Business Analytics
High-Dimensional Problems

Prof. Bichler

Decision Sciences & Systems

Department of Informatics

Technical University of Munich

Course Content

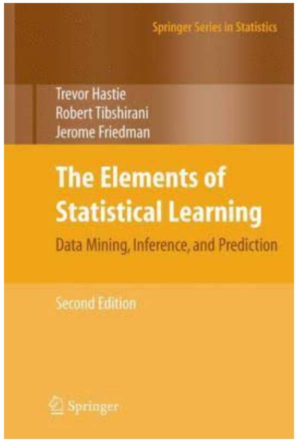
- Introduction
- Regression Analysis
- Regression Diagnostics
- Logistic and Poisson Regression
- Naive Bayes and Bayesian Networks
- Decision Tree Classifiers
- Data Preparation and Causal Inference
- Model Selection and Learning Theory
- Ensemble Methods and Clustering
- **High-Dimensional Problems**
- Association Rules and Recommenders
- Neural Networks

Primary Literature

- **The Elements of Statistical Learning**
  - Trevor Hastie, Robert Tibshirani, Jerome Friedman
  - https://web.stanford.edu/~hastie/ElemStatLearn/
  - Section: 3.4 – 3.6

- **An Introduction to Statistical Learning: With Applications in R**
  - Gareth James, Trevor Hastie, Robert Tibshirani, 2014
  - http://www-bcf.usc.edu/~gareth/ISL/
  - Section: 6, 10.2

  - https://www.cs.princeton.edu/picasso/mats/PCA-Tutorial-Intuition_jp.pdf



Springer Series in Statistics

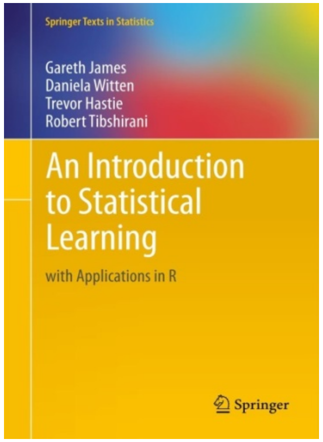
Trevor Hastie
Robert Tibshirani
Jerome Friedman

The Elements of Statistical Learning

Data Mining, Inference, and Prediction

Second Edition

Springer



Springer Texts in Statistics

Gareth James
Daniela Witten
Trevor Hastie
Robert Tibshirani

An Introduction to Statistical Learning

with Applications in R

Springer

Outline for today

- **Overview**
- Linear algebra revisited
- Principal Component Analysis
- Singular Value Decomposition
- PCA regression and regularization

Principal Component Analysis (PCA)

- Principal component analysis (PCA) converts a set of possibly correlated variables into a (possibly smaller) set of values of linearly uncorrelated variables called principal components.

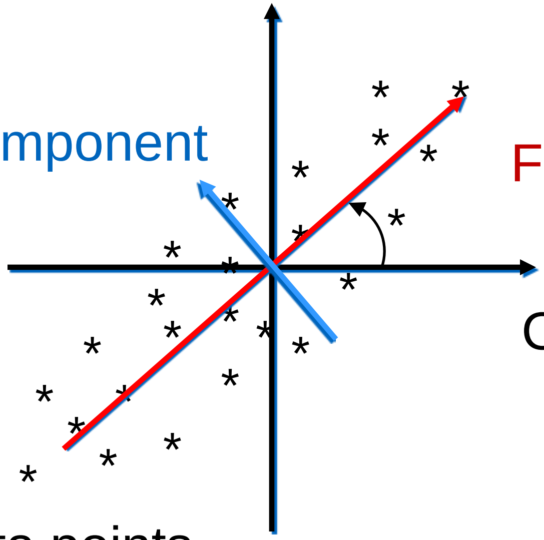
- The first principal component has the largest possible variance, and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to (i.e., uncorrelated with) the preceeding components.

- The principal components are orthogonal (they are the Eigenvectors of the symmetric covariance matrix).

- PCA is sometimes referred to as the Karhunen–Loève transform (KLT) and related to singular value decomposition (SVD).

Principal Component Analysis

Suppose we have a population measured on p random variables $X_1, \ldots, X_p$. Note that these random variables represent the p axes of the Cartesian coordinate system in which the population resides. Our goal is to develop a new set of $k \leq p$ axes (linear combinations of the original p axes) in the directions of greatest variability. This is accomplished by rotating the axes.



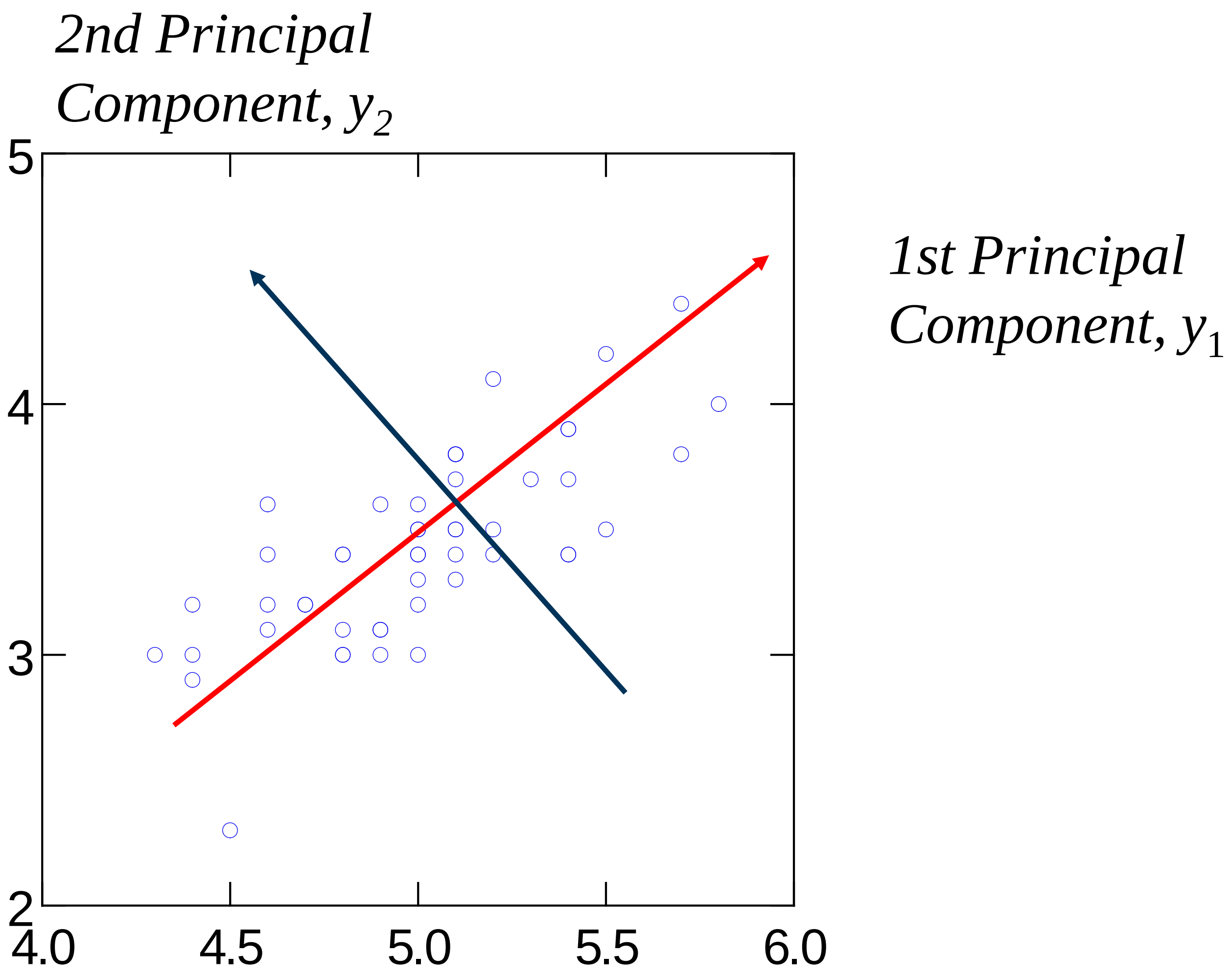
Second principal component

First principal component

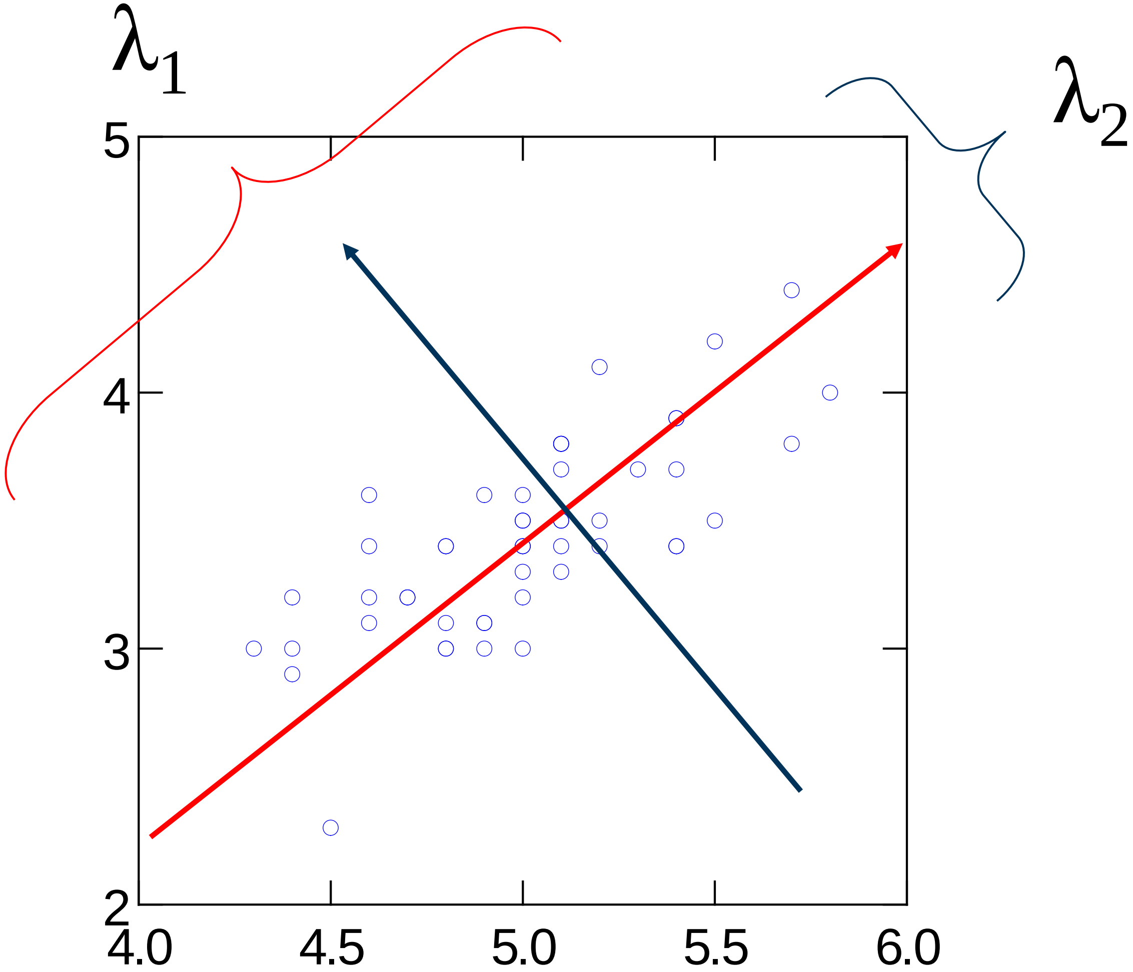
Original axes

Data points

Principal Components are Eigenvectors

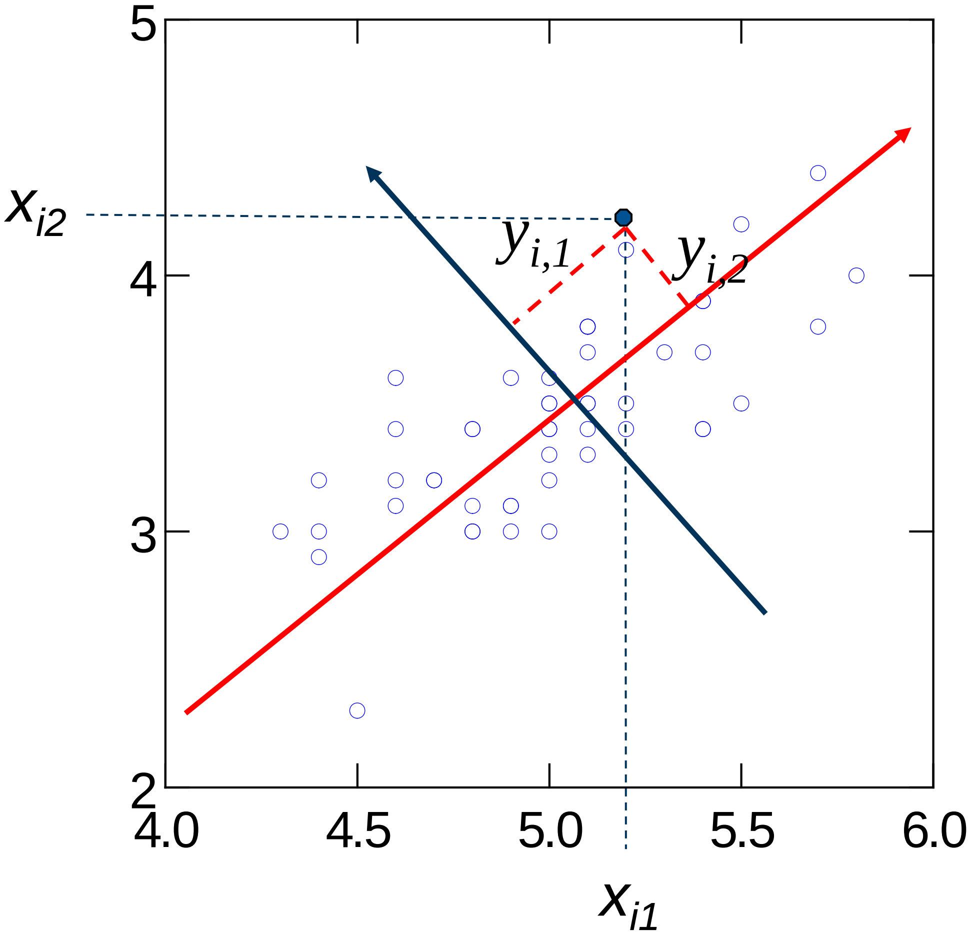


PCA Eigenvalues



Eigenvalues λ_1 explain the proportion of variance explained by PC1.

PCA Scores



The PCA score for any of the x is just it's coefficient in each of the y.

PCA

From p original variables: $x_1, x_2, \ldots, x_p$:
Produce k (or less) new variables: $y_1, y_2, \ldots, y_k$ as linear combinations of the original variables x_i.

$$\left.\begin{aligned} y_1 &= a_{11}x_1 + a_{12}x_2 + \ldots + a_{1p}x_p \\ y_2 &= a_{21}x_1 + a_{22}x_2 + \ldots + a_{2p}x_p \\ &\ldots \\ y_k &= a_{k1}x_1 + a_{k2}x_2 + \ldots + a_{kp}x_p \end{aligned}\right\} \quad \begin{array}{c} y_k\text{'s are the} \\ \text{Principal Components} \end{array}$$

such that:

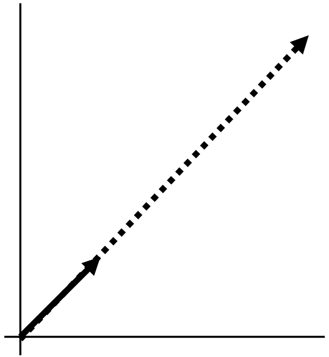
y_k's are uncorrelated (orthogonal)
y_1 explains as much as possible of original variance in data
y_2 explains as much as possible of remaining variance
etc.

Outline for today

- Overview
- **Linear algebra revisited**
- Principal Component Analysis
- Singular Value Decomposition
- PCA regression and regularization

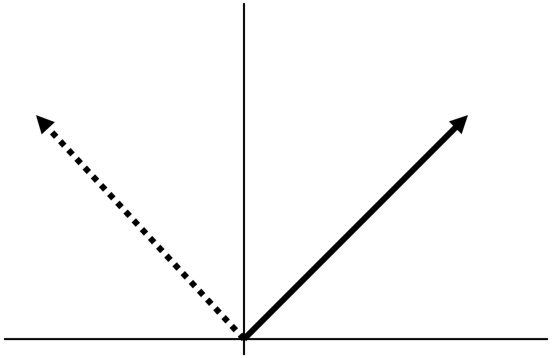
Matrices as Linear Transformations

$$\begin{pmatrix} 5 & 0 \\ 0 & 5 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$



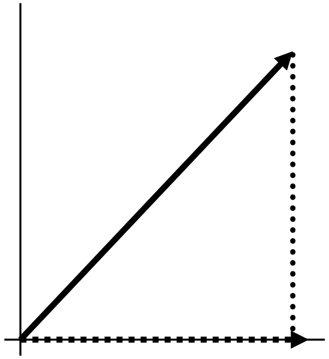
(stretching)

$$\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$



(rotation)

$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

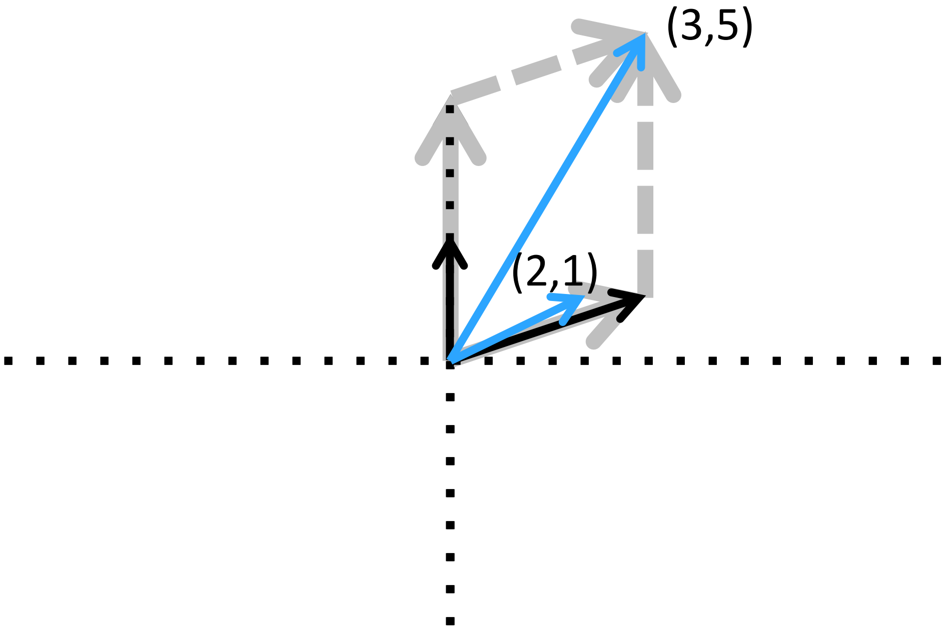


(projection)

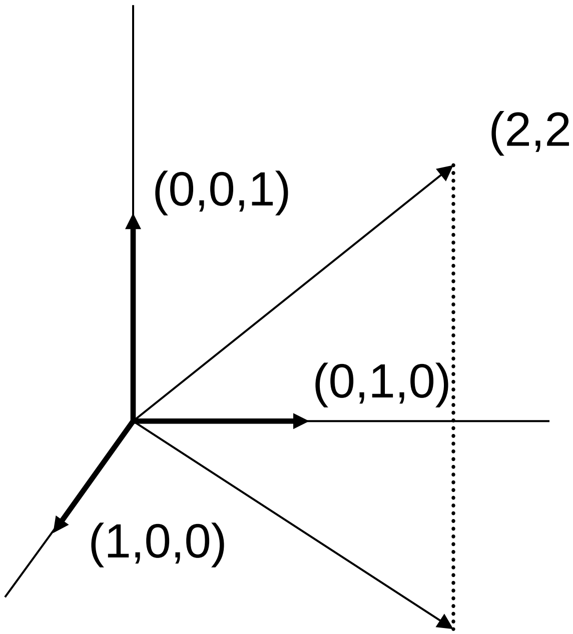
Rotation and Scaling

$$\begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 3 \\ 5 \end{pmatrix}$$

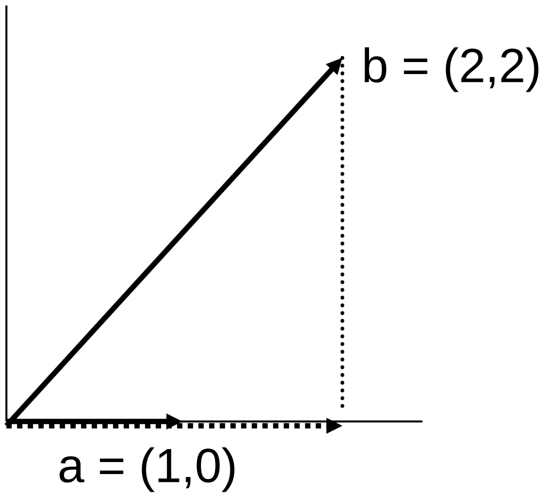
The new vector is rotatet and scaled!



Projections



$$\begin{pmatrix} 2 \\ 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}$$



In general, to project b onto some vector a:

$$c = \frac{a^T b}{a^T a} a = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$$

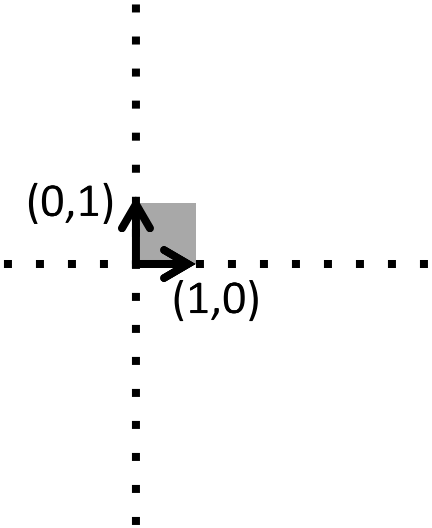
Determinant

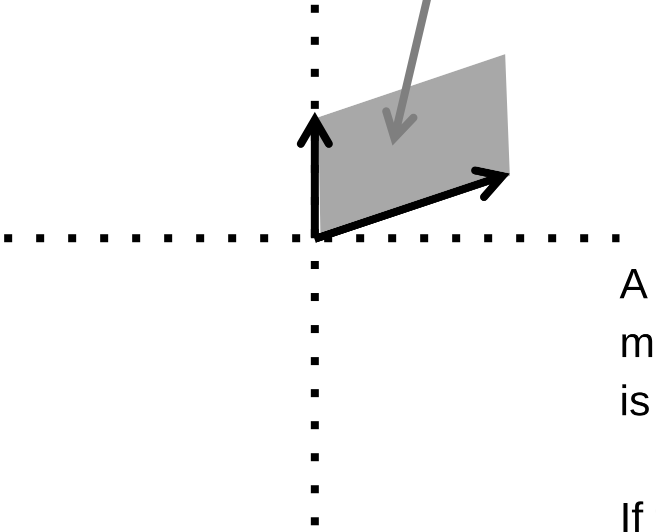
Geometrically, a determinant can be viewed as the volume **scaling** factor of the linear transformation described by the **matrix**.

$$Area = \left| det(\overleftrightarrow{M}) \right|$$

$$\begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix}$$

$$\left| det \begin{pmatrix} a & b \\ c & d \end{pmatrix} \right| = |ad - bc|$$





$$det \begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix} = -6$$

A negative determinant means that the orientation is flipped.

If the determinant is zero, ther volume is zero!

Eigenvalues and Eigenvectors

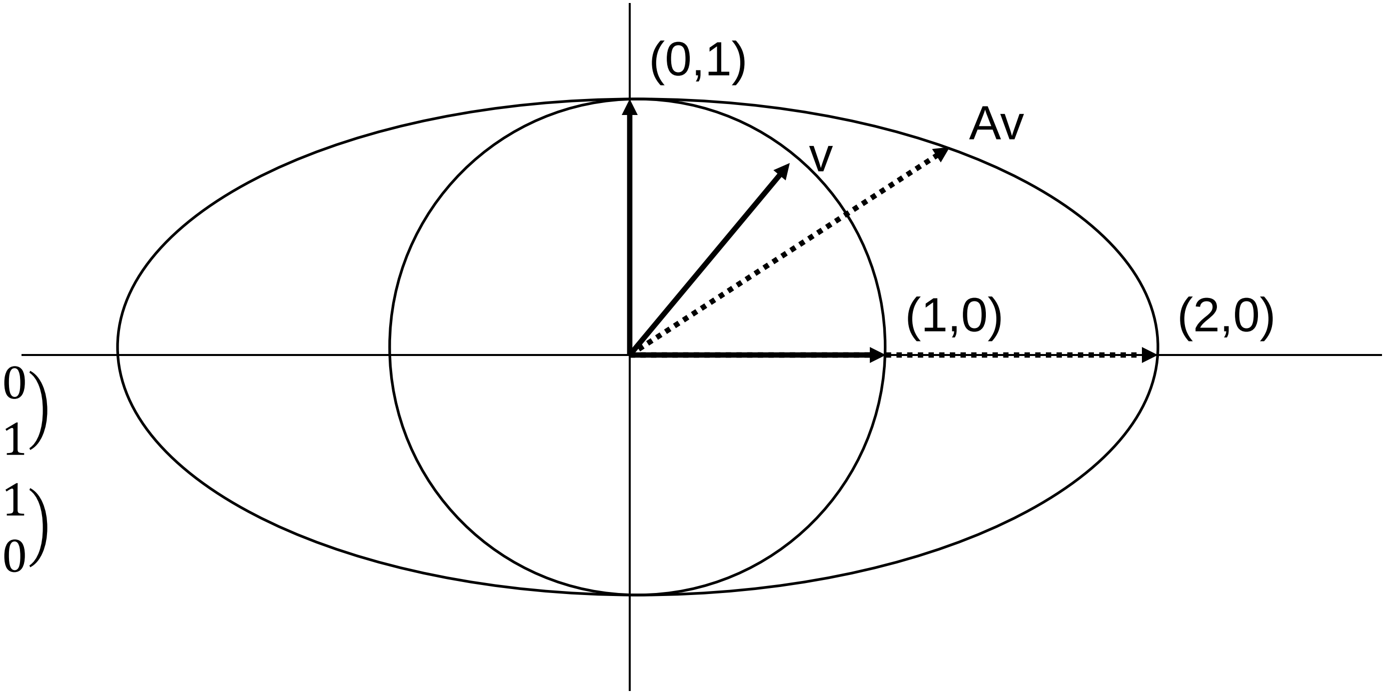
Eigenvalues λ = 2, 1 with
Eigenvectors (1,0), (0,1)

Eigenvectors of a linear transformation $\boldsymbol{A}$ are not rotated (but will be scaled by the corresponding Eigenvalue) when $\boldsymbol{A}$ is applied.

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = 1\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = 2\begin{pmatrix} 1 \\ 0 \end{pmatrix}$$



Eigenvalues and Eigenvectors

Let $\boldsymbol{A}$ be an $p * p$ matrix with Eigenvalue λ and corresponding Eigenvector v. Thus $\mathbf{A}v \ = \ \lambda v$. This equation may be written

$$\mathbf{A}v - \ \lambda v \ = \ 0$$

given

$$\left(\mathbf{A} - \lambda\mathbf{I}_p\right)v \ = \ 0$$

- Solving the equation $|\mathbf{A} – \lambda\mathbf{I}_p| = 0$ for λ leads to all the Eigenvalues of **A**
    - A determinant can be viewed as the volume scaling factor of the linear transformation described by the matrix!
    - If the determinant is 0, then the space described by the transformation via **A** is 0, i.e. we get a scalar λ that scales the vector.
- On expending the determinant $|\mathbf{A} – \lambda\mathbf{I}_p|$, we get a polynomial in λ
- This polynomial is called the **characteristic polynomial** of **A**
- The equation $|\mathbf{A} – \lambda\mathbf{I}_p| = 0$ is called the **characteristic equation** of **A**

Eigenvalues and Eigenvectors

$\mathbf{A} = \begin{bmatrix} -4 & -6 \\ 3 & 5 \end{bmatrix}$

- For the equation $|A - \lambda I_p| = 0$ let us first derive the characteristic polynomial of $\boldsymbol{A}$
- We get:

$$\mathbf{A} - \lambda \mathbf{I}_2 = \begin{bmatrix} -4 & -6 \\ 3 & 5 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -4-\lambda & -6 \\ 3 & 5-\lambda \end{bmatrix}$$

$$|\mathbf{A} - \lambda \mathbf{I}_2| = (-4-\lambda)(5-\lambda) + 18 = \lambda^2 - \lambda - 2$$

- We now solve the characteristic equation of **A**

$$\lambda^2 - \lambda - 2 = 0 \Rightarrow (\lambda - 2)(\lambda + 1) = 0 \Rightarrow \lambda = 2 \; or \; -1$$

- The Eigenvalues of $\boldsymbol{A}$ are 2 and –1
- The corresponding Eigenvectors are found by using these values of λ in the equation $(\mathbf{A} - \lambda \mathbf{I}_2)v = 0$

Reminder: Eigenvectors for λ = 2

We solve the equation $(\mathbf{A} - 2\mathbf{I}_2)v = 0$ for v. The matrix $(\mathbf{A} - 2\mathbf{I}_2)$ is obtained by subtracting 2 from the diagonal elements of **A.**

- We get:

$$\begin{bmatrix} -6 & -6 \\ 3 & 3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = 0$$

- This leads to the system of equations:

$$\begin{gathered} -6v_1 - 6v_2 = 0 \\ 3v_1 + 3v_2 = 0 \end{gathered}$$

giving $v_1 = -v_2$. The solutions to this system of equations are $v_1 = -r, v_2 = r$, where *r* is a scalar. Thus the Eigenvectors of **A** corresponding to λ = 2 are nonzero vectors of the form

$$r \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Eigenvectors for λ = –1

We solve the equation $(\mathbf{A} + 1\mathbf{I}_2)v = 0$ for v. The matrix $(\mathbf{A} + 1\mathbf{I}_2)$ is obtained by adding 1 to the diagonal elements of **A.**

We get:

$$\begin{bmatrix} -3 & -6 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = 0$$

This leads to the system of equations

$$-3v_1 - 6v_2 = 0$$
$$3v_1 + 6v_2 = 0$$

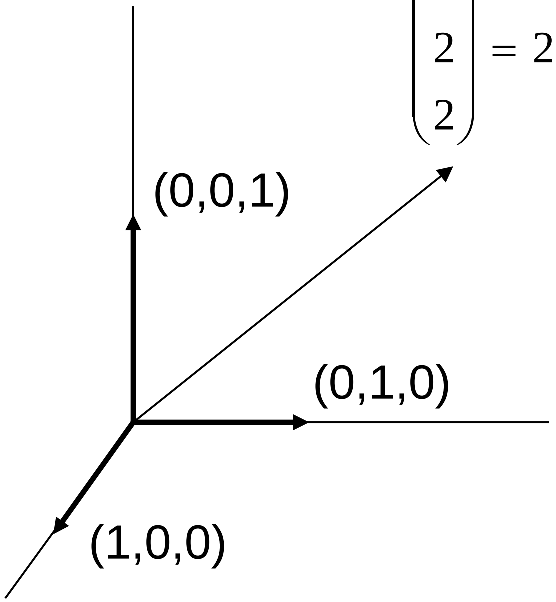
Thus $v_1 = -2v_2$. The solutions to this system of equations are $v_1 = -2s$ and $v_2 = s$, where s is a scalar. Thus the Eigenvectors of **A** corresponding to λ = –1 are nonzero vectors of the form

$$s\begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

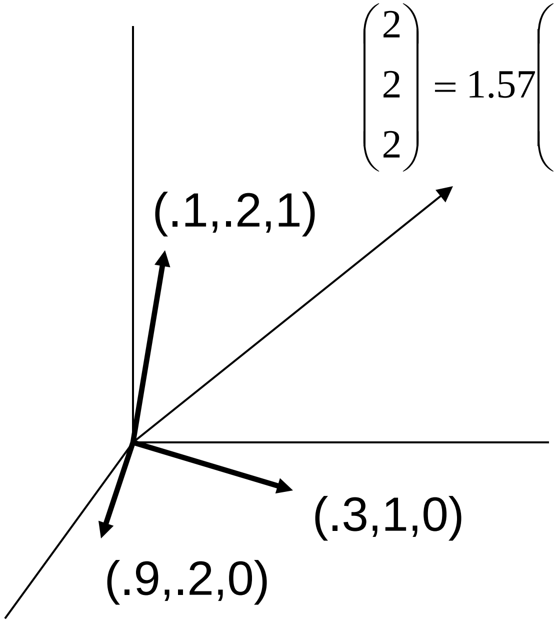
Linear Independence and Basis

If all vectors in a vector space may be expressed as linear combinations of $v_1,\ldots,v_k$, then $v_1,\ldots,v_k$ *span* the space.

$$\begin{pmatrix}2\\2\\2\end{pmatrix} = 2\begin{pmatrix}1\\0\\0\end{pmatrix} + 2\begin{pmatrix}0\\1\\0\end{pmatrix} + 2\begin{pmatrix}0\\0\\1\end{pmatrix}$$



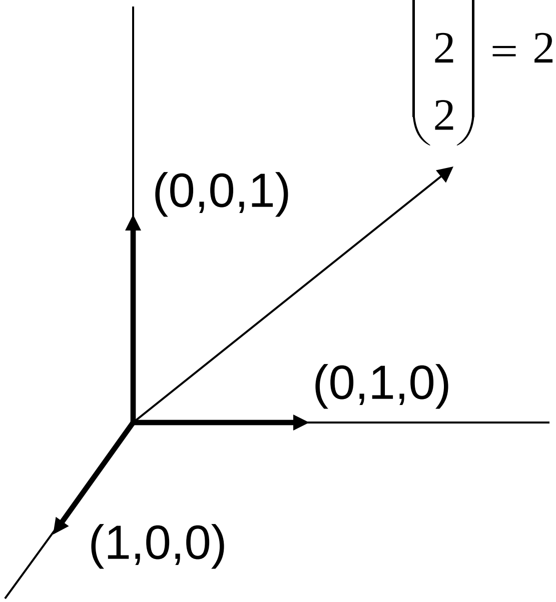
$$\begin{pmatrix}2\\2\\2\end{pmatrix} = 1.57\begin{pmatrix}.9\\.2\\0\end{pmatrix} + 1.29\begin{pmatrix}.3\\1\\0\end{pmatrix} + 2\begin{pmatrix}.1\\.2\\1\end{pmatrix}$$



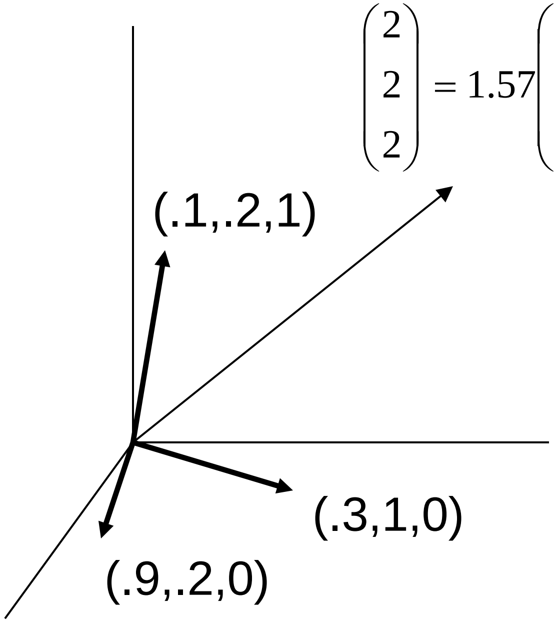
Linear Independence and Basis

A basis is a set of linearly independent vectors which span the space.
A basis is a maximal set of *linearly independent* (not necessarily orthogonal) vectors and a minimal set of spanning vectors.

$$\begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix} = 2\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + 2\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + 2\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$



$$\begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix} = 1.57\begin{pmatrix} .9 \\ .2 \\ 0 \end{pmatrix} + 1.29\begin{pmatrix} .3 \\ 1 \\ 0 \end{pmatrix} + 2\begin{pmatrix} .1 \\ .2 \\ 1 \end{pmatrix}$$



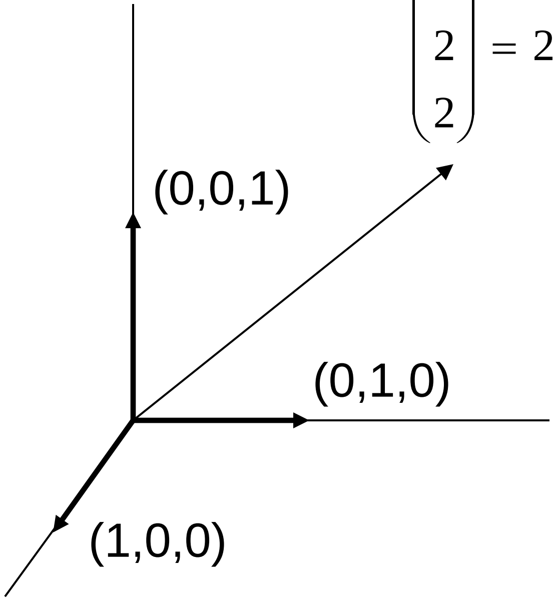
Linear Independence and Basis

Two vectors are orthogonal if their dot product is 0.

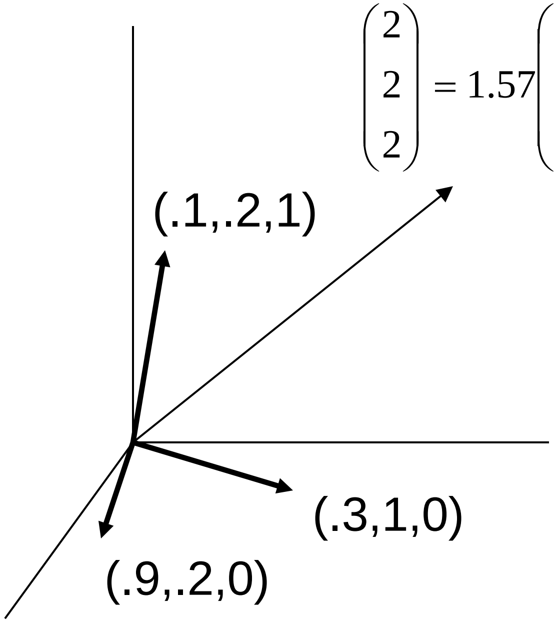
An orthogonal basis consists of orthogonal vectors.

An orthonormal basis consists of orthogonal vectors of unit length.

$$\begin{pmatrix}2\\2\\2\end{pmatrix} = 2\begin{pmatrix}1\\0\\0\end{pmatrix} + 2\begin{pmatrix}0\\1\\0\end{pmatrix} + 2\begin{pmatrix}0\\0\\1\end{pmatrix}$$



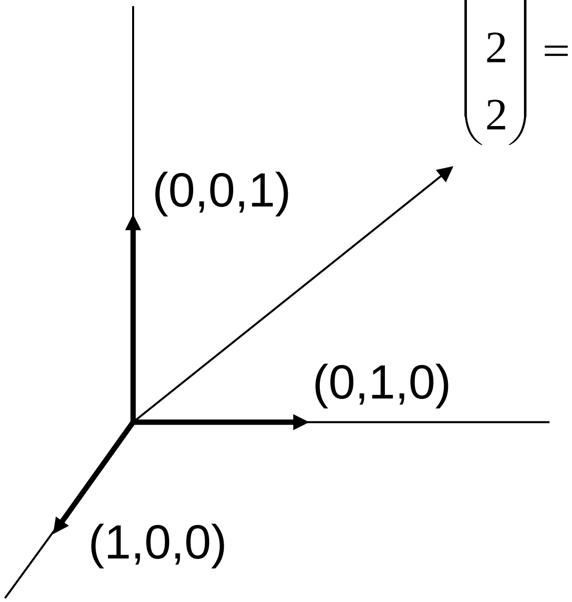
$$\begin{pmatrix}2\\2\\2\end{pmatrix} = 1.57\begin{pmatrix}.9\\.2\\0\end{pmatrix} + 1.29\begin{pmatrix}.3\\1\\0\end{pmatrix} + 2\begin{pmatrix}.1\\.2\\1\end{pmatrix}$$



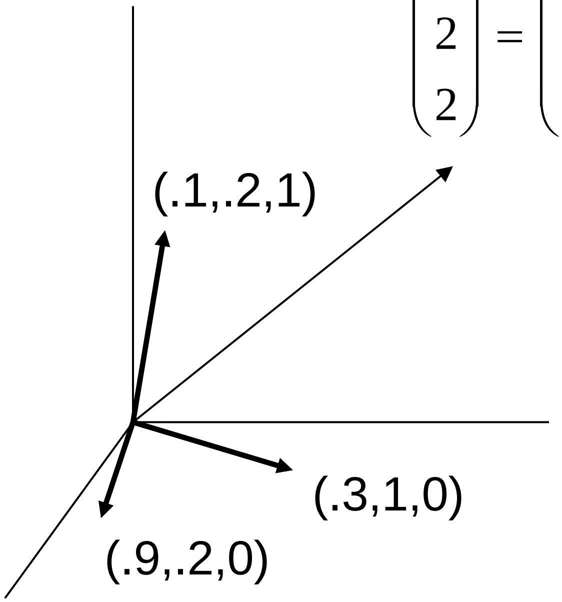
Basis Transformations

We may write v=(2,2,2) in terms of an alternate basis:

$$\begin{pmatrix}2\\2\\2\end{pmatrix} = \begin{pmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{pmatrix}\begin{pmatrix}2\\2\\2\end{pmatrix}$$

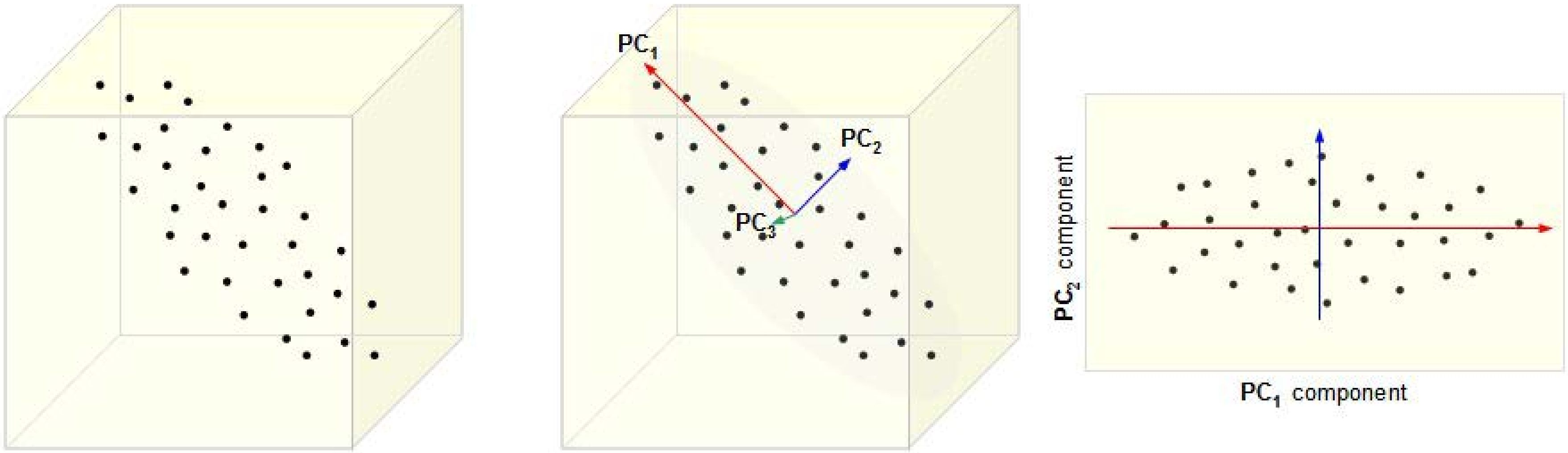


$$\begin{pmatrix}2\\2\\2\end{pmatrix} = \begin{pmatrix}.9 & .3 & .1\\.2 & 1 & .2\\0 & 0 & 1\end{pmatrix}\begin{pmatrix}1.57\\1.29\\2\end{pmatrix}$$



Components of (1.57,1.29,2) are projections of v onto new basis vectors, normalized so that the new v still has same length.

Principal Components in 3 Dimensions



- Principal components are the eigenvectors of the covariance or correlation matrix.
- The Eigenvalue λ explains the amount of variance along that axis, and the proportion of the overall variance explained by the PC.
- So **PCA** simply takes points expressed in the standard **basis** and transforms them into points expressed in an eigenvector **basis**.

Can you explain the Eigenvectors and Eigenvalues geometrically?

Outline for today

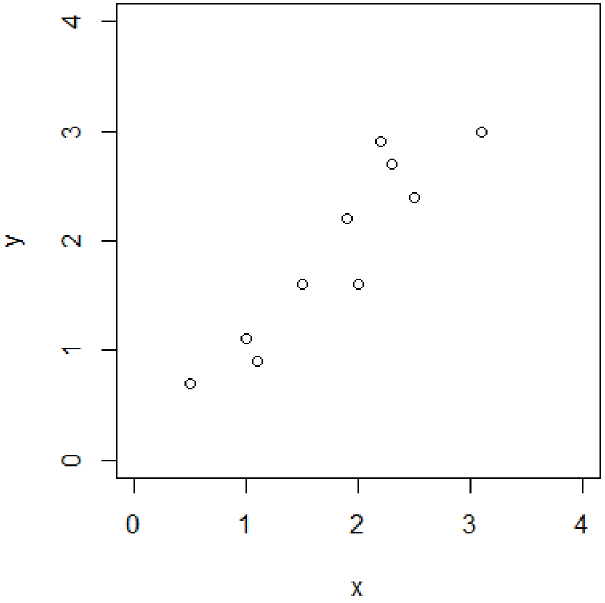
- Overview
- Linear algebra revisited
- **Principal Component Analysis**
- Singular Value Decomposition
- PCA regression and regularization

PCA Example –STEP 1

Center data by subtracting the mean as we are rotating axes around the origin later on.

Data (A):	
x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

Zero mean Data (X)	
x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01



PCA Example –STEP 2

- Calculate the covariance matrix, which summarizes the relationship between variables

$$cov = \sum = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

- Since the non-diagonal elements in this covariance matrix are positive, we should expect that both the x and y variable increase together

Variance-Covariance vs. Correlation Matrix

$$\Sigma = \begin{bmatrix} \mathrm{E}[(X_1-\mu_1)(X_1-\mu_1)] & \mathrm{E}[(X_1-\mu_1)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_1-\mu_1)(X_n-\mu_n)] \\ \mathrm{E}[(X_2-\mu_2)(X_1-\mu_1)] & \mathrm{E}[(X_2-\mu_2)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_2-\mu_2)(X_n-\mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}[(X_n-\mu_n)(X_1-\mu_1)] & \mathrm{E}[(X_n-\mu_n)(X_2-\mu_2)] & \cdots & \mathrm{E}[(X_n-\mu_n)(X_n-\mu_n)] \end{bmatrix}.$$

The **covariance matrix** centers each variable on the mean, but the scale matters. Should be used only, when the variables are measured in comparable units and the differences in variance are important for interpretation.
A covariance matrix implicitly involves centering of the data already.

If variables are measured in different units use the **correlation matrix**:

$$\rho = corr(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y}$$

Use correlation if differences in variance across the variables are not meaningful.

PCA Example –STEP 3

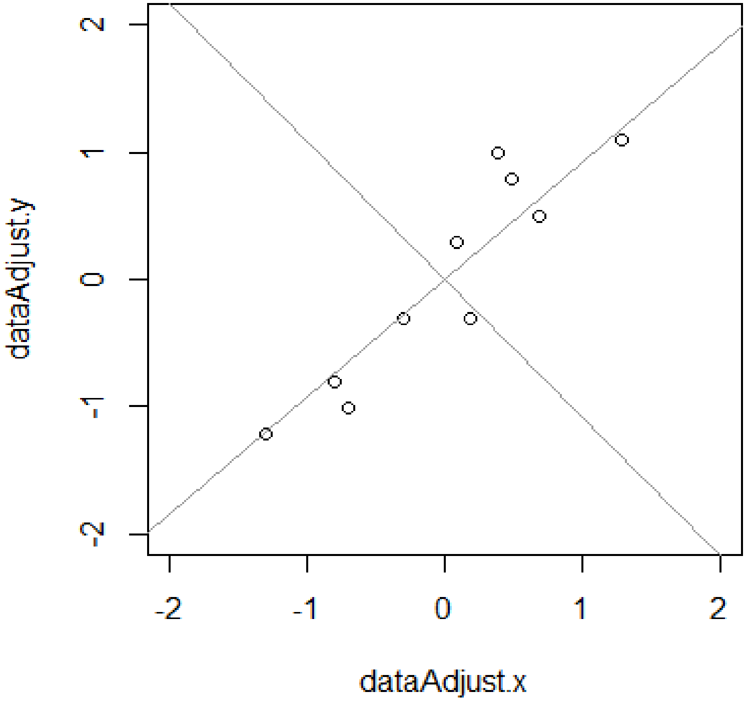
Calculate the Eigenvectors and Eigenvalues of the covariance matrix $\sum$

$$Eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$Eigenvectors = \begin{pmatrix} -.735178656 & .677873399 \\ .677873399 & .735178656 \end{pmatrix}$$

Once Eigenvectors are found from the covariance matrix, the next step is to order them by Eigenvalue, highest to lowest. This gives you the components in order of significance: Φ

PCA Example –STEP 3



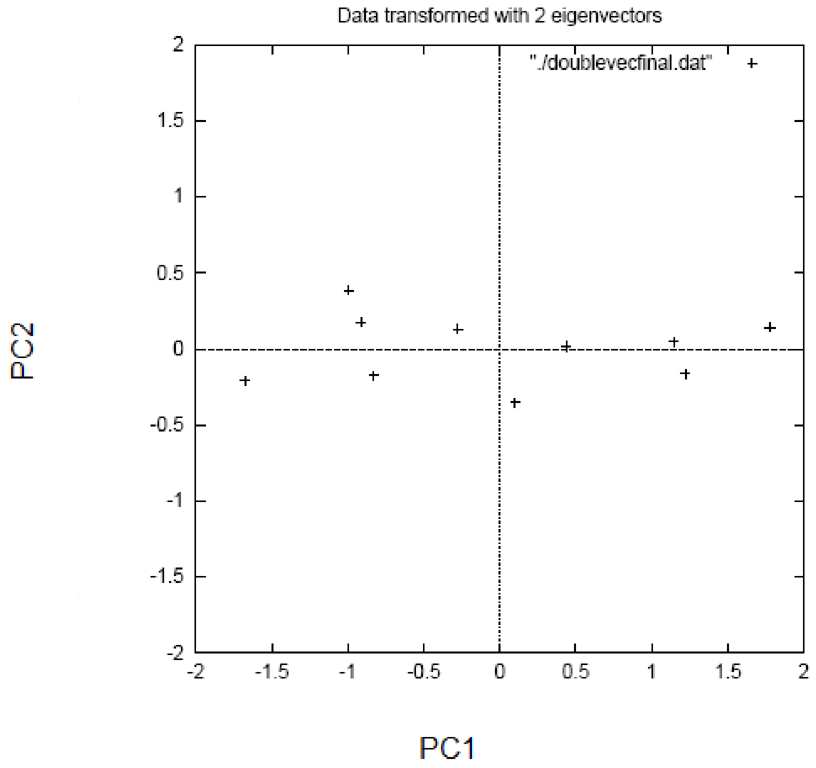
- Eigenvectors are plotted as diagonal dotted lines on the plot
- Note they are perpendicular to each other
- The second Eigenvector gives us the other, less important, pattern in the data, that all the points follow the main line, but are off to the side of the main line by some amount

Taking both Eigenvectors and Rotate

Final data: $\mathbf{Z} = \mathbf{X} * \mathbf{\Phi}$

If you take all Eigenvectors in $\mathbf{\Phi}$, you will get the original data rotated so that the Eigenvectors are the axes! Rotation is equivalent to a basis transformation by an orthonormal basis.

Z1 (PC1)	Z2 (PC2)
0.827970186	$-\ 0.175115307$
-1.77758033	0.142857227
0.992197494	0.384374989
0.274210416	0.130417207
1.67580142	$-\ 0.209498461$
0.912949103	0.175282444
-0.0991094375	$-\ 0.349824698$
-1.14457216	0.0464172582
-0.438046137	0.0177646297
-1.22382056	$-\ 0.162675287$



PCA Example –STEP 4

- Now, if you like, you can decide to ignore the components of lesser significance
- You do lose some information, but if the Eigenvalues are small, you don’t lose much
  - p dimensions in your data
  - calculate p Eigenvectors and Eigenvalues
  - choose only the first k Eigenvectors
  - final data set has only k dimensions

PCA Example –STEP 4

- Feature vector = $(v_1\ v_2\ v_3 \ldots v_{\ k})$
- We can either form a feature vector with both of the Eigenvectors:

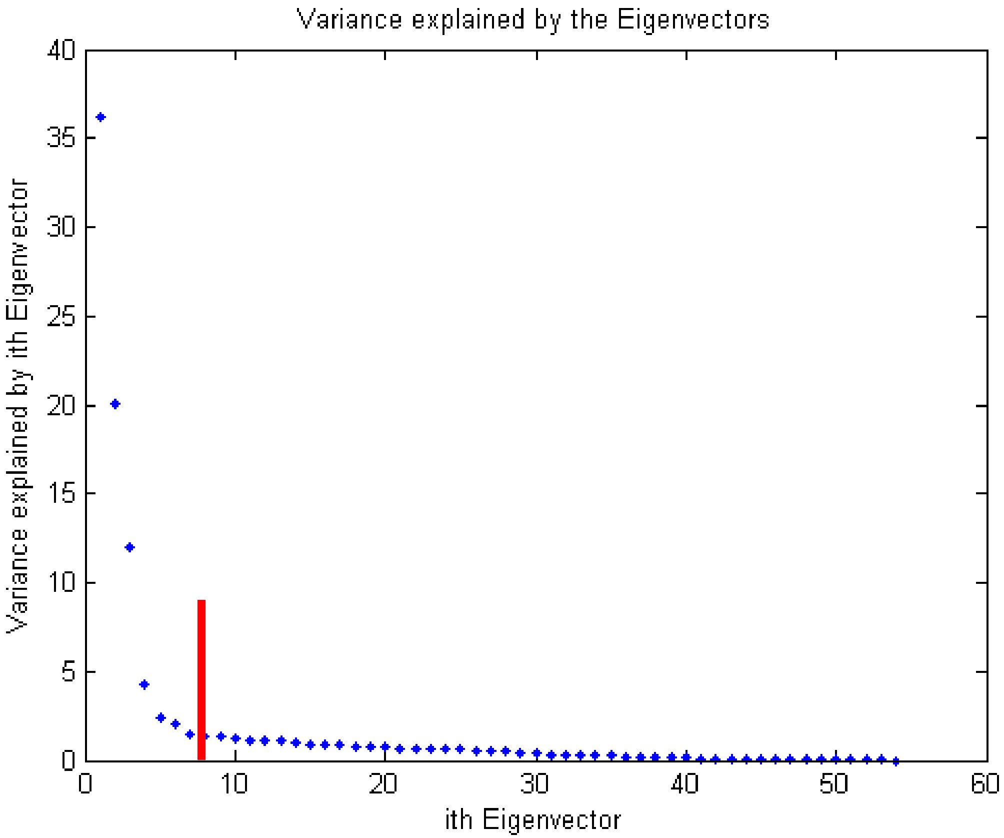
$$\begin{pmatrix} 0.677873399 & -.735178656 \\ 0.735178656 & .677873399 \end{pmatrix}$$

- or, we can choose to leave out the smaller, less significant component and only have a single column:

$$\begin{pmatrix} 0.677873399 \\ 0.735178656 \end{pmatrix}$$

How many Principal Components?

- Check the distribution of Eigenvalues in a scree plot (below)
- Take enough Eigenvalues to cover 80-90% of the total variance



Taking only 1 Eigenvector: Data Reduction

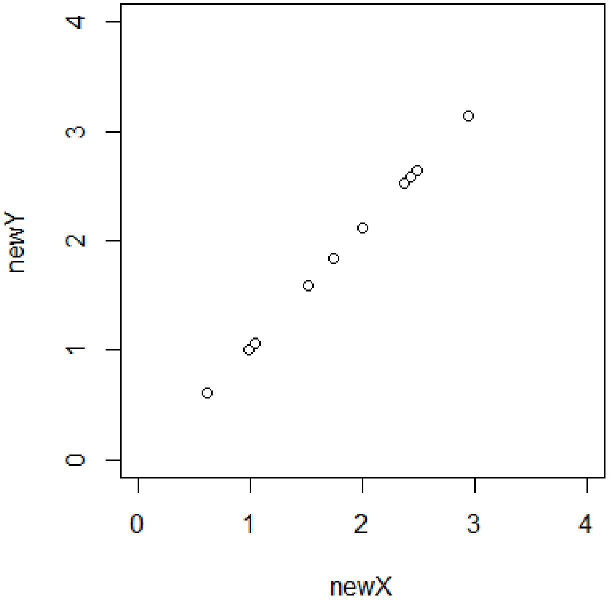
Taking only one Eigenvector: $\mathbf{Z} = \mathbf{X} * \mathbf{\Phi}$

$$\begin{pmatrix} 0.69 & 0.49 \\ -1.31 & -1.21 \\ 0.39 & 0.99 \\ 0.09 & 0.29 \\ 1.29 & 1.09 \\ 0.49 & 0.79 \\ 0.19 & -0.31 \\ -0.81 & -0.81 \\ -0.31 & -0.31 \\ -0.71 & -1.01 \end{pmatrix} * \begin{pmatrix} 0.677873399 \\ 0.735178656 \end{pmatrix} = \begin{pmatrix} 0.827970186 \\ -1.77758033 \\ 0.992197494 \\ 0.274210416 \\ 1.67580142 \\ 0.912949103 \\ -0.0991094375 \\ -1.14457216 \\ -0.438046137 \\ -1.22382056 \end{pmatrix}$$

Reconstruction of Original Data with one EV

- Restore original data: $\hat{\boldsymbol{X}} \approx \mathbf{Z}\mathbf{\Phi}^{\mathrm{T}} + original\ mean$
  - $\mathbf{X} \approx$ PCA Scores * Eigenvectors + original mean

$\mathbf{Z}$	$\mathbf{\Phi}$
0.827970186	0.6778734
-1.77758033	0.7351787
0.992197494	
0.274210416	
1.67580142	
0.912949103	
-0.0991094375	
-1.14457216	
-0.438046137	
-1.22382056	



- If we reduce the dimensionality, then, when reconstructing the data, we lose those dimensions we chose to discard.
- Note that if all p Eigenvectors are used, then $\mathbf{\Phi}\mathbf{\Phi}^{\mathrm{T}} = I$

Summary of Steps 1-3

$\mathbf{A} =$

Data (A):

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

$\mathbf{X} =$

Zero mean Data (X)

x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

$$\sum = \begin{pmatrix} 0.616555556 & 0.615444444 \\ 0.615444444 & 0.716555556 \end{pmatrix}$$

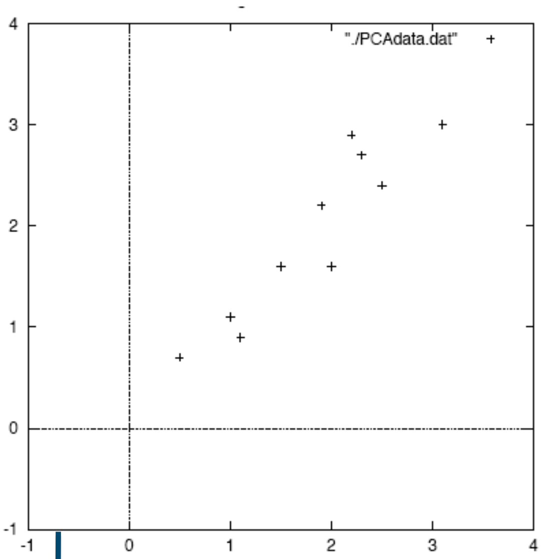
$$Eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$Eigenvectors = \begin{pmatrix} -.735178656 & 0.677873399 \\ 0.677873399 & 0.735178656 \end{pmatrix}$$

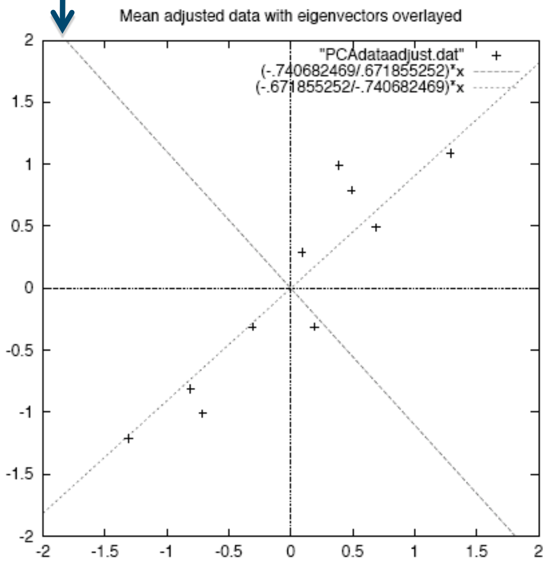
Then order Eigenvectors after decresing Eigenvalues in matrix $\mathbf{\Phi}$, the principal components:

$$\mathbf{\Phi} = \begin{pmatrix} 0.677873399 & -.735178656 \\ 0.735178656 & 0.677873399 \end{pmatrix}$$

$\mathbf{A} =$



$\mathbf{X} =$



PCA and Aggregation of Attributes

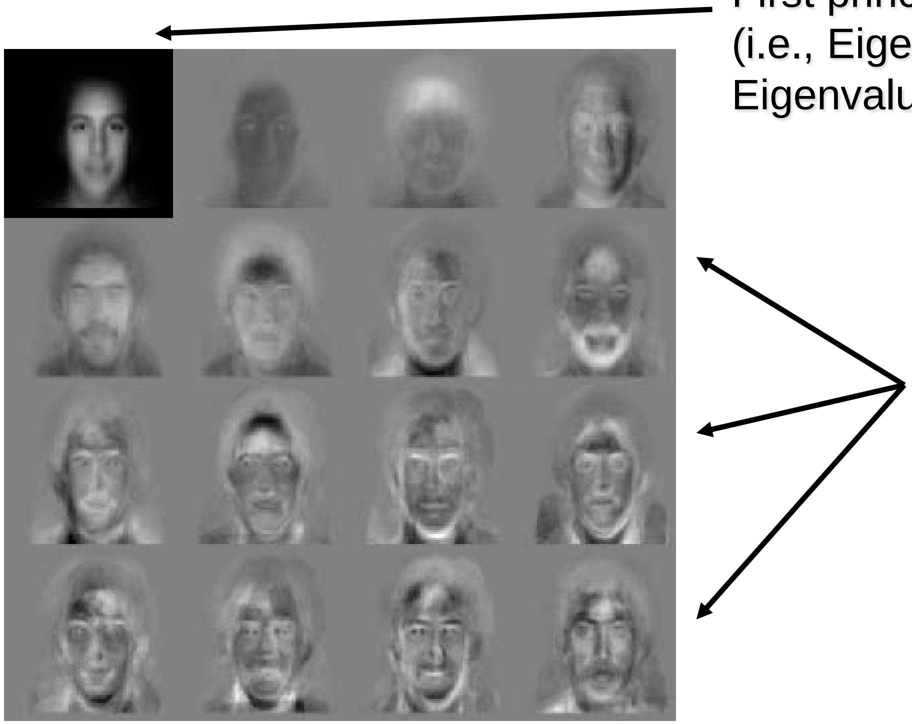
- Matrix **Φ** also allows aggregating similar attributes
- Each element of the Eigenvectors represents the contribution of a given variable to a component
  - In the example below the attributes volume, length, width, and depth all have high impact on the first component. They could as well be considered representations of a latent variable “size”, while speed1 and speed2 could be aggregated to “speed”
  - Similarly, determine *latent variables* such as „social status“ in customer data.

Principal Components – Matrix **Φ**

	1	2	3	4	5	6	7	8
Volume	**0.948**	−0.094	−0.129	0.228	0.040	0.036	0.136	0.055
Length	**0.906**	0.302	−0.064	−0.209	0.128	−0.144	−0.007	−0.050
Width	**0.977**	−0.128	−0.031	0.032	0.103	−0.017	−0.014	0.129
Depth	**0.934**	−0.276	−0.061	0.014	0.074	0.129	0.154	−0.038
Speed1	0.552	**0.779**	−0.196	−0.133	−0.099	0.143	−0.038	0.018
Speed2	−0.520	**0.798**	−0.157	0.222	0.109	−0.038	0.071	0.004
Radius	0.398	0.311	**0.862**	0.038	0.008	0.022	−0.002	−0.005

Applications to Image Compression

- Images: here the rows of a matrix are grey scale values of a pixel
- Compression: each image can be approximated by projection onto the first few principal components
- Recognition: measure difference to existing pictures on the new axes derived from the PCA
  PCA is also known as Karhunen-Loève transformation



First principal component (i.e., Eigenvector with largest Eigenvalue)

Other principal components

Assumptions of PCA

PCA assumes relationships among variables are LINEAR

- cloud of points in p-dimensional space has linear dimensions that can be effectively summarized by the principal axes
- If the structure in the data is NONLINEAR, the principal axes will not be an efficient and informative summary of the data.

PCA uses the Euclidean distance among points assuming continuous variables. With discrete variables special techniques are in order (e.g., correspondence analysis).

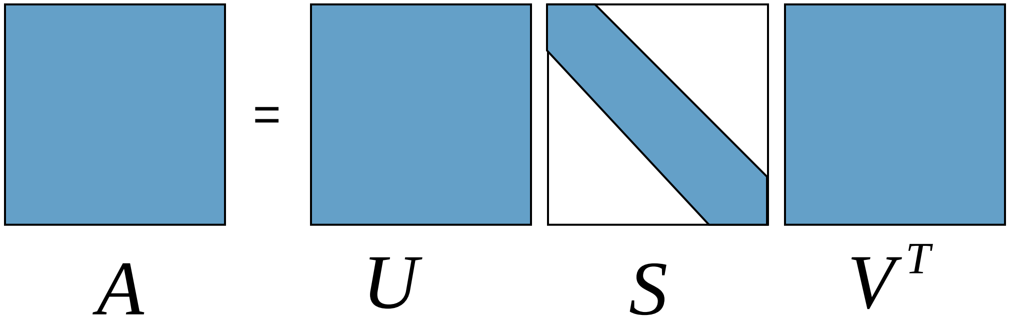
Outline for today

- Overview
- Linear algebra revisited
- Principal Component Analysis
- **Singular Value Decomposition**
- PCA regression and regularization

Singular Value Decomposition (SVD)

- SVD of the data matrix $\boldsymbol{A} \in R^{m\times n}$ can be seen as an alternative technique to compute the same Eigenvectors.
- For any matrix $\boldsymbol{A} \in R^{m\times n}$, there exist orthogonal matrices $\boldsymbol{U}$**,** $\boldsymbol{V}$ and a diagonal matrix $\boldsymbol{S}$, such that all the diagonal values σ_{i} of $\boldsymbol{S}$ are non-negative and

$$A = USV^T$$



You can compute PCA via SVD

SVD is a matrix factorization technique used to compute PCA. Data matrix **A**, ***rows = data points***, ***columns = variables*** (attributes, features, parameters)

1. Center the data by subtracting the mean of each column
2. Compute the SVD of the centered matrix **X** (i.e., find the first **k** singular values/vectors) $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T$
3. The principal components are the columns of **V**, the coordinates of the data in the basis defined by the principal components are **US**

SVD and PCA

- The diagonal values of $\boldsymbol{S}$ are called the **singular values**. It is accustomed to sort them by size.
- The columns of $\boldsymbol{U}$ are called the **left singular vectors**.
- The columns of $\boldsymbol{V}$ are called the **right singular vectors**.

$$A = USV^T$$

- The columns of $\boldsymbol{V}$ are the **principal axes**.
- The columns of $\boldsymbol{US}$ are principal component scores
- **Singular values** of the SVD decomposition of the matrix $\boldsymbol{A}$ **is the square root of the Eigenvalues** of the matrix ($\boldsymbol{AA^t}$) or ($\boldsymbol{A^tA}$).
- Finding $\boldsymbol{U, S, V}$ is equivalent to finding Eigenvectors of $\boldsymbol{A^tA}$.

SVD (on Uncentred Data)

Data $\boldsymbol{A}$ =

10	20	10
2	5	2
8	17	7
9	20	10
12	22	11

1st row of $\boldsymbol{A}$ $= [10\ 20\ 10]$

Since $\boldsymbol{A} = U\,S\,V^t$,

then $A(1,) = U(1,) * \boldsymbol{S} * V^t$

$= [24.3\ \ 0.21\ -0.228] * V^t$

$\Rightarrow A(1,) = 24.3\, v_1 + 0.21\, v_2 + -0.228\, v_3$

where v_1, v_2, v_3 are rows of V^t

$[24.3, 0.21, -0.228] * [0.41,0.73,0.55]^T = \mathbf{10}$

$$A = U\,S\,Vt$$

where U =

$$\begin{array}{ccc} \boxed{0.50 \quad 0.14 \quad -0.19} \\ 0.12 & -0.35 & 0.07 \\ 0.41 & -0.54 & 0.66 \\ 0.49 & -0.35 & -0.67 \\ 0.56 & 0.66 & 0.27 \end{array}$$

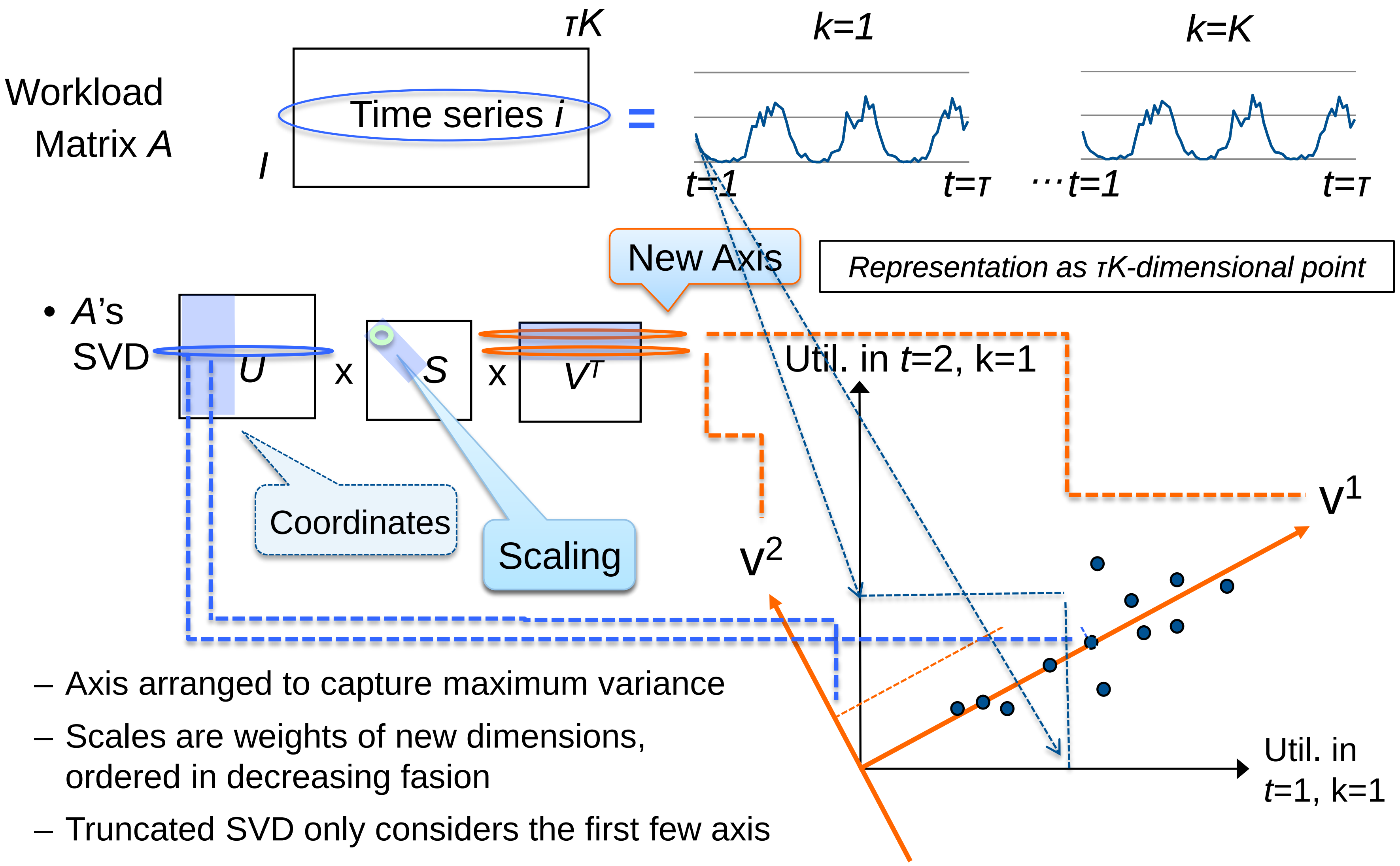
where S =

$$\begin{array}{ccc} 48.6 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1.2 \end{array}$$

and V^t =

$$\begin{array}{ccc} 0.41 & 0.82 & 0.40 \\ 0.73 & -0.56 & 0.41 \\ 0.55 & 0.12 & -0.82 \end{array}$$

Example: Time Series of *I* Customers



- *A*'s SVD
  - Axis arranged to capture maximum variance
  - Scales are weights of new dimensions, ordered in decreasing fasion
  - Truncated SVD only considers the first few axis

Principal Components: Summary

- PCA takes a data matrix of n objects by p variables, which may be correlated, and summarizes it by uncorrelated axes (principal components or principal axes) that are linear combinations of the original p variables.
- The first k components display as much as possible of the variation in the data.
- The first PC is the direction of maximum variance from origin. Subsequent PCs are orthogonal to first PC and describe maximum residual variance.
- PCA can be quite useful to combat multicollinearity in the regression analysis.

How do Eigenvectors play a role in Principal Component Analysis?

Outline for today

- Overview
- Linear algebra revisited
- Principal Component Analysis
- Singular Value Decomposition
- **PCA regression and regularization**

Reminder: Multiple Linear Regression

- Equation: $y = X\beta + \varepsilon$
  - y: $n * 1$ vector of observed values
  - X: $n * p$ matrix of independent values
  - β: $p * 1$ vector of regression parameters
  - ε: $n * 1$ vector of residuals

- OLS estimator: $\hat{\beta} = (\mathrm{X}^{\mathrm{T}}\mathrm{X})^{-1}\mathrm{X}^{\mathrm{T}}y$

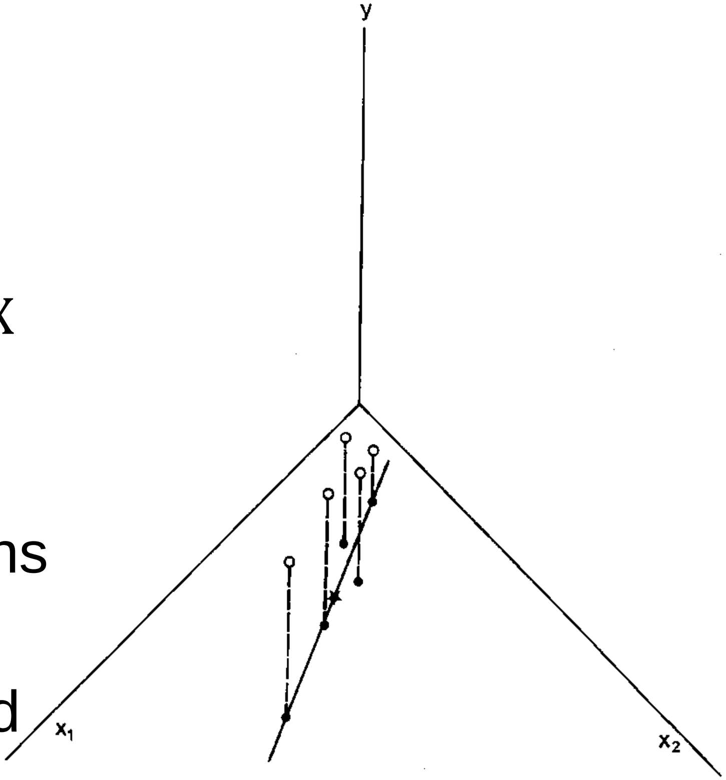
Multicollinearity in Linear Regression Models

- $\min(\|X\beta - y\|^2)$

- MLR solution, requires $n \geq p$
  (variable selection) and nearly orthogonal X

- If $\mathrm{X}^{\mathrm{T}}\mathrm{X}$ is not full rank
  - No unique solution to normal equations

- If the columns of X are highly correlated
  - Leads to unstable equation/plane
    in the direction with little variability



Considerations in High Dimensions

While p can be extremely large, the number of observations n is often limited due to cost, sample availability, etc.

Data sets containing more features than observations are often referred to a *high-dimensional*.

When the number of features p is as large as, or larger than, the number of observations n, OLS should not be performed.

- It is too *flexible* and hence overfits the data.

Forward stepwise selection, ridge regression, lasso, and PCR are particularly useful for performing regression in the high-dimensional setting.

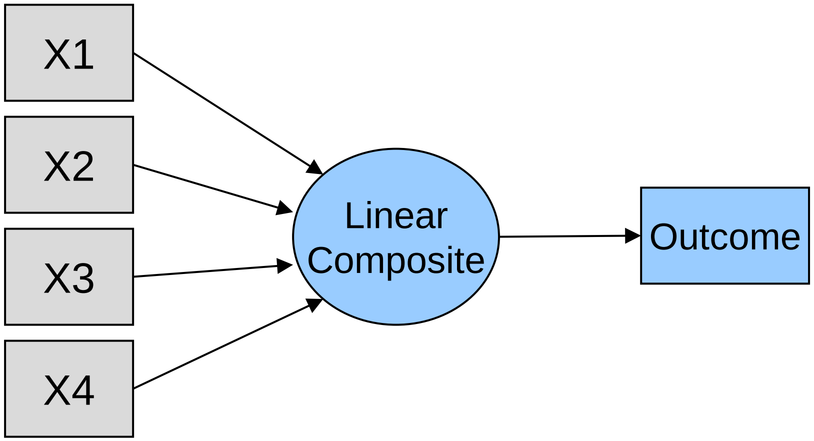
Solutions to Multicollinearity

- Subset selection
  - best subset, backward, forward, stepwise selection of features
  - (already discussed in the context of the linear regression)

- Using derived input
  - **Principal component regression**
  - Partial least squares

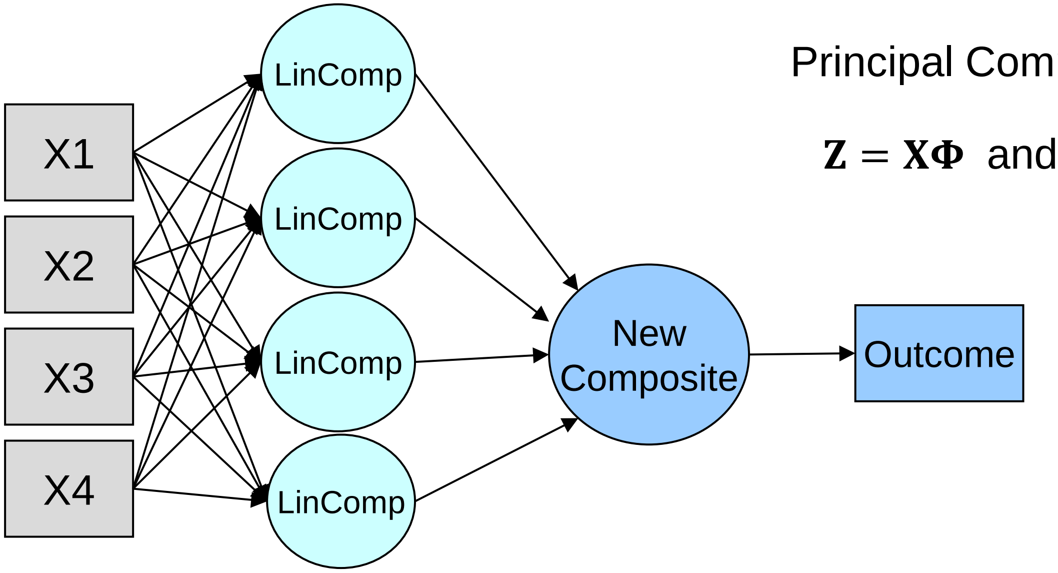
- Coefficient shrinkage (regularization)
  - Ridge regression
  - Lasso (least absolute shrinkage and selection operator)

Multiple Regression vs. Principal Component Regression



Multiple Linear Regression

$$y = \mathbf{X}\beta + \varepsilon$$



Principal Component Regression

$\mathbf{Z} = \mathbf{X}\boldsymbol{\Phi}$ and $y = \mathbf{Z}\gamma + \varepsilon$

Back to our Example

$$\begin{pmatrix} 0.69 & 0.49 \\ -1.31 & -1.21 \\ 0.39 & 0.99 \\ 0.09 & 0.29 \\ 1.29 & 1.09 \\ 0.49 & 0.79 \\ 0.19 & -0.31 \\ -0.81 & -0.81 \\ -0.31 & -0.31 \\ -0.71 & -1.01 \end{pmatrix} * \begin{pmatrix} 0.677873399 \\ 0.735178656 \end{pmatrix} = \begin{pmatrix} 0.827970186 \\ -1.77758033 \\ 0.992197494 \\ 0.274210416 \\ 1.67580142 \\ 0.912949103 \\ -0.0991094375 \\ -1.14457216 \\ -0.438046137 \\ -1.22382056 \end{pmatrix}$$

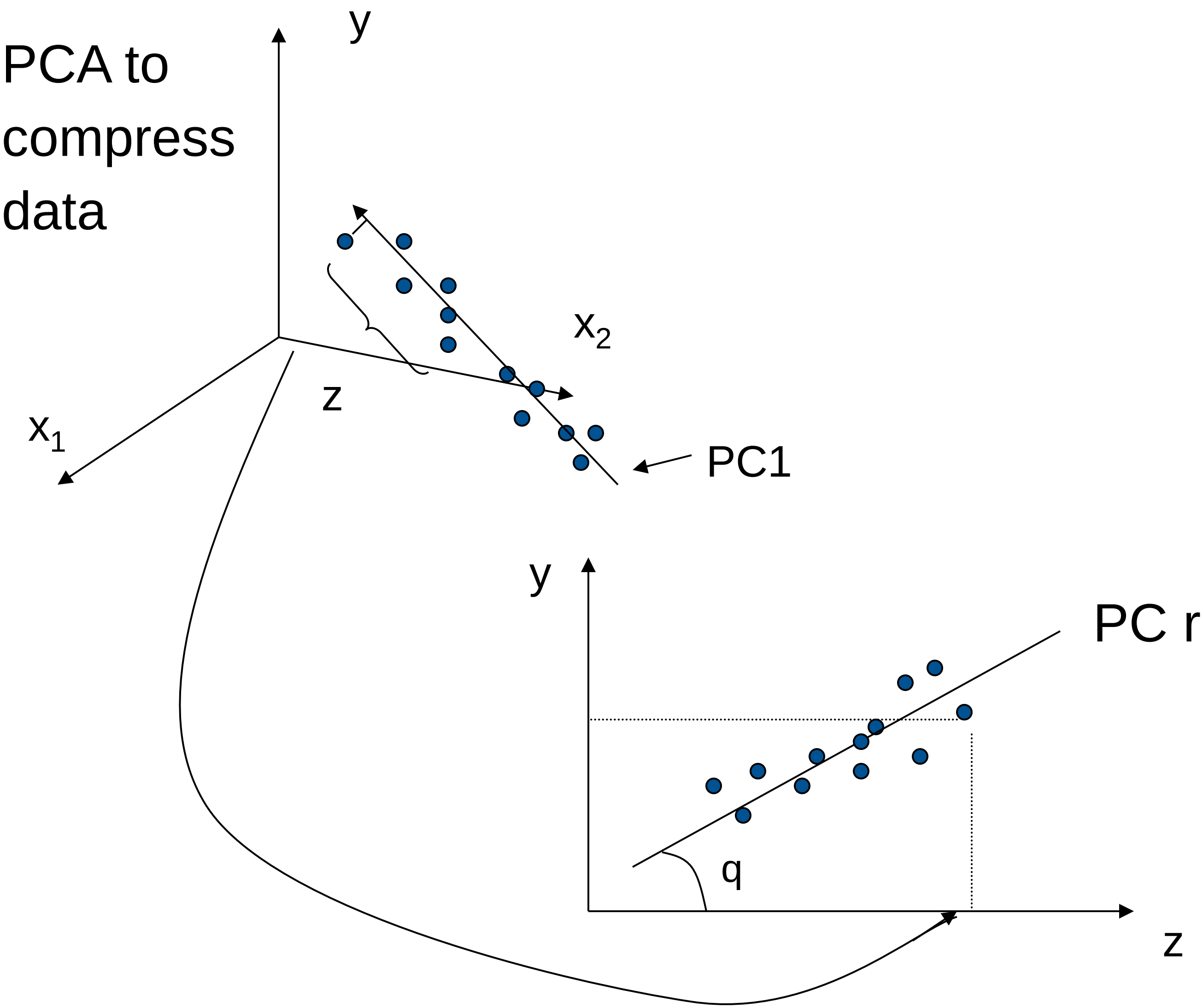
$$\mathbf{X} * \mathbf{\Phi} = \mathbf{Z}$$

PC regression: $y = \mathbf{Z}\gamma + \varepsilon$

Now, we need to estimate γ via the OLS estimator $\hat{\gamma} = (\mathbf{Z}^t\mathbf{Z})^{-1}\mathbf{Z}^t y$

- The independent variables are now the principal components in $\mathbf{Z}$.
- If only a subset of the principal components in $\mathbf{Z}$ is used for the regression, the coefficients in γ remain the same as the PCs are orthogonal.
- Remember, the PCs are linear combinations of all original variables.

Principal Component Regression



PCA to compress data

PC regression

Principal Components Regression

PCR will tend to do well in cases when the first few principal components are sufficient to capture most of the variation in the predictors as well as the relationship with the response.

We note that even though PCR provides a simple way to perform regression using $k < p$ predictors, it *is not* a feature selection method.

In PCR, the number of principal components is typically chosen by cross-validation.

Related Topic: Partial Least Squares

- Partial Least Squares (PLS) is just like PC Regression except in how the components are computed
- Unlike PCR, PLS identifies these new features in a supervised way;
  - PLS makes use of the response y in order to identify new features that not only approximate the old features well, but also that are related to the response.
- PCR = weights are calculated from the covariance matrix of the predictors
- PLS = weights reflect the covariance structure between predictors and response y
  - While conceptually not too much of a stretch, it requires a more complicated iterative algorithm

Regularization

- If the linear model is correct for a given problem, then the OLS prediction is unbiased, and has the lowest variance among all linear unbiased estimators.
- But there can be (and often exist) biased estimators with smaller MSE.
- Generally, by regularizing the estimator in some way, its variance will be reduced;
- if the corresponding increase in bias is small, this will be worthwhile.
- Examples of regularization:
  - subset selection (forward, backward, all subsets)
  - ridge regression
  - the lasso

Regularization

Objective function:

$$J(\theta) = L(\theta) + \Omega(\theta)$$

- $L(\theta)$ is training loss: how well model fit on training data.
- $\Omega(\theta)$ is regularization, measures complexity of model.
- Lower training loss result in more predictive model.
- Lower regularization result in simpler (more biased) model.

Regularization methods introduce bias into the regression solution that can reduce variance considerably relative to the ordinary least squares (OLS) solution.

Ridge Regression

Ridge coefficient minimize a penalized RSS. The parameter λ>0 penalizes β_j proportional to its size β_j^2.

$$\hat{\beta}^{ridge} = \arg\min_{\beta}\{\sum_i (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2\}$$

Or

$$\underset{\beta}{Minimize}\{\sum_i (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2\}$$

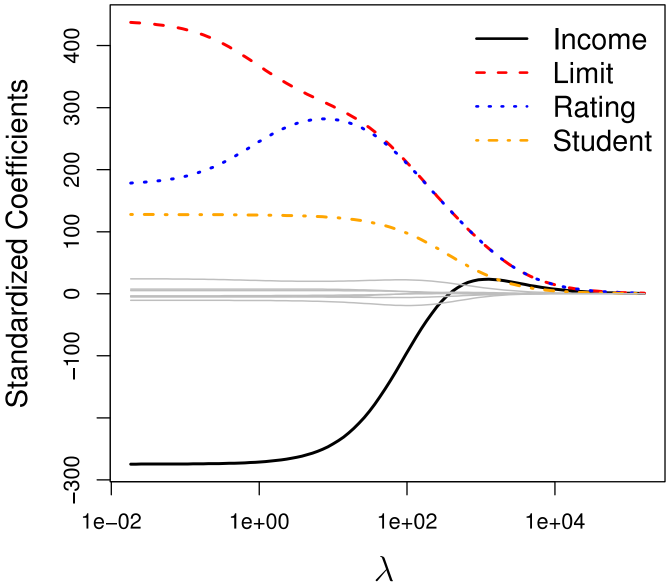
$$subject \text{ to} \sum_{j=1}^{p} \beta_j^2 \le s$$

This is a biased estimator that for some value of λ>0 may have smaller mean squared error than the least squares estimator.

Ridge Regression (cont.)

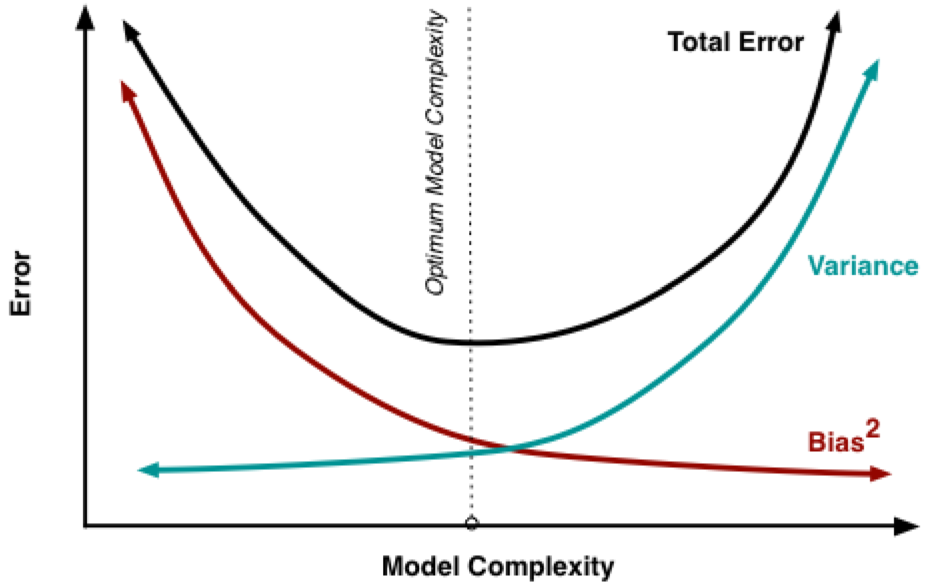
- As λ increases, the standardized ridge regression coefficients shrink to zero.
- Thus, when λ is extremely large, then all of the ridge coefficient estimates are basically zero; this corresponds to the *null model* that contains no predictors.

It is best to apply ridge regression after *standardizing the predictors*



Ridge Regression

- In general, the ridge regression estimates will be *more biased* than the OLS ones but have *lower variance*.
- Ridge regression will work best in situations where the OLS estimates have high variance.



The Lasso

The lasso coefficients minimize the quantity:

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}|\beta_j| \;=\mathrm{RSS}+\lambda\sum_{j=1}^{p}|\beta_j|$$

The key difference from ridge regression is that the lasso uses an ℓ_1 penalty instead of an ℓ_2, which has the effect of forcing some of the coefficients to be exactly equal to zero when the tuning parameter λ is sufficiently large.

Thus, the lasso performs variable/feature selection.

Lasso vs. Ridge Regression

- The lasso has a major advantage over ridge regression, in that it produces simpler and more interpretable models that involved only a subset of predictors.
- The lasso leads to qualitatively similar behavior to ridge regression, in that as λ increases, the variance decreases and the bias increases.
- The lasso can generate more accurate predictions compared to ridge regression.
- Cross-validation can be used in order to determine which approach is better on a particular data set.

Selecting the Tuning Parameter λ

- As for subset selection, for ridge regression and lasso we require a method to determine which of the models under consideration is best; thus, we required a method selecting a value for the tuning parameter λ or equivalently, the value of the constraint *s*.
- Select a grid of potential values; use cross-validation to estimate the error rate on test data (for each value of λ) and select the value that gives the smallest error rate.
- Finally, the model is re-fit using all of the variable observations and the selected value of the parameter λ.

Ridge vs. PCA vs. PLS vs. Lasso

- A recent study has shown that ridge regression and PCR outperform PLS in prediction.
- Lasso outperforms ridge when there are a moderate number of sizable effects, rather than many small effects. It also produces more interpretable models.
- Regularization is still a topic for ongoing research.