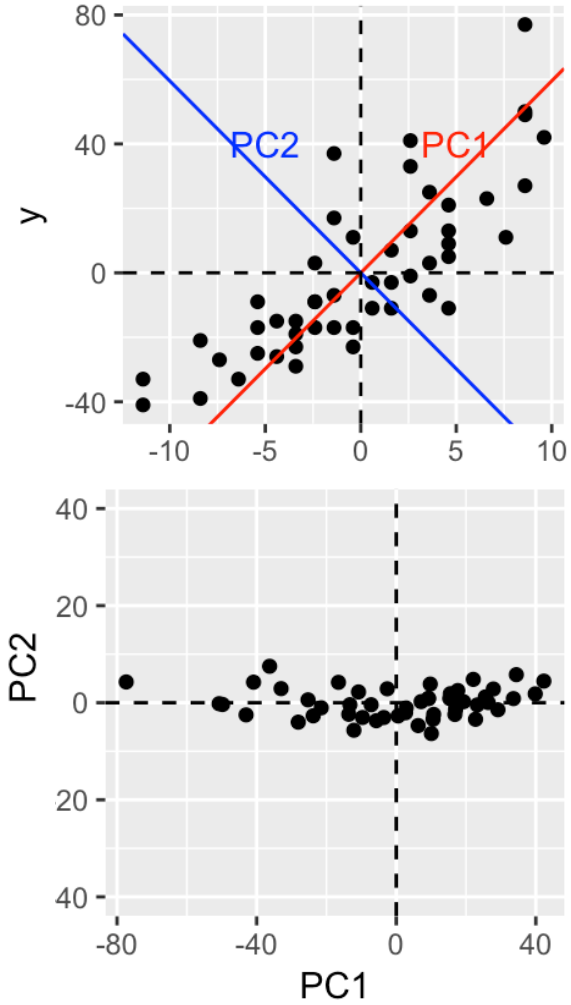
What do the PCs themselves represent?

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

The PCs are weighted sums of the original variables. They also represent a new, rotated coordinate system in which we can view the data: (u_1, u_2) instead of (x_1, x_2).

A plot of the data in the (u_1, u_2) plane is called a biplot. In this case, the biplot shows no pattern because almost all of the information is explained in the 1st 2 PCs.



http://www.sthda.com/english/articles/31-principal-component-methods-in-r-practical-guide/112-pca-principal-component-analysis-essentials/

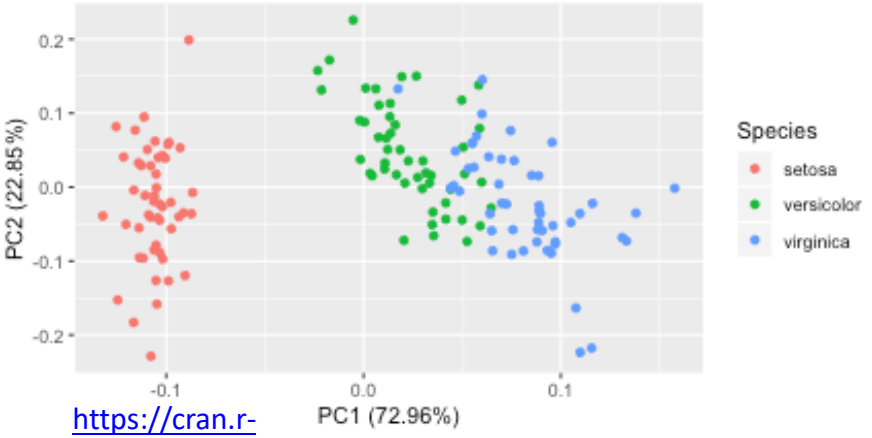
What do the PCs themselves represent?

Agenda

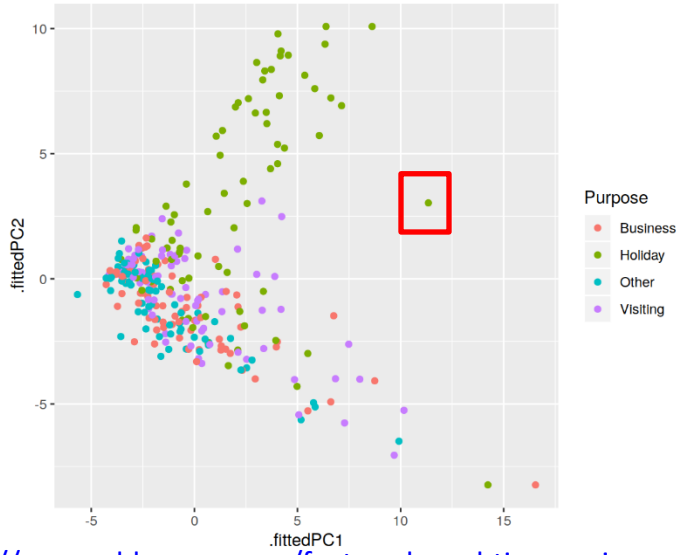
- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

But sometimes the first 2 PCs separate the data into clusters, or reveal additional modes of variability beyond the 1st 2 PCs.

They can also reveal multivariate outliers.



https://cran.r-project.org/web/packages/ggfortify/vignettes/plot_pca.html



https://www.r-bloggers.com/feature-based-time-series-analysis/

Additional features of the biplot

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

We can also project the original data variables as vectors onto the 1st 2 PCs and display them on the biplot.

The projection for variable X_i in the 1st two PCs is the vector $(e_1^T b_i, e_2^T b_i)$ where b_i is a column vector of 0s except for 1 in element *i:*

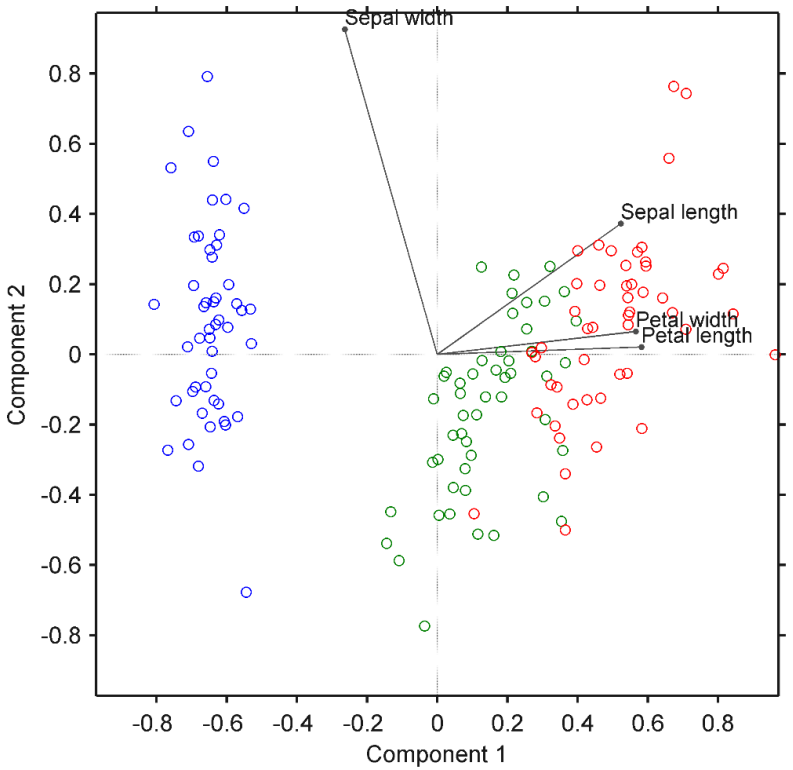
$$b_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \; b_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \; \ldots \; b_m = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

Projecting the original data variables onto the biplot

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

Similar to the loadings plot, the length of the vector indicates the importance of that variable, and the extent to which variables point in the same direction indicates their correlation in the first 2 modes of variability.



https://iaisidro.wordpress.com/2015/10/09/biplotg/

```
# View the data in the first 2 PCs
biplot(xdmgnd.pca, main="Biplot of Extreme Accident Metrics")

# If you've installed ggbiplot
ggbiplot(xdmgnd.pca, varname.size = 5, labels=row(xdmgnd)[,1],
         main="Biplot of Extreme Accident Metrics")
```

Summary

Agenda

- Recap: What is PCA?
- Mathematical derivation
- Interpreting PCA outputs
  - Eigenvectors
  - Eigenvalues
  - Principal Components
- Summary

You will not need to know how to derive the weight vector for PCA, but you should know what comes out of that derivation:

1. The PCs represent the directions of greatest variability of the data, conditional on them all being orthogonal.

   We plot the 1st 2 PCs on a biplot

2. The PCs are weighted sums of the original data variables. The weights for each PC are the elements of the corresponding eigenvector. These weights are called loadings.

   We plot the loadings on bar plots

3. The variance in the direction of each PC is equal to the corresponding eigenvalue.

   We plot the eigenvalues on a scree plot or cumulative variance plot

Principal Components Analysis 2 (PCA)

Laura Barnes and Julianne Quinn

Organization of lecture

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

1. Recap:
   a) What is PCA?
   b) How does it work?
2. Potential shortcomings of performing PCA as described so far and an alternative
3. Performing PCA on gridded data
4. Summary

Recap: What is PCA?

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Visualizing and modeling data in more than 2 dimensions is conceptually difficult.

PCA transforms our data into a new variables (principal components) that sequentially maximize the information content (variance) of the original data.

The principal components (PCs) are orthogonal, linear projections of the original variables that represent the directions of greatest variability of the data.

We can use a subset of these PCs to compress our data and view it in an alternative coordinate system.

Recap: What is PCA?

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

What are some of the benefits of these properties of PCA?

1. The PCs are orthogonal

   Regression models require independent predictors. Since the PCs are orthogonal, we can use them as predictors in a regression model rather than the original variables.

2. The PCs are linear projections

   The PCs are weighted sums of the original variables, and these weights (loadings) reveal the importance and relationship between the original data variables in different modes of variability.

3. The PCs represent the directions of greatest variability

   We can retain the first K < M PCs rather than all M variables without losing much information.

Recap: How does PCA work?

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

PCA finds linear combinations (i.e. weighted sums), U, of the original data variables, X, that "frontload" the information content in X:

$$[U] = [X][W]$$

$$\begin{bmatrix} u_{11} & u_{12} & \dots u_{1m} \\ u_{21} & u_{22} & \dots u_{2m} \\ u_{31} & u_{32} & \dots u_{3m} \\ \dots & \dots & \dots \ \dots \\ u_{n1} & u_{n2} & \dots u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots x_{1m} \\ x_{21} & x_{22} & \dots x_{2m} \\ x_{31} & x_{32} & \dots x_{3m} \\ \dots & \dots & \dots \ \dots \\ x_{n1} & x_{n2} & \dots x_{nm} \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} & \dots w_{1m} \\ w_{21} & w_{22} & \dots w_{2m} \\ w_{31} & w_{32} & \dots w_{3m} \\ \dots & \dots & \dots \ \dots \\ w_{m1} & w_{m2} & \dots w_{mm} \end{bmatrix}$$

These transformed variables, U, are called the principal components.

Recap: How does PCA work?

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

$$[U] = [X][W]$$

$$\begin{bmatrix} u_{11} & u_{12} & \dots u_{1m} \\ u_{21} & u_{22} & \dots u_{2m} \\ u_{31} & u_{32} & \dots u_{3m} \\ \dots & \dots & \dots \ \dots \\ u_{n1} & u_{n2} & \dots u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots x_{1m} \\ x_{21} & x_{22} & \dots x_{2m} \\ x_{31} & x_{32} & \dots x_{3m} \\ \dots & \dots & \dots \ \dots \\ x_{n1} & x_{n2} & \dots x_{nm} \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} & \dots w_{1m} \\ w_{21} & w_{22} & \dots w_{2m} \\ w_{31} & w_{32} & \dots w_{3m} \\ \dots & \dots & \dots \ \dots \\ w_{m1} & w_{m2} & \dots w_{mm} \end{bmatrix}$$

It turns out, the weights that sequentially maximize the variance in the PCs, conditional on being orthogonal, are the eigenvectors of the covariance matrix ([S]) of [X]:

$$[U] = [X][E]$$

$$\begin{bmatrix} u_{11} & u_{12} & \dots u_{1m} \\ u_{21} & u_{22} & \dots u_{2m} \\ u_{31} & u_{32} & \dots u_{3m} \\ \dots & \dots & \dots \ \dots \\ u_{n1} & u_{n2} & \dots u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots x_{1m} \\ x_{21} & x_{22} & \dots x_{2m} \\ x_{31} & x_{32} & \dots x_{3m} \\ \dots & \dots & \dots \ \dots \\ x_{n1} & x_{n2} & \dots x_{nm} \end{bmatrix} \begin{matrix} \overrightarrow{e_1} \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \overrightarrow{e_2} \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \overrightarrow{e_m} \\ \\ \\ \\ \\ \end{matrix} \begin{bmatrix} e_{11} & e_{12} & \dots e_{1m} \\ e_{21} & e_{22} & \dots e_{2m} \\ e_{31} & e_{32} & \dots e_{3m} \\ \dots & \dots & \dots \ \dots \\ e_{m1} & e_{m2} & \dots e_{mm} \end{bmatrix}$$

Understanding the eigenvector elements

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1m} \\ u_{21} & u_{22} & \cdots & u_{2m} \\ u_{31} & u_{32} & \cdots & u_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \cdots & u_{nm} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ x_{31} & x_{32} & \cdots & x_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1m} \\ e_{21} & e_{22} & \cdots & e_{2m} \\ e_{31} & e_{32} & \cdots & e_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m2} & \cdots & e_{mm} \end{bmatrix}$$

The elements of each eigenvector are the loadings on each variable in that PC:

e_{11} = weight on X_1 (variable 1) in 1st PC
e_{21} = weight on X_2 (variable 2) in 1st PC
e_{31} = weight on X_3 (variable 3) in 1st PC
…
e_{m1} = weight on X_m (variable m) in 1st PC

What might be a shortcoming of PCA as described so far?

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Consider a dataset with the following variables:

X_1 = People's height in meters

X_2 = People's weight in pounds

Which variable do you think would have a higher loading? What if we changed the units of X_1 to millimeters?

In the first case, X_2 would likely be more important, while in the second case X_1 would be. But in reality, the variables are the same in both cases!

Using the covariance matrix of the data for PCA will show the variables with the greatest magnitudes explain the most variability, when the magnitude depends on the data units.

Overcoming this shortcoming

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

When we perform PCA, we first "center" our data [X] by subtracting each variable's mean $\bar{X}$ from each observation to obtain $X_i{}' = X_i - \bar{X}_i$.

We can also standardize our data [X] by dividing each centered observation X' by that variable's standard deviation s to obtain $Z_i = \frac{X_i - \bar{X}_i}{s(X_i)}$.

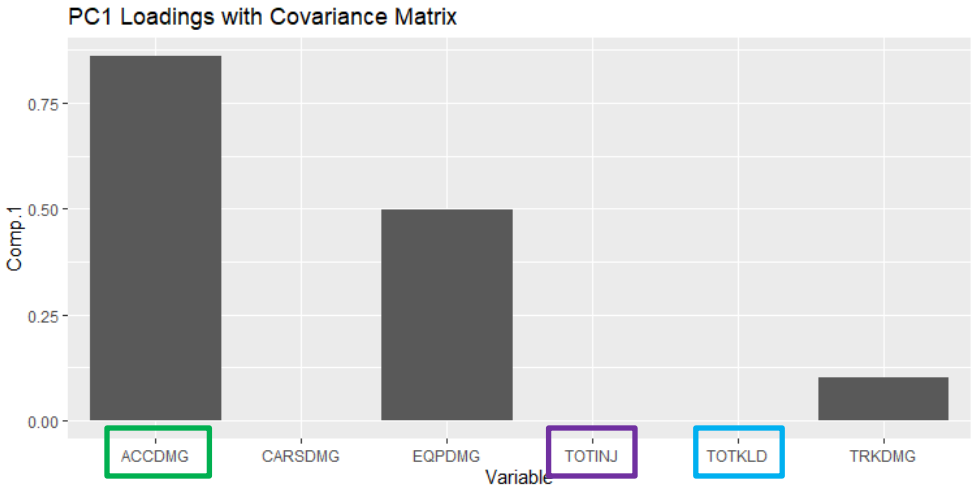
Performing PCA using the covariance matrix of [Z] is equivalent to performing PCA using the correlation matrix of [X], i.e. our weight matrix is the eigenvector matrix of the correlation matrix [R] instead of the covariance matrix [S].

```
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG", "TOTKLD", "TOTINJ")])
> xdmgnd.pca <- princomp(xdmgnd[,c("CARSDMG","EQPDMG","TRKDMG",
+                                  "ACCDMG", "TOTKLD", "TOTINJ")],cor=T)
```

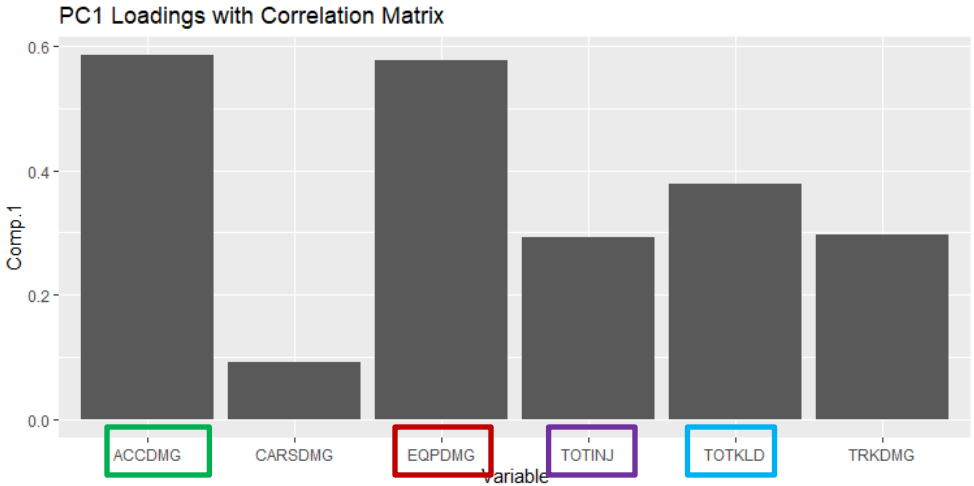
PCA w/ the correlation vs. covariance matrix

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary



Using the covariance matrix, ACCDMG dominates everything because it encompasses the other damages. TOTKLD and TOTINJ are extremely small because they're different units ($ vs. people)

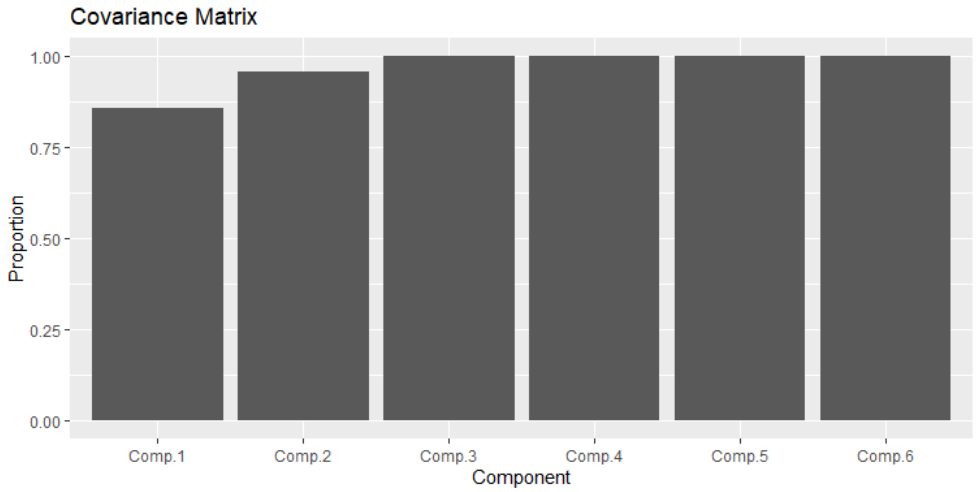


Using the correlation matrix, TOTKLD and TOTINJ become more significant. And EQPDMG is almost as important as ACCDMG, suggesting that is most of the damages.

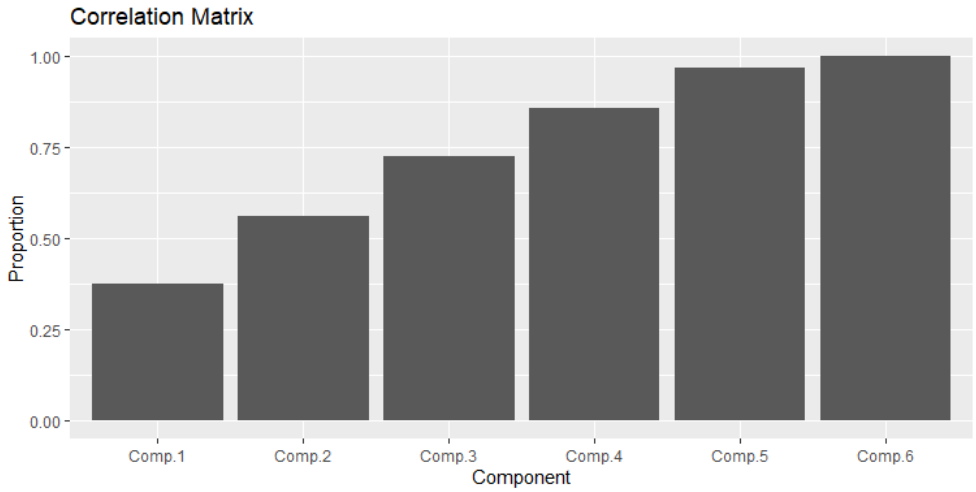
PCA w/ the correlation vs. covariance matrix

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary



Using the covariance matrix, almost all of the information is explained by the first PC, which is dominated by ACCDMG.



Using the correlation matrix, several PCs are needed to explain the variability, as TOTKLD and TOTINJ are more important, and vary in different ways than ACCDMG.

PCA w/ covariance vs. correlation matrix

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

If multiple variables with widely different magnitudes are used for PCA, it is generally best to use the correlation matrix to remove this influence.

But if the data consists of the same variable at multiple locations, the covariance matrix might be preferred even if the magnitudes across locations vary.

Examples of data like this include gridded spatial data and images.

How do you perform PCA on gridded data?

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Our data, X, are often composed of a time series of multiple variables, e.g. temperature at multiple locations:

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} \cdots & x_{1m} \\ x_{21} & x_{22} & x_{23} \cdots & x_{2m} \\ x_{31} & x_{32} & x_{33} \cdots & x_{3m} \\ \cdots & \cdots & \cdots \cdots & \cdots \\ x_{n1} & x_{n2} & x_{n3} \cdots & x_{nm} \end{bmatrix}$$

Pixel 1 Pixel 2 Pixel 3 … Pixel *m*

Year 1
Year 2
Year 3
…
Year *n*

Instead of storing *M* variables, can we reduce our data to *K* transformed variables with almost as much information?

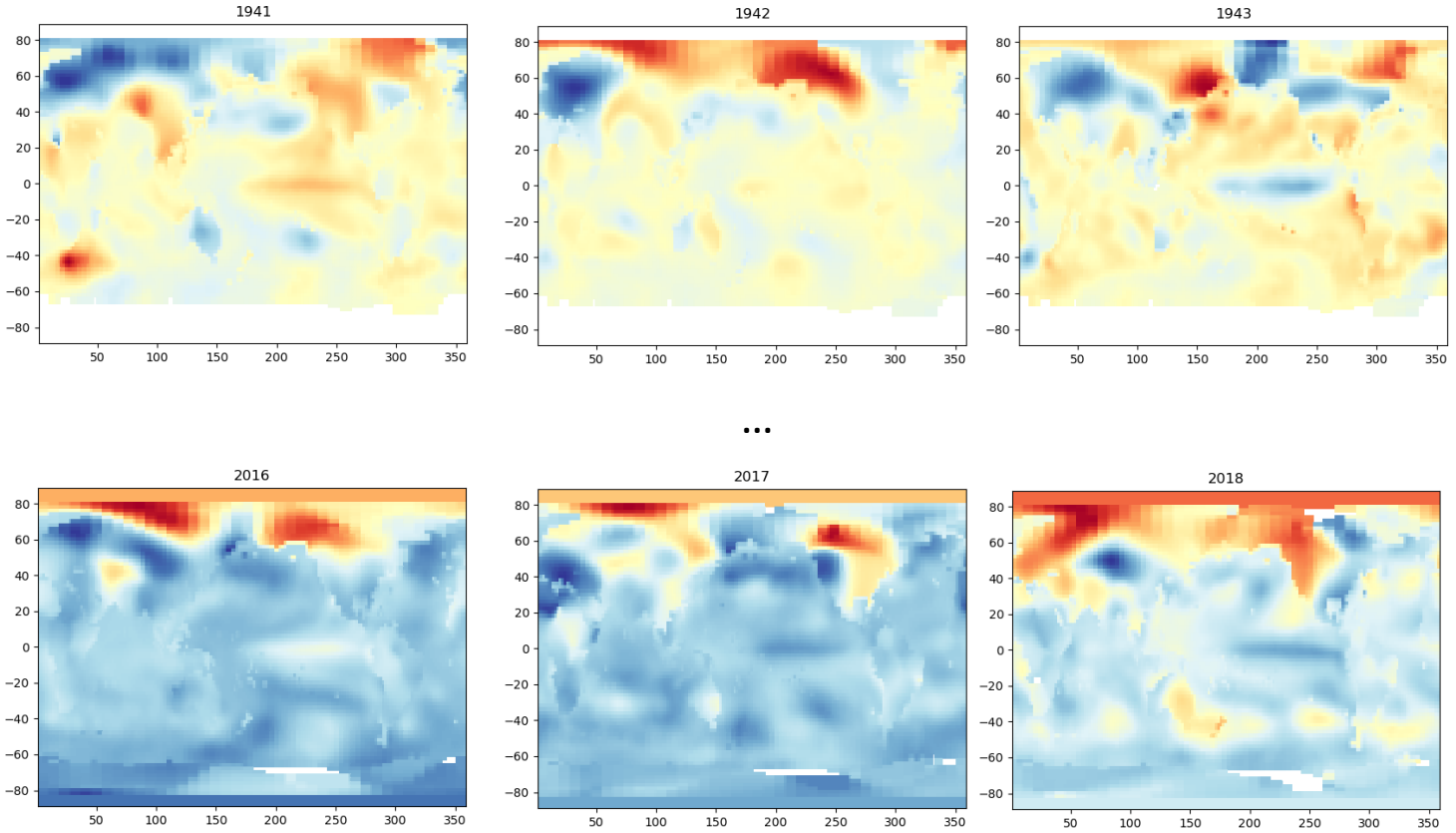
And are there common patterns of spatial variability in our data?

PCA on global temperature data

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Are there common patterns of global variability in 78 years of annual temperature anomalies at 13,253 locations?



PCA on gridded data

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

$$[U] = [X][E]$$

[X] is an (N x M) matrix (M variables with N observations)

Normally:

[E] is an (M x M) matrix (M eigenvectors with M variable weights)

[U] is an (N x M) matrix (M PCs with N observations)

We can only find [E] by eigen-decomposition if N > M. But here N=78 years and M=13,253 grid cells. This is often a problem with gridded data.

PCA when N < M

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

When N < M, we need to use singular value decomposition (SVD). This uses a different function in R (`prcomp` instead of `princomp`).

With SVD, instead of getting M PCs, we will only get N PCs, so [E] won't be an M x M matrix of eigenvectors, but an M x N matrix [W] of N singular values, each with M variable weights:

$$[U] = [X][W]$$

[X] is an (N x M) matrix (M variables with N observations)
[U] is an (N x N) matrix (N PCs with N observations)
[W] is an (M x N) matrix (N singular values with M variable weights)

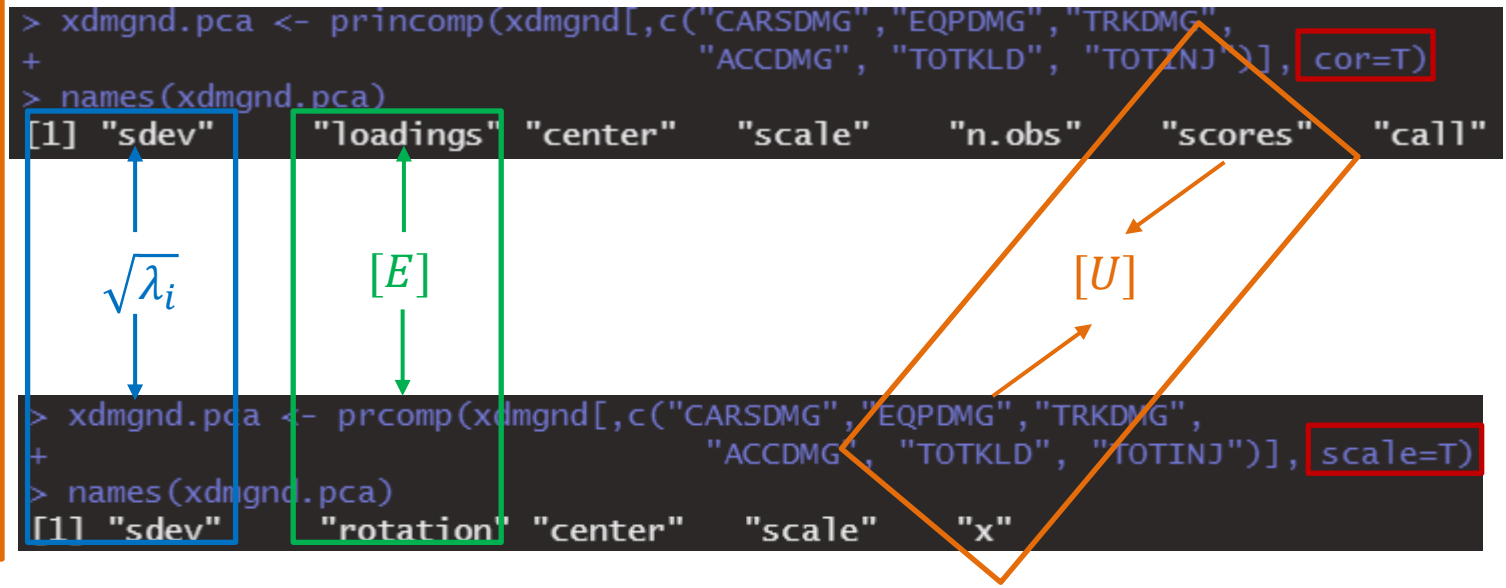
`prcomp` vs. `princomp`

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

If M > N, we need to use `prcomp` with SVD instead of `princomp` with eigen decomposition.

If M < N, we will get the same things: the singular values will be the same as the eigenvalues.



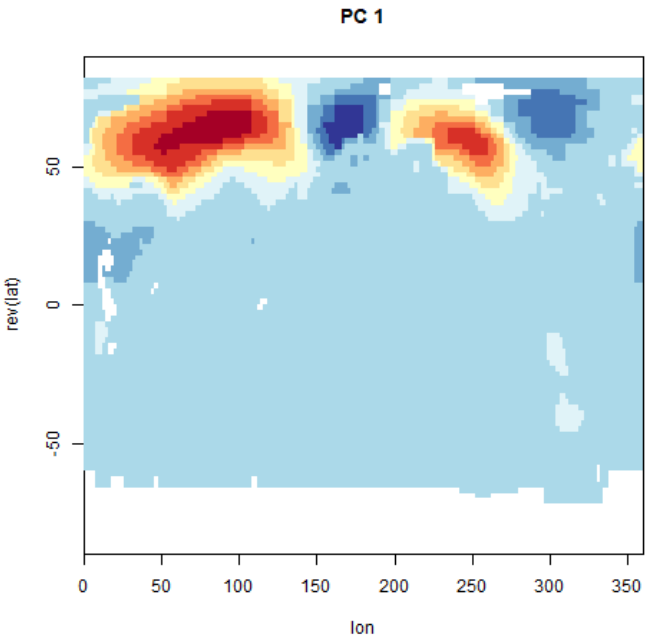
PCA loadings on a map

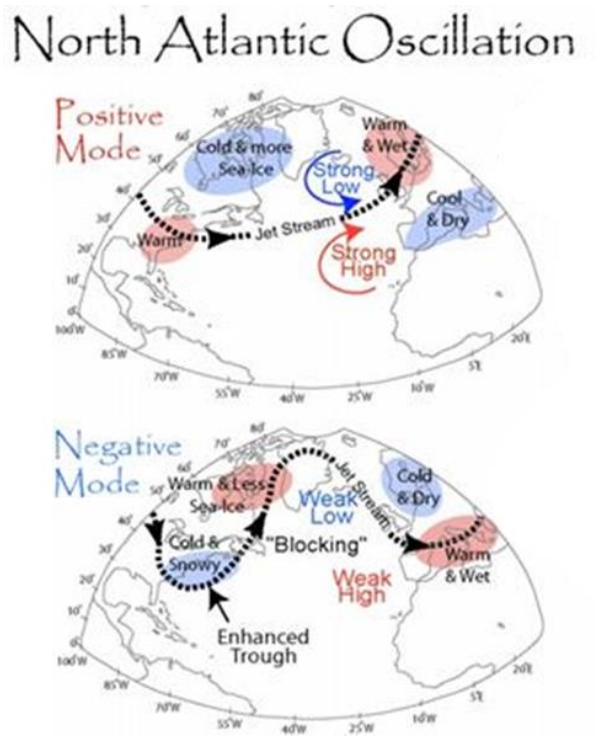
Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Since each variable is a grid cell, for each singular value, we can plot the weights/loadings associated with each grid cell on a map.

Here, this reveals common patterns of global variability in annual temperature anomalies.





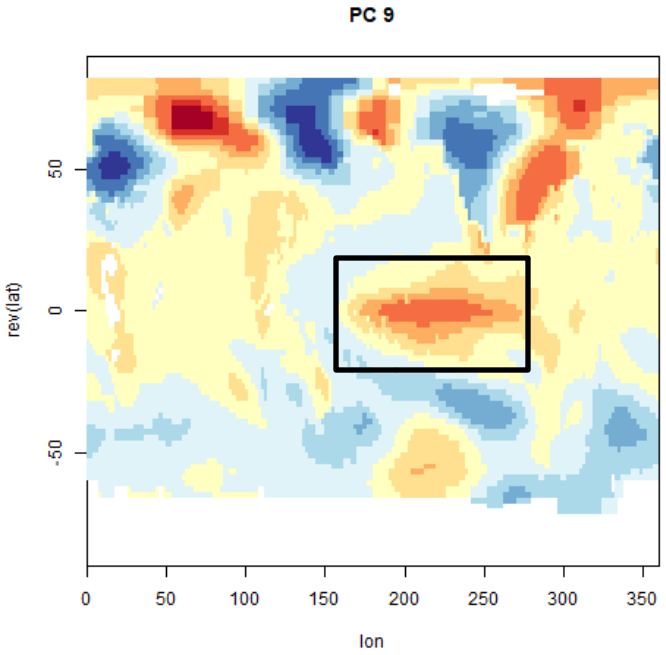
PCA loadings on a map

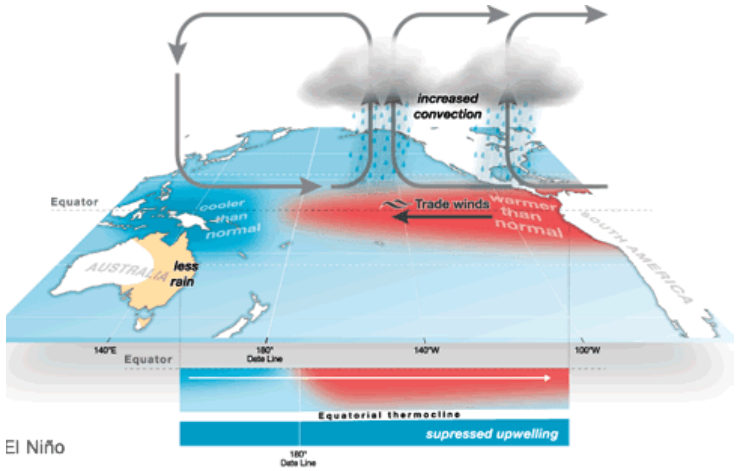
Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Since each variable is a grid cell, for each singular value, we can plot the weights/loadings associated with each grid cell on a map.

Here, this reveals common patterns of global variability in annual temperature anomalies.





El Nino Southern Oscillations

Analyzing images with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

We can apply the same approach to analyze image data, like faces. In this case, our data, X, are composed of RGB values in different pixels across multiple images (such as faces).

If it's in black and white, you have one RGB color (variable) per pixel:

$$X = \begin{array}{c} \begin{matrix} \text{Pixel 1} & \text{Pixel 2} & \text{Pixel 3} & \dots & \text{Pixel } m \end{matrix} \\ \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1m} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2m} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3m} \\ \dots & \dots & \dots & \cdots & \dots \\ x_{n1} & x_{n2} & x_{n3} & \cdots & x_{nm} \end{bmatrix} \end{array} \begin{matrix} \\ \text{Image 1} \\ \text{Image 2} \\ \text{Image 3} \\ \dots \\ \text{Image } n \end{matrix}$$

Analyzing images with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

We can apply the same approach to analyze image data, like faces. In this case, our data, X, are composed of RGB values in different pixels across multiple images (such as faces).

If it's in color, you have three RGB colors (variables) per pixel:

Pixel 1R Pixel 1G Pixel 1B … Pixel (*m/3)B*

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} \cdots & x_{1m} \\ x_{21} & x_{22} & x_{23} \cdots & x_{2m} \\ x_{31} & x_{32} & x_{33} \cdots & x_{3m} \\ \cdots & \cdots & \cdots \cdots & \cdots \\ x_{n1} & x_{n2} & x_{n3} \cdots & x_{nm} \end{bmatrix}$$

Image 1
Image 2
Image 3
…
Image *n*

Can PCA compress our data in a better way than coarsening the pixel resolution? Can we use this for facial recognition?

Facial image compression with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Consider N=1000 faces like those at the right.

Each is composed of 64x64 pixels = 4096 variables (M).

Can we store approximations of these faces with K<N PCs?

https://towardsdatascience.com/eigenfaces-recovering-humans-from-ghosts-17606c328184

Loadings when using PCA on images

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

Again, for each singular value, we can plot the loadings associated with each of our pixels at that pixel location. These are called "eigenfaces."

Image compression with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

We can reconstruct the faces with a subset K < N of the PCs and singular values. This is what it looks like for K=50:

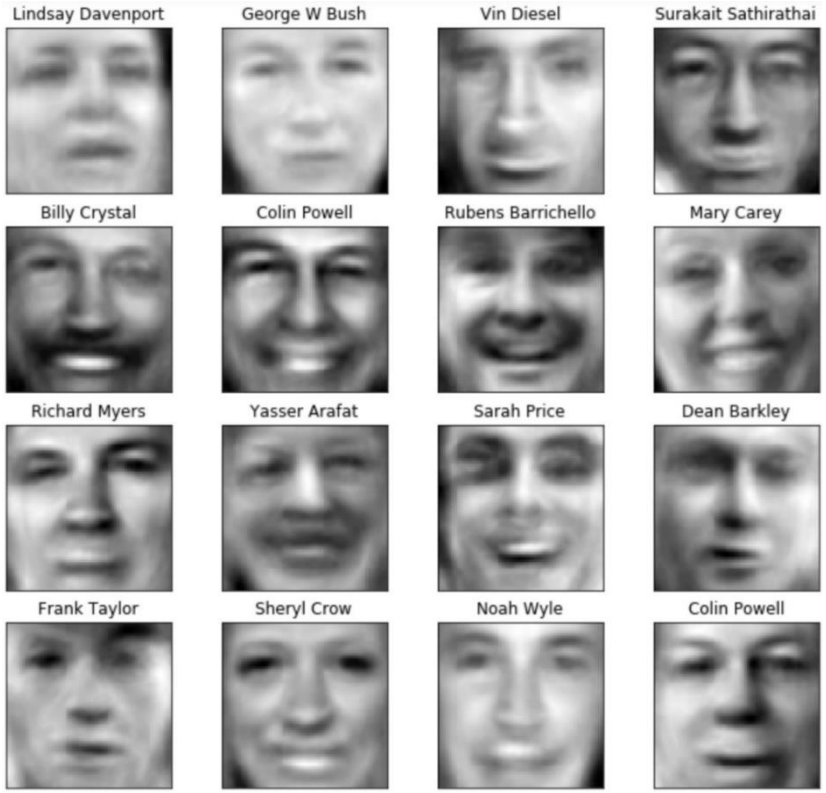


Image compression with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

We can reconstruct the faces with a subset K < N of the PCs and singular values. This is what it looks like for K=100:

Image compression with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

We can reconstruct the faces with a subset K < N of the PCs and singular values. This is what it looks like for K=250:

With only 250 PCs (1/4 of the 1000), we can reproduce the images quite well!

Facial recognition with PCA

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

PCA can be used as part of facial recognition algorithms as well, not just image compression.

Just as we can use the PCs as predictors in a regression, we can use them as predictors in classifying faces.

In the above example with images of different people, we would be using it to identify a face from other objects.

If we wanted to recognize one face as opposed to other faces, we would use multiple images of the same face.

Summary

Agenda

- Recap
- Potential shortcomings and an alternative
- PCA on gridded data
- Summary

1. PCA is a useful tool for visualization and data compression, especially for geographic and image data.

2. But we should be careful about how we perform it.
   a) If our data variables are all different magnitudes and units, it is best to use the correlation matrix.
   b) If our data variables are different magnitudes but of the same variable at different locations/pixels, use the covariance matrix.

3. If we have more variables than observations, we need to use singular value decomposition rather than eigen decomposition.