How Principal Component Analysis works ?

Define the essential terms

1. **Dimensionality Reduction** : This is the main objective of PCA. It's the process of reducing the number of features (or variables) in a dataset. By doing so, you can simplify the data without losing much of the important information.

2. **Principal Components** : These are new, uncorrelated variables that are constructed from the original features. The principal components are ordered so that the first principal component accounts for the largest possible variance in the data, the second accounts for the next largest, and so on.

3. **Eigenvector** : In the context of PCA, the eigenvectors of the covariance matrix are the **principal components**. They are the directions in the feature space along which the data has the most variance.

4. **Eigenvalue** : Each eigenvector has a corresponding eigenvalue. The **eigenvalue** represents the magnitude of the variance along its corresponding eigenvector. A larger eigenvalue means that the principal component (eigenvector) captures more of the data's variance.

5. **Explained Variance** : This refers to the percentage of the dataset's total variance that is captured by each principal component. You typically use this to decide how many principal components to keep, as you want to retain enough components to explain a high percentage (e.g., 95%) of the variance.

Steps of Principal Component Analysis ?

Based on the steps you provided, here are the names for the five steps in the PCA algorithm :

1. **Data Normalization :** The first step involves finding the center of the data. This is typically done by **standardizing** the data, which means subtracting the mean from each data point so that the data is centered around the origin (0,0). This is a crucial preprocessing step for PCA.

2. **Find the First Principal Component (PC1) :** This step involves identifying the axis (a line in your example) that best captures the variance in the data. This axis is called the **First Principal Component (PC1)**, and it's the direction with the greatest amount of variance.

3. **Find Subsequent Principal Components :** After finding PC1, the algorithm finds the next best axis for capturing variance. This new axis, the **Second Principal Component (PC2)**, is always **orthogonal** (perpendicular) to the first.

4. **Calculate All Principal Components :** This step involves repeating the process until you have found a principal component for every original dimension. The result is a new set of orthogonal axes that are ordered by how much variance they capture.

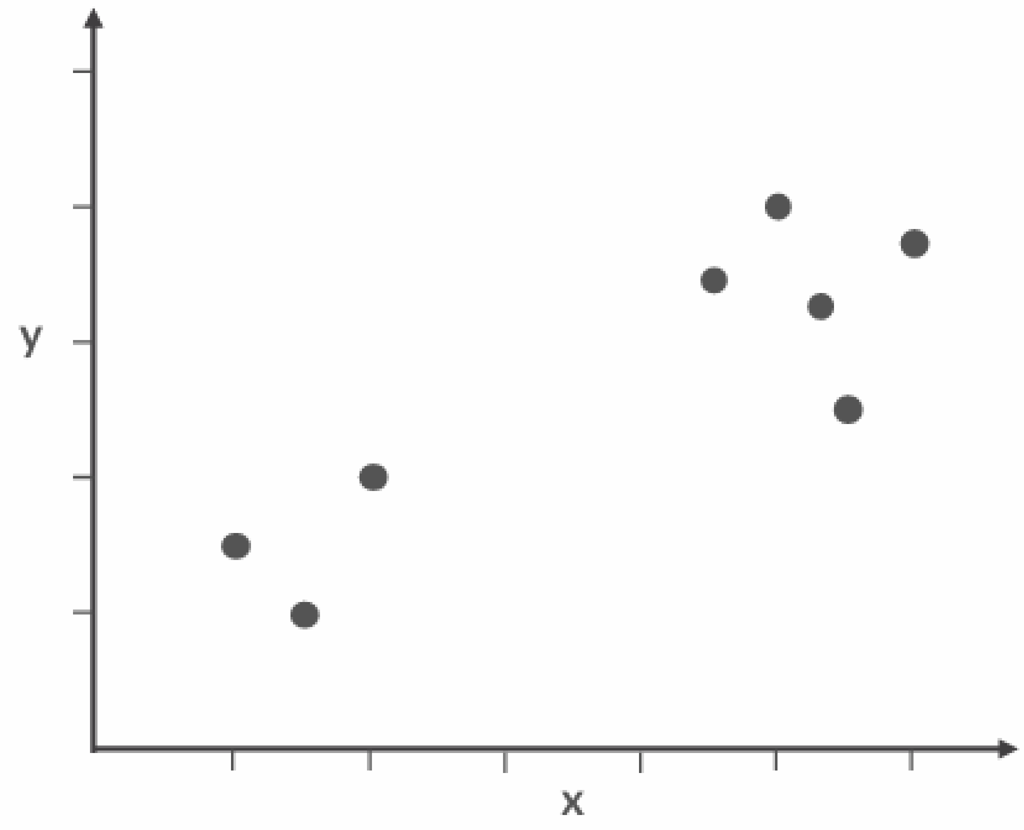
5. **Dimensionality Reduction :** In the final step, you choose a subset of the most important principal components (e.g., just PC1) to represent your data. By discarding the components that explain the least variance, you effectively reduce the dimensionality of your dataset.

Step 1 - Data Normalization

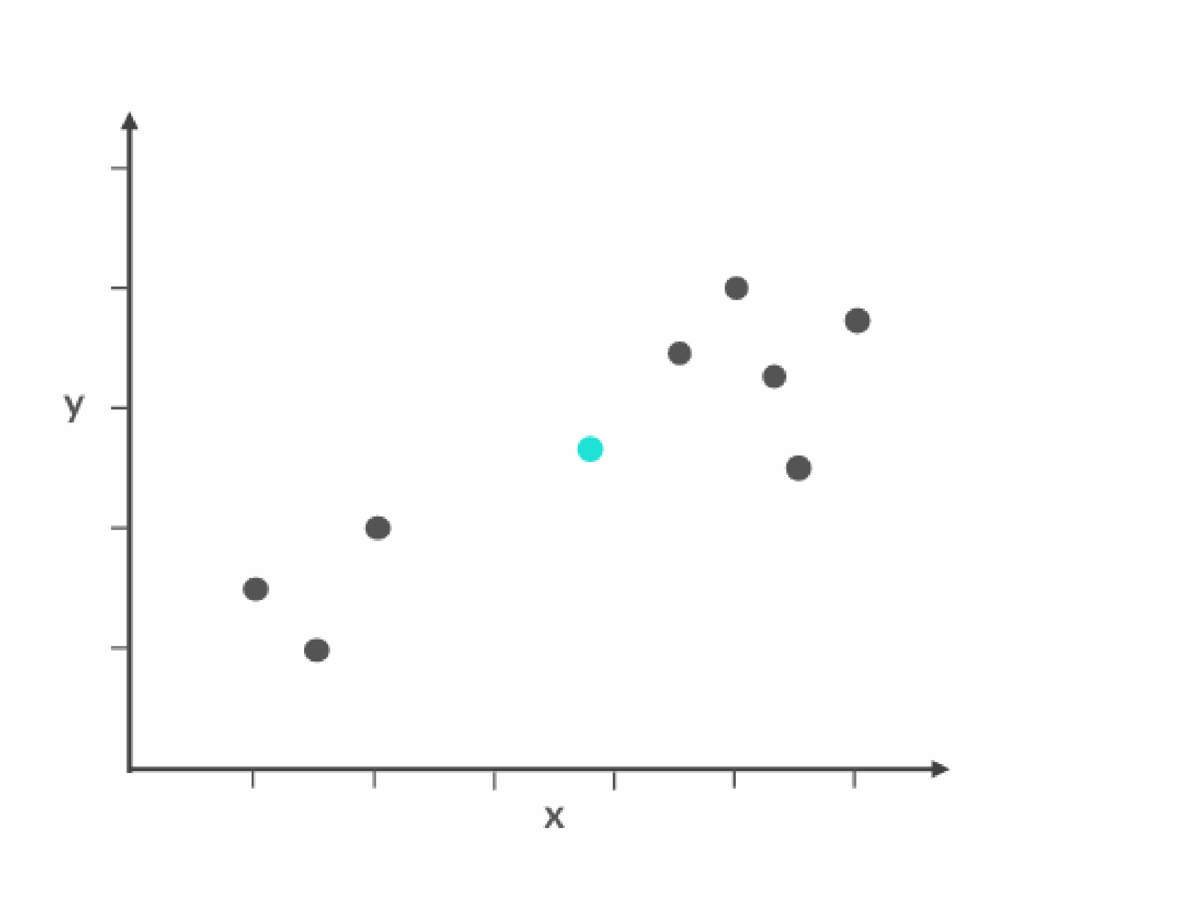
Step 1 - Initialization

The first step involves finding the center of the data. This is typically done by **standardizing** the data, which means subtracting the mean from each data point so that the data is centered around the origin (0,0). This is a crucial preprocessing step for PCA.

Step 1.1 - Create a scatter plot and find the center of the data



Step 1.2 - Create a scatter plot and find the center of the data

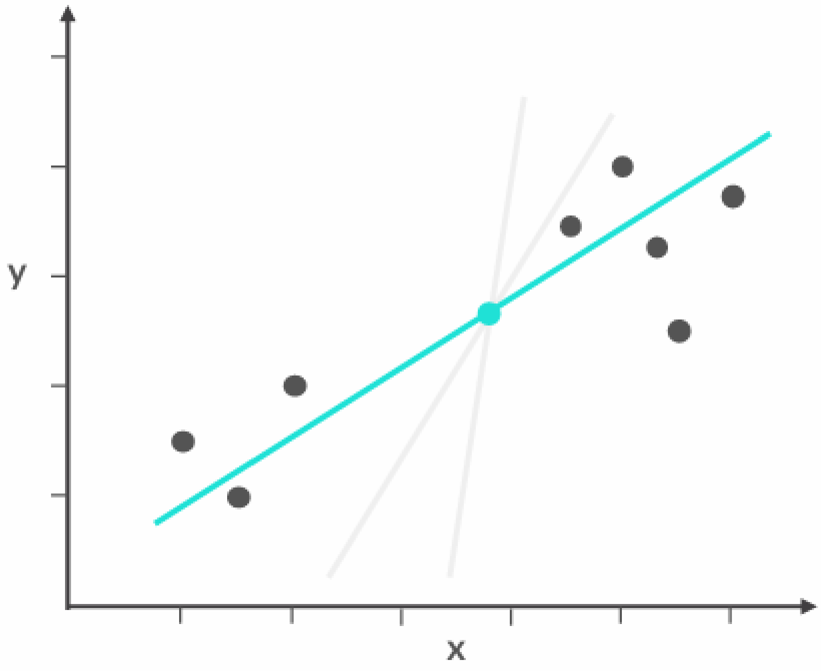


Step 2 - Principal Components

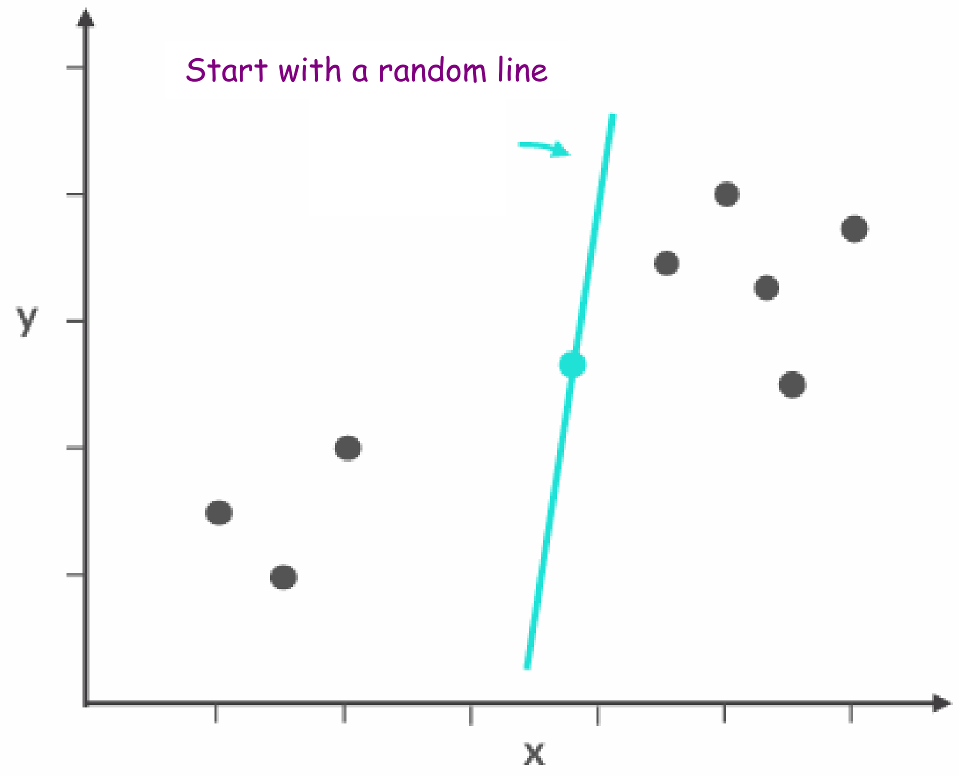
Step 2 – Principal Components

These are new, uncorrelated variables that are constructed from the original features. The principal components are ordered so that the first principal component accounts for the largest possible variance in the data, the second accounts for the next largest, and so on.

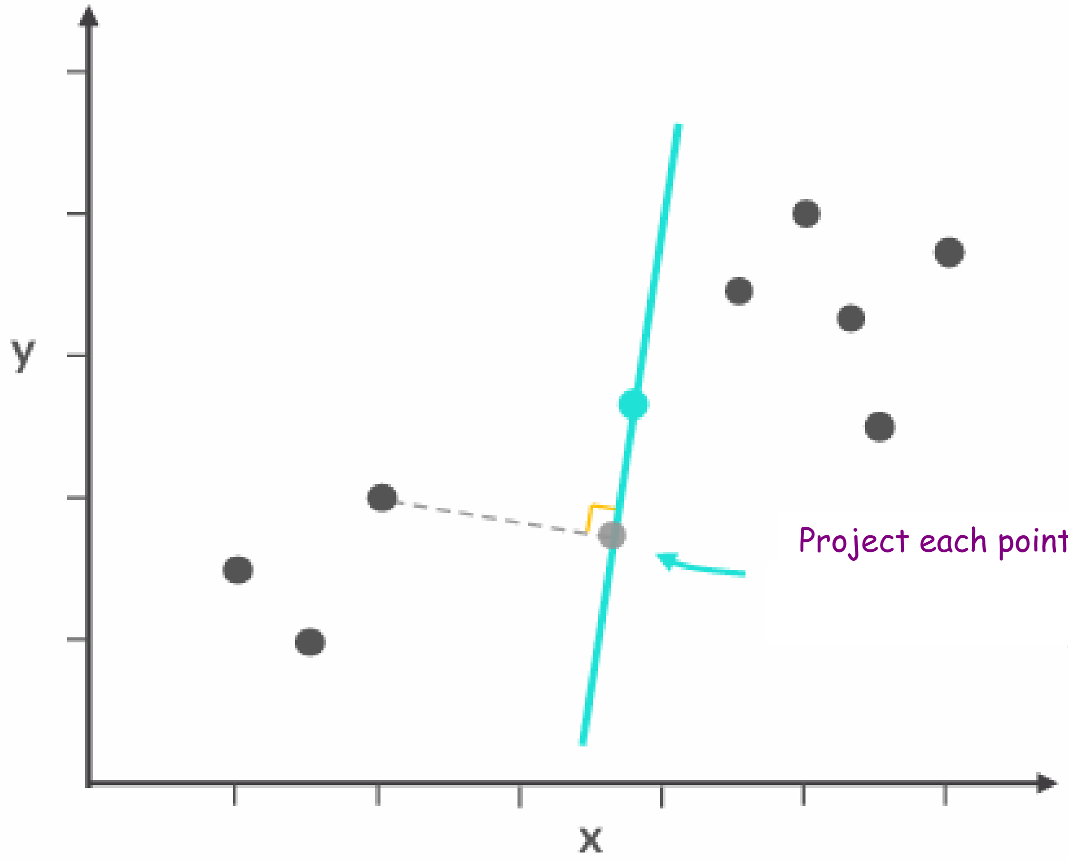
Step 2.1 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



Step 2.2 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)

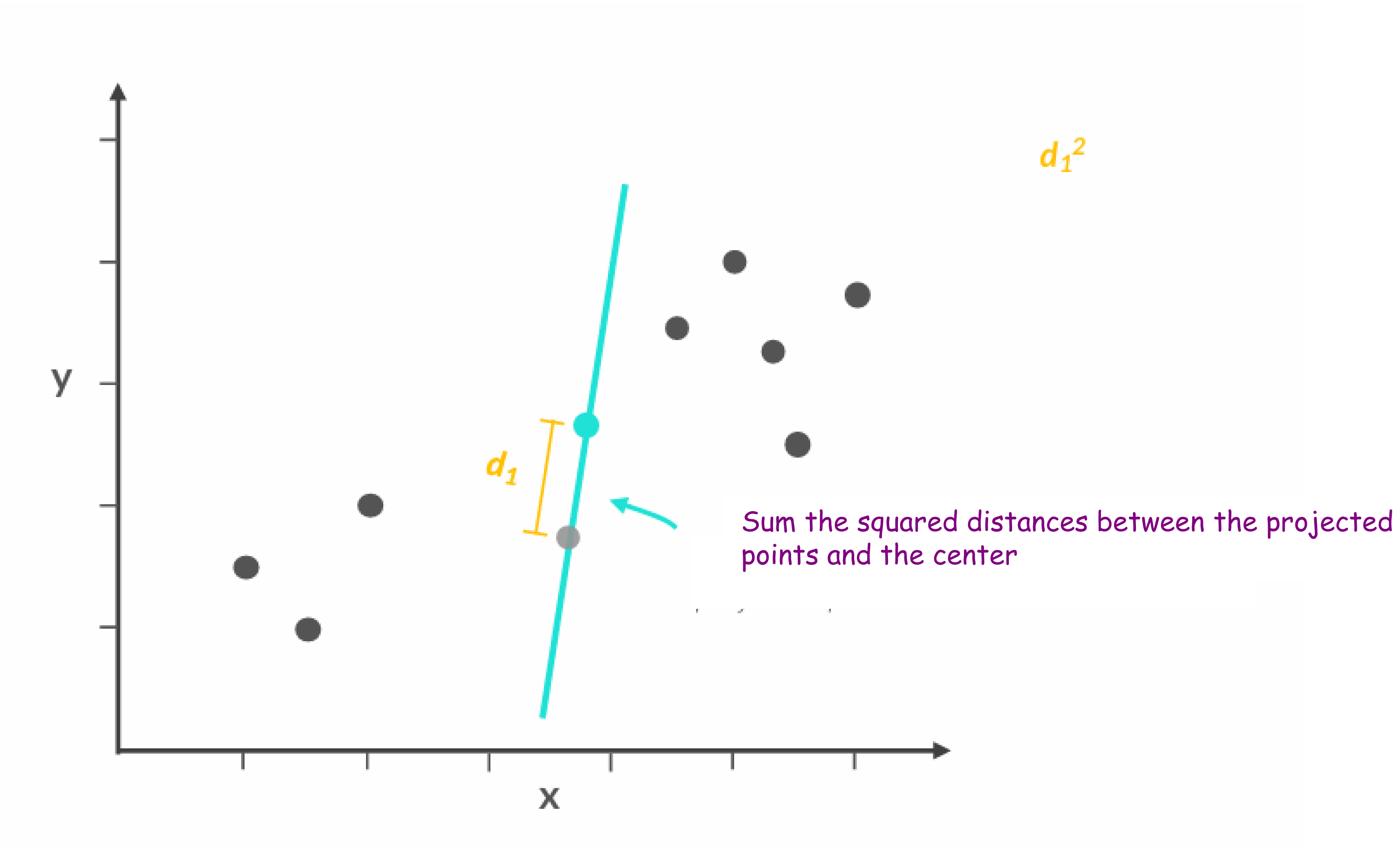


Step 2.3 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



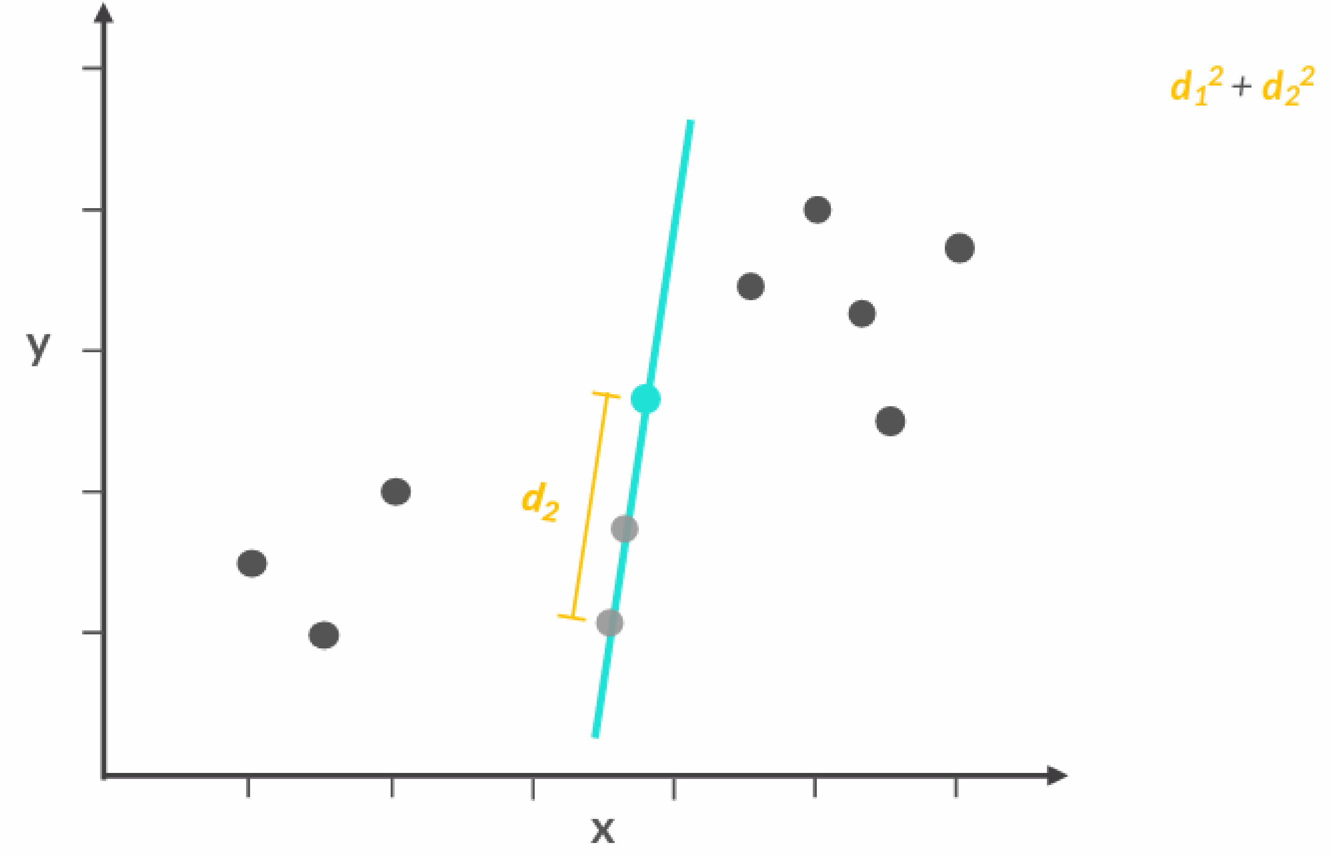
Project each point onto the line at a 90° angle

Step 2.4 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)

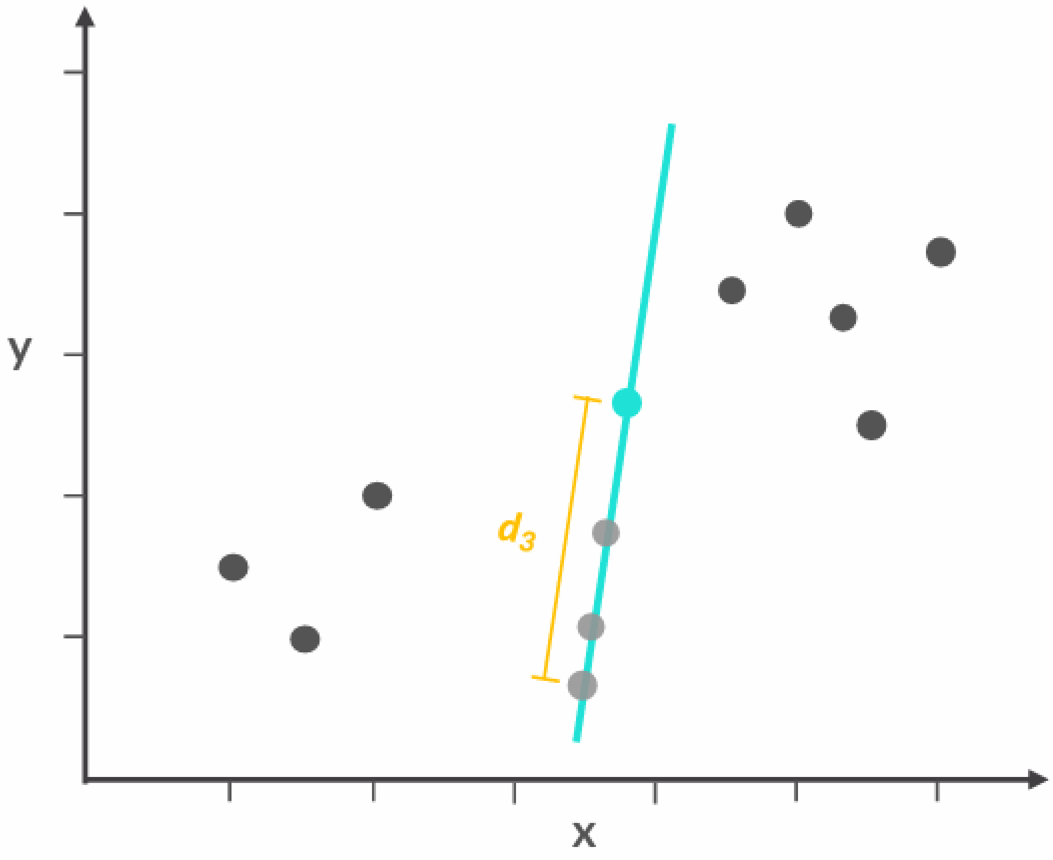


Sum the squared distances between the projected points and the center

Step 2.5 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)

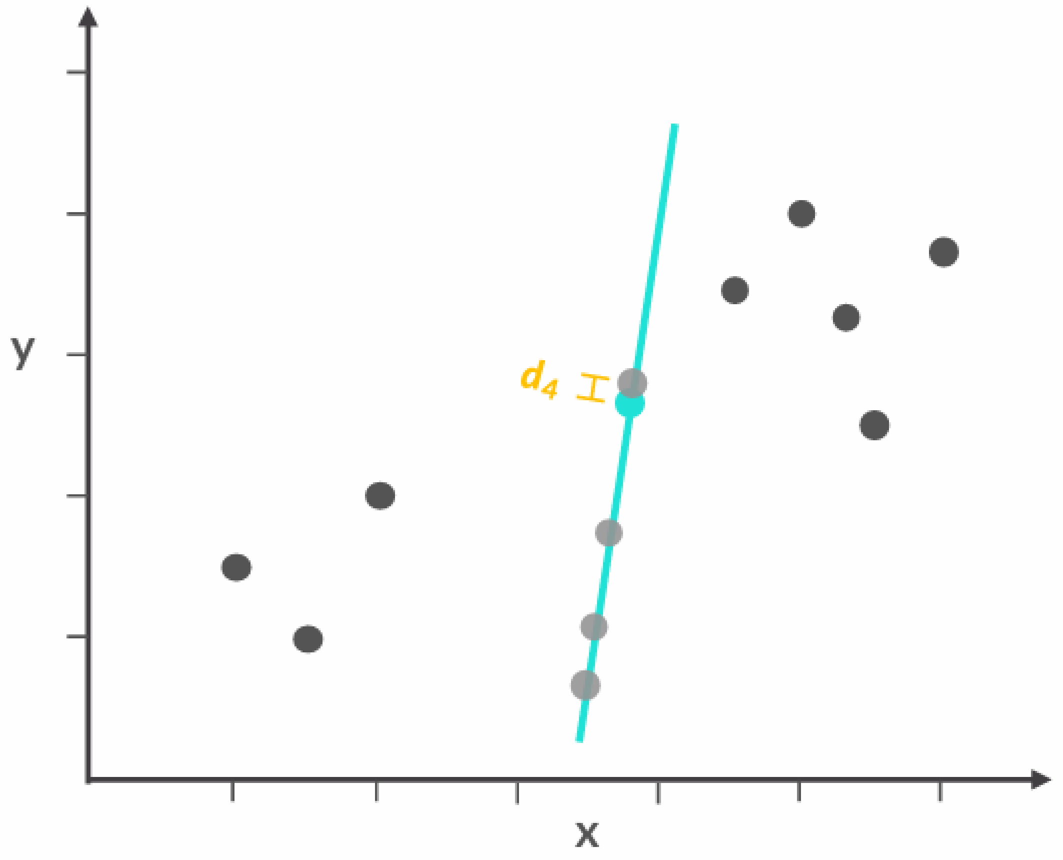


Step 2.6 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



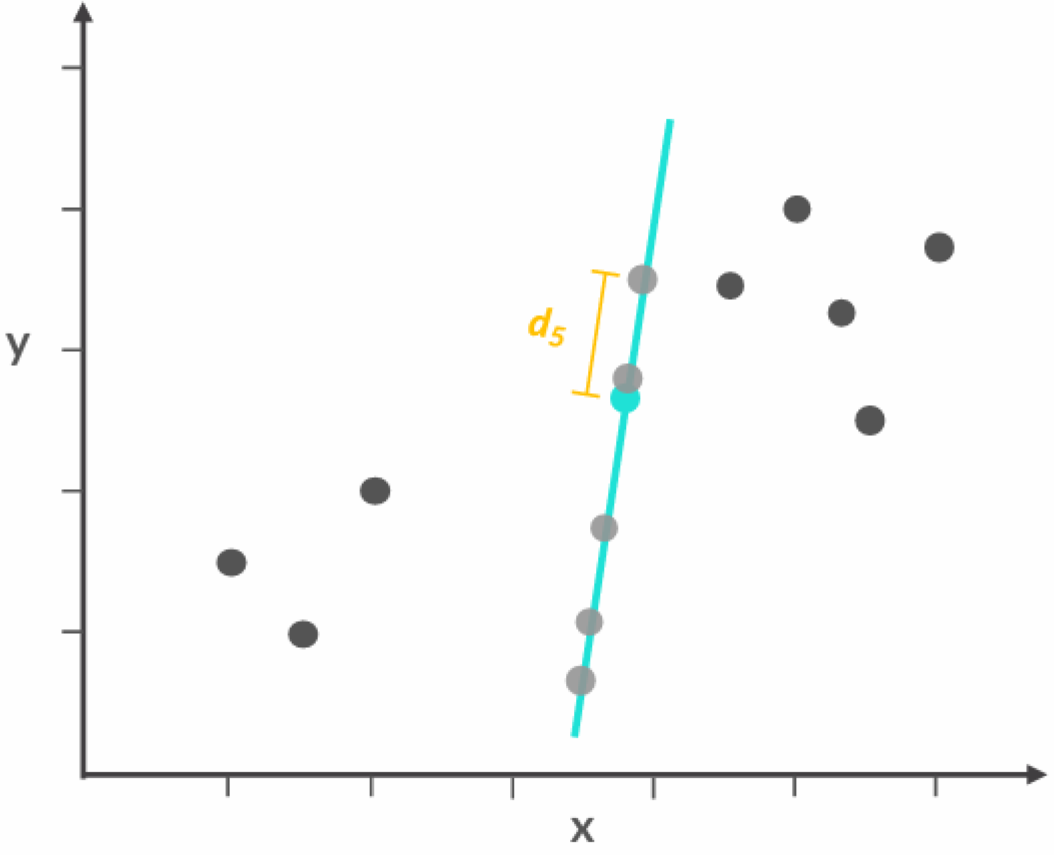
$d_1^2 + d_2^2 + d_3^2$

Step 2.7 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



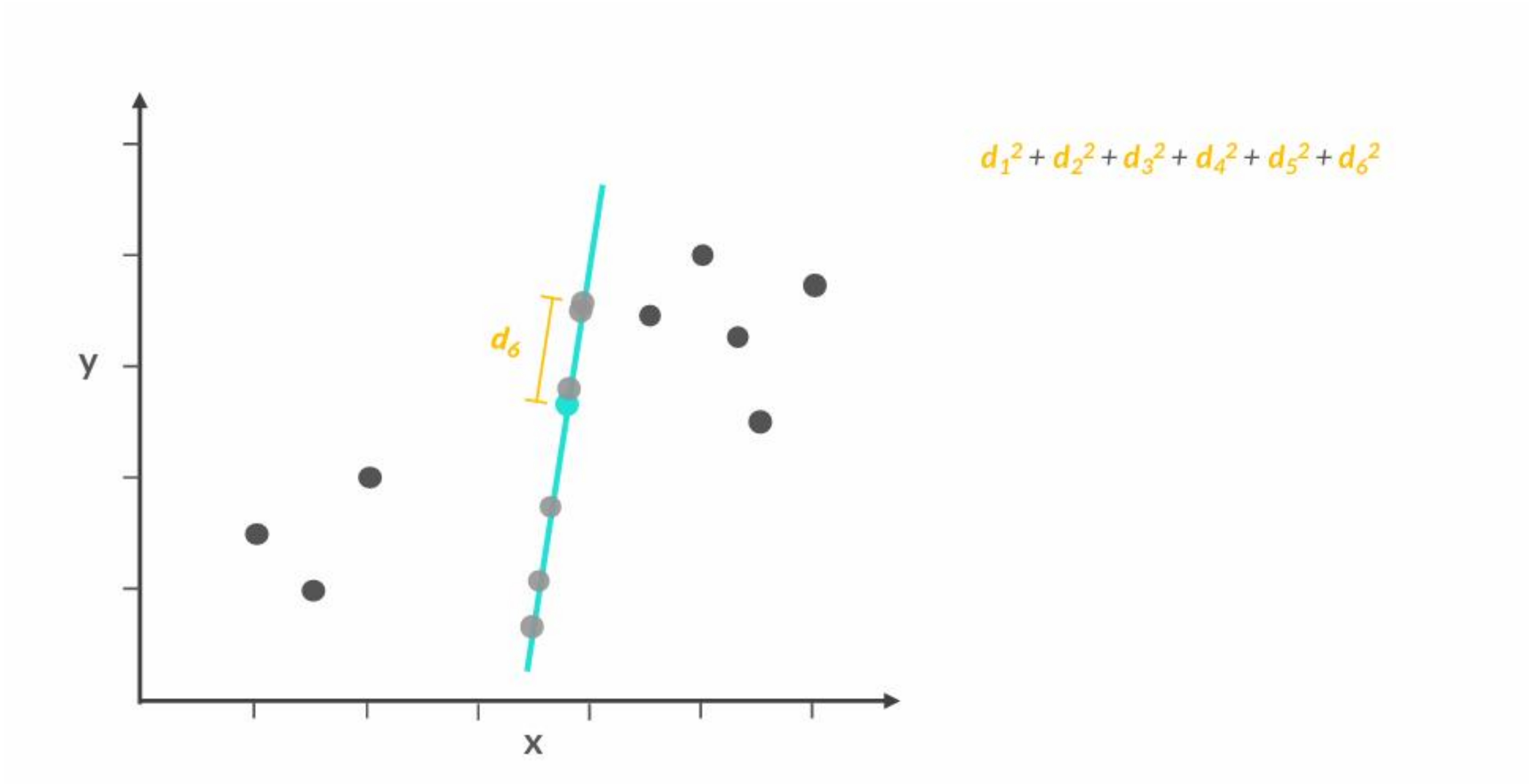
$d_1^2 + d_2^2 + d_3^2 + d_4^2$

Step 2.8 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



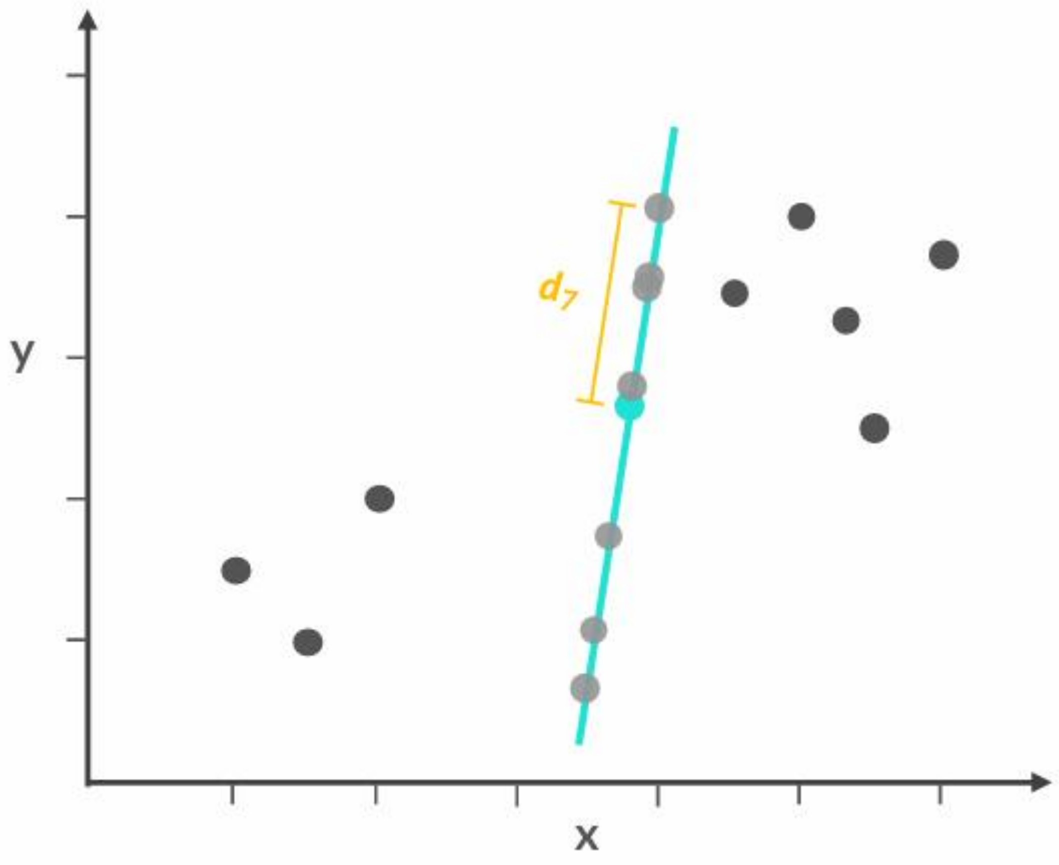
$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2$

Step 2.9 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



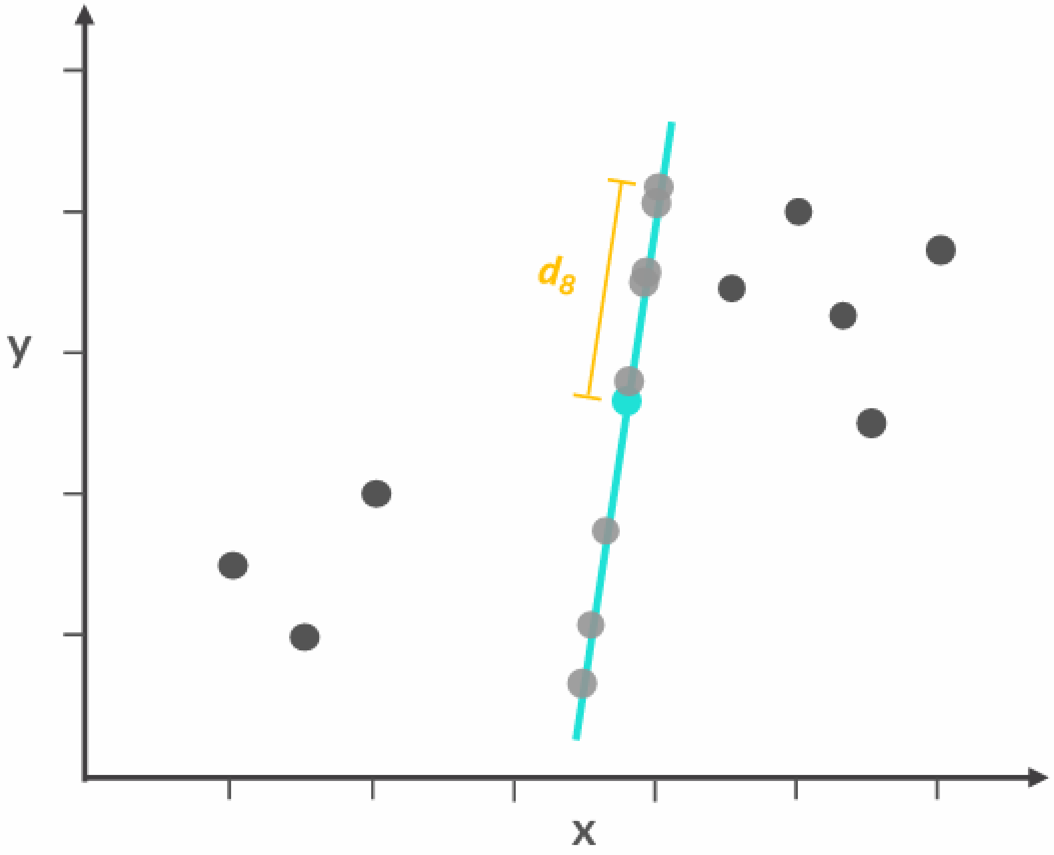
$$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2$$

Step 2.10 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



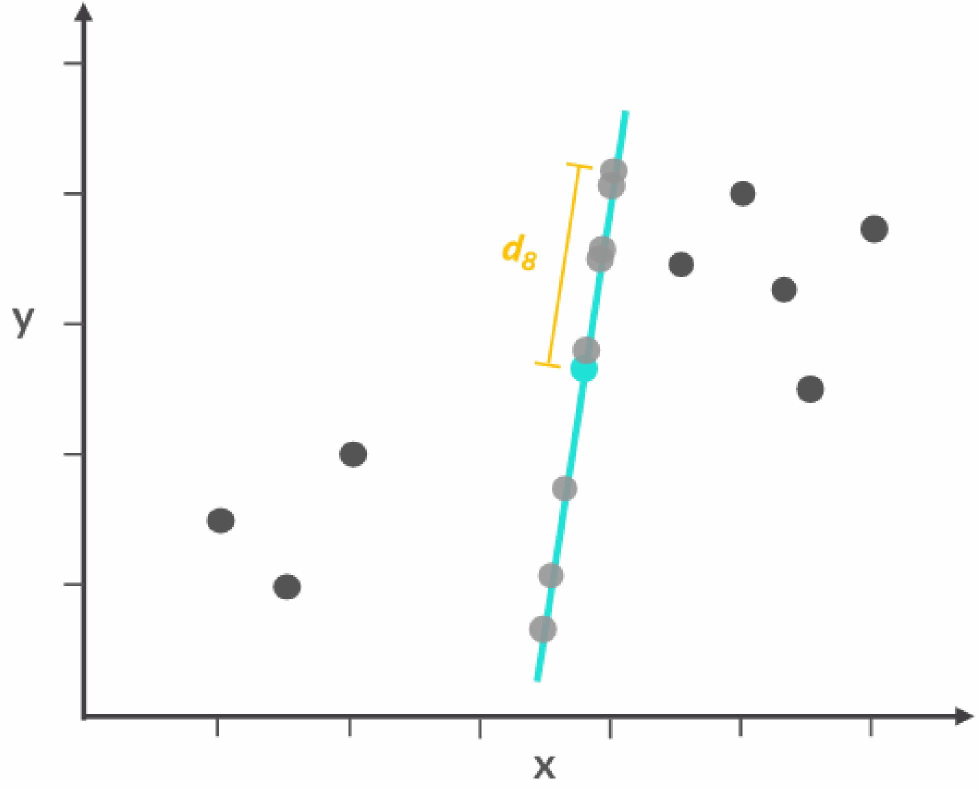
$$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2 + d_7^2$$

Step 2.11 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2 + d_7^2 + d_8^2 = \mathbf{4.8}$

Step 2.12 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



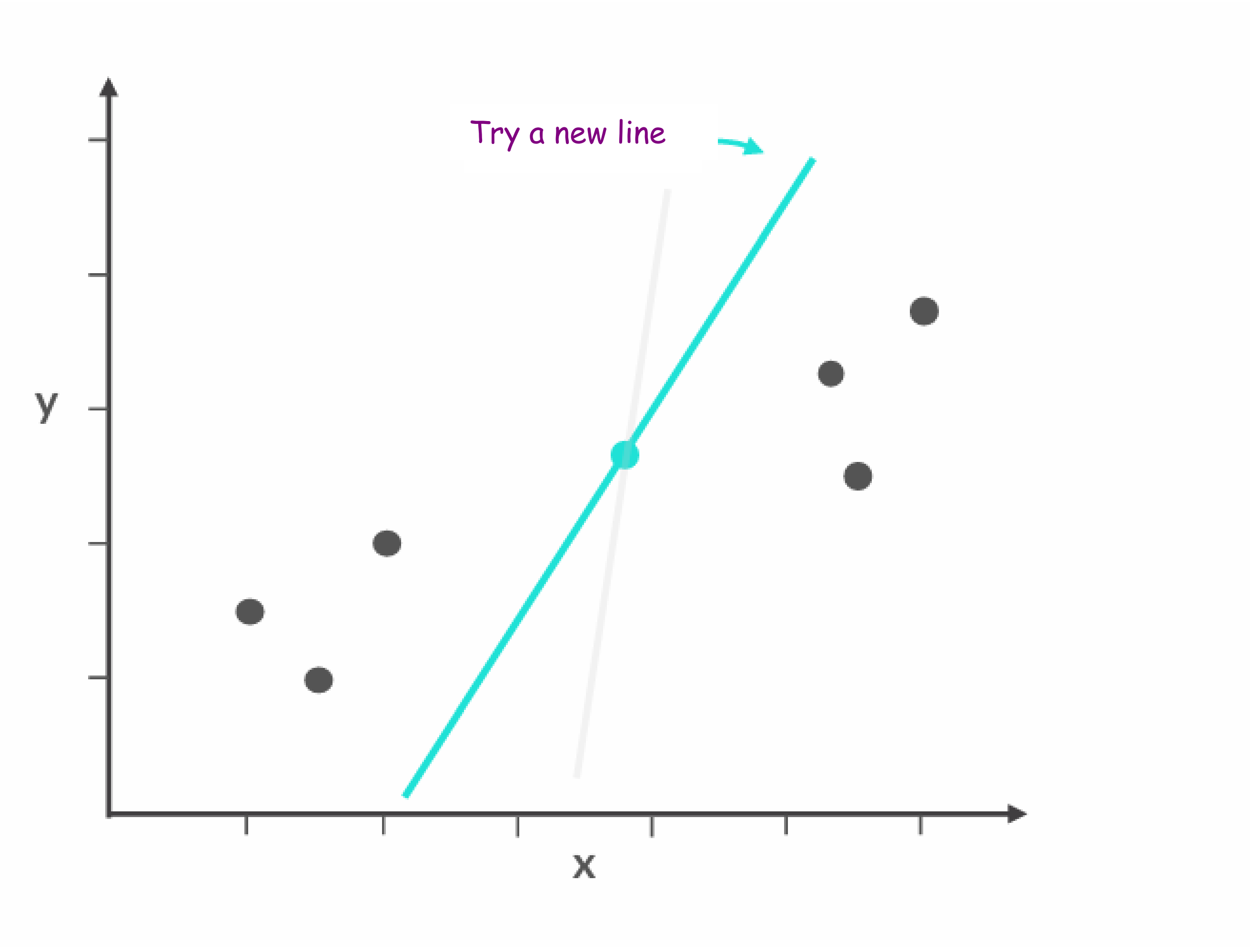
$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2 + d_7^2 + d_8^2 = 4.8$

Find the line that maximizes the sum of squared distances?

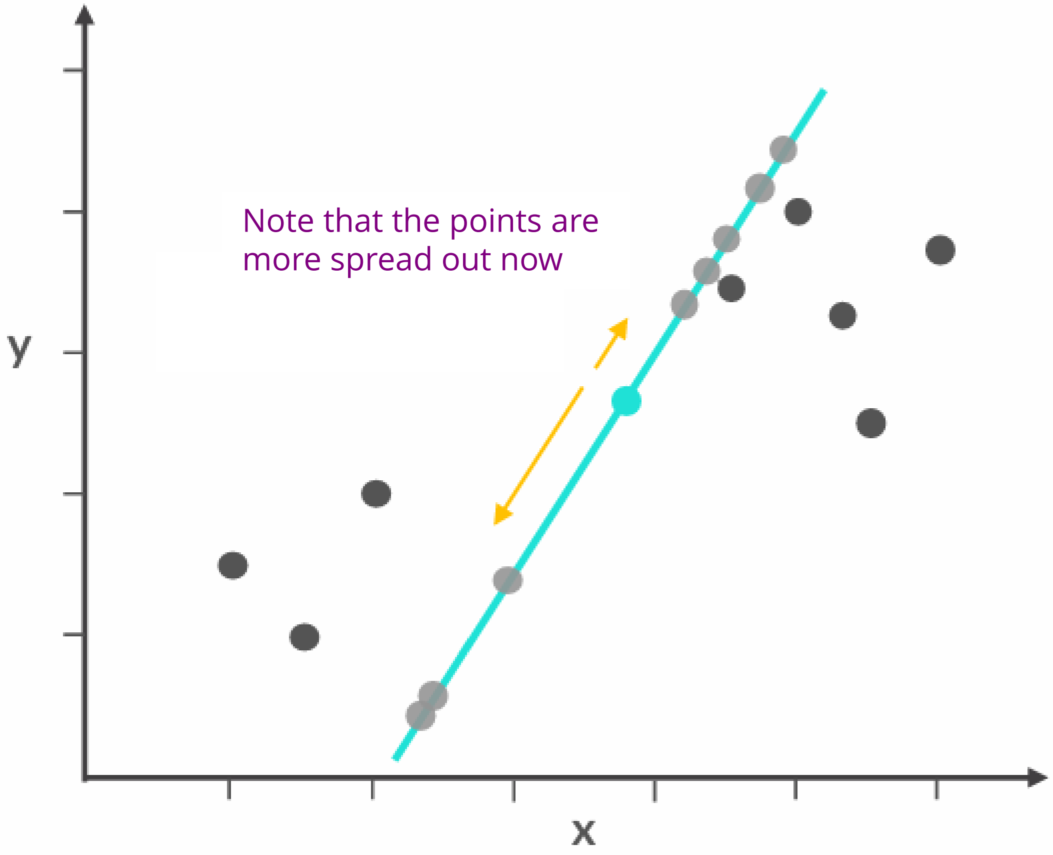
Why maximize the distance?

- **This guarantees that the principal component line captures the most variation in the data**

Step 2.12 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



Step 2.13 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)

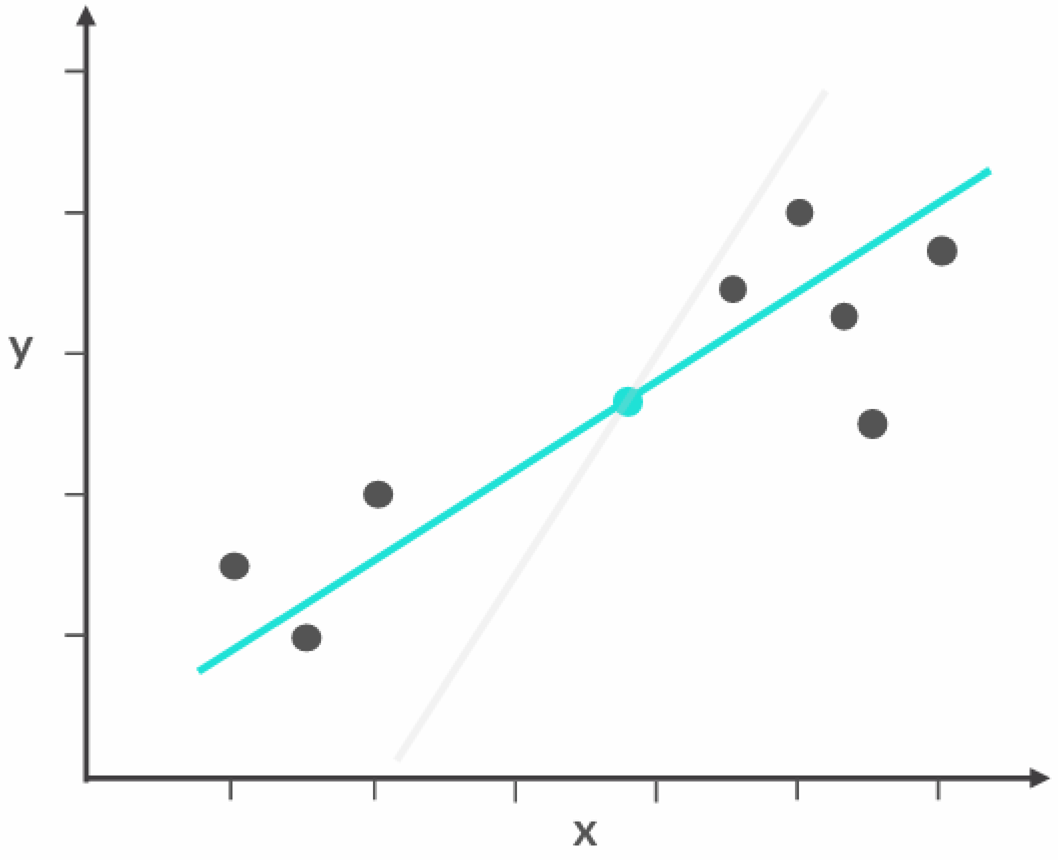


Note that the points are more spread out now

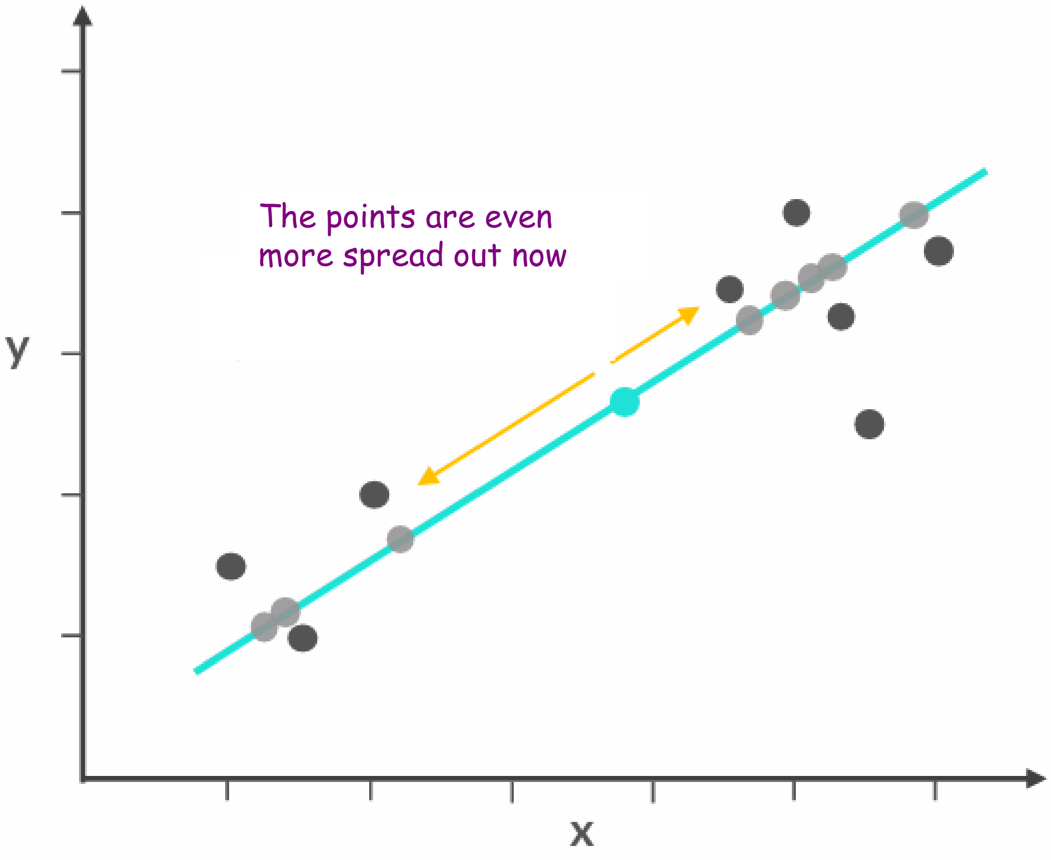
$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2 + d_7^2 + d_8^2 = 9.4$

The distance almost doubled!

Step 2.14 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



Step 2.15 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)

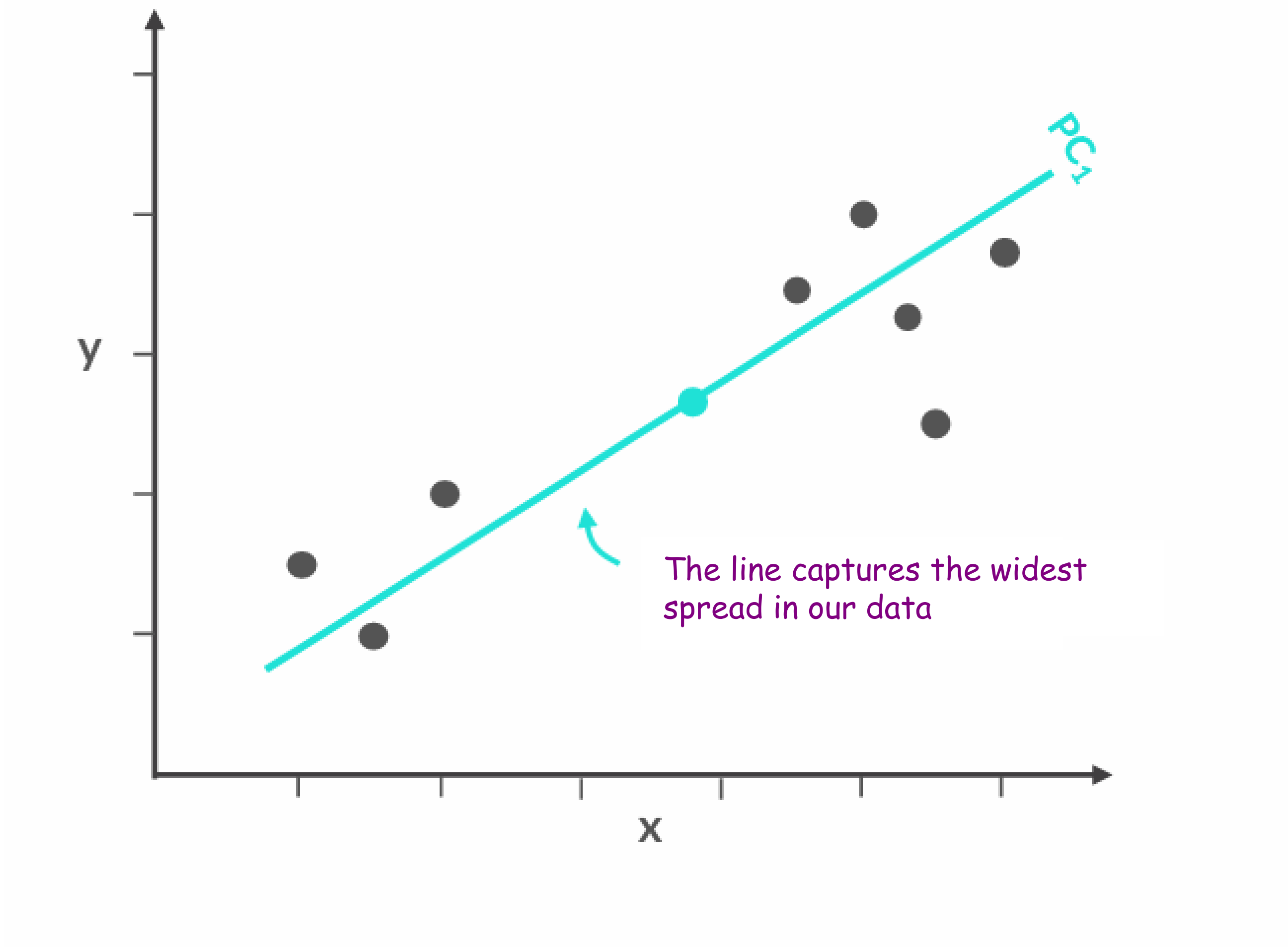


The points are even more spread out now

$$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2 + d_7^2 + d_8^2 = \mathbf{12.2}$$

This is the maximum sum of squared distances

Step 2.16 - Trace a line through the center that captures the most spread, or variation, in the data – this is the first principal component (PC1)



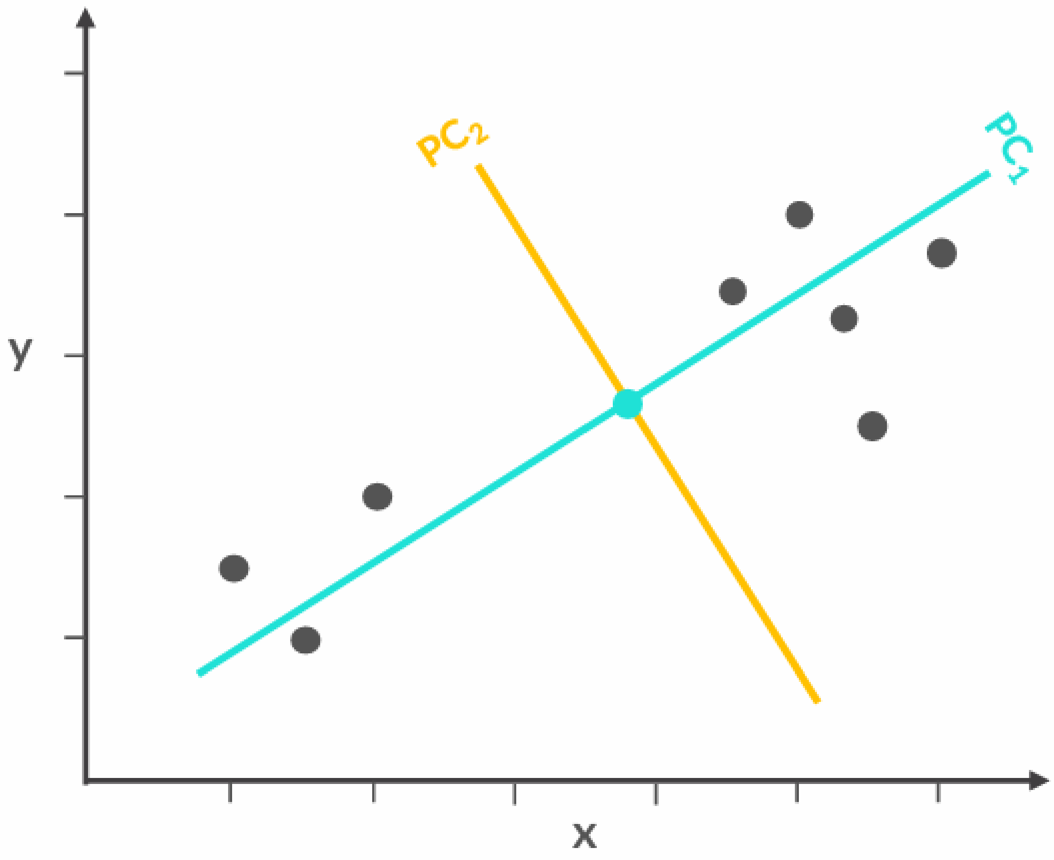
The line captures the widest spread in our data

Step 3 – Find Subsequent Principal Component

Step 3 – Find Subsequent Principal Component

After finding PC1, the algorithm finds the next best axis for capturing variance. This new axis, the **Second Principal Component (PC2)**, is always **orthogonal** (perpendicular) to the first.

Step 3.1 - Create another line perpendicular to the first one that captures the most spread, or variation, in the data – this is PC2

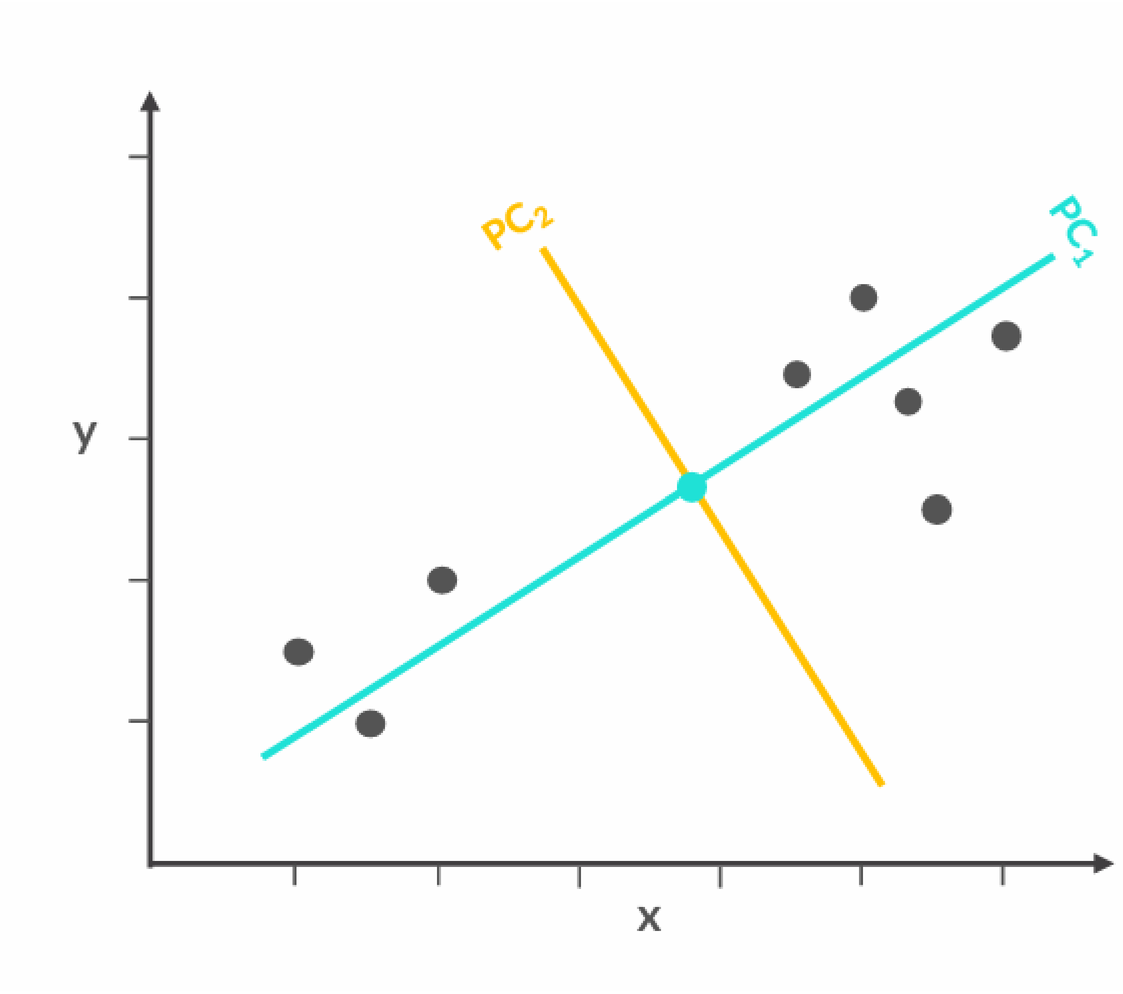


Step 4 – Calculate All Principal Components

Step 4 – Calculate All Principal Components

This step involves repeating the process until you have found a principal component for every original dimension. The result is a new set of orthogonal axes that are ordered by how much variance they capture.

Step 4.1 - Repeat Step 3 until you have as many PCs as original columns



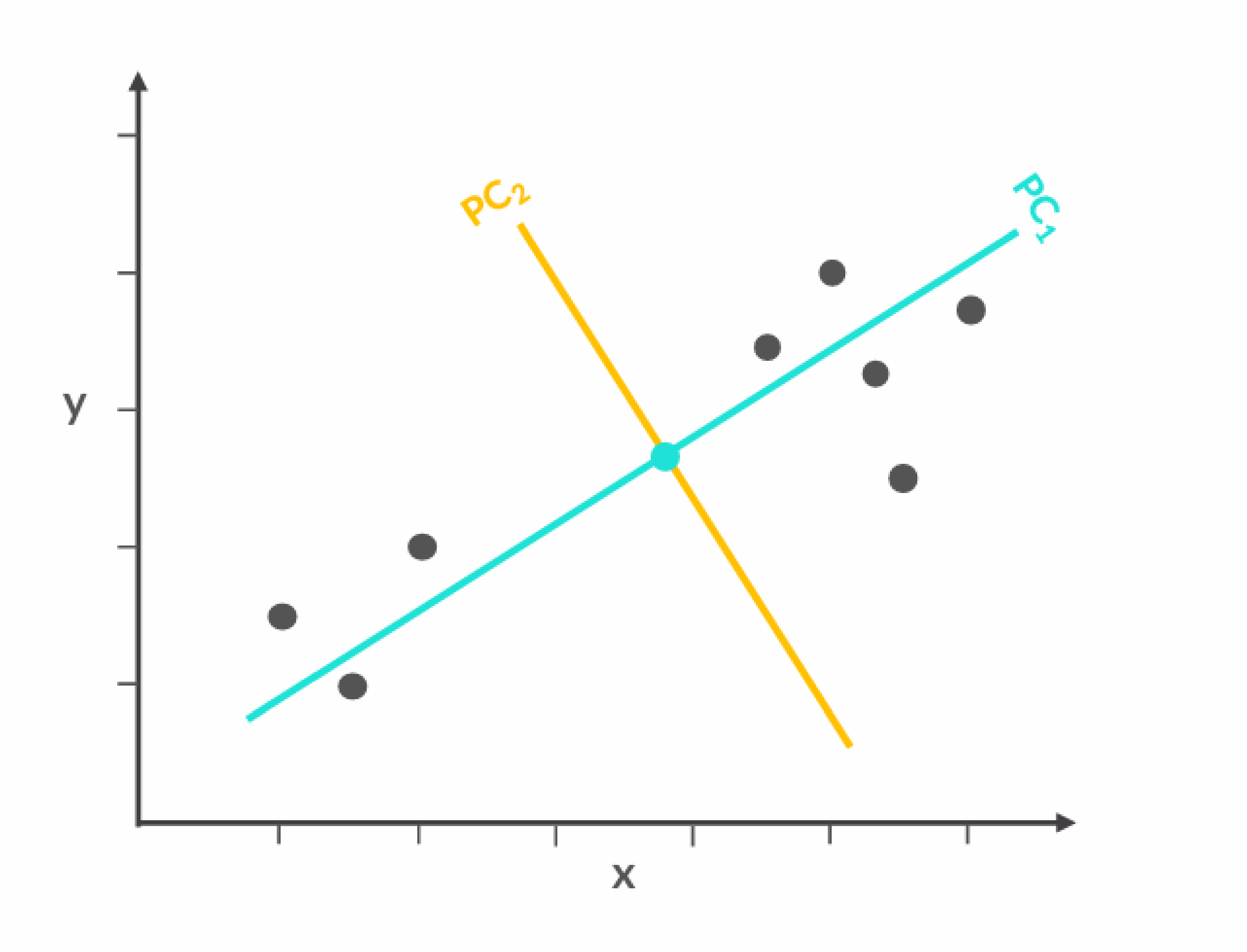
In practice, these steps are completed using a linear algebra technique called eigen decomposition

Step 5 - Dimensionality Reduction

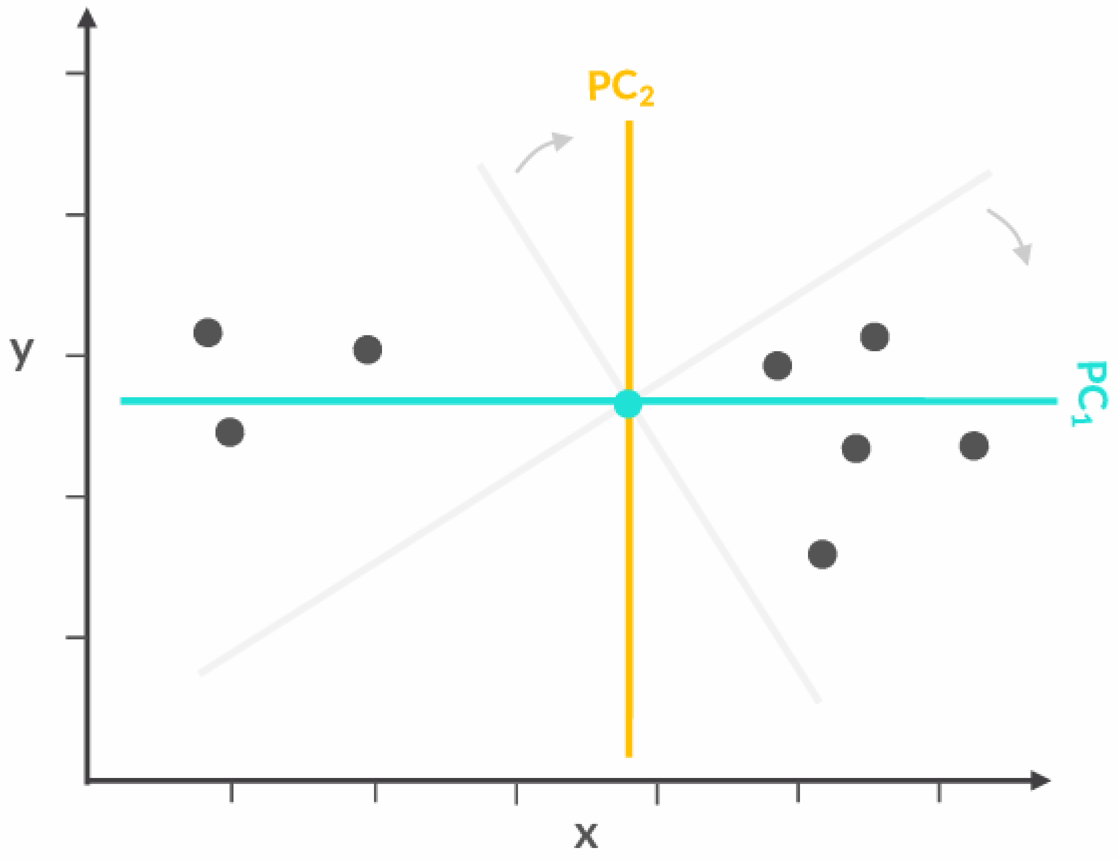
Step 5 – Dimensionality Reduction

In the final step, you choose a subset of the most important principal components (e.g., just PC1) to represent your data. By discarding the components that explain the least variance, you effectively reduce the dimensionality of your dataset.

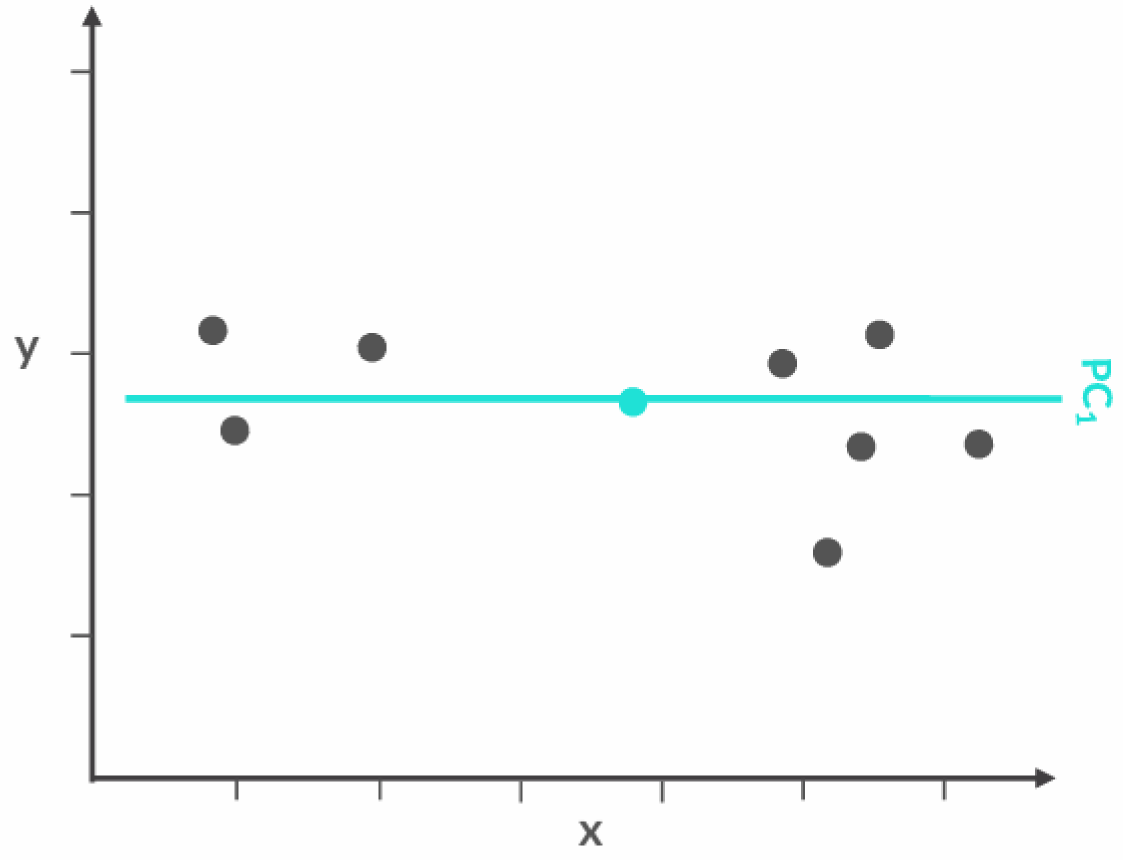
Step 5.1 - Transform the data into a new space with the PC1 line as the x-axis, the PC2 line as the y-axis, and so on



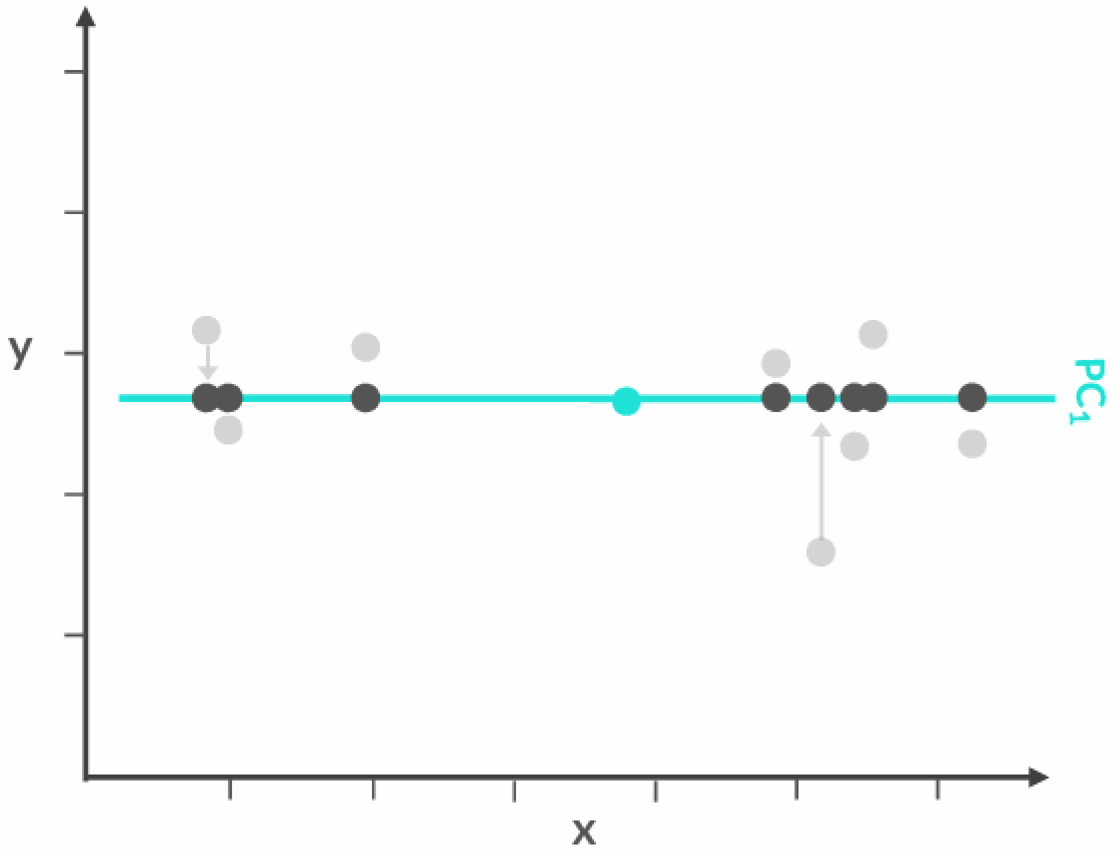
Step 5.2 - Transform the data into a new space with the PC1 line as the x-axis, the PC2 line as the y-axis, and so on



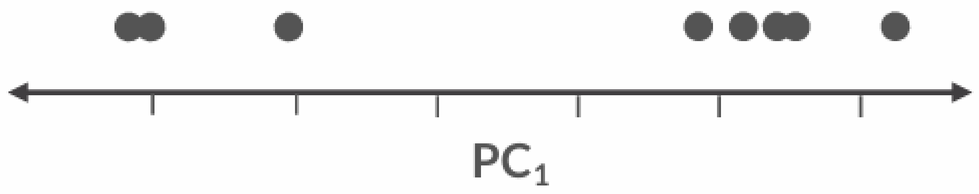
Step 5.3 - To reduce dimensions, keep a subset of principal components – for example, remove PC2 and project the points onto PC1, which is your new x-axis



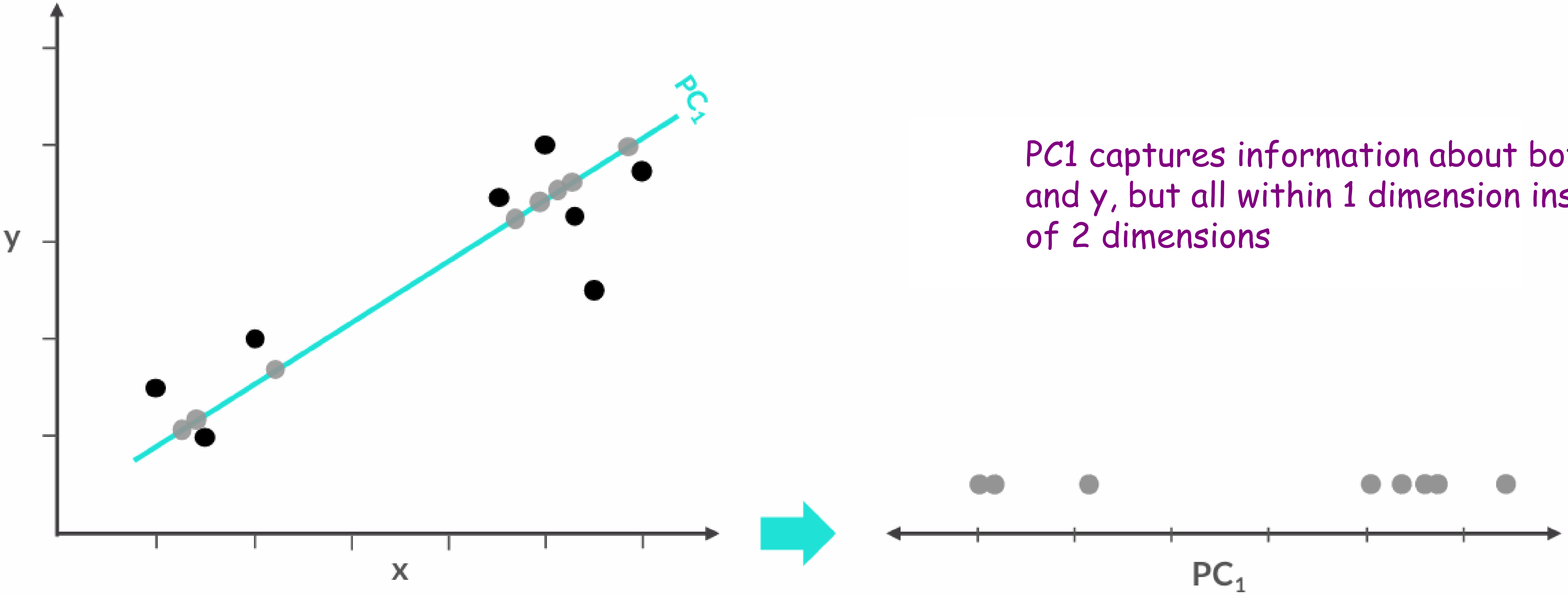
Step 5.4 - To reduce dimensions, keep a subset of principal components – for example, remove PC2 and project the points onto PC1, which is your new x-axis



Step 5.5 - To reduce dimensions, keep a subset of principal components – for example, remove PC2 and project the points onto PC1, which is your new x-axis

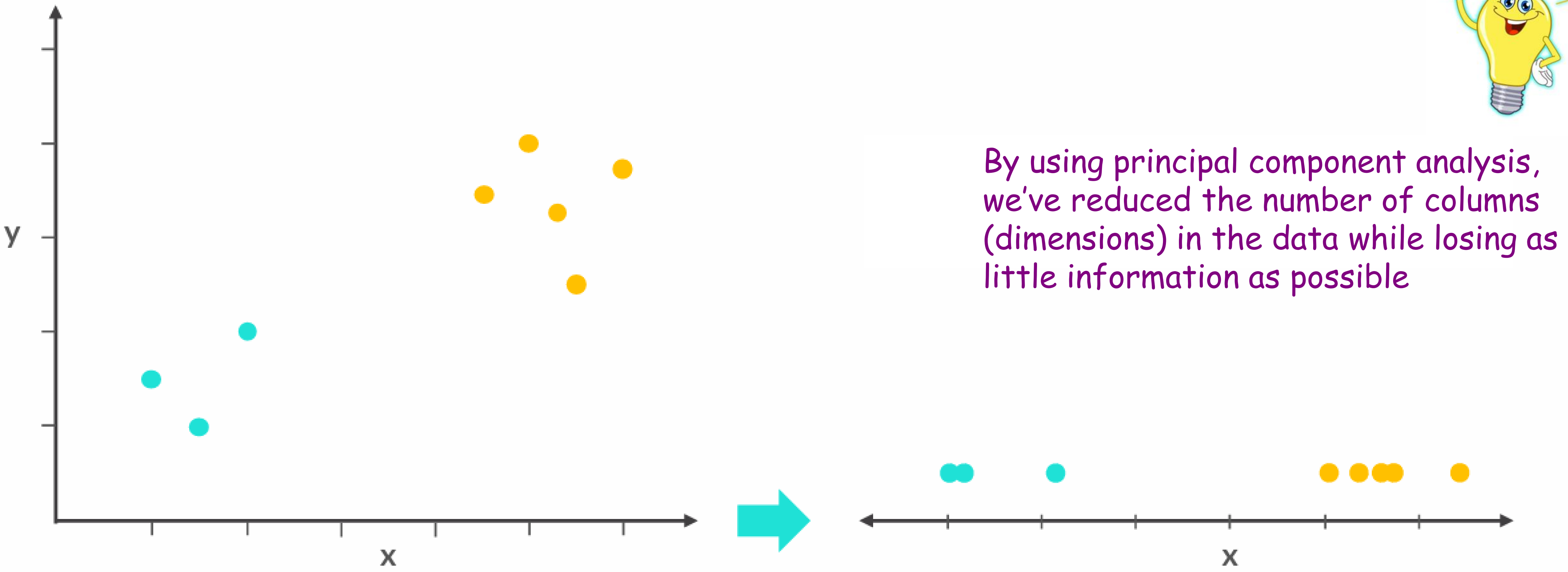


Step 5.6 - By using principal component analysis, we've reduced the number of columns (dimensions) in the data while losing as little information as possible



PC1 captures information about both x and y, but all within 1 dimension instead of 2 dimensions

Step 5.7 - By using principal component analysis, we've reduced the number of columns (dimensions) in the data while losing as little information as possible



By using principal component analysis, we've reduced the number of columns (dimensions) in the data while losing as little information as possible

Principal Component Analysis summarization

Principal Component Analysis (PCA) is a powerful **dimensionality reduction** technique used in machine learning. Its main goal is to simplify a dataset by transforming a large set of correlated variables into a smaller set of new, uncorrelated variables called **principal components**.

The Core Idea

PCA works by finding the directions (or axes) in the data that capture the most variance. The first principal component (PC1) is the direction with the highest variance, the second principal component (PC2) is the direction with the next highest variance, and so on. These new axes are always perpendicular to each other.

By keeping only a few of the top principal components, you can represent the data in a much lower dimension while retaining most of the important information. This is useful for visualization, speeding up machine learning algorithms, and reducing noise in the data.

LET’S GO