1. Given the following training data:

	y_1	y_2
x_1	0	0
x_2	4	0
x_3	2	1
x_4	6	3

(a) Compute the K-L transformation.

(b) What is the rotation applied to go from the original to the eigenvector coordinate system?

(c) Which eigenvector is most significant?

(d) Can we apply the Kaiser criterion?

(e) Map the points onto the most significant dimension.

The K-L transform is the **matrix** that rotates the coordinate system from the original one to the eigenvector one. It is given by the eigenvectors of the covariance matrix of the data. The most significant eigenvector is the one with the largest eigenvalue, of course. Moreover, the **Kaiser Criterion** tells us that we should keep only the eigenvectors with eigenvalues greater than 1.

Let's start by calculating the K-L transform of the given data. For starters, we need to calculate the covariance matrix (and, therefore, the mean) of the data.

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i = \frac{\left(\begin{bmatrix}0\\0\end{bmatrix} + \begin{bmatrix}4\\0\end{bmatrix} + \begin{bmatrix}2\\1\end{bmatrix} + \begin{bmatrix}6\\3\end{bmatrix}\right)}{4} = \begin{bmatrix}3\\1\end{bmatrix}$$

$$\Sigma = \frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu)(x_i-\mu)^T = \frac{1}{3}\left[\left(\begin{bmatrix}0\\0\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right)\left(\begin{bmatrix}0\\0\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right)^T + \cdots\right] = \begin{bmatrix}6.66667 & 2.66667\\ 2.66667 & 2\end{bmatrix}$$

In order to calculate the K-L transform, we need to calculate the eigenvectors of the covariance matrix. We can do this by determining the roots of the characteristic polynomial of the covariance matrix: its **eigenvalues**. As we have learned before, the eigenvalues are λ's such that, for a given non-zero vector v, we have that $\Sigma v = \lambda v$. In other words, we need to find the roots of the following polynomial:

$$\begin{aligned}\det(\Sigma - \lambda I) &= \det\left(\begin{bmatrix}6.66667 & 2.66667\\ 2.66667 & 2\end{bmatrix} - \lambda\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right)\\ &= (20/3 - \lambda)(2-\lambda) - (8/3)^2 = 0\end{aligned}$$

In the end, we're able to gather both eigenvalues, $\lambda_1 = 0.79 \vee \lambda_2 = 7.88$. As has been noted before, the Kaiser criterion tells us that we should keep only the eigenvectors with eigenvalues greater than 1. In this case, we should keep only the eigenvector associated with $\lambda_2 = 7.88$, v_2, discarding v_1.

We now want, of course, to find the eigenvectors of the covariance matrix (and normalize them afterwards). We can do this by solving the following system of equations:

$$
\begin{aligned}
\begin{cases} \Sigma v_1 = \lambda_1 v_1 \\ \Sigma v_2 = \lambda_2 v_2 \end{cases} &= \begin{cases} \begin{bmatrix} 6.66667 & 2.66667 \\ 2.66667 & 2 \end{bmatrix} v_1 = \lambda_1 v_1 \\ \begin{bmatrix} 6.66667 & 2.66667 \\ 2.66667 & 2 \end{bmatrix} v_2 = \lambda_2 v_2 \end{cases} \\
&= \begin{cases} \left(\begin{bmatrix} 6.66667 & 2.66667 \\ 2.66667 & 2 \end{bmatrix} - \lambda_1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = 0 \\ \left(\begin{bmatrix} 6.66667 & 2.66667 \\ 2.66667 & 2 \end{bmatrix} - \lambda_2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) \begin{bmatrix} v_{21} \\ v_{22} \end{bmatrix} = 0 \end{cases} \\
&= \cdots \\
&= \begin{cases} \begin{bmatrix} (20/3 - \lambda_1) v_{11} + 8/3 v_{12} \\ 8/3 v_{11} + (2 - \lambda_1) v_{12} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} (20/3 - \lambda_2) v_{21} + 8/3 v_{22} \\ 8/3 v_{21} + (2 - \lambda_2) v_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{cases}
\end{aligned}
$$

After performing some magical *continhas* with `numpy.linalg`'s `eig` method, we're able to find the following eigenvectors:

$$
v_1 = \begin{bmatrix} -0.413216 \\ 0.910633 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 0.910633 \\ 0.413216 \end{bmatrix}
$$

Note that these vectors have already been normalized (i.e, $v = \frac{v}{\|v\|}$).

The **K-L transform** is, then, given by:

$$
U_{K-L} = \begin{bmatrix} -0.413216 & 0.910633 \\ 0.910633 & 0.413216 \end{bmatrix}
$$

As a side note, we do this since we're trying to transform the original Σ into a diagonal matrix (utilizing Linear Algebra's $U^{-1}MU = D$ equality). This diagonal matrix is particularly relevant since we'll essentially want an equivalent matrix to Σ, but where every feature is "orthogonal" to each other: that way, we can look at each feature by itself, being then able to gather which are relevant or not.

To understand the rotation applied, we must first how the rotation concept applies to this context: we basically want to rotate the data so that the new axes are aligned with the eigenvectors of the covariance matrix. Mathematically, we can express this (considering e_k as the k-th canonical basis vector) as:

$$
\begin{aligned}
U_{K-L} e_1 &= v_1 \\
U_{K-L} e_2 &= v_2
\end{aligned}
$$

To gather the rotation angle, we can use an old notation from trigonometry:

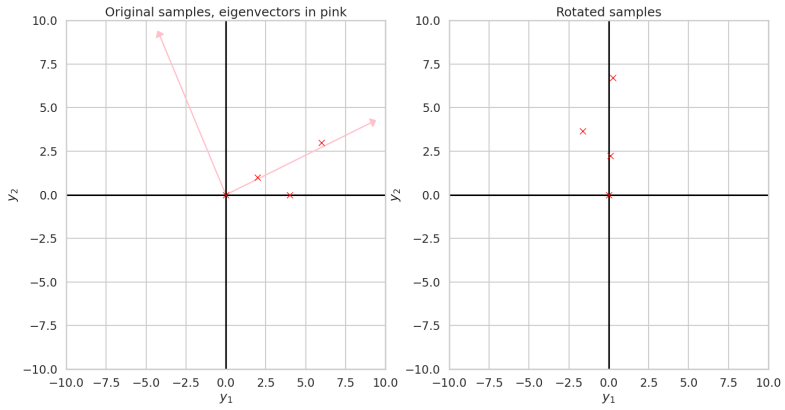
$$v_1 e_1 = \|v_1\|\|e_1\| \cos\theta \leftrightarrow \theta = \arccos\left(\frac{v_1 e_1}{\|v_1\|\|e_1\|}\right) = \arccos(-0.4138) \approx 114.4^\circ$$

In simpler terms, we're basically mapping the points to a new coordinate system, the **eigenspace**, where the eigenvectors are the new axes. The rotation angle is the angle between the original axes and the new ones. This mapping is done through $x' = U_{K-L}^T x$:

$$x' = U_{K-L}^T x = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} v_{11}x_1 + v_{21}x_2 \\ v_{12}x_1 + v_{22}x_2 \end{bmatrix}$$

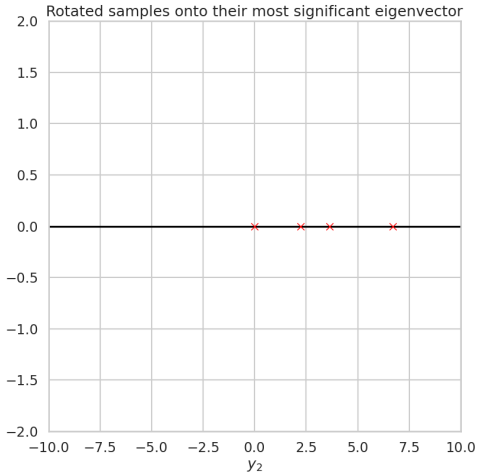
$$x'_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad x'_2 = \begin{bmatrix} -1.65287 \\ 3.64253 \end{bmatrix}, \quad x'_3 = \begin{bmatrix} 0.0842003 \\ 2.23448 \end{bmatrix}, \quad x'_4 = \begin{bmatrix} 0.252601 \\ 6.70345 \end{bmatrix}$$

Our mapping will, therefore, be something like this (note that the covariance associated with this new, "mapped dataset" would be diagonal):

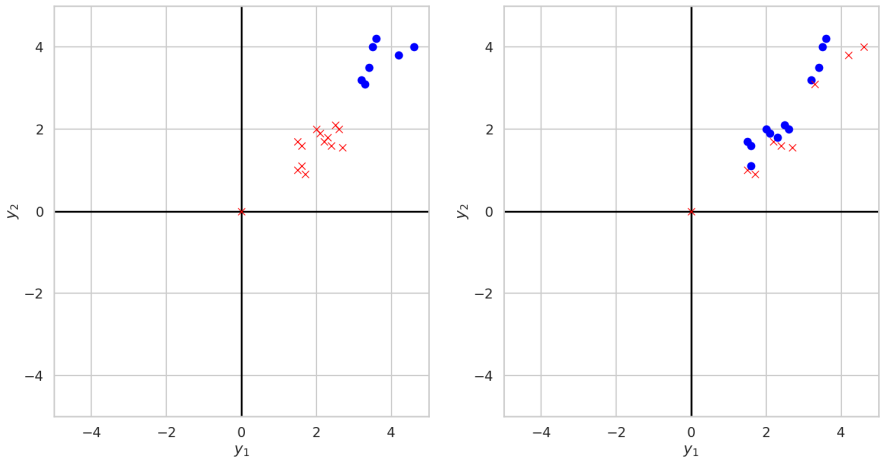


When talking about the **most significant eigenvector**, we can say that it is the eigenvector with the **largest associated eigenvalue** - largest eigenvalues are associated with eigenvectors along which data varies the most. In this case, we have that $\lambda_1 < \lambda_2$, hence the most significant eigenvector is v_2. Looking to the plots above we can note, as expected, that v_2 does look like it "fits the data the best".

Finally, we could also be interested in mapping the samples onto the dataset's **most significant dimension**: here, we would be interested in the projection of the samples onto the eigenvector v_2, with the greater eigenvalue. As such, we'd only consider the second coordinate of the previously calculated mapped samples, which would lead us to the following plot:



2. Given the following datasets where observations are in $\mathbb{R}^2$ and belong to one of two classes, which principal components can accurately discriminate between the class per dataset?

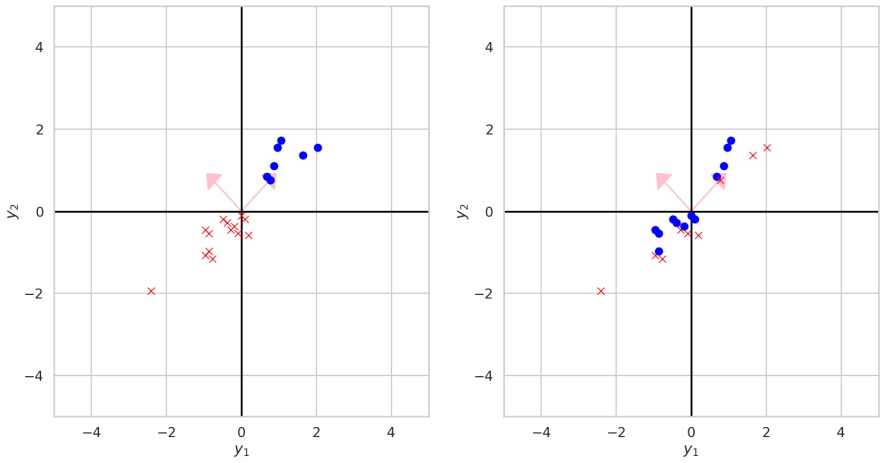


Note: the datasets used here are close, but not equal to the ones used in the original exercise. The logic applied, however, is the same.

Eigenvalues are directly related to the variance of the data along the eigenvectors. With that said, we can think about this problem as "how close are our points along the eigenvector" - with that put, we can see that along an $y = x$-ish slope, points vary way more than along an $y = -x$-ish slope. Therefore, with larger eigenvalues representing more variance, and considering that the eigenvectors do, in fact, follow those lines, $\lambda_1 > \lambda_2$.

With all this gathered, we'll be able to tell, and considering both these eigenvectors, that v_2 actually splits the data the best for the left-most plot, while v_1 does it for the right-most one.

Note, of course, how we'd be quick to discard v_2, with it showing a smaller variance, but in fact it would actually perform way better in certain scenarios! PCA doesn't, then, work well in *every* scenario.



3. The following top-7 eigenvalues explain 90% of the variation of dataset X:

$$\lambda_1 = 20, \quad \lambda_2 = 10, \quad \lambda_3 = 5, \quad \lambda_4 = 4, \quad \lambda_5 = 3, \quad \lambda_6 = 2, \quad \lambda_7 = 1$$

What is the most accurate information regarding X?

(a) X has less than 7 attributes.

(b) X has 7 attributes.

(c) X has more than 7 attributes.

(d) X has 11 attributes.

We know that the proportion of variance explained by a given feature is given by the quotient between its eigenvalue and the sum of all eigenvalues. In a similar manner, we can say that the n most significant features explain the proportion of variance given by the sum of their eigenvalues divided by the sum of all eigenvalues. Let's try to decode this exercise:

$$0.9 = \frac{\sum_{i=1}^{7} \lambda_i}{\sum_{i=1}^{n} \lambda_i} = \frac{45}{\sum_{i=1}^{n} \lambda_i}$$

Putting it in another way, an eigenvalue of 45 would be able to explain 90% of the variance of the dataset. As such, we can say that a unit-like eigenvalue, $\lambda_k = 1$, would be able to explain 2% of the variance. Considering that the least significant eigenvalue from the top-7 is 1, all other eigenvalues after it would be able to explain, at best, 2% of the variance. As such, we can say that, to explain all the variance in the dataset, we'll need **at least** $7 + \frac{100-90}{2} = 12$ features.

The fourth option is, then, the most accurate one.

4. Given a set of data points in $\mathbb{R}^3$, the following covariance matrix was obtained:

$$\Sigma = \begin{bmatrix} 91.43 & 171.92 & 297.99 \\ 171.92 & 373.92 & 545.21 \\ 297.99 & 545.21 & 1297.26 \end{bmatrix}$$

as well as the following eigenvectors retrieved:

$$v_1 = \begin{bmatrix} 0.2179 \\ 0.4145 \\ 0.8836 \end{bmatrix}, \quad v_2 = \begin{bmatrix} -0.2466 \\ -0.8525 \\ 0.4608 \end{bmatrix}, \quad v_3 = \begin{bmatrix} 0.9443 \\ -0.3183 \\ -0.0836 \end{bmatrix}$$

Please select the more complete answer:

(a) eigenvalue λ_1 is approximately 1626

(b) eigenvalue λ_2 is approximately 129

(c) eigenvalues λ_1 and λ_2 explain > 99% of the variation in the data

(d) all of the above

This is a rather straight forward answer. We must first calculate the eigenvalues, of course, from the characteristic polynomial:

$$\det(\Sigma - \lambda I) = 0$$

Performing several more *continhas* (which won't be here), we can see that the eigenvalues are:

$$\lambda_1 \approx 1626.5, \quad \lambda_2 \approx 129.0, \quad \lambda_3 \approx 7.1$$

Therefore, both a) and b) are correct. As for c), we can calculate the proportion of variance explained by the top-k eigenvalues as follows:

$$p_k = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i}$$

$$p_2 = \frac{1626.5 + 129.0}{1626.5 + 129.0 + 7.1} = 0.99597$$

We can, therefore, assert that c) is also correct, hence the correct option here is d).

5. Given the following dataset:

	y_1	y_2
x_1	1	-1
x_2	0	1
x_3	-1	0

and the corresponding eigenvectors and eigenvalues:

$$\lambda_1 = 3/2, \qquad \lambda_2 = 1/2$$
$$v_1 = \begin{bmatrix} 0.707107 \\ -0.707107 \end{bmatrix}, \qquad v_2 = \begin{bmatrix} 0.707107 \\ 0.707107 \end{bmatrix}$$

Transform the input data using PCA.

Given both eigenvectors, we can, of course, write the transformation matrix as:

$$U_{K-L} = \begin{bmatrix} 0.707107 & 0.707107 \\ -0.707107 & 0.707107 \end{bmatrix}$$

As it had been stated a few questions ago, the mapping from the original axes to the new ones is given by $x' = U_{K-L}^T x$. As such, we can rewrite the dataset as:

$$x_1' = \begin{bmatrix} 0.707107 & -0.707107 \\ 0.707107 & 0.707107 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1.41421 \\ 0 \end{bmatrix}$$
$$x_2' = \begin{bmatrix} 0.707107 & -0.707107 \\ 0.707107 & 0.707107 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -0.707107 \\ 0.707107 \end{bmatrix}$$
$$x_3' = \begin{bmatrix} 0.707107 & -0.707107 \\ 0.707107 & 0.707107 \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix} = \begin{bmatrix} -0.707107 \\ -0.707107 \end{bmatrix}$$

Below we can see the before and after of the transformation: