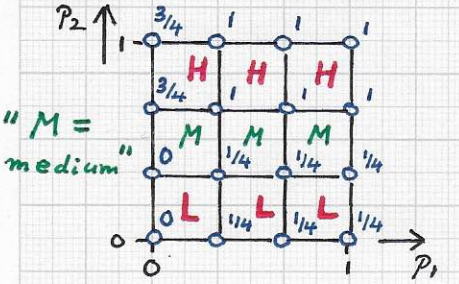
■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...

Special convex combination of $F_1(p_1)$ and $F_2(p_2)$ – from examples on previous page – where $w_1 = 1/4$ and $w_2 = 3/4$:



We must recall that the numerical values shown in the sketch – 0, 1/4, 3/4, 1 – are Bernstein-Bézier function coefficient values and not actual function values of $F(p_1,p_2)$. Nevertheless, since function values of functions in Bernstein-Bézier representation closely "follow" the values of the polynomials' coefficients, one can deduce the low-, medium- and high-value regions from the coefficient values of $b_{i,j}$ – which are shown.

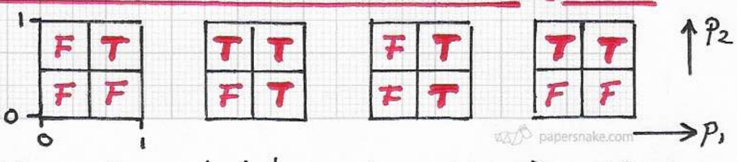
The general convex combination is defined as

$$F(p_1,p_2) = \sum_{j=0}^{n}\sum_{i=0}^{n} w_1 b_i^1 \cdot w_2 b_j^2 \cdot B_i^n(p_1)\, B_j^n(p_2).$$

The figure (left) shows the 4×4 Bernstein-Bézier control coefficient/point net for another convex combination of $F_1(p_1)$ and $F_2(p_2)$ (using the same functions F_1 and F_2 again):

$$F(p_1,p_2) = 1/4\, F_1(p_1) + 3/4\, F_2(p_2).$$

The 4×4 nodes in the sketch show their associated $b_{i,j}$-values next to them. This special convex combination can be interpreted as a numerical approximation of the Logic function that yields TRUE when combining H_2 with H_1 or H_2 with L_1 and yields FALSE otherwise. **In summary, the four constructed decider functions $F = F_1 \cdot F_2$; $F = 1/2\, F_1 + 1/2\, F_2$; $F = 1 \cdot F_1 + 0 \cdot F_2$; $F = 1/4\, F_1 + 3/4\, F_2$ are examples of the four types that are logically and physically meaningful with the following TRUE (T) and FALSE (F) regions:** !



These four sketches show the described combinations of the specific functions F_1 and F_2.

...

Bernd Hamann
Stratovan

6/13/2022

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

- Laplacian eigenfunctions and neural networks:...

Various possibilities for combining, for example, univariate functions have varying degrees of "representational power."

- Basic additive combination:

$$\mathcal{F}_1(x) = a_0 + a_1 x + a_2 x^2$$
$$\mathcal{F}_2(y) = b_0 + b_1 y + b_2 y^2$$

$$\mathcal{F}(x,y) = w_1 \mathcal{F}_1(x) + w_2 \mathcal{F}_2(y),$$
$$w_1, w_2 \geq 0,\ w_1 + w_2 = 1$$

$$\Rightarrow \mathcal{F}(x,y) = w_1 a_0 + w_2 b_0 + w_1(a_1 x + a_2 x^2) + w_2(b_1 y + b_2 y^2)$$
$$= A_{0,0} + A_{1,0}x + A_{2,0}x^2 + A_{0,1}y + A_{0,2}y^2$$

- Basic multiplicative combination:

$$\mathcal{F}(x,y) = \mathcal{F}_1(x) \cdot \mathcal{F}_2(y)$$

$$\Rightarrow \mathcal{F}(x,y) = a_0 b_0 + \ldots + a_2 b_2 x^2 y^2$$
$$= \sum_{j=0}^{2} \sum_{i=0}^{2} a_i b_j x^i y^j$$
$$= \sum_{j=0}^{2} \sum_{i=0}^{2} \bar{A}_{i,j} x^i y^j$$
$$= \bar{A}_{0,0} + \ldots + \bar{A}_{2,2} x^2 y^2$$

- Coefficient "summary":

y^2	$A_{0,2}$	O	O
y	$A_{0,1}$	O	O
1	$A_{0,0}$	$A_{1,0}$	$A_{2,0}$
	1	x	x^2

No. coefficients $= 2n+1$

y^2 $\bar{A}_{0,2}$	xy^2 $\bar{A}_{1,2}$	x^2y^2 $\bar{A}_{2,2}$
y $\bar{A}_{0,1}$	xy $\bar{A}_{1,1}$	x^2y $\bar{A}_{2,1}$
1 $\bar{A}_{0,0}$	x $\bar{A}_{1,0}$	x^2 $\bar{A}_{2,0}$

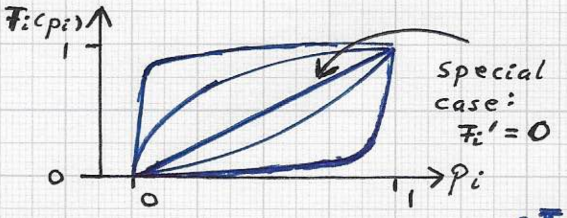
No. coefficients $= (n+1)^2$

It is evident at this point that several mathematical topics, mathematical branches, come together to address the material classification problem: numerical data approximation; optimization; numerical linear algebra; Boolean logic; numerical analysis; computational geometry and more. Several areas at the interface of mathematics and computer science are also relevant: graphs and networks and their processing; graph and network computations; logic gates and circuits; signal processing; computational efficiency and error; efficient representation and processing of hierarchical and multi-resolution data structures; convolutional data processing and more. OUR ULTIMATE GOAL IS THE CONSTRUCTION OF A MULTIVARIATE DECIDER FUNCTION $\mathcal{F}$ THAT HAS AS ITS VALUE "TRUE", "1", "YES", "HIGH" ETC. WHEN $\mathcal{F}$'S ARGUMENT VARIABLES DEFINE AN OBJECT/MATERIAL CLASS TO BE RECOGNIZED BY $\mathcal{F}$. ...

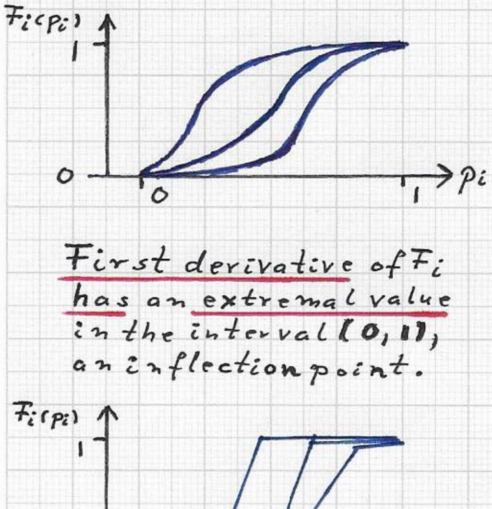
OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• **Laplacian eigenfunctions and neural networks:...**

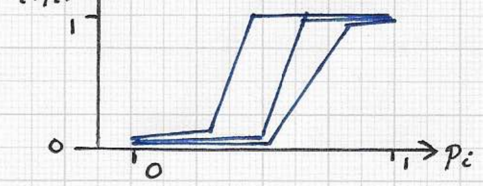
Examples of types of monotonically increasing univariate decider function representations:



First derivative of F_i is monotonically increasing or decreasing; but it has no extremum in the interval $(0,1)$.



First derivative of F_i has an extremal value in the interval $(0,1)$, an inflection point.



The F_i function is a linear spline with three (or more) linear spline segments.

⇒ Based on a p_i-value, $F_i(p_i)$ computes a probability for the unclassified segment – indicating whether it belongs to the class of the classified comparison segment.

We should rewiew meaning and definitions:

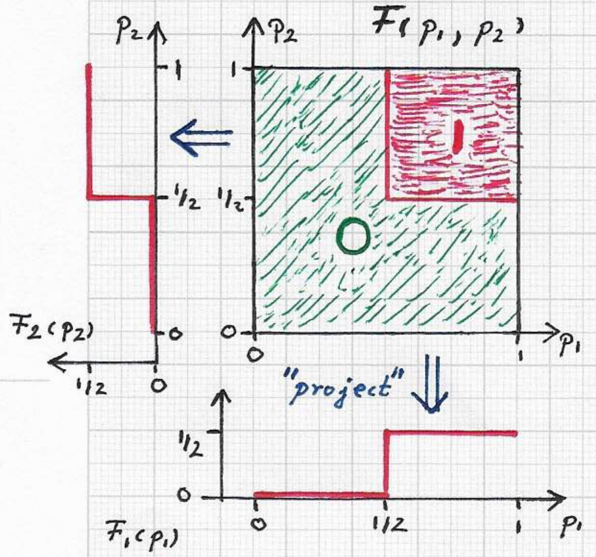
- **p_i** : p_i measures the similarity, at scale i, of two normalized eigenfunction coefficient histograms – between one already classified image sample segment and a given unclassified segment.
- **$F_i(p_i)$** : The univariate decider function F_i has a value between 0 and 1 that serves as a probability; a high F_i-value indicates that the given unclassified segment is very likely to belong to the class of the classified image sample segment used in the comparison – when one considers only scale i.
- **$F(p_1, p_2, \ldots)$**: F is the wanted, unknown multivariate decider function that considers (all or some of) the univariate functions F_i to compute a value between 0 and 1 that serves as a combined probability; just like F_i, F's value indicates a class match probability – considering many scales.

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:...

High-level view of the construction of the decider function as a RE-CONSTRUCTION problem:



One can view F_1 and F_2 as projections of an UNKNOWN function F:

$$F_1(p_1) = \int_{p_2=0}^{1} F(p_1, p_2)\, dp_2$$

$$F_2(p_2) = \int_{p_1=0}^{1} F(p_1, p_2)\, dp_1$$

Considering the simple function $F(p_1, p_2)$ sketched above, the projections are:

$$F_1(p_1) = \begin{cases} 0, & \text{if } p_1 \in [0, 1/2), \\ 1/2, & \text{otherwise} \end{cases} \quad p_1 \in [0,1]$$

$$F_2(p_2) = \begin{cases} 0, & \text{if } p_2 \in [0, 1/2), \\ 1/2, & \text{otherwise} \end{cases} \quad p_2 \in [0,1]$$

It is commonly said that "The multivariate decider function, i.e., the classification function or classifier, is '**learned**' via training." Of course, it is highly desirable to involve sound mathematical methods in the overall construction process of a classification function – to the maximal degree possible.

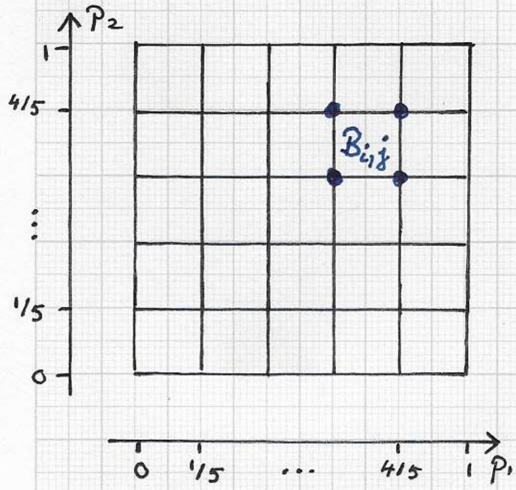
• **Note.** **One can view the construction of a classification function as a RECONSTRUCTION problem: Considering the previous discussion, we want to RE-construct in the two-scale setting, for example, the classification function $F(p_1, p_2)$ from two univariate decider functions $F_1(p_1)$ and $F_2(p_2)$ – that we can understand as PROJECTIONS of F.** Thus, one can understand the construction of F as a FUNCTION RECONSTRUCTION problem – without the physical constraints typical for IMAGE RECONSTRUCTION.

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

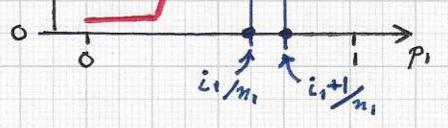
• Laplacian eigenfunctions and neural networks:...

Discretization of F's domain via uniform hypercubes and defining F via a constant spline:



In this 2D example, the grid resolution n has the value $n=5$. A BOX basis function $B_{i,j}$, $i,j = 0 \ldots (n-1)$, has value 1 over its associated grid cell and has value 0 outside this cell.

Discretization of univariate function $F_1(p_1)$ via uniform intervals:



For reconstruction the function $F_1(p_1)$ is discretized via intervals $[i_1/n_1, (i_1+1)/n_1)$, $i_1 = 0 \ldots n_1-1$, with associated constant values.

By adopting this RECONSTRUCTION view for decider function construction, one can employ the numerical methods used in traditional image reconstruction from projection data: discretization methods; best approximation methods; and efficient methods for solving large linear equation systems. In the following, we use a simple constant spline representation for the (unknown) function $F(p_1, p_2, \ldots)$. Considering only scales 1 and 2, this function is defined as

$$F(p_1, p_2) = \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} c_{i,j} \, B_{i,j}(p_1, p_2), \text{ where}$$

$$B_{i,j}(p_1, p_2) = \begin{cases} 1, & \text{if } p_1 \in [i/n, (i+1)/n) \\ & \text{and } p_2 \in [j/n, (j+1)/n) \\ 0, & \text{otherwise} \end{cases}.$$

The coefficient values of $c_{i,j}$ of the BOX basis functions $B_{i,j}(p_1, p_2)$ are the n^2 unknown values that must be determined optimally via the given univariate functions $F_1(p_1)$ and $F_2(p_2)$ – to be discretized as well.

...