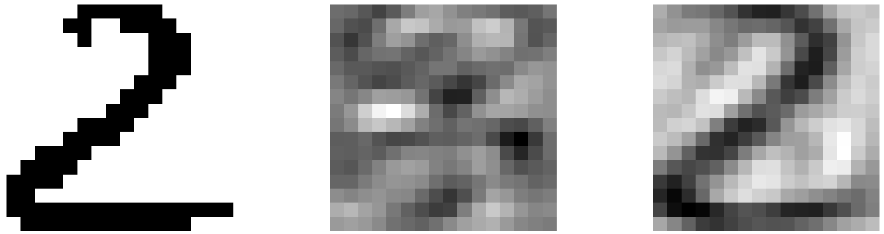
Advanced Machine Learning

Principal Component Analysis (PCA)

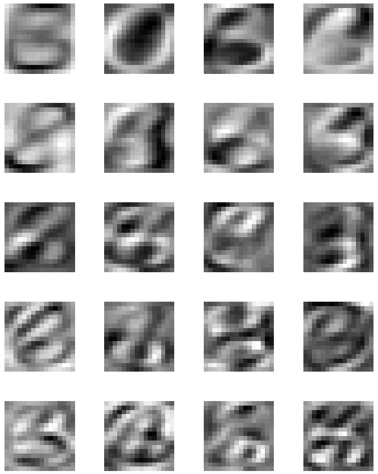
1.6 -1.1 -1.6 2.1 -0.52 2.8 0.72 0.7 -0.68 -0.41 -1.4 -1.5 -0.54 -0.62 1.3 -1.4 -0.27 0.74 0.77 -1



Covariance matrices, dimensionality reduction, PCA, Duality

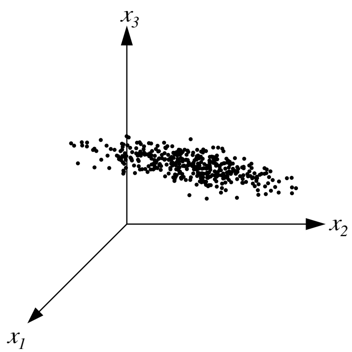
Outline

1. **Covariance Matrices**
2. Principal Component Analysis
3. Duality



Spread of Data

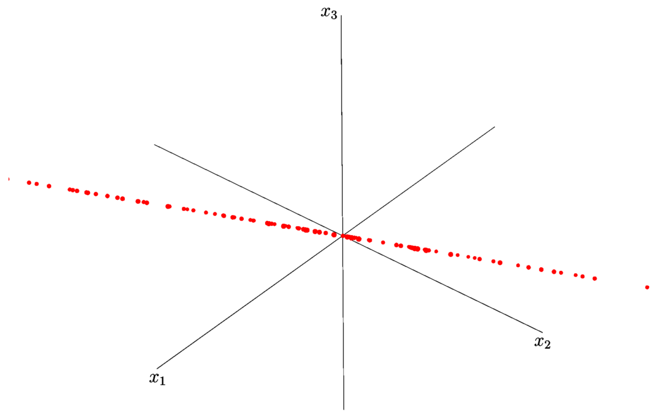
- Often data varies significantly in only some directions



- Reduce dimensions by projecting onto low dimensional subspace with maximum variation

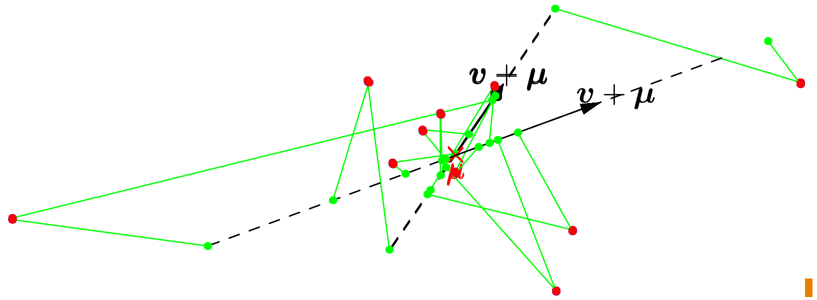
Looking is not Enough

Can't spot low dimensional data by looking at numbers



Dimensionality Reduction

- Often helpful to consider only directions where data varies significantly
- Want to find directions along which data has its greatest variation



Direction of Maximum Variation

- Look for the vector $\boldsymbol{v}$ with $\|\boldsymbol{v}\|^2 = 1$ to maximise

$$\sigma^2 = \frac{1}{m-1}\sum_{i=1}^{m}\left(\boldsymbol{v}^\mathsf{T}(\boldsymbol{x}_i - \boldsymbol{\mu})\right)^2$$

- This is a constrained optimisation problem
- Solve by maximising Lagrangian

$$\mathcal{L} = \frac{1}{m-1}\sum_{k=1}^{m}\left(\boldsymbol{v}^\mathsf{T}(\boldsymbol{x}_k - \boldsymbol{\mu})\right)^2 - \lambda\left(\|\boldsymbol{v}\|^2 - 1\right)$$

- λ is a Lagrange multiplier

Direction of Maximum Variation

- Expanding the Lagrangian

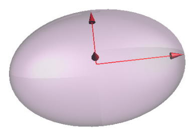
$$\begin{aligned}
\mathcal{L} &= \frac{1}{m-1}\sum_{k=1}^{m}\left(\boldsymbol{v}^\mathsf{T}(\boldsymbol{x}_k - \boldsymbol{\mu})\right)^2 - \lambda\left(\|\boldsymbol{v}\|^2 - 1\right) \\
&= \frac{1}{m-1}\sum_{k=1}^{m}\left(\boldsymbol{v}^\mathsf{T}(\boldsymbol{x}_k - \boldsymbol{\mu})(\boldsymbol{x}_k - \boldsymbol{\mu})^\mathsf{T}\boldsymbol{v}\right) - \lambda\left(\|\boldsymbol{v}\|^2 - 1\right) \\
&= \boldsymbol{v}^\mathsf{T}\left(\frac{1}{m-1}\sum_{k=1}^{m}(\boldsymbol{x}_k - \boldsymbol{\mu})(\boldsymbol{x}_k - \boldsymbol{\mu})^\mathsf{T}\right)\boldsymbol{v} - \lambda\left(\|\boldsymbol{v}\|^2 - 1\right) \\
&= \boldsymbol{v}^\mathsf{T}\mathbf{C}\boldsymbol{v} - \lambda\left(\boldsymbol{v}^\mathsf{T}\boldsymbol{v} - 1\right)
\end{aligned}$$

- Extrema of the Lagrangian

$$\boldsymbol{\nabla}\mathcal{L} = 2(\mathbf{C}\boldsymbol{v} - \lambda\boldsymbol{v}) = 0 \qquad \Rightarrow \qquad \mathbf{C}\boldsymbol{v} = \lambda\boldsymbol{v}$$

Direction of Maximum Variation

- The eigenvectors are directions that are extrema of the variance



- The variance in direction $\boldsymbol{v}$ is equal to

$$\begin{aligned}
\sigma^2 &= \frac{1}{m-1}\sum_{i=1}^{m}\left(\boldsymbol{v}^\mathsf{T}(\boldsymbol{x}_i - \boldsymbol{\mu})\right)^2 \\
&= \boldsymbol{v}^\mathsf{T}\mathbf{C}\boldsymbol{v} = \lambda\boldsymbol{v}^\mathsf{T}\boldsymbol{v} = \lambda
\end{aligned}$$

- The variance is maximised by the eigenvector with the maximum eigenvalue

Covariance Matrix

- The **covariance matrix** is defined as

$$\mathbf{C} = \frac{1}{m-1}\sum_{k=1}^{m} (\boldsymbol{x}_k - \boldsymbol{\mu})(\boldsymbol{x}_k - \boldsymbol{\mu})^{\mathsf{T}}$$

- The components C_{ij} measure how the i^{th} and j^{th} components co-vary

$$C_{ij} = \frac{1}{m-1}\sum_{k=1}^{m} (x_{ik} - \mu_i)(x_{jk} - \mu_j)$$

- C.f. covariance of random variables

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

Outer Product

- Remember that the outer-product of two vectors is defined as

$$\boldsymbol{x}\boldsymbol{y}^{\mathsf{T}} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \begin{pmatrix} y_1 & y_2 & \cdots & y_n \end{pmatrix} = \begin{pmatrix} x_1y_1 & x_1y_2 & \cdots & x_1y_n \\ x_2y_1 & x_2y_2 & \cdots & x_2y_n \\ \vdots & \vdots & \ddots & \vdots \\ x_ny_1 & x_ny_2 & \cdots & x_ny_n \end{pmatrix}$$

- C.f. Inner product

$$\boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = x_1y_1 + x_2y_2 + \cdots + x_ny_n$$

Matrix Form

- The covariance matrix is

$$\mathbf{C} = \frac{1}{m-1}\sum_{k=1}^{m} (\boldsymbol{x}_k - \boldsymbol{\mu})(\boldsymbol{x}_k - \boldsymbol{\mu})^{\mathsf{T}}$$

- Define the matrix

$$\mathbf{X} = \frac{1}{\sqrt{m-1}}(\boldsymbol{x}_1 - \boldsymbol{\mu}, \boldsymbol{x}_2 - \boldsymbol{\mu}, \cdots \boldsymbol{x}_m - \boldsymbol{\mu})$$

- We can write the covariance matrix as

$$\mathbf{C} = \mathbf{X}\mathbf{X}^{\mathsf{T}}$$

Properties of Covariance Matrix

- The **quadratic form** of a vector and matrix is defined as

$$\boldsymbol{v}^{\mathsf{T}}\mathbf{M}\boldsymbol{v}$$

- The quadratic form of a covariance matrix is non-negative for any vector

$$\boldsymbol{v}^{\mathsf{T}}\mathbf{C}\boldsymbol{v} = \boldsymbol{v}^{\mathsf{T}}\mathbf{X}\mathbf{X}^{\mathsf{T}}\boldsymbol{v} = \boldsymbol{u}^{\mathsf{T}}\boldsymbol{u} = \|\boldsymbol{u}\|^2 \geq 0$$

  where $\boldsymbol{u} = \mathbf{X}^{\mathsf{T}}\boldsymbol{v}$

- Matrices with non-negative quadratic forms are known as **positive semi-definite**

Eigenvalue Decomposition

- The eigenvectors of $\mathbf{C}$ with the largest eigenvalues are known as the **principal components**
- The eigenvalues are all greater than or equal to zero
- Recall an eigenvector $\boldsymbol{v}$ satisfies the equation

$$\mathbf{C}\boldsymbol{v} = \lambda \boldsymbol{v}$$

- Multiplying both sides by $\boldsymbol{v}^{\mathsf{T}}$

$$\boldsymbol{v}^{\mathsf{T}}\mathbf{C}\boldsymbol{v} = \lambda \boldsymbol{v}^{\mathsf{T}}\boldsymbol{v} = \lambda \|\boldsymbol{v}\|^2$$

but $\boldsymbol{v}^{\mathsf{T}}\mathbf{C}\boldsymbol{v} \geq 0$ and $\|\boldsymbol{v}\|^2 > 0$ so

$$\lambda = \frac{\boldsymbol{v}^{\mathsf{T}}\mathbf{C}\boldsymbol{v}}{\|\boldsymbol{v}\|^2} \geq 0$$

Surface Defined by Matrix

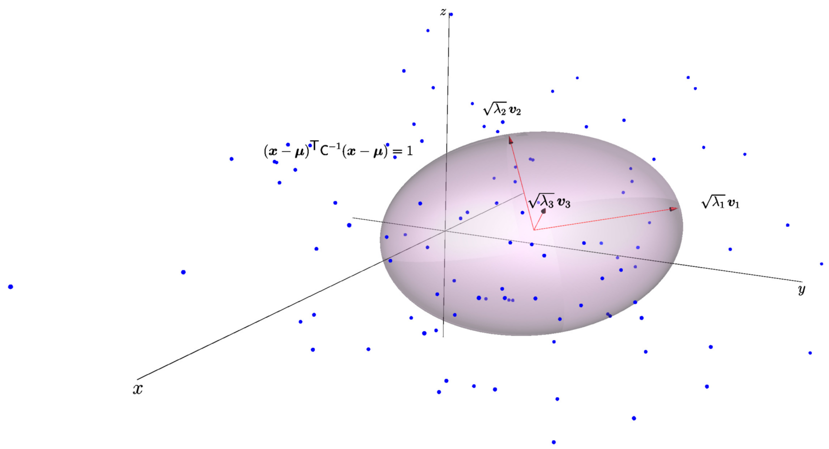
- The set of vectors $\boldsymbol{x}$ such that

$$\boldsymbol{x}^{\mathsf{T}}\mathbf{C}^{-1}\boldsymbol{x} = 1$$

defines a surface

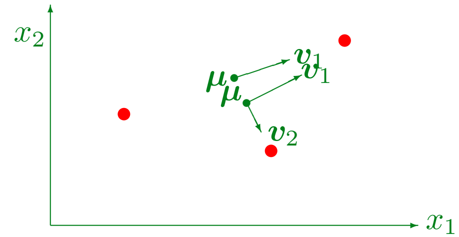
- The surface is an ellipsoid, $\mathcal{E}$
- The eigenvectors point in the direction of the principal axes of the ellipsoid
- The radii of the principal axes are equal to the square root of the eigenvalues

Ellipsoid and Eigen Space



Spanning Input Space

- A covariance matrix will have a zero eigenvalue only if there is no variation in the direction of the corresponding eigenvector
- A covariance matrix will have zero eigenvalues if the number of patterns are less than or equal to the number of dimensions
- A covariance matrix formed from $p+1$ patterns that are linearly independent (i.e. you cannot form any one out of p of the other patterns) will have no zero eigenvalues

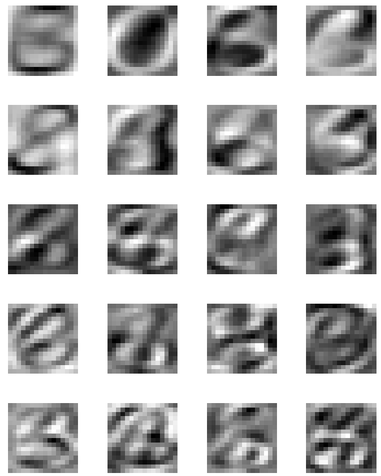


Positive Definite

- Matrices with no zero eigenvalues are called **full rank** matrices (as opposed to rank deficient)
- Full rank matrices are invertible, rank deficient matrices are singular and non-invertible
- Full rank covariance matrices have positive eigenvalues only and are said to be **positive definite**
- We would expect that when $m > p$ the covariance matrix will be positive definite unless there are some symmetries that linearly constrain the patterns

Outline

1. Covariance Matrices
2. **Principal Component Analysis**
3. Duality



Principal Component Analysis

- PCA occurs as follows
  - ⋆ Construct the covariance matrix
  - ⋆ Find the eigenvalues and eigenvectors
  - ⋆ Keep the eigenvectors with the largest eigenvalues (principal components)
  - ⋆ Project the inputs into the space spanned by the principal components
- We then use the projected inputs as inputs to our learning machine

Projection Matrix

- To project the inputs construct the projection matrix

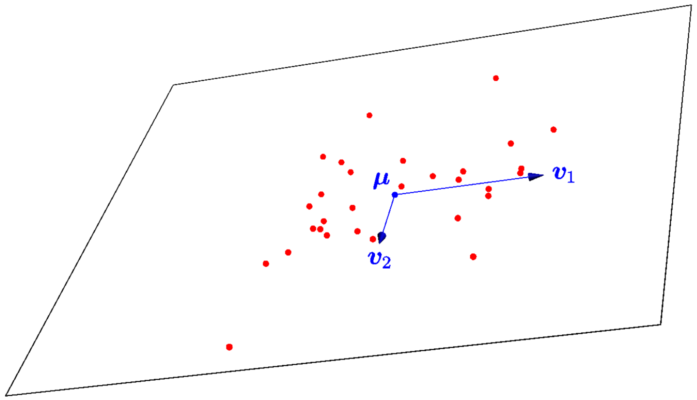
$$\mathbf{P} = \begin{pmatrix} \boldsymbol{v}_1^{\mathsf{T}} \\ \boldsymbol{v}_2^{\mathsf{T}} \\ \vdots \\ \boldsymbol{v}_k^{\mathsf{T}} \end{pmatrix}$$

- $k < p$ is the number of principal components we keep
- Given a p-dimensional input pattern $\boldsymbol{x}$ we can construct a k-dimensional representation $\boldsymbol{z}$

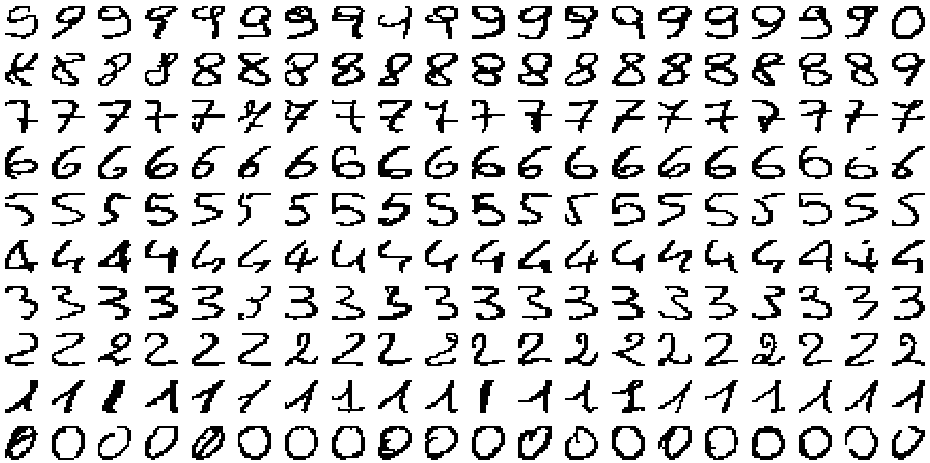
$$\boldsymbol{z} = \mathbf{P}(\boldsymbol{x} - \boldsymbol{\mu})$$

- Use $\boldsymbol{z}$ as our new inputs

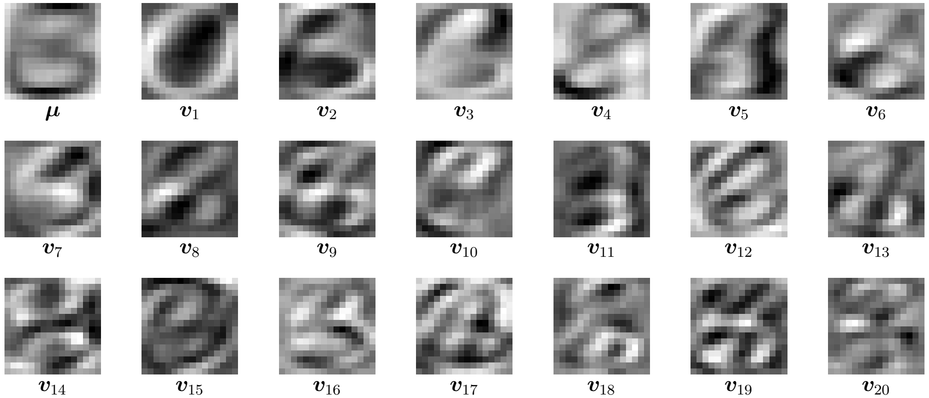
Subspace Projection



Hand Written Digits



Eigenvectors



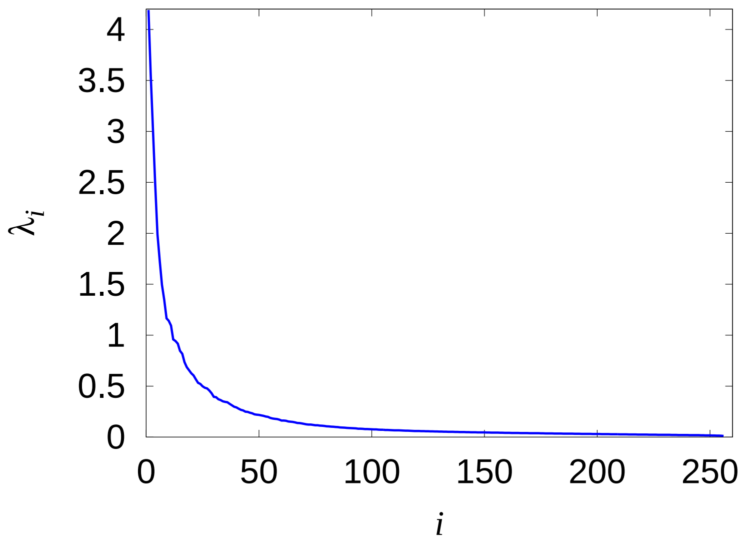
Reconstruction

- Projecting into a subspace of eigenvectors can be seen as approximating the inputs by

$$\hat{\boldsymbol{x}}_i = \boldsymbol{\mu} + \sum_{j=1}^{k} z_j^i \boldsymbol{v}_j, \qquad z_j^i = \boldsymbol{v}_j^{\mathsf{T}}(\boldsymbol{x}_i - \boldsymbol{\mu}), \qquad \|\boldsymbol{v}_j\| = 1$$

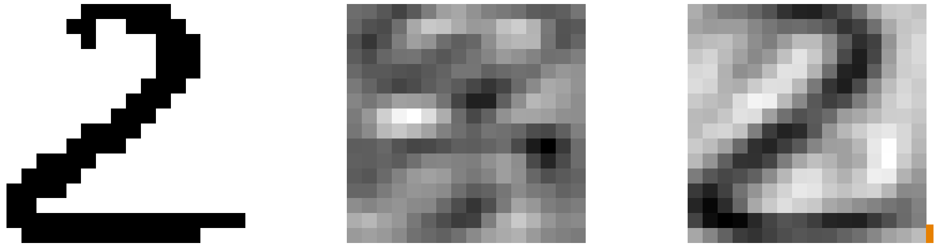
- Principle component analysis projects the data into a subspace of size m with the minimal approximation error $\mathbb{E}\left[\|\hat{\boldsymbol{x}}_i - \boldsymbol{x}_i\|^2\right]$
- The loss of "energy" (or squared error) is equal to the sum of the eigenvalues in the directions that are ignored

Eigenvalues for Digits



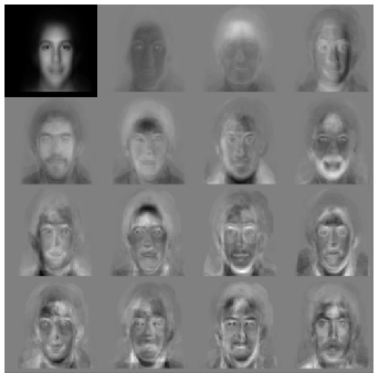
Reconstruction from Eigenvectors

1.6 -1.1 -1.6 2.1 -0.52 2.8 0.72 0.7 -0.68 -0.41 -1.4 -1.5 -0.54 -0.62 1.3 -1.4 -0.27 0.74 0.77 -1



Outline

1. Covariance Matrices
2. Principal Component Analysis
3. **Duality**



PCA for Images

- An image often contains around $p = 256 \times 256 = 64k$ pixels
- In standard PCA we would create an $p \times p$ matrix with over 4×10^9 elements
- This is intractable
- m images span at most a $m - 1$ dimensional subspace
- Usually this subspace will be much smaller than the space of all images $m \ll p$

Dual Matrix

- The covariance $\mathbf{C} = \mathbf{X}\mathbf{X}^\mathsf{T}$ is a $p \times p$ matrix
- Consider the $m \times m$ matrix $\mathbf{D} = \mathbf{X}^\mathsf{T}\mathbf{X}$
- Suppose $\boldsymbol{v}$ is an eigenvector of $\mathbf{D}$

$$\begin{aligned}
\mathbf{D}\boldsymbol{v} &= \lambda\boldsymbol{v} \\
\mathbf{X}^\mathsf{T}\mathbf{X}\boldsymbol{v} &= \lambda\boldsymbol{v} \\
\mathbf{X}\mathbf{X}^\mathsf{T}\mathbf{X}\boldsymbol{v} &= \lambda\mathbf{X}\boldsymbol{v} \\
\mathbf{C}\mathbf{X}\boldsymbol{v} &= \lambda\mathbf{X}\boldsymbol{v} \quad \Rightarrow \quad \mathbf{C}\boldsymbol{u} = \lambda\boldsymbol{u}
\end{aligned}$$

- $\boldsymbol{u} = \mathbf{X}\boldsymbol{v}$ (and $\boldsymbol{v} \propto \mathbf{X}^\mathsf{T}\boldsymbol{u}$)

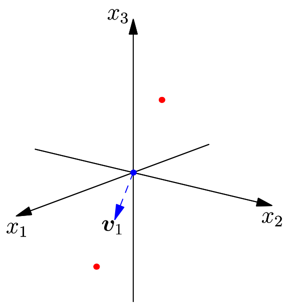
Dual Matrix

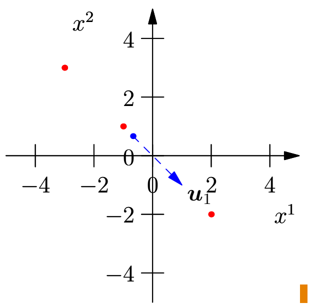
- Matrices $\mathbf{C} = \mathbf{X}\mathbf{X}^\mathsf{T}$ and $\mathbf{D} = \mathbf{X}^\mathsf{T}\mathbf{X}$ have the same eigenvalues
- Can use the dual $m \times m$ matrix $\mathbf{D}$ to find eigenvalues and eigenvectors of $\mathbf{C}$
- Note that $\mathbf{D} = \mathbf{X}^\mathsf{T}\mathbf{X}$ has components $D_{kl} \propto (\boldsymbol{x}_k - \boldsymbol{\mu})^\mathsf{T}(\boldsymbol{x}_l - \boldsymbol{\mu})$
- Takes $O(p \times m \times m)$ time to construct $\mathbf{D}$
- We work in a "dual space" which is the space spanned by the examples

What Does a Subspace Look Like?

- Consider $\boldsymbol{y}^1 = \begin{pmatrix} 2 \\ 4 \\ 4 \end{pmatrix}$, $\boldsymbol{y}^2 = \begin{pmatrix} 8 \\ 6 \\ 2 \end{pmatrix}$ with mean $\boldsymbol{\mu} = \begin{pmatrix} 5 \\ 5 \\ 3 \end{pmatrix}$
- Subtracting the mean $\boldsymbol{x}^i = \boldsymbol{y}^i - \boldsymbol{\mu}$ we can construct matrix

$$\mathbf{X} = \begin{pmatrix} x_1^1 & x_1^2 \\ x_2^1 & x_2^2 \\ x_3^1 & x_3^2 \end{pmatrix} = \begin{pmatrix} -3 & 3 \\ -1 & 1 \\ 2 & -2 \end{pmatrix}$$





Summary

- PCA allows us to reduce the dimensionality of the inputs
- We project the inputs into a sub-space where the data varies the most
- We can work in either the original space ($\mathbf{X}\mathbf{X}^\mathsf{T}$) or the dual space ($\mathbf{X}^\mathsf{T}\mathbf{X}$)
- When we have many more features than examples (i.e. $p \gg m$) then it is more efficient working in the dual space
- We will see examples of dual spaces again when we look at SVMs