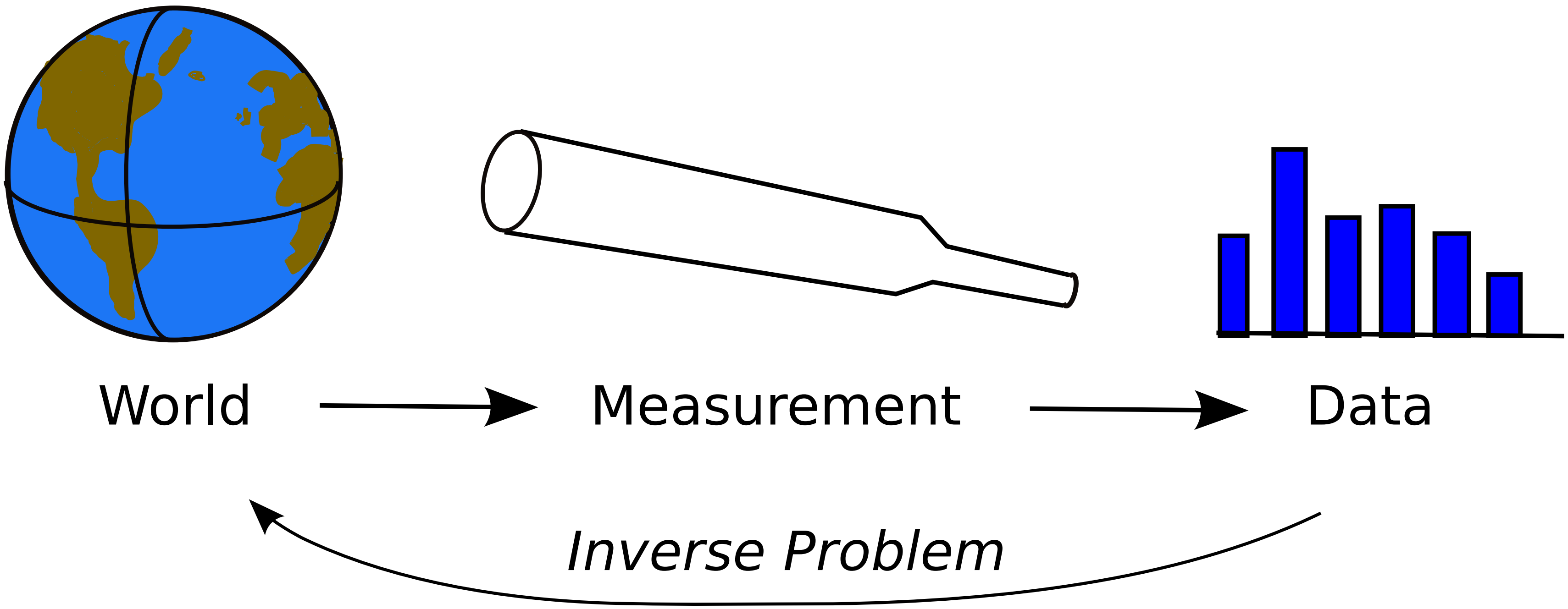
Advanced Machine Learning

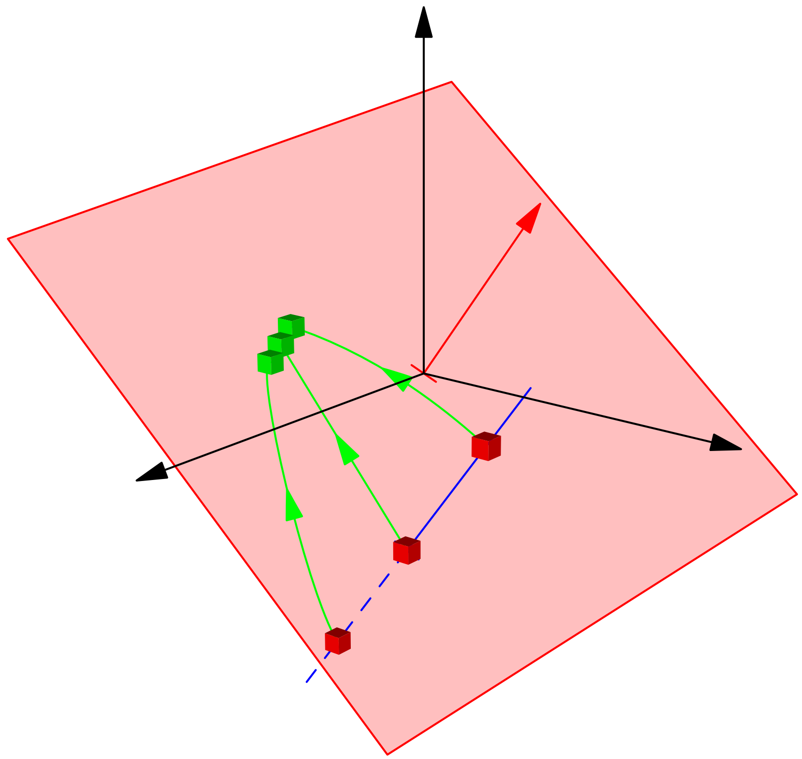
Understand Mappings



Mappings, Eigenvectors

Outline

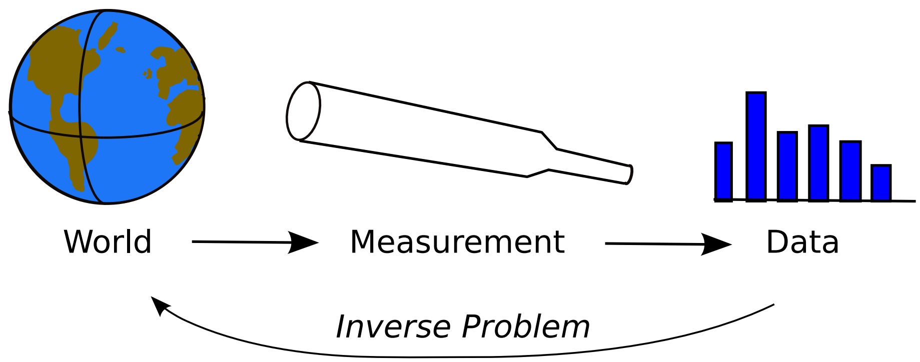
1. **Mappings**
2. Linear Maps
3. Eigenvectors



Transforming Data

- In the last lecture we spent time developing a sophisticate view of vector spaces and operators

- At a mathematical level machine learning can be viewed as performing an inverse mapping

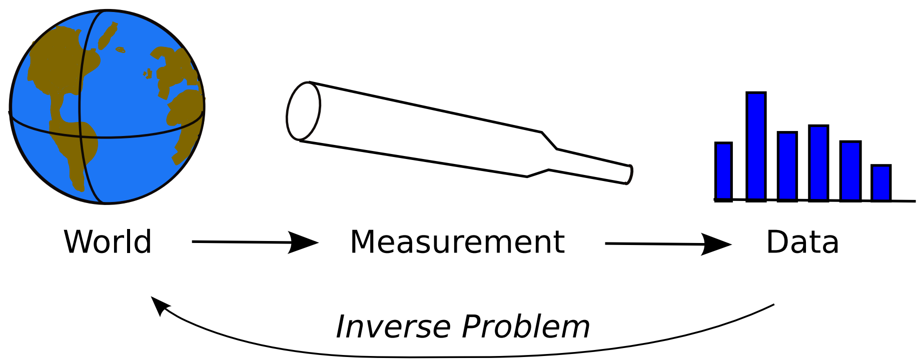


- Although our mappings are not necessarily linear in either direction we learn a lot by understanding linear operators

Transforming Data

- In the last lecture we spent time developing a sophisticate view of vector spaces and operators

- At a mathematical level machine learning can be viewed as performing an inverse mapping

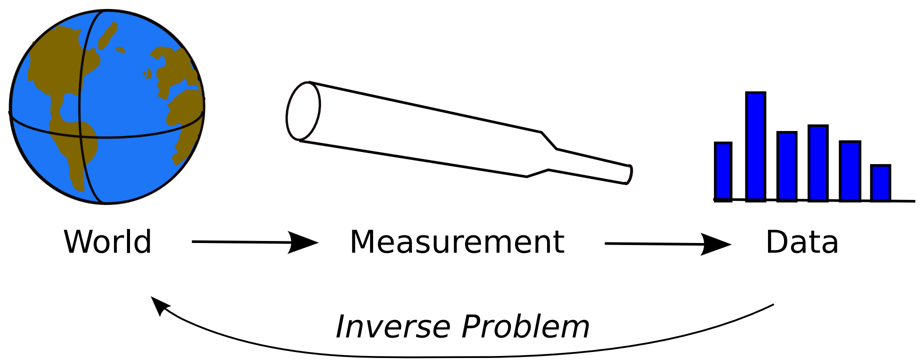


- Although our mappings are not necessarily linear in either direction we learn a lot by understanding linear operators

Transforming Data

- In the last lecture we spent time developing a sophisticate view of vector spaces and operators

- At a mathematical level machine learning can be viewed as performing an inverse mapping



- Although our mappings are not necessarily linear in either direction we learn a lot by understanding linear operators

Inverse Problems

- Given m observations $\{(\boldsymbol{x}_k, y_k) | k = 1, \ldots, m\}$ and p unknown $\boldsymbol{w} = (w_1, w_2, \ldots w_p)$ such that $\boldsymbol{x}_k^\mathsf{T} \boldsymbol{w} = y_k$ then to find $\boldsymbol{w}$
- Define the *design matrix* as the matrix of feature vectors

$$\mathbf{X} = \begin{pmatrix} \boldsymbol{x}_1^\mathsf{T} \\ \boldsymbol{x}_2^\mathsf{T} \\ \ldots \\ \boldsymbol{x}_m^\mathsf{T} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mp} \end{pmatrix}$$

- and the target vector $\boldsymbol{y} = (y_1, y_2, \cdots, y_m)^\mathsf{T}$
- Then if $m = p$ we have $\boldsymbol{y} = \mathbf{X}\boldsymbol{w}$ or $\boldsymbol{w} = \mathbf{X}^{-1}\boldsymbol{y}$

Inverse Problems

- Given m observations $\{(\boldsymbol{x}_k, y_k) | k = 1, \dots, m\}$ and p unknown $\boldsymbol{w} = (w_1, w_2, \dots w_p)$ such that $\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} = y_k$ then to find $\boldsymbol{w}$

- Define the *design matrix* as the matrix of feature vectors

$$\mathbf{X} = \begin{pmatrix} \boldsymbol{x}_1^{\mathsf{T}} \\ \boldsymbol{x}_2^{\mathsf{T}} \\ \dots \\ \boldsymbol{x}_m^{\mathsf{T}} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mp} \end{pmatrix}$$

- and the target vector $\boldsymbol{y} = (y_1, y_2, \cdots, y_m)^{\mathsf{T}}$

- Then if $m = p$ we have $\boldsymbol{y} = \mathbf{X}\boldsymbol{w}$ or $\boldsymbol{w} = \mathbf{X}^{-1}\boldsymbol{y}$

Inverse Problems

- Given m observations $\{(\boldsymbol{x}_k, y_k) | k = 1, \ldots, m\}$ and p unknown $\boldsymbol{w} = (w_1, w_2, \ldots w_p)$ such that $\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} = y_k$ then to find $\boldsymbol{w}$

- Define the *design matrix* as the matrix of feature vectors

$$\mathbf{X} = \begin{pmatrix} \boldsymbol{x}_1^{\mathsf{T}} \\ \boldsymbol{x}_2^{\mathsf{T}} \\ \ldots \\ \boldsymbol{x}_m^{\mathsf{T}} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mp} \end{pmatrix}$$

- and the target vector $\boldsymbol{y} = (y_1, y_2, \cdots, y_m)^{\mathsf{T}}$

- Then if $m = p$ we have $\boldsymbol{y} = \mathbf{X}\boldsymbol{w}$ or $\boldsymbol{w} = \mathbf{X}^{-1}\boldsymbol{y}$

Inverse Problems

- Given m observations $\{(\boldsymbol{x}_k, y_k) | k = 1, \ldots, m\}$ and p unknown $\boldsymbol{w} = (w_1, w_2, \ldots w_p)$ such that $\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} = y_k$ then to find $\boldsymbol{w}$

- Define the *design matrix* as the matrix of feature vectors

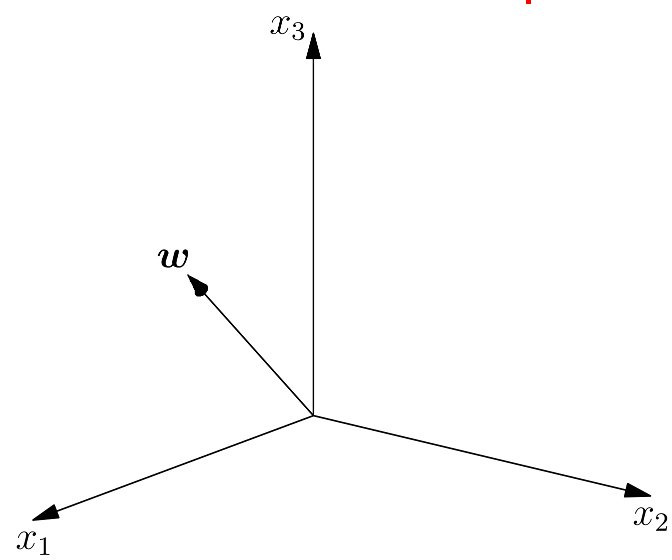
$$\mathbf{X} = \begin{pmatrix} \boldsymbol{x}_1^{\mathsf{T}} \\ \boldsymbol{x}_2^{\mathsf{T}} \\ \ldots \\ \boldsymbol{x}_m^{\mathsf{T}} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mp} \end{pmatrix}$$

- and the target vector $\boldsymbol{y} = (y_1, y_2, \cdots, y_m)^{\mathsf{T}}$

- Then if $m = p$ we have $\boldsymbol{y} = \mathbf{X}\boldsymbol{w}$ or $\boldsymbol{w} = \mathbf{X}^{-1}\boldsymbol{y}$

Linear Regression

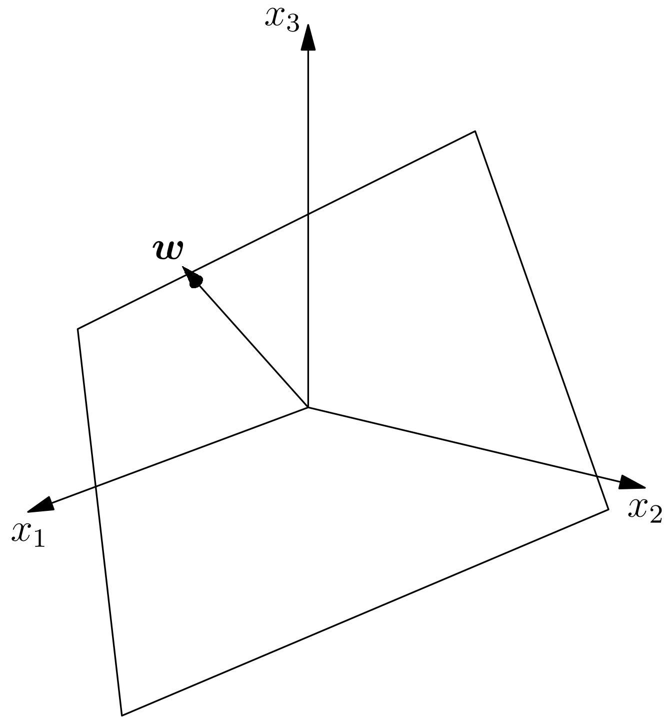
- $y_k = \boldsymbol{x}_k^\mathsf{T} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

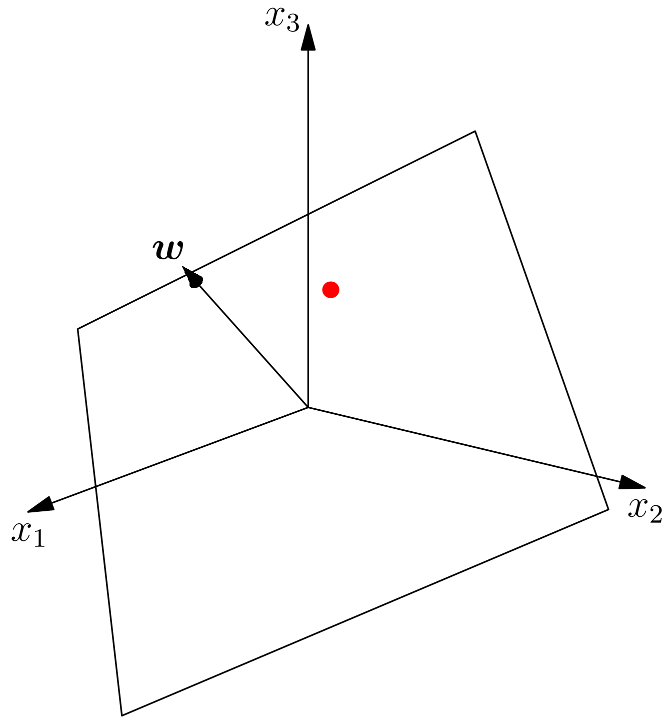
- $y_k = \boldsymbol{x}_k^\mathsf{T} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

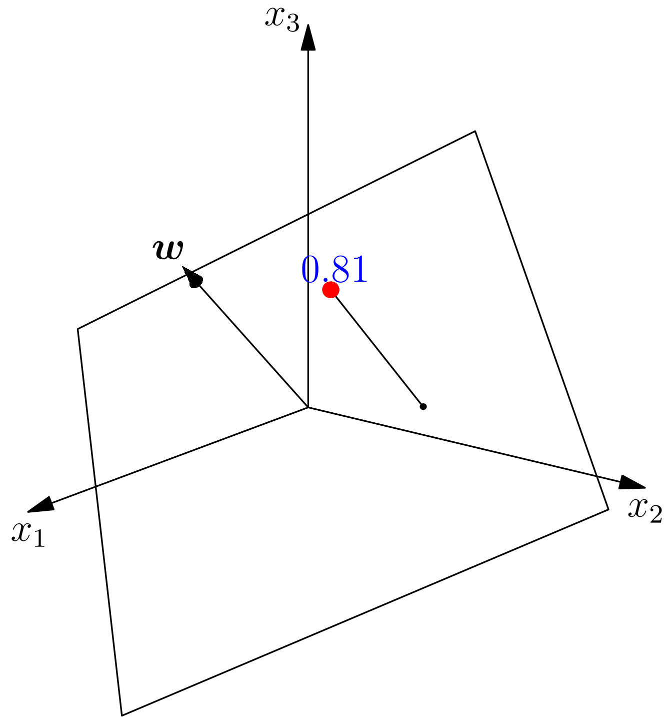
- $y_k = \boldsymbol{x}_k^\mathsf{T}\boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

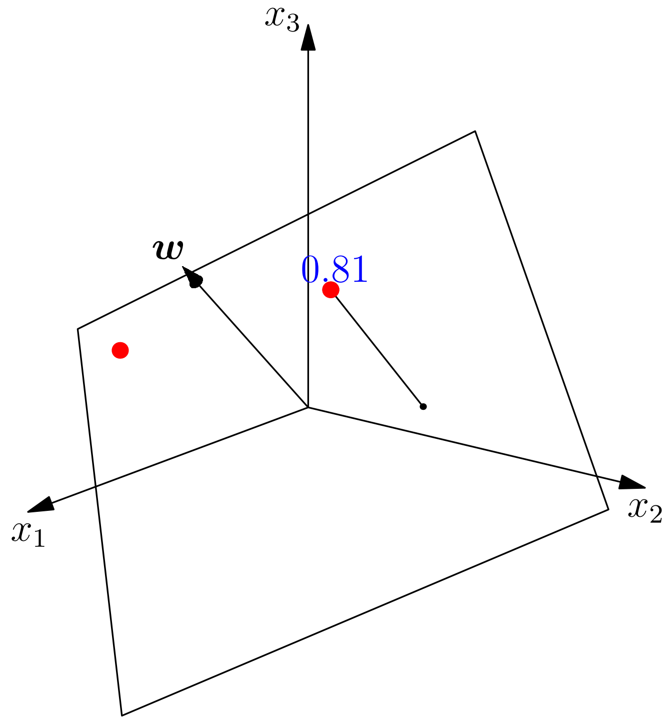
- $y_k = \boldsymbol{x}_k^\mathsf{T} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

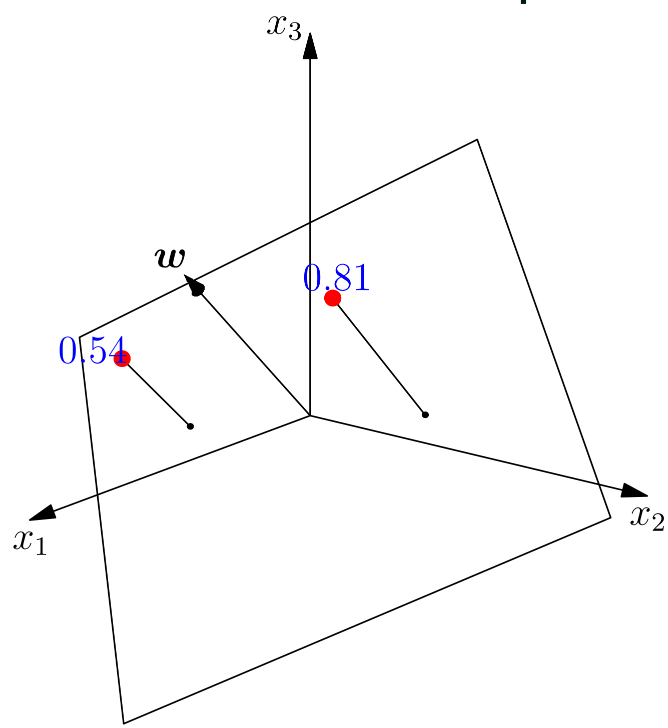
- $y_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

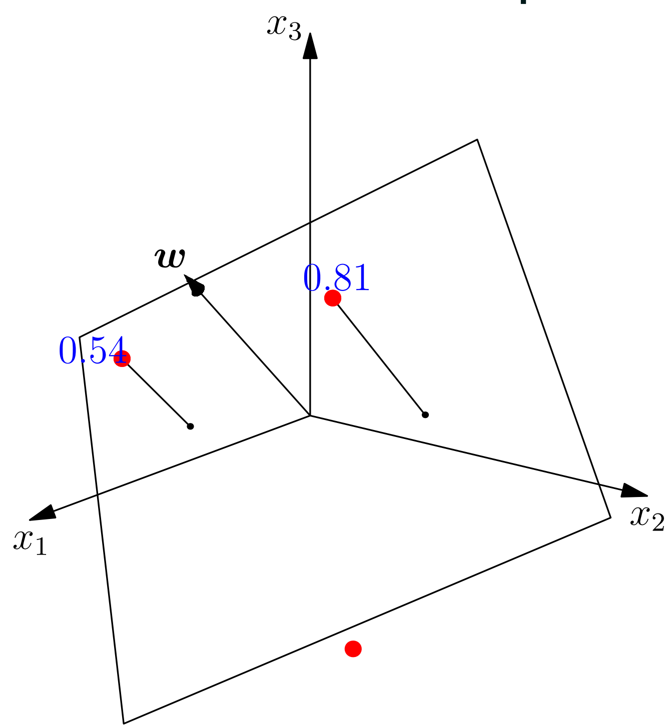
- $y_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

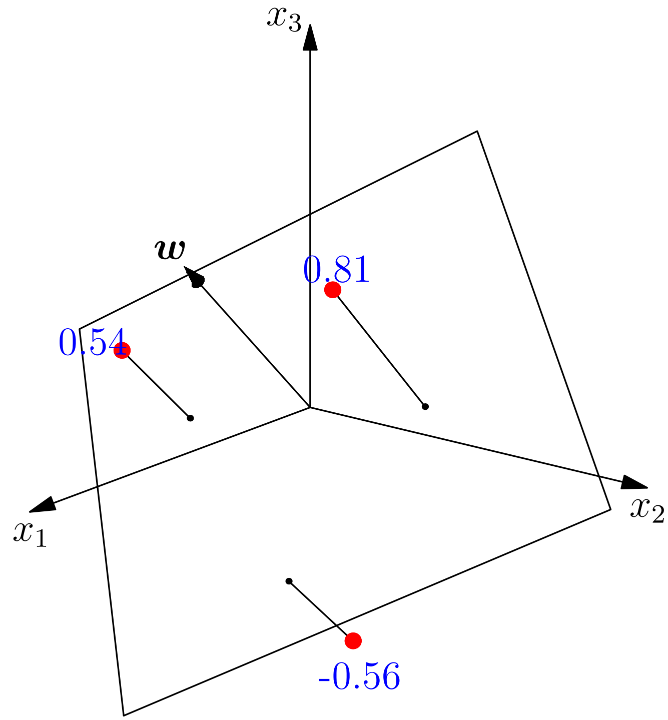
- $y_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

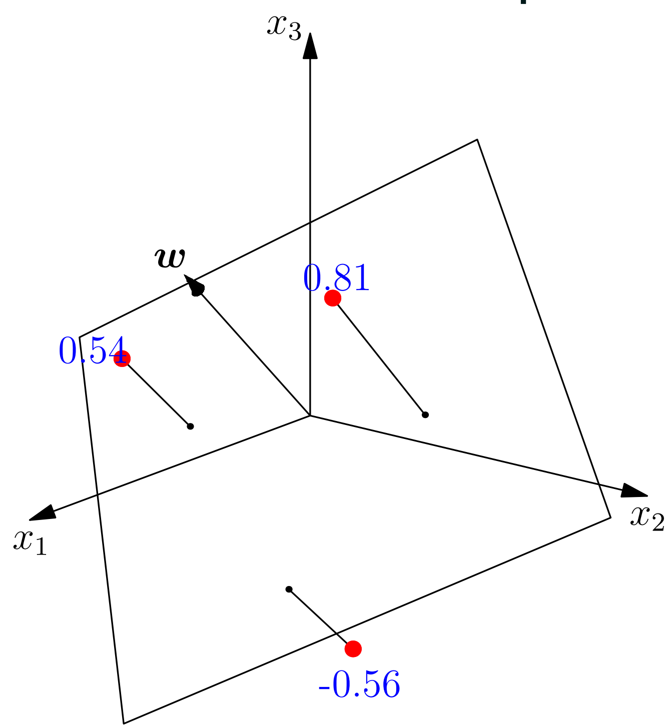
- $y_k = \boldsymbol{x}_k^\mathsf{T} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

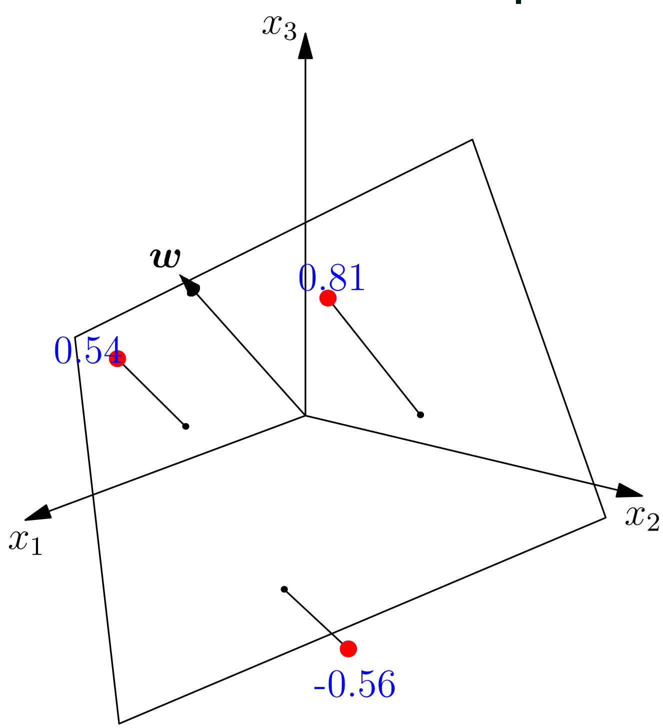
- $y_k = \boldsymbol{x}_k^\top \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

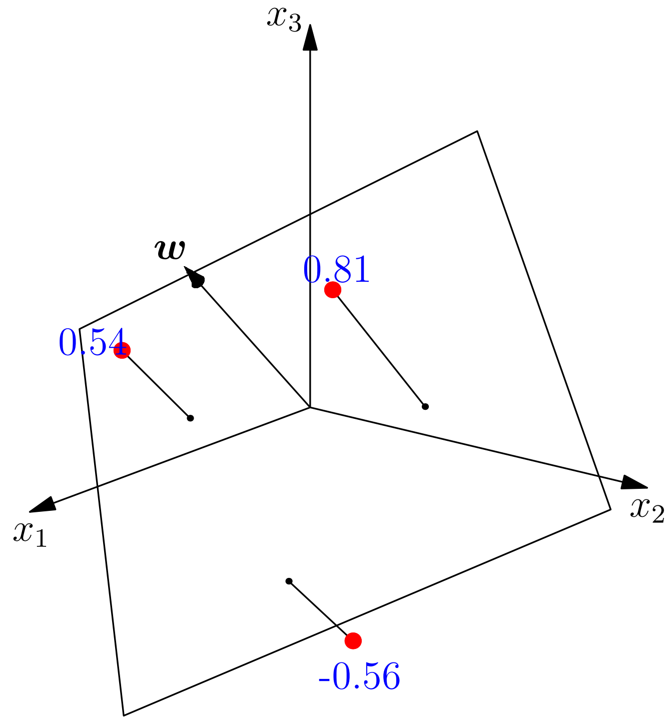
- $y_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Regression

- $y_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w}$ depends on distance from separating plane



- If $m > p$ then $\mathbf{X}$ isn't square so doesn't have an inverse
- Worse unless the data is accurate $\boldsymbol{y} \approx \mathbf{X}\boldsymbol{w} \Rightarrow$ no "solution"
- Problem solved by Gauss to predict the orbit of the asteroid Ceres

Linear Least Squares

- The error of input pattern $\boldsymbol{x}_k$ is

$$\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$$

- The squared error

$$E(\boldsymbol{w}|\mathcal{D}) = \sum_{k=1}^{P} \left(\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k\right)^2 = \sum_{k=1}^{P} \epsilon_k^2 = \|\boldsymbol{\epsilon}\|^2$$

- We can define the error vector

$$\boldsymbol{\epsilon} = \mathbf{X}\boldsymbol{w} - \boldsymbol{y}$$

(note that $\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$)

- Minimising this error is known as the least squares problem

Linear Least Squares

- The error of input pattern $\boldsymbol{x}_k$ is

$$\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$$

- The squared error

$$E(\boldsymbol{w}|\mathcal{D}) = \sum_{k=1}^{P} \left(\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k\right)^2 = \sum_{k=1}^{P} \epsilon_k^2 = \|\boldsymbol{\epsilon}\|^2$$

- We can define the error vector

$$\boldsymbol{\epsilon} = \mathbf{X}\boldsymbol{w} - \boldsymbol{y}$$

(note that $\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$)

- Minimising this error is known as the least squares problem

Linear Least Squares

- The error of input pattern $\boldsymbol{x}_k$ is

$$\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$$

- The squared error

$$E(\boldsymbol{w}|\mathcal{D}) = \sum_{k=1}^{P} \left(\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k\right)^2 = \sum_{k=1}^{P} \epsilon_k^2 = \|\boldsymbol{\epsilon}\|^2$$

- We can define the error vector

$$\boldsymbol{\epsilon} = \mathbf{X}\boldsymbol{w} - \boldsymbol{y}$$

(note that $\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$)

- Minimising this error is known as the least squares problem

Linear Least Squares

- The error of input pattern $\boldsymbol{x}_k$ is

$$\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$$

- The squared error

$$E(\boldsymbol{w}|\mathcal{D}) = \sum_{k=1}^{P} \left(\boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k\right)^2 = \sum_{k=1}^{P} \epsilon_k^2 = \|\boldsymbol{\epsilon}\|^2$$

- We can define the error vector

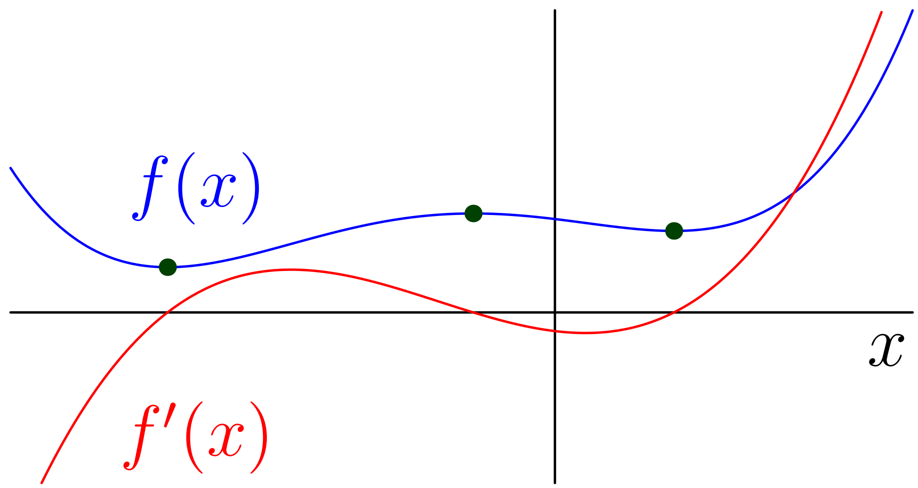
$$\boldsymbol{\epsilon} = \mathbf{X}\boldsymbol{w} - \boldsymbol{y}$$

(note that $\epsilon_k = \boldsymbol{x}_k^{\mathsf{T}} \boldsymbol{w} - y_k$)

- Minimising this error is known as the least squares problem

Finding a Minimum

- The minima of a one dimensional function, $f(x)$, are given by $f'(x) = 0$

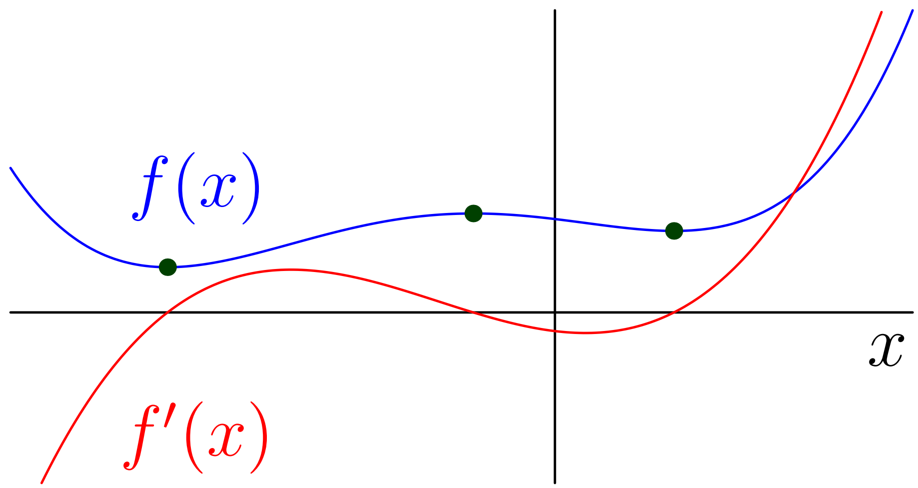


- The minima of an n-dimensions function $f(\boldsymbol{x})$ are given by the set of equations

$$\frac{\partial f(\boldsymbol{x})}{\partial x_i} = 0 \quad \forall i = 1, \ldots n$$

Finding a Minimum

- The minima of a one dimensional function, $f(x)$, are given by $f'(x) = 0$



- The minima of an n-dimensions function $f(\boldsymbol{x})$ are given by the set of equations

$$\frac{\partial f(\boldsymbol{x})}{\partial x_i} = 0 \quad \forall i = 1, \ldots n$$

Gradients

- The **grad** operator $\boldsymbol{\nabla}$ is the gradient operator in high dimensions

$$\boldsymbol{\nabla} f(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial f(\boldsymbol{x})}{\partial x_1} \\ \frac{\partial f(\boldsymbol{x})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\boldsymbol{x})}{\partial x_n} \end{pmatrix}$$

- The partial derivatives (curly d's)

$$\frac{\partial f(\boldsymbol{x})}{\partial x_i}$$

means differentiate with respect to x_i treating all other components x_j as constants

Gradients

- The **grad** operator $\boldsymbol{\nabla}$ is the gradient operator in high dimensions

$$\boldsymbol{\nabla} f(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial f(\boldsymbol{x})}{\partial x_1} \\ \frac{\partial f(\boldsymbol{x})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\boldsymbol{x})}{\partial x_n} \end{pmatrix}$$

- The partial derivatives (curly d's)

$$\frac{\partial f(\boldsymbol{x})}{\partial x_i}$$

means differentiate with respect to x_i treating all other components x_j as constants

Least Squares Solution

- The least squared solution is give by

$$\begin{aligned} \nabla E(\boldsymbol{w}|\mathcal{D}) = \nabla \|\boldsymbol{\epsilon}\|^2 &= \nabla \|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\ &= \nabla \left(\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} + \boldsymbol{y}^{\mathsf{T}}\boldsymbol{y} \right) \\ &= 2\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^{\mathsf{T}}\boldsymbol{y} \right) = 0 \end{aligned}$$

- Or

$$\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$\begin{aligned}\boldsymbol{\nabla} E(\boldsymbol{w}|\mathcal{D}) &= \boldsymbol{\nabla}\|\boldsymbol{\epsilon}\|^2 = \boldsymbol{\nabla}\|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\ &= \boldsymbol{\nabla}\left(\boldsymbol{w}^\mathsf{T}\mathbf{X}^\mathsf{T}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^\mathsf{T}\mathbf{X}^\mathsf{T}\boldsymbol{y} + \boldsymbol{y}^\mathsf{T}\boldsymbol{y}\right) \\ &= 2\left(\mathbf{X}^\mathsf{T}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^\mathsf{T}\boldsymbol{y}\right) = 0\end{aligned}$$

- Or

$$\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$
\begin{aligned}
\boldsymbol{\nabla} E(\boldsymbol{w}|\mathcal{D}) = \boldsymbol{\nabla} \|\boldsymbol{\epsilon}\|^2 &= \boldsymbol{\nabla} \|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\
&= \boldsymbol{\nabla} \left(\boldsymbol{w}^\mathsf{T} \mathbf{X}^\mathsf{T} \mathbf{X}\, \boldsymbol{w} - 2\boldsymbol{w}^\mathsf{T} \mathbf{X}^\mathsf{T} \boldsymbol{y} + \boldsymbol{y}^\mathsf{T} \boldsymbol{y} \right) \\
&= 2 \left(\mathbf{X}^\mathsf{T} \mathbf{X}\, \boldsymbol{w} - \mathbf{X}^\mathsf{T} \boldsymbol{y} \right) \; = 0
\end{aligned}
$$

- Or

$$
\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T} \mathbf{X} \right)^{-1} \mathbf{X}^\mathsf{T} \boldsymbol{y} = \mathbf{X}^+ \boldsymbol{y}
$$

- $\mathbf{X}^+ = \left(\mathbf{X}^\mathsf{T} \mathbf{X} \right)^{-1} \mathbf{X}^\mathsf{T}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$\begin{aligned}\boldsymbol{\nabla} E(\boldsymbol{w}|\mathcal{D}) &= \boldsymbol{\nabla}\|\boldsymbol{\epsilon}\|^2 = \boldsymbol{\nabla}\|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\ &= \boldsymbol{\nabla}\left(\boldsymbol{w}^\mathsf{T}\mathbf{X}^\mathsf{T}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^\mathsf{T}\mathbf{X}^\mathsf{T}\boldsymbol{y} + \boldsymbol{y}^\mathsf{T}\boldsymbol{y}\right) \\ &= 2\left(\mathbf{X}^\mathsf{T}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^\mathsf{T}\boldsymbol{y}\right) = 0\end{aligned}$$

- Or

$$\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$
\begin{aligned}
\boldsymbol{\nabla} E(\boldsymbol{w}|\mathcal{D}) &= \boldsymbol{\nabla} \|\boldsymbol{\epsilon}\|^2 = \boldsymbol{\nabla} \|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\
&= \boldsymbol{\nabla} \left(\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} + \boldsymbol{y}^{\mathsf{T}}\boldsymbol{y}\right) \\
&= 2\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^{\mathsf{T}}\boldsymbol{y}\right) = 0
\end{aligned}
$$

- Or

$$
\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}
$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$\begin{aligned} \nabla E(\boldsymbol{w}|\mathcal{D}) &= \nabla \|\boldsymbol{\epsilon}\|^2 = \nabla \|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\ &= \nabla \left(\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} + \boldsymbol{y}^{\mathsf{T}}\boldsymbol{y}\right) \\ &= 2\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^{\mathsf{T}}\boldsymbol{y}\right) = 0 \end{aligned}$$

- Or

$$\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$
\begin{aligned}
\boldsymbol{\nabla} E(\boldsymbol{w}|\mathcal{D}) &= \boldsymbol{\nabla}\|\boldsymbol{\epsilon}\|^2 = \boldsymbol{\nabla}\|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\
&= \boldsymbol{\nabla}\left(\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} + \boldsymbol{y}^{\mathsf{T}}\boldsymbol{y}\right) \\
&= 2\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^{\mathsf{T}}\boldsymbol{y}\right) = 0
\end{aligned}
$$

- Or

$$
\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}
$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Least Squares Solution

- The least squared solution is give by

$$\begin{aligned}\boldsymbol{\nabla} E(\boldsymbol{w}|\mathcal{D}) &= \boldsymbol{\nabla}\|\boldsymbol{\epsilon}\|^2 = \boldsymbol{\nabla}\|\mathbf{X}\boldsymbol{w} - \boldsymbol{y}\|^2 \\ &= \boldsymbol{\nabla}\left(\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - 2\boldsymbol{w}^{\mathsf{T}}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} + \boldsymbol{y}^{\mathsf{T}}\boldsymbol{y}\right) \\ &= 2\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\,\boldsymbol{w} - \mathbf{X}^{\mathsf{T}}\boldsymbol{y}\right) = 0\end{aligned}$$

- Or

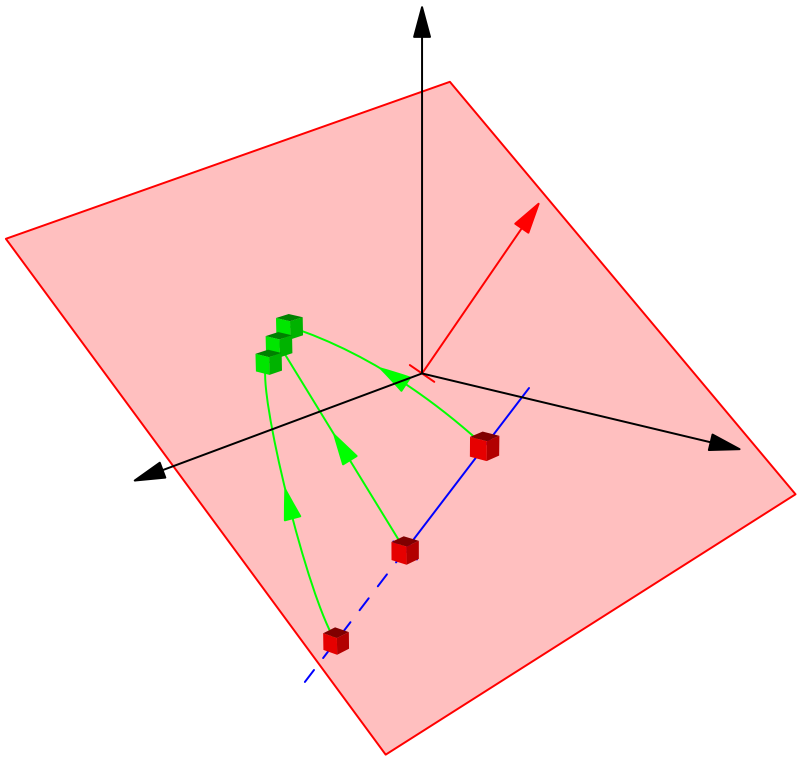
$$\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y} = \mathbf{X}^{+}\boldsymbol{y}$$

- $\mathbf{X}^{+} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}$ is known as the pseudo inverse
- For non-square matrices Matlab uses the pseudo inverse so in Matlab we can write

```
w = X\y
```

Outline

1. Mappings
2. **Linear Maps**
3. Eigenvectors



Solving Inverse Problems

- Gauss showed us how to solve **over-constrained** problems (we have more observations than parameters)
- We seek a solution which isn't necessarily exact but minimises an error
- But, what if we have more parameters than observations
- That is, we are **under-constrained**
- Note that in some directions you might be over-constrained and in other directions under-constrained

Solving Inverse Problems

- Gauss showed us how to solve **over-constrained** problems (we have more observations than parameters)
- We seek a solution which isn't necessarily exact but minimises an error
- But, what if we have more parameters than observations
- That is, we are **under-constrained**
- Note that in some directions you might be over-constrained and in other directions under-constrained

Solving Inverse Problems

- Gauss showed us how to solve **over-constrained** problems (we have more observations than parameters)
- We seek a solution which isn't necessarily exact but minimises an error
- But, what if we have more parameters than observations
- That is, we are **under-constrained**
- Note that in some directions you might be over-constrained and in other directions under-constrained

Solving Inverse Problems

- Gauss showed us how to solve **over-constrained** problems (we have more observations than parameters)
- We seek a solution which isn't necessarily exact but minimises an error
- But, what if we have more parameters than observations
- That is, we are **under-constrained**
- Note that in some directions you might be over-constrained and in other directions under-constrained

Solving Inverse Problems

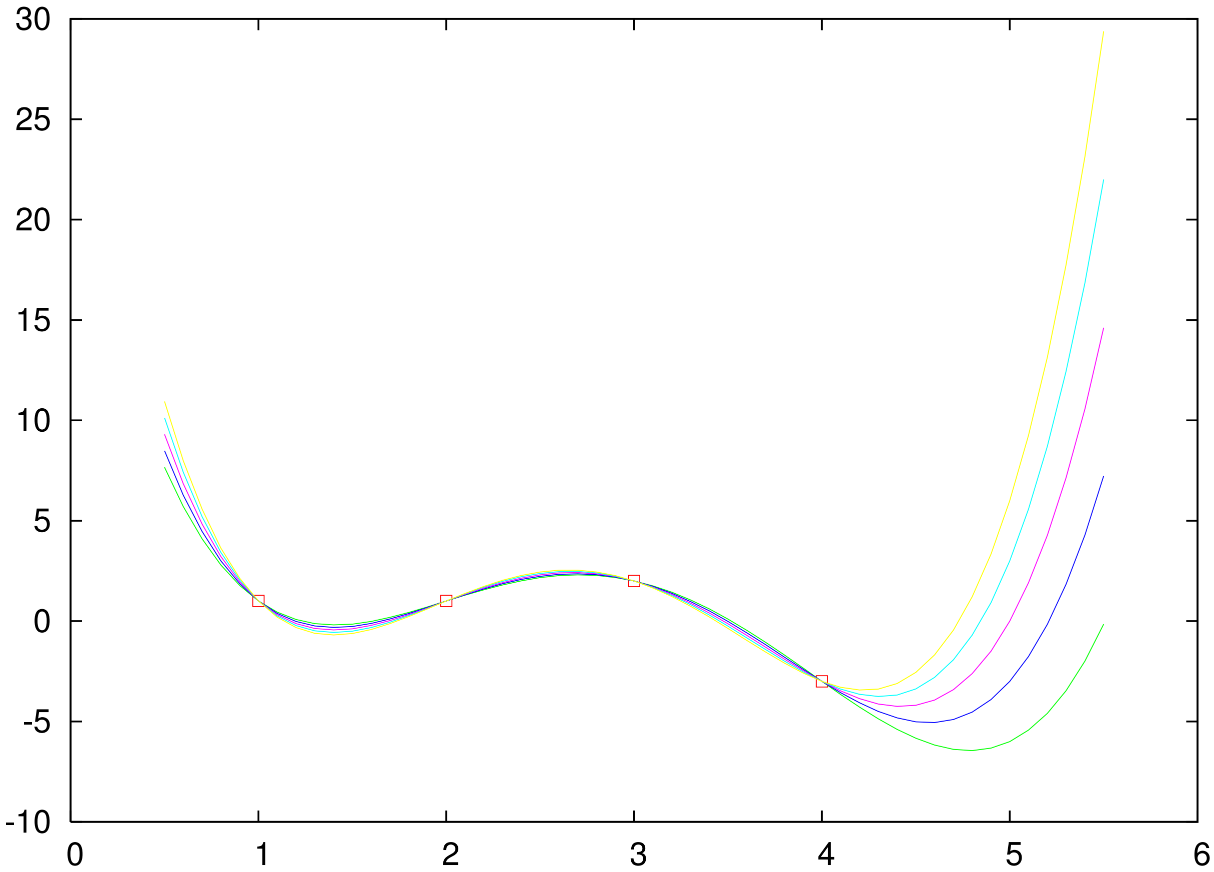
- Gauss showed us how to solve **over-constrained** problems (we have more observations than parameters)
- We seek a solution which isn't necessarily exact but minimises an error
- But, what if we have more parameters than observations
- That is, we are **under-constrained**
- Note that in some directions you might be over-constrained and in other directions under-constrained

Solving Inverse Problems

- Gauss showed us how to solve **over-constrained** problems (we have more observations than parameters)
- We seek a solution which isn't necessarily exact but minimises an error
- But, what if we have more parameters than observations
- That is, we are **under-constrained**
- Note that in some directions you might be over-constrained and in other directions under-constrained
- This is very typical of most machine learning problems

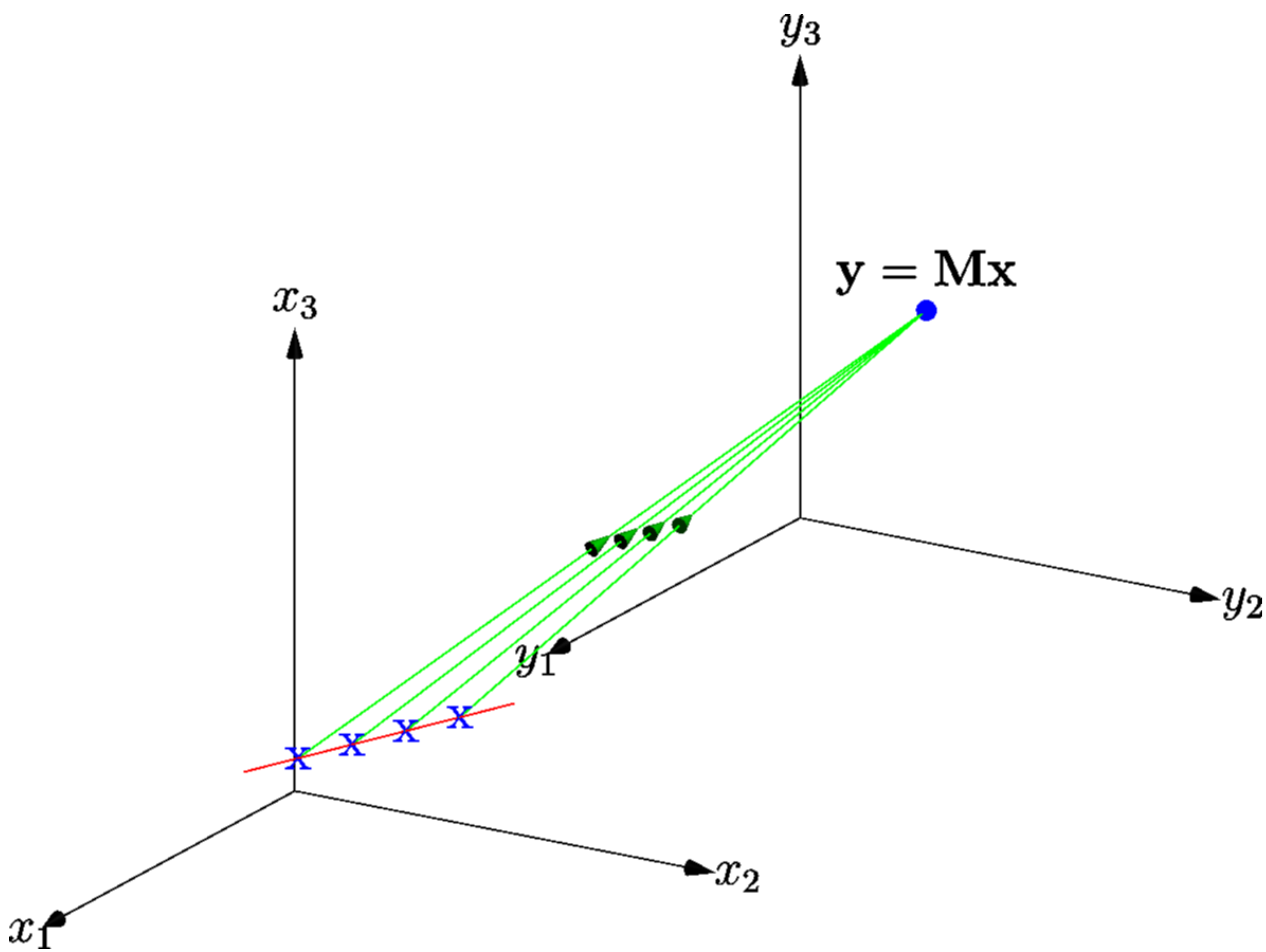
Under Constrained Systems

- If we have less data-points than parameters then there will be multiple solutions



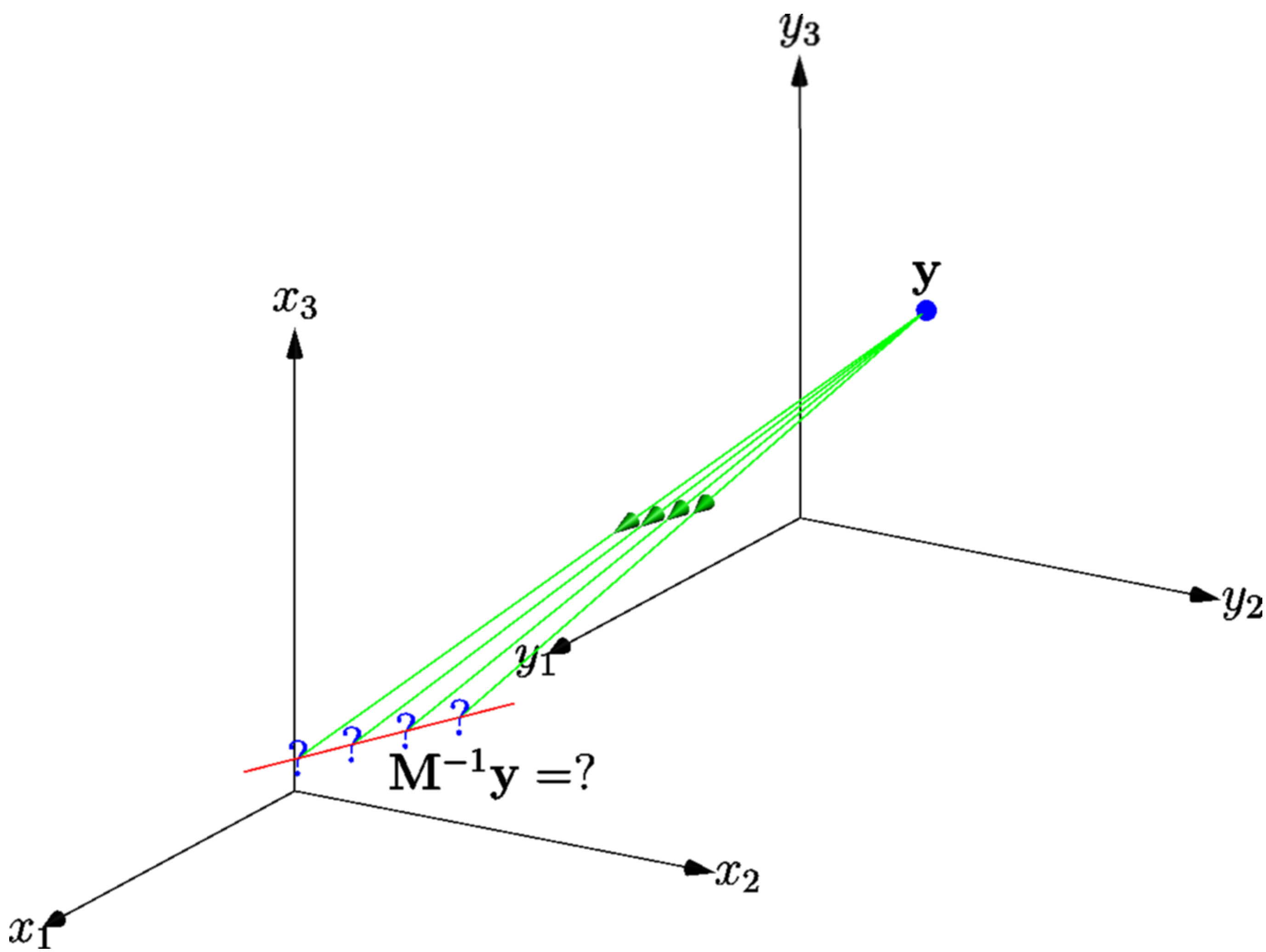
What is the Inverse?

- Many points can map to the same points



What is the Inverse?

- Many points can map to the same points



Under-constrained Systems

- The system is **under-constrained**
- We have more unknowns than equations
- The inverse is not unique
- Solving the inverse problem ($\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y}$) is said to be **ill-posed**
- The inverse $\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}$ doesn't exist
- If we have a complicated learning machine and not sufficient data we often end with an ill-posed inverse problem (there are lots of sets of parameters that explain the data)

Under-constrained Systems

- The system is **under-constrained**
- We have more unknowns than equations
- The inverse is not unique
- Solving the inverse problem ($\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y}$) is said to be **ill-posed**
- The inverse $\left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}$ doesn't exist
- If we have a complicated learning machine and not sufficient data we often end with an ill-posed inverse problem (there are lots of sets of parameters that explain the data)

Under-constrained Systems

- The system is **under-constrained**
- We have more unknowns than equations
- The inverse is not unique
- Solving the inverse problem ($\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y}$) is said to be **ill-posed**
- The inverse $\left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}$ doesn't exist
- If we have a complicated learning machine and not sufficient data we often end with an ill-posed inverse problem (there are lots of sets of parameters that explain the data)

Under-constrained Systems

- The system is **under-constrained**
- We have more unknowns than equations
- The inverse is not unique
- Solving the inverse problem ($\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y}$) is said to be **ill-posed**
- The inverse $\left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}$ doesn't exist
- If we have a complicated learning machine and not sufficient data we often end with an ill-posed inverse problem (there are lots of sets of parameters that explain the data)

Under-constrained Systems

- The system is **under-constrained**
- We have more unknowns than equations
- The inverse is not unique
- Solving the inverse problem ($\boldsymbol{w} = \left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathsf{T}}\boldsymbol{y}$) is said to be **ill-posed**
- The inverse $\left(\mathbf{X}^{\mathsf{T}}\mathbf{X}\right)^{-1}$ doesn't exist
- If we have a complicated learning machine and not sufficient data we often end with an ill-posed inverse problem (there are lots of sets of parameters that explain the data)

Under-constrained Systems

- The system is **under-constrained**
- We have more unknowns than equations
- The inverse is not unique
- Solving the inverse problem ($\boldsymbol{w} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{y}$) is said to be **ill-posed**
- The inverse $\left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}$ doesn't exist
- If we have a complicated learning machine and not sufficient data we often end with an ill-posed inverse problem (there are lots of sets of parameters that explain the data)

Ill-Conditions

- Singular matrices are rare (although they occur when we don't have enough data), but matrices that are close to being singular are common
- If a matrix is close to singular it is ill-conditioned
- Ill-conditioned matrices have some small eigenvalues
- All points get contracted towards a plane
- Large matrices are very often ill conditioned

Ill-Conditions

- Singular matrices are rare (although they occur when we don't have enough data), but matrices that are close to being singular are common
- If a matrix is close to singular it is ill-conditioned
- Ill-conditioned matrices have some small eigenvalues
- All points get contracted towards a plane
- Large matrices are very often ill conditioned

Ill-Conditions

- Singular matrices are rare (although they occur when we don't have enough data), but matrices that are close to being singular are common
- If a matrix is close to singular it is ill-conditioned
- Ill-conditioned matrices have some small eigenvalues
- All points get contracted towards a plane
- Large matrices are very often ill conditioned

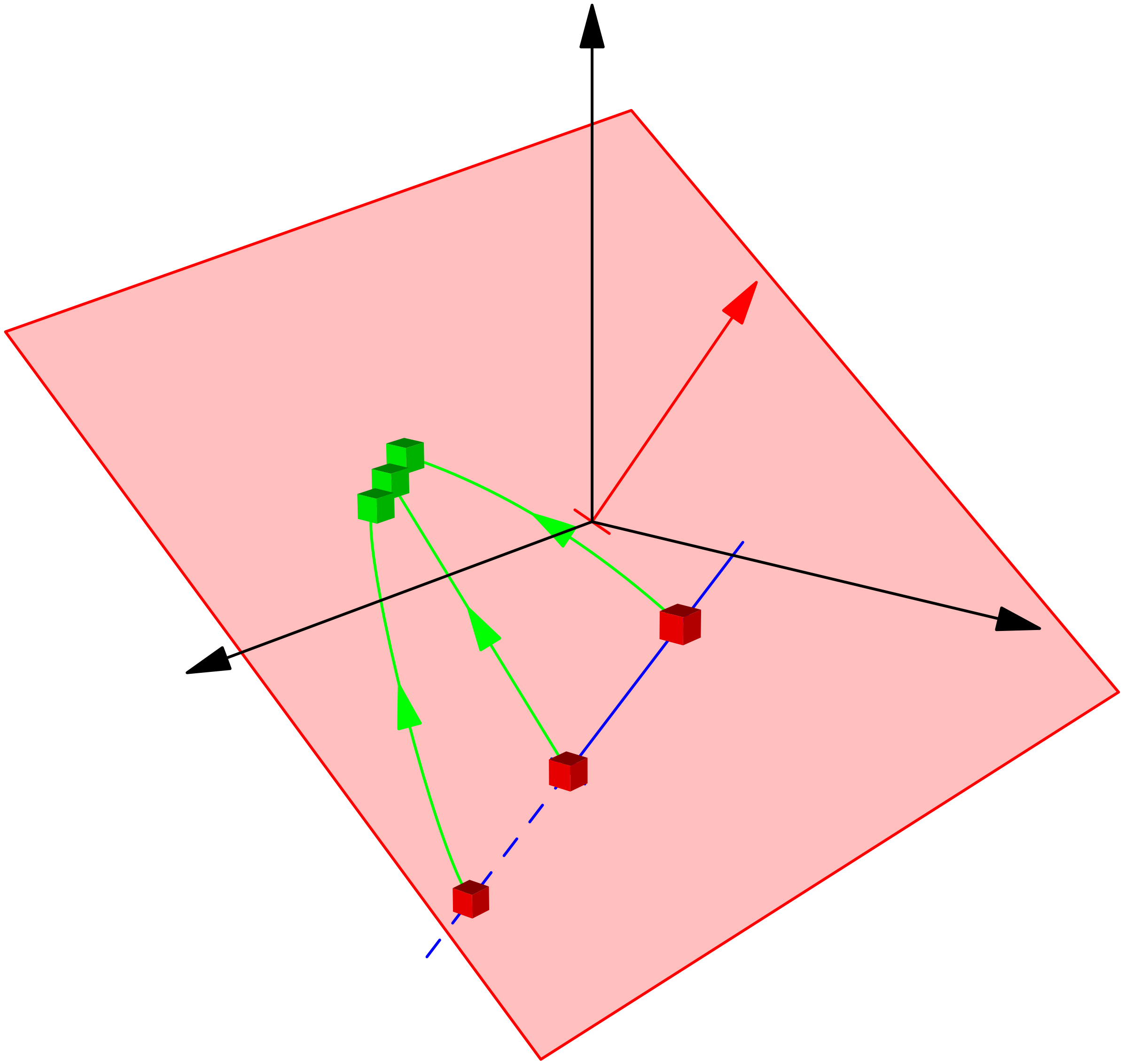
Ill-Conditions

- Singular matrices are rare (although they occur when we don't have enough data), but matrices that are close to being singular are common
- If a matrix is close to singular it is ill-conditioned
- Ill-conditioned matrices have some small eigenvalues
- All points get contracted towards a plane
- Large matrices are very often ill conditioned

Ill-Conditions

- Singular matrices are rare (although they occur when we don't have enough data), but matrices that are close to being singular are common
- If a matrix is close to singular it is ill-conditioned
- Ill-conditioned matrices have some small eigenvalues
- All points get contracted towards a plane
- Large matrices are very often ill conditioned

Ill-Conditioned Matrices



Ill-Conditioning in ML

- Ill-conditioning in machine learning occurs when a very small change in the learning data causes a large change in the predictions of the learning machine
- In linear regression the matrix $\mathbf{X}^\mathsf{T}\mathbf{X}$ is ill-conditioned when we have as many data points as parameters
- Much of machine learning is concerned with making learning machines better conditioned
- Adding regularisers is one approach to achieve this

Ill-Conditioning in ML

- Ill-conditioning in machine learning occurs when a very small change in the learning data causes a large change in the predictions of the learning machine
- In linear regression the matrix $\mathbf{X}^\mathsf{T}\mathbf{X}$ is ill-conditioned when we have as many data points as parameters
- Much of machine learning is concerned with making learning machines better conditioned
- Adding regularisers is one approach to achieve this

Ill-Conditioning in ML

- Ill-conditioning in machine learning occurs when a very small change in the learning data causes a large change in the predictions of the learning machine
- In linear regression the matrix $\mathbf{X}^\top\mathbf{X}$ is ill-conditioned when we have as many data points as parameters
- Much of machine learning is concerned with making learning machines better conditioned
- Adding regularisers is one approach to achieve this

Ill-Conditioning in ML

- Ill-conditioning in machine learning occurs when a very small change in the learning data causes a large change in the predictions of the learning machine

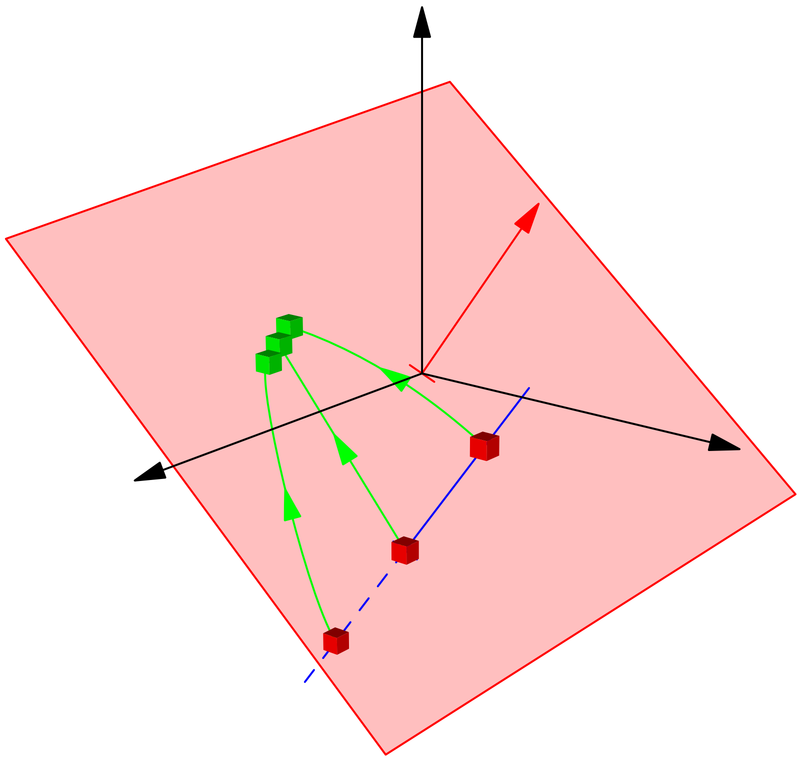
- In linear regression the matrix $\mathbf{X}^\mathsf{T}\mathbf{X}$ is ill-conditioned when we have as many data points as parameters

- Much of machine learning is concerned with making learning machines better conditioned

- Adding regularisers is one approach to achieve this

Outline

1. Mappings
2. Linear Maps
3. **Eigenvectors**



Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

- $\mathbf{M}$ is square (i.e. $n \times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

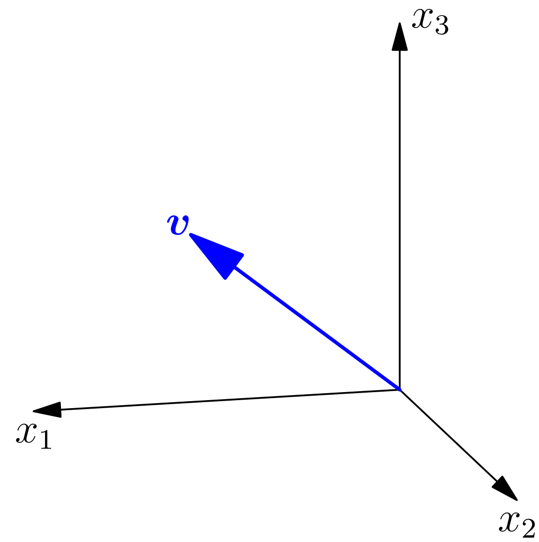
$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

- $\mathbf{M}$ is square (i.e. $n{\times}n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

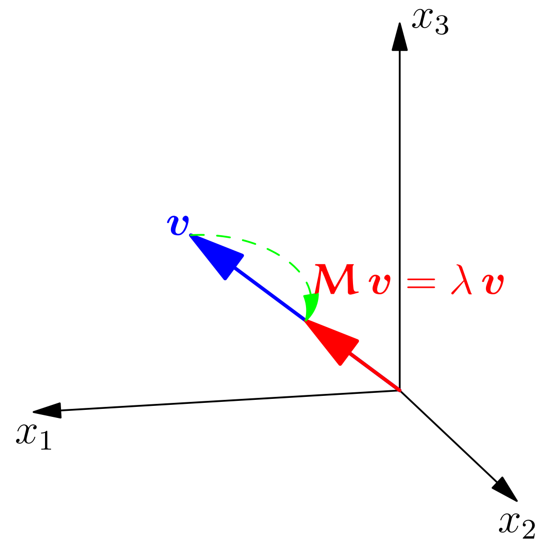


- $\mathbf{M}$ is square (i.e. $n \times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

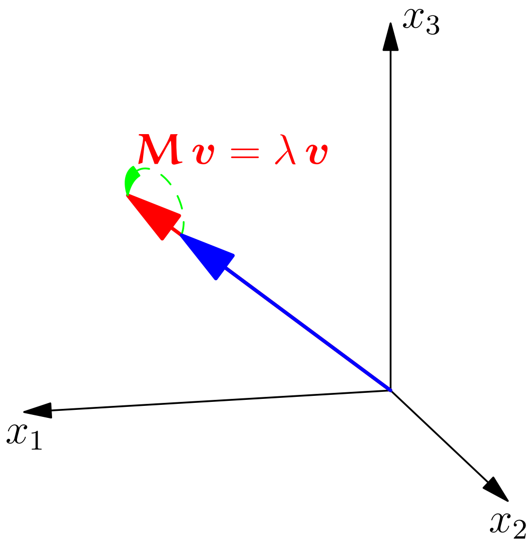


- $\mathbf{M}$ is square (i.e. $n \times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

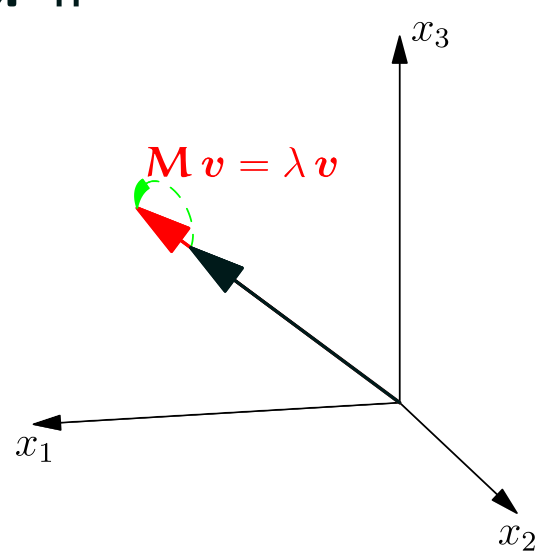


- $\mathbf{M}$ is square (i.e. $n \times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

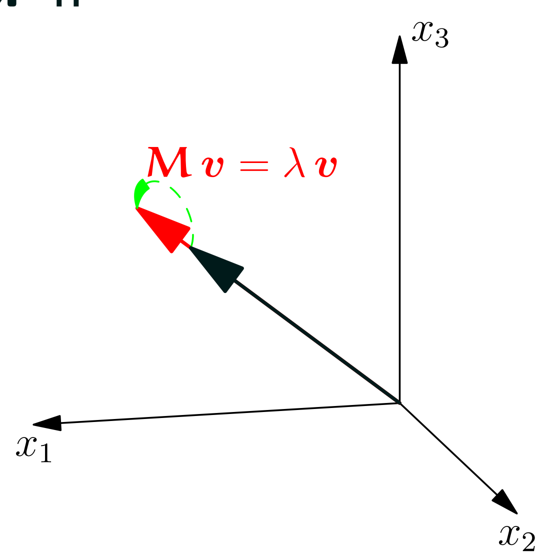


- $\mathbf{M}$ is square (i.e. $n\times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$

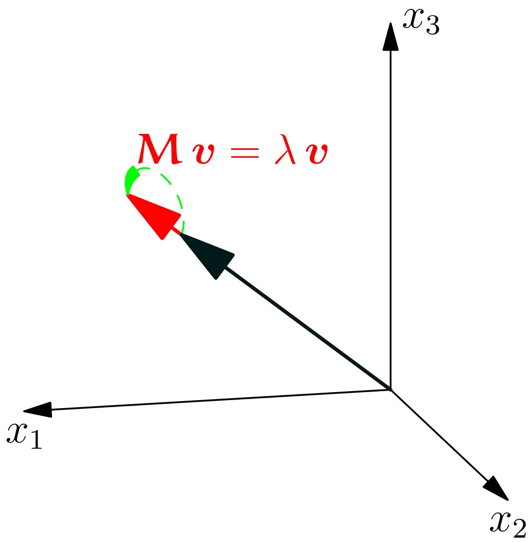


- $\mathbf{M}$ is square (i.e. $n \times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Eigenvalue equation

- Eigen-systems help us to understand mappings
- A vector $\boldsymbol{v}$ is said to be an **eigenvector** if

$$\mathbf{M}\boldsymbol{v} = \lambda\,\boldsymbol{v}$$



- $\mathbf{M}$ is square (i.e. $n \times n$)
- Where the number λ is the **eigenvalue**
- Eigenvalues play a fundamental role in understanding operators

Symmetric Matrices

- If $\boldsymbol{M}$ is an $n \times n$ **symmetric** matrix then it has n real orthogonal eigenvectors with real eigenvalues

- We denote the i^{th} eigenvector by $\boldsymbol{v}_i$ and the corresponding eigenvalue by λ_i so that

$$\boldsymbol{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

- Orthogonal means that if $i \neq j$ then

$$\boldsymbol{v}_i^{\mathsf{T}} \boldsymbol{v}_j = 0$$

- (We can always normalise eigenvectors if we want)

Symmetric Matrices

- If $\mathbf{M}$ is an $n \times n$ **symmetric** matrix then it has n real orthogonal eigenvectors with real eigenvalues

- We denote the i^{th} eigenvector by $\boldsymbol{v}_i$ and the corresponding eigenvalue by λ_i so that

$$\mathbf{M}\boldsymbol{v}_i = \lambda_i \, \boldsymbol{v}_i$$

- Orthogonal means that if $i \neq j$ then

$$\boldsymbol{v}_i^{\mathsf{T}} \boldsymbol{v}_j = 0$$

- (We can always normalise eigenvectors if we want)

Symmetric Matrices

- If $\mathbf{M}$ is an $n \times n$ **symmetric** matrix then it has n real orthogonal eigenvectors with real eigenvalues

- We denote the i^{th} eigenvector by $\boldsymbol{v}_i$ and the corresponding eigenvalue by λ_i so that

$$\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

- Orthogonal means that if $i \neq j$ then

$$\boldsymbol{v}_i^{\top} \boldsymbol{v}_j = 0$$

- (We can always normalise eigenvectors if we want)

Symmetric Matrices

- If $\mathbf{M}$ is an $n \times n$ **symmetric** matrix then it has n real orthogonal eigenvectors with real eigenvalues
- We denote the i^{th} eigenvector by $\boldsymbol{v}_i$ and the corresponding eigenvalue by λ_i so that

$$\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

- Orthogonal means that if $i \neq j$ then

$$\boldsymbol{v}_i^{\mathsf{T}} \boldsymbol{v}_j = 0$$

- (We can always normalise eigenvectors if we want)

Proof of Orthogonality

- $(\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i)^{\mathsf{T}}$ implies $\boldsymbol{v}_i^{\mathsf{T}} \mathbf{M}^{\mathsf{T}} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- When $\mathbf{M}$ is symmetric then $\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i \Rightarrow \boldsymbol{v}_i^{\mathsf{T}} \mathbf{M} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- Consider two eigenvectors $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ of $\mathbf{M}$

$$\begin{aligned} \boldsymbol{v}_i^{\mathsf{T}} \mathbf{M} \boldsymbol{v}_j &= (\boldsymbol{v}_i^{\mathsf{T}} \mathbf{M}) \boldsymbol{v}_j = \lambda_i \boldsymbol{v}_i^{\mathsf{T}} \boldsymbol{v}_j \\ &= \boldsymbol{v}_i^{\mathsf{T}} (\mathbf{M} \boldsymbol{v}_j) = \lambda_j \boldsymbol{v}_i^{\mathsf{T}} \boldsymbol{v}_j \end{aligned}$$

- So either $\lambda_i = \lambda_j$ or $\boldsymbol{v}_i^{\mathsf{T}} \boldsymbol{v}_j = 0$
- If $\lambda_i = \lambda_j$ then any vector in the plane spanned by $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are eigenvectors and we can always choose two orthogonal vectors in the plane

Proof of Orthogonality

- $\left(\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i\right)^\mathsf{T}$ implies $\boldsymbol{v}_i^\mathsf{T}\mathbf{M}^\mathsf{T} = \lambda_i \boldsymbol{v}_i^\mathsf{T}$
- When $\mathbf{M}$ is symmetric then $\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i \Rightarrow \boldsymbol{v}_i^\mathsf{T}\mathbf{M} = \lambda_i \boldsymbol{v}_i^\mathsf{T}$
- Consider two eigenvectors $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ of $\mathbf{M}$

$$\begin{aligned} \boldsymbol{v}_i^\mathsf{T}\mathbf{M}\boldsymbol{v}_j &= (\boldsymbol{v}_i^\mathsf{T}\mathbf{M})\boldsymbol{v}_j = \lambda_i \boldsymbol{v}_i^\mathsf{T}\boldsymbol{v}_j \\ &= \boldsymbol{v}_i^\mathsf{T}(\mathbf{M}\boldsymbol{v}_j) = \lambda_j \boldsymbol{v}_i^\mathsf{T}\boldsymbol{v}_j \end{aligned}$$

- So either $\lambda_i = \lambda_j$ or $\boldsymbol{v}_i^\mathsf{T}\boldsymbol{v}_j = 0$
- If $\lambda_i = \lambda_j$ then any vector in the plane spanned by $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are eigenvectors and we can always choose two orthogonal vectors in the plane

Proof of Orthogonality

- $\left(\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i\right)^{\mathsf{T}}$ implies $\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M}^{\mathsf{T}} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- When $\mathbf{M}$ is symmetric then $\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i \Rightarrow \boldsymbol{v}_i^{\mathsf{T}}\mathbf{M} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- Consider two eigenvectors $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ of $\mathbf{M}$

$$\begin{aligned} \boldsymbol{v}_i^{\mathsf{T}}\mathbf{M}\boldsymbol{v}_j &= (\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M})\boldsymbol{v}_j = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j \\ &= \boldsymbol{v}_i^{\mathsf{T}}(\mathbf{M}\boldsymbol{v}_j) = \lambda_j \boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j \end{aligned}$$

- So either $\lambda_i = \lambda_j$ or $\boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j = 0$
- If $\lambda_i = \lambda_j$ then any vector in the plane spanned by $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are eigenvectors and we can always choose two orthogonal vectors in the plane

Proof of Orthogonality

- $\left(\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i\right)^{\mathsf{T}}$ implies $\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M}^{\mathsf{T}} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- When $\mathbf{M}$ is symmetric then $\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i \Rightarrow \boldsymbol{v}_i^{\mathsf{T}}\mathbf{M} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- Consider two eigenvectors $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ of $\mathbf{M}$

$$\begin{aligned}\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M}\boldsymbol{v}_j &= (\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M})\boldsymbol{v}_j = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j \\ &= \boldsymbol{v}_i^{\mathsf{T}}(\mathbf{M}\boldsymbol{v}_j) = \lambda_j \boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j\end{aligned}$$

- So either $\lambda_i = \lambda_j$ or $\boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j = 0$
- If $\lambda_i = \lambda_j$ then any vector in the plane spanned by $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are eigenvectors and we can always choose two orthogonal vectors in the plane

Proof of Orthogonality

- $\left(\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i\right)^{\mathsf{T}}$ implies $\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M}^{\mathsf{T}} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- When $\mathbf{M}$ is symmetric then $\mathbf{M}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i \Rightarrow \boldsymbol{v}_i^{\mathsf{T}}\mathbf{M} = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}$
- Consider two eigenvectors $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ of $\mathbf{M}$

$$\begin{aligned} \boldsymbol{v}_i^{\mathsf{T}}\mathbf{M}\boldsymbol{v}_j &= (\boldsymbol{v}_i^{\mathsf{T}}\mathbf{M})\boldsymbol{v}_j = \lambda_i \boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j \\ &= \boldsymbol{v}_i^{\mathsf{T}}(\mathbf{M}\boldsymbol{v}_j) = \lambda_j \boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j \end{aligned}$$

- So either $\lambda_i = \lambda_j$ or $\boldsymbol{v}_i^{\mathsf{T}}\boldsymbol{v}_j = 0$
- If $\lambda_i = \lambda_j$ then any vector in the plane spanned by $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are eigenvectors and we can always choose two orthogonal vectors in the plane

Orthogonal Matrices

- We can construct an **orthogonal** matrix $\mathbf{V}$ from the eigenvectors

$$\mathbf{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n)$$

- Matrix $\mathbf{V}$ is an $n \times n$ matrix

- Because of the orthogonality of the vectors $\boldsymbol{v}_i$

$$\mathbf{V}^{\mathsf{T}}\mathbf{V} = \begin{pmatrix} \boldsymbol{v}_1^{\mathsf{T}}\boldsymbol{v}_1 & \boldsymbol{v}_1^{\mathsf{T}}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_1^{\mathsf{T}}\boldsymbol{v}_n \\ \boldsymbol{v}_2^{\mathsf{T}}\boldsymbol{v}_1 & \boldsymbol{v}_2^{\mathsf{T}}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_2^{\mathsf{T}}\boldsymbol{v}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_n^{\mathsf{T}}\boldsymbol{v}_1 & \boldsymbol{v}_n^{\mathsf{T}}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n^{\mathsf{T}}\boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = \mathbf{I}$$

Orthogonal Matrices

- We can construct an **orthogonal** matrix $\mathbf{V}$ from the eigenvectors

$$\mathbf{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n)$$

- Matrix $\mathbf{V}$ is an $n \times n$ matrix

- Because of the orthogonality of the vectors $\boldsymbol{v}_i$

$$\mathbf{V}^\top \mathbf{V} = \begin{pmatrix} \boldsymbol{v}_1^\top \boldsymbol{v}_1 & \boldsymbol{v}_1^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_1^\top \boldsymbol{v}_n \\ \boldsymbol{v}_2^\top \boldsymbol{v}_1 & \boldsymbol{v}_2^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_2^\top \boldsymbol{v}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_n^\top \boldsymbol{v}_1 & \boldsymbol{v}_n^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n^\top \boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = \mathbf{I}$$

Orthogonal Matrices

- We can construct an **orthogonal** matrix $\mathbf{V}$ from the eigenvectors

$$\mathbf{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n)$$

- Matrix $\mathbf{V}$ is an $n \times n$ matrix

- Because of the orthogonality of the vectors $\boldsymbol{v}_i$

$$\mathbf{V}^\mathsf{T}\mathbf{V} = \begin{pmatrix} \boldsymbol{v}_1^\mathsf{T}\boldsymbol{v}_1 & \boldsymbol{v}_1^\mathsf{T}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_1^\mathsf{T}\boldsymbol{v}_n \\ \boldsymbol{v}_2^\mathsf{T}\boldsymbol{v}_1 & \boldsymbol{v}_2^\mathsf{T}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_2^\mathsf{T}\boldsymbol{v}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_n^\mathsf{T}\boldsymbol{v}_1 & \boldsymbol{v}_n^\mathsf{T}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n^\mathsf{T}\boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = \mathbf{I}$$

Orthogonal Matrices

- We can construct an **orthogonal** matrix $\mathbf{V}$ from the eigenvectors

$$\mathbf{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n)$$

- Matrix $\mathbf{V}$ is an $n \times n$ matrix

- Because of the orthogonality of the vectors $\boldsymbol{v}_i$

$$\mathbf{V}^{\mathsf{T}}\mathbf{V} = \begin{pmatrix} \boldsymbol{v}_1^{\mathsf{T}}\boldsymbol{v}_1 & \boldsymbol{v}_1^{\mathsf{T}}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_1^{\mathsf{T}}\boldsymbol{v}_n \\ \boldsymbol{v}_2^{\mathsf{T}}\boldsymbol{v}_1 & \boldsymbol{v}_2^{\mathsf{T}}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_2^{\mathsf{T}}\boldsymbol{v}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_n^{\mathsf{T}}\boldsymbol{v}_1 & \boldsymbol{v}_n^{\mathsf{T}}\boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n^{\mathsf{T}}\boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = \mathbf{I}$$

Orthogonal Matrices

- We can construct an **orthogonal** matrix $\mathbf{V}$ from the eigenvectors

$$\mathbf{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n)$$

- Matrix $\mathbf{V}$ is an $n \times n$ matrix

- Because of the orthogonality of the vectors $\boldsymbol{v}_i$

$$\mathbf{V}^\top \mathbf{V} = \begin{pmatrix} \boldsymbol{v}_1^\top \boldsymbol{v}_1 & \boldsymbol{v}_1^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_1^\top \boldsymbol{v}_n \\ \boldsymbol{v}_2^\top \boldsymbol{v}_1 & \boldsymbol{v}_2^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_2^\top \boldsymbol{v}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_n^\top \boldsymbol{v}_1 & \boldsymbol{v}_n^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n^\top \boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = \mathbf{I}$$

Orthogonal Matrices

- We can construct an **orthogonal** matrix $\mathbf{V}$ from the eigenvectors

$$\mathbf{V} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n)$$

- Matrix $\mathbf{V}$ is an $n \times n$ matrix

- Because of the orthogonality of the vectors $\boldsymbol{v}_i$

$$\mathbf{V}^\top \mathbf{V} = \begin{pmatrix} \boldsymbol{v}_1^\top \boldsymbol{v}_1 & \boldsymbol{v}_1^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_1^\top \boldsymbol{v}_n \\ \boldsymbol{v}_2^\top \boldsymbol{v}_1 & \boldsymbol{v}_2^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_2^\top \boldsymbol{v}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_n^\top \boldsymbol{v}_1 & \boldsymbol{v}_n^\top \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n^\top \boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = \mathbf{I}$$

The Other Way Around

- We have shown that $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$
- Thus multiply both sides on the left by $\mathbf{V}$

$$\mathbf{V}\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}$$

- $\mathbf{V}$ will have an inverse, $\mathbf{V}^{-1}$, such that $\mathbf{V}\mathbf{V}^{-1} = \mathbf{I}$
- Multiplying the equation on the right by $\mathbf{V}^{-1}$

$$(\mathbf{V}\mathbf{V}^{\mathsf{T}})\mathbf{V}\mathbf{V}^{-1} = \mathbf{V}\mathbf{V}^{-1}$$

- Note that, $\mathbf{V}^{-1} = \mathbf{V}^{\mathsf{T}}$ (definition of orthogonal matrix)

The Other Way Around

- We have shown that $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$
- Thus multiply both sides on the left by $\mathbf{V}$

$$\mathbf{V}\,\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}$$

- $\mathbf{V}$ will have an inverse, $\mathbf{V}^{-1}$, such that $\mathbf{V}\,\mathbf{V}^{-1} = \mathbf{I}$
- Multiplying the equation on the right by $\mathbf{V}^{-1}$

$$(\mathbf{V}\,\mathbf{V}^{\mathsf{T}})\,\mathbf{V}\,\mathbf{V}^{-1} = \mathbf{V}\,\mathbf{V}^{-1}$$

- Note that, $\mathbf{V}^{-1} = \mathbf{V}^{\mathsf{T}}$ (definition of orthogonal matrix)

The Other Way Around

- We have shown that $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$
- Thus multiply both sides on the left by $\mathbf{V}$

$$\mathbf{V}\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}$$

- $\mathbf{V}$ will have an inverse, $\mathbf{V}^{-1}$, such that $\mathbf{V}\mathbf{V}^{-1} = \mathbf{I}$
- Multiplying the equation on the right by $\mathbf{V}^{-1}$

$$(\mathbf{V}\mathbf{V}^{\mathsf{T}})\mathbf{V}\mathbf{V}^{-1} = \mathbf{V}\mathbf{V}^{-1}$$

- Note that, $\mathbf{V}^{-1} = \mathbf{V}^{\mathsf{T}}$ (definition of orthogonal matrix)

The Other Way Around

- We have shown that $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$
- Thus multiply both sides on the left by $\mathbf{V}$

$$\mathbf{V}\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}$$

- $\mathbf{V}$ will have an inverse, $\mathbf{V}^{-1}$, such that $\mathbf{V}\mathbf{V}^{-1} = \mathbf{I}$
- Multiplying the equation on the right by $\mathbf{V}^{-1}$

$$(\mathbf{V}\mathbf{V}^{\mathsf{T}})\mathbf{V}\mathbf{V}^{-1} = \mathbf{V}\mathbf{V}^{-1}$$

- Note that, $\mathbf{V}^{-1} = \mathbf{V}^{\mathsf{T}}$ (definition of orthogonal matrix)

The Other Way Around

- We have shown that $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{I}$
- Thus multiply both sides on the left by $\mathbf{V}$

$$\mathbf{V}\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}$$

- $\mathbf{V}$ will have an inverse, $\mathbf{V}^{-1}$, such that $\mathbf{V}\mathbf{V}^{-1} = \mathbf{I}$
- Multiplying the equation on the right by $\mathbf{V}^{-1}$

$$(\mathbf{V}\mathbf{V}^\mathsf{T})\mathbf{V}\mathbf{V}^{-1} = \mathbf{V}\mathbf{V}^{-1}$$

$$\mathbf{V}\mathbf{V}^\mathsf{T} = \mathbf{I}$$

- Note that, $\mathbf{V}^{-1} = \mathbf{V}^\mathsf{T}$ (definition of orthogonal matrix)

The Other Way Around

- We have shown that $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{I}$
- Thus multiply both sides on the left by $\mathbf{V}$

$$\mathbf{V}\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}$$

- $\mathbf{V}$ will have an inverse, $\mathbf{V}^{-1}$, such that $\mathbf{V}\mathbf{V}^{-1} = \mathbf{I}$
- Multiplying the equation on the right by $\mathbf{V}^{-1}$

$$(\mathbf{V}\mathbf{V}^\mathsf{T})\mathbf{V}\mathbf{V}^{-1} = \mathbf{V}\mathbf{V}^{-1}$$
$$\mathbf{V}\mathbf{V}^\mathsf{T} = \mathbf{I}$$

- Note that, $\mathbf{V}^{-1} = \mathbf{V}^\mathsf{T}$ (definition of orthogonal matrix)

Invertible Matrices

- A matrix, $\mathbf{M}$, will be singular (uninvertible) if there exists a vector $\boldsymbol{x}$ $(\neq \mathbf{0})$ such that

$$\mathbf{M}\,\boldsymbol{x} = \mathbf{0}$$

- Now if there exists such a vector such that $\mathbf{V}\,\boldsymbol{x} = \mathbf{0}$ then multiply by $\mathbf{V}^\mathsf{T}$ we get

$$\mathbf{V}^\mathsf{T}\mathbf{V}\,\boldsymbol{x} = \mathbf{V}^\mathsf{T}\,\mathbf{0}$$

Invertible Matrices

- A matrix, $\mathbf{M}$, will be singular (uninvertible) if there exists a vector $\boldsymbol{x}$ ($\neq \mathbf{0}$) such that

$$\mathbf{M}\,\boldsymbol{x} = \mathbf{0}$$

- Now if there exists such a vector such that $\mathbf{V}\,\boldsymbol{x} = \mathbf{0}$ then multiply by $\mathbf{V}^{\mathsf{T}}$ we get

$$\mathbf{V}^{\mathsf{T}}\mathbf{V}\,\boldsymbol{x} = \mathbf{V}^{\mathsf{T}}\,\mathbf{0}$$

Invertible Matrices

- A matrix, $\mathbf{M}$, will be singular (uninvertible) if there exists a vector $\boldsymbol{x}$ $(\neq \mathbf{0})$ such that

$$\mathbf{M}\,\boldsymbol{x} = \mathbf{0}$$

- Now if there exists such a vector such that $\mathbf{V}\,\boldsymbol{x} = \mathbf{0}$ then multiply by $\mathbf{V}^{\mathsf{T}}$ we get

$$\mathbf{V}^{\mathsf{T}}\mathbf{V}\,\boldsymbol{x} = \mathbf{V}^{\mathsf{T}}\mathbf{0}$$

$$\boldsymbol{x} = \mathbf{0}$$

since $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$

Invertible Matrices

- A matrix, $\mathbf{M}$, will be singular (uninvertible) if there exists a vector $\boldsymbol{x}$ ($\neq \mathbf{0}$) such that

$$\mathbf{M}\,\boldsymbol{x} = \mathbf{0}$$

- Now if there exists such a vector such that $\mathbf{V}\,\boldsymbol{x} = \mathbf{0}$ then multiply by $\mathbf{V}^{\mathsf{T}}$ we get

$$\mathbf{V}^{\mathsf{T}}\mathbf{V}\,\boldsymbol{x} = \mathbf{V}^{\mathsf{T}}\,\mathbf{0}$$

$$\boldsymbol{x} = \mathbf{0}$$

since $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$

- Thus $\boldsymbol{V}$ is invertible

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$

- As a consequent they define rotations (and possibly a reflection)

- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}\,\mathbf{V}^\mathsf{T} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^\mathsf{T}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^\mathsf{T}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^\mathsf{T}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2 \, \|\boldsymbol{y}\|_2 \cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$

- As a consequent they define rotations (and possibly a reflection)

- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}\,\mathbf{V}^\mathsf{T} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^\mathsf{T}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^\mathsf{T}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^\mathsf{T}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$

- As a consequent they define rotations (and possibly a reflection)

- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}\,\mathbf{V}^\mathsf{T} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^\mathsf{T}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^\mathsf{T}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^\mathsf{T}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$

- As a consequent they define rotations (and possibly a reflection)

- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}\,\mathbf{V}^\mathsf{T} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^\mathsf{T}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^\mathsf{T}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^\mathsf{T}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2 \cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{V}\,\mathbf{V}^\mathsf{T} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^\mathsf{T}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^\mathsf{T}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^\mathsf{T}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^\mathsf{T}\mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^\mathsf{T}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\, \|\boldsymbol{y}\|_2 \cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$

- As a consequent they define rotations (and possibly a reflection)

- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Rotations

- Orthogonal matrices satisfy $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}$
- As a consequent they define rotations (and possibly a reflection)
- Consider a vector $\boldsymbol{x}$ and $\boldsymbol{x}' = \mathbf{V}\boldsymbol{x}$, now

$$\|\boldsymbol{x}'\|_2^2 = \boldsymbol{x}'^{\mathsf{T}}\boldsymbol{x}' = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{x} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_2^2$$

- Similarly if additionally $\boldsymbol{y}' = \mathbf{V}\boldsymbol{y}$ then

$$\langle \boldsymbol{x}', \boldsymbol{y}' \rangle = (\mathbf{V}\boldsymbol{x})^{\mathsf{T}}(\mathbf{V}\boldsymbol{y}) = \boldsymbol{x}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{y} = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{y} = \langle \boldsymbol{x}, \boldsymbol{y} \rangle = \|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2\,\cos(\theta)$$

- Rotations and reflections preserve lengths and angles

Matrix Decomposition

- Taking the matrix of eigenvectors, $\boldsymbol{V}$, then

$$\mathbf{M}\,\mathbf{V} = \mathbf{M}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) = (\lambda_1\,\boldsymbol{v}_1, \lambda_2\,\boldsymbol{v}_2, \ldots, \lambda_n\,\boldsymbol{v}_n) = \mathbf{V}\,\boldsymbol{\Lambda}$$

- where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$

- Now

$$\mathbf{M} = \mathbf{M}\,\mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}}$$

- Very important *similarity transform*

Matrix Decomposition

- Taking the matrix of eigenvectors, $\boldsymbol{V}$, then

$$\mathbf{M}\,\mathbf{V} = \mathbf{M}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) = (\lambda_1\,\boldsymbol{v}_1, \lambda_2\,\boldsymbol{v}_2, \ldots, \lambda_n\,\boldsymbol{v}_n) = \mathbf{V}\,\boldsymbol{\Lambda}$$

- where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$

- Now

$$\mathbf{M} = \mathbf{M}\,\mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}}$$

- Very important *similarity transform*

Matrix Decomposition

- Taking the matrix of eigenvectors, $\boldsymbol{V}$, then

$$\mathbf{M}\,\mathbf{V} = \mathbf{M}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) = (\lambda_1\,\boldsymbol{v}_1, \lambda_2\,\boldsymbol{v}_2, \ldots, \lambda_n\,\boldsymbol{v}_n) = \mathbf{V}\,\boldsymbol{\Lambda}$$

- where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$

- Now

$$\mathbf{M} = \mathbf{M}\,\mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}}$$

- Very important *similarity transform*

Matrix Decomposition

- Taking the matrix of eigenvectors, $\boldsymbol{V}$, then

$$\mathbf{M}\,\mathbf{V} = \mathbf{M}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) = (\lambda_1\,\boldsymbol{v}_1, \lambda_2\,\boldsymbol{v}_2, \ldots, \lambda_n\,\boldsymbol{v}_n) = \mathbf{V}\,\boldsymbol{\Lambda}$$

- where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$

- Now

$$\mathbf{M} = \mathbf{M}\,\mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}}$$

- Very important *similarity transform*

Matrix Decomposition

- Taking the matrix of eigenvectors, $\boldsymbol{V}$, then

$$\mathbf{M}\,\mathbf{V} = \mathbf{M}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) = (\lambda_1\,\boldsymbol{v}_1, \lambda_2\,\boldsymbol{v}_2, \ldots, \lambda_n\,\boldsymbol{v}_n) = \mathbf{V}\,\boldsymbol{\Lambda}$$

- where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$

- Now

$$\mathbf{M} = \mathbf{M}\,\mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}}$$

- Very important *similarity transform*

Matrix Decomposition

- Taking the matrix of eigenvectors, $\boldsymbol{V}$, then

$$\mathbf{M}\,\mathbf{V} = \mathbf{M}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) = (\lambda_1\,\boldsymbol{v}_1, \lambda_2\,\boldsymbol{v}_2, \ldots, \lambda_n\,\boldsymbol{v}_n) = \mathbf{V}\,\boldsymbol{\Lambda}$$

- where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$

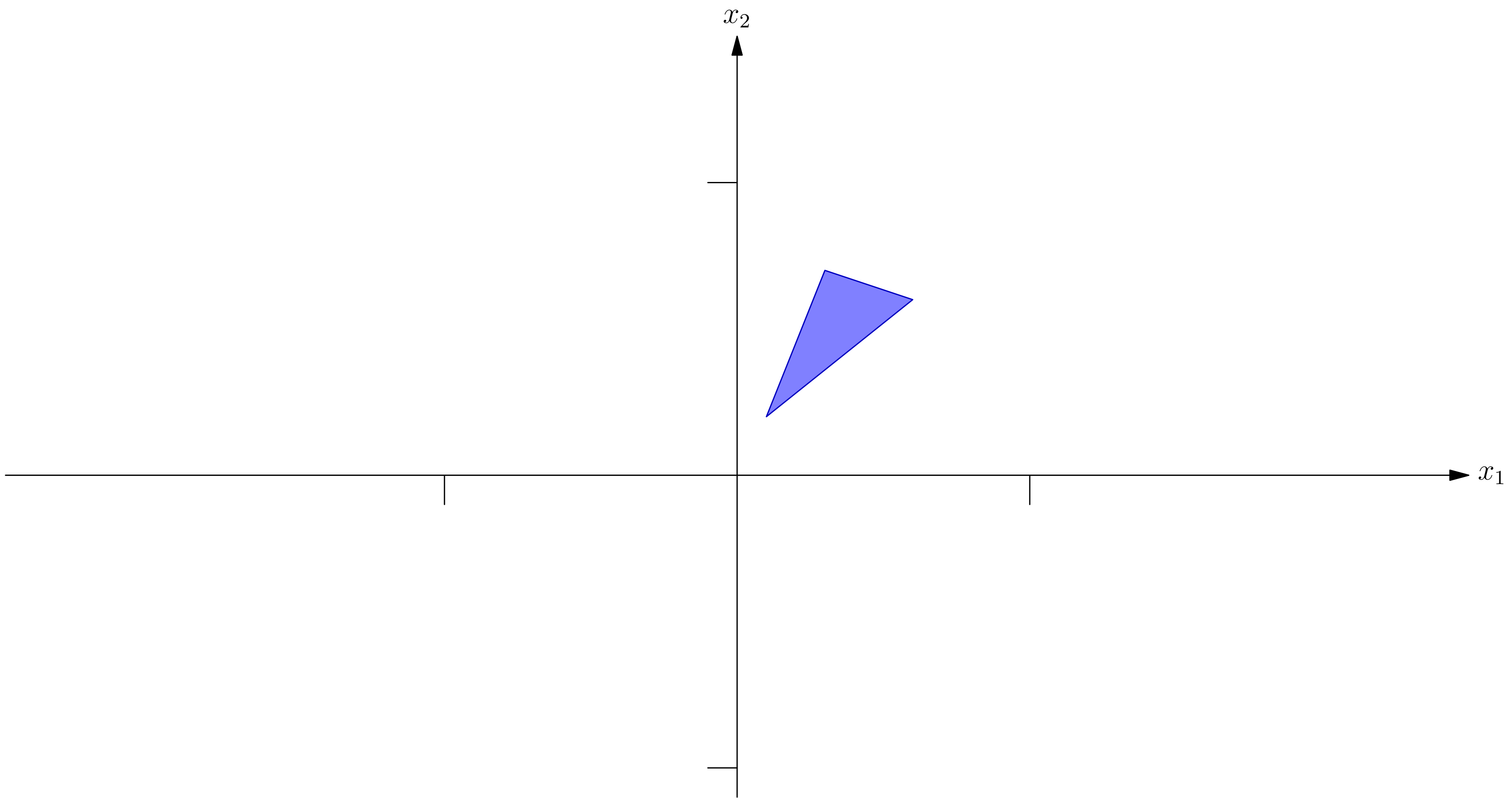
- Now

$$\mathbf{M} = \mathbf{M}\,\mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}}$$

- Very important *similarity transform*

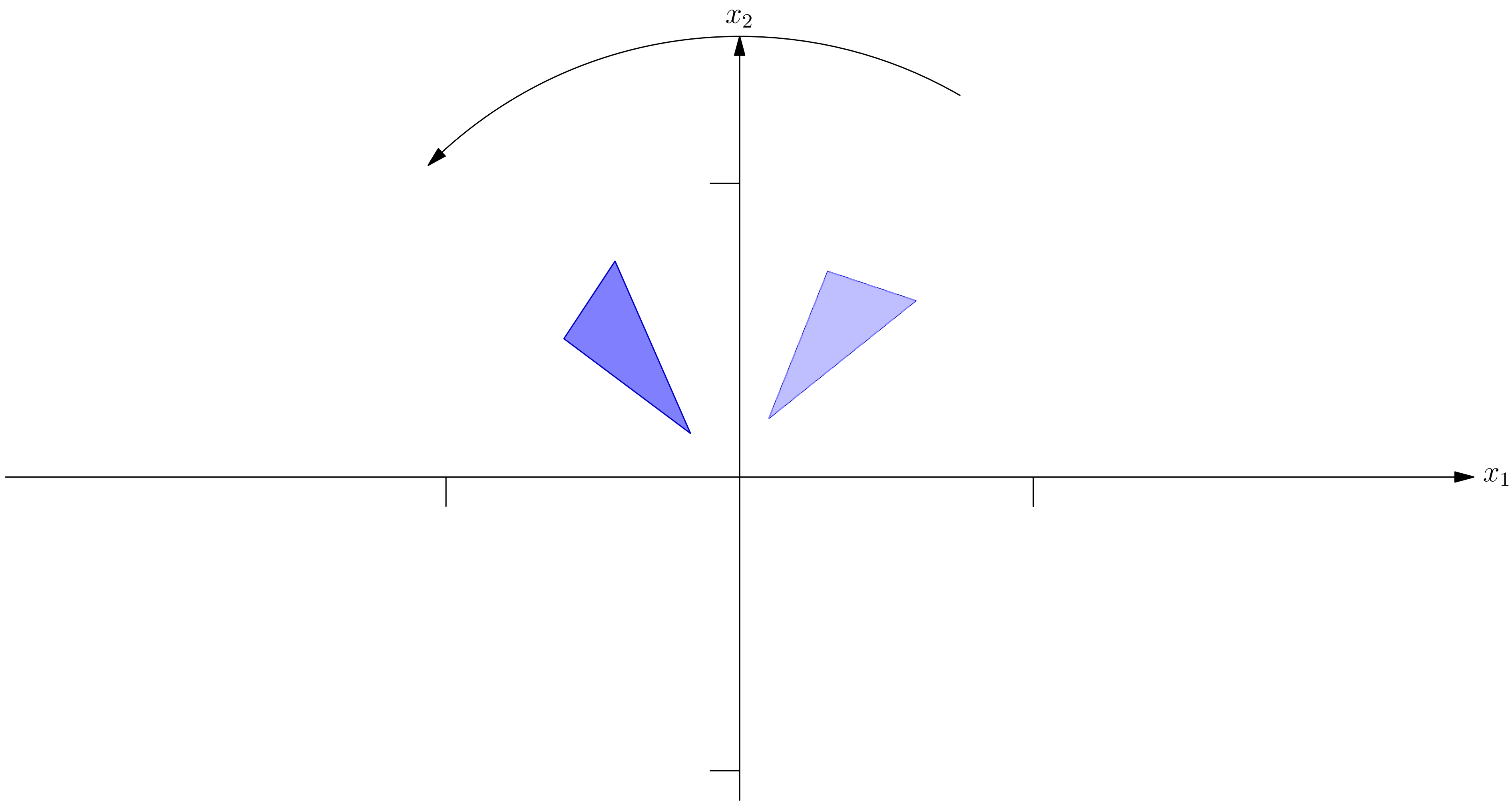
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^\mathsf{T} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



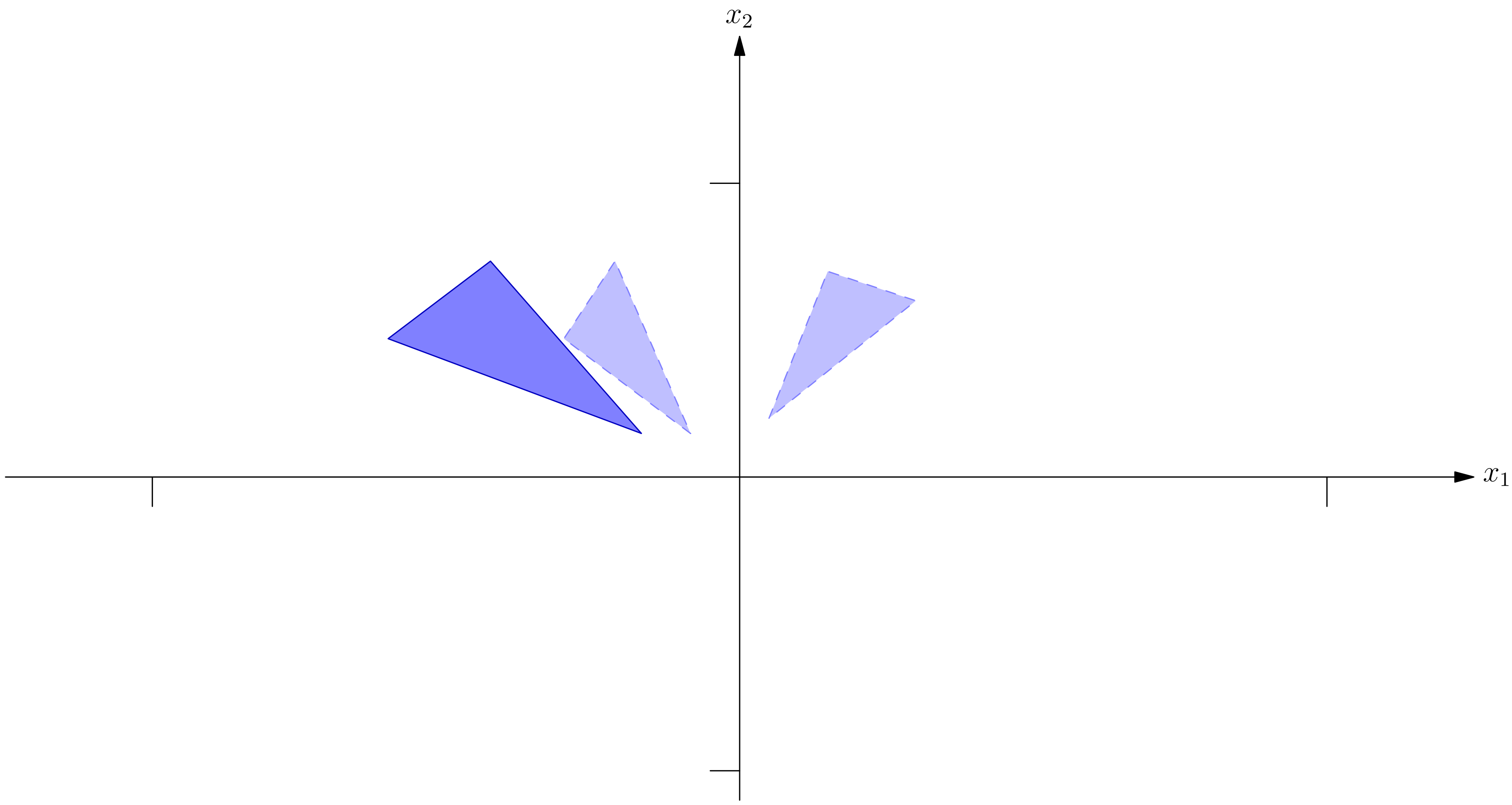
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^{\mathsf{T}} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



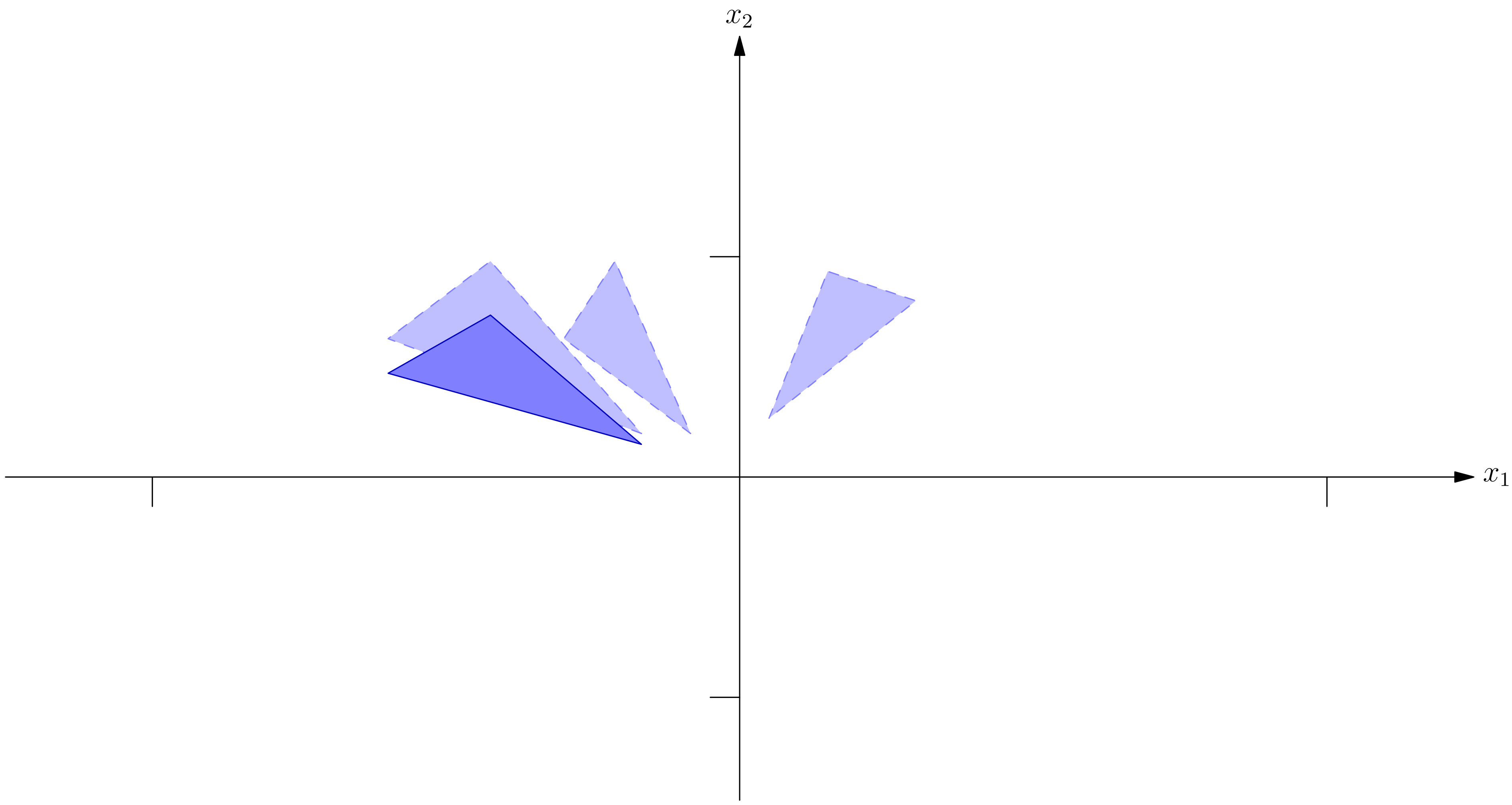
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^{\mathsf{T}} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



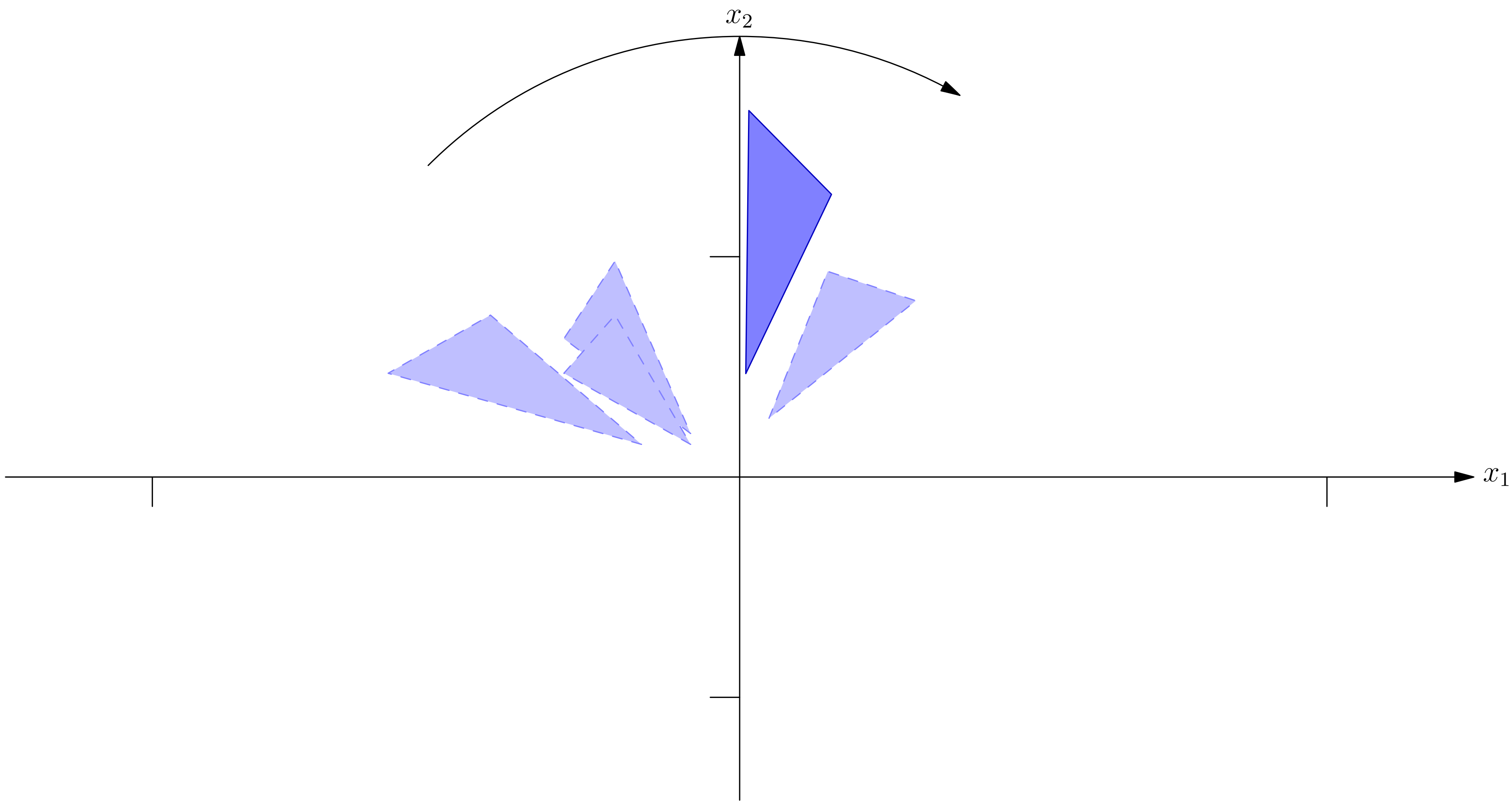
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^\mathsf{T} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



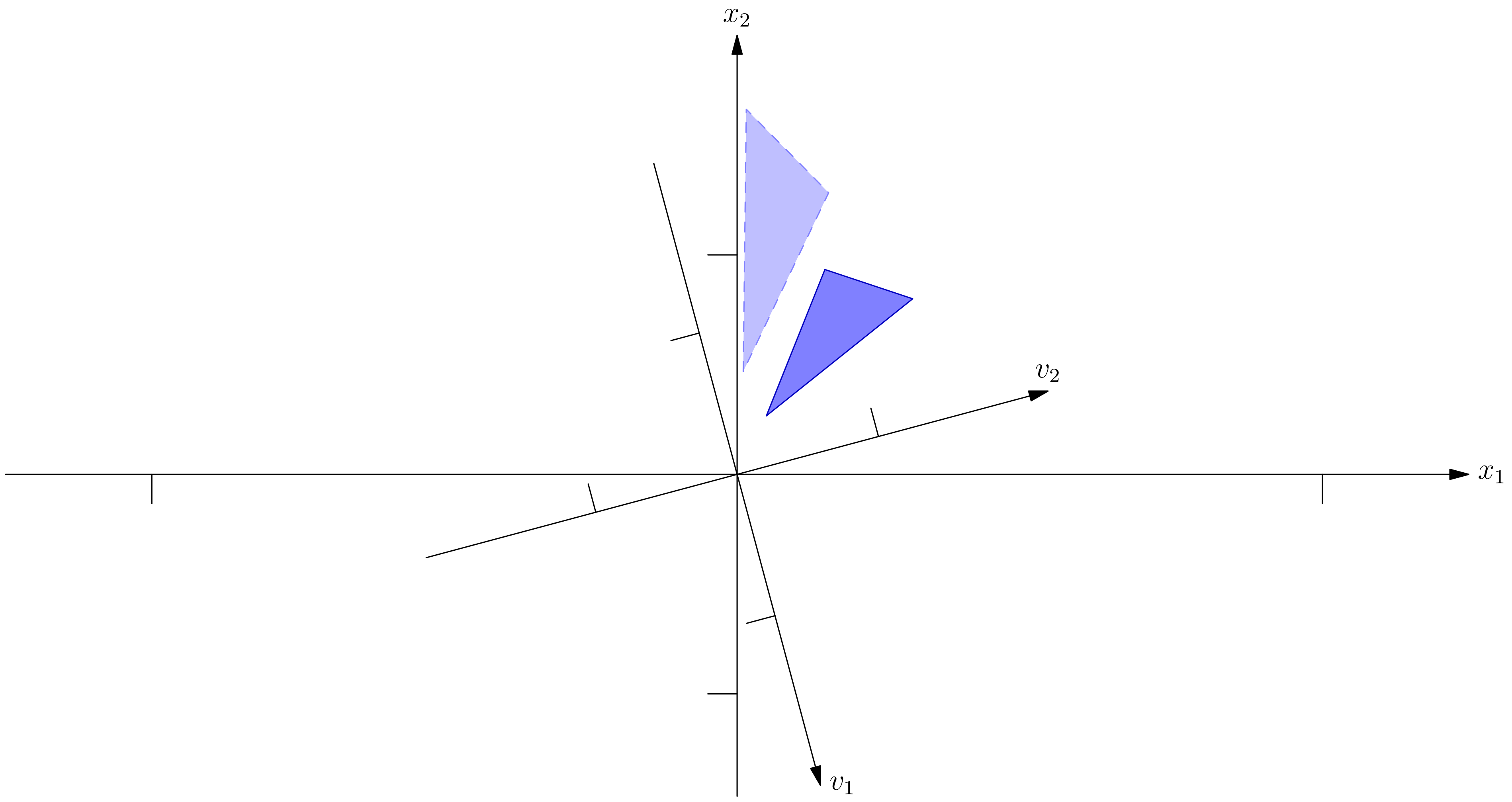
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^{\mathsf{T}} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



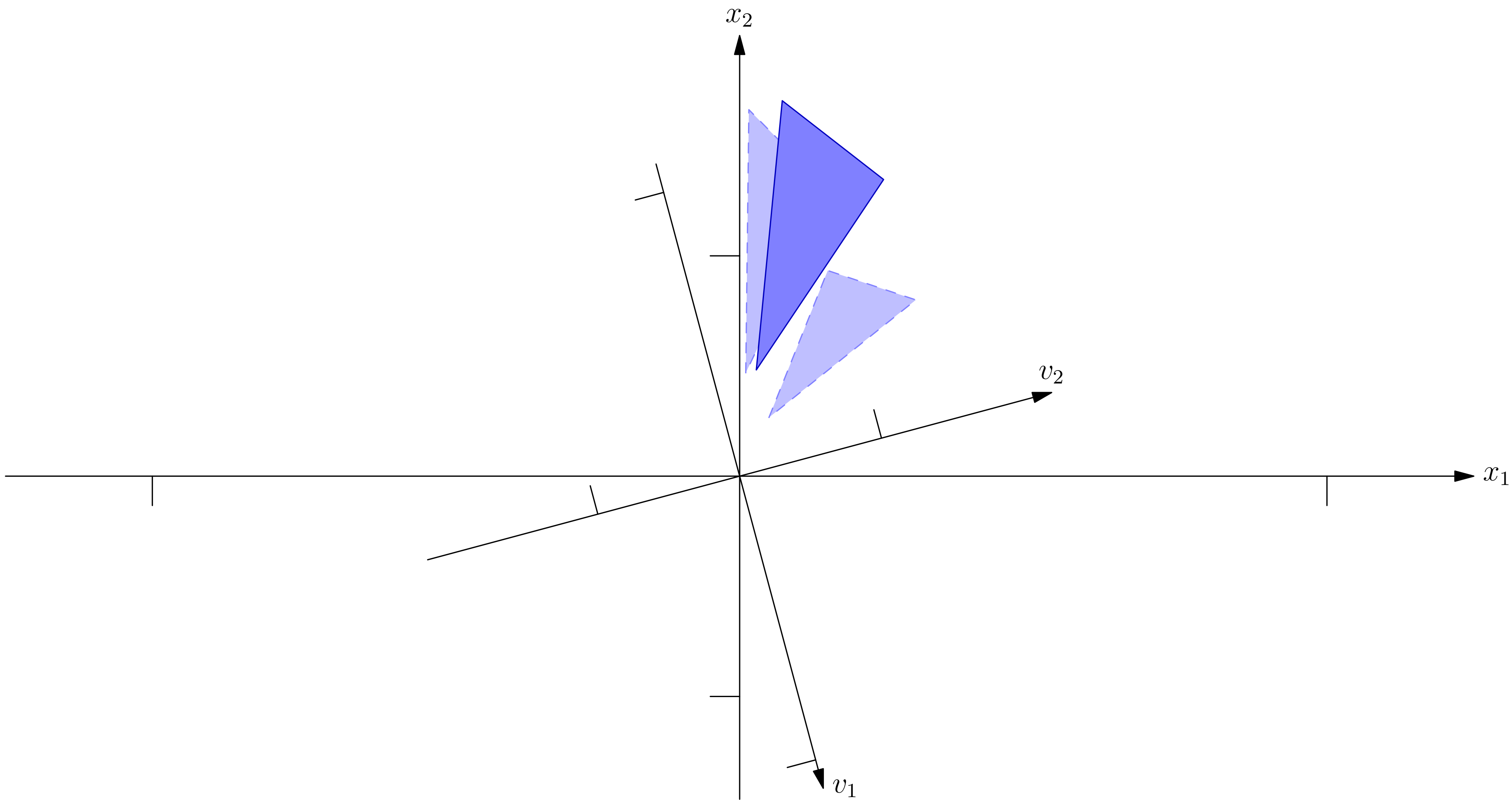
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^\mathsf{T} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



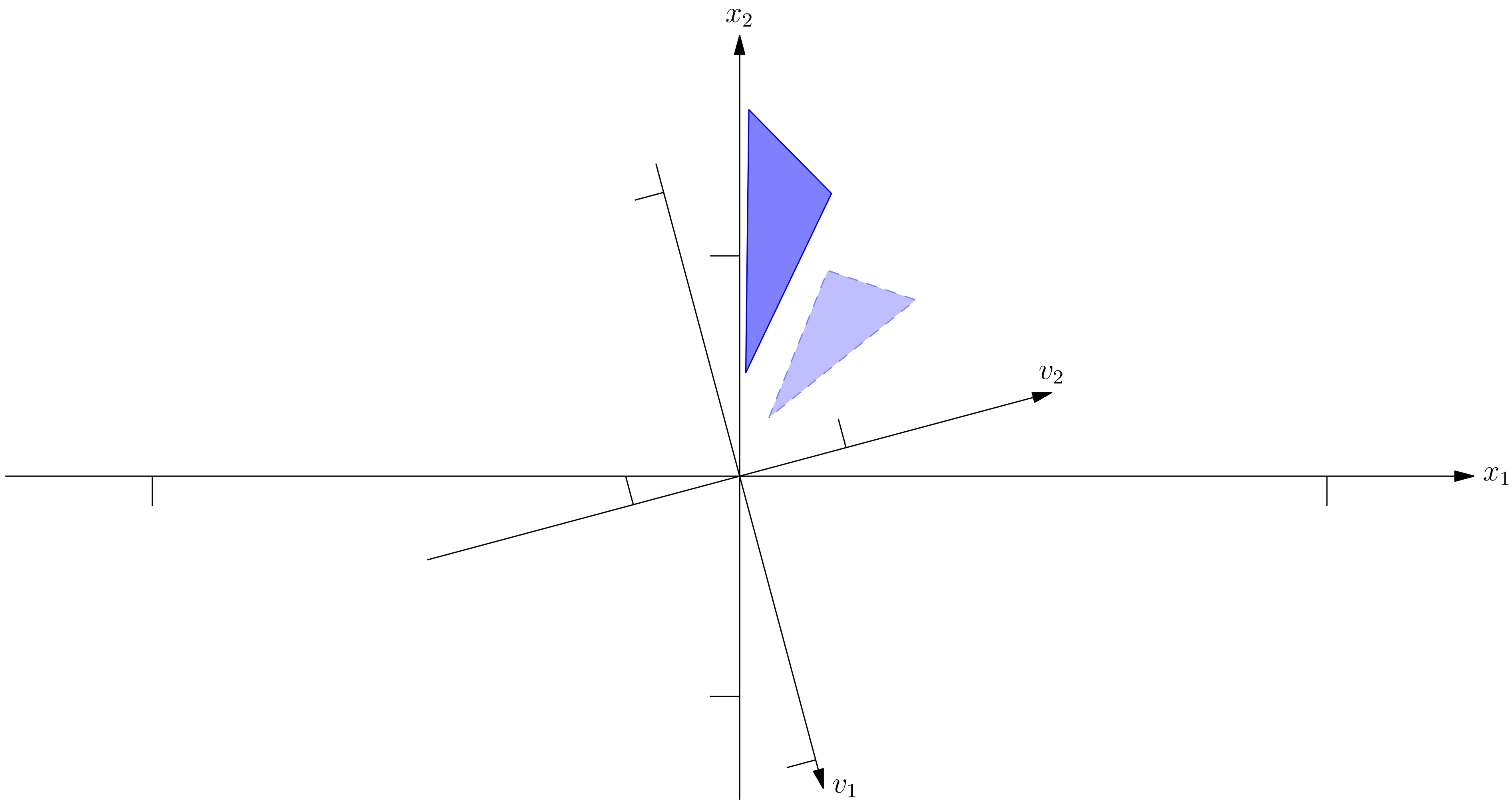
Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^{\mathsf{T}} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



Mappings by Symmetric Matrices

$$\mathbf{M} = \begin{pmatrix} 0.83 & -0.31 \\ -0.31 & 1.9 \end{pmatrix} = \mathbf{V}\mathbf{S}\mathbf{V}^\mathsf{T} = \begin{pmatrix} \cos(-75) & \sin(-75) \\ -\sin(-75) & \cos(-75) \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 0.75 \end{pmatrix} \begin{pmatrix} \cos(75) & \sin(75) \\ -\sin(75) & \cos(75) \end{pmatrix}$$



Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\mathbf{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\mathbf{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\boldsymbol{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\boldsymbol{\Lambda}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\mathbf{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\mathbf{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\mathbf{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} = (\mathbf{V}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) &= \mathbf{V}\,\mathbf{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\mathbf{\Lambda}\,\mathbf{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\boldsymbol{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\boldsymbol{\Lambda}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned}\mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\boldsymbol{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}})\\ &= \mathbf{V}\,\boldsymbol{\Lambda}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I}\end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

- Where $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\boldsymbol{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\boldsymbol{\Lambda}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

- I.e, Small eigenvalues become large eigenvalues and visa verse

Inverses

- For any square matrix

$$\mathbf{M} = \mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}} \qquad\qquad \mathbf{M}^{-1} = \mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}$$

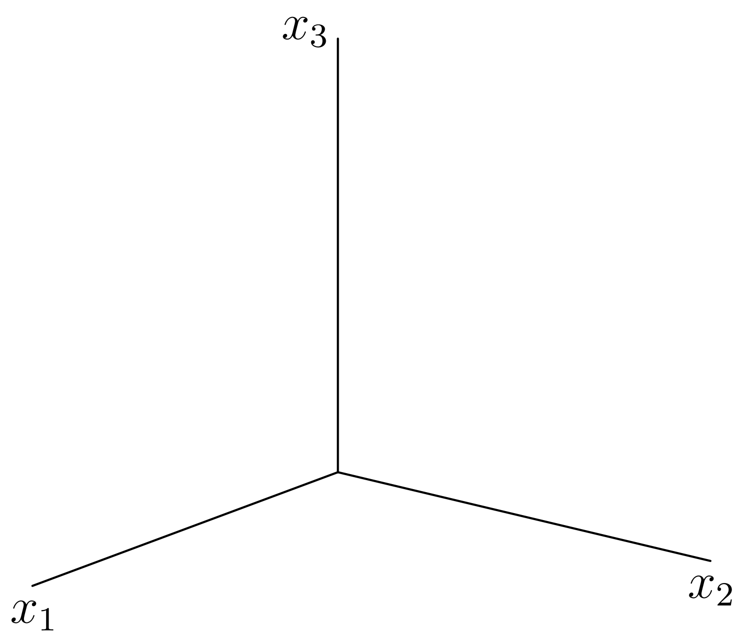
- Where $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}) = \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\lambda_n} \end{pmatrix}$

- Since

$$\begin{aligned} \mathbf{M}\,\mathbf{M}^{-1} &= (\mathbf{V}\,\boldsymbol{\Lambda}\,\mathbf{V}^{\mathsf{T}})\,(\mathbf{V}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) = \mathbf{V}\,\boldsymbol{\Lambda}\,(\mathbf{V}^{\mathsf{T}}\,\mathbf{V})\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}}) \\ &= \mathbf{V}\,\boldsymbol{\Lambda}\,\boldsymbol{\Lambda}^{-1}\,\mathbf{V}^{\mathsf{T}} = \mathbf{V}\,\mathbf{V}^{\mathsf{T}} = \mathbf{I} \end{aligned}$$

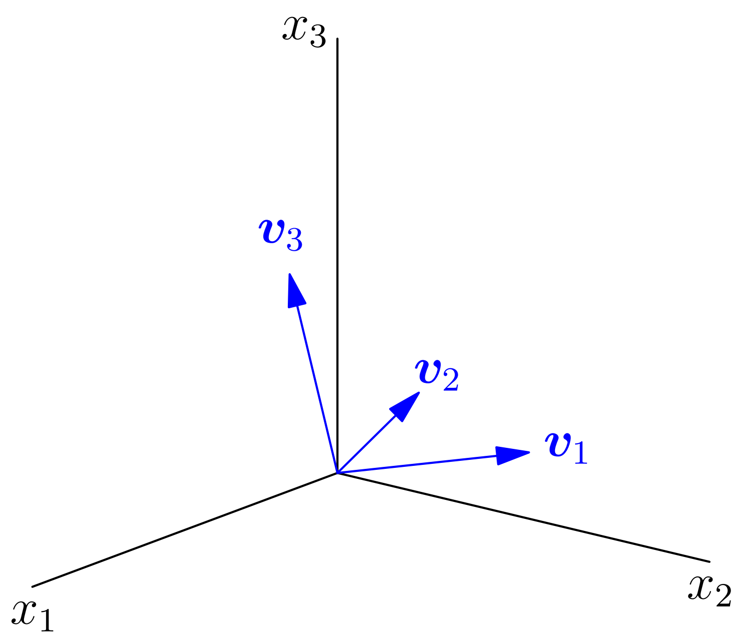
- I.e, Small eigenvalues become large eigenvalues and visa verse

Ill-Conditioning Again



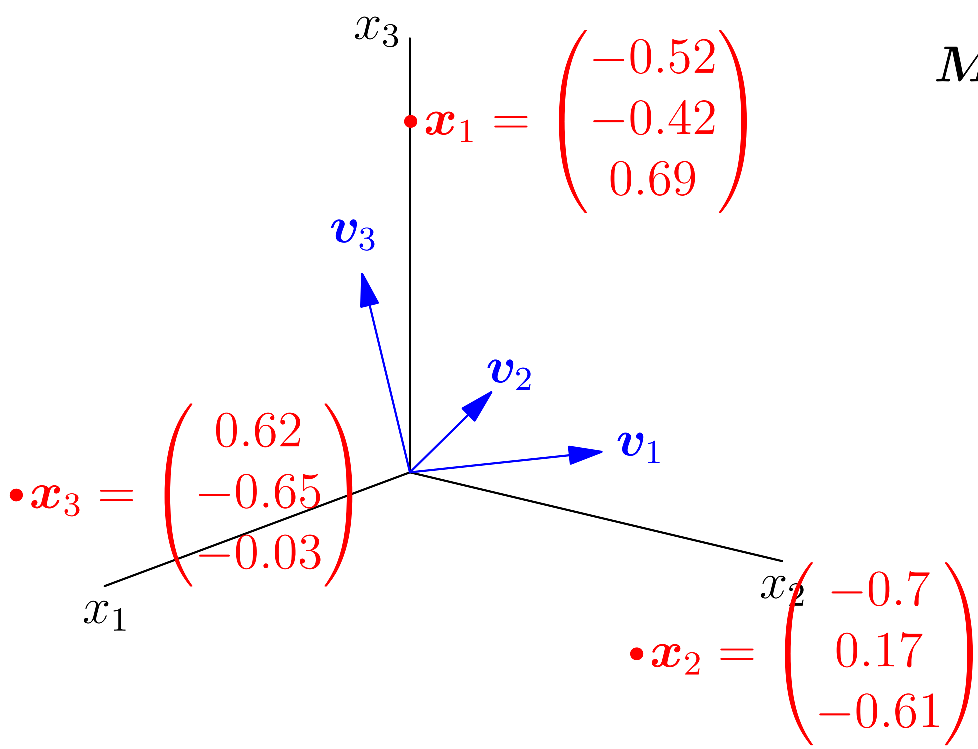
$$\boldsymbol{M} = \begin{pmatrix} 0.80 & 0.01 & 0.25 \\ 0.01 & 0.85 & 0.48 \\ 0.25 & 0.48 & 0.5 \end{pmatrix}$$
$$= \boldsymbol{V} \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \boldsymbol{V}^{\mathsf{T}}$$

Ill-Conditioning Again



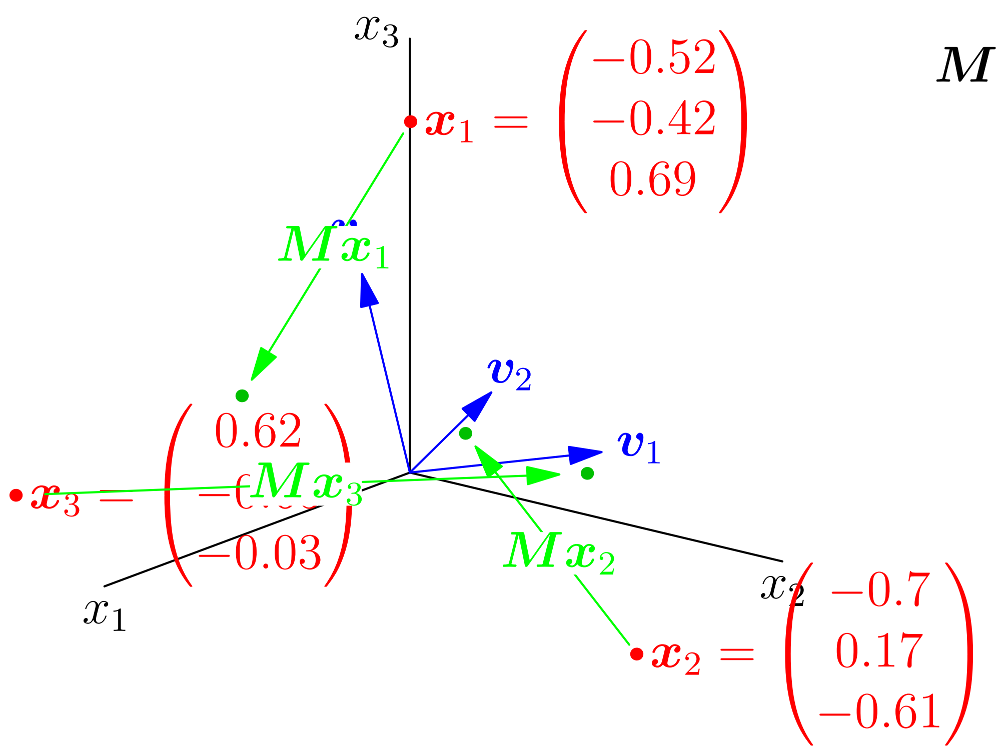
$$\boldsymbol{M} = \begin{pmatrix} 0.80 & 0.01 & 0.25 \\ 0.01 & 0.85 & 0.48 \\ 0.25 & 0.48 & 0.5 \end{pmatrix}$$
$$= \boldsymbol{V} \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \boldsymbol{V}^{\mathsf{T}}$$

Ill-Conditioning Again



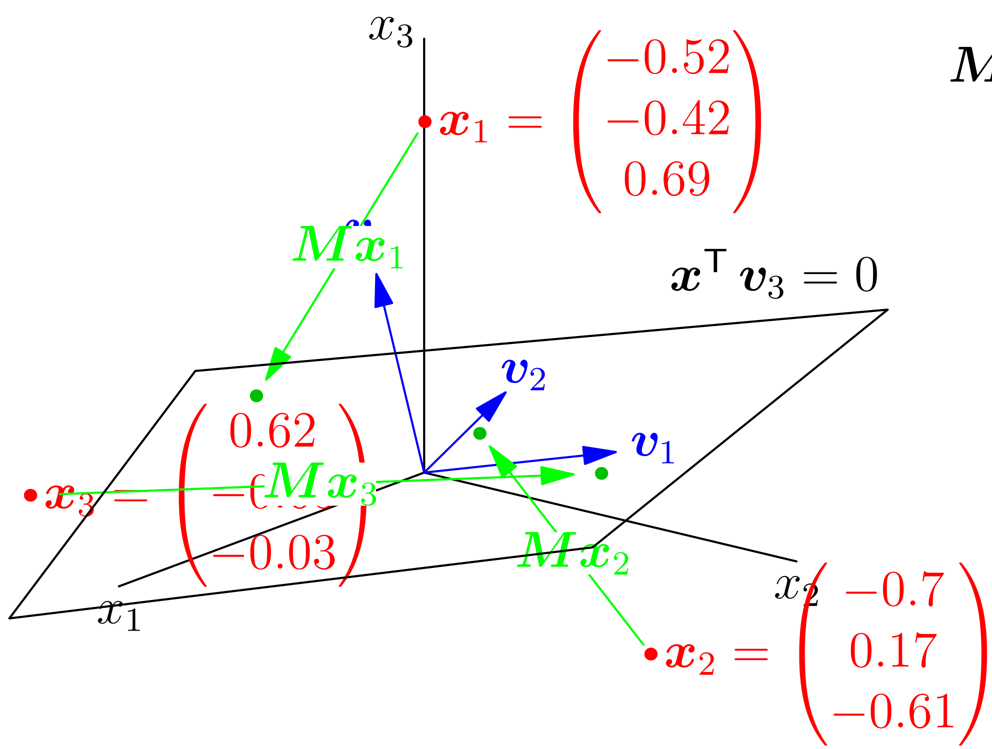
$$\begin{aligned} \boldsymbol{M} &= \begin{pmatrix} 0.80 & 0.01 & 0.25 \\ 0.01 & 0.85 & 0.48 \\ 0.25 & 0.48 & 0.5 \end{pmatrix} \\ &= \boldsymbol{V} \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \boldsymbol{V}^{\mathsf{T}} \end{aligned}$$

Ill-Conditioning Again



$$
\boldsymbol{M} = \begin{pmatrix} 0.80 & 0.01 & 0.25 \\ 0.01 & 0.85 & 0.48 \\ 0.25 & 0.48 & 0.5 \end{pmatrix}
= \boldsymbol{V} \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \boldsymbol{V}^{\mathsf{T}}
$$

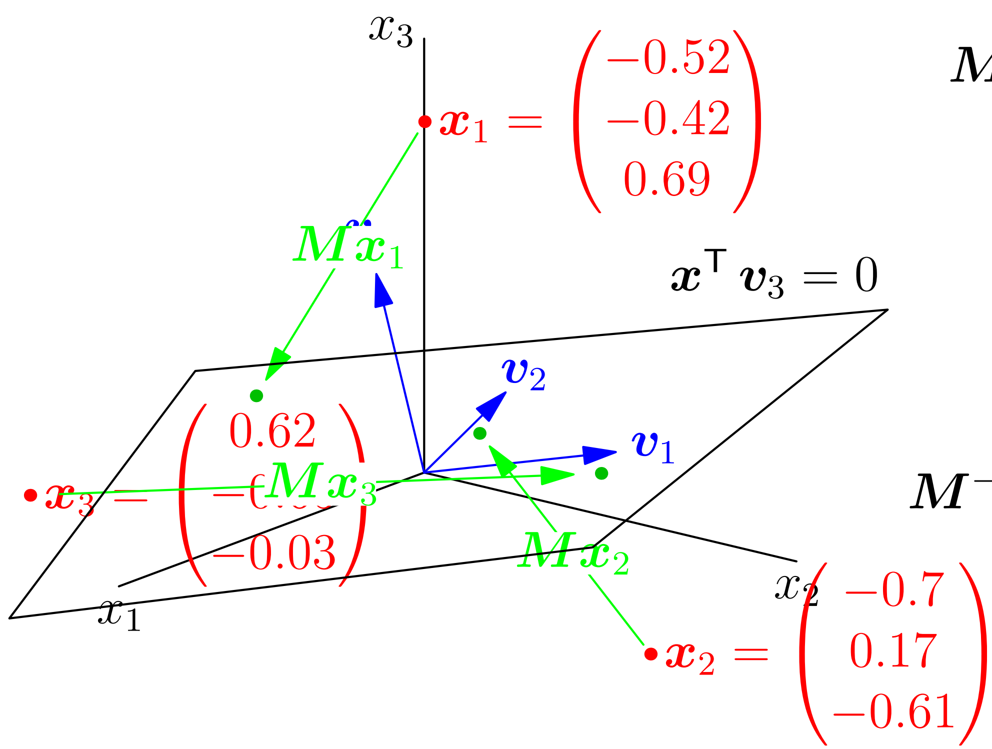
Ill-Conditioning Again



$$\boldsymbol{M} = \begin{pmatrix} 0.80 & 0.01 & 0.25 \\ 0.01 & 0.85 & 0.48 \\ 0.25 & 0.48 & 0.5 \end{pmatrix}$$

$$= \boldsymbol{V} \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \boldsymbol{V}^\mathsf{T}$$

Ill-Conditioning Again



$$\boldsymbol{M} = \begin{pmatrix} 0.80 & 0.01 & 0.25 \\ 0.01 & 0.85 & 0.48 \\ 0.25 & 0.48 & 0.5 \end{pmatrix}$$

$$= \boldsymbol{V} \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 1.25 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \boldsymbol{V}^\mathsf{T}$$

$$\boldsymbol{M}^{-1} = \begin{pmatrix} 1.9 & 1.17 & -2.1 \\ 1.17 & 3.34 & -3.8 \\ -2.1 & -3.8 & 6.8 \end{pmatrix}$$

$$= \boldsymbol{V} \begin{pmatrix} 1.25 & 0 & 0 \\ 0 & 0.8 & 0 \\ 0 & 0 & 10 \end{pmatrix} \boldsymbol{V}^\mathsf{T}$$

Condition Number

- Taking matrix inverses can be inherently unstable
- Any small error can be amplified by taking the inverse
- The stability of the inverse depends on the ratio of smallest eigenvalue to the largest eigenvalue (i.e. the biggest possible amplification compared to the smallest)
- Note that the Hilbert-norm of a matrix is the absolute value of the largest eigenvalue
- The condition number is given by

$$\|\mathbf{M}\|_H \times \|\mathbf{M}^{-1}\|_H = \frac{|\lambda_{\mathsf{max}}|}{|\lambda_{\mathsf{min}}|}$$

- Large condition number implies very ill-conditioned

Condition Number

- Taking matrix inverses can be inherently unstable
- Any small error can be amplified by taking the inverse
- The stability of the inverse depends on the ratio of smallest eigenvalue to the largest eigenvalue (i.e. the biggest possible amplification compared to the smallest)
- Note that the Hilbert-norm of a matrix is the absolute value of the largest eigenvalue
- The condition number is given by

$$\|\mathbf{M}\|_H \times \|\mathbf{M}^{-1}\|_H = \frac{|\lambda_{\text{max}}|}{|\lambda_{\text{min}}|}$$

- Large condition number implies very ill-conditioned

Condition Number

- Taking matrix inverses can be inherently unstable
- Any small error can be amplified by taking the inverse
- The stability of the inverse depends on the ratio of smallest eigenvalue to the largest eigenvalue (i.e. the biggest possible amplification compared to the smallest)
- Note that the Hilbert-norm of a matrix is the absolute value of the largest eigenvalue
- The condition number is given by

$$\|\mathbf{M}\|_H \times \|\mathbf{M}^{-1}\|_H = \frac{|\lambda_{\mathsf{max}}|}{|\lambda_{\mathsf{min}}|}$$

- Large condition number implies very ill-conditioned

Condition Number

- Taking matrix inverses can be inherently unstable
- Any small error can be amplified by taking the inverse
- The stability of the inverse depends on the ratio of smallest eigenvalue to the largest eigenvalue (i.e. the biggest possible amplification compared to the smallest)
- Note that the Hilbert-norm of a matrix is the absolute value of the largest eigenvalue
- The condition number is given by

$$\|\mathbf{M}\|_H \times \|\mathbf{M}^{-1}\|_H = \frac{|\lambda_{\mathsf{max}}|}{|\lambda_{\mathsf{min}}|}$$

- Large condition number implies very ill-conditioned

Condition Number

- Taking matrix inverses can be inherently unstable
- Any small error can be amplified by taking the inverse
- The stability of the inverse depends on the ratio of smallest eigenvalue to the largest eigenvalue (i.e. the biggest possible amplification compared to the smallest)
- Note that the Hilbert-norm of a matrix is the absolute value of the largest eigenvalue
- The condition number is given by

$$\|\mathbf{M}\|_H \times \|\mathbf{M}^{-1}\|_H = \frac{|\lambda_{\mathsf{max}}|}{|\lambda_{\mathsf{min}}|}$$

- Large condition number implies very ill-conditioned

Condition Number

- Taking matrix inverses can be inherently unstable
- Any small error can be amplified by taking the inverse
- The stability of the inverse depends on the ratio of smallest eigenvalue to the largest eigenvalue (i.e. the biggest possible amplification compared to the smallest)
- Note that the Hilbert-norm of a matrix is the absolute value of the largest eigenvalue
- The condition number is given by

$$\|\mathbf{M}\|_H \times \|\mathbf{M}^{-1}\|_H = \frac{|\lambda_{\text{max}}|}{|\lambda_{\text{min}}|}$$

- Large condition number implies very ill-conditioned

Summary

- Linear mappings are commonly used in machine learning algorithms such as regression
- We will often meet the pseudo-inverse involving inverting $\mathbf{X}^{\mathsf{T}}\mathbf{X}$
- They can be inherently unstable to noise in the inputs
- We can understand symmetric operators by looking at their eigenvectors
- Function spaces can similarly be understood in terms of eigenfunctions

Summary

- Linear mappings are commonly used in machine learning algorithms such as regression
- We will often meet the pseudo-inverse involving inverting $\mathbf{X}^{\mathsf{T}}\mathbf{X}$
- They can be inherently unstable to noise in the inputs
- We can understand symmetric operators by looking at their eigenvectors
- Function spaces can similarly be understood in terms of eigenfunctions

Summary

- Linear mappings are commonly used in machine learning algorithms such as regression
- We will often meet the pseudo-inverse involving inverting $\mathbf{X}^\mathsf{T}\mathbf{X}$
- They can be inherently unstable to noise in the inputs
- We can understand symmetric operators by looking at their eigenvectors
- Function spaces can similarly be understood in terms of eigenfunctions

Summary

- Linear mappings are commonly used in machine learning algorithms such as regression
- We will often meet the pseudo-inverse involving inverting $\mathbf{X}^\mathsf{T}\mathbf{X}$
- They can be inherently unstable to noise in the inputs
- We can understand symmetric operators by looking at their eigenvectors
- Function spaces can similarly be understood in terms of eigenfunctions

Summary

- Linear mappings are commonly used in machine learning algorithms such as regression
- We will often meet the pseudo-inverse involving inverting $\mathbf{X}^\mathsf{T}\mathbf{X}$
- They can be inherently unstable to noise in the inputs
- We can understand symmetric operators by looking at their eigenvectors
- Function spaces can similarly be understood in terms of eigenfunctions