Interpolating Values in Code Space

Dr. Adam Bourne

✉ a.bourne@physicist.net

Abstract

A method is described for interpolating un-sampled values attributed to points in code space. A metric is used which counts the number of non-equal corresponding indices shared by two given points.

A generalised interpolation equation is derived for values ascribed to nodes on undirected graphs. The equation is then applied specifically to values at points in code space. This interpolation equation is then solved in general for a set of given sampled values in the space.

1 Introduction

This paper lays out a clear method for predictively interpolating values associated with points in code space given at least one sampled value for other points as a training set. Possible applications include in AI where strings of symbols are classified into various types and then predictions are made for untrained examples. Another application would be predicting values for unfilled cells in database entries from the other entries.

Solving the interpolation equation analytically has two main advantages, the obvious one being the exactitude of the solution. In addition, an iterative numerical solution using the sampled points as seed values would be prohibitively costly. For the code space described, each iteration would create an ever increasing ball of values around each sampled point; soaking up memory, with ever increasing computational times for each iteration state.

The method described here reduces the solution to a simple linear problem and for N training examples requires the computation matrix determinants of rank N.

2 Interpolating values on an undirected graph

2.1 Defining terms

We consider an undirected graph, $G = G(\mathcal{V}, \mathcal{E})$, with the set of vertices $\mathcal{V}$ and edges $\mathcal{E}$. Each vertex can be assigned an associated scalar value $\rho = \rho(v \in \mathcal{V})$. The set of nearest-neighbour vertices of a given vertex v are denoted by $\mathcal{V}(v)$.

A measure of how similar a vertex value is to the vertex values of its nearest neighbours can be defined as the mean difference squared.

$$\sigma(v) = \frac{1}{V} \sum_{v' \in \mathcal{V}(v)} (\rho(v') - \rho(v))^2 \tag{1}$$

where $V \equiv \sum_{v' \in \mathcal{V}} 1$ is the total number of vertices. It is assumed that at least one of the vertex values is known from sampling.

2.2 Derivation of interpolation equation

The interpolation equation is the condition arising from minimising the mean nearest neighbour measure over all the vertices which are un-sampled, $\mathcal{V}$, in the graph G. Formally this can be written,

$$\delta \sum_{v \in \mathcal{V}} \sigma(v) = 0 \tag{2}$$

Performing this optimisation yields the following equations on G.

$$\left\{\frac{\partial}{\partial\rho(v)}\sum_{v''\in\mathcal{V}}\sigma(v'')\equiv 2V\sum_{v'\in N(v)}(\rho(v')-\rho(v))=0\right\} \tag{3}$$

$$\Rightarrow \rho^{\text{unsampled}}(v)=\frac{1}{N(v)}\sum_{v'\in\mathcal{V}(v)}\rho(v') \tag{4}$$

where $N(v)=\sum_{v'\in\mathcal{V}(v)}1$ is the number of edges emanating from vertex v. Notice that this is simply the statement that un-sampled vertex values are interpolated to be the mean of their nearest-neighbour vertex values.

Of course, the sampled vertices can take arbitrary vertex values, so would not generally satisfy this equation. In general, a set of independent values assigned to all the vertices in a graph $\{\rho(v)\}$ can be uniquely defined by assigning each vertex a value $Q(v)$, given by the expression,

$$\sum_{v'\in\mathcal{V}(v)}(\rho(v')-\rho(v))=Q(v) \tag{5}$$

where $\{Q(v)\}$ is some set of undetermined values, which we shall call the *pseudo-charge* distribution associated with the vertex values $\{\rho(v)\}$. Equation (5) is now the general interpolation equation over the entire graph, with $Q(v)=0$ for the un-sampled vertices. This can be rewritten in terms of the adjacency matrix [3], $G_{v,v'}$, defined by,

$$G_{v,v'}\equiv\left\{\begin{matrix}0 & \text{vertices } v \text{ and } v' \text{ share no edge}\\ 1 & \text{vertices } v \text{ and } v' \text{ share one edge}\end{matrix}\right\} \tag{6}$$

in terms of which the interpolation eqution is,

$$\sum_{v'}G_{v,v'}(\rho(v')-\rho(v))=Q(v) \tag{7}$$

and the number of edges emanating from vertex v is $N(v)=\sum_{v'}G_{v,v'}$.

2.3 Pseudo-charge neutrality

Theorem 2.1. *For any graph $\mathcal{G}$, the total pseudo-charge, $\sum_{v\in\mathcal{V}}Q(v)=0$*

Proof. Summing over the pseudo-charges associated with $\mathcal{G}$,

$$\sum_{v}Q(v)=\sum_{v,v'}G_{v,v'}(\rho(v')-\rho(v))\equiv\sum_{v'}\rho(v')N(v')-\sum_{v}\rho(v)N(v)\equiv 0$$

□

Hence all possible distributions of $\{\rho(v)\}$ have vanishing overall pseudo-charge.

3 Application to code space

3.1 Description of code space

Consider the special case of the graph's vertices forming the points in a code space (Σ, d) [1, pp. 125-133], with each vertex now being labelled by a total of D integer indices $(m_1, m_2, \cdots, m_D)$, such that,

$$v\rightarrow\mathbf{m}=(m_1,m_2,\cdots,m_D) \quad , \quad 0\leq m_q\leq\mu_q-1 \tag{8}$$

We then define the set of points in the code space by, $\Sigma:\mathbf{m}\in\Sigma \iff 0\leq m_q\leq\mu_q-1\,\forall q:1\leq q\leq D$. Notice that the number of permissible values for m_q is μ_q, which depends on its position index, q.

Vertices in the graph, $G(\mathcal{V},\mathcal{E})$ now map onto this D-dimensional code space. Traversing an edge from one vertex (point) to another involves changing exactly one index $m_q\rightarrow m'_q\neq m_q$ for a given

q. A metric will be used that counts the minimum number of edges that can be crossed in traversing between two nodes, namely,

$$d : d(\mathbf{m}', \mathbf{m}) = \sum_{q=1}^{D} \overline{\delta_{m'_q, m_q}} \tag{9}$$

The left-hand side 'gradient' term in (7) can now be rewritten, with the set of all nearest neighbours to (say) $\mathbf{m}$ being $\cup_{q=1}^{D} \cup_{m'_q \neq m_q} \{\mathbf{m} : m_q \to m'_q\}$

$$\Delta_{\mathbf{m}} \equiv \sum_{v'} G_{v,v'}(\rho(v') - \rho(v)) = \sum_{q=1}^{D} \sum_{\substack{m=0 \\ m \neq m_q}}^{\mu_q - 1} \left(\rho_{\mathbf{m}:m_q \to m} - \rho_{\mathbf{m}}\right) = \sum_{q=1}^{D} \sum_{\alpha=1}^{\mu_q - 1} \left(\rho_{\mathbf{m}:m_q \to m_q + \alpha \bmod \mu_q} - \rho_{\mathbf{m}}\right) \tag{10}$$

where in the final expression the incremental index α sums over 'rotations' in each m_q in succession. NB note $\alpha \neq 0$ as $\mathbf{m} : m_q \to m_q + 0 \bmod \mu_q \equiv \mathbf{m}$ is not a nearest neighbour to $\mathbf{m}$.

3.2 The rotation operator

We can define a rotation operator, R, write down its eigenfunctions and re-frame the gradient operator in terms of it,

$$R : R\, f(m) = f(m + 1 \bmod \mu) \tag{11}$$

which has the eigenfunctions $\{\phi_\mu^\alpha(m)\}$ and corresponding eigenvalues $\{\lambda_\alpha\}$, given by,

$$\left\{\phi_\mu^\alpha(m) = e^{\frac{2\pi i \alpha m}{\mu}} \quad , \quad \lambda_\alpha = e^{\frac{2\pi i \alpha}{\mu}}\right\} \tag{12}$$

3.3 The gradient operator

A single-index gradient term can be rewritten $\sum_{\alpha=1}^{\mu-1} (\rho_{m+\alpha \bmod \mu} - \rho_m) = \Delta\, \rho_m$, where we have defined the *operator*,

$$\Delta = \sum_{\alpha=1}^{\mu-1} (R^\alpha - 1) \tag{13}$$

of which the $\left\{\phi_\mu^\alpha(m) = e^{\frac{2\pi i \alpha m}{\mu}}\right\}$ are eigenfunctions with respective eigenvalues,

$$L_\alpha = \sum_{\beta=1}^{\mu-1} \left(e^{\frac{2\pi i \alpha \beta}{\mu}} - 1\right) = \sum_{\beta=0}^{\mu-1} e^{\frac{2\pi i \alpha \beta}{\mu}} - 1 - (\mu - 1) = \mu\,(\delta_{\alpha,0} - 1) \tag{14}$$

NB notice $L_0 = 0$. Now we can construct a Green's function [2] for the gradient operator. A *truncated* Green's function of Δ, centred on $m = 0$ is,

$$g(m) = \frac{1}{\mu} \sum_{\alpha=1}^{\mu-1} \frac{\phi_\alpha^\mu(m)}{L_\alpha} \tag{15}$$

where the sum is truncated and does not include the term for $\alpha = 0$ as this would create a singularity. To check this is correct one writes,

$$\Delta\, g(m) = \frac{1}{\mu} \sum_{\alpha=1}^{\mu-1} \frac{\Delta\, \phi_\alpha^\mu(m)}{L_\alpha} = \frac{1}{\mu} \sum_{\alpha=1}^{\mu-1} \phi_\alpha^\mu(m) = \frac{1}{\mu} \sum_{\alpha=1}^{\mu-1} e^{\frac{2\pi i \alpha m}{\mu}} = \delta_{m,0} - \frac{1}{\mu} \tag{16}$$

which is a point pseudo-charge in a constant neutralising background, which is a result of truncating the Green's function. Notice the solution has a vanishing overall net charge as stipulated by theorem 2.1.

3.4 Multidimensional code space

Generalising to the D-dimensional code space we define the rotation operators $\{R_q\}$,

$$\{R_q : f(\mathbf{m} = \{m_j\}) = f(\mathbf{m} : m_q \to m_q + 1 \bmod \mu_q)\} \tag{17}$$

with the corresponding eigenfunctions,

$$\phi_{\{\alpha_j\}}(\mathbf{m}) = \prod_{q=1}^{D} \phi_{\alpha_q}(m_q) = \prod_{q=1}^{D} e^{\frac{2\pi i \alpha_q m_q}{\mu_q}} \tag{18}$$

Then the interpolation equation gradient is,

$$\Delta = \sum_{q=1}^{D} \Delta_q = \sum_{q=1}^{D} \sum_{\beta}^{\mu_q - 1} \left(R_q^{\beta} - 1\right) \tag{19}$$

which has the corresponding eigenvalues,

$$L_{\{\alpha_j\}} = \sum_{q=1}^{D} L_{\alpha_q} = \sum_{q=1}^{D} \mu_q \left(\delta_{\alpha_q,0} - 1\right) \tag{20}$$

Noting that this expression is vanishing for $\{\alpha_j\} = \mathbf{0} \equiv \{0,0,0,\cdots\}$, then the truncated Green's function can be constructed as follows,

$$g(\mathbf{m}) = \frac{1}{V} \sum_{\substack{\{\alpha_q=0\} \\ \{\alpha_q\}\neq \mathbf{0}}}^{\{\mu_q-1\}} \frac{\phi_{\{\alpha_j\}}(\mathbf{m})}{L_{\{\alpha_j\}}} \tag{21}$$

where $V \equiv \sum_{\mathbf{m}} 1 = \prod_{j=1}^{D} \mu_j$ is the total number of micro-states or in the graph model, vertices. Checking this corresponds to a unit pseudo-charge in a uniform neutralising background,

$$\Delta\, g(\mathbf{m}) = \frac{1}{V} \sum_{\substack{\{\alpha_q=0\} \\ \{\alpha_q\}\neq \mathbf{0}}}^{\{\mu_q-1\}} \frac{\Delta\, \phi_{\{\alpha_j\}}(\mathbf{m})}{L_{\{\alpha_j\}}} = \frac{1}{V} \sum_{\substack{\{\alpha_q=0\} \\ \{\alpha_q\}\neq \mathbf{0}}}^{\{\mu_q-1\}} \phi_{\{\alpha_j\}}(\mathbf{m}) = \frac{1}{V} \left(\sum_{\{\alpha_q=0\}}^{\{\mu_q-1\}} \prod_{j=1}^{D} e^{\frac{2\pi i \alpha_j m_j}{\mu_j}} - 1 \right) \tag{22}$$

$$= \prod_{j=1}^{D} \left(\frac{1}{\mu_j} \sum_{\alpha=0}^{\mu_q-1} e^{\frac{2\pi i \alpha m_q}{\mu_q}} \right) - \frac{1}{V} = \delta_{\{\alpha_q\},\mathbf{0}} - \frac{1}{V} \tag{23}$$

The eigenvalues in (20) have terms in the summand that only discriminate between $\alpha_q = 0$ or $\alpha_q \neq 0$. So the expression (21) can be simplified by separating the terms in the sum over the $\{\alpha_q\}$ into incidences such that $\alpha_q = 0$ and $\alpha_q \neq 0$ are explicitly separated out. The summation operator,

$$\sum_{\substack{\{\alpha_q=0\} \\ \{\alpha_q\}\neq \mathbf{0}}}^{\{\mu_q-1\}} \times = \sum_{n=1}^{D} \sum_{j_1<\cdots<j_n=0}^{D} \sum_{\{\alpha_q=0\}}^{\{\mu_q-1\}} \prod_{k\in \mathbf{j}} \overline{\delta_{\alpha_k,0}} \prod_{k\notin \mathbf{j}} \delta_{\alpha_k,0} \times \tag{24}$$

where n is the number of non-zero values in the micro-state indices $\{\alpha_q\}$. Notice $n = 0$ is not included so as to exclude the case $\{\alpha_q\} = \mathbf{0}$. The factor $\overline{\delta_{\alpha_k,0}} \equiv 1 - \delta_{\alpha_k,0}$ picks out the n indices with $\alpha_q \neq 0$, the other factor $\delta_{\alpha_k,0}$ ensures the remaining $D - n$ indices are zero. The subset $\mathbf{j} = \{j_1, \cdots, j_n\}$. The truncated Green's function is then,

$$g(\mathbf{m}) = \frac{1}{V} \sum_{n=1}^{D} \sum_{j_1<\cdots<j_n=0}^{D} \sum_{\{\alpha_q=0\}}^{\{\mu_q-1\}} \prod_{k\in \mathbf{j}} \overline{\delta_{\alpha_k,0}} \prod_{k\notin \mathbf{j}} \delta_{\alpha_k,0} \frac{\prod_{q=1}^{D} \phi_{\alpha_q}(m_q)}{\sum_{q=1}^{D} \mu_q \left(\delta_{\alpha_q,0} - 1\right)} \tag{25}$$

$$= \frac{1}{V} \sum_{n=1}^{D} \sum_{j_1<\cdots<j_n=0}^{D} \sum_{\{\alpha_q=0\}}^{\{\mu_q-1\}} \prod_{k\in \mathbf{j}} \overline{\delta_{\alpha_k,0}} \prod_{k\notin \mathbf{j}} \delta_{\alpha_k,0} \frac{\prod_{q=1}^{D} \phi_{\alpha_q}(m_q)}{\sum_{q=1}^{D} \mu_q \left(\delta_{\alpha_q,0} - 1\right)} \tag{26}$$

in which the following fraction becomes,

$$\frac{\prod_{q=1}^{D}\phi_{\alpha_q}(m_q)}{\sum_{q=1}^{D}\mu_q\left(\delta_{\alpha_q,0}-1\right)}=\frac{\prod_{q\in\mathbf{j}}e^{\frac{2\pi i\alpha_q m_q}{\mu_q}}\cdot\prod_{q\notin\mathbf{j}}e^{\frac{2\pi i\alpha_q m_q}{\mu_q}}}{\sum_{q\in\mathbf{j}}\mu_q\left(\delta_{\alpha_q,0}-1\right)+\sum_{q\notin\mathbf{j}}\mu_q\left(\delta_{\alpha_q,0}-1\right)} \tag{27}$$

$$=\frac{\prod_{q\in\mathbf{j}}e^{\frac{2\pi i\alpha_q m_q}{\mu_q}}\cdot\prod_{q\notin\mathbf{j}}1}{\sum_{q\in\mathbf{j}}\mu_q\ (0-1)+\sum_{q\notin\mathbf{j}}\mu_q\ (1-1)}=-\frac{\prod_{q\in\mathbf{j}}e^{\frac{2\pi i\alpha_q m_q}{\mu_q}}}{\sum_{q\in\mathbf{j}}\mu_q} \tag{28}$$

The sums over $\alpha_{q\notin\mathbf{j}}$ now all contribute a trivial factor of unity. The sums over $\alpha_{q\notin\mathbf{j}}$ exclude $\alpha_q=0$ terms, giving a truncated Green's function,

$$g(\mathbf{m})=-\frac{1}{V}\sum_{n=1}^{D}\sum_{j_1<\cdots<j_n}\frac{1}{\sum_{q\in\mathbf{j}}\mu_q}\sum_{\{\alpha_{q\in\mathbf{j}}=1\}}^{\{\mu_q-1\}}\prod_{k\in\mathbf{j}}e^{\frac{2\pi i\alpha_k m_k}{\mu_k}} \tag{29}$$

The factor,

$$\sum_{\{\alpha_{q\in\mathbf{j}}=1\}}^{\{\mu_q-1\}}\prod_{k\in\mathbf{j}}e^{\frac{2\pi i\alpha_k m_k}{\mu_k}}=\prod_{k\in\mathbf{j}}\sum_{\alpha=1}^{\mu_k-1}e^{\frac{2\pi\alpha m_k}{\mu_k}}=\prod_{k\in\mathbf{j}}\left(\sum_{\alpha=0}^{\mu_k-1}e^{\frac{2\pi\alpha m_k}{\mu_k}}-1\right)=\prod_{k\in\mathbf{j}}\left(\mu_k\,\delta_{m_k,0}-1\right) \tag{30}$$

which gives the truncated Green's function,

$$g(\mathbf{m})=-\frac{1}{V}\sum_{n=1}^{D}\sum_{j_1<\cdots<j_n}\frac{1}{\sum_{q\in\mathbf{j}}\mu_q}\prod_{k\in\mathbf{j}}\left(\mu_k\,\delta_{m_k,0}-1\right) \tag{31}$$

3.5 Code space subspaces

We define the subset of elements from $\{1,2,\ldots,D\}$,

$$\sigma_\mu : q\in\sigma_\mu \Leftrightarrow \mu_q=\mu \tag{32}$$

We define the subspaces of (Σ,d), to be the set of spaces $\{(\Sigma_\mu,d_\mu)\}$, where Σ_μ is a code space with member points being labelled by the indices $(m_{q\in\sigma_\mu})$

The corresponding metrics are,

$$\left\{d_\mu\equiv\sum_{k\in\sigma_\mu}\overline{\delta_{m'_k,m_k}}\right\} \tag{33}$$

4 Projection of truncated Green's function onto code subspaces

It can be seen by inspection that in (31), the factor $\mu_k\,\delta_{m_k,0}-1=-(1-\mu_k)^{\delta_{m_k,0}}$. The product over a given permutation $\{j_1,\cdots,j_n\}$ can then be separated into products over the subsets of indices $\{\sigma_\mu\}$,

$$\prod_{k\in\mathbf{j}}\left(\mu_k\,\delta_{m_k,0}-1\right)=\prod_{\mu=2}^{\mu_{\max}}\prod_{\substack{k\in\mathbf{j}\\k\in\sigma_\mu}}(-1)(1-\mu)^{\delta_{m_k,0}}=\prod_{\mu=2}^{\mu_{\max}}(-1)^{n_\mu(\mathbf{j})}(1-\mu)^{n_\mu(\mathbf{j})-q_\mu(\mathbf{j})} \tag{34}$$

where $n_\mu(\mathbf{j})\equiv\sum_{j\in\sigma_\mu,\mathbf{j}}1$ is the number of the members of $\mathbf{j}$ that are also members of σ_μ. The parameter $q_\mu(\mathbf{j},\mathbf{m})\equiv\sum_{k\in\sigma_\mu,\mathbf{j}}\overline{\delta_{m_k,0}}$ is the number of members of $\mathbf{j}\equiv\{j_{k\in\mathbf{j}}\}$ that are also members of σ_μ AND have $m_k\neq0$. From this it follows that $q_\mu(\mathbf{j})\leq n_\mu(\mathbf{j})$ and $\sum_\mu n_\mu(\mathbf{j})=n$. Also, the sum $\sum_{q\in\mathbf{j}}\mu_q\equiv\sum_\mu\mu\,n_\mu(\mathbf{j})$ and the truncated Green's function,

$$g(\mathbf{m})=-\sum_{n=1}^{D}\sum_{j_1<\cdots<j_n=0}^{D}\frac{(-1)^n}{\sum_\mu n_\mu(\mathbf{j})}\prod_{\mu=2}^{\mu_{\max}}(1-\mu)^{n_\mu(\mathbf{j})-q_\mu(\mathbf{j},\mathbf{m})} \tag{35}$$

The summation operator in (35) can decomposed into the independent subspace variables $\{n_\mu = n_\mu(\mathbf{j})\}$ and $\{q_\mu = q_\mu(\mathbf{j},\mathbf{m})\}$ in the manner,

$$\sum_{n=1}^{D} \sum_{j_1<\cdots<j_n=0}^{D} \times \equiv \sum_{\{n_\mu\}} \sum_{\{q_\mu\}} \sum_{n=1}^{D} \sum_{j_1<\cdots<\jmath_n=0}^{D} \delta_{n,\sum_\mu n_\mu(\mathbf{j})} \left(\prod_\mu \delta_{n_\mu,n_\mu(\mathbf{j})} \delta_{q_\mu,q_\mu(\mathbf{j},\mathbf{m})} \right) \times \tag{36}$$

This gives the truncated Green's function in terms of the subspace variables,

$$g(\mathbf{m}) = -\sum_{\{n_\mu,q_\mu\}} \sum_{n=1}^{D} \frac{(-1)^n \, \delta_{n,\sum_\mu n_\mu}}{\sum_\mu \mu \, n_\mu} \left(\prod_{\mu=2}^{\mu_{\max}} (1-\mu)^{n_\mu - q_\mu} \right) w_{\mathbf{n},\mathbf{q}}(\mathbf{m}) \tag{37}$$

where the coefficient,

$$w_{\mathbf{n},\mathbf{q}}(\mathbf{m}) \equiv \sum_{j_1<\cdots j_n=0}^{D} \left(\prod_\mu \delta_{n_\mu,n_\mu(\mathbf{j})} \, \delta_{q_\mu,q_\mu(\mathbf{j},\mathbf{m})} \right) \tag{38}$$

4.1 Calculation of coefficent

The coefficient $w_{\mathbf{n},\mathbf{q}}(\mathbf{m})$ counts the number of ways of choosing n_μ elements from σ_μ such that exactly q_μ of them have $\alpha_q \neq$ for each of the subspaces $\{\sigma_\mu\}$ simultaneously. This can be split into a product over the subspace labels,

$$w_{\mathbf{n},\mathbf{q}}(\mathbf{m}) \equiv \prod_\mu w_{n_\mu,q_\mu}(\{m_{j\in\sigma_\mu}\}) \tag{39}$$

Each subspace has a total of $D_\mu \equiv \sum_{j\in\sigma_\mu} 1$ elements, of which $d_\mu \equiv \sum_{k\in\sigma_\mu} \overline{\delta_{m_k,0}} = d_\mu(\mathbf{m})$ have $\alpha_k \neq 0$. So $w_{n_\mu,q_\mu}(\{m_{j\in\sigma_\mu}\})$ counts the number of ways of choosing q_μ objects from a choice of d_μ in total AND choosing $(n_\mu - q_\mu)$ objects from a possible total of $D_\mu - d_\mu$.

$$w_{n_\mu,q_\mu}(\{m_{j\in\sigma_\mu}\}) = w_{n_\mu,q_\mu}(d_\mu) = \begin{bmatrix} d_\mu \\ q_\mu \end{bmatrix} \begin{bmatrix} D_\mu - d_\mu \\ n_\mu - q_\mu \end{bmatrix} \tag{40}$$

$$= \frac{d_\mu!}{(d_\mu - q_\mu)! \, q_\mu!} \frac{(D_\mu - d_\mu)!}{(D_\mu - d_\mu - n_\mu + q_\mu)! \, (n_\mu - q_\mu)!} \tag{41}$$

where

$$\begin{bmatrix} n \\ r \end{bmatrix} = \frac{n!}{(n-r)! \, r!}$$

is the binomial coeeficient.

4.2 Truncated Green's function in terms of subspace coordinates

The truncated Green's function now becomes,

$$g(\mathbf{m}) = -\sum_{\{n_\mu,q_\mu\}} \sum_{n=1}^{D} \frac{(-1)^n \, \delta_{n,\sum_\mu n_\mu}}{\sum_\mu \mu \, n_\mu} \prod_\mu \left((1-\mu)^{n_\mu - q_\mu} \, w_{n_\mu,q_\mu}(d_\mu) \right) \tag{42}$$

$$= -\sum_{\substack{\{n_\mu\} \\ \{n_\mu\}\neq \mathbf{0}}} \frac{(-1)^{\sum_\mu n_\mu}}{\sum_\mu \mu \, n_\mu} \sum_{\{q_\mu\}} \prod_\mu \left((1-\mu)^{n_\mu - q_\mu} \, w_{n_\mu,q_\mu}(d_\mu) \right) \tag{43}$$

$$= -\sum_{\substack{\{n_\mu\} \\ \{n_\mu\}\neq \mathbf{0}}} \frac{1}{\sum_\mu \mu \, n_\mu} \prod_\mu (-1)^{n_\mu} \sum_{q_\mu} \left((1-\mu)^{n_\mu - q_\mu} \, w_{n_\mu,q_\mu}(d_\mu) \right) \tag{44}$$

or more succinctly, in terms of the subspace 'radial' coordinates $\{d_\mu\}$,

$$g(\{d_\mu\}) = -\sum_{\substack{\{n_\mu\} \\ \{n_\mu\}\neq \mathbf{0}}} \frac{1}{\sum_\mu \mu \, n_\mu} \prod_\mu \varphi_{n_\mu}^{\mu,D_\mu}(d_\mu) \tag{45}$$

where,

$$\varphi_n^{\mu,D_\mu}(r) \equiv (-1)^n \sum_q (1-\mu)^{n-q} \begin{bmatrix} r \\ q \end{bmatrix} \begin{bmatrix} D_\mu - r \\ n - q \end{bmatrix} \tag{46}$$

5 General solution to interpolation equation

Given the linearity of (5), its general solution can be constructed from a linear superposition of truncated Green's functions, each being centred on every one of the micro-state code space points $\mathbf{m} = (m_1, m_2, \ldots, m_D)$ respectively,

$$\rho(\mathbf{m}) = \lambda + \sum_{\mathbf{m}'} Q(\mathbf{m}')\, g\left(\{d_\mu(\mathbf{m}, \mathbf{m}')\}\right) \tag{47}$$

where λ is some undetermined constant and the metric measuring the minimum number of pairwise digit changes between points $\mathbf{m}$ and $\mathbf{m}'$ in the subspace labelled μ is,

$$d_\mu(\mathbf{m}, \mathbf{m}') \equiv \sum_{k \in \sigma_\mu} \overline{\delta_{m_k, m'_k}} \tag{48}$$

That this is a solution is confirmed by operating on it with the gradient operator Δ,

$$\Delta\, \rho(\mathbf{m}) = \sum_{\mathbf{m}'} Q(\mathbf{m}')\, \Delta\, g(\{d_\mu(\mathbf{m}, \mathbf{m}')\}) = \sum_{\mathbf{m}'} Q(\mathbf{m}') \left(\delta_{\mathbf{m},\mathbf{m}'} - \frac{1}{V}\right) = Q(\mathbf{m}) - \sum_{\mathbf{m}'} Q(\mathbf{m}') \tag{49}$$

which yields the correct pseudo-charge distribution because $\sum_{\mathbf{m}'} Q(\mathbf{m}') = 0$, as required.

5.1 Determining the pseudo-charge distribution from the sampled values

Remember that the un-sampled points in code space have a vanishing pseudo-charge associated with them. We will assume there are a total of a set of N sampled values $\rho(\mathbf{m}_n)$ for the sample points $\{\mathbf{m}_n\}$ with $1 \leq n \leq N$. The associated pseudo-charge distribution is then,

$$Q(\mathbf{m}) = \sum_{n=1}^{N} Q(\mathbf{m}_n)\, \delta_{\mathbf{m},\mathbf{m}_n} \tag{50}$$

Substituting (50) into (47) gives the distribution of the sampled and interpolated values,

$$\rho(\mathbf{m}) = \lambda + \sum_{n=1}^{N} Q(\mathbf{m}_n)\, g(\{d_\mu(\mathbf{m}, \mathbf{m}_n)\}) \tag{51}$$

where the non-zero pseudo-charges at the sample points act as 'sources' for the field of un-sampled, interpolated values. In particular, the actual sample values $\{\rho(\mathbf{m}_n)\}$ are related to the pseudo-charges by,

$$\rho(\mathbf{m}_{n'}) = \lambda + \sum_{n=1}^{N} Q(\mathbf{m}_n)\, g(\{d_\mu(\mathbf{m}_{n'}, \mathbf{m}_n)\}) \tag{52}$$

This N-dimensional linear equation can be written,

$$\mathbf{r} = \lambda\, \mathbf{1} + \mathbf{g} \cdot \mathbf{q} \tag{53}$$

where the vectors $\mathbf{r} = (\{r_n = \rho(\mathbf{m}_n)\})$, $\mathbf{1} = (\{1, 1, \cdots, 1\})$, $\mathbf{q} = (\{q_n = Q(\mathbf{m}_n)\})$ and the rank N, symmetric matrix $\mathbf{g}$, has components $g_{n'n} \equiv g(\{d_\mu(\mathbf{m}_{n'}, \mathbf{m}_n)\})$. In addition, the constraint of charge neutrality is $\sum_n q_n \equiv \mathbf{1}^{\mathrm{T}} \cdot \mathbf{q} = 0$. Combining this constraint with (53) gives the linear equation describing the relationship between the sample values and the associated pseudo charges,

$$\begin{pmatrix} \mathbf{r} \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{g} & \mathbf{1} \\ \mathbf{1}^{\mathrm{T}} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{q} \\ \lambda \end{pmatrix} \tag{54}$$

The linear equation (54) can then be solved for $\mathbf{q}$ and λ, for example, using Cramer's rule.

References

[1] *Fractals everywhere* (Academic Press, San Diego, 1994).

[2] *Green's functions with applications* (Chapman & Hall/CRC, London, 2001).

[3] *Adjacency matrix* (Available from: `http://mathworld.wolfram.com/AdjacencyMatrix.html`, Accessed 4 April 2019).