Principle Component Analysis (PCA)

Dr Jamie A Ward

This week

Vector spaces

- Linear dependence
- Mean and variance
- Covariance

Principle Component Analysis (PCA)

- Maximise the covariance
- Linearly independent features

Linearly dependent vectors

How can you tell if two vectors are linearly dependent or independent?

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

If one can be written as a linear combination of the other, they are linearly dependent. Otherwise, they are linearly independent.

x and y are linearly dependent ***iff*** there exists scalars a, b such that:

$$a \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = b \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \qquad \text{or equivalently,} \qquad \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Otherwise, they are linearly independent.

Orthogonal vectors

How can you tell if two vectors are orthogonal (at right angles) to each other?

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

If the dot product of two (non-zero) vectors is 0, the are orthogonal:

$$\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{x}^T \boldsymbol{y} = [x_1 \; x_2 \; x_3] \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = x_1 y_1 + x_2 y_2 + x_3 y_3 = 0$$

e.g.

$$\boldsymbol{y} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \boldsymbol{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \;\Rightarrow\; \boldsymbol{x}.\boldsymbol{y} = 0$$

Orthogonal vectors

How can you tell if two vectors are orthogonal (at right angles) to each other?

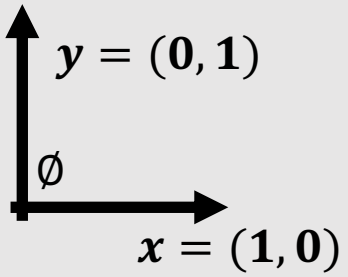
$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

If the dot product of two (non-zero) vectors is 0, the are orthogonal:

$$\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{x}^T \boldsymbol{y} = [x_1\ x_2\ x_3] \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = x_1 y_1 + x_2 y_2 + x_3 y_3 = 0$$

This is because the dot product is proportional to the cosine of the angle $\emptyset$ between $\boldsymbol{x}$ and $\boldsymbol{y}$

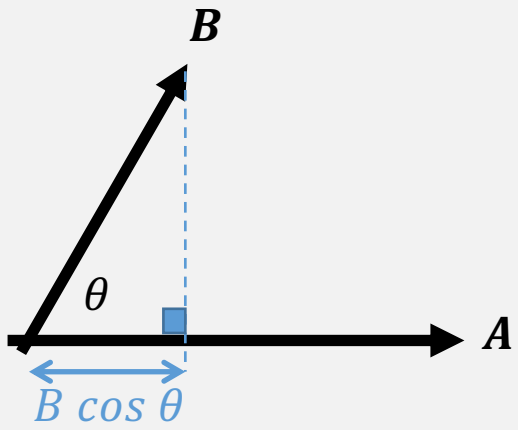
$$\boldsymbol{y} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \boldsymbol{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Rightarrow \boldsymbol{x}.\boldsymbol{y} \propto \cos(\emptyset) = \cos(90°) = 0$$



Aside: Geometry of the dot product

The dot product of two vectors is the magnitude of one times the projection of the second onto the first.

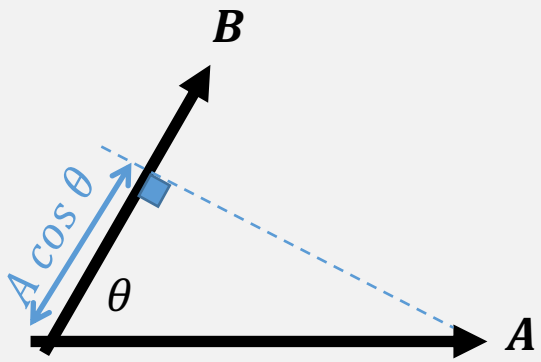
$$\boldsymbol{A} \cdot \boldsymbol{B} = AB\cos\theta$$



Aside: Geometry of the dot product

The dot product of two vectors is the magnitude of one times the projection of the second onto the first.

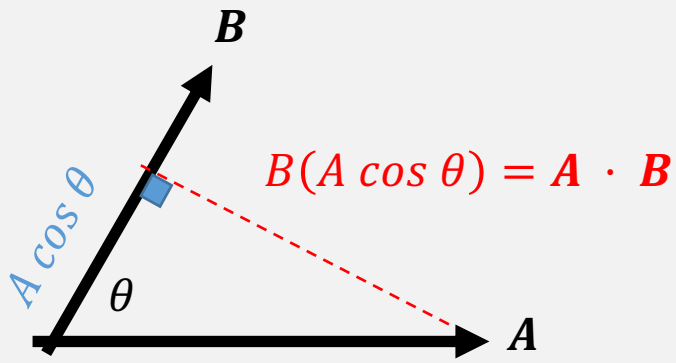
$$\boldsymbol{A} \cdot \boldsymbol{B} = AB\cos\theta$$



Aside: Geometry of the dot product

The dot product of two vectors is the magnitude of one times the projection of the second onto the first.

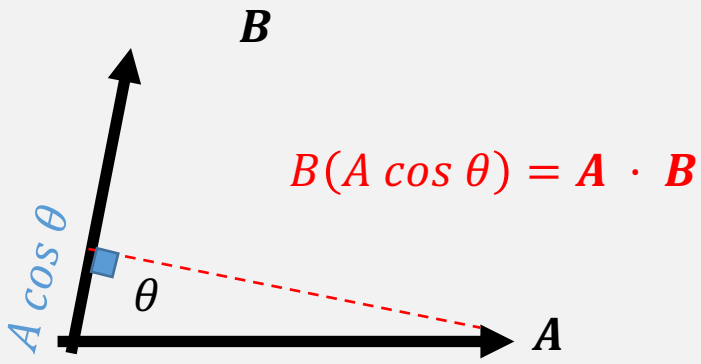
$$\boldsymbol{A} \cdot \boldsymbol{B} = AB\cos\theta$$



Aside: Geometry of the dot product

The dot product of two vectors is the magnitude of one times the projection of the second onto the first.

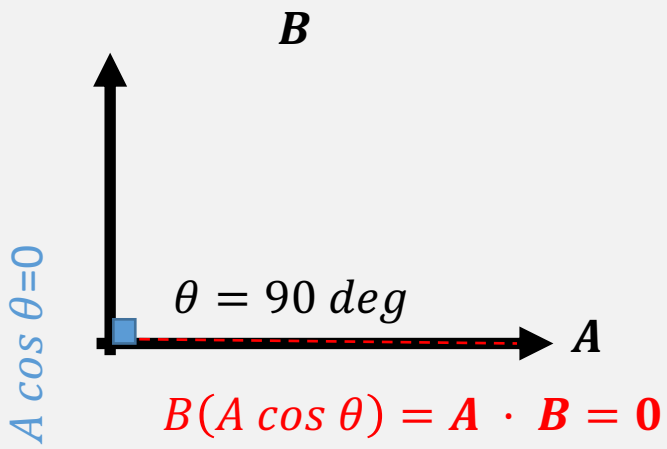
$$\boldsymbol{A} \cdot \boldsymbol{B} = AB\cos\theta$$



Aside: Geometry of the dot product

The dot product of two vectors is the magnitude of one times the projection of the second onto the first.

$$\boldsymbol{A} \cdot \boldsymbol{B} = AB\cos\theta$$



Linear dependence and orthogonality

Given vectors $\boldsymbol{x}$ and $\boldsymbol{y}$

Two tests

1. If dot product $\boldsymbol{x} \cdot \boldsymbol{y} = 0$, then the vectors are orthogonal and thus linearly independent.
2. They are linearly dependent if there exists non-zero scalars, a and b, that satisfy:

$$a\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = b\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \quad \text{or equivalently,} \quad \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \end{bmatrix}\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Otherwise the vectors are linearly independent.

Linear dependence question

- Is x linearly dependent on y?

$$\boldsymbol{x} = \begin{bmatrix} 2 \\ 4 \\ 0 \\ 1 \end{bmatrix} \quad \boldsymbol{y} = \begin{bmatrix} 4 \\ 8 \\ 0 \\ 2 \end{bmatrix}$$

Yes, since $ay = bx, \quad a = 2, b = 1$

$$\boldsymbol{x} = \begin{bmatrix} 0 \\ 4 \\ 0 \end{bmatrix} \quad \boldsymbol{y} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$$

No, since $x^T y = 0$, *i.e.*

$$x^T y = [0\ 4\ 0] \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} = 0 * 2 + 4 * 0 + 0 * 1 = 0$$

Or, equivalently, $\nexists \{a, b\}$ s.t. $ay = bx$

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \boldsymbol{y} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

No, since $\nexists \{a, b\}$ s.t. $ay = bx$

$\nexists$ means does not exist

Sample mean and variance

Given a random variable X that takes on a set of N values $x^{(i)}$, then the expected value of X is defined as:

$$E[x] = \sum_{i=1}^{N} x^{(i)} p_i$$

(where p_i is the *probability* of x_i appearing)

If we assume $p_i = \frac{1}{N}$, **then:**

$$E[x] = mean(x) = \bar{\boldsymbol{x}} = \frac{1}{N} \sum_{i=1}^{N} x^{(i)}$$

The variance of X can be written:

$$var(x) = \sigma^2 = \frac{1}{N} \sum_{i=1}^{N} \left(x^{(i)} - \bar{\boldsymbol{x}}\right)^2$$

Sample mean and variance: example

You are given a dataset with the scalar values:

$$X = [0,1,2,3,4]$$

Estimate the mean as:

$$\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} x^{(i)} = \frac{0+1+2+3+4}{5} = 2$$

Using this, the variance of X can be estimated as:

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \bar{\boldsymbol{x}}\right)^2 = \frac{(-2)^2 + (-1)^2 + 0 + 1 + 2^2}{5} = \frac{10}{5} = 2$$

How does $\bar{x}$ and σ^2 change if we grow the dataset as follows (does it increase, decrease, or stay the same)?

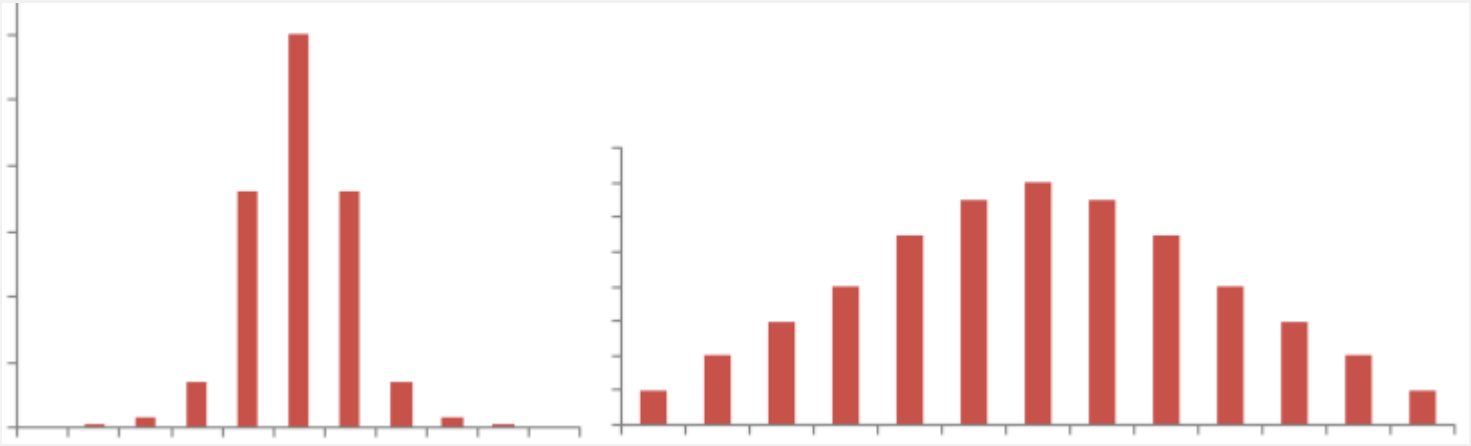
1. $X = [0,1,2,3,4,2,2,2,2]$ $\quad\quad \bar{x} = 2 \;\; \sigma^2 = 1.11$
2. $X = [0,1,2,3,4,-1,5]$ $\quad\quad \bar{x} = 2 \;\; \sigma^2 = 4$

Variance: how far data is from the mean

$$X = [0,1,2,3,4] \qquad \bar{x} = 2 \;\; \sigma^2 = 2$$

$$X = [0,1,2,3,4,2,2,2,2] \qquad \bar{x} = 2 \;\; \sigma^2 = 1.11$$

$$X = [0,1,2,3,4,-1,5] \qquad \bar{x} = 2 \;\; \sigma^2 = 4$$



smaller σ

larger σ

Mean and Covariance over >1 features

You are given a data matrix $\boldsymbol{X}$ with F features and N samples:

$$\boldsymbol{X} = \left[\boldsymbol{x}^{(1)}\boldsymbol{x}^{(2)}, \ldots \boldsymbol{x}^{(N)}\right], \qquad where\ \boldsymbol{X} \in \mathbb{R}^{F\times N}$$

The sample mean is then:

$$\overline{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}^{(i)}, \qquad where\ \boldsymbol{x}^{(i)} \in \mathbb{R}^{F\times 1}$$

The covariance is estimated as:

$$cov(\boldsymbol{X}) = \boldsymbol{S} = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \overline{\boldsymbol{x}}\right)\left(x^{(i)} - \overline{\boldsymbol{x}}\right)^{T}$$

This captures how the N variables vary together – how they change together with respect to the mean.

Mean, variance, and covariance

1 Dimension

$$\boldsymbol{X} = [0,1,2,3,4]$$ Data

$$\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} x^{(i)} = 2$$ Mean (scalar)

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \bar{\boldsymbol{x}}\right)^2 = 2$$ Variance (scalar)

> 1 dimension (x is an F-dimensional vector)

$$\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}^{(i)}$$ Mean (Fx1)

$$cov(\boldsymbol{X}) = \boldsymbol{S} = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \bar{\boldsymbol{x}}\right)\left(x^{(i)} - \bar{\boldsymbol{x}}\right)^T$$ Covariance (FxF)

Covariance

- How two feature vectors change together is based on the product of two scalars, e.g. s1 and s2:

S1	S2	S1*S2
−	−	+
+	+	+
−	+	−
+	−	−

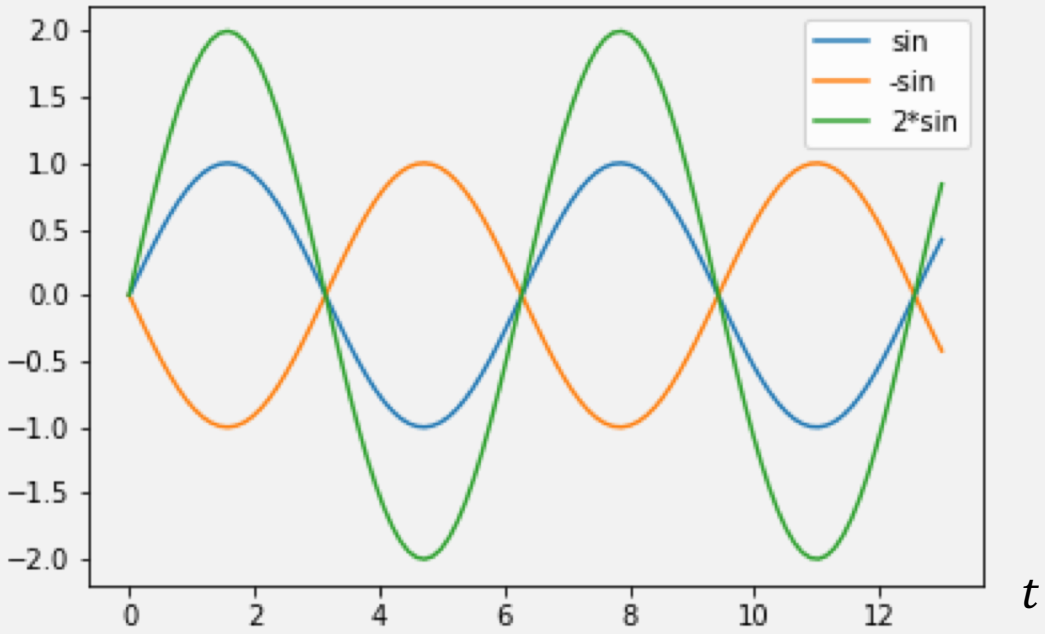
+ve sign: Covariance increase

-ve sign: Covariance decrease

Covariance example

feature1 $x_1 = \sin(t)$
feature2 $x_2 = -\sin(t)$
feature3 $x_3 = 2 * \sin(t)$

Given data with N=100 samples,
F=3 features, and zero mean:



1. What are the dimensions of feature matrix X?
2. What are the dimensions of covariance matrix S?
3. Is the covariance of feature1 to feature3 positive or negative?
4. What about the covariance of feature1 to feature2?

Covariance example

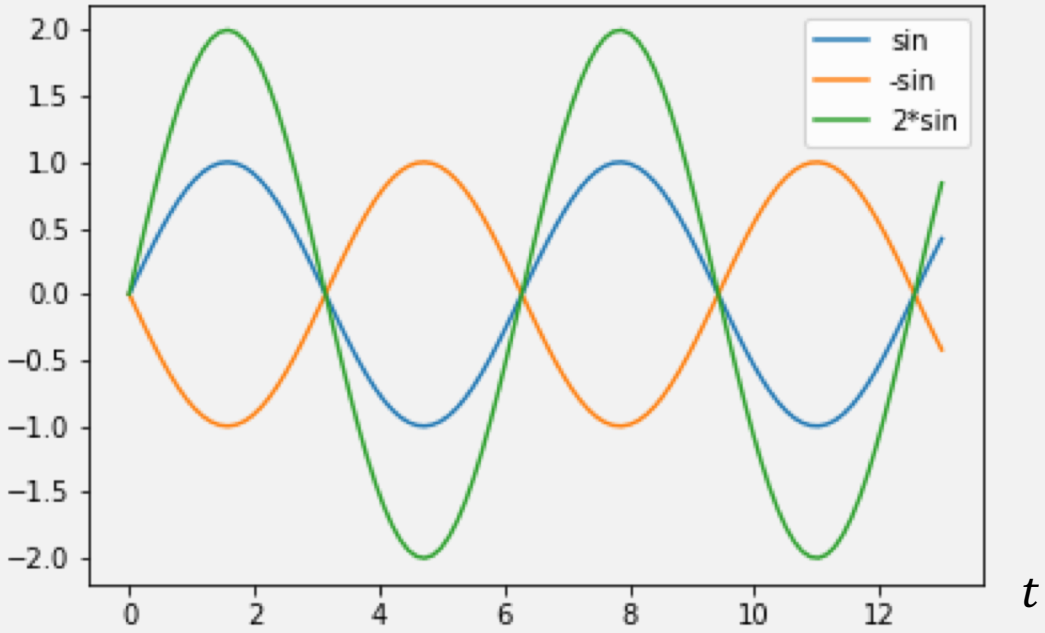
feature1 $x_1 = \sin(t)$
feature2 $x_2 = -\sin(t)$
feature3 $x_3 = 2 * \sin(t)$

N=100 samples, F=3 features
$X \in \mathbb{R}^{F \times N}, F = 3, N = 100$

Dimensions of S are 3×3
(Features x Features)

$$S = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \bar{\boldsymbol{x}}\right)\left(x^{(i)} - \bar{\boldsymbol{x}}\right)^T$$

$$S = \begin{bmatrix} 0.5 & -0.5 & 1 \\ -0.5 & 0.5 & -1 \\ 1 & -1 & 2 \end{bmatrix}$$



1. What are the dimensions of feature matrix X?
2. What are the dimensions of covariance matrix S?
3. Is the covariance of feature1 to feature3 positive or negative?
4. What about the covariance of feature1 to feature2?

Covariance example

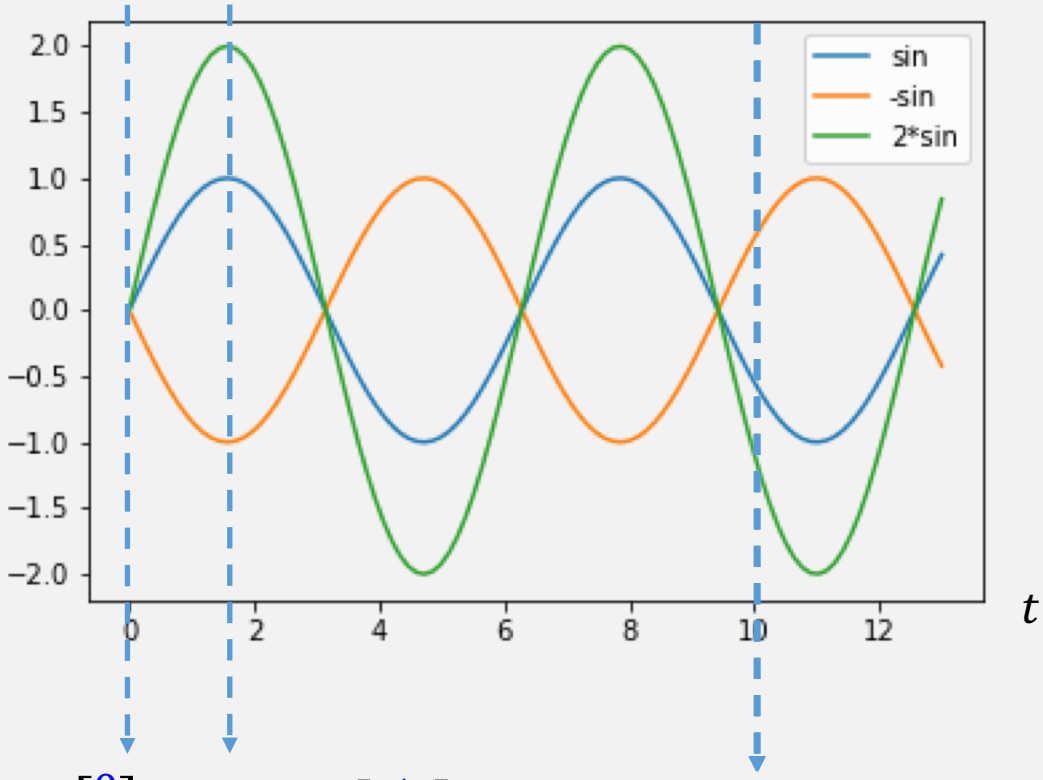
feature1 $x_1 = \sin(t)$
feature2 $x_2 = -\sin(t)$
feature3 $x_3 = 2 * \sin(t)$

N=100 samples, F=3 features
$X \in \mathbb{R}^{F \times N}, F = 3, N = 100$

Dimensions of S are 3×3
(Features x Features)

$$S = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \bar{\boldsymbol{x}}\right)\left(x^{(i)} - \bar{\boldsymbol{x}}\right)^T$$

$$S = \begin{bmatrix} 0.5 & -0.5 & 1 \\ -0.5 & 0.5 & -1 \\ 1 & -1 & 2 \end{bmatrix}$$

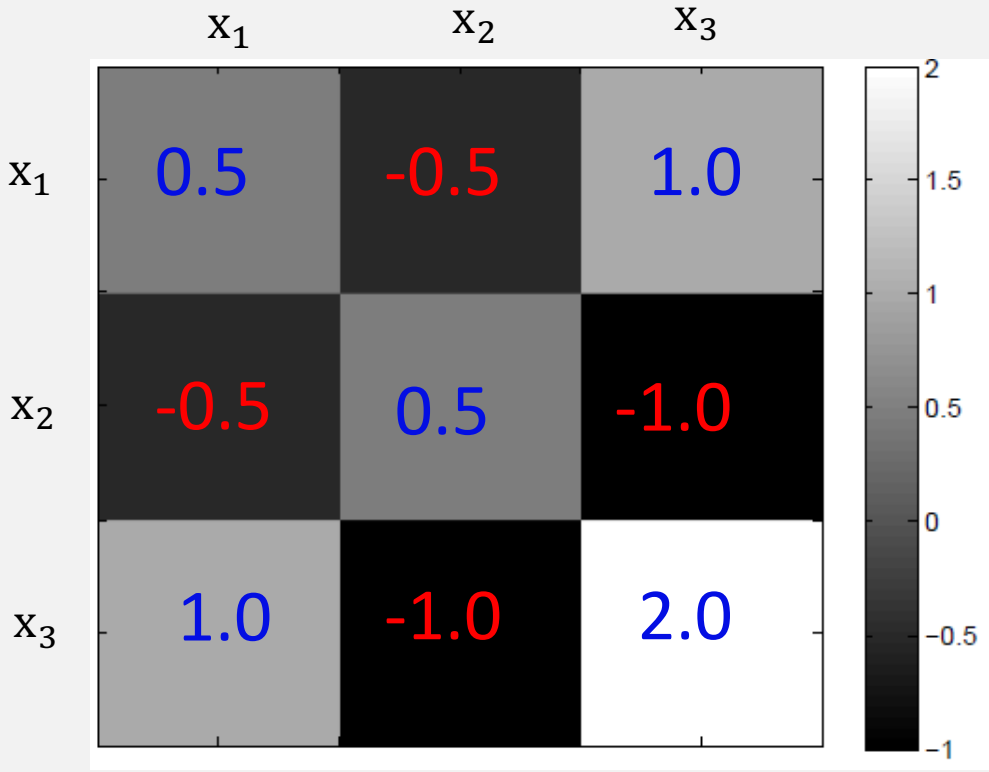


$$\boldsymbol{x}^{(0)} = \begin{bmatrix} x_1^{(0)} & x_2^{(0)} & x_3^{(0)} \end{bmatrix}^T = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \boldsymbol{x}^{(12)} = \begin{bmatrix} 1 \\ -1 \\ 2 \end{bmatrix} \quad \boldsymbol{x}^{(76)} = \begin{bmatrix} -0.53 \\ 0.52 \\ -1.05 \end{bmatrix}$$

Covariance example

$$S = \frac{1}{N}\sum_{i=1}^{N}\left(x^{(i)} - \bar{\boldsymbol{x}}\right)\left(x^{(i)} - \bar{\boldsymbol{x}}\right)^{T}$$

$$S = \begin{bmatrix} 0.5 & -0.5 & 1 \\ -0.5 & 0.5 & -1 \\ 1 & -1 & 2 \end{bmatrix}$$



Cov(x_1, x_3) = Cov(x_3, x_1) = 1 → +ve: x_1 and x_3 move together

Cov(x_1, x_2) = Cov(x_2, x_1) = -0.5 → -ve: x_1 and x_2 move apart

This week

Vector spaces

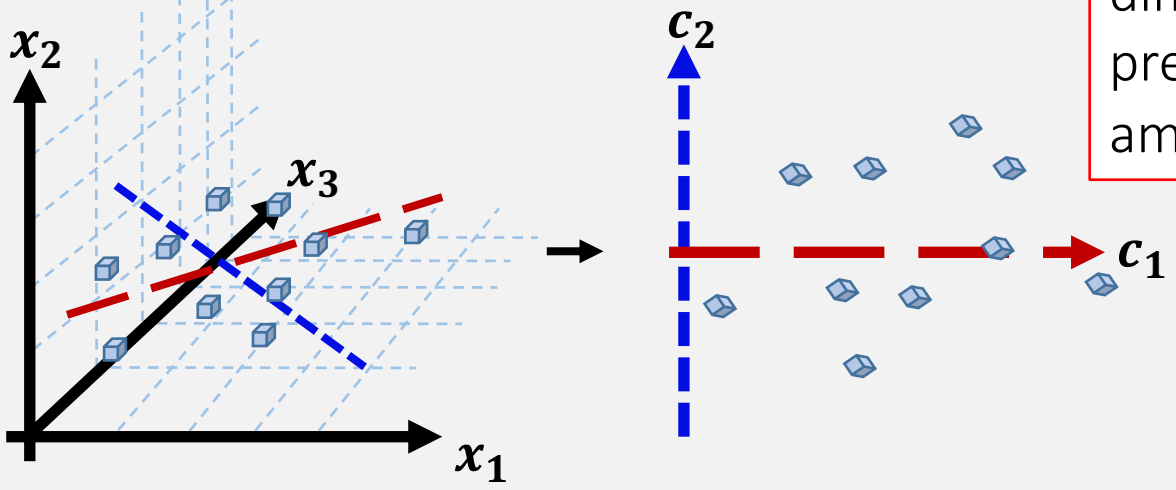
- Linear dependence
- Mean and variance
- Covariance

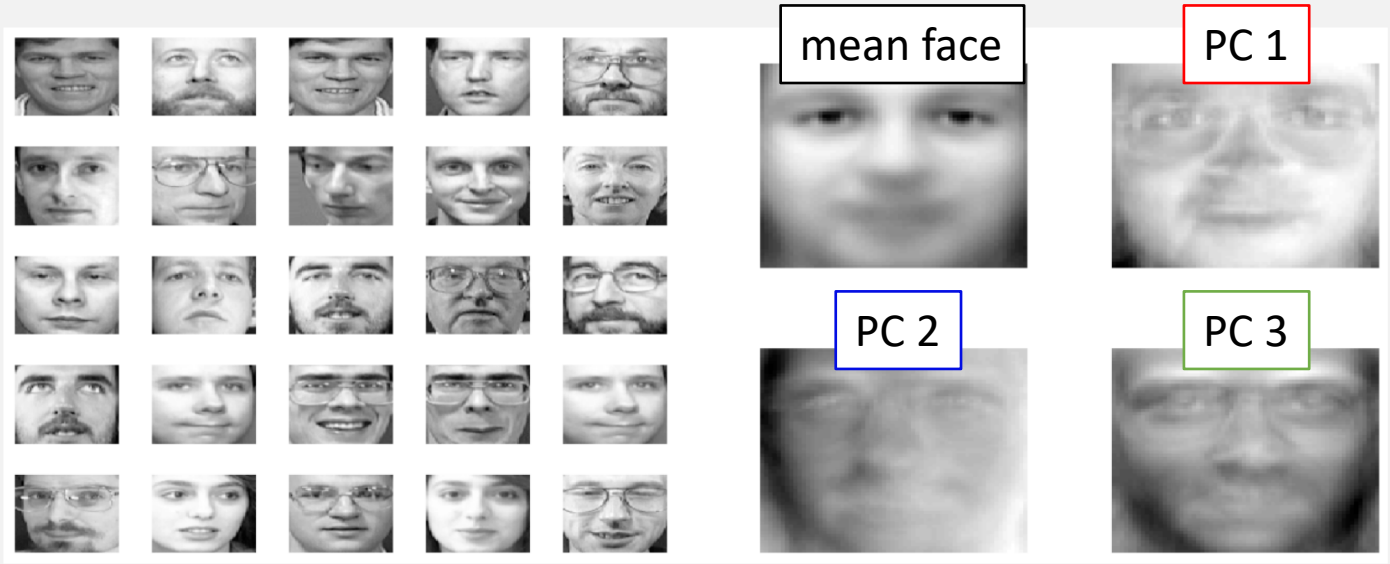
Principle Component Analysis (PCA)

- Maximise the covariance
- Linearly independent features

Linear dimensionality reduction

PCA Goal: reduce dimensionality, while preserving the maximum amount of variability

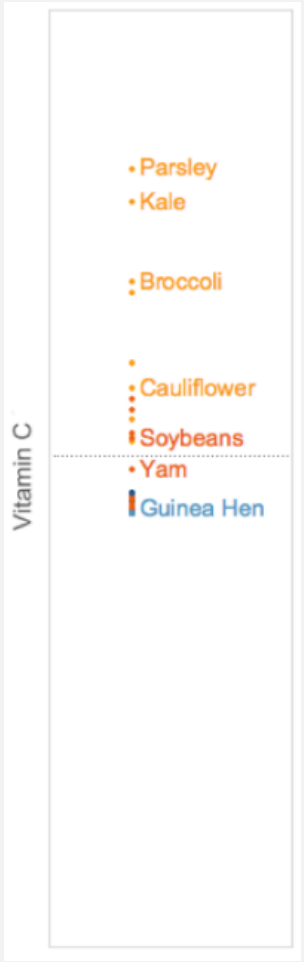




Maximising the variance

We have a dataset of food with the following features (F=4)

- Vitamin C
- Fat
- Protein
- Fiber



Maximising the variance

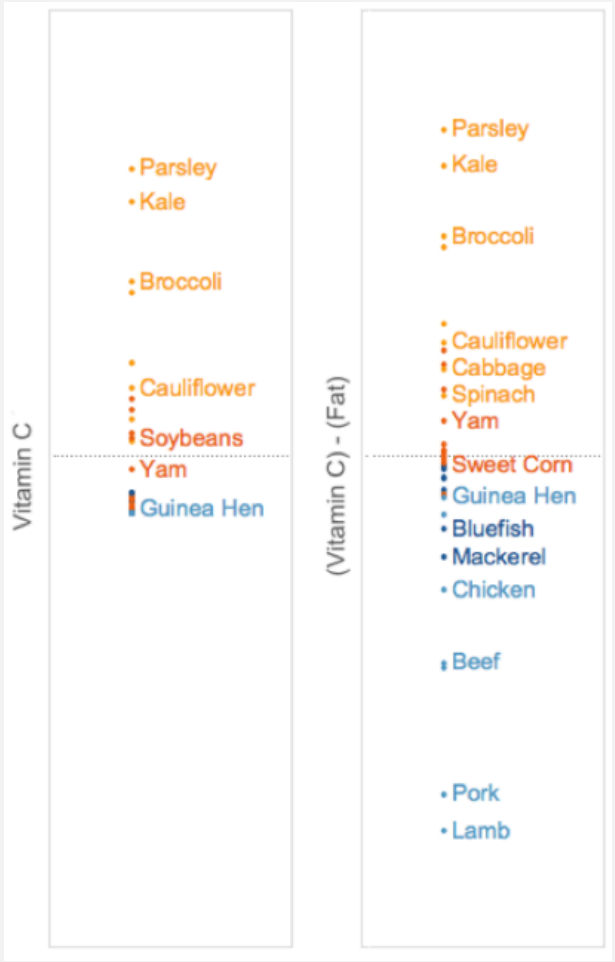
We have a dataset of food with the following features (F=4)

- Vitamin C
- Fat
- Protein
- Fiber

Make a new feature: VitaminC - Fat

(taking care to normalise the values to make the comparable)

- The data spreads out more



Maximising the variance

We have a dataset of food with the following features (F=4)

- Vitamin C
- Fat
- Protein
- Fiber

Make a new feature: VitaminC - Fat

(taking care to normalise the values to make the comparable)

- The data spreads out more

Now sum VitaminC – Fat + Fiber

- Even better spread

These are linear combinations of the original feature space

Principle components and food

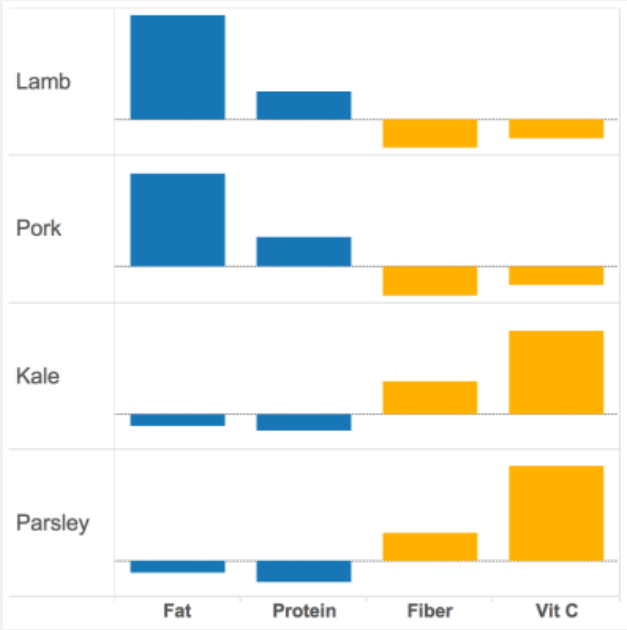
We can see correlations in co-occurrence of some features

- fat and protein 'move together'
- protein and fiber 'move apart'

Idea:

- instead of using all 4 features, combine those that are correlated
- This is the basis of PCA
- Create new features based on linear combination of the old ones (principle components)
- Use weights to indicate the importance of each feature's contribution:

Covariance in features



	PC1	PC2	PC3	PC4
Fat	-0.45	0.66	0.58	0.18
Protein	-0.55	0.21	-0.46	-0.67
Fiber	0.55	0.19	0.43	-0.69
Vitamin C	0.44	0.70	-0.52	0.22

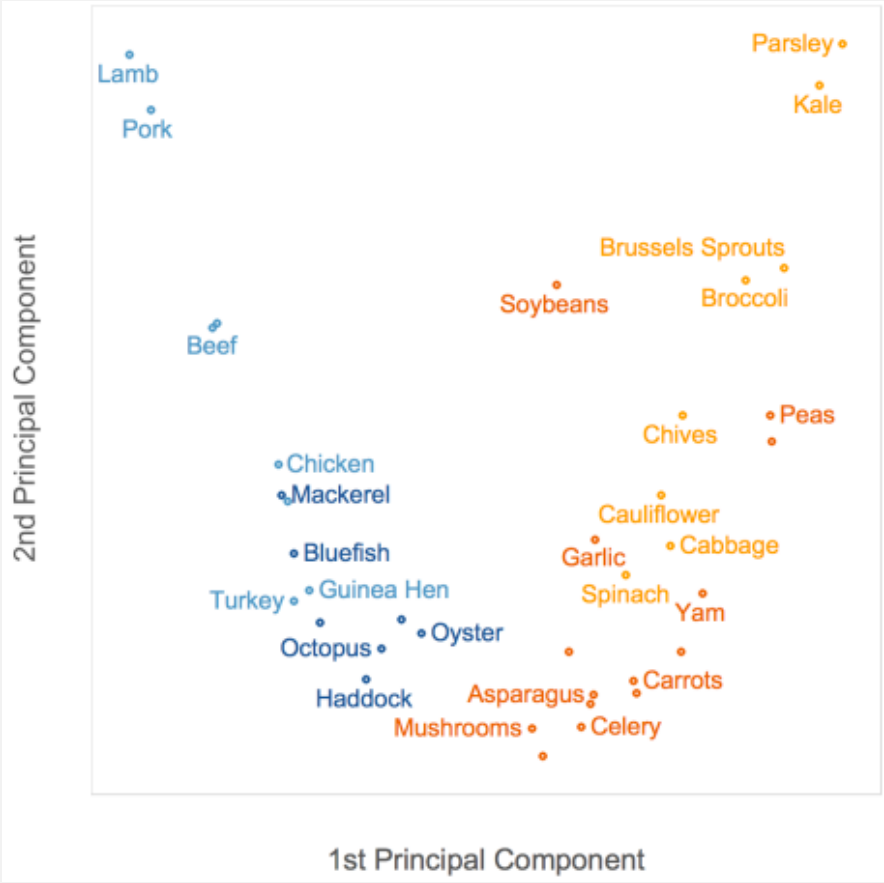
PCA and food

Using our top 2 principle components, we can visualise the data clearly

Calculation:

PC1 = 0.55*fiber + 0.44*VitC
- 0.55*protein – 0.45*fat

PC2 = 0.66*fat + 0.7*VitC
+ 0.21*protein +0.19*fiber



	PC1	PC2	PC3	PC4
Fat	-0.45	0.66	0.58	0.18
Protein	-0.55	0.21	-0.46	-0.67
Fiber	0.55	0.19	0.43	-0.69
Vitamin C	0.44	0.70	-0.52	0.22

Principle Component Analysis (PCA)

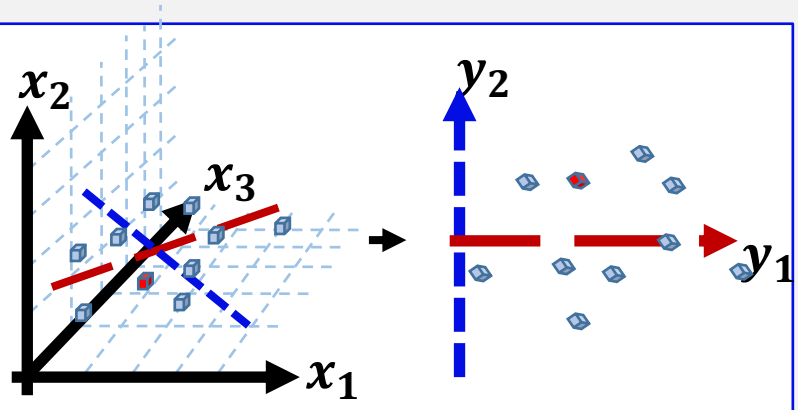
PCA maximises the covariance in a linearly projected space

If X represents our data

$$Y = W^T X$$ A linear projection

Linear projection and dimensionality reduction

e.g. matrix W maps (transforms) data X onto Y



2D

If (3 to 2 mapping) $W = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 1 & 0 \end{bmatrix}$ and (3x1 feature) data $X = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$, then

From 3D to 2D

$$\mathrm{Y} = W^T X = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} (1*1)+(0*2)+(1*3) \\ (0*1)+(2*2)+(0*3) \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$$

Linear projection examples

Using single 3x1 data data point

$$X = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

From 3 to 2 dimensions: $W = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 1 & 0 \end{bmatrix}$

$$\mathrm{Y} = W^T X = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} (1*1)+(0*2)+(1*3) \\ (0*1)+(2*2)+(0*3) \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$$

From 3 to 1 dimension: $W = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$

$$\mathrm{Y} = W^T X = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = [(1*1)+(0*2)+(1*3)] = 4$$

From 3 to 3 dimensions: $W = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 0 & 2 \end{bmatrix}$

$$\mathrm{Y} = W^T X = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 2 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} (1*1)+(0*2)+(1*3) \\ (0*1)+(2*2)+(0*3) \\ (2*1)+(0*2)+(2*3) \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 8 \end{bmatrix}$$

Principle Component Analysis (PCA)

PCA maximises the covariance in a linearly projected space

If X represents our data

$$Y = W^T X \quad \text{A linear projection}$$
$$cov(Y) = cov(W^T X) \quad \text{What we want to maximise}$$

We can write $cov(Y)$ as a function of $cov(X)$ directly

$$cov(Y) = cov(W^T X) = W^T cov(X) W$$

With $S = cov(X)$, we need to find the best W that maximises:

Square matrix (DxD)

$$W^T S W$$

aside: argmax notation

The argmax notation provides a way to ask

"What is the best value of V that maximises the value of function $f(V)$?"

$$\underset{V}{\text{argmax}}\left(f(V)\right)$$

Principle Component Analysis (PCA)

PCA maximises the covariance in a linearly projected space

If X represents our data

$$Y = W^T X \quad \text{A linear projection}$$
$$cov(Y) = cov(W^T X) \quad \text{What we want to maximise}$$

We can write $cov(Y)$ as a function of $cov(X)$ directly

$$cov(Y) = cov(W^T X) = W^T cov(X) W$$

With $S = cov(X)$, we need to find the best W that maximises:

Square matrix (DxD)

$$W^T S W$$

That is, find: $\underset{W}{\text{argmax}}(W^T S W)$

Principle Component Analysis (PCA)

PCA maximises the covariance of our data in a linearly projected space

- Find the best values for W that maximise $W^T S W$:

$$\underset{W}{\operatorname{argmax}} (W^T S W)$$

- What will happen if you maximise this as it is (think about the 1D case)?

By setting values in W to infinity (∞) we get infinitely big covariance

- But we wouldn't know what to do with this, so…
- Add a constraint to ensure W is finite:

$$W^T W = \boldsymbol{I}$$

- This constraint also ensures no linear dependencies in the projected data.
- Why?...

No linear dependencies?

If the dot product of two (non-zero) vectors is zero, they are linearly independent (orthogonal)

$$\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{x}^T \boldsymbol{y} = [x_1\ x_2\ x_3] \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = x_1 y_1 + x_2 y_2 + x_3 y_3 = 0$$

Consider a W with 2 components, where $W^T W = \mathrm{I}$:

$$I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad W = [\boldsymbol{w}_1 \quad \boldsymbol{w}_2], \qquad where\ \boldsymbol{w}_i = \begin{bmatrix} w_{i1} \\ w_{i2} \\ \vdots \\ w\_iD \end{bmatrix}$$

What does the $W^T W$ product look like?

$$W^T W = \begin{bmatrix} \boldsymbol{w}_1^2 & \boldsymbol{w}_2^T \boldsymbol{w}_1 \\ \boldsymbol{w}_1^T \boldsymbol{w}_2 & \boldsymbol{w}_2^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Every vector w_i of W will be linearly independent of all other vectors w_j

- $\boldsymbol{w}_i^T \boldsymbol{w}_j = 0$ (for all $i \neq j$)

Principle Component Analysis (PCA)

PCA maximises the covariance of our data in a linearly projected space

- Find the best values for W that maximise $W^T S W$:

$$\underset{W}{\operatorname{argmax}} (W^T S W) \quad \text{s.t.} \quad W^T W = \boldsymbol{I}$$

Constraint $W^T W = \boldsymbol{I}$ ensures:

- We don't get infinite values for W
- That the projected features are linearly independent

We can use a technique called eigenanalysis to get the optimal values of W.

eigenvectors eigenvalues original data

$$[W, \Lambda] = eig(S), \qquad \text{where S} = \text{cov(X)}$$

The data can finally be transformed using: $Y = W^T X$

Eigenanalysis (briefly)

$$[\boldsymbol{W}, \Lambda] = eig(\boldsymbol{S}), \qquad \text{where } \mathbf{S} = \text{cov}(\text{X})$$

This function finds a matrix of vectors ($\boldsymbol{W}$) and corresponding scalars (Λ), such that:

$$\boldsymbol{SW} = \Lambda \boldsymbol{W}$$

→multiplying the vectors in $\boldsymbol{W}$ by covariance matrix $\boldsymbol{S}$, is the same as multiplying them by the scalar values Λ

→$\boldsymbol{W}$ are called the eigenvectors

→$\boldsymbol{\Lambda}$ are called eigenvalues

Principle Component Analysis (PCA)

PCA maximises the covariance of our data in a linearly projected space

- Find the best values for W that maximise $W^T S W$:

$$\underset{W}{\operatorname{argmax}} (W^T S W) \quad \text{s.t.} \quad W^T W = \boldsymbol{I}$$

Constraint $W^T W = \boldsymbol{I}$ ensures:

- We don't get infinite values for W
- That the projected features are linearly independent

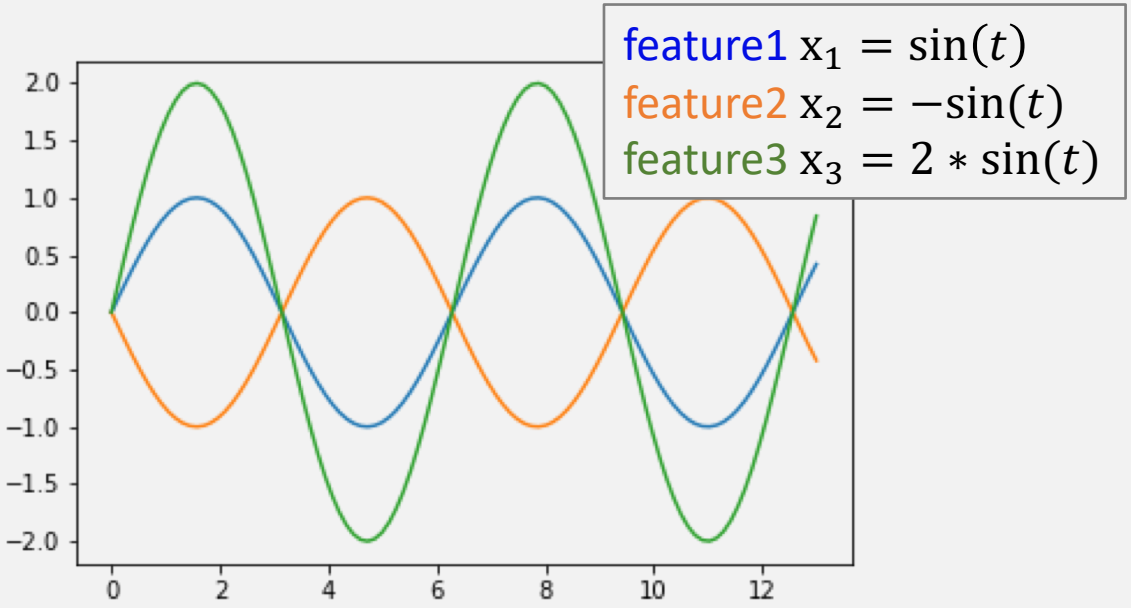
We can use a technique called **eigenanalysis** to get the optimal values of W.

eigenvectors eigenvalues original data

$$[W, \Lambda] = eig(S), \qquad \text{where S} = \text{cov(X)}$$

The data can finally be transformed using: $Y = W^T X$

PCA example



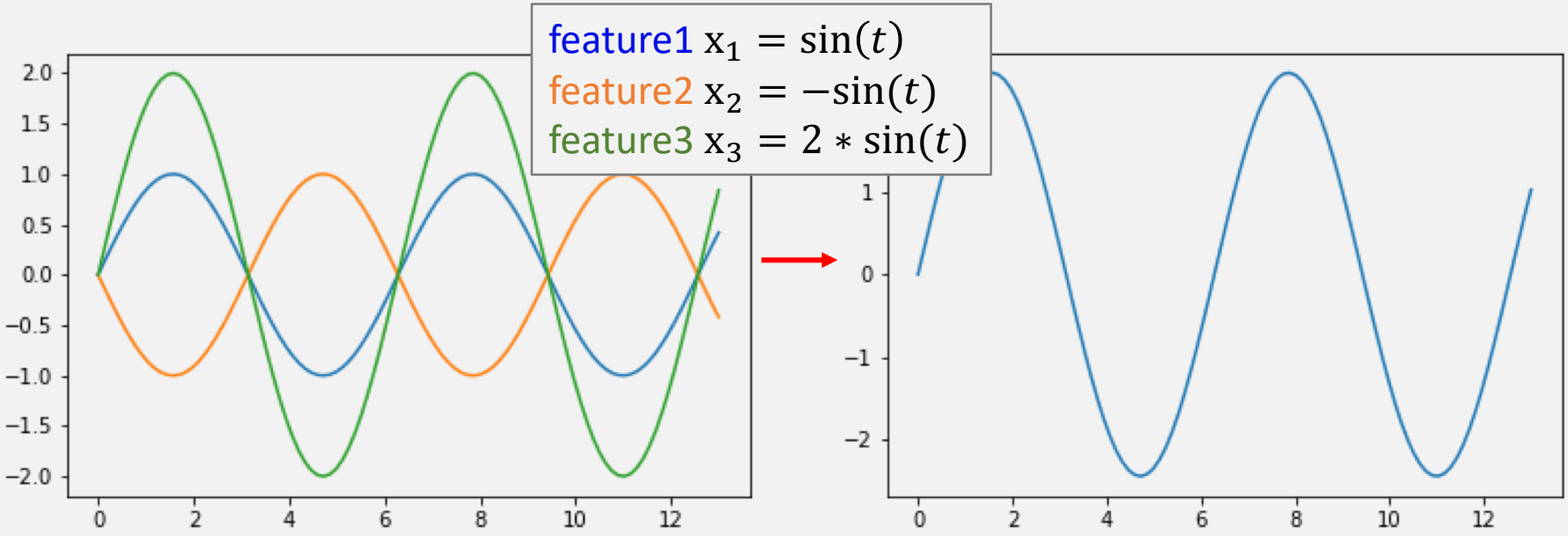
Recall that PCA removes any linear dependencies in the data

Questions:

1. How many dimensions do we need to describe the data above?

2. How many dimensions would PCA need to recover?

PCA example



Recall that PCA removes any linear dependencies in the data

Questions:

1. How many dimensions do we need to describe the data above?
   Feature2 is linearly dependent on feature1: $x_2 = -1 * x_1$
   Feature3 is linearly dependent on feature2: $x_3 = -2 * x_2 = 2 * x_1$
2. How many dimensions would PCA need to recover?

Applying PCA with Eigenanalysis

Given data X, say in 3D, we want a 2D representation Y that maximises the covariance (using PCA)

1. Centre data around its mean, i.e. $X = X - \bar{X}$
2. Calculate covariance matrix of the data, $S = cov(X)$
3. Calculate eigenvalues and eigenvectors, $[\mathbf{W}, \Lambda] = eig(S)$
4. Order the eigenvalues Λ from large to small→ this gives us the components of the corresponding eigenvectors, $\mathbf{W}$

   e.g. for a 2D output, we keep 2 eigenvectors w_i from $\boldsymbol{W}$ that correspond to the 2 biggest eigenvalues. We can call this matrix $\boldsymbol{W_2}$
5. Finally, project the original data into 2D by multiplying $\boldsymbol{W_2}$ with X, i.e. $Y = \boldsymbol{W_2^T} X$

```
import numpy as np
X = np.array([[1,2],[.1,.3],[4,3]]) # 2 x 3D data points
S = np.cov(X)
v, U = np.linalg.eig(S)
top = np.flip( np.argsort(v), 0)
U2 = U[:,top[:2]]
Y = U2.T.dot(X)
```

Data compression

Once we have a feature reduction map $\boldsymbol{W}$:

$$Y_2 = \boldsymbol{W_2^T} X$$

e.g. $\boldsymbol{W_2^T}$ $\boldsymbol{X}$ Y_2

$$= \begin{bmatrix} .7 & -.7 & .2 \\ -.14 & .14 & .98 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ .5 \end{bmatrix} \approx \begin{bmatrix} 1.44 \\ 0.23 \end{bmatrix}$$

We can reverse the operation to recreate our original data:

$$\tilde{X} = \boldsymbol{W_2} Y_2$$

$\boldsymbol{W_2}$ $\tilde{X}$

$$= \begin{bmatrix} .7 & -.14 \\ -.7 & .14 \\ .2 & .98 \end{bmatrix} \begin{bmatrix} 1.44 \\ 0.23 \end{bmatrix} \approx \begin{bmatrix} 1 \\ -1 \\ .5 \end{bmatrix}$$

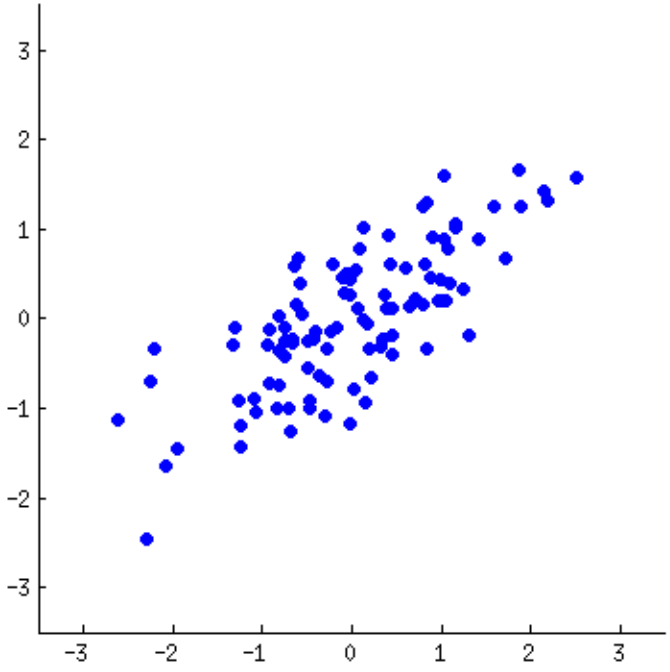
Some data will be lost (unless we retain all components), but PCA ensures the most descriptive data remains.

Same as above, but using a 1D mapping

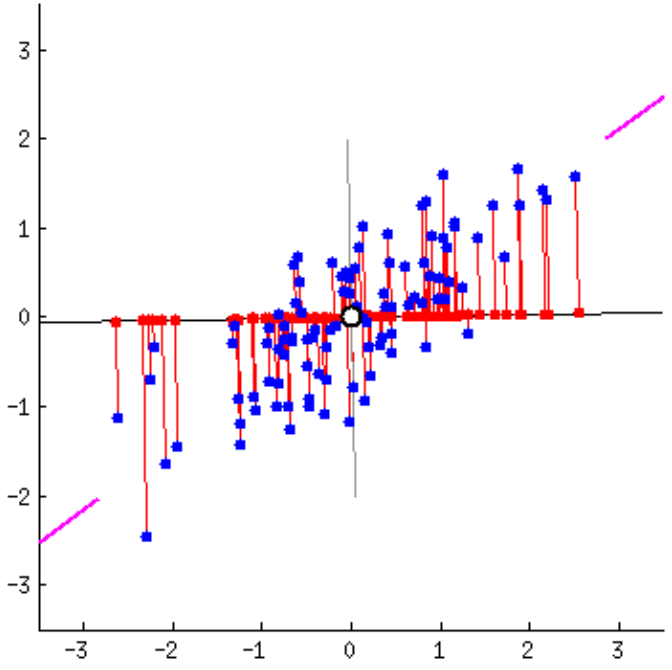
$$\tilde{X} = \boldsymbol{W_1} Y_1$$

$$\begin{bmatrix} .7 \\ -.7 \\ .2 \end{bmatrix} [1.44] \approx \begin{bmatrix} 1 \\ -1 \\ .3 \end{bmatrix}$$

PCA: maximise variance, minimise error



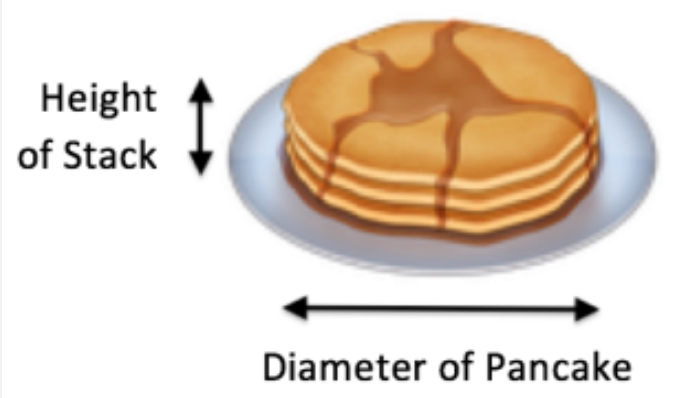
PCA: maximise variance, minimise error



Limitation: the pancake problem

Maximising variance isn't always what we need

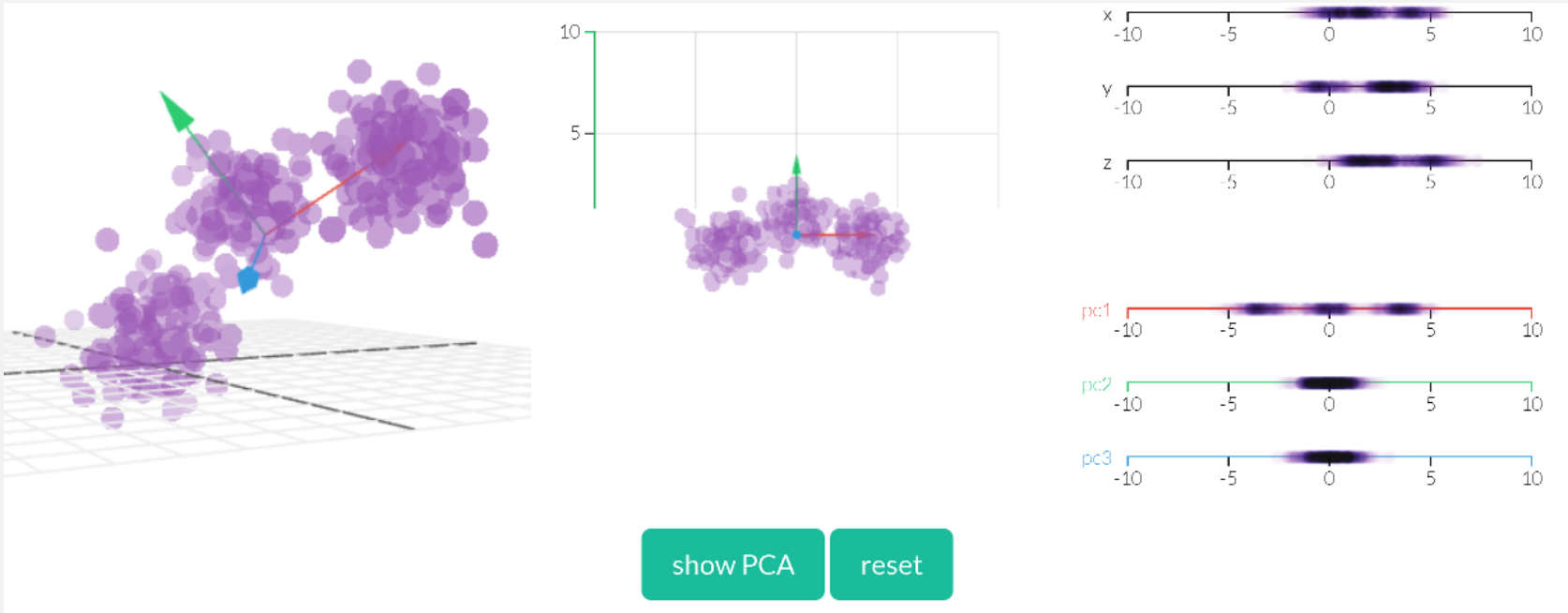
- If we are interested in the number of pancakes, then height is a more useful dimension than diameter
- But PCA will select diameter if it has more variance (i.e. for a small number of pancakes)



Visualisations

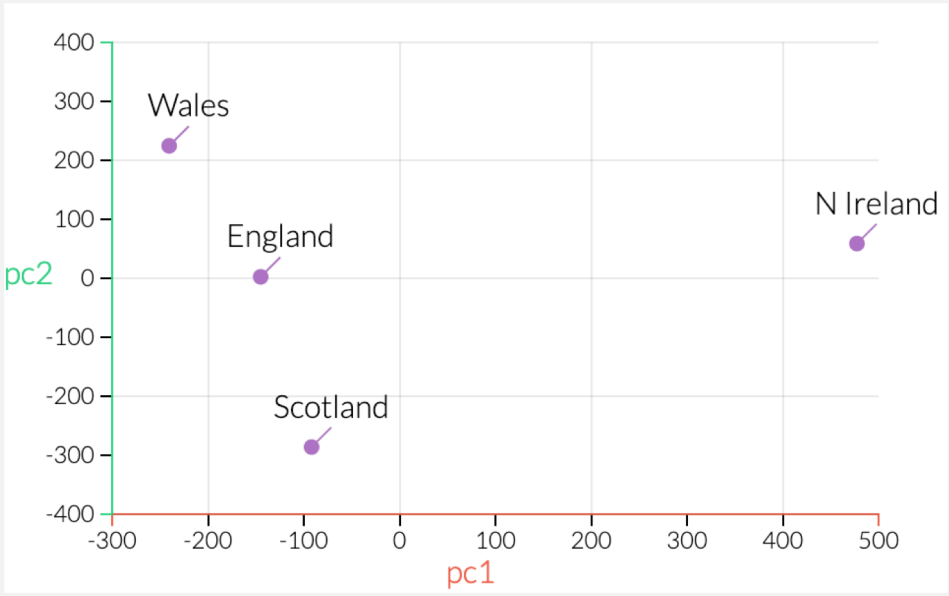
Some nice interactive examples:

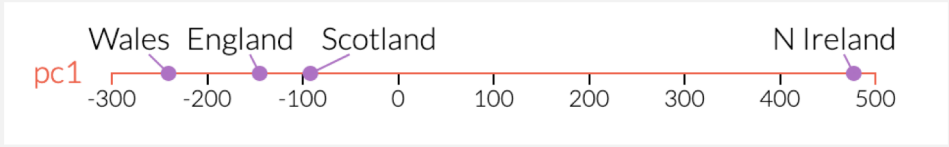
http://setosa.io/ev/principal-component-analysis/



Eating in the UK (a 17D example)

	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175





Interactive examples:

http://setosa.io/ev/principal-component-analysis/