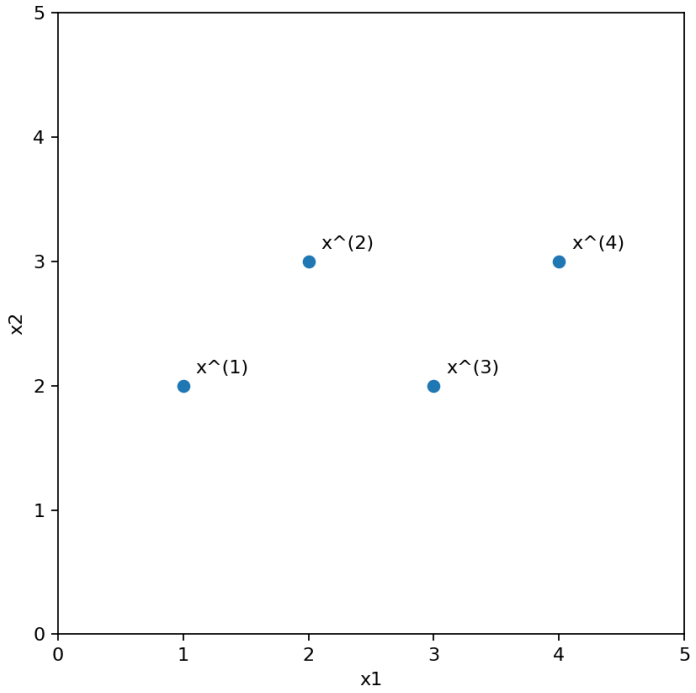
# 1 PCA: Basic Concepts

Consider dataset $\mathcal{D} = \{\mathbf{x}^{(1)} = \begin{bmatrix}1\\2\end{bmatrix}, \mathbf{x}^{(2)} = \begin{bmatrix}2\\3\end{bmatrix}, \mathbf{x}^{(3)} = \begin{bmatrix}3\\2\end{bmatrix}, \mathbf{x}^{(4)} = \begin{bmatrix}4\\3\end{bmatrix}\}$. A visualization of the dataset is as below.




## 1.1 Centering Data

Centering is crucial for PCA. We must preprocess data so that all features have zero mean before applying PCA, i.e.

$$\frac{1}{N}\sum_{i=1}^{N}\mathbf{x}^{(i)} = \vec{0}$$

Compute the centered dataset:

$\mathbf{x}^{(1)} =$ _____ $\mathbf{x}^{(2)} =$ _____

$\mathbf{x}^{(3)} =$ _____ $\mathbf{x}^{(4)} =$ _____

First, note that $\hat{E}[x_1] = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_1^{(i)} = 2.5$ and $\hat{E}[x_2] = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_2^{(i)} = 2.5$

$\mathbf{x}^{(1)} = \begin{bmatrix}-1.5\\-0.5\end{bmatrix}$ $\mathbf{x}^{(2)} = \begin{bmatrix}-0.5\\0.5\end{bmatrix}$

$\mathbf{x}^{(3)} = \begin{bmatrix}0.5\\-0.5\end{bmatrix}$ $\mathbf{x}^{(4)} = \begin{bmatrix}1.5\\0.5\end{bmatrix}$

## 1.2 Unit vector

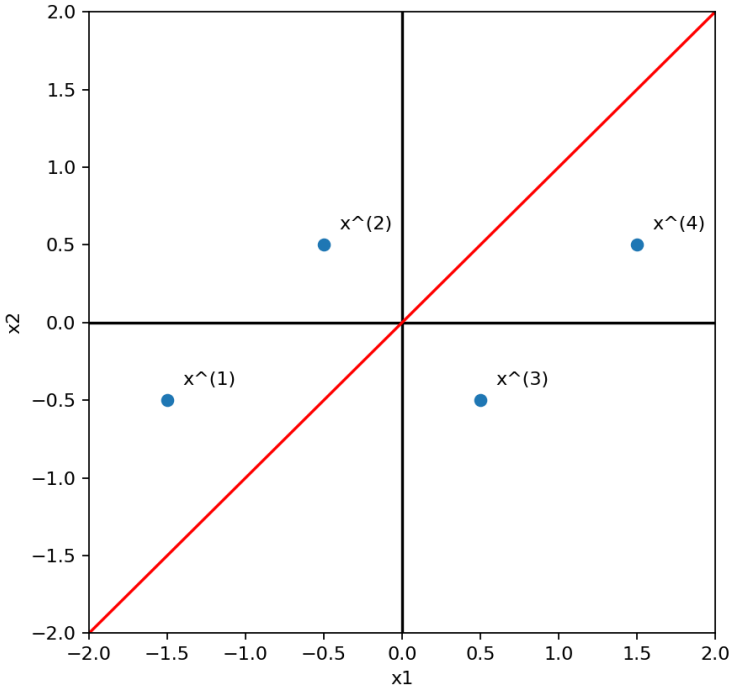
In order to easily compute the projected coordinates of data, we need to make the projected directions unit vectors. Suppose we want to project our data onto the vector $\mathbf{v} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. Normalize $\mathbf{v}$ to be a unit vector.

$\mathbf{v} =$ _____

$\mathbf{v} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$

## 1.3 Project Data

The centered data should now look like the following:




Suppose we want to project the centered data onto $\mathbf{v}$, where $\mathbf{v}$ goes through the origin.

Compute the magnitude of the projections, i.e. compute $z^{(i)} = \mathbf{v}^T \mathbf{x}^{(i)}, \forall 1 \leq i \leq N$.

$z^{(1)} =$ _____ $z^{(2)} =$ _____

$z^{(3)} =$ _____ $z^{(4)} =$ _____

$$z^{(1)} = [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}] \begin{bmatrix} -1.5 \\ -0.5 \end{bmatrix} = \frac{-3}{2\sqrt{2}} - \frac{1}{2\sqrt{2}} = -\frac{4}{2\sqrt{2}} = -\sqrt{2}$$
$$z^{(2)} = [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}] \begin{bmatrix} -0.5 \\ 0.5 \end{bmatrix} = \frac{-1}{2\sqrt{2}} + \frac{1}{2\sqrt{2}} = 0$$
$$z^{(3)} = [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}] \begin{bmatrix} 0.5 \\ -0.5 \end{bmatrix} = \frac{-1}{2\sqrt{2}} + \frac{1}{2\sqrt{2}} = 0$$
$$z^{(4)} = [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}] \begin{bmatrix} 1.5 \\ 0.5 \end{bmatrix} = \frac{3}{2\sqrt{2}} + \frac{1}{2\sqrt{2}} = \frac{4}{2\sqrt{2}} = \sqrt{2}$$

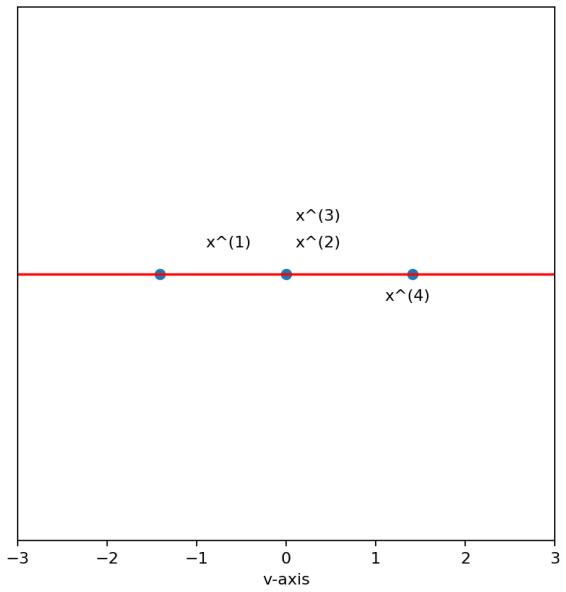
Let $\mathbf{x}^{(i)'}$ be the projected point of $\mathbf{x}^{(i)}$. Note that $\mathbf{x}^{(i)'} = \mathbf{v}^T\mathbf{x}^{(i)}\mathbf{v} = z^{(i)}\mathbf{v}$. Compute the projected coordinates:

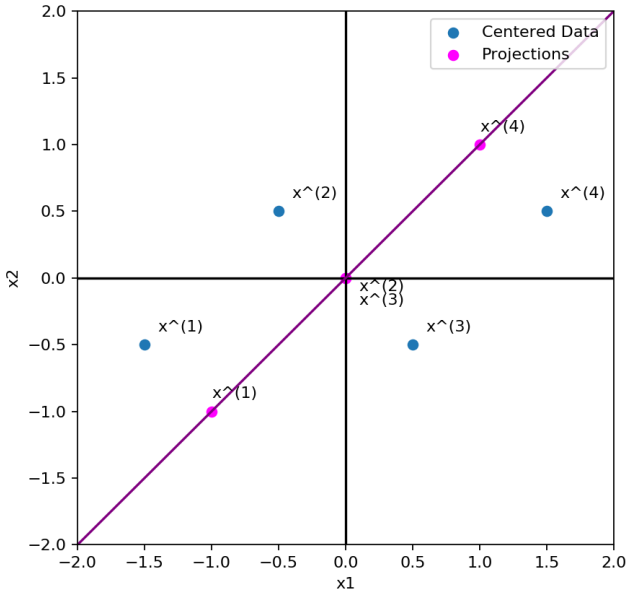
$\mathbf{x}^{(1)'} =$ _____ $\mathbf{x}^{(2)'} =$ _____

$\mathbf{x}^{(3)'} =$ _____ $\mathbf{x}^{(4)'} =$ _____

$$\mathbf{x}^{(1)'} = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \quad \mathbf{x}^{(2)'} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$\mathbf{x}^{(3)'} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \mathbf{x}^{(4)'} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Below is a visualization of the projections:



## 1.4 Reconstruction Error

One of the two goals of PCA is to find new directions to project our dataset onto such that it **minimizes the reconstruction error**, where the reconstruction error is defined as following:

$$\text{Reconstruction Error} = \sum_{i=1}^{N} \|\mathbf{x}^{(i)'} - \mathbf{x}^{(i)}\|_2^2$$

What is the reconstruction error in our case?

$$\text{Reconstruction Error} = \_\_\_\_\_$$

$$\begin{aligned}
\text{Reconstruction Error} &= \sum_{i=1}^{N} \|\mathbf{x}^{(i)'} - \mathbf{x}^{(i)}\|_2^2 \\
&= \| \begin{bmatrix} -1 \\ -1 \end{bmatrix} - \begin{bmatrix} -1.5 \\ -0.5 \end{bmatrix} \|_2^2 + \| \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} -0.5 \\ 0.5 \end{bmatrix} \|_2^2 + \| \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.5 \\ -0.5 \end{bmatrix} \|_2^2 + \| \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1.5 \\ 0.5 \end{bmatrix} \|_2^2 \\
&= \sqrt{(\frac{1}{2})^2 + (-\frac{1}{2})^2} + \sqrt{(\frac{1}{2})^2 + (-\frac{1}{2})^2} + \sqrt{(-\frac{1}{2})^2 + (\frac{1}{2})^2} + \sqrt{(-\frac{1}{2})^2 + (\frac{1}{2})^2} \\
&= \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} \\
&= \frac{4}{\sqrt{2}} = 2\sqrt{2}
\end{aligned}$$

## 1.5 Variance of Projected Data

Another goal is to find new directions to project our dataset onto such that it **maximizes the variance of the projections**, where the variance of projections is defined as following:

$$\begin{aligned}
\text{variance of projection} &= \sum_{i=1}^{N} (z^{(i)} - \hat{E}[z])^2 \\
&= \sum_{i=1}^{N} (\mathbf{v}^T \mathbf{x}^{(i)} - \frac{1}{N} \sum_{j=1}^{N} \{\mathbf{v}^T \mathbf{x}^{(j)})^2 \\
&= \sum_{i=1}^{N} (\mathbf{v}^T \mathbf{x}^{(i)} - \mathbf{v}^T (\frac{1}{N} \sum_{j=1}^{N} \mathbf{x}^{(j)}))^2 \\
&= \sum_{i=1}^{N} (\mathbf{v}^T \mathbf{x}^{(i)} - \mathbf{v}^T \vec{0})^2 \\
&= \sum_{i=1}^{N} (\mathbf{v}^T \mathbf{x}^{(i)})^2
\end{aligned}$$

What is the variance of the projections?

$$\text{variance} = \_\_\_\_\_$$

$$\begin{aligned}\text{variance} &= \sum_{i=1}^{N} (z^{(i)})^2 \\ &= (-\sqrt{2})^2 + 0^2 + 0^2 + (\sqrt{2})^2 \\ &= 2 + 0 + 0 + 2 \\ &= 4\end{aligned}$$

## 2 Deriving the second principal component

1. Recall that PCA tries to minimize the reconstruction error between the data points and the projections of the data points onto the principle componenets. We have derived the first principle component in lecture using maximum variance. This week we will derive the second principle component using minimum reconstruction error. Let $J(\mathbf{v_2}) = \frac{1}{n}\sum_{i=1}^{n}||\mathbf{x^{(i)}} - z_1^{(i)}\mathbf{v_1} - z_2^{(i)}\mathbf{v_2}||_2^2$ given the constraints $\mathbf{v_1}^T\mathbf{v_2} = 0$ and $\mathbf{v_2}^T\mathbf{v_2} = 1$. Here, n is the number of data points, $\mathbf{v_1}, \mathbf{v_2}$ are the first and the second principle component, and $\mathbf{z^{(i)}}$ denotes the principle encoding of the ith data point $\mathbf{x^{(i)}}$. Recall that we've defined $z_1^{(i)} = \mathbf{v_1}^T\mathbf{x^{(i)}}$. Define $z_2^{(i)}$, which is the second principle encoding of $\mathbf{x^{(i)}}$.

   $z_2^{(i)} = \mathbf{v_2}^T\mathbf{x^{(i)}}$

2. Show that the value of $\mathbf{v_2}$ that minimizes $J$ is given by the eigenvector of $\mathbf{C} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x^{(i)}}\mathbf{x^{(i)}}^T)$ with the second largest eigenvalue. Assumed we have already proved the $\mathbf{v_1}$ is the eigenvector of $\mathbf{C}$ with the largest eigenvalue.

   Plug in $z_2^{(i)}$ and the constraints into $J(\mathbf{v_2})$ (here k denotes some constant that does not depend on $\mathbf{v_2}$), we have

$$\begin{aligned}
J(\mathbf{v_2}) &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x^{(i)}}^T\mathbf{x^{(i)}} - z_1^{(i)}\mathbf{v_1}^T\mathbf{x^{(i)}} - z_2^{(i)}\mathbf{v_2}^T\mathbf{x^{(i)}} - z_1^{(i)}\mathbf{x^{(i)}}^T\mathbf{v_1} + z_1^{(i)2}\mathbf{v_1}^T\mathbf{v_1} - z_2^{(i)}\mathbf{v_2}^T\mathbf{x^{(i)}} + z_2^{(i)2}\mathbf{v_2}^T\mathbf{v_2}) \\
&= \frac{1}{n}\sum_{i=1}^{n}(k - 2z_2^{(i)}\mathbf{v_2}^T\mathbf{x^{(i)}} + z_2^{(i)2}) \\
&= \frac{1}{n}\sum_{i=1}^{n}(-2\mathbf{v_2}^T\mathbf{x^{(i)}}\mathbf{x^{(i)}}^T\mathbf{v_2} + \mathbf{v_2}^T\mathbf{x^{(i)}}\mathbf{x^{(i)}}^T\mathbf{v_2} + k) \\
&= -\mathbf{v_2}^T\mathbf{C}\mathbf{v_2} + k
\end{aligned}$$

   In order to minimize J with constraints $\mathbf{v_2}^T\mathbf{v_2} = 1$, we use method of Lagrange multipliers and so we have $L = -\mathbf{v_2}^T\mathbf{C}\mathbf{v_2} + \lambda(\mathbf{v_2}^T\mathbf{v_2} - 1)$. Take derivative of $\mathbf{v_2}$, we have

$$\frac{\partial L}{\partial \mathbf{v_2}} = -2\mathbf{C}\mathbf{v_2} + 2\lambda\mathbf{v_2} = 0$$

   Therefore, we have

$$\mathbf{C}\mathbf{v_2} = \lambda\mathbf{v_2}$$

# 3 Equivalence Between Maximum Variance and Minimum Reconstruction Error

As alluded to above, maximizing the variance is equivalent to minimizing the reconstruction error. Argue why.

Hint:

$$||x^{(i)} - (v^T x^{(i)})v||^2 = ||x^{(i)}||^2 - (v^T x^{(i)})^2$$

via orthogonality and the unit norm of v.

$$v^* = argmin_{||v||^2=1} \frac{1}{n} \sum_{i=1}^{n} ||x^{(i)} - v^T xv||^2 = argmin_{||v||^2=1} \frac{1}{n} \sum_{i=1}^{n} ||x^{(i)}||^2 - (v^T x^{(i)})^2$$

$$= argmax_{||v||^2=1} \frac{1}{n} \sum_{i=1}^{n} (v^T x^{(i)})^2$$

# 4 SVD

1. Find the SVD of $X = \begin{bmatrix} 4 & 4 \\ 3 & -3 \end{bmatrix}$

   To find the SVD of $X$, we first compute the matrices $X^TX$ and $XX^T$.
   $X^TX = \begin{bmatrix} 25 & 7 \\ 7 & 25 \end{bmatrix}$
   $XX^T = \begin{bmatrix} 32 & 0 \\ 0 & 18 \end{bmatrix}$
   The singular values of $X$ are square roots of eigenvalues of $X^TX$ and $XX^T$ (they have the same eigenvalues). They are $4\sqrt{2}$ and $3\sqrt{2}$.
   Then we know that $S = \begin{bmatrix} 4\sqrt{2} & 0 \\ 0 & 3\sqrt{2} \end{bmatrix}$
   Now, We notice that the singular value decomposition of $X$ is $X = USV^T$, where columns of $U$ are eigenvectors of $XX^T$ and columns of $V$ are eigenvectors of $X^TX$.
   We also note that $U$ and $V$ are both orthogonal matrices, which means that their columns are orthonormal.
   We first find two orthonormal eigenvectors of $X^TX$. They are $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$.
   Now, we find two orthonormal eigenvectors of $XX^T$. They are $(1, 0)$ and $(0, -1)$.
   So we have the SVD of $X$ is $X = USV^T$, where $U = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, $S = \begin{bmatrix} 4\sqrt{2} & 0 \\ 0 & 3\sqrt{2} \end{bmatrix}$, $V = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$.

2. How does SVD relate to PCA?

   Recall that the principle components $v$ in the PCA algorithm are precisely the eigenvectors of the covariance matrix $X^TX$. On the other hand, the columns of the matrix $V$ are an orthonormal set of eigenvectors for $X^TX$. So, given the SVD of $X$, it is trivial to find the principle components of $X$.

3. How does SVD relate to Matrix Factorization?

   Matrix Factorization is a latent-variable method for building recommender systems, classified under Collaborative Filtering. In Matrix Factorization, we are given a sparse matrix $R$ of ratings, where the rows are users and columns are items, and the entries are the user's ratings/preferences of the item.
   Suppose $R$ has $n$ rows and $m$ columns.
   In rank-$k$ matrix factorization, we want to factorize $R$ into $R \approx \tilde{U}\tilde{V}^T$, where $\tilde{U}$ is $n \times k$ and $\tilde{V}$ is $m \times k$. The different columns of $\tilde{U}$ represent our latent variables, and our latent space has dimension $k$. $\tilde{U}$ is a mapping of each user to the low dimensional space. Likewise, $\tilde{V}$ is a mapping of each item to the low dimensional space. Our objective is to make the difference between $R$ and $\tilde{U}\tilde{V}^T$ small.
   The SVD of $R$ is $R = USV^T$. Now, let $\tilde{U} = U'S$, and $\tilde{V} = V'$. Then we obtain a rank-$m$ factorization of $R$, with difference 0 between $R$ and $\tilde{U}\tilde{V}^T$. So the SVD of $R$ gives us an optimal rank-$m$ factorization.
   Now, if we want a rank $k$ matrix factorization, we can take the first $k$ columns of $U$ and $V$ to get $U_k$ and $V_k$, and take the top-left $k \times k$ sub-matrix of $S$ to get $S_k$. Then let $\tilde{U_k} = U_kS_k$ and $\tilde{V_k} = V_k$, and we have a rank-$k$ matrix factorization of $R$.