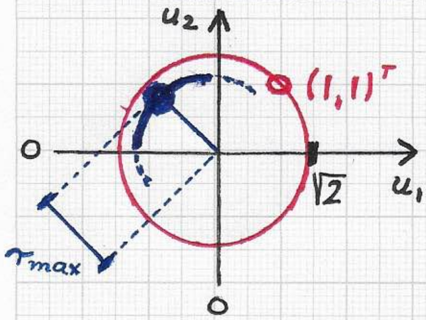
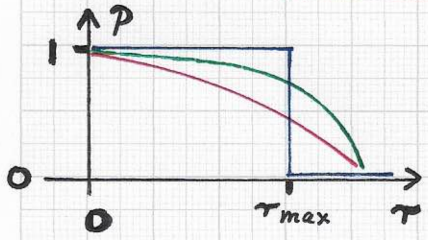
OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...



x.) All points u_i available as samples of the specific class of interest satisfy $(u_1^i)^2 + (u_2^i)^2 \leq 2$, as a consequence of the (minimal) bounding box construction in the (X_1, X_2)-plane and the applied linear coordinate transformation.

The figure (above) shows a point '●' $\in \{u_i\}$ that has maximal distance from the origin, considering all points in $\{u_i\}$. This distance is called r_{max}, and it is true that all points u_i satisfy $(u_1^i)^2 + (u_2^i)^2 \leq (r_{max})^2$.



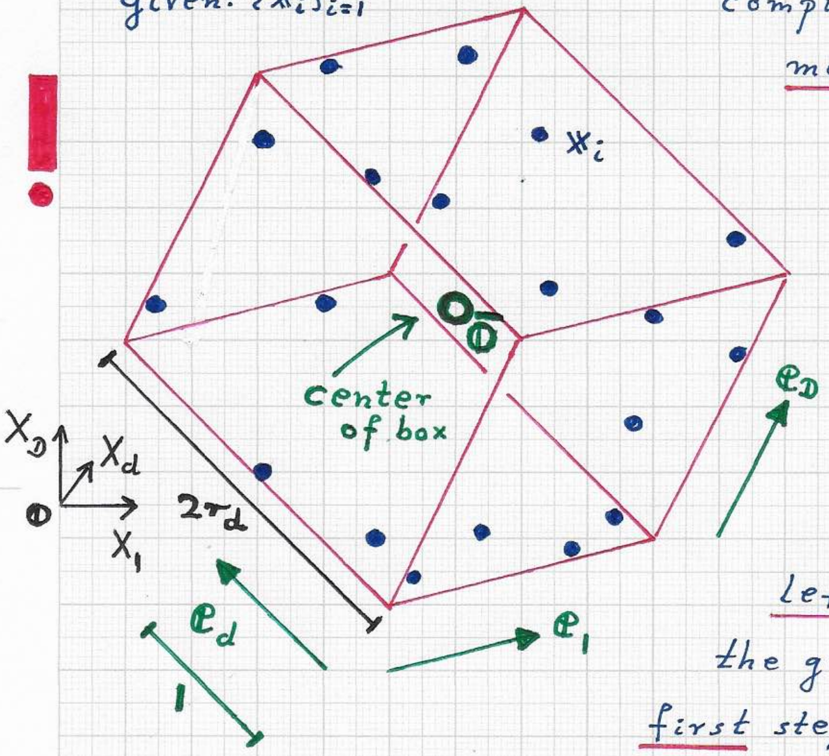
xi.) The left figure sketches possible p-functions (probabilities) that one could consider for the purpose of assigning a higher or lower value of "certainty" to a point u_i. A relatively higher value should/would indicate that a point u_i is more likely to represent the specific class of interest and represented by the point set $\{u_i\}$. For example, one could design, via experimentation with data, an "optimal" p-function that has the value 1 for $r = 0$ and monotonically decreases to the value 0 — very rapidly decreasing to 0 when $r > r_{max}$. (This concept could be beneficial when two point clouds, representing two classes, "overlap" in (X_1, X_2)-space. The r-value and a p-function might be helpful to improve classification in this situation.) ...

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

GENERAL CASE

• Laplacian eigenfunctions and neural networks:... We briefly describe the generalization of these concepts and computations for the D-dimensional case, where the given set of points $\{\mathbf{x}_i\}$ is embedded in $(X_1, \ldots, X_d, \ldots, X_D)$-space and point $\mathbf{x}_i$ has the associated coordinate tuple $(x_1^i, \ldots, x_d^i, \ldots, x_D^i)^T$. The left figure illustrates the general case. In the first step, we perform mean subtraction, by subtracting the mean/centroid/average from each point $\mathbf{x}_i$: (i) $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$; (ii) $\mathbf{x}_i \mapsto \mathbf{x}_i - \bar{\mathbf{x}}$.

given: $\{\mathbf{x}_i\}_{i=1}^{n}$



→ After mean subtraction, one can compute the eigenvalues and eigenvectors of the D-by-D covariance matrix $\mathbf{C}$ of all points '•'. The **normalized, unit-length eigenvectors define the D orthonormal eigendirections $\mathbf{e}_1, \ldots, \mathbf{e}_d, \ldots, \mathbf{e}_D$** shown in the above figure.

→ The eigendirections $\mathbf{e}_d = (e_1^d, \ldots, e_D^d)^T$, $d = 1 \ldots D$, define the gradients/gradient vectors of the linear polynomials $F_d(X_1, \ldots, X_D) = e_1^d X_1 + \ldots + e_D^d X_D$, $d = 1 \ldots D$.

• • •

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:... → We evaluate the linear polynomials F_d — which all have the function value 0 at $(X_1,...,X_D)=(0,...,0)$ — for each given $\mathbf{x}_i$, $i=1...n$, i.e., for each given point $\mathbf{x}_i$ **before** applying mean subtraction to the points' coordinate values. For each linear polynomial, i.e., for each direction/dimension d, we determine minimal and maximal values F_d^{min} and F_d^{max}. Specifically, we compute

$$F_d^{min} = \min\{F_d(x_1^i,...,x_D^i)\}_{i=1}^{n} \quad \text{and}$$
$$F_d^{max} = \max\{F_d(x_1^i,...,x_D^i)\}_{i=1}^{n} \quad .$$

→ **A** pair of contours $F_d=F_d^{min}$ and $F_d=F_d^{max}$ defines two boundary segments of the needed (minimal) bounding hyper-box, having **2D** boundary segments totally. We can compute the values of parameters defining the second, local coordinate system $\mathbf{S_2}$ as follows:

$$2r_d = F_d^{max} - F_d^{min} \quad ; \quad \overline{\mathbf{O}} = \frac{1}{2}\sum_{d=1}^{D}(F_d^{min}+F_d^{max})\,\mathbf{e}_d \, .$$

→ The local origin of the second system $(\overline{\mathbf{O}})$, the orthonormal eigendirections $(\mathbf{e}_d)$ and the "radii" (r_d) define the three (homogeneous) transformation matrices $\mathbf{S}^{-1}$, $\mathbf{R}^{-1}$ and $\mathbf{T}^{-1}$ that determine $\mathbf{M}^{-1}=\mathbf{S}^{-1}\mathbf{R}^{-1}\mathbf{T}^{-1}$:

$$\mathbf{T}^{-1} = \left[\begin{array}{c|c} \begin{matrix} \mathbf{1} & & 0 \\ & \ddots & \\ 0 & & \mathbf{1} \end{matrix} & \begin{bmatrix} \\ -\overline{\mathbf{O}} \\ \\ \end{bmatrix} \\ \hline 0 & \mathbf{1} \end{array}\right] , \quad \mathbf{R}^{-1} = \left[\begin{array}{c|c} \begin{matrix} [\ \mathbf{e}_1^T\ ] \\ \vdots \\ [\ \mathbf{e}_D^T\ ] \end{matrix} & 0 \\ \hline 0 & \mathbf{1} \end{array}\right] , \quad \mathbf{S}^{-1} = \left[\begin{array}{c|c} \begin{matrix} 1/r_1 & & 0 \\ & \ddots & \\ 0 & & 1/r_D \end{matrix} & 0 \\ \hline 0 & \mathbf{1} \end{array}\right] .$$

...

OBJECT AND MATERIAL EIGENFUNCTIONS - Cont'd.

• Laplacian eigenfunctions and neural networks: ... → The mapping defined by the matrix M^{-1} can be understood most simply by reviewing the special case of the two-dimensional case summarized on p. 15 (4/8/2023). Here,

$$_2\mathbb{x} = M^{-1}\,_1\mathbb{x} \iff \mathbb{u} = M^{-1}\mathbb{X} \,,$$

where $\mathbb{X} = (X_1, X_2)^T$ is a point's representation relative to the first, original coordinate system and $\mathbb{u} = (u_1, u_2)^T$ is the representation relative to the second coordinate system. Specifically, the mapping of the individual coordinate values is given as

$$u_1 = X_1\, e_1^1/r_1 + X_2\, e_2^1/r_1 - (t_1 e_1^1 + t_2 e_2^1)/r_1 \,,$$
$$u_2 = X_1\, e_1^2/r_2 + X_2\, e_2^2/r_2 - (t_1 e_1^2 + t_2 e_2^2)/r_2 \,.$$

One can shorten these expressions by using the notation for scalar (inner) products of vectors, $\langle \cdot , \cdot \rangle$:

$$u_1 = 1/r_1 \left(X_1 e_1^1 + X_2 e_2^1 - (t_1 e_1^1 + t_2 e_2^1) \right) = 1/r_1 \left(\langle \mathbb{X}, \mathbb{e}_1 \rangle - \langle \mathbb{t}, \mathbb{e}_1 \rangle \right) \,,$$
$$u_2 = 1/r_2 \left(X_1 e_1^2 + X_2 e_2^2 - (t_1 e_1^2 + t_2 e_2^2) \right) = 1/r_2 \left(\langle \mathbb{X}, \mathbb{e}_2 \rangle - \langle \mathbb{t}, \mathbb{e}_2 \rangle \right) \,.$$

Thus, the value of u_d in the **D**-dimensional case is

$$\boxed{u_d = 1/r_d \left(\langle \mathbb{X}, e_d \rangle - \langle \mathbb{t}, e_d \rangle \right)} \,, \; d = 1 \ldots D.$$

One can write this equation even more compactly as

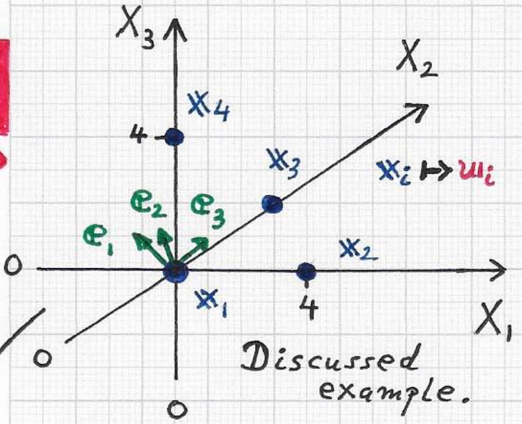
$$\boxed{u_d = \langle e_d, \mathbb{X} - \mathbb{t} \rangle / r_d} \,.$$

The translation vector $\mathbb{t}$ is the positional vector $\mathbb{t} = \overline{\Phi}$.

...

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:...



Discussed example.

• We discuss a simple three-dimensional example to clarify these steps. We are given four points x_i, $i=1...4$:

$x_1 = (0,0,0)^T$, $x_2 = (4,0,0)^T$, $x_3 = (0,4,0)^T$, $x_4 = (0,0,4)^T$.

The mean is $\bar{x} = 1/4 \sum_{i=1}^{4} x_i = (1,1,1)^T$. After mean subtraction, one obtains

$x_1 = (-1,-1,-1)^T$, $x_2 = (3,-1,-1)^T$, $x_3 = (-1,3,-1)^T$, $x_4 = (-1,-1,3)^T$.

These four points define the covariance matrix C,

$$C = \begin{bmatrix} -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 \\ 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{bmatrix} = \begin{bmatrix} 12 & -4 & -4 \\ -4 & 12 & -4 \\ -4 & -4 & 12 \end{bmatrix}.$$

The eigenvalues of C are $\lambda_1 = 16$, $\lambda_2 = 16$ and $\lambda_3 = 4$.

The corresponding normalized eigenvectors are $e_1 = (-\sqrt{2}/2, 0, \sqrt{2}/2)^T$, $e_2 = (-\sqrt{2}/2, \sqrt{2}/2, 0)^T$ and $e_3 = (\sqrt{3}/3, \sqrt{3}/3, \sqrt{3}/3)^T$. The figure above provides an illustration. The resulting linear polynomials are $F_1 = -\sqrt{2}/2\, X_1 + \sqrt{2}/2\, X_3$, $F_2 = -\sqrt{2}/2\, X_1 + \sqrt{2}/2\, X_2$ and $F_3 = \sqrt{3}/3\, X_1 + \sqrt{3}/3\, X_2 + \sqrt{3}/3\, X_3$. Evaluating these three linear polynomials for all points x_i (using their coordinates BEFORE mean subtraction) yields $F_1^{min} = -2\sqrt{2}$, $F_1^{max} = 2\sqrt{2}$; $F_2^{min} = -2\sqrt{2}$, $F_2^{max} = 2\sqrt{2}$; and $F_3^{min} = 0$, $F_3^{max} = 4\sqrt{3}/3$. The respective radius values are $2r_1 = 4\sqrt{2}$; $2r_2 = 4\sqrt{2}$; and $2r_3 = 4\sqrt{3}/3$. Further, the origin $\bar{\Phi}$ of the implied (minimal) bounding box is $\bar{\Phi} = \frac{1}{2} \sum_{d=1}^{3} (F_d^{min} + F_d^{max})\, e_d = (2/3, 2/3, 2/3)^T$. The translation vector $t = -\bar{\Phi}$, the unit eigendirections e_d and the radii r_d define the coordinate transformation $u_d = \langle e_d, X - t \rangle / r_d$.

$$\begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{3} & 0 & 1 \\ 0 & -\frac{1}{3} & 1 & 0 \\ -1 & -1 & -1 & -1 \end{bmatrix}$$

...