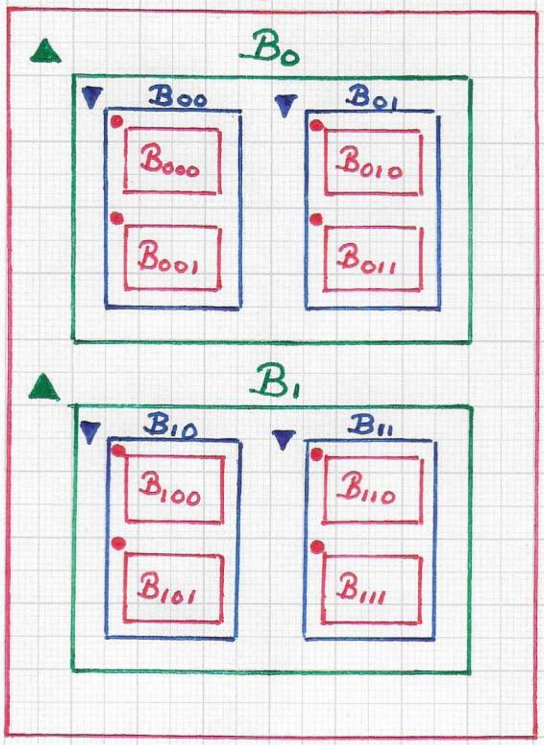
OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

- Laplacian eigenfunctions and neural networks: ...

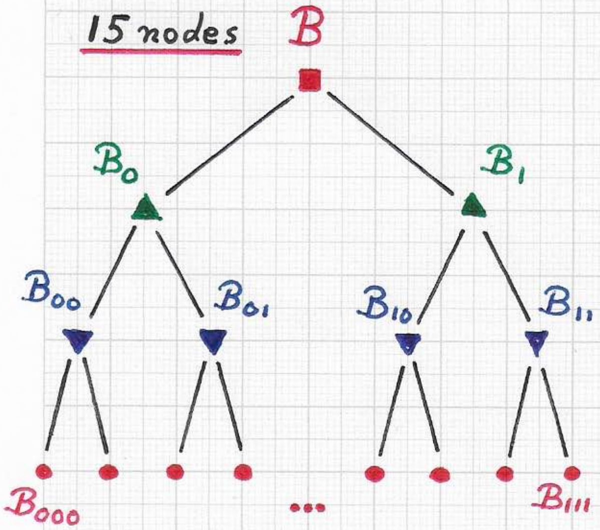
We briefly consider a geometrically "perfect nesting" example, where box splitting is performed three times.



	■	▲ 0	▲ 1	▼ 00	▼ 01	▼ 10	▼ 11	● 000	● 001	● 010	● 011	● 100	● 101	● 110	● 111	
0	0	·	·	·	·	·	·	·	·	·	·	·	·	·	·	← outside B
1	1	·	·	·	·	·	·	·	·	·	·	·	·	·	·	
2	1	1	0	·	·	·	·	·	·	·	·	·	·	·	·	
3	1	0	1	·	·	·	·	·	·	·	·	·	·	·	·	
4	1	1	0	1	0	·	·	·	·	·	·	·	·	·	·	
5	1	1	0	0	1	·	·	·	·	·	·	·	·	·	·	
6	1	0	1	·	·	1	0	·	·	·	·	·	·	·	·	
7	1	0	1	·	·	0	1	·	·	·	·	·	·	·	·	
8	1	1	0	1	0	·	·	1	0	·	·	·	·	·	·	15 cases
9	1	1	0	1	0	·	·	0	1	·	·	·	·	·	·	
10	1	1	0	0	1	·	·	·	·	1	0	·	·	·	·	
11	1	1	0	0	1	·	·	·	·	0	1	·	·	·	·	
12	1	0	1	·	·	1	0	·	·	·	·	1	0	·	·	
13	1	0	1	·	·	1	0	·	·	·	·	0	1	·	·	
14	1	0	1	·	·	0	1	·	·	·	·	·	·	1	0	
15	1	0	1	·	·	0	1	·	·	·	·	·	·	0	1	· = 0

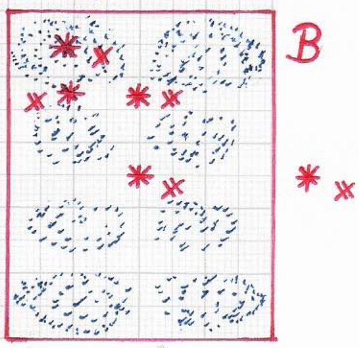
Traversal of tree for classification.

15 nodes



Binary box tree obtained after splitting three times.

The illustrations and table provided on this page present the same essential information summarizing the repeatedly applied box splitting – applied to a given point set that is sketched qualitatively in the right figure.



Possible locations of point *.

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks ... The figure in the bottom-right corner of the previous page shows a point set in 2D space — that can be thought of as a set of samples belonging to / representing a specific material class. The largest bounding box $\mathcal{B}$, while tightly enclosing the point set, is not sufficient. One can clearly see eight distinct point clusters inside $\mathcal{B}$, with "substantial empty space" between the clusters. Therefore, when calculating the proposed "tightness" measure T, and considering the defined value of a minimally required T-value, the box $\mathcal{B}$ most likely will be subject to multiple, repeated binary subdivision steps. (More precisely, the point set will be subject to multiple, repeated binary subdivision steps, generating sub-boxes $\mathcal{B}_0$, $\mathcal{B}_1$, $\mathcal{B}_{00}$, ..., $\mathcal{B}_{11}$, $\mathcal{B}_{000}$, ..., $\mathcal{B}_{111}$.) Since "perfect nesting" is most important for the majority of practically arising cases, this detailed example sketched and represented in various "equivalent ways" on the previous page emphasizes the relationships between (i) the point set exhibiting cluster characteristics (bottom-right figure); (ii) the cluster-induced geometrical hierarchy of bounding boxes / box regions (top-left illustration); (iii) the abstract binary tree data structure representing the hierarchy (bottom-left sketch); and (iv) a "traversal table" (top-right) for classification. •••

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

- Laplacian eigenfunctions and neural networks:... The "traversal table" on p. 11 (7/18/2023) defines how the 15-node tree used in the example is traversed to calculate a class-match probability for a to-be-classified point. The $16 = 2^4$ rows of the "traversal table" — numbered from 0 to 15 — show all possible traversal cases for this "perfect nesting" scenario. We consider some of the 16 cases defined by the rows in the "traversal table" in more detail.

Row 0: Point $*$ is outside box B; $P(*) = 0$.

Row 1: Point $*$ is inside box B AND outside box B_0 AND outside box B_1; $P(*) = 0$.

Row 2: Point $*$ is inside box B AND inside box B_0 AND outside box B_{00} AND outside box B_{01}; $P(*) = 0$.

Row 4: Point $*$ is inside box B AND inside box B_0 AND inside box B_{00} AND outside box B_{000} AND outside box B_{001}; $P(*) = 0$.

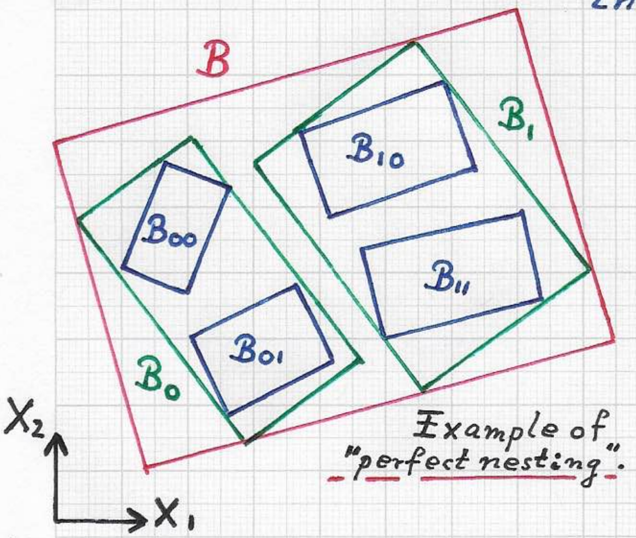
Row 8: **Point** $*$ is inside box B AND inside box B_0 AND inside box B_{00} AND inside box B_{000}; $P(*) = P_{000}$.

In summary, it is only possible to potentially obtain a value $P(*) > 0$ when the to-be-classified point $*$ lies inside box B_{000}, B_{001}, B_{010}, ... or B_{111}.

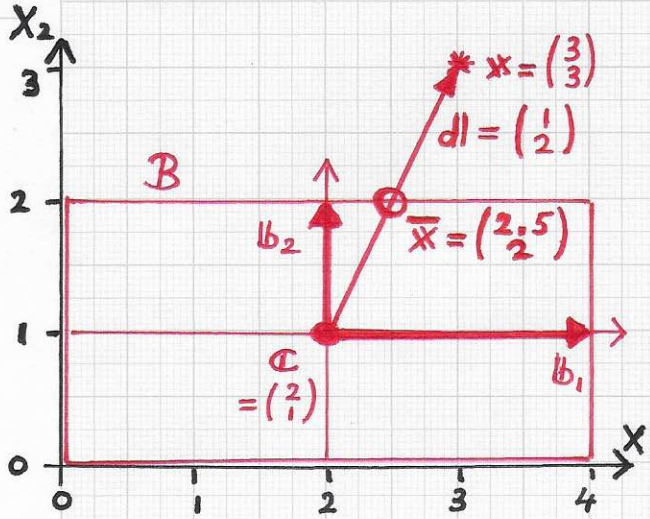
...

OBJECT AND MATERIAL EIGENFUNCTIONS - Cont'd.

- **L**aplacian eigenfunctions and neural networks:...



Example of "perfect nesting".



Since it is — in principle — possible that the geometrical conditions for a "perfect nesting" of a box hierarchy are not fully satisfied, one could consider the design and use of a box manipulation algorithm. The left figure shows an example of a "perfect nesting" where all subset/superset and non-intersection conditions are satisfied in the entire **7-box** hierarchy. For example, if one of the box corner vertices of B_0 lied outside B, one could employ a scaling operation "moving" that corner vertex of B_0 onto the box boundary of B. The second figure on this page refers to such a corner vertex outside B as $\ast$. Box B used in this example has $\mathbb{c} = (2,1)^T$ as its center and $\mathbb{b}_1 = (2,0)^T$ and $\mathbb{b}_2 = (0,1)^T$ as its orthogonal box basis vectors. The point $\ast$ has the global coordinate tuple $(X_1, X_2)^T = (3,3)^T$; its coordinate tuple relative to the local system $\{\mathbb{c}, \mathbb{b}_1, \mathbb{b}_2\}$ is $\ast_{loc} = (1/2, 2)^T$. ...

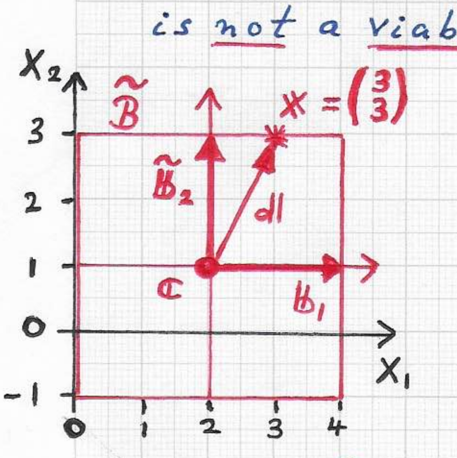
■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:... The coordinate tuple for $\mathbb{x}_{loc}$ is obtained by projecting the vector $\mathbb{d} = \mathbb{x} - \mathbb{c} = (3,3)^T - (2,1)^T = (1,2)^T$ onto the two orthogonal basis vectors $\mathbb{b}_1$ and $\mathbb{b}_2$, i.e.,

$$\langle \mathbb{d}, \mathbb{b}_1 \rangle / \|\mathbb{b}_1\|^2 = \langle (1,2)^T, (2,0)^T \rangle / 4 = 1/2 \,,$$

$$\langle \mathbb{d}, \mathbb{b}_2 \rangle / \|\mathbb{b}_2\|^2 = \langle (1,2)^T, (0,1)^T \rangle / 1 = 2 \,.$$

Thus, $\mathbb{x}_{loc} = (1/2, 2)^T$. This coordinate tuple indicates the the point lies outside box $\mathbb{B}$ – since the second coordinate value is **2**. A point lies inside or on the boundary of box $\mathbb{B}$ if the ABSOLUTE VALUES of all coordinates of $\mathbb{x}_{loc}$ are smaller than or equal to one. The lower figure on the previous page shows the point $\bar{\mathbb{x}} = (2.5, 2)^T$ that lies on the top boundary of $\mathbb{B}$ and would thus be "acceptable"; but changing the point's location is not a viable operation. Box $\mathbb{B}$ must be scaled.



Scaling box $\mathbb{B}$ by factor 2 in "second box direction."

Since only the absolute value of the second coordinate of $\mathbb{x}_{loc}$ is larger than one, we only must expand $\mathbb{B}$ in its box-inherent second direction. For example, we can scale basis vector $\mathbb{b}_2$ by 2, producing $\tilde{\mathbb{b}}_2 = (0,2)^T$, see left figure. The new coordinates of the point relative to box $\tilde{\mathbb{B}}$ are $\langle \mathbb{d}, \mathbb{b}_1 \rangle / \|\mathbb{b}_1\|^2 = 1/2$ and $\langle \mathbb{d}, \tilde{\mathbb{b}}_2 \rangle / \|\tilde{\mathbb{b}}_2\|^2 = \langle (1,2)^T, (0,2)^T \rangle / 4 = 1$.

•••