Now we show that the kernel trick can be applied to PCA.

In this lecture we set the means to zero \( \mu_{x_i} = 0 \). This is easy to arrange (by subtraction).

\[ \sum_{i=1}^{N} \phi(x_i) = 0 \]

The output of PCA is the projection of the data \( \{x_i: i \in \mathbb{N}\} \) onto the subspace defined by \( \{e_1, \ldots, e_N\} \), i.e., the coefficients \( (x_i \cdot e_1, x_i \cdot e_2, \ldots, x_i \cdot e_N) \), \( i \in \mathbb{N} \).

This depends on dot products, which suggests we can use the kernel trick if we replace \( x \) by \( \phi(x) \).

But what about the \( e_i \)'s? They are eigenvectors of the correlation function. How do they change if we replace \( x \) by \( \phi(x) \)?

To understand this, we need another way to compute the eigenvector of \( \Sigma = \frac{1}{N} \sum_{i=1}^{N} x_i x_i^T \).

Claim: The eigenvectors with non-zero eigenvalues can be expressed in form \( e_i = \frac{1}{\sqrt{\lambda_i}} x_i e_i \). The eigenvectors with zero eigenvalues are of the form \( e_i, \forall i \in \mathbb{N} \).

Proof: Suppose \( x_i \cdot e \neq 0 \), \( i \in \mathbb{N} \).

then \( e_i e = \frac{1}{N} \sum_{i=1}^{N} x_i (x_i \cdot e) = 0 \), hence \( e_i \) is an eigenvector with zero eigenvalue. This proves the last sentence.

The remaining eigenvectors must be orthogonal to the zero eigenvectors. Hence they are of form \( \sum_{i=1}^{N} \lambda_i e_i \).
(2) **Intuition:** Suppose we have data in 3-2D space spanned by axes $i, j, k$. Suppose all the data lies in the plane spanned by $i, j$.

\[
\begin{align*}
\mathbf{e}_i &= x_i \mathbf{i} + y_i \mathbf{j} \\
\mathbf{e}_j &= x_j \mathbf{i} + y_j \mathbf{j} \\
\mathbf{e}_k &= x_k \mathbf{i} + y_k \mathbf{j}
\end{align*}
\]

Then all vectors in the $i, j$ plane can be written as $y = \frac{y_i}{x_i} x_i \mathbf{i}$.

The eigenvectors of the correlation function lie in the $i, j$ plane, except for the zero eigenvalue with eigenvector $e = k$.

Now replace $x$ by $\Phi(x)$

\[
\mathbf{C} = \frac{1}{N} \sum_{n=1}^{N} \Phi(x_n) \Phi(x_n)
\]

All non-zero eigenvectors $\mathbf{e}$ of $\mathbf{C}$ are of form

\[
\mathbf{e} = \sum_{j=1}^{N} \alpha_j \Phi(x_j)
\]

**Substitution:** \( e = \frac{1}{e} \mathbf{e} \)

\[
\mathbf{C} \mathbf{e} = \lambda \mathbf{e}
\]

\[
\mathbf{C} = \frac{1}{N} \sum_{k=1}^{N} \Phi(x_k) \Phi(x_k) \mathbf{e}
\]

Equating coefficients of $\Phi(x_j)$ gives new eigenvalue equations

\[
\sum_{j} K(x_k, x_j) \alpha_j = \lambda \alpha_k
\]

**Index:** $n$
(3)

\[ \gamma N \sum_{j} K(x_k, x_j) x_j = \sum_{k} \lambda_k^\mu x_k \quad \mu = 1, \ldots, D. \]

Solving this, gives us the eigenvectors

\[ e_\mu = \sum_{j=1}^{N} \lambda_j^\mu \phi(x_j) , \text{ eigenvalue } \lambda_\mu. \]

But the projections \( e_\mu \cdot \phi(x) \) of the data are

\[ e_\mu \cdot \phi(x) = \sum_{j=1}^{N} \lambda_j^\mu K(x_k, x_j), \]

which is independent of \( \phi \)

(depending only on \( K \)).

Hence:

the projections of the data onto the eigenvectors require only knowing the kernel \( K(x_k, x_j) \) (i.e. not knowing \( \phi \))

Knowledge of the kernel is used twice:

1) to compute the \( \lambda_j^\mu \)
2) to compute the projections \( e_\mu \cdot \phi(x) \).