Big Data Computing

Master's Degree in Computer Science

2022-2023

SAPIENZA
UNIVERSITÀ DI ROMA

Gabriele Tolomei

Department of Computer Science

Sapienza Università di Roma

tolomei@di.uniroma1.it

Recap from Last Lecture(s)

- High-dimensional naïve representation (i.e., feature space) of text data

Recap from Last Lecture(s)

- High-dimensional naïve representation (i.e., feature space) of text data
- Clustering high-dimensional data may be problematic
  - Due to the curse of dimensionality

Recap from Last Lecture(s)

- High-dimensional naïve representation (i.e., feature space) of text data
- Clustering high-dimensional data may be problematic
  - Due to the curse of dimensionality
- Many other data sources (e.g., images) share the same issue

Recap from Last Lecture(s)

- High-dimensional naïve representation (i.e., feature space) of text data
- Clustering high-dimensional data may be problematic
  - Due to the curse of dimensionality
- Many other data sources (e.g., images) share the same issue
- Good news! High-dimensionality is often not real!
  - Due to the way in which we observe/collect data

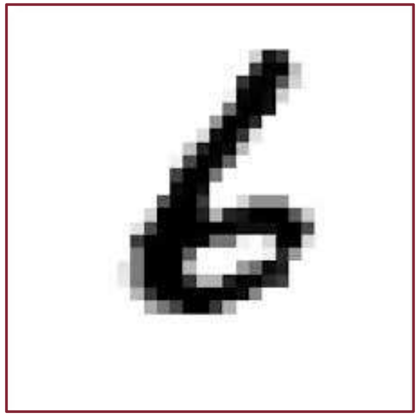
DIMENSIONALITY REDUCTION

Modeled vs. True Dimensionality

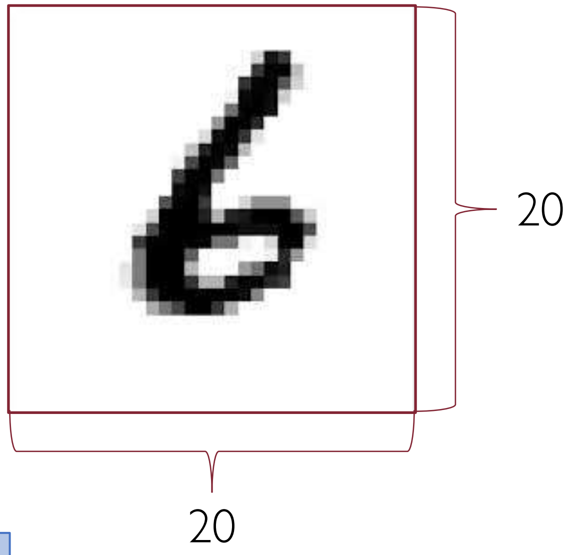
<u>Example</u>

Handwritten digit recognition

Modeled vs. True Dimensionality



Modeled vs. True Dimensionality

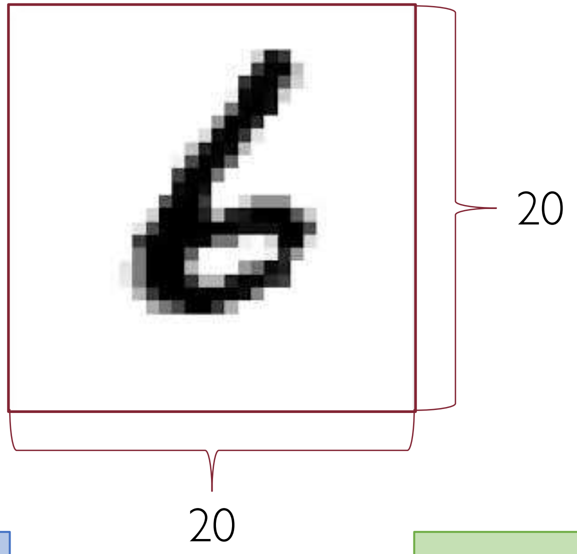


Modeled dimensionality

Each digit represented by 20x20 bitmap

400-dimensional binary vector

Modeled vs. True Dimensionality



Modeled dimensionality

Each digit represented by 20x20 bitmap

400-dimensional binary vector

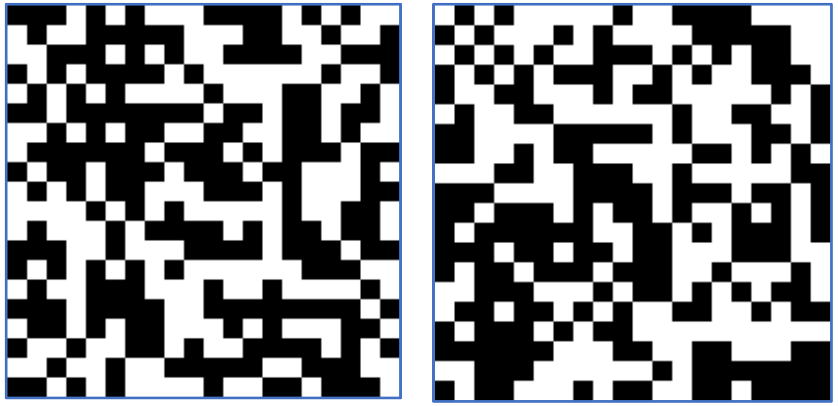
True dimensionality

Actual digits just cover a tiny fraction of all this huge space

Small variations of the pen-stroke

Modeled vs. True Dimensionality

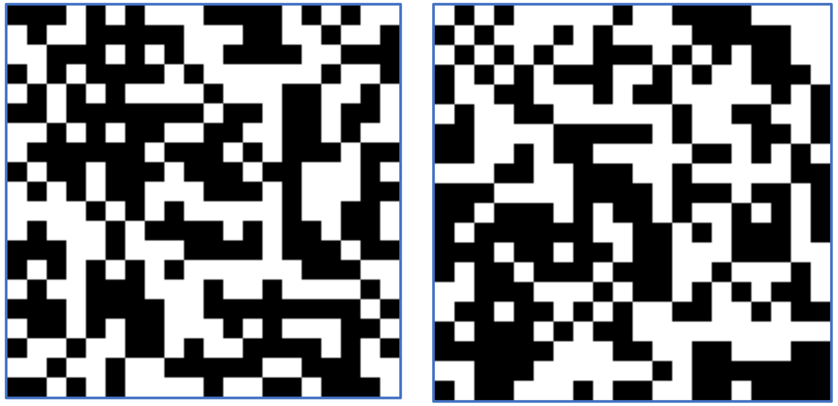
Random samples from 400-d space

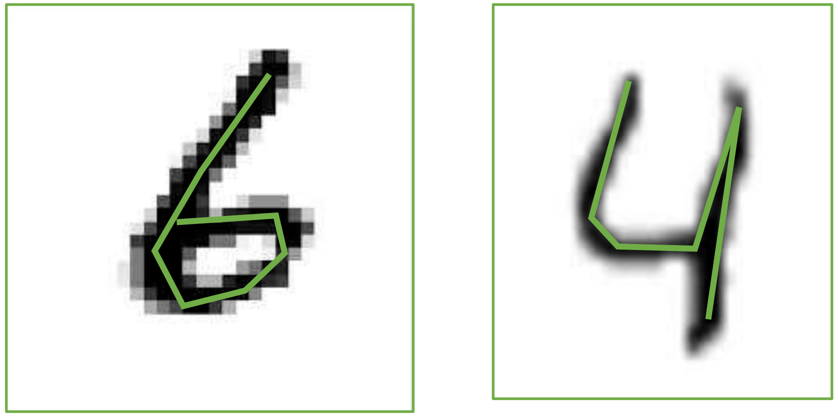


Modeled vs. True Dimensionality

Random samples from 400-d space

True digits living in a 400-d space

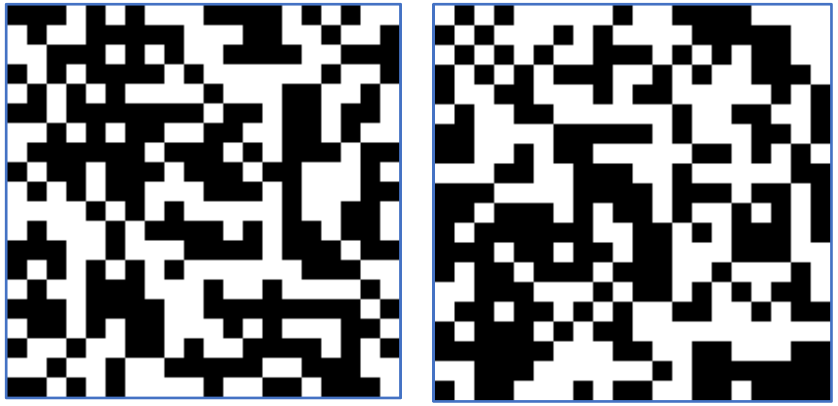


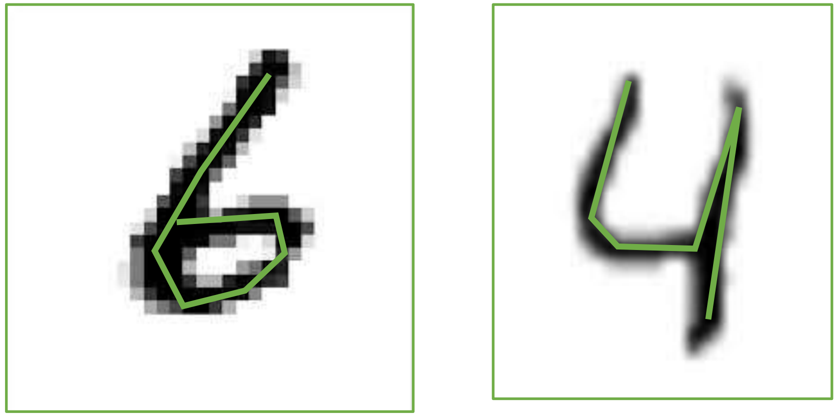


Modeled vs. True Dimensionality

Random samples from 400-d space

True digits living in a 400-d space





We model data (i.e., digits) as very high dimensional…

… In fact, they are not so

The Curse of Dimensionality

As dimensionality grows fewer examples in each region of the feature space
(assuming # examples is constant)

The Curse of Dimensionality

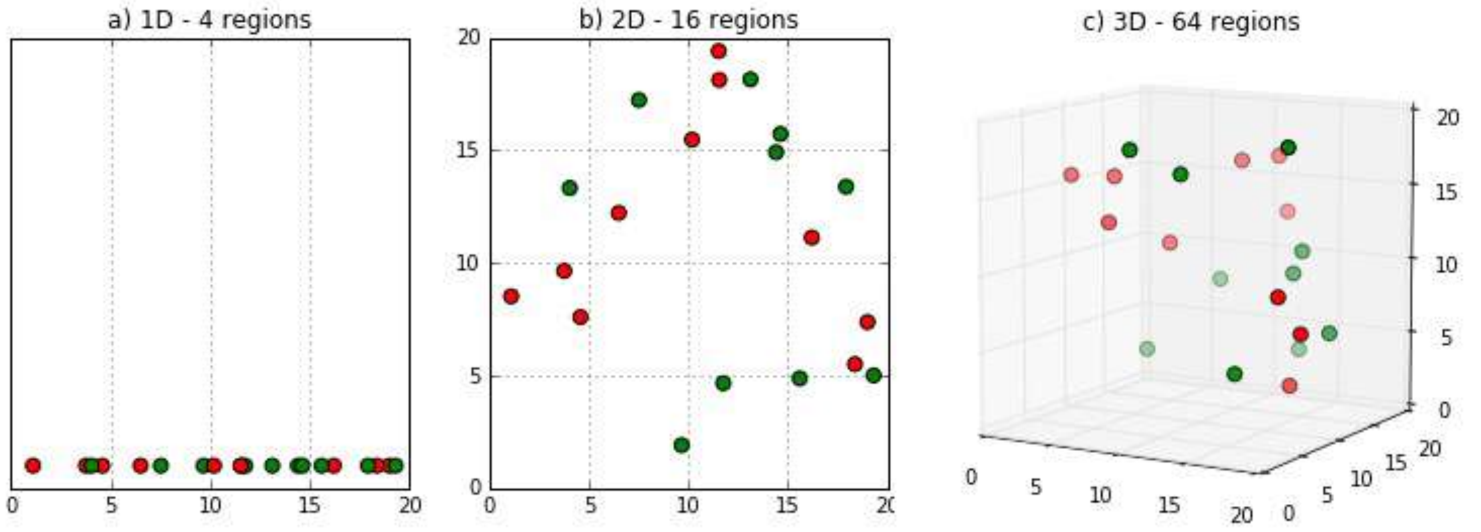
As dimensionality grows fewer examples in each region of the feature space (assuming # examples is constant)

Put it another way:
The number of examples must grow exponentially with dimensionality if we want to maintain the same "density"

The Curse of Dimensionality

As dimensionality grows fewer examples in each region of the feature space (assuming # examples is constant)

Put it another way:
The number of examples must grow exponentially with dimensionality if we want to maintain the same "density"



Dealing with High Dimensionality

3 possible approaches

Feature Engineering
(using domain knowledge)

e.g., SIFT in computer vision

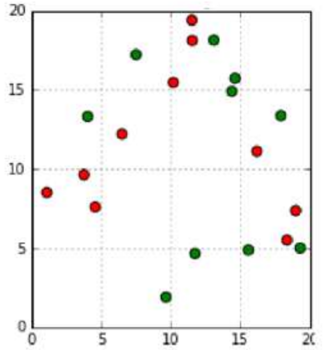
Dealing with High Dimensionality

3 possible approaches

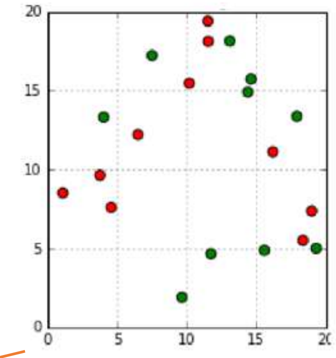
Feature Engineering
(using domain knowledge)

e.g., SIFT in computer vision

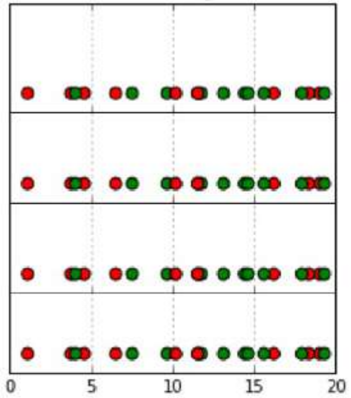
Making Assumptions



Dealing with High Dimensionality: Assumptions

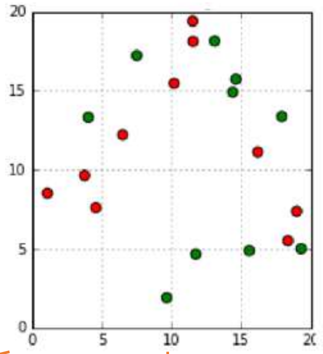


independence

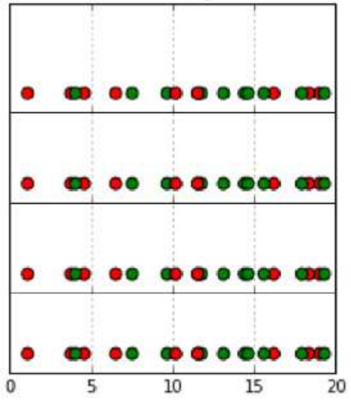


Count along each dimension separately

Dealing with High Dimensionality: Assumptions

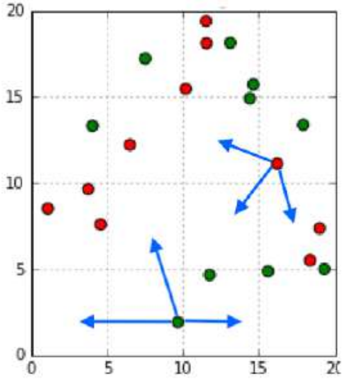


independence



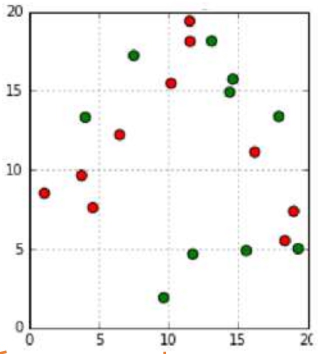
Count along each dimension separately

smoothness

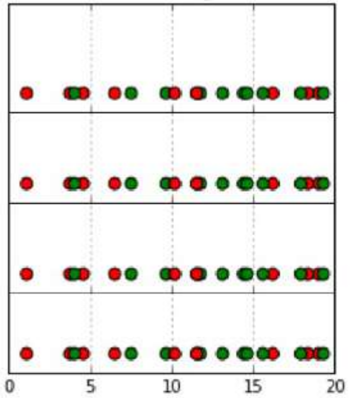


Propagate counts to neighboring regions

Dealing with High Dimensionality: Assumptions

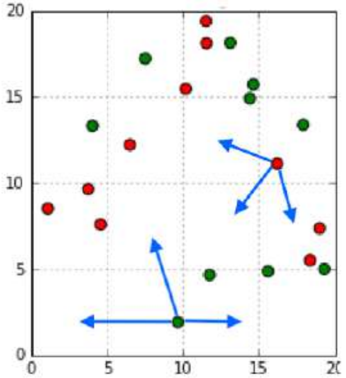


independence



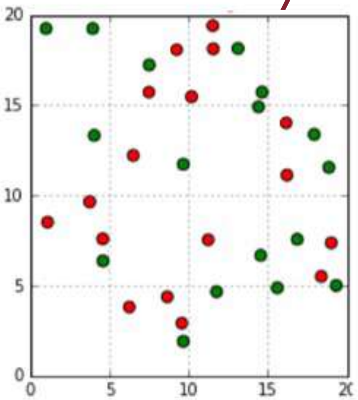
Count along each dimension separately

smoothness



Propagate counts to neighboring regions

simmetry



Invariance to the order of dimensions

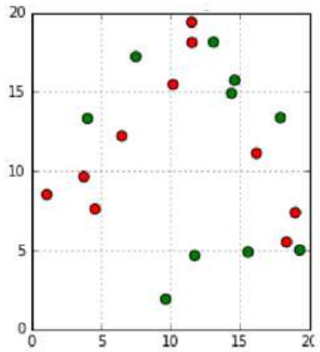
Dealing with High Dimensionality

3 possible approaches

Feature Engineering
(using domain knowledge)

e.g., SIFT in computer vision

Making Assumptions



Reduce Dimensionality

Create a new set of dimensions (i.e., variables)

Dimensionality Reduction

- A technique to unveil the actual (i.e., meaningful) dimensions of data
- A pre-processing step for representing data with fewer features
- Preserve as much "structure" of the data as possible
- Retained structure must be discriminative affecting data separability

"structure" here means **variance**

Dimensionality Reduction

$\boxed{\text{2 main approaches}}$

Dimensionality Reduction

2 main approaches

Feature Selection

Pick a subset of the original dimensions that are good predictors (e.g., using information gain)

$x_1, x_2, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots, x_{d-1}, x_d$

Dimensionality Reduction

2 main approaches

Feature Selection

Pick a subset of the original dimensions that are good predictors (e.g., using information gain)

$x_1, x_2, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots, x_{d-1}, x_d$

Feature Extraction

Build a new set of $k < d$ dimensions as a (linear) combination of the originals

$e_1, e_2, \ldots, e_k$

$e_i = f(x_1, x_2, \ldots, x_d)$

Dimensionality Reduction

2 main approaches

Feature Selection

Pick a subset of the original dimensions that are good predictors (e.g., using information gain)

$x_1, x_2, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots, x_{d-1}, x_d$

Feature Extraction

Build a new set of $k < d$ dimensions as a (linear) combination of the originals

$e_1, e_2, \ldots, e_k$

$e_i = f(x_1, x_2, \ldots, x_d)$

Principal Component Analysis (PCA)

Dimensionality reduction technique based on feature extraction

High-dimensional data is in fact embedded into some lower dimensional space

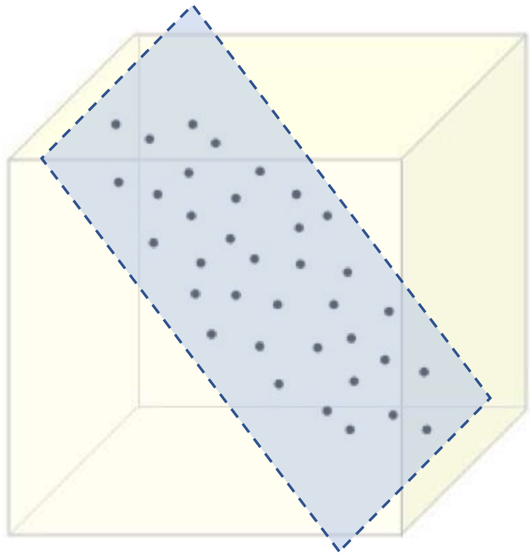
Principal Component Analysis (PCA)

Dimensionality reduction technique based on feature extraction

High-dimensional data is in fact embedded into some lower dimensional space

Example

A 3-d set of points embedded into a 2-d hyperplane



Principal Component Analysis (PCA)

PCA defines a set of principal components as follows:

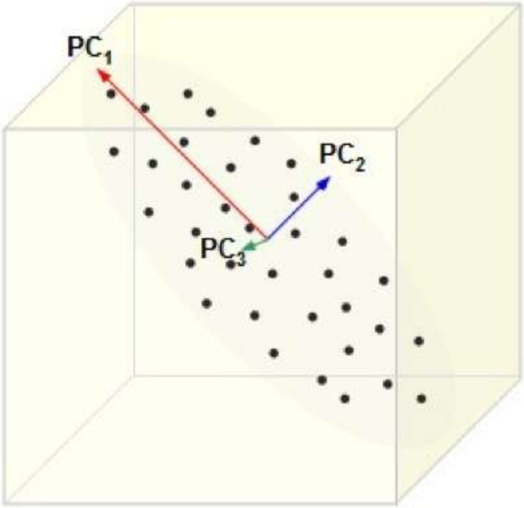
- 1st: direction of the greatest variance of data
- 2nd: perpendicular to 1st and greatest variance of what's left
- … and so on until d

Principal Component Analysis (PCA)

PCA defines a set of principal components as follows:

- 1st: direction of the greatest variance of data
- 2nd: perpendicular to 1st and greatest variance of what's left
- … and so on until d

The top $k < d$ components become the new dimensions



e.g., $k = 2$

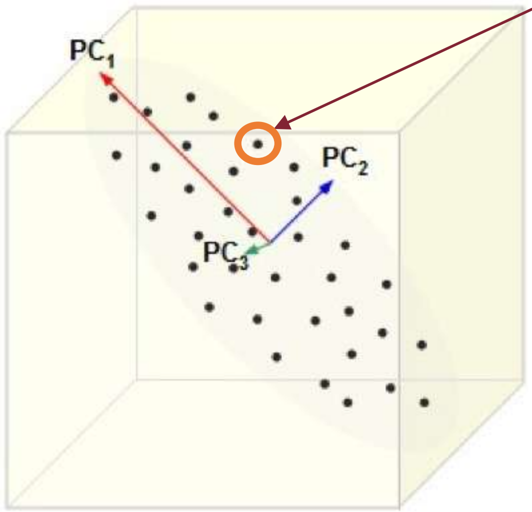
Principal Component Analysis (PCA)

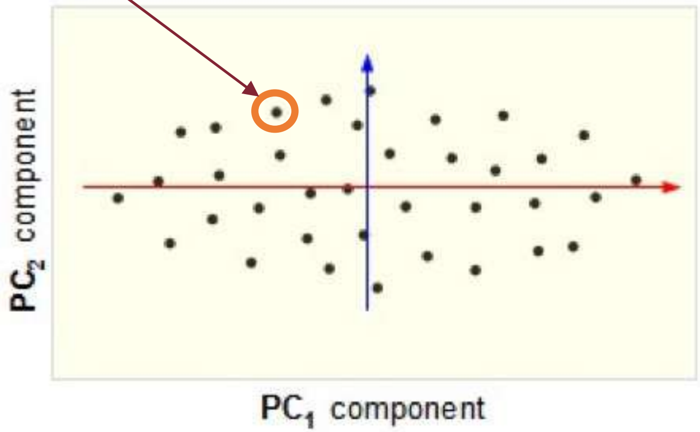
PC_1 and PC_2 are the top-2 principal components

Principal Component Analysis (PCA)

PC_1 and PC_2 are the top-2 principal components

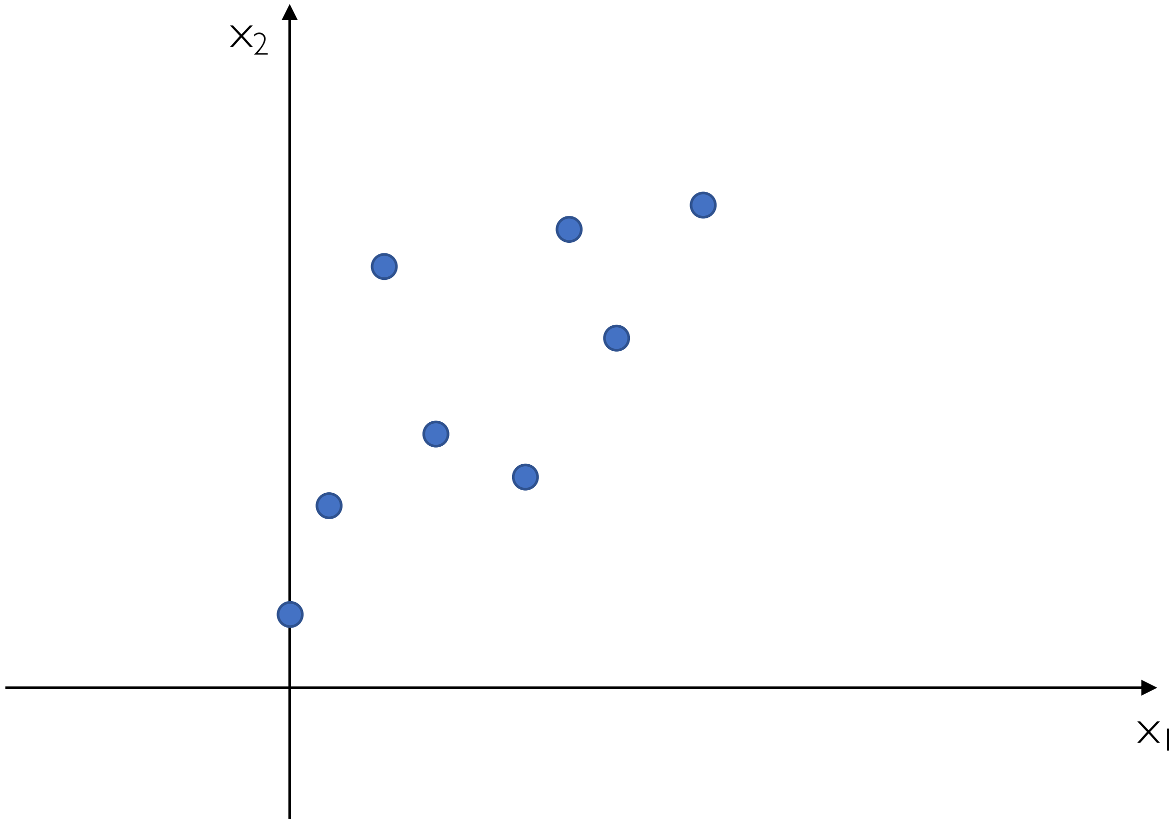
Change the coordinates of every point according to the new dimensions





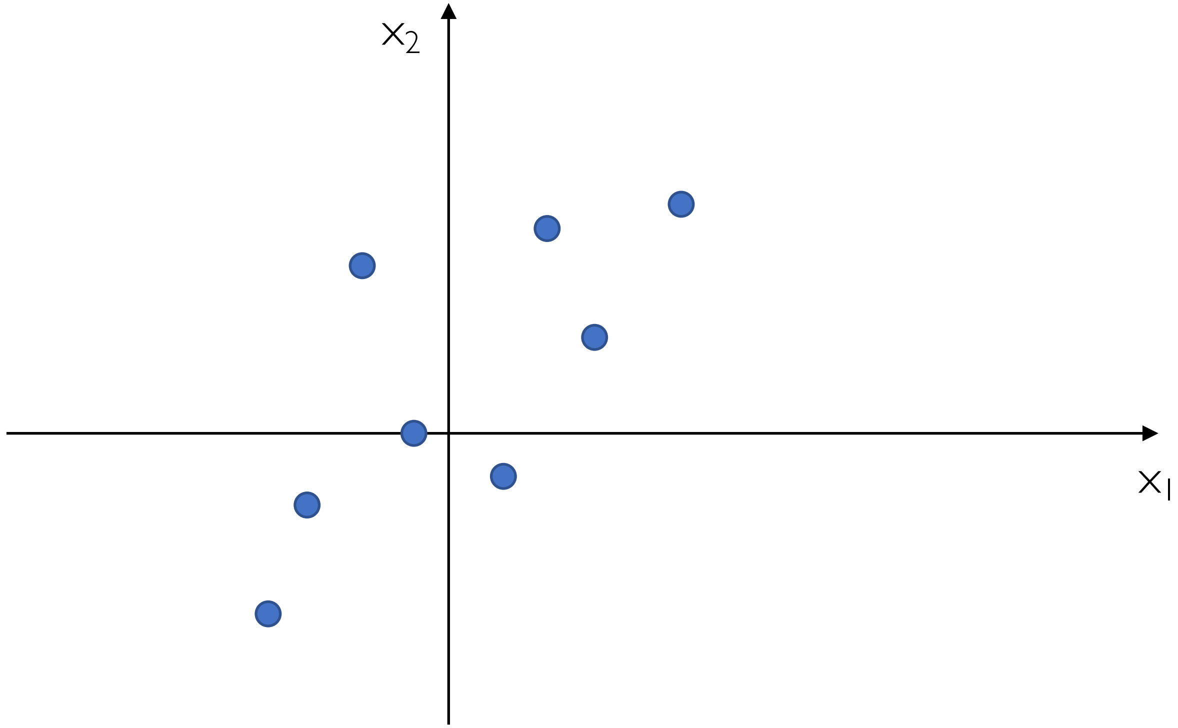
Why Do We Look for Greatest Variance?

Example: Reduce 2-dimensional data to 1-d



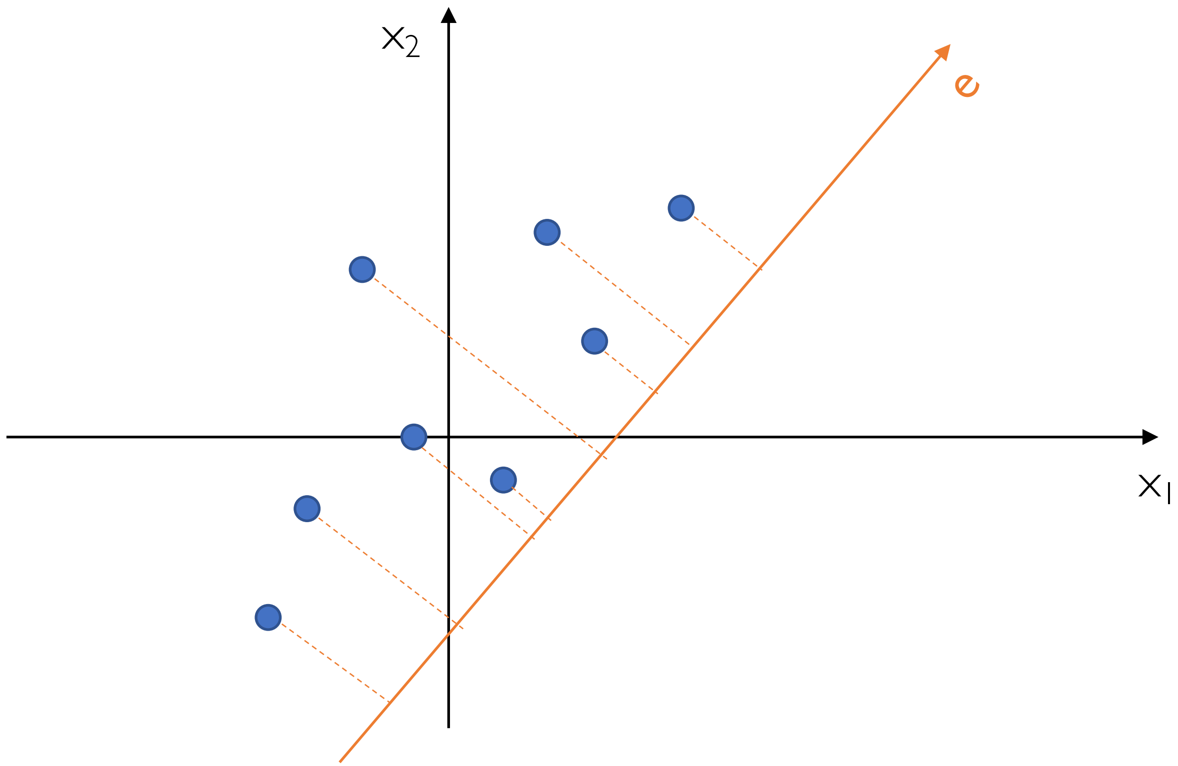
Why Do We Look for Greatest Variance?

First of all, let's center the points around the mean along x_1 and x_2



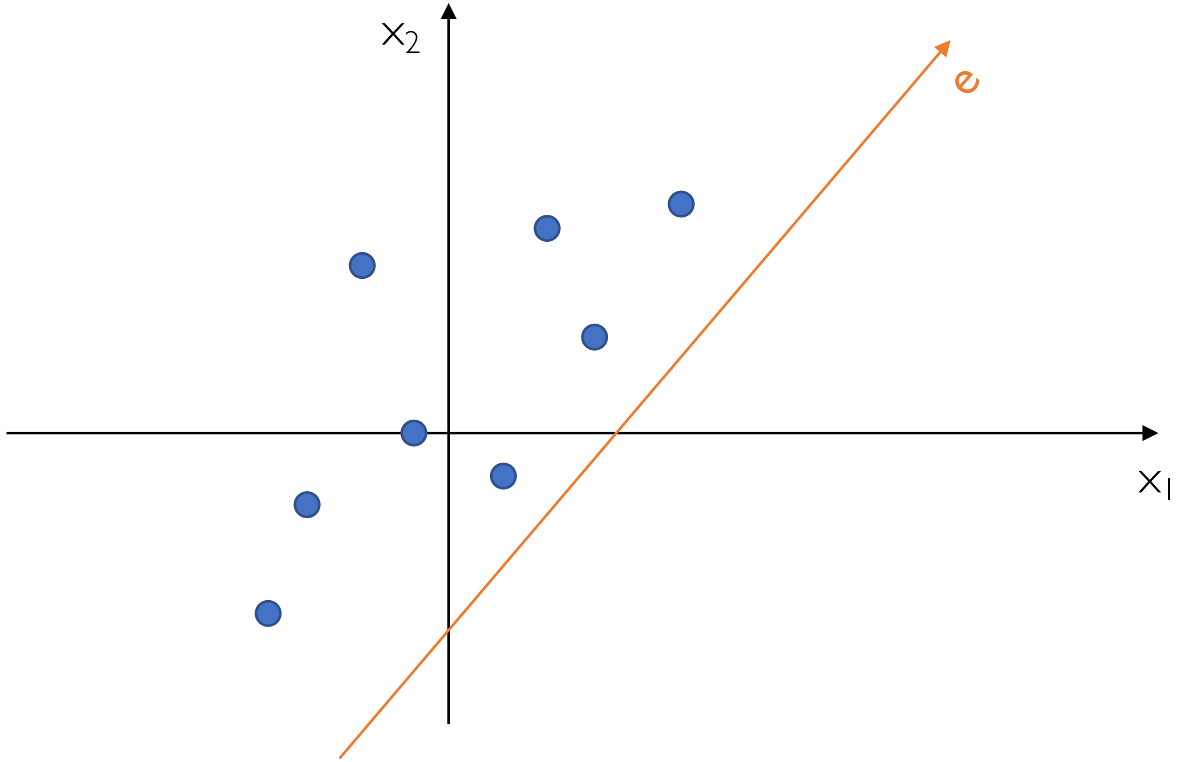
Why Do We Look for Greatest Variance?

Map, i.e., project (x_1, x_2) to a new single dimension axis **e**



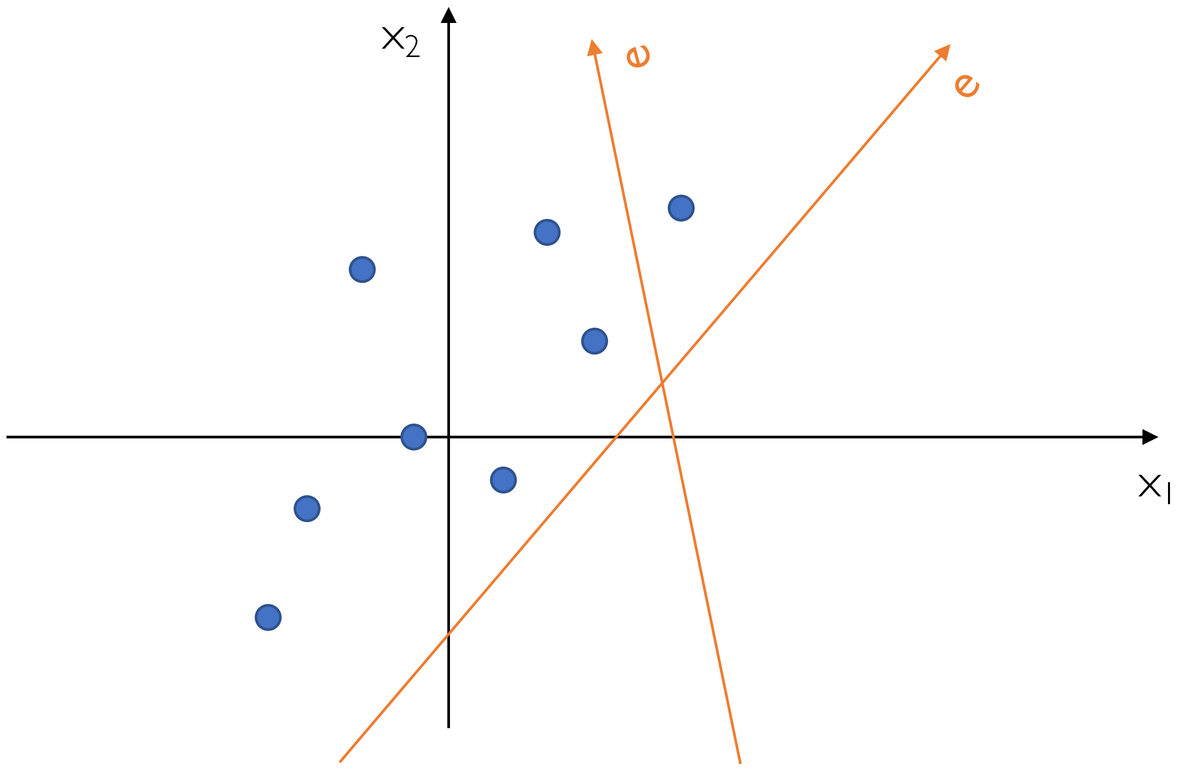
Why Do We Look for Greatest Variance?

Map, i.e., project (x_1, x_2) to a new single dimension axis $\mathbf{e}$



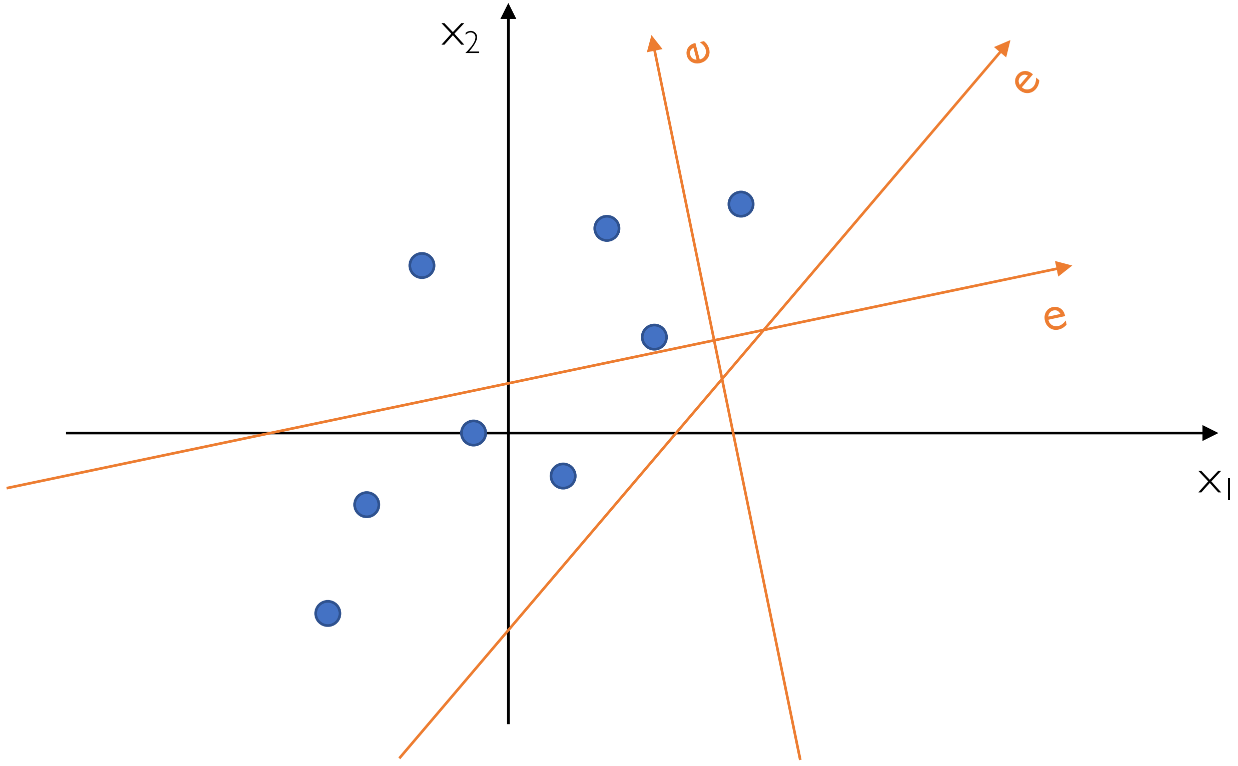
Why Do We Look for Greatest Variance?

Map, i.e., project (x_1, x_2) to a new single dimension axis **e**



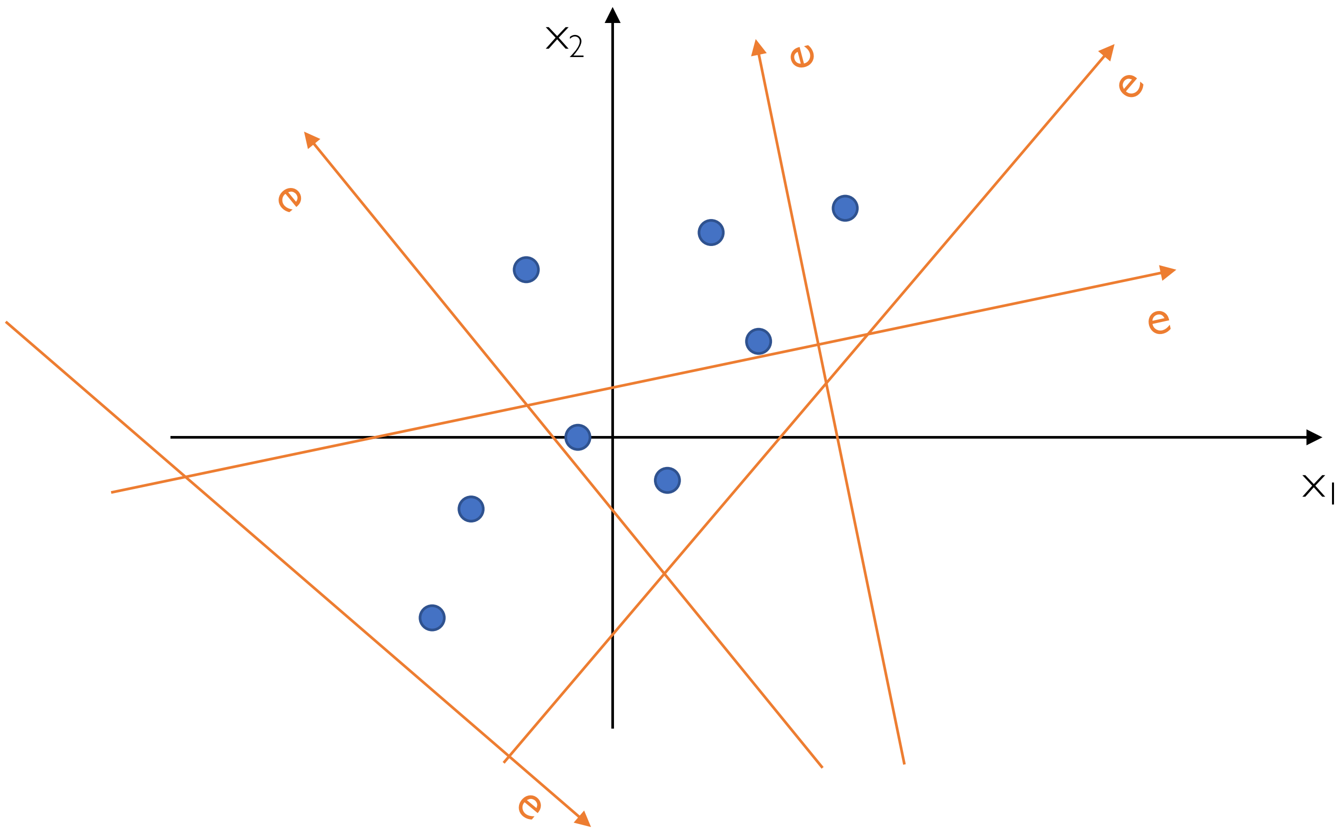
Why Do We Look for Greatest Variance?

Map, i.e., project (x_1, x_2) to a new single dimension axis **e**



Why Do We Look for Greatest Variance?

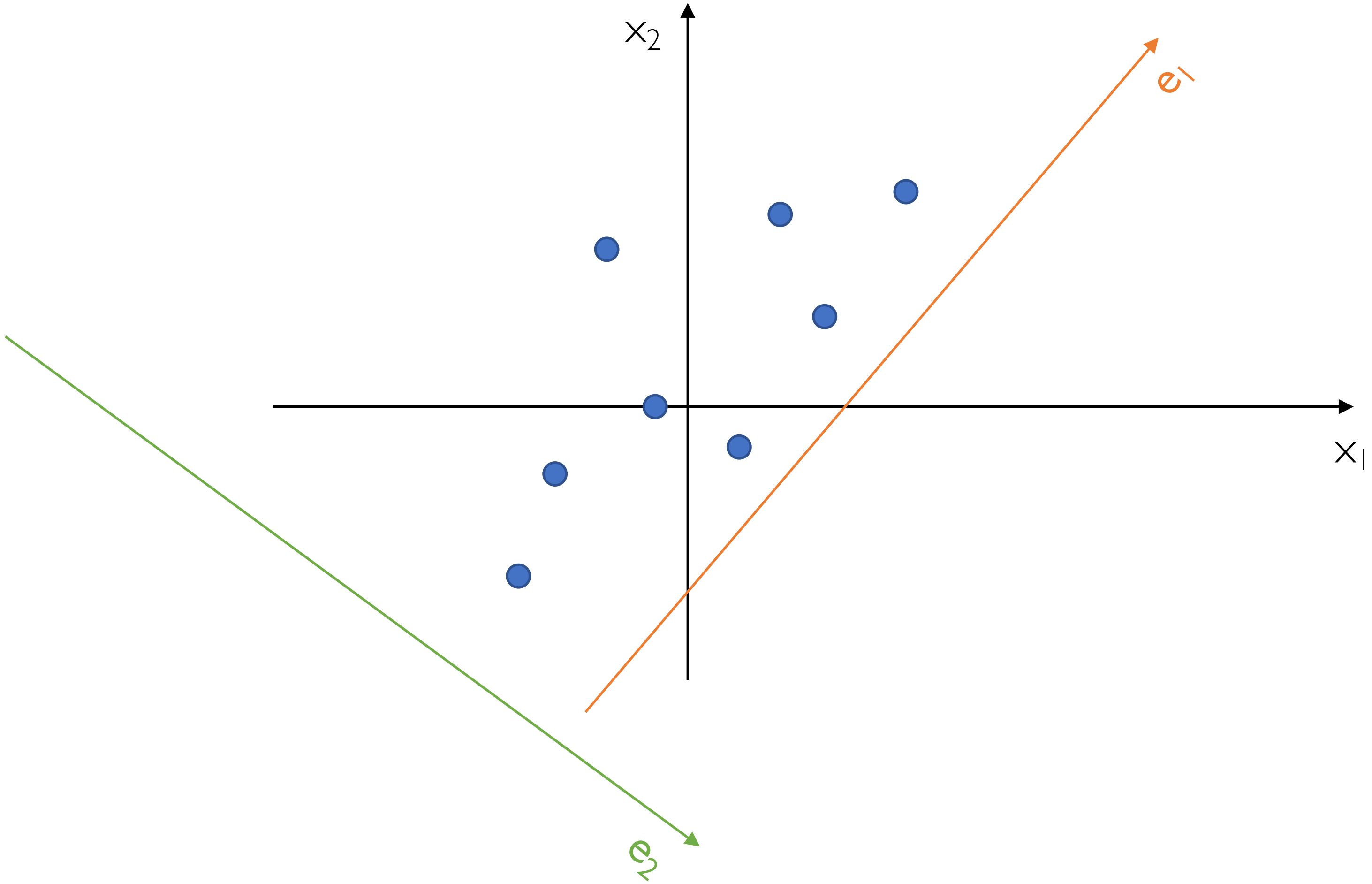
Map, i.e., project (x_1, x_2) to a new single dimension axis **e**



infinitely many mappings from (x_1, x_2) to a new axis **e**

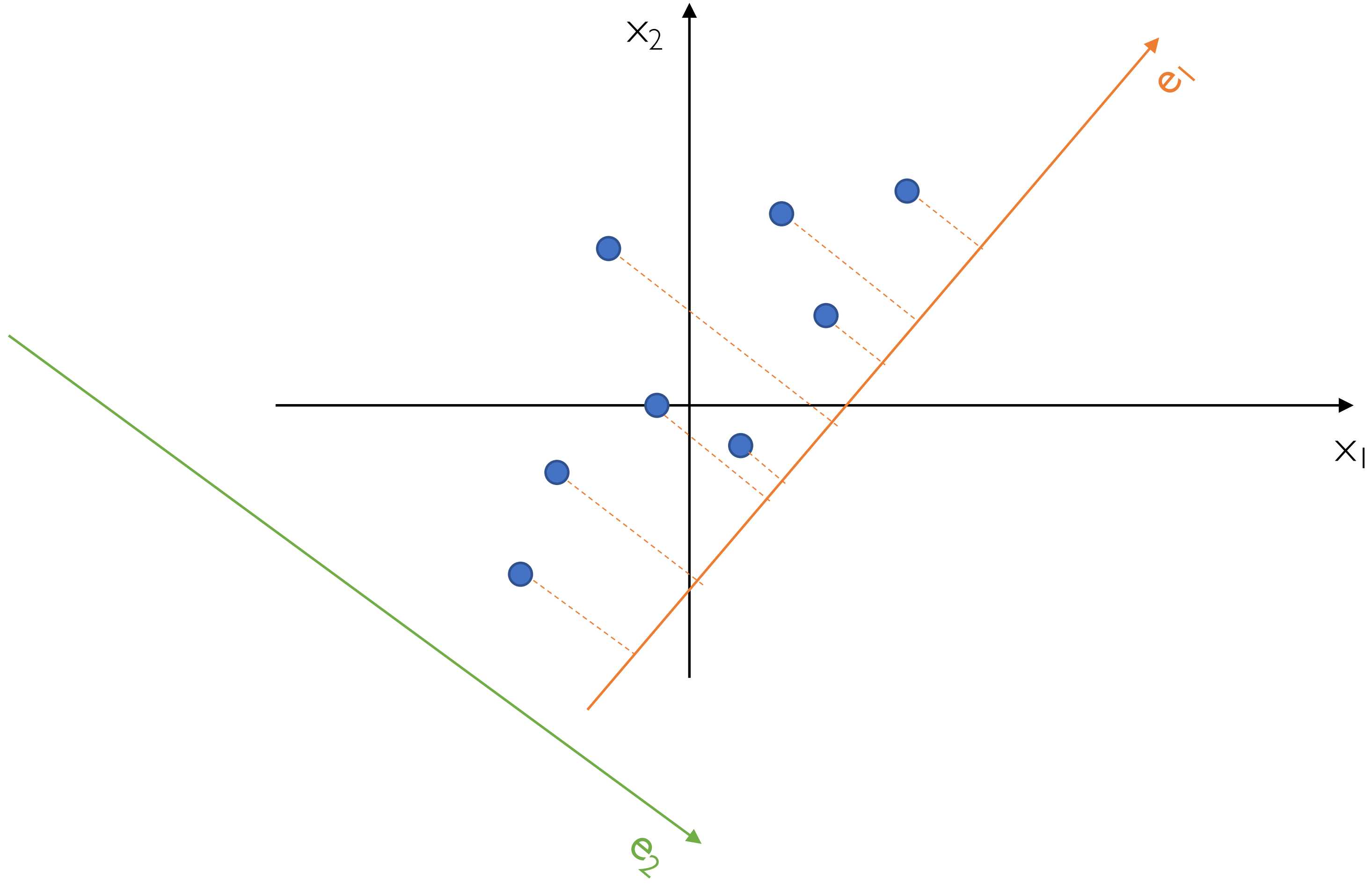
Why Do We Look for Greatest Variance?

Let's consider 2 different mappings e_1 and e_2



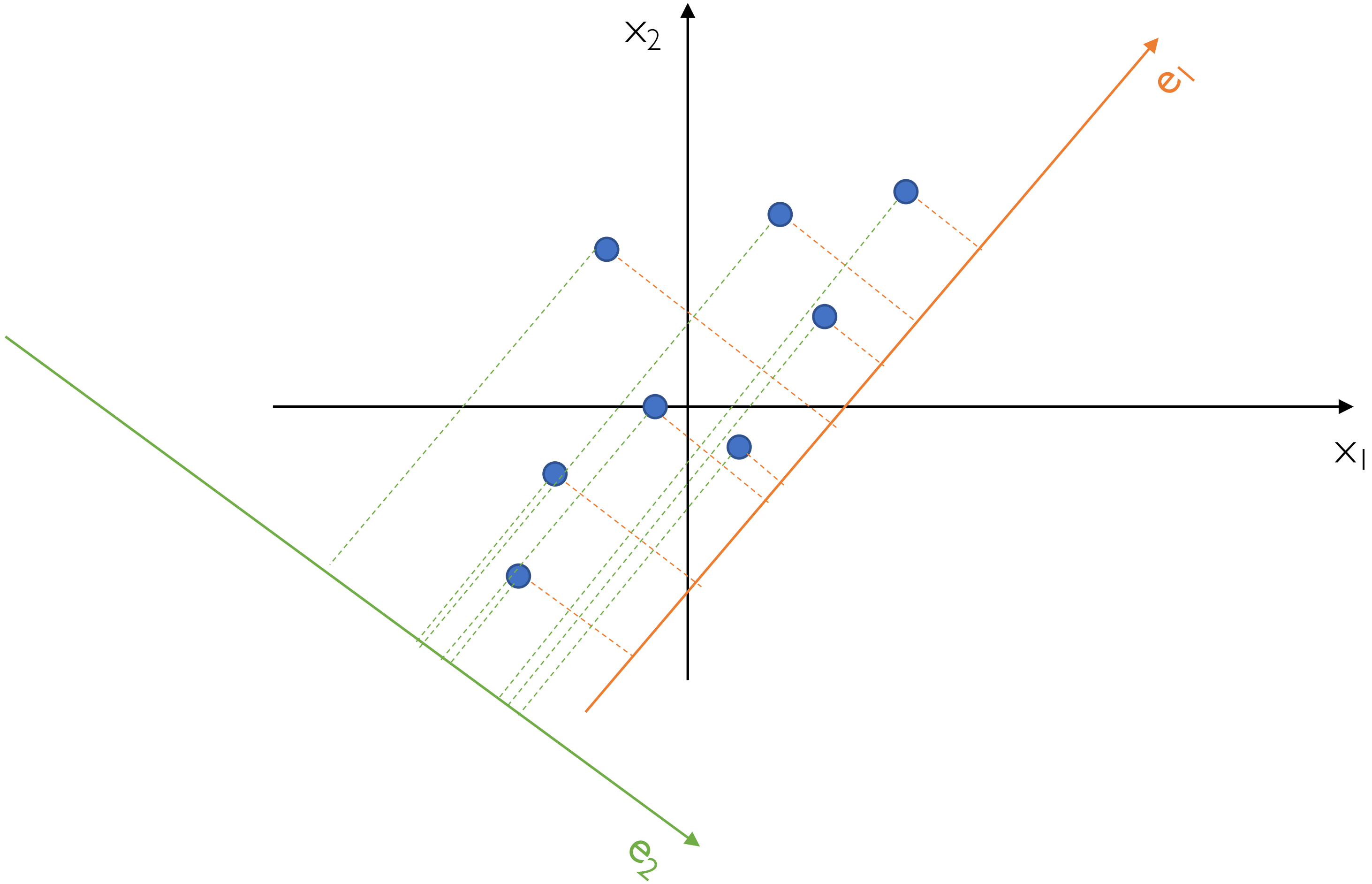
Why Do We Look for Greatest Variance?

Let's consider 2 different mappings e_1 and e_2



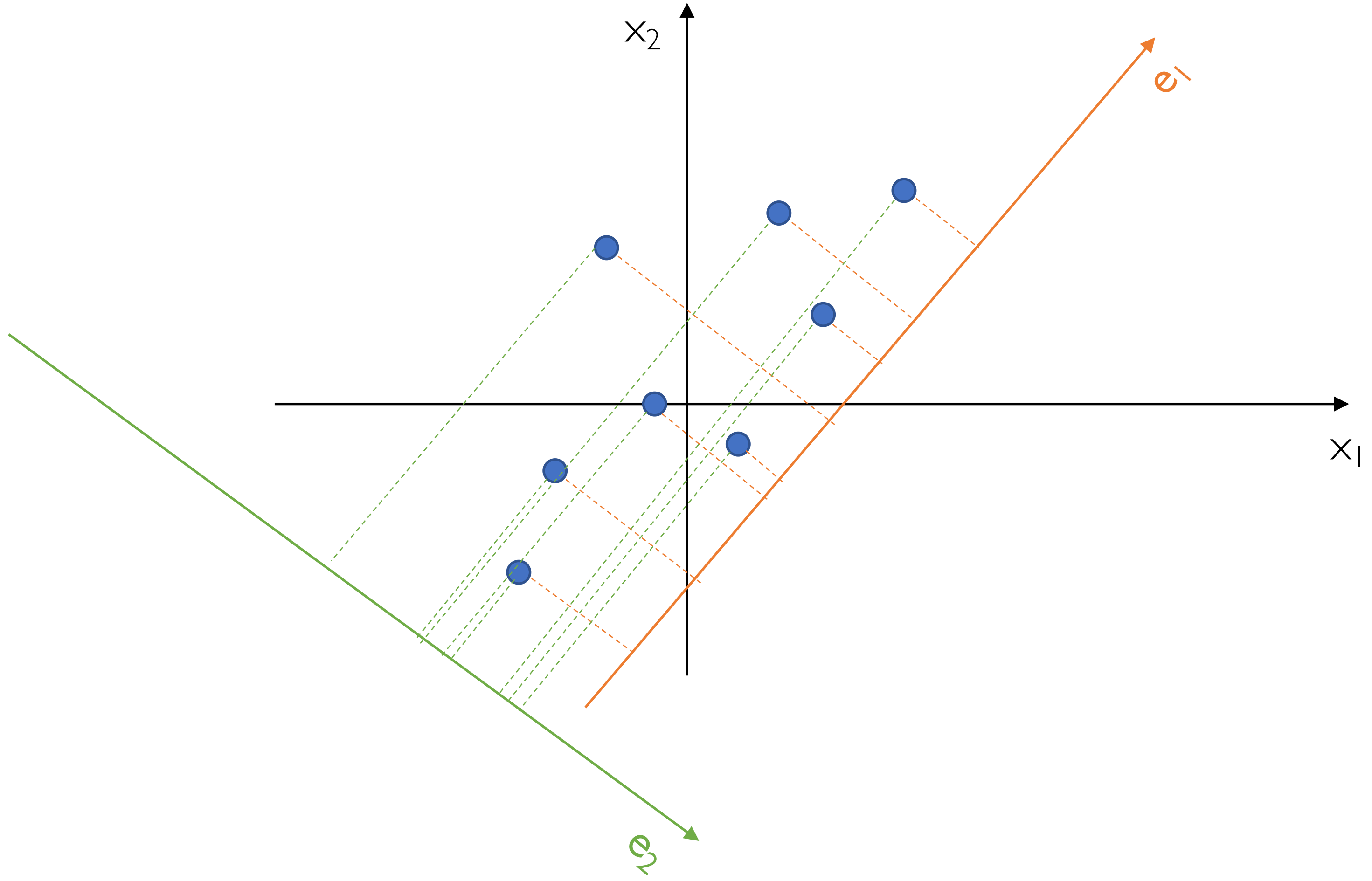
Why Do We Look for Greatest Variance?

Let's consider 2 different mappings e_1 and e_2



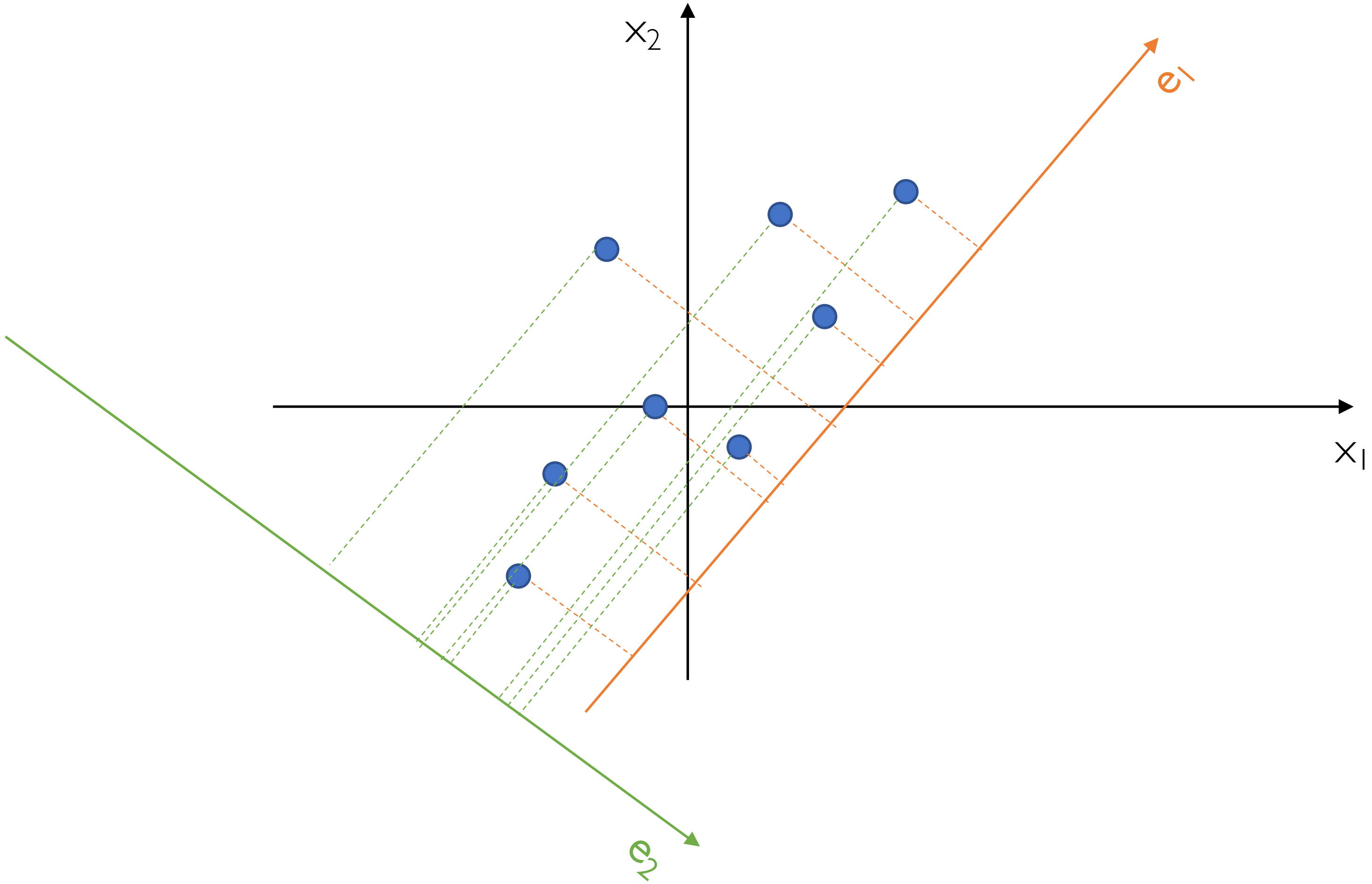
Why Do We Look for Greatest Variance?

Which one is better?



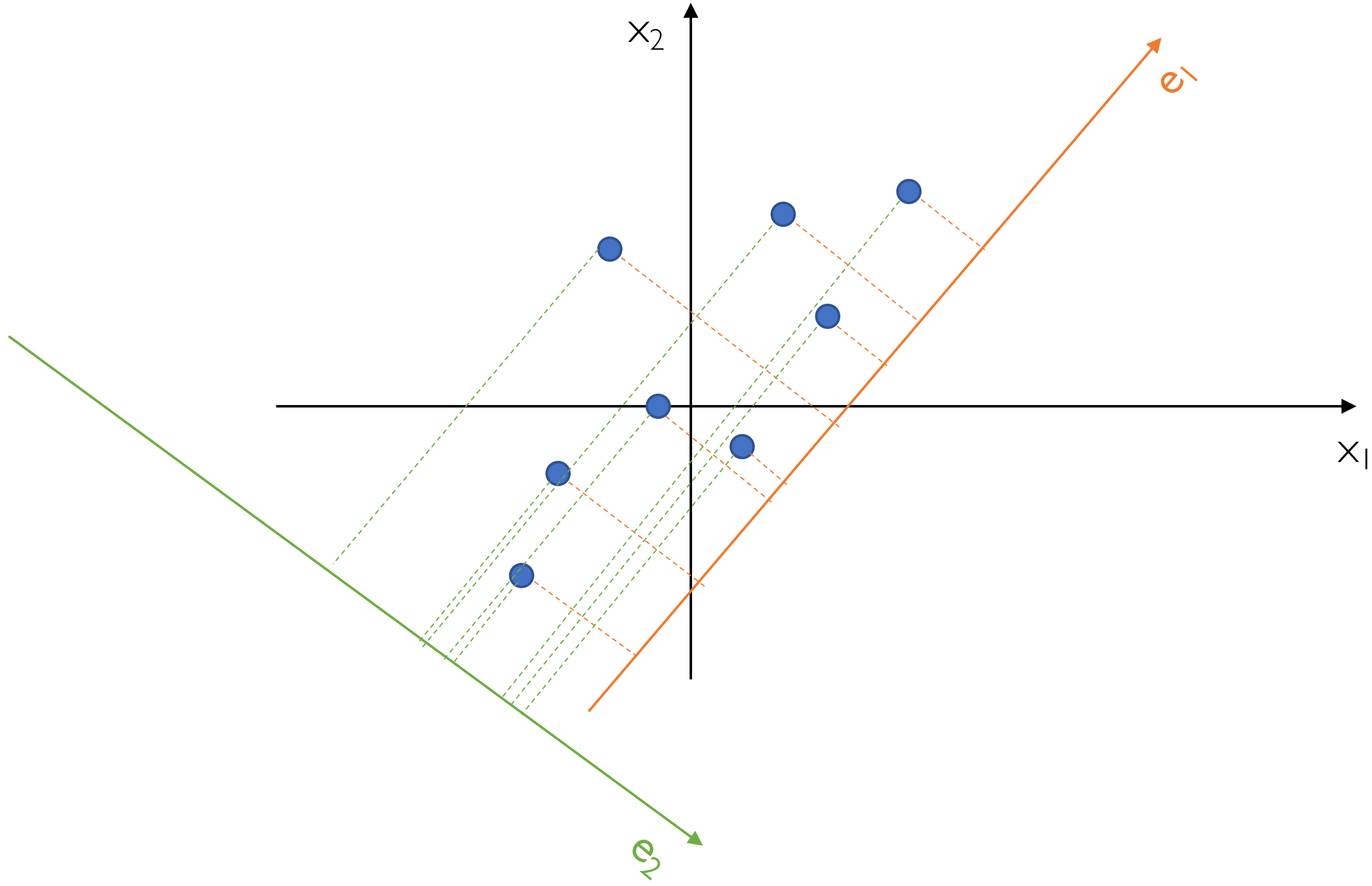
Why Do We Look for Greatest Variance?

Points projected onto e_1 look more **spread-out** than onto e_2



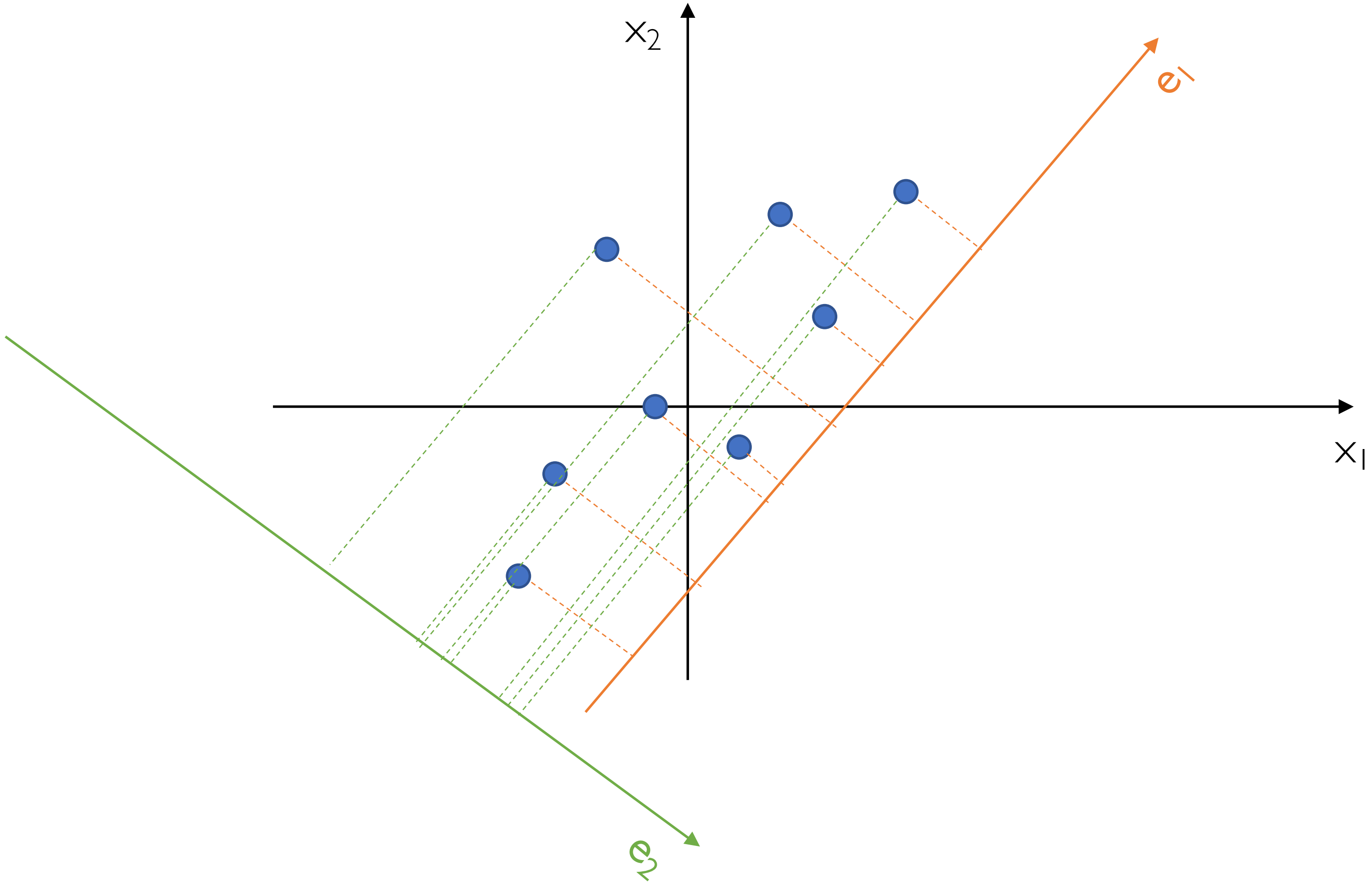
Why Do We Look for Greatest Variance?

The variance along e_1 is larger than along e_2



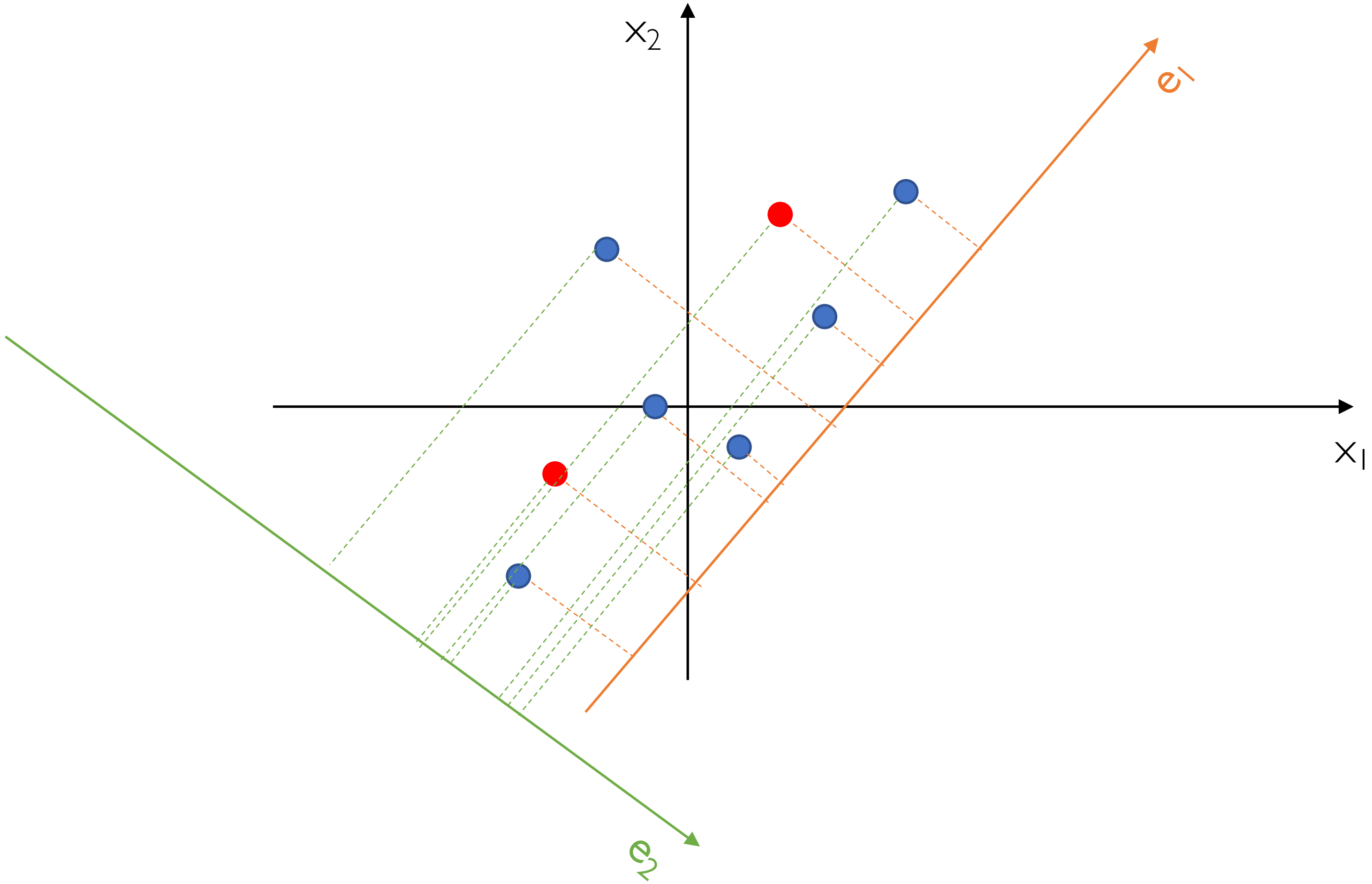
Why Do We Look for Greatest Variance?

Why is that good?



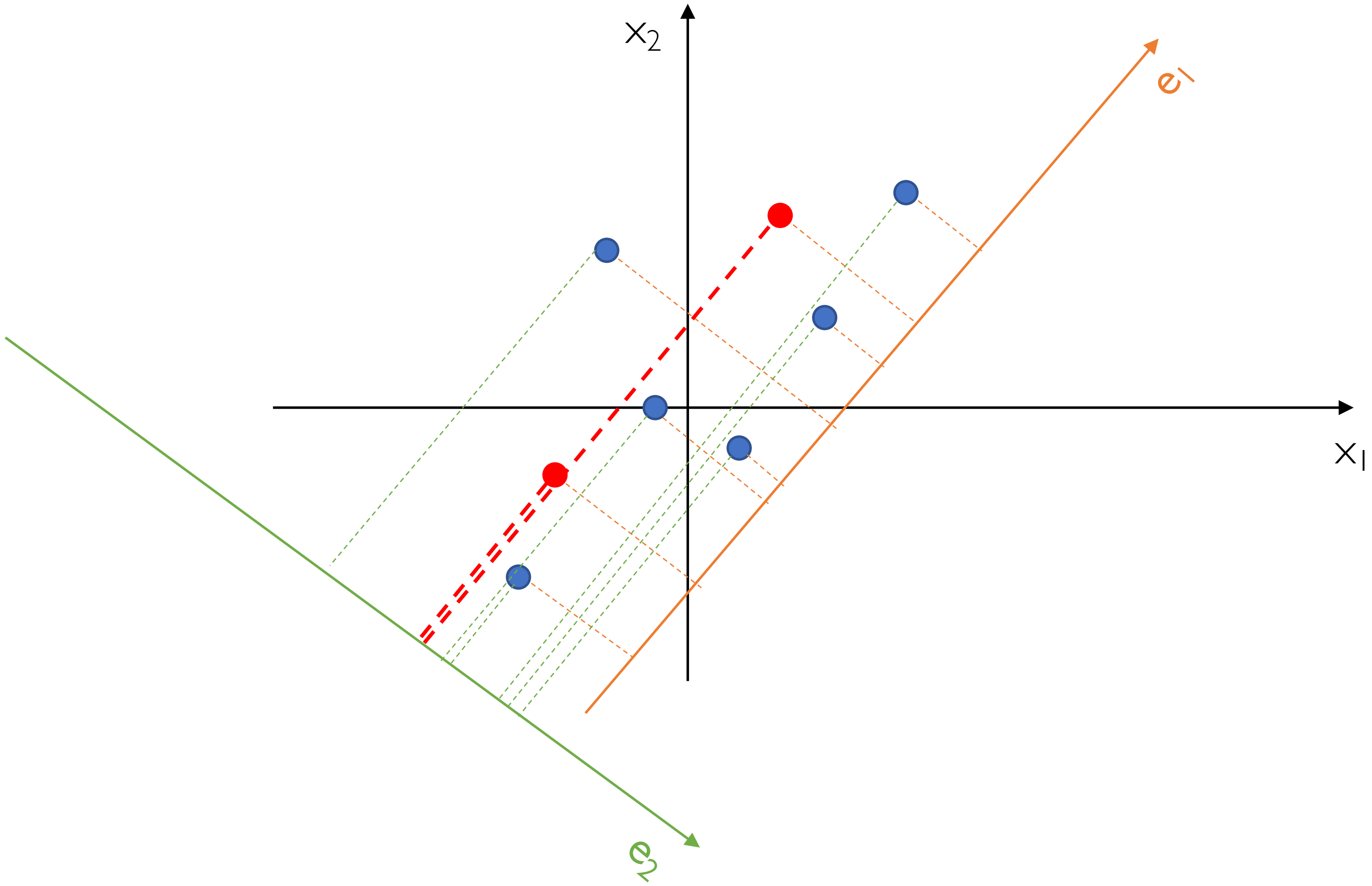
Why Do We Look for Greatest Variance?

Consider the 2 red points below



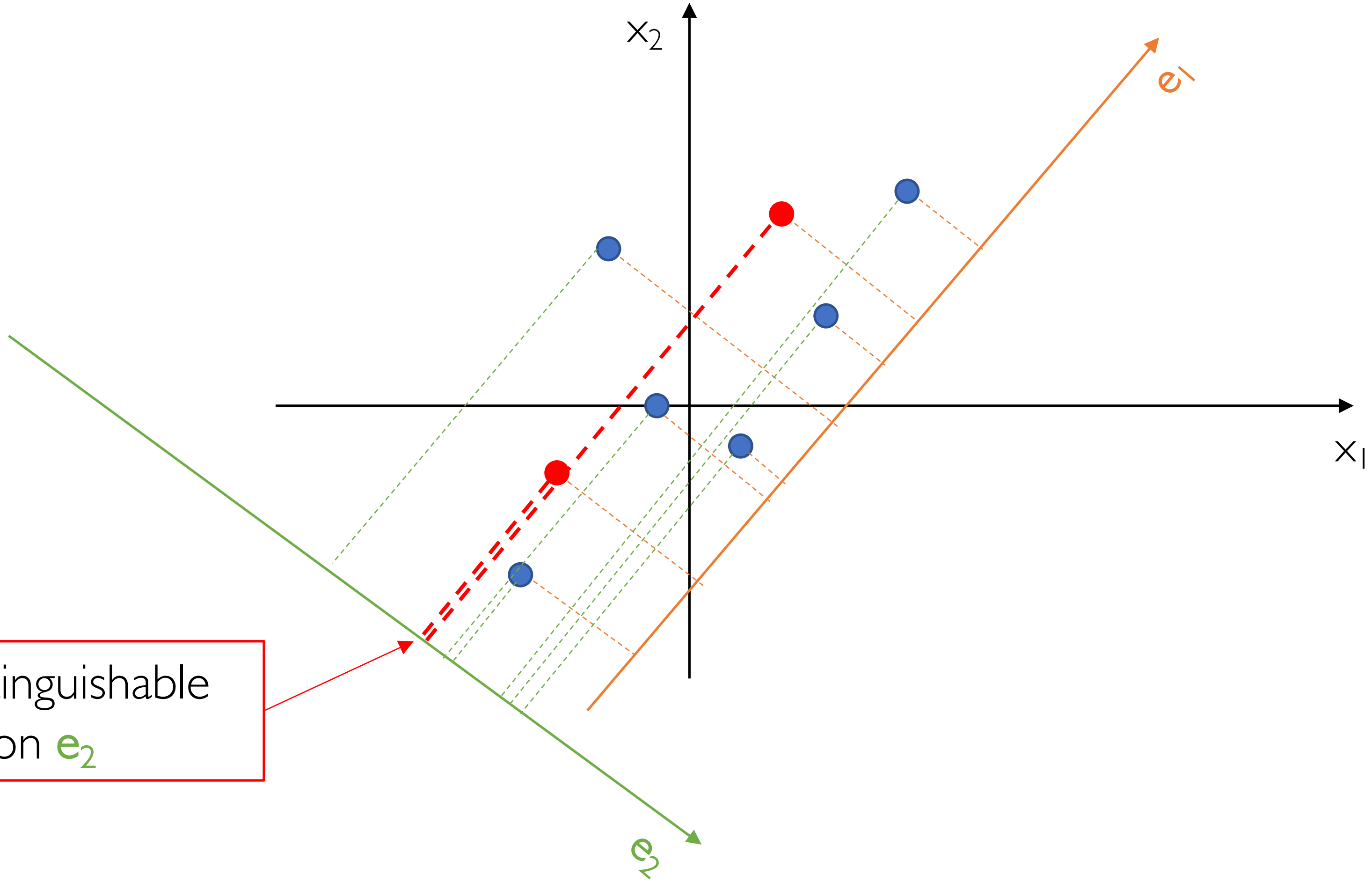
Why Do We Look for Greatest Variance?

On (x_1, x_2) far away from each other, end up close if projected onto e_2



Why Do We Look for Greatest Variance?

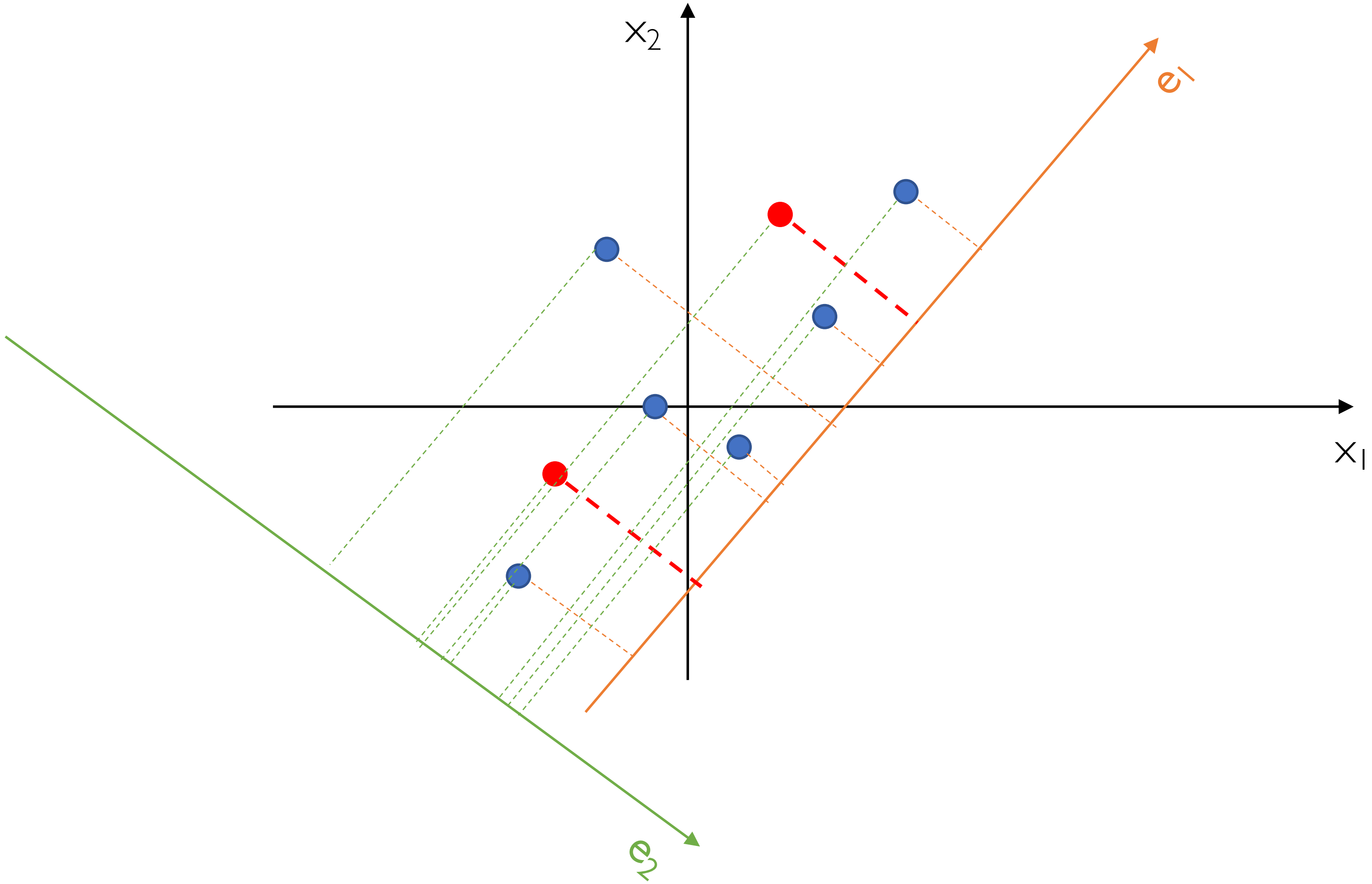
On (x_1, x_2) far away from each other, end up close if projected onto e_2



Almost indistinguishable points on e_2

Why Do We Look for Greatest Variance?

If projected onto e_1 they better preserve their distance



Why Do We Look for Greatest Variance?

- Intuitively, we want to minimize the chance that 2 points that are far in the original space end up close in the lower dimensional space

Why Do We Look for Greatest Variance?

- Intuitively, we want to minimize the chance that 2 points that are far in the original space end up close in the lower dimensional space
- Minimize distances between points as measured on (x_1, x_2) space and those measured on **e**

Why Do We Look for Greatest Variance?

- Intuitively, we want to minimize the chance that 2 points that are far in the original space end up close in the lower dimensional space
- Minimize distances between points as measured on (x_1, x_2) space and those measured on e

⇩

Solution

Pick e so as to maximize variance of projected data

Variance of a Random Variable

- The variance of a random variable X measures how far a set of (random) numbers are spread out from their mean value

Variance of a Random Variable

- The variance of a random variable X measures how far a set of (random) numbers are spread out from their mean value
- Formally, it is the expected value of the squared deviation from its mean

$$\mathrm{Var}(X) = \mathrm{E}[(X - \mu)^2]$$

$$\text{where } \mu = \mathrm{E}[X]$$

Covariance of Two Random Variables

- A measure of the joint variability of two random variables X and Y
  - Do X and Y increase/decrease together, or when one increases/decreases the other decreases/increases?

Covariance of Two Random Variables

- A measure of the joint variability of two random variables X and Y
  - Do X and Y increase/decrease together, or when one increases/decreases the other decreases/increases?
- Formally, it is the expected value of the product of their deviations from their individual means

$$\mathrm{Cov}(X,Y) = \mathrm{E}[(X-\mu_X)(Y-\mu_Y)]$$

$$\boxed{\mathrm{Cov}(X,X) = \mathrm{Var}(X)}$$

$$\text{where } \mu_X = \mathrm{E}[X] \text{ and } \mu_Y = \mathrm{E}[Y]$$

Covariance Matrix

- Given a random vector $\mathbf{X} = (X_1, \ldots, X_d)$ its covariance matrix K is a dxd square matrix with the covariance between each pair of elements

Covariance Matrix

- Given a random vector $\mathbf{X} = (X_1, \ldots, X_d)$ its covariance matrix K is a dxd square matrix with the covariance between each pair of elements
- In the matrix diagonal there are variances, i.e., the covariance of each element with itself

$$\mathrm{K}[i, j] = \mathrm{Cov}(X_i, X_j)$$

Covariance Matrix of Original Dimensions

- The original set of dimensions is a random vector $\mathbf{X} = (X_1, \ldots, X_d)$

Covariance Matrix of Original Dimensions

- The original set of dimensions is a random vector $\mathbf{X} = (X_1, \ldots, X_d)$
- In our example, $d = 2$ and $\mathbf{X} = (X_1, X_2)$

Covariance Matrix of Original Dimensions

- The original set of dimensions is a random vector $\mathbf{X} = (X_1, \ldots, X_d)$
- In our example, $d = 2$ and $\mathbf{X} = (X_1, X_2)$
- The covariance matrix K is a 2-by-2 matrix

Covariance Matrix of Original Dimensions

- The original set of dimensions is a random vector $\mathbf{X} = (X_1, \ldots, X_d)$
- In our example, $d = 2$ and $\mathbf{X} = (X_1, X_2)$
- The covariance matrix K is a 2-by-2 matrix
- To ease the covariance computation, we center each data point at zero
  - Subtracting the mean of each attribute/dimension
  - The mean of each dimension becomes then 0

Covariance Matrix of Original Dimensions

Let n be the total number of data points: $\mathbf{x}_1, \ldots, \mathbf{x}_n$

Each data point is represented by a (x_1, x_2) pair

$\mathbf{x}_i = (x_{i,1}, x_{i,2})$

We associate 2 random variables X_1, X_2 to each dimension, and we compute:

$$\mu_1 = \mathrm{E}[X_1] = \frac{1}{n}\sum_{i=1}^{n} x_{i,1}$$

$$\mu_2 = \mathrm{E}[X_2] = \frac{1}{n}\sum_{i=1}^{n} x_{i,2}$$

$$\mathbf{x}_i = (x_{i,1} - \mu_1, x_{i,2} - \mu_2)$$

Covariance Matrix of Original Dimensions

Let us rewrite each data point $\mathbf{x}_i$ as follows:

$\mathbf{x}_i = (x'_{i,1}, x'_{i,2})$ where:

$x'_{i,1} = x_{i,1} - \mu_1; x'_{i,2} = x_{i,2} - \mu_2$

$$\mu_1^{\text{new}} = \mathrm{E}[X_1] = \frac{1}{n}\sum_{i=1}^{n} x'_{i,1} = \frac{1}{n}\sum_{i=1}^{n}(x_{i,1} - \mu_1)$$

$$\mu_2^{\text{new}} = \mathrm{E}[X_2] = \frac{1}{n}\sum_{i=1}^{n} x'_{i,2} = \frac{1}{n}\sum_{i=1}^{n}(x_{i,2} - \mu_2)$$

Covariance Matrix of Original Dimensions

$$\mu_1^{\text{new}} = \frac{1}{n}\sum_{i=1}^{n}(x_{i,1}-\mu_1) = \frac{1}{n}\left(\underbrace{\sum_{i=1}^{n} x_{i,1}}_{n\mu_1} - \underbrace{\sum_{i=1}^{n} \mu_1}_{n\mu_1}\right) = 0$$

$$\mu_2^{\text{new}} = \frac{1}{n}\sum_{i=1}^{n}(x_{i,2}-\mu_2) = \frac{1}{n}\left(\underbrace{\sum_{i=1}^{n} x_{i,2}}_{n\mu_2} - \underbrace{\sum_{i=1}^{n} \mu_2}_{n\mu_2}\right) = 0$$

0-mean

Covariance Matrix of Original Dimensions

Scaling data so as to have 0-mean on all dimensions allow computing covariance much easily

$$\mathrm{Cov}(X_1, X_2) = \mathrm{E}[(X_1 - \underbrace{\mu_1^{\mathrm{new}}}_{=0})(X_2 - \underbrace{\mu_2^{\mathrm{new}}}_{=0})] = \mathrm{E}[X_1 X_2]$$

Covariance Matrix of Original Dimensions

Scaling data so as to have 0-mean on all dimensions
allow computing covariance much easily

$$\mathrm{Cov}(X_1, X_2) = \mathrm{E}[(X_1 - \underbrace{\mu_1^{\text{new}}}_{=0})(X_2 - \underbrace{\mu_2^{\text{new}}}_{=0})] = \mathrm{E}[X_1 X_2]$$

As a consequence, the covariance matrix is also easier to compute!

Covariance Matrix of Original Dimensions

Let's assume the following is our 2-by-2 covariance matrix

$$
\begin{array}{cc}
 & \begin{array}{cc} x_1 & x_2 \end{array} \\
\begin{array}{c} x_1 \\ x_2 \end{array} & \begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}
\end{array}
$$

Covariance Matrix of Original Dimensions

Let's assume the following is our 2-by-2 covariance matrix

$$\begin{array}{cc} & \begin{array}{cc} x_1 & x_2 \end{array} \\ \begin{array}{c} x_1 \\ x_2 \end{array} & \begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix} \end{array}$$

$$\boxed{\mathrm{Cov}(X_1, X_2) = \frac{1}{n}\sum_{i=1}^{n} x'_{i,1} * x'_{i,2}}$$

Covariance Matrix of Original Dimensions

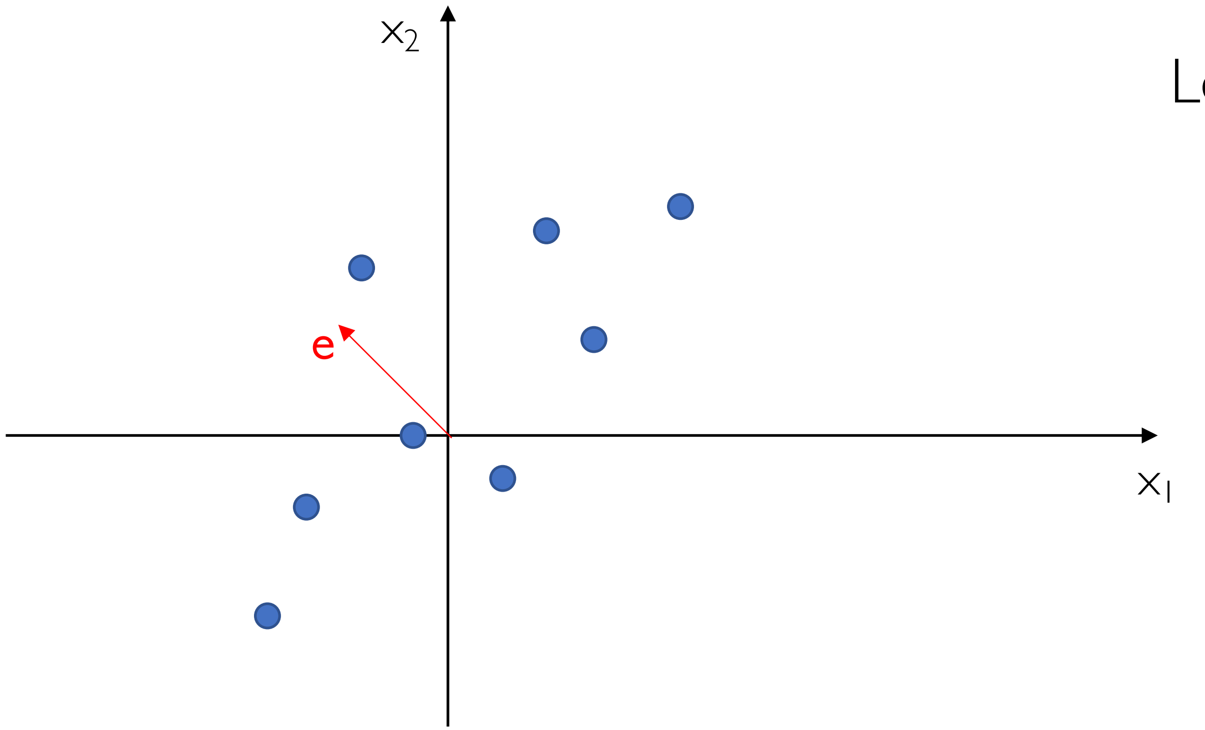
Let's assume the following is our 2-by-2 covariance matrix

$$\begin{array}{c|cc} & x_1 & x_2 \\ \hline x_1 & 2 & 4/5 \\ x_2 & 4/5 & 3/5 \end{array} \quad \text{i.e.} \quad \begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}$$

$$\boxed{\mathrm{Cov}(X_1, X_2) = \frac{1}{n}\sum_{i=1}^{n} x'_{i,1} * x'_{i,2}}$$

$$\boxed{\mathrm{Cov}(X_2, X_2) = \mathrm{Var}(X_2) = \frac{1}{n}\sum_{i=1}^{n} (x'_{i,2})^2}$$

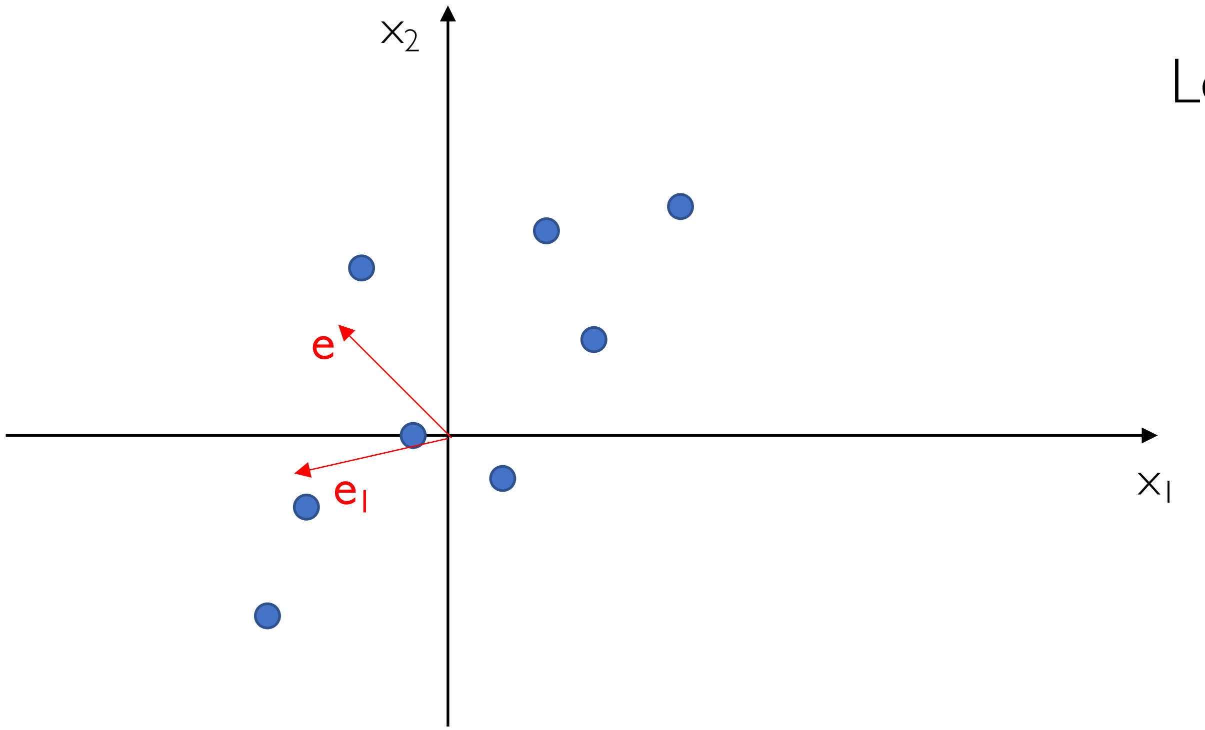
Covariance Matrix of Original Dimensions



Let's multiply our 2-by-2 covariance matrix K by a random vector e = (-1,1)

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{e}} =$$

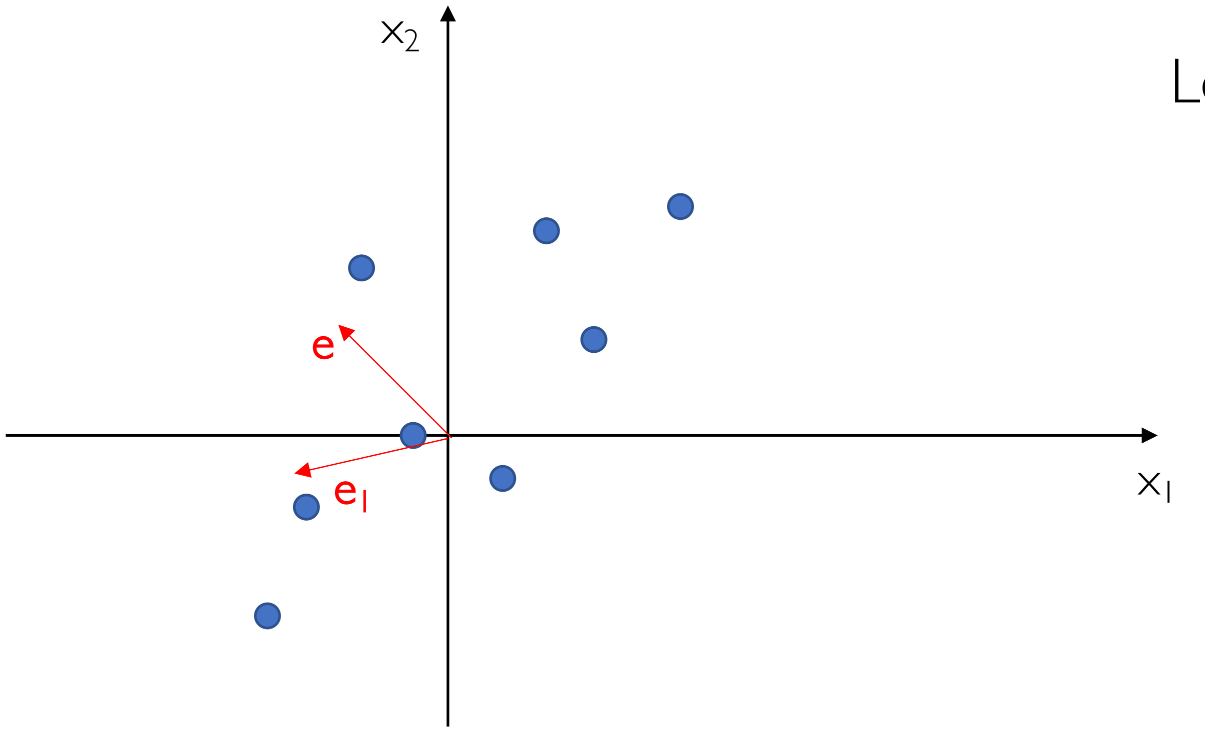
Covariance Matrix of Original Dimensions



Let's multiply our 2-by-2 covariance matrix K by a random vector **e** = (-1,1)

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{e}} = \underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}}$$

Covariance Matrix of Original Dimensions

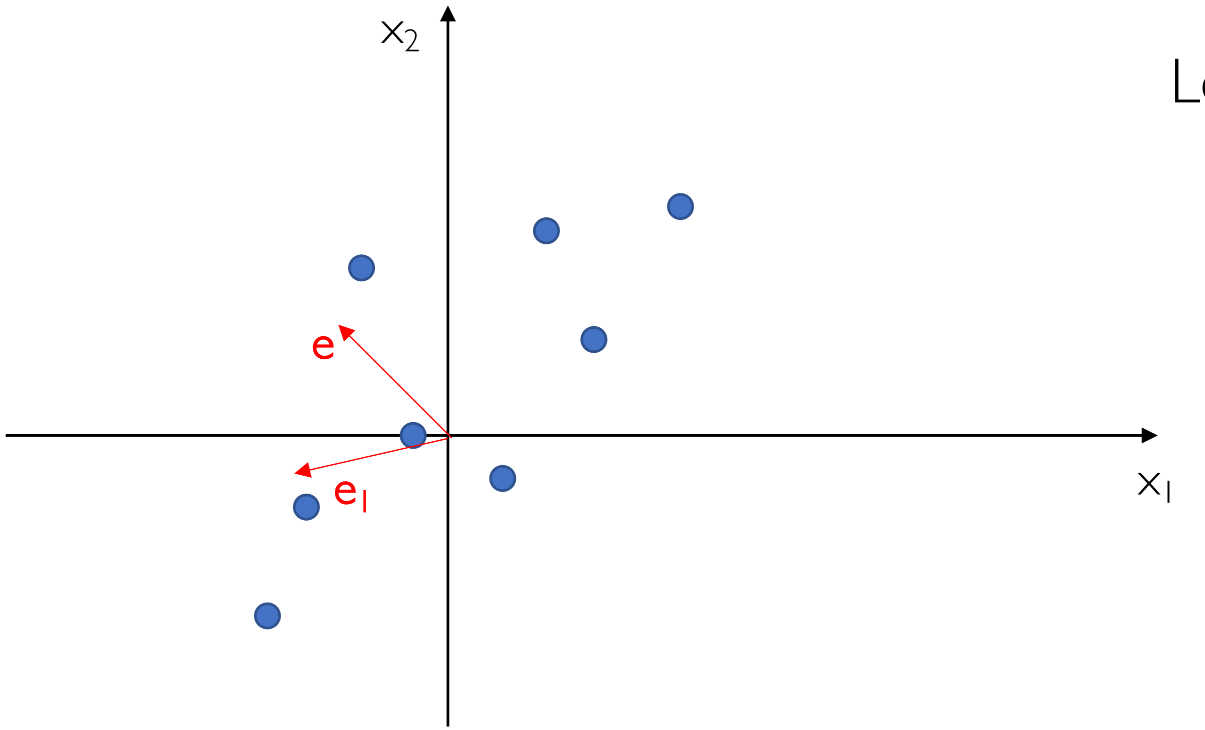


Let's multiply our 2-by-2 covariance matrix K by a random vector **e** = (-1,1)

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{e}} = \underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}}$$

$\boxed{\text{slope} = 1/\text{-}1 = \text{-}1}$

Covariance Matrix of Original Dimensions



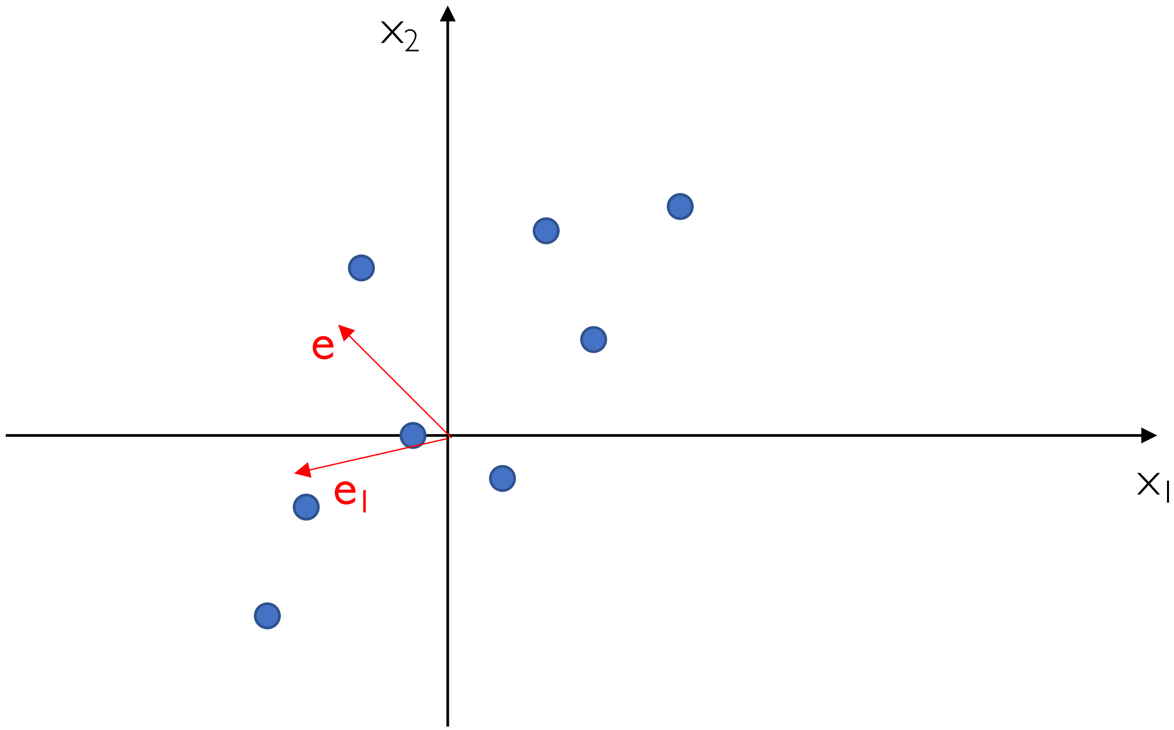
Let's multiply our 2-by-2 covariance matrix K by a random vector e = (-1,1)

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{e}} = \underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}}$$

slope = 1/-1 = -1

new slope = -(1/5)/-(6/5) = 1/6

Covariance Matrix of Original Dimensions



Let's multiply our 2-by-2 covariance matrix K by a random vector e = (-1,1)

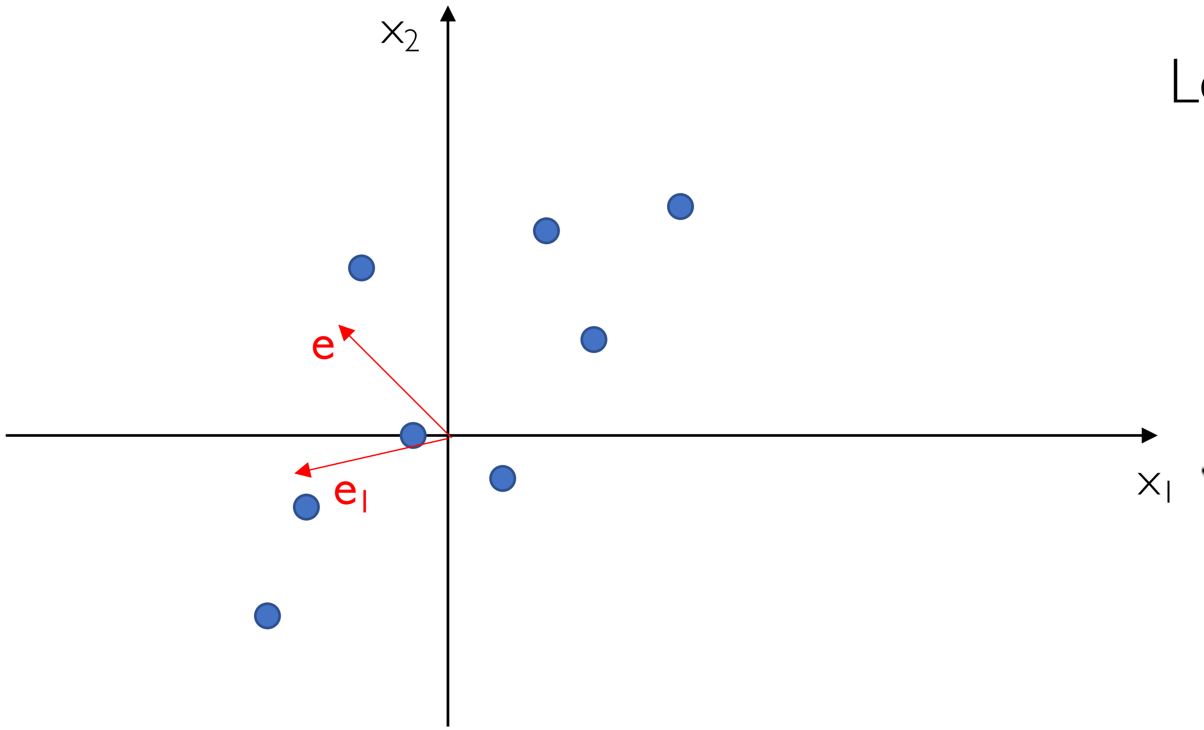
$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{e}} = \underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}}$$

slope = 1/-1 = -1

Turns towards the direction of the greatest variance

new slope = -(1/5)/-(6/5) = 1/6

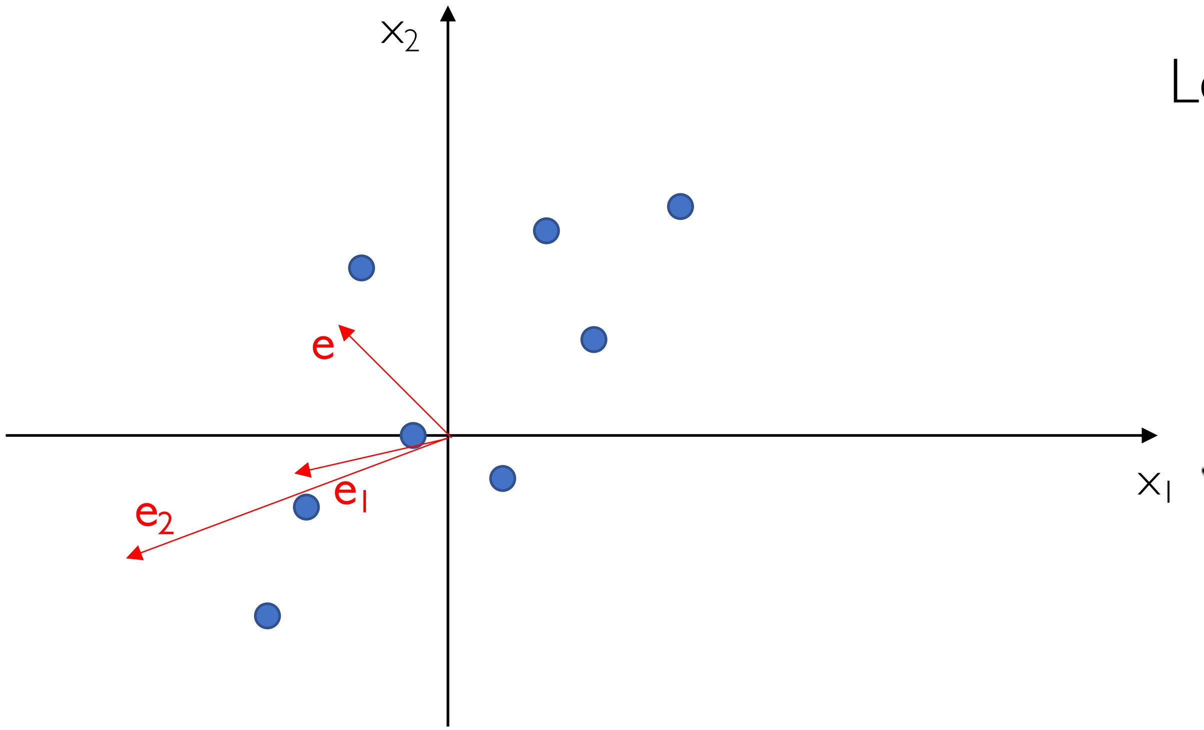
Covariance Matrix of Original Dimensions



Let's repeat the previous step multiplying the covariance matrix K by $\mathbf{e}_1$ = (-6/5,-1/5)

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K}\underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}} = \qquad ,$$

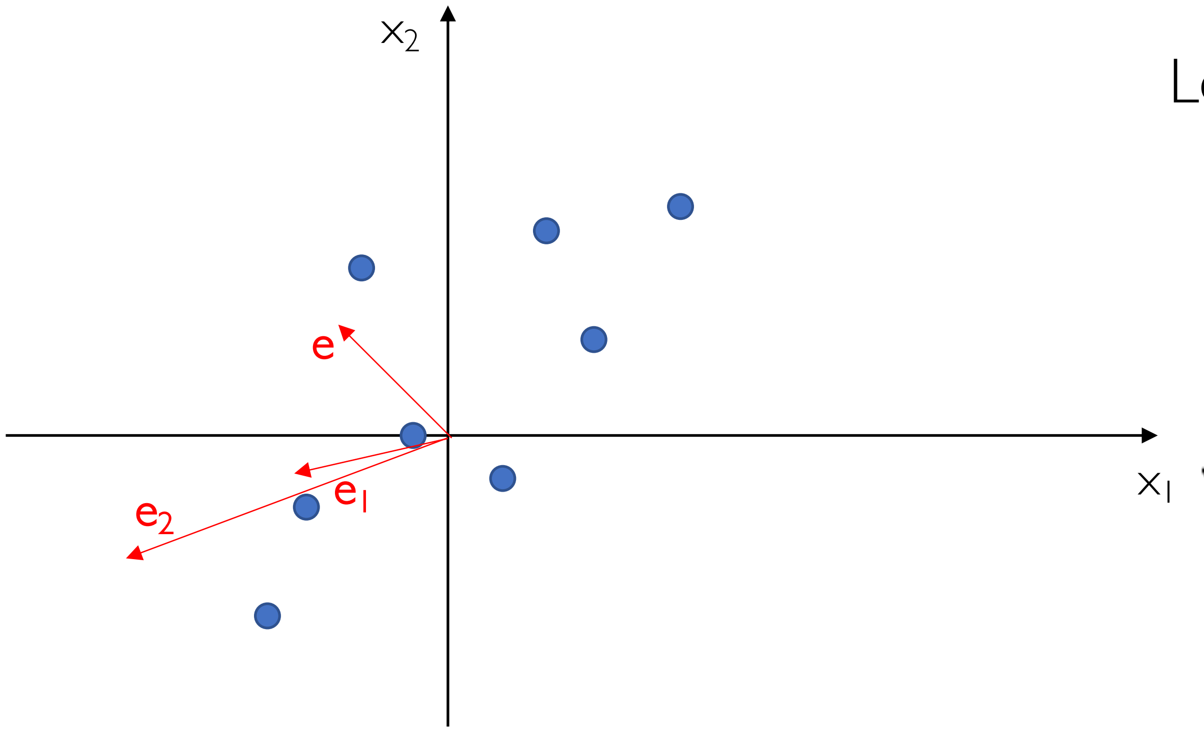
Covariance Matrix of Original Dimensions



Let's repeat the previous step multiplying the covariance matrix K by $\mathbf{e}_1 = (-6/5,-1/5)$

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}} = \underbrace{\begin{bmatrix} -64/25 \\ -27/25 \end{bmatrix}}_{\mathbf{e_2}}$$

Covariance Matrix of Original Dimensions



Let's repeat the previous step multiplying the covariance matrix K by $\mathbf{e}_1$ = (-6/5,-1/5)

$$\underbrace{\begin{bmatrix} 2 & 4/5 \\ 4/5 & 3/5 \end{bmatrix}}_{K} \underbrace{\begin{bmatrix} -6/5 \\ -1/5 \end{bmatrix}}_{\mathbf{e_1}} = \underbrace{\begin{bmatrix} -64/25 \\ -27/25 \end{bmatrix}}_{\mathbf{e_2}}$$

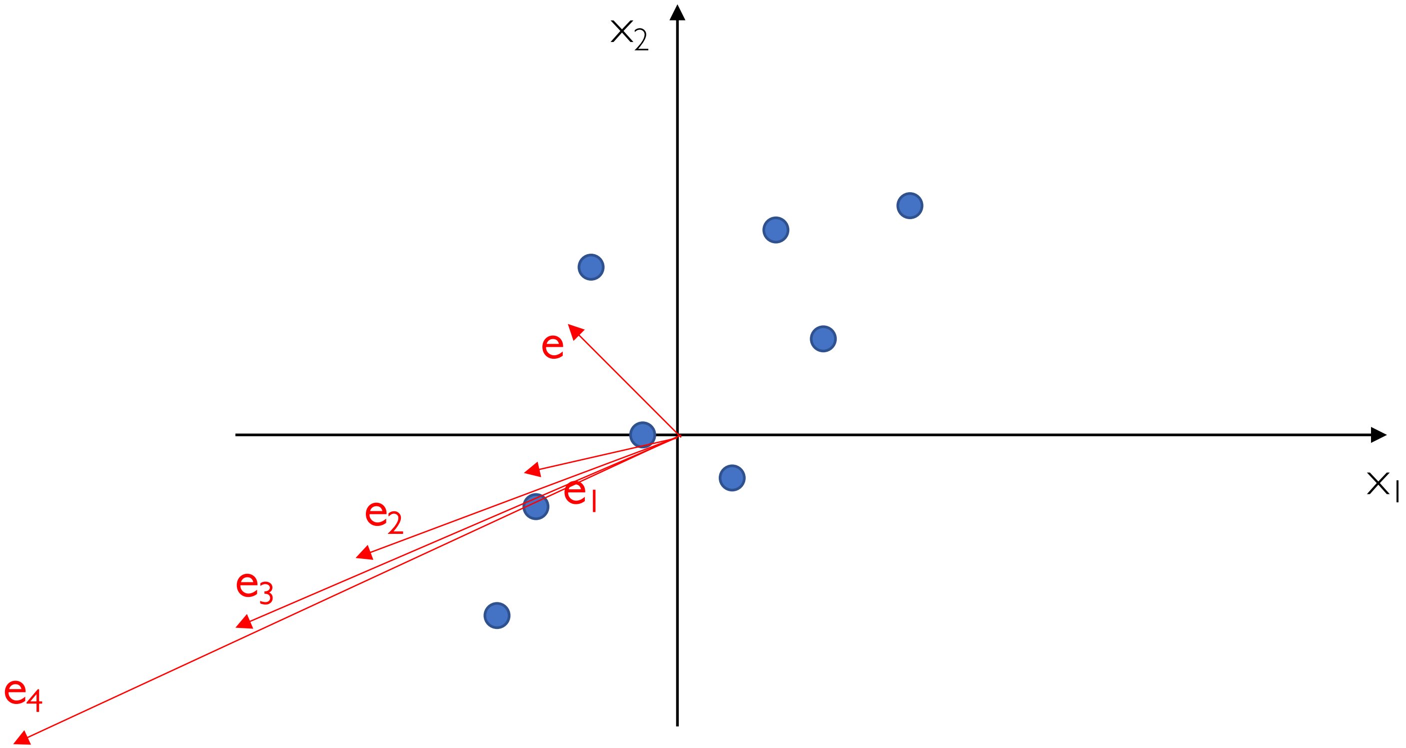
slope = 1/6

new slope = 27/64

Turns towards the direction of the greatest variance

Covariance Matrix of Original Dimensions

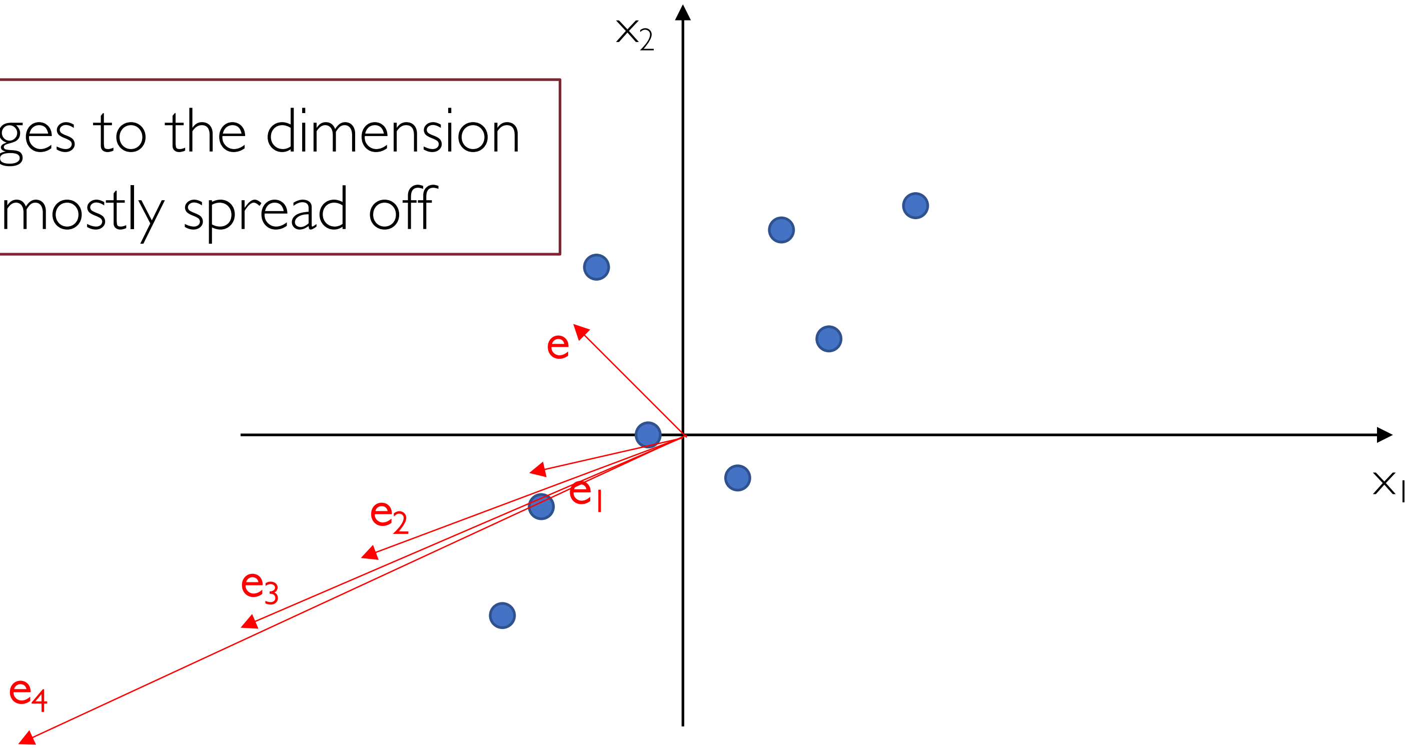
If we keep doing this the resulting vector is getting longer and turns towards the direction of the largest variance



Covariance Matrix of Original Dimensions

If we keep doing this the resulting vector is getting longer and turns towards the direction of the largest variance

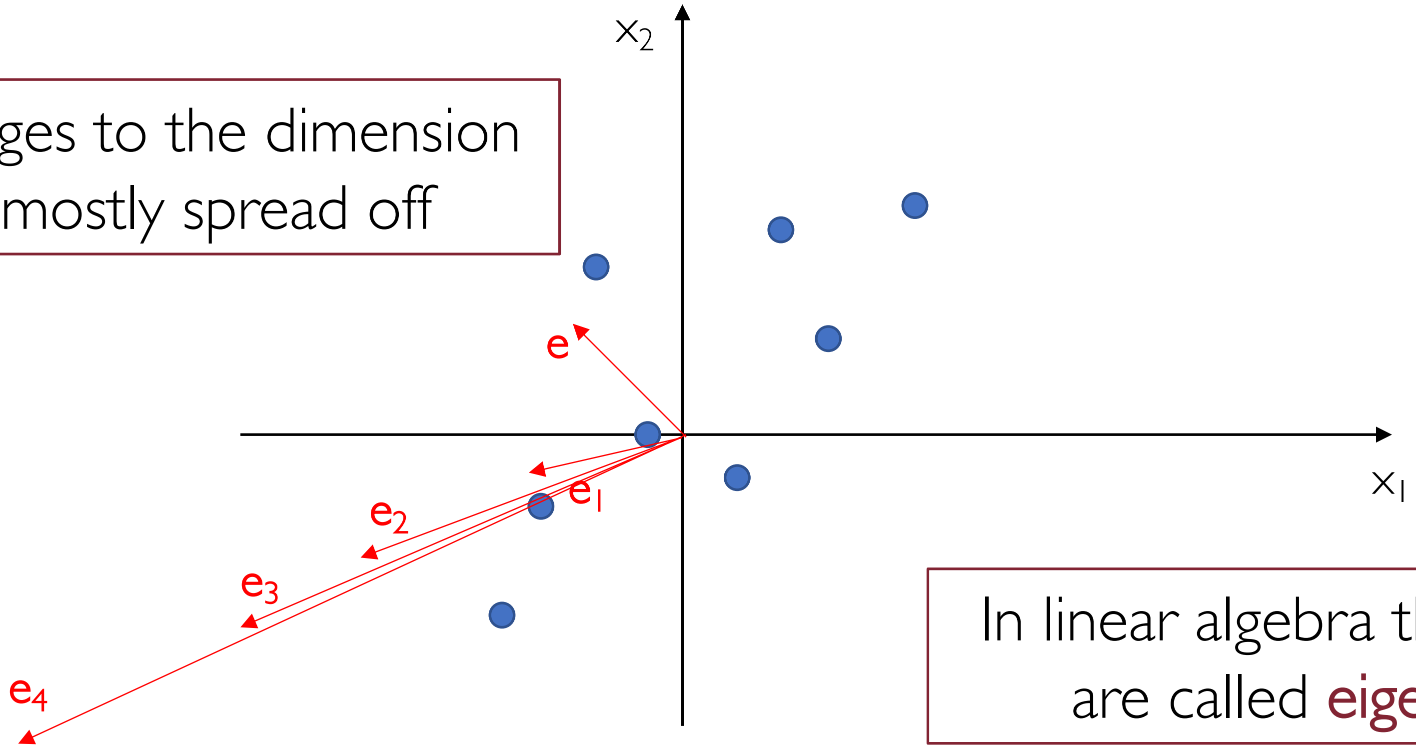
The slope converges to the dimension where data is mostly spread off



Covariance Matrix of Original Dimensions

If we keep doing this the resulting vector is getting longer and turns towards the direction of the largest variance

The slope converges to the dimension where data is mostly spread off



In linear algebra those vectors are called eigenvectors

Take-Home Message of Today

- Raw data are often embedded within high-dimensional spaces
- Dimensionality reduction techniques allow to extract "important" features
- PCA is a dimensionality reduction technique which tries to represent high-dimensional data into a low-dimensional linear subspace
- The intuition behind PCA is to find a change of basis so that the first component maximizes the preserved variance of the data
- Suggested video: https://www.youtube.com/watch?v=PFDu9oVAE-g