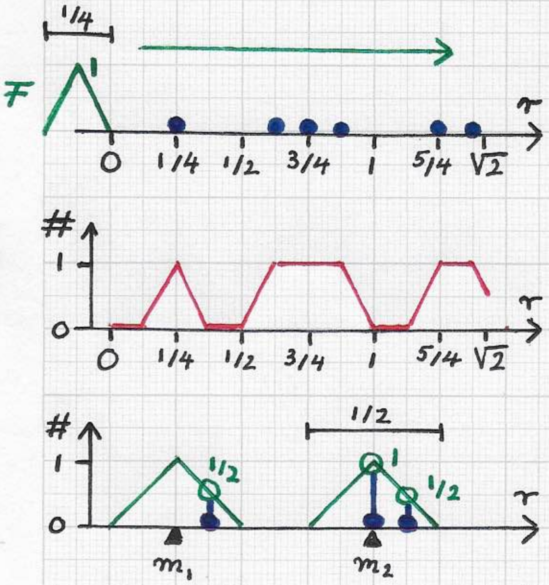
OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

- Laplacian eigenfunctions and neural networks: ...



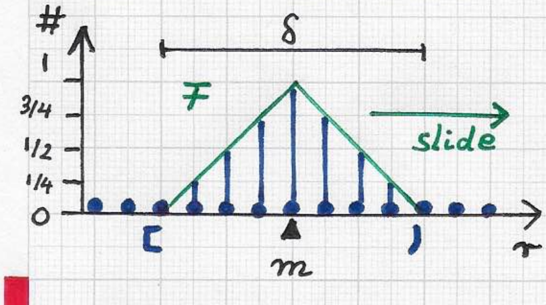
$\#(m_1) = 1/2$, $\#(m_2) = 1 + \frac{1}{2} = \frac{3}{2}$

The sketches shown here (left) illustrate an example for $D=2$ where a "hat function" of domain/window width 1/4 and peak value 1 is moved on the r-axis from left to right. The "hat function" serves as the filter F that is used for convolution with the set {•}. The third sketch explains the essential calculation of the value of the function called '#': We determine all r-values (•) inside the window of the (sliding) "hat function" and the values of F, i.e., the "hat function", for these r-values; and we sum up the resulting values of the "hat function" and assign the sum as value of # at the current location of the midpoint m_i. Sliding the filter F over the chosen set {•} of 6 r-values produces the piecewise linear function shown in the second sketch. Our goal is the estimation of probability density function values with respect to the D-dimensional domain space in which the actual points reside. We are not interested in densities relative to the r-axis.

...

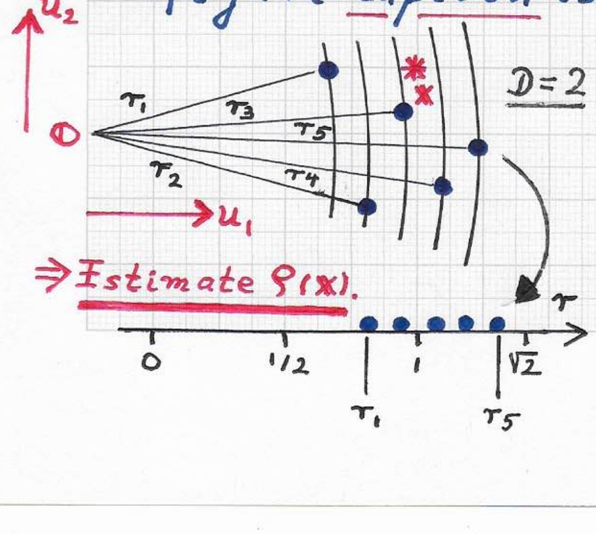
■ OBJECT AND MATERIAL EIGENFUNCTIONS - Cont'd.

• Laplacian eigenfunctions and neural networks: ...



$\#(m) = 0 + 1/4 + 1/2 + 3/4 + 1 + 3/4 + 1/2 + 1/4 + (0) = 3.$

While BOX function and HAT function filter masks F are simple and can be implemented efficiently, many functions can be considered to serve as a "counting" filter function F used in the convolution process. For example, normalized B-spline basis functions of any degree can be considered as they have many desirable properties one wants F to have, including non-negativity and partition of unity. The top-left figure illustrates how F serves as a "weight function": An r-value (•) lying in that part of F's domain where F has a positive value is assigned this positive value as its weight. The computation of the value of #(m) in the figure caption is an example. To summarize the most important aspects of this discussion, we must (1) choose a filter F; (2) define the "width" δ of F; and (3) normalize the resulting #-values to serve as proper values supporting the computation of probability density function values.

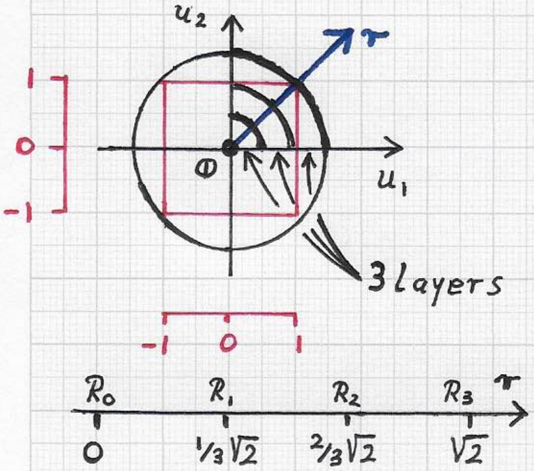


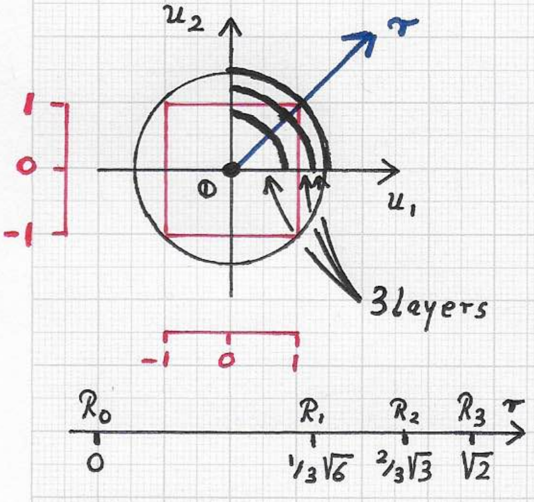
⇒ Estimate $\rho(x)$.

•••

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:...





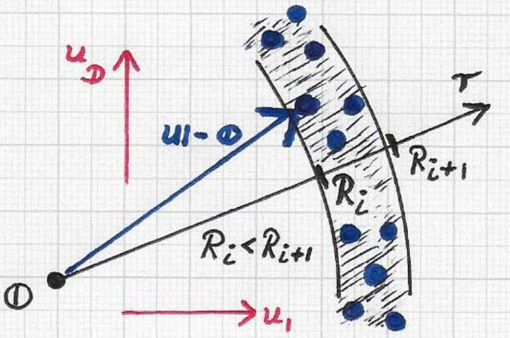
Two layer models for ρ function.

We re-consider the issue of using R_i-values, $i = 0 \ldots N$, such that $|R_{i+1} - R_i| = \text{const}$ or all layer hyper-volumes in the $\mathbf{u}$-space domain have the same hyper-volume. This issue is relevant for the calculation of proper probability density values – relative to $\mathbf{u}$-space and NOT relative to the radial r-line. We consider a simple example for $D = 2$ and $N = 3$. The top-left figure sketches the first case: The maximal 2-ball hyper-volume is $V = 2\pi$. The equidistantly spaced R_i-values are $R_i := \frac{i}{3}\sqrt{2}$. The hyper-volume of the i^{th} layer, V_i, is $V_i = (R_{i+1}^2 - R_i^2)\pi$, i.e., $V_0 = \frac{2}{9}\pi$, $V_1 = \frac{2}{3}\pi$ and $V_2 = \frac{10}{9}\pi$. ($\sum V_i = V$.)

The next figure sketches the second case: Each layer must have the same hyper-volume $V_i = \frac{2}{3}\pi$. The R_i-values are given as $R_i = \sqrt{i/3}\,\sqrt{2}$, i.e., $R_0 = 0$, $R_1 = \frac{1}{3}\sqrt{6}$, $R_2 = \frac{2}{3}\sqrt{3}$ and $R_3 = \sqrt{2}$. (See p. 8 (5/27/2023).)

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...



If one defines point (•) density only relatively to the one-dimensional r-line, then the probability density estimate for the interval $[R_i, R_{i+1})$ will be $\varrho_i = n_i / (R_{i+1} - R_i)$, where n_i is the number of u-points with positional vector $(\mathrm{u} - \mathrm{o})$ that satisfy the layer condition $R_i^2 \leq \|\mathrm{u} - \mathrm{o}\|^2 < R_{i+1}^2$, see top-left figure. Considering the actual hyper-volume of the u-space layer sketched in the figure for $D=2$, the probability density estimate is ${}^2\varrho_i = n_i / (\pi (R_{i+1}^2 - R_i^2))$. For the case $D=3$, one obtains the density estimate ${}^3\varrho_i = n_i / (\frac{4}{3}\pi (R_{i+1}^3 - R_i^3))$. For the general D-dimensional case, one obtains two formulae, see p. 9 (4/25/2023):

$$ {}^D\varrho_i = n_i \Big/ \left(\frac{1}{(D/2)!} \pi^{D/2} (R_{i+1}^D - R_i^D) \right), \; D = 0, 2, 4, \ldots, $$

$$ {}^D\varrho_i = n_i \Big/ \left(\frac{2^{\lceil D/2 \rceil}}{D!!} \pi^{\lfloor D/2 \rfloor} (R_{i+1}^D - R_i^D) \right), \; D = 1, 3, 5, \ldots . $$

Thus, a "D-dimensional density ${}^D\varrho_i$" is a certain multiple, scaled version of $\varrho_i = {}^1\varrho_i$. By computing ϱ_i, one can obtain ${}^D\varrho_i$, $D \in \{2, 3, 4, \ldots\}$, via multiplication.

•••

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

- Laplacian eigenfunctions and neural networks: ... More compactly, we can write a "D-dimensional density ${}^D\varrho_i$" as ${}^D\varrho_i = n_i / (c_D (R_{i+1}^D - R_i^D))$, where c_D is the factor that depends on the dimension D. We can express ϱ_i in terms of ${}^D\varrho_i$ (and vice versa):

$$\frac{\varrho_i}{{}^D\varrho_i} = \frac{n_i}{R_{i+1} - R_i} : \frac{n_i}{c_D (R_{i+1}^D - R_i^D)} = c_D \frac{R_{i+1}^D - R_i^D}{R_{i+1} - R_i}.$$

It is possible to symbolically compute the term $(R_{i+1}^D - R_i^D)/(R_{i+1} - R_i)$ exactly. We obtain:

$D = 1$: $(R_{i+1} - R_i)/(R_{i+1} - R_i) = 1$

$D = 2$: $(R_{i+1}^2 - R_i^2)/(R_{i+1} - R_i) = R_{i+1} + R_i$

$D = 3$: $(R_{i+1}^3 - R_i^3)/(R_{i+1} - R_i) = R_{i+1}^2 + R_{i+1} R_i + R_i^2$

$D = 4$: $(R_{i+1}^4 - R_i^4)/(R_{i+1} - R_i) = R_{i+1}^3 + R_{i+1}^2 R_i + R_{i+1} R_i^2 + R_i^3$

$$\Rightarrow (R_{i+1}^D - R_i^D)/(R_{i+1} - R_i) = R_{i+1}^{D-1} + R_{i+1}^{D-2} R_i + R_{i+1}^{D-3} R_i^2 + \ldots + R_{i+1} R_i^{D-2} + R_i^{D-1}$$

$$= \sum_{j=0}^{D-1} R_{i+1}^{D-1-j} R_i^j = \Sigma_D .$$

It is now possible to express ϱ_i as

$$\boxed{\varrho_i = {}^D\varrho_i \, c_D \, \Sigma_D \quad \Leftrightarrow \quad {}^D\varrho_i = \varrho_i / (c_D \Sigma_D)}.$$

We consider a simple example for the case $D=2$ and $R_i = i/4\sqrt{2}$, $i=0\ldots4$ (equidistant spacing of R_i-values). Here, ${}^2\varrho_i = \varrho_i / (\pi (R_{i+1} + R_i))$, $i=0\ldots3$. Further, $(R_{i+1} + R_i) = (i+1)/4\sqrt{2} + i/4\sqrt{2} = (2i+1)/4\sqrt{2}$.

...