Adaptive Neural Networks and Random Distribution for Cancer Tissue Detection and Localization with Random multivariable

Akash Singh, PhD
IBM Corporation
Sacramento CA

Abstract

In this paper a novel approach is described to perform detection of Cancer Tissues by directly modeling the statistical characteristics of the Cancer Cells. This approach allows us to represent Cancer Tissue Acquisitions in the form of pattern that will be analyzed and monitored using Adaptive Self Organizing Maps and Mathematical framework of Cancer Random Tissue Distributions and Localization of Cancer Cells.MRI Images are stacked and pattern recognition techniques are applied to determine Cancer Tissue Image Segmentation and Registration.

1. Introduction

Growth of Cancer patients is increasing and many patients diagnosed with Cancer at a Late Stage. So there is a high requirement of innovation in the field of Early Cancer Detection. Cancer Screening can help to find the cancer in early stage of development and there are more chances better treatment results. However, some of the cancer types still don't have screening test available and few populations with certain genetic code.

Cancer is a disease in which abnormal cells start dividing and there is no immunity defense to control the cell division and more likely invade other connected tissue structure. Cancer cells can distribution in random order through blood stream and lymph systems. There are more than 100 different types of cancer.

Requirements engineering is an attempt to define a discipline for the management of requirements across the system development life cycle. In particular, the discipline addresses the stages preceding the better understood, downstream activities of detailed design, implementation, testing, and maintenance, for which there exist reasonably formal engineering practices and procedures often supported by computer tools.

At the front end of the life cycle, the task is to understand the customer's requirements. Most requirements begin as natural language statements embedded within formal project specification documents, often hundreds of pages in length. These documents normally represent the unresolved views of a group of individuals and will, in most cases be fragmentary, inconsistent, contradictory, seldom be prioritized and often be overstated, beyond actual needs. There is very little in the way of formal process and tool support in this area. This is unfortunate, as the front end tasks represent the key leverage points in the entire design and development process. Mistakes and misunderstandings at this stage may result in enormous economic and technical problems later on in the life cycle.

2. Statistical modeling of Joint Cumulant

The Independence among signals means there is no statistical dependence among them. For the 2nd order statistics with Gaussian random variables, independence means their mutual correlation is zero. For higher order statistics, the dependence is judged by joint cumulant, which means their mutual joint cumulants are zero. Generally speaking, we deal with non-Gaussian random variables. So it is necessary to consider higher order statistics for independence [6-11].

For a set of n real random variable {x1, x2, …, xn}, their joint moment of order r = k1 + k2 + … + kn are given by Papoulis [2]:

$$\mathrm{Mom}\left[x_1^{k1}, x_2^{k2}, \ldots, x_n^{kn}\right] = E\left\{x_1^{k1} x_2^{k2} \ldots x_n^{kn}\right\} = (-j)^r \left.\frac{\partial^r \Phi(\omega_1, \omega_2, \ldots, \omega_n)}{\partial\omega_1^{k1}\partial\omega_2^{k2}\ldots\partial\omega_n^{kn}}\right|_{\omega_1=\omega_2=\ldots=\omega_n=0}$$

where

$$\Phi(\omega_1, \omega_2, \ldots \omega_n) = E\{\exp(j(\omega_1 x_1 + \omega_2 x_2 + \ldots + \omega_n x_n))\}$$

is there joint characteristic function.

Another form of joint characteristic function is defined as the natural logarithm of

$$\Phi(\omega_1, \omega_2, \ldots \omega_n);\ \text{i.e.,}$$

$$\tilde{\Psi}(\omega_1, \omega_2, \ldots \omega_n) = \ln[\Phi(\omega_1, \omega_2, \ldots \omega_n)]$$

Joint cumulants can be defined as the coefficients in the Taylor series expansion of the above characteristic function about zero:

$$\mathrm{Cum}\left[x_1^{k_1}, x_2^{k_2}, \ldots, x_n^{k_n}\right] = (-j)^r \left.\frac{\partial^r \tilde{\Psi}(\omega_1, \omega_2, \ldots, \omega_n)}{\partial\omega_1^{k_1}\partial\omega_2^{k_2}\ldots\partial\omega_n^{k_n}}\right|_{\omega_1=\omega_2=\ldots=\omega_n=0}$$

The general relationship between moments of {x1, x2, …, xn} and joint cumulants Cum[x1, x2, …,Xn] of order r = n is given by Rosenblatt [3]:

$$\mathrm{Cum}[x_1, x_2, \ldots, x_n] = \sum(-1)^{p-1}(p-1)!\,E\left\{\prod_{i\in s_1} x_i\right\}\left\{\prod_{i\in s_2} x_i\right\}\ldots\left\{\prod_{i\in s_p} x_i\right\},$$

where the summation extends over all partitions (s1, s2, …, sp), p = 1, 2, …, n, of
the set of integers (1, 2, …, n).
From statistical point of view, for a set of n real random variable {x1, x2, …, xn}, if their mutual joint cumulants up to order n are all zero, then, they can be claimed independent. For calculation simplicity, we consider only the 3rd and 4th orders. We define a penalty function P,

$$P(x_1, x_2, \ldots, x_j) = \sum_{\text{all } i<j}\left|\mathrm{Cum}(x_i, x_j, x_i)\right|/2 + \sum_{\text{all } i<j}\left|\mathrm{Cum}(x_i, x_j, x_j)\right|/2 + \sum_{\text{all } i<j}\left|\mathrm{Cum}(x_i, x_j, x_i, x_j)\right|^S$$

Separation matrix W can be obtain by minimizing P with regard to W. W is $[w_1^T \quad w_2^T \ \ldots \ w_n^T]^T$, where w1 through wn are row vectors. Gradient decent method is used in experiments. Independent components are extracted one by one. When extracting j-th component xj, joint cumulants are calculated for all i and j combination with i < j. We first assume input signals Y be whitened, i.e. zero mean, unit variance. For non-whitened signals, simply do a PCA whitening. Consider the 1st component. It can be extracted by maximizing its fourth order cumulant, kurtosis [1], and the separating function is,

$$w_1(k+1) = E[\ Y\ (Y^T w_1(k))^3] - 3\ w_1(k)$$

For the j-th component, j>1

$$P(j) = \sum_{i<j}\left|\mathrm{Cum}(z_i, z_j, z_i)\right|/2 + \sum_{i<j}\left|\mathrm{Cum}(z_i, z_j, z_j)\right|/2 + \sum_{i<j}\left|\mathrm{Cum}(z_i, z_j, z_i, z_j)\right|$$

$$= \sum_{i<j}\left|E\left[(w_iY)^2 \cdot w_jY\right]\right|/2 + \sum_{i<j}\left|E\left[w_iY \cdot (w_jY)^2\right]\right|/2 +$$

$$\sum_{i<j}\left|E\left[(w_iY)^2 \cdot (w_jY)^2\right] - 2E^2\left[(w_iY)\cdot(w_jY)\right] - E\left[(w_iY)^2\right]E\left[(w_jY)^2\right]\right|$$

and

$$\Delta w_j = \frac{\partial P(j)}{\partial w_j} = \sum_{i<j}\mathrm{Cum}(z_i, z_j, z_i)\frac{\partial \mathrm{Cum}(z_i, z_j, z_i)}{\partial w_j}/2 +$$

$$\sum_{i<j}\mathrm{Cum}(z_i, z_j, z_j)\frac{\partial \mathrm{Cum}(z_i, z_j, z_j)}{\partial w_j}/2 + \sum_{i<j}\mathrm{Cum}(z_i, z_j, z_i, z_j)\frac{\partial \mathrm{Cum}(z_i, z_j, z_i, z_j)}{\partial w_j}$$

$$= \sum_{i<j}E\left[(w_iY)^2 \cdot w_jY\right]\cdot E\left[(w_iY)^2 Y'\right]/2 + \sum_{i<j}E\left[w_iY \cdot (w_jY)^2\right]\cdot E\left[(w_iY)\cdot(w_jY)Y'\right] +$$

$$\sum_{i<j}\left(E\left[(w_iY)^2 \cdot (w_jY)^2\right] - 2E^2\left[(w_iY)\cdot(w_jY)\right] - E\left[(w_iY)^2\right]E\left[(w_jY)^2\right]\right)\cdot$$

$$\left(2E\left[(w_iY)^2 \cdot (w_jY)Y_i'\right] - 4E\left[(w_iY)\cdot(w_jY)\right]E\left[(w_iY)Y'\right] - 2E\left[(w_iY)^2\right]E\left[(w_jY)Y'\right]\right)$$

$$w_j(k+1) = w_j(k) + \alpha\,\Delta w_j$$

To extract the j-th component, j > 1, we need to calculate 3*j –3 terms of joint cumulants.

We consider the following software engineering model field equations defined over an open bounded piece of network and /or feature space $\Omega \subset R^d$. They describe the dynamics of the mean Software models of each of p node populations.

$$\begin{cases} \left(\frac{d}{dt} + l_i\right)V_i(t,r) = \sum_{j=1}^{p}\int_\Omega J_{ij}(r,\bar{r})S[(V_j(t-\tau_{ij}(r,\bar{r}),\bar{r}) - h_{|j})]d\bar{r} \\ \qquad\qquad + I_i^{ext}(r,t), \qquad t \geq 0, 1 \leq i \leq p, \\ V_i(t,r) = \phi_i(t,r) \qquad\qquad t \in [-T,0] \end{cases} \qquad (1)$$

We give an interpretation of the various parameters and functions that appear in (1), Ω is finite piece of nodes and/or feature space and is represented as an open bounded set of R^d . The vector r and $\bar{r}$ represent points in Ω . The function $S : R \to (0,1)$ is the normalized sigmoid function:

$$S(z) = \frac{1}{1+e^{-z}} \qquad (2)$$

It describes the relation between the input software model rate v_i of population i as a function of the software requirement potential, for example, $V_i = v_i = S[\sigma_i(V_i - h_i)]$. We note V the $p-$ dimensional vector $(V_1,\ldots,V_p)$. The p

function $\phi_i, i=1,\ldots,p,$ represent the initial conditions, see below. We note ϕ the $p-$ dimensional vector $(\phi_1,\ldots,\phi_p)$. The p function $I_i^{ext}, i=1,\ldots,p,$ represent external factors from other network areas. We note I^{ext} the $p-$ dimensional vector $(I_1^{ext},\ldots,I_p^{ext})$. The $p\times p$ matrix of functions $J=\{J_{ij}\}_{i,j=1,\ldots,p}$ represents the connectivity between populations i and j, see below. The p real values $h_i, i=1,\ldots,p,$ determine the threshold of activity for each population, that is, the value of the nodes potential corresponding to 50% of the maximal activity. The p real positive values $\sigma_i, i=1,\ldots,p,$ determine the slopes of the sigmoids at the origin. Finally the p real positive values $l_i, i=1,\ldots,p,$ determine the speed at which each requirement node potential decreases exponentially toward its real value. We also introduce the function $S: R^p \to R^p$, defined by $S(x)=[S(\sigma_1(x_1-h_1)),\ldots,S(\sigma_p-h_p))]$, and the diagonal $p\times p$ matrix $L_0=diag(l_1,\ldots,l_p)$. Is the intrinsic dynamics of the software requirement model population given by the linear response of functional and non functional requirement design pattern. $(\frac{d}{dt}+l_i)$ is replaced by $(\frac{d}{dt}+l_i)^2$ to use the alpha function response. We use $(\frac{d}{dt}+l_i)$ for simplicity although our analysis applies to more general intrinsic software moule requirement dynamics. For the sake, of generality, the propagation delays are not assumed to be identical for all populations, hence they are described by a matrix $\tau(r,\bar{r})$ whose element $\tau_{ij}(r,\bar{r})$ is the propagation delay between population j at $\bar{r}$ and population i at r. The reason for this assumption is that it is still unclear from business requirements if propagation delays are independent of the populations. We assume for technical reasons that τ is continuous, that is $\tau\in C^0(\overline{\Omega}^2, R_+^{p\times p})$. Moreover software data indicate that τ is not a symmetric function i.e., $\tau_{ij}(r,\bar{r})\neq\tau_{ij}(\bar{r},r)$, thus no assumption is made about this symmetry unless otherwise stated. In order to compute the right-hand side of (1), we need to know the node potential factor V on interval $[-T,0]$. The value of T is obtained by considering the maximal delay:

$$\tau_m = \max_{i,j(r,\bar{r}\in\overline{\Omega}\times\overline{\Omega})} \tau_{i,j}(r,\bar{r}) \qquad (3)$$

Hence we choose $T=\tau_m$

A. Software Requirement Mathematical Framework

A convenient functional setting for the non-delayed software requirement model field equations is to use the space $F=L^2(\Omega,R^p)$ which is a Hilbert space endowed with the usual inner product:

$$\langle V,U\rangle_F = \sum_{i=1}^{p}\int_\Omega V_i(r)U_i(r)dr \qquad (1)$$

To give a meaning to (1), we defined the history space $C=C^0([-\tau_m,0],F)$ with $\|\phi\|=\sup_{t\in[-\tau_m,0]}\|\phi(t)\|F$, which is the Banach phase space associated with equation (3). Using the notation $V_t(\theta)=V(t+\theta), \theta\in[-\tau_m,0]$, we write (1) as

$$\begin{cases} V(t)=-L_0V(t)\overset{.}{+}L_1S(V_t)+I^{ext}(t), \\ \qquad V_0=\phi\in C, \end{cases} \qquad (2)$$

Where

$$\begin{cases} L_1: C\to F, \\ \phi\to\int_\Omega J(.,\bar{r})\phi(\bar{r},-\tau(.,\bar{r}))d\bar{r} \end{cases}$$

Is the linear continuous operator satisfying $\|L_1\|\le\|J\|_{L^2(\Omega^2,R^{p\times p})}$. Notice that most of the papers on this subject assume Ω infinite, hence requiring $\tau_m=\infty$.

Proposition 1.0 If the following software requirement model assumptions are satisfied.

1. $J \in L^2(\Omega^2, R^{p\times p}),$
2. The external current $I^{ext} \in C^0(R, F),$
3. $\tau \in C^0(\overline{\Omega^2}, R_+^{p\times p}), \sup_{\overline{\Omega^2}} \tau \le \tau_m.$

Then for any $\phi \in C,$ there exists a unique solution $V \in C^1([0,\infty), F) \cap C^0([-\tau_m, \infty, F)$ to (3)

Notice that this result gives existence on $R_+,$ finite-time explosion is impossible for this delayed differential equation. Nevertheless, a particular solution could grow indefinitely, we now prove that this cannot happen.

B. Boundedness of Solutions

A valid model of software neural networks requirement model should only feature bounded software node potentials.

Theorem 1.0 All the software integration model trajectories are ultimately bounded by the same constant R if $I \equiv \max_{t\in R^+} \left\| I^{ext}(t) \right\|_F < \infty.$

Proof :Let us defined $f : R \times C \to R^+$ as

$$f(t,V_t) \overset{def}{=} \left\langle -L_0 V_t(0) + L_1 S(V_t) + I^{ext}(t), V(t) \right\rangle_F = \frac{1}{2}\frac{d\|V\|_F^2}{dt}$$

We note $l = \min_{i=1,\dots p} l_i$

$$f(t,V_t) \le -l\|V(t)\|_F^2 + (\sqrt{p|\Omega|}\,\|J\|_F + I)\|V(t)\|_F$$

Thus, if

$$\|V(t)\|_F \ge 2\frac{\sqrt{p|\Omega|}.\|J\|_F + I}{l} \overset{def}{=} R, f(t,V_t) \le -\frac{lR^2}{2} \overset{def}{=} -\delta < 0$$

Let us show that the open data route of F of center 0 and radius $R, B_R,$ is stable under the dynamics of equation. We know that $V(t)$ is defined for all $t \ge 0s$ and that $f < 0$ on $\partial B_R,$ the boundary of B_R. We consider three cases for the initial condition V_0. If $\|V_0\|_C < R$ and set $T = \sup\{t \mid \forall s \in [0,t], V(s) \in \overline{B_R}\}.$ Suppose that $T \in R,$ then $V(T)$ is defined and belongs to $\overline{B_R},$ the closure of $B_R,$ because $\overline{B_R}$ is closed, in effect to $\partial B_R,$ we also have $\frac{d}{dt}\|V\|_F^2 \,|_{t=T} = f(T, V_T) \le -\delta < 0$ because $V(T) \in \partial B_R$. Thus we deduce that for $\varepsilon > 0$ and small enough, $V(T+\varepsilon) \in \overline{B_R}$ which contradicts the definition of T. Thus $T \notin R$ and $\overline{B_R}$ is stable.

Because f<0 on $\partial B_R, V(0) \in \partial B_R$ implies that $\forall t > 0, V(t) \in B_R$. Finally we consider the case $V(0) \in C\overline{B_R}$. Suppose that $\forall t > 0, V(t) \notin \overline{B_R},$ then $\forall t > 0, \frac{d}{dt}\|V\|_F^2 \le -2\delta,$ thus $\|V(t)\|_F$ is monotonically decreasing and reaches the value of R in finite time when $V(t)$ reaches ∂B_R. This contradicts our assumption. Thus $\exists T > 0 \mid V(T) \in B_R$.

Proposition 1.1 : Let s and t be measured software requirement functions on X. for $E\varepsilon M$, define

$$\phi(E) = \int_E s\,d\mu \qquad (1)$$

Then ϕ is a measure on M .

$$\int_X (s+t)\,d\mu = \int_X s\,d\mu + \int_X t\,d\mu \qquad (2)$$

Proof : If s and if $E_1, E_2, \dots$ are disjoint members of M whose union is E, the countable additivity of μ shows that

$$\phi(E) = \sum_{i=1}^{n} \alpha_i \mu(A_i \cap E) = \sum_{i=1}^{n} \alpha_i \sum_{r=1}^{\infty} \mu(A_i \cap E_r)$$

$$= \sum_{r=1}^{\infty} \sum_{i=1}^{n} \alpha_i \mu(A_i \cap E_r) = \sum_{r=1}^{\infty} \phi(E_r)$$

Also, $\varphi(\phi)=0,$ so that φ is not identically ∞. Next, let s be as before, let $\beta_1,...,\beta_m$ be the distinct values of t, and let $B_j=\{x:t(x)=\beta_j\}$ if $E_{ij}=A_i\cap B_j$, the $\int_{E_{ij}}(s+t)d\mu=(\alpha_i+\beta_j)\mu(E_{ij})$

and $\int_{E_{ij}}sd\mu+\int_{E_{ij}}td\mu=\alpha_i\mu(E_{ij})+\beta_j\mu(E_{ij})$

Thus (2) holds with E_{ij} in place of X. Since X is the disjoint union of the sets E_{ij} $(1\le i\le n, 1\le j\le m)$, the first half of our proposition implies that (2) holds.

Theorem 1.1: If K is a compact set in the plane whose complement is connected, if f is a continuous complex function on K which is holomorphic in the interior of , and if $\varepsilon>0$, then there exists a polynomial P such that $|f(z)=P(z)|<\varepsilon$ for all $z\varepsilon K$. If the interior of K is empty, then part of the hypothesis is vacuously satisfied, and the conclusion holds for every $f\varepsilon C(K)$. Note that K need to be connected.

Proof: By Tietze's theorem, f can be extended to a continuous function in the plane, with compact support. We fix one such extension and denote it again by f. For any $\delta>0$, let $\omega(\delta)$ be the supremum of the numbers $|f(z_2)-f(z_1)|$ Where z_1 and z_2 are subject to the condition $|z_2-z_1|\le\delta$. Since f is uniformly continous, we have $\lim_{\delta\to 0}\omega(\delta)=0$ (1)

From now on, δ will be fixed. We shall prove that there is a software module API calls polynomial P such that

$$|f(z)-P(z)|<10{,}000\ \omega(\delta)\quad (z\varepsilon K)\qquad (2)$$

By (1), this proves the theorem. Our first objective is the construction of a function $\Phi\varepsilon C_c^{'}(R^2)$, such that for all z

$$|f(z)-\Phi(z)|\le\omega(\delta),\qquad (3)$$

$$|(\partial\Phi)(z)|<\frac{2\omega(\delta)}{\delta},\qquad (4)$$

And

$$\Phi(z)=-\frac{1}{\pi}\iint_X\frac{(\partial\Phi)(\zeta)}{\zeta-z}d\zeta d\eta\qquad (\zeta=\xi+i\eta),\qquad (5)$$

Where X is the set of all points in the support of Φ whose distance from the software runtime parameter and dynamic software object binding complement of K does not δ. (Thus X contains no point which is "far within" K.) We construct Φ as the convolution of f with a smoothing function A. Put $a(r)=0$ if $r>\delta$, put

$$a(r)=\frac{3}{\pi\delta^2}(1-\frac{r^2}{\delta^2})^2\qquad (0\le r\le\delta),\qquad (6)$$

And define

$$A(z)=a(|z|)\qquad (7)$$

For all complex z. It is clear that $A\varepsilon C_c^{'}(R^2)$. We claim that

$$\iint_{R^s}A=1,\qquad (8)$$

$$\iint_{R^2}\partial A=0,\qquad (9)$$

$$\iint_{R^3}|\partial A|=\frac{24}{15\delta}<\frac{2}{\delta},\qquad (10)$$

The constants are so adjusted in (6) that (8) holds. (Compute the integral in polar coordinates), (9) holds simply because A has compact support. To compute (10), express ∂A in polar coordinates, and note that $\partial A/\partial\theta=0,$

$$\partial A/\partial r=-a^{'},$$

Now define

$$\Phi(z)=\iint_{R^2}f(z-\zeta)Ad\xi d\eta=\iint_{R^2}A(z-\zeta)f(\zeta)d\xi d\eta\qquad (11)$$

Since f and A have compact support, so does Φ. Since

$$\Phi(z) - f(z)$$
$$= \iint_{R^2} [f(z-\zeta) - f(z)] A(\xi)\, d\xi d\eta \quad (12)$$

And $A(\zeta)=0$ if $|\zeta|>\delta$, (3) follows from (8). The difference quotients of A converge boundedly to the corresponding software abstract layer partial derivatives, since $A\varepsilon C_c^{'}(R^2)$. Hence the last expression in (11) may be differentiated under the integral sign, and we obtain

$$(\partial\Phi)(z) = \iint_{R^2} (\overline{\partial A})(z-\zeta) f(\zeta)\, d\xi d\eta$$
$$= \iint_{R^2} f(z-\zeta)(\partial A)(\zeta)\, d\xi d\eta$$
$$= \iint_{R^2} [f(z-\zeta)-f(z)](\partial A)(\zeta)\, d\xi d\eta \quad (13)$$

The last equality depends on (9). Now (10) and (13) give (4). If we write (13) with Φ_x and Φ_y in place of $\partial\Phi$, we see that Φ has continuous partial derivatives, if we can show that $\partial\Phi=0$ in G, where G is the set of all $z\varepsilon K$ whose distance from the complement of K exceeds δ. We shall do this by showing that

$$\Phi(z)=f(z) \quad (z\varepsilon G); \quad (14)$$

Note that $\partial f=0$ in G, since f is holomorphic there. Now if $z\varepsilon G$, then $z-\zeta$ is in the interior of K for all ζ with $|\zeta|<\delta$. The mean value property for harmonic functions therefore gives, by the first equation in (11),

$$\Phi(z)=\int_0^\delta a(r)rdr\int_0^{2\pi} f(z-re^{i\theta})d\theta$$
$$=2\pi f(z)\int_0^\delta a(r)rdr = f(z)\iint_{R^2} A = f(z) \quad (15)$$

For all $z\ \varepsilon\ G$, we have now proved (3), (4), and (5) The definition of X shows that X is compact and that X can be covered by finitely many open discs $D_1,...,D_n$, of radius 2δ, whose centers are not in K. Since S^2-K is connected, the center of each D_j can be joined to ∞ by a polygonal path in S^2-K. It follows that each D_j contains a compact connected set E_j, of diameter at least 2δ, so that S^2-E_j is connected and so that $K\cap E_j=\phi$. with $r=2\delta$. There are functions $g_j\varepsilon H(S^2-E_j)$ and constants b_j so that the inequalities.

$$|Q_j(\zeta,z)|<\frac{50}{\delta}, \quad (16)$$
$$\left|Q_j(\zeta,z)-\frac{1}{z-\zeta}\right|<\frac{4,000\delta^2}{|z-\zeta|^2} \quad (17)$$

Hold for $z\notin E_j$ and $\zeta\in D_j$, if

$$Q_j(\zeta,z)=g_j(z)+(\zeta-b_j)g_j^2(z) \quad (18)$$

Let Ω be the complement of $E_1\cup...\cup E_n$. Then Ω is an open set which contains K. Put $X_1=X\cap D_1$ and $X_j=(X\cap D_j)-(X_1\cup...\cup X_{j-1})$, for $2\le j\le n$,

Define

$$R(\zeta,z)=Q_j(\zeta,z) \quad (\zeta\varepsilon X_j, z\,\varepsilon\,\Omega) \quad (19)$$

And

$$F(z)=\frac{1}{\pi}\iint_X(\partial\Phi)(\zeta)R(\zeta,z)\,d\zeta\, d\eta \quad (20)$$
$$(z\ \varepsilon\ \Omega)$$

Since,

$$F(z)=\sum_{j=1}\frac{1}{\pi}\iint_{X_i}(\partial\Phi)(\zeta)Q_j(\zeta,z)\,d\xi d\eta, \quad (21)$$

(18) shows that F is a Software Quality of Service (QoS) finite linear combination of the functions g_j and g_j^2. Hence $F\varepsilon H(\Omega)$. By (20), (4), and (5) we have

$$|F(z)-\Phi(z)|<\frac{2\omega(\delta)}{\pi\delta}\iint_X|R(\zeta,z)$$
$$-\frac{1}{z-\zeta}|\,d\xi d\eta \quad (z\ \varepsilon\ \Omega) \quad (22)$$

Observe that the inequalities (16) and (17) are valid with R in place of Q_j if $\zeta\ \varepsilon\ X$ and $z\ \varepsilon\ \Omega.$ now fix $z\ \varepsilon\ \Omega.$, put $\zeta=z+\rho e^{i\theta}$, and estimate the integrand in (22) by (16) if $\rho < 4\delta$, by (17) if $4\delta \le \rho$. The integral in (22) is then seen to be less than the sum of

$$2\pi\int_0^{4\delta}\left(\frac{50}{\delta}+\frac{1}{\rho}\right)\rho d\rho = 808\pi\delta \qquad (23)$$

And

$$2\pi\int_{4\delta}^{\infty}\frac{4,000\delta^2}{\rho^2}\rho d\rho = 2,000\pi\delta. \qquad (24)$$

Hence (22) yields

$$\left|F(z)-\Phi(z)\right| < 6,000\omega(\delta) \qquad (z\ \varepsilon\ \Omega) \qquad (25)$$

Since $F\ \varepsilon\ H(\Omega), K\subset\Omega$, and S^2-K is connected, Runge's theorem shows that F can be uniformly approximated on K by polynomials. Hence (3) and (25) show that (2) can be satisfied. This completes the proof.

Lemma 1.0 : Suppose software model requirement $f\varepsilon C_c^{'}(R^2)$, the space of all continuously differentiable functions in the plane, with compact support. Put

$$\partial = \frac{1}{2}\left(\frac{\partial}{\partial x}+i\frac{\partial}{\partial y}\right) \qquad (1)$$

Then the following "Cauchy formula" holds:

$$f(z) = -\frac{1}{\pi}\iint_{R^2}\frac{(\partial f)(\zeta)}{\zeta - z}d\xi d\eta$$

$$(\zeta=\xi+i\eta) \qquad (2)$$

Proof: This may be deduced from Green's theorem. However, here is a simple direct proof:

Put $\varphi(r,\theta)=f(z+re^{i\theta}),\ r>0,\ \theta$ real

If $\zeta = z+re^{i\theta}$, the chain rule gives

$$(\partial f)(\zeta) = \frac{1}{2}e^{i\theta}\left[\frac{\partial}{\partial r}+\frac{i}{r}\frac{\partial}{\partial\theta}\right]\varphi(r,\theta) \qquad (3)$$

The right side of (2) is therefore equal to the limit, as $\varepsilon\to 0$, of

$$-\frac{1}{2}\int_\varepsilon^\infty\int_0^{2\pi}\left(\frac{\partial\varphi}{\partial r}+\frac{i}{r}\frac{\partial\varphi}{\partial\theta}\right)d\theta dr \qquad (4)$$

For each $r>0,\varphi$ is periodic in θ, with period 2π. The integral of $\partial\varphi/\partial\theta$ is therefore 0, and (4) becomes

$$-\frac{1}{2\pi}\int_0^{2\pi}d\theta\int_\varepsilon^\infty\frac{\partial\varphi}{\partial r}dr = \frac{1}{2\pi}\int_0^{2\pi}\varphi(\varepsilon,\theta)d\theta \qquad (5)$$

As $\varepsilon\to 0$, $\varphi(\varepsilon,\theta)\to f(z)$ uniformly.
This gives (2)

If $X^\alpha\in a$ and $X^\beta\in k\left[X_1,...X_n\right]$, then $X^\alpha X^\beta = X^{\alpha+\beta}\in a$, and so A satisfies the condition $(*)$. Conversely,

$$(\sum_{\alpha\in A}c_\alpha X^\alpha)(\sum_{\beta\in\square^n}d_\beta X^\beta)=\sum_{\alpha,\beta}c_\alpha d_\beta X^{\alpha+\beta} \qquad (\textit{finite sums}),$$

and so if A satisfies $(*)$, then the subspace generated by the monomials $X^\alpha,\alpha\in a$, is an ideal. The proposition gives a classification of the monomial ideals in $k\left[X_1,...X_n\right]$: they are in one to one correspondence with the subsets A of $\square^n$ satisfying $(*)$. For example, the monomial ideals in $k\left[X\right]$ are exactly the ideals $(X^n),\ n\ge 1$, and the zero ideal (corresponding to the empty set A). We write $\left\langle X^\alpha \mid \alpha\in A\right\rangle$ for the ideal corresponding to A (subspace generated by the $X^\alpha,\alpha\in a$).

LEMMA 1.1. Let S be a Software non functional requirement subset of $\square^n$. The ideal a generated by $X^\alpha,\alpha\in S$ is the monomial ideal corresponding to

$$A\overset{df}{=}\left\{\beta\in\square^n \mid \beta-\alpha\in\square^n, \quad some\ \alpha\in S\right\}$$

Thus, a monomial is in a if and only if it is divisible by one of the $X^\alpha,\alpha\in\mid S$

PROOF. Clearly A satisfies $(*)$, and $a\subset\left\langle X^\beta\mid\beta\in A\right\rangle$. Conversely, if $\beta\in A$, then $\beta-\alpha\in\square^n$ for some $\alpha\in S$, and $X^\beta = X^\alpha X^{\beta-\alpha}\in a$. The last statement follows from the fact that $X^{\alpha} | X^{\beta} \Leftrightarrow \beta - \alpha \in \square^{n}$. Let $A \subset \square^{n}$ satisfy$(*)$. From the geometry of A, it is clear that there is a finite set of elements $S = \left\{\alpha_1, ...\alpha_s\right\}$ of A such that $A = \left\{\beta \in \square^{n} \mid \beta - \alpha_i \in \square^{2},\ some\ \ \alpha_i \in S\right\}$ (The α_i's are the corners of A) Moreover, $a \overset{df}{=} \left\langle X^{\alpha} \mid \alpha \in A \right\rangle$ is generated by the monomials $X^{\alpha_i}, \alpha_i \in S$.

DEFINITION 1.0. For a nonzero ideal a in $k\left[X_1, ..., X_n\right]$, we let $(LT(a))$ be the ideal generated by
$\left\{LT(f) \mid f \in a\right\}$

LEMMA 1.2 Let a be a nonzero ideal in $k\left[X_1, ..., X_n\right]$; then $(LT(a))$ is a monomial ideal, and it equals $(LT(g_1), ..., LT(g_n))$ for some $g_1, ..., g_n \in a$.

PROOF. Since $(LT(a))$ can also be described as the ideal generated by the leading monomials (rather than the leading terms) of elements of a.

THEOREM 1.2. Every *ideal* a in $k\left[X_1, ..., X_n\right]$ is finitely generated; more precisely, $a = (g_1, ..., g_s)$ where $g_1, ..., g_s$ are any elements of a whose leading terms generate $LT(a)$

PROOF. Let $f \in a$. On applying the division algorithm, we find $f = a_1 g_1 + ... + a_s g_s + r, \qquad a_i, r \in k\left[X_1, ..., X_n\right]$, where either $r = 0$ or no monomial occurring in it is divisible by any $LT(g_i)$. But $r = f - \sum a_i g_i \in a$, and therefore $LT(r) \in LT(a) = (LT(g_1), ..., LT(g_s))$, implies that every monomial occurring in r is divisible by one in $LT(g_i)$. Thus $r = 0$, and $g \in (g_1, ..., g_s)$.

DEFINITION 1.1. A finite subset $S = \left\{g_1, | ..., g_s\right\}$ of an ideal a is a standard ($(Gr\ddot{o}bner)$ bases for a if $(LT(g_1), ..., LT(g_s)) = LT(a)$. In other words, S is a standard basis if the leading term of every element of a is divisible by at least one of the leading terms of the g_i.

THEOREM 1.3 *The ring $k[X_1, ..., X_n]$ is Noetherian i.e., every ideal is finitely generated.*

PROOF. For $n = 1$, $k[X]$ is a principal ideal domain, which means that every ideal is generated by single element. We shall prove the theorem by induction on n. Note that the obvious map $k[X_1, ...X_{n-1}][X_n] \rightarrow k[X_1, ...X_n]$ is an isomorphism – this simply says that every polynomial f in n variables $X_1, ...X_n$ can be expressed uniquely as a polynomial in X_n with coefficients in $k[X_1, ..., X_n]$:

$$f(X_1, ...X_n) = a_0(X_1, ...X_{n-1})X_n^{r} + ... + a_r(X_1, ...X_{n-1})$$

Thus the next lemma will complete the proof

LEMMA 1.3. If A is Noetherian, then so also is $A[X]$

PROOF. For a polynomial

$$f(X) = a_0 X^{r} + a_1 X^{r-1} + ... + a_r, \qquad a_i \in A, \qquad a_0 \neq 0,$$

r is called the degree of f, and a_0 is its leading coefficient. We call 0 the leading coefficient of the polynomial 0. Let a be an ideal in $A[X]$. The leading coefficients of the polynomials in a form an ideal $a^{'}$ in A, and since A is Noetherian, $a^{'}$ will be finitely generated. Let $g_1,\ldots,g_m$ be elements of a whose leading coefficients generate a', and let r be the maximum degree of g_i. Now let $f \in a$, and suppose f has degree $s > r$, say, $f = aX^s + \ldots$ Then $a \in a'$, and so we can write

$a = \sum b_i a_i, \qquad b_i \in A,$

$a_i = leading\ coefficient\ of\ g_i$

Now

$f - \sum b_i g_i X^{s-r_i}, \quad r_i = \deg(g_i),$ has degree $< \deg(f)$. By continuing in this way, we find that $f \equiv f_t \quad \mod(g_1,\ldots g_m)$ With f_t a polynomial of degree $t < r$. For each $d < r$, let a_d be the subset of A consisting of 0 and the leading coefficients of all polynomials in a of degree d; it is again an ideal in A. Let $g_{d,1},\ldots,g_{d,m_d}$ be polynomials of degree d whose leading coefficients generate a_d. Then the same argument as above shows that any polynomial f_d in a of degree d can be written $f_d \equiv f_{d-1} \quad \mod(g_{d,1},\ldots g_{d,m_d})$ With f_{d-1} of degree $\leq d-1$. On applying this remark repeatedly we find that $f_t \in (g_{r-1,1},\ldots g_{r-1,m_{r-1}},\ldots g_{0,1},\ldots g_{0,m_0})$ Hence

$f_t \in (g_1,\ldots g_m g_{r-1,1},\ldots g_{r-1,m_{r-1}},\ldots,g_{0,1},\ldots,g_{0,m_0})$

and so the polynomials $g_1,\ldots,g_{0,m_0}$ generate a.

One of the great successes of category theory in computer science has been the development of a “unified theory” of the constructions underlying denotational semantics. In the untyped λ-calculus, any term may appear in the function position of an application. This means that a model D of the λ-calculus must have the property that given a term t whose interpretation is $d \in D$, Also, the interpretation of a functional abstraction like $\lambda x\,.\,x$ is most conveniently defined as a function from $D to D$, which must then be regarded as an element of D. Let $\psi : [D \to D] \to D$ be the function that picks out elements of D to represent elements of $[D \to D]$ and $\phi : D \to [D \to D]$ be the function that maps elements of D to functions of D. Since $\psi(f)$ is intended to represent the function f as an element of D, it makes sense to require that $\phi(\psi(f)) = f$, that is, $\psi\, o\, \psi = id_{[D\to D]}$ Furthermore, we often want to view every element of D as representing some function from $D\ to\ D$ and require that elements representing the same function be equal – that is

$\psi(\varphi(d)) = d$

or

$\psi\, o\, \phi = id_D$

The latter condition is called extensionality. These conditions together imply that $\phi\ and\ \psi$ are inverses--- that is, D is isomorphic to the space of functions from $D\ to\ D$ that can be the interpretations of functional abstractions: $D \cong [D \to D]$ .Let us suppose we are working with the untyped $\lambda - calculus$, we need a solution ot the equation $D \cong A + [D \to D]$, where A is some predetermined domain containing interpretations for elements of *C*. Each element of D corresponds to either an element of A or an element of $[D \to D]$, with a tag. This equation can be solved by finding least fixed points of the function $F(X) = A + [X \to X]$ from domains to domains --- that is, finding domains X such that $X \cong A + [X \to X]$, and such that for any domain Y also satisfying this equation, there is an embedding of X to Y --- a pair of maps

$$X \quad \overset{f}{\underset{f^R}{\square}} \quad Y$$

Such that

$f^R\, o\, f = id_X$

$f\, o\, f^R \subseteq id_Y$

Where $f \subseteq g$ means that f *approximates* g in some ordering representing their information content. The key shift of perspective from the domain-theoretic to the more general category-theoretic approach lies in considering *F* not as a function on domains, but as a *functor* on a category of domains. Instead of a least fixed point of the function, *F*.

Definition 1.3: Let ***K*** be a category and $F: K \to K$ as a functor. A fixed point of *F* is a pair (A,a), where A is a ***K-object*** and $a: F(A) \to A$ is an isomorphism. A prefixed point of F is a pair (A,a), where A is a ***K-object*** and a is any arrow from F(A) to A

Definition 1.4 : An $\omega - chain$ in a category ***K*** is a diagram of the following form:

$\Delta = D_o \xrightarrow{f_o} D_1 \xrightarrow{f_1} D_2 \xrightarrow{f_2} .....$

Recall that a cocone μ of an $\omega - chain\ \Delta$ is a *K*-object X and a collection of K *–arrows* $\{\mu_i : D_i \to X \mid i \geq 0\}$ such that $\mu_i = \mu_{i+1} o\, f_i$ for all $i \geq 0$. We sometimes write $\mu : \Delta \to X$ as a reminder of the arrangement of μ's components Similarly, a colimit $\mu : \Delta \to X$ is a cocone with the property that if $v : \Delta \to X^{'}$ is also a cocone then there exists a unique mediating arrow $k: X \to X^{'}$ such that for all $i \geq 0,, v_i = k\, o\, \mu_i$. Colimits of $\omega - chains$ are sometimes referred to as $\omega - co\lim its$. Dually, an $\omega^{op} - chain$ in ***K*** is a diagram of the following form:

$\Delta = D_o \xleftarrow{f_o} D_1 \xleftarrow{f_1} D_2 \xleftarrow{f_2} .....$ A cone $\mu : X \to \Delta$ of an $\omega^{op} - chain\ \Delta$ is a ***K***-object X and a collection of **K**-arrows $\{\mu_i : D_i \mid i \geq 0\}$ such that for all $i \geq 0,\ \mu_i = f_i\, o\, \mu_{i+1}$. An ω^{op}-limit of an $\omega^{op} - chain\ \ \Delta$ is a cone $\mu : X \to \Delta$ with the property that if $v : X^{'} \to \Delta$ is also a cone, then there exists a unique mediating arrow $k : X^{'} \to X$ such that for all $i \geq 0, \mu_i\, o\, k = v_i$. We write $\perp_k$ (or just $\perp$) for the distinguish initial object of ***K***, when it has one, and $\perp \to A$ for the unique arrow from $\perp$ to each ***K***-object A. It is also convenient to write $\Delta^{-} = D_1 \xrightarrow{f_1} D_2 \xrightarrow{f_2} .....$ to denote all of Δ except D_o and f_0. By analogy, μ^{-} is $\{\mu_i \mid i \geq 1\}$. For the images of Δ and μ under ***F*** we write

$F(\Delta) = F(D_o) \xrightarrow{F(f_o)} F(D_1) \xrightarrow{F(f_1)} F(D_2) \xrightarrow{F(f_2)} .....$

and $F(\mu) = \{F(\mu_i) \mid i \geq 0\}$

We write F^i for the ***i***-fold iterated composition of ***F*** – that is, $F^o(f) = f, F^1(f) = F(f), F^2(f) = F(F(f))$,etc. With these definitions we can state that every monitonic function on a complete lattice has a least fixed point:

Lemma 1.4. Let ***K*** be a category with initial object $\perp$ and let $F: K \to K$ be a functor. Define the $\omega - chain\, \Delta$ by

$\Delta = \perp \xrightarrow{!\perp \to F(\perp)} F(\perp) \xrightarrow{F(!\perp \to F(\perp))} F^2(\perp) \xrightarrow{F^2(!\perp \to F(\perp))} ....$

If both $\mu : \Delta \to D$ and $F(\mu) : F(\Delta) \to F(D)$ are colimits, then (D,d) is an intial F-algebra, where $d : F(D) \to D$ is the mediating arrow from $F(\mu)$ to the cocone μ^{-}

Theorem 1.4 Let a DAG G given in which each node is a random variable, and let a discrete conditional probability distribution of each node given values of its parents in G be specified. Then the product of these conditional distributions yields a joint probability distribution P of the variables, and (G,P) satisfies the Markov condition.

Proof. Order the nodes according to an ancestral ordering. Let $X_1, X_2, ...X_n$ be the resultant ordering. Next define.

$$P(x_1,x_2,....x_n)=P(x_n|pa_n)P(x_{n-1}|Pa_{n-1})...$$
$$..P(x_2|pa_2)P(x_1|pa_1),$$

Where PA_i is the set of parents of X_i of in G and $P(x_i|pa_i)$ is the specified conditional probability distribution. First we show this does indeed yield a joint probability distribution. Clearly, $0\le P(x_1,x_2,...x_n)\le 1$ for all values of the variables. Therefore, to show we have a joint distribution, as the variables range through all their possible values, is equal to one. To that end, Specified conditional distributions are the conditional distributions they notationally represent in the joint distribution. Finally, we show the Markov condition is satisfied. To do this, we need show for $1\le k\le n$ that

$$P(pa_k)\neq 0, if\ \ P(nd_k|pa_k)\neq 0$$

whenever $\quad and\ \ P(x_k|pa_k)\neq 0$

$$then\ P(x_k|nd_k,pa_k)=P(x_k|pa_k),$$

Where ND_k is the set of nondescendents of X_k of in G. Since $PA_k\subseteq ND_k$, we need only show $P(x_k|nd_k)=P(x_k|pa_k)$. First for a given k, order the nodes so that all and only nondescendents of X_k precede X_k in the ordering. Note that this ordering depends on k, whereas the ordering in the first part of the proof does not. Clearly then

$$ND_k=\{X_1,X_2,....X_{k-1}\}$$

Let

$$D_k=\{X_{k+1},X_{k+2},....X_n\}$$

follows $\sum_{d_k}$

We define the m^{th} *cyclotomic field to be the field* $Q[x]/(\Phi_m(x))$ Where $\Phi_m(x)$ is the m^{th} cyclotomic polynomial. $Q[x]/(\Phi_m(x))$ $\Phi_m(x)$ *has degree* $\varphi(m)$ *over* Q *since* $\Phi_m(x)$ has degree $\varphi(m)$. *The roots of* $\Phi_m(x)$ are just the primitive m^{th} roots of unity, so the complex embeddings of $Q[x]/(\Phi_m(x))$ *are simply the* $\varphi(m)$ *maps*

$$\sigma_k:Q[x]/(\Phi_m(x))\mapsto C,$$
$$1\le k\prec m,(k,m)=1,\quad where$$
$$\sigma_k(x)=\xi_m^k,$$

ξ_m being our fixed choice of primitive m^{th} root of unity. Note that $\xi_m^k\in Q(\xi_m)$ for every k; it follows that $Q(\xi_m)=Q(\xi_m^k)$ for all k relatively prime to m. In particular, the images of the σ_i coincide, so $Q[x]/(\Phi_m(x))$ *is Galois over* Q. *This means that we can write* $Q(\xi_m)$ *for* $Q[x]/(\Phi_m(x))$ *without much fear of ambiguity; we will do so from now on, the identification being* $\xi_m\mapsto x$. *One advantage of this is that one can easily talk about cyclotomic fields being extensions of one another,or intersections or compositums; all of these things take place considering them as subfield of C.* We now investigate some basic properties of cyclotomic fields. The first issue is whether or not they are all distinct; to determine this, we need to know which roots of unity lie in $Q(\xi_m)$ .Note, for example, that if m is odd, then $-\xi_m$ is a $2m^{th}$ root of unity. We will show that this is the only way in which one can obtain any non- m^{th} roots of unity.

LEMMA 1.5 If m divides n, then $Q(\xi_m)$ *is contained in* $Q(\xi_n)$

PROOF. Since $\xi^{n/m}=\xi_m$, *we have* $\xi_m\in Q(\xi_n)$, *so the result is clear*

LEMMA 1.6 If m and n are relatively prime, then

$$Q(\xi_m,\xi_n)=Q(\xi_{nm})$$

and

$$Q(\xi_m)\cap Q(\xi_n)=Q$$

(Recall the $Q(\xi_m,\xi_n)$ is the compositum of $Q(\xi_m)\ and\ Q(\xi_n)$)

PROOF. One checks easily that $\xi_m\xi_n$ is a primitive mn^{th} root of unity, so that

$$Q(\xi_{mn}) \subseteq Q(\xi_m,\xi_n)$$

$$\left[Q(\xi_m,\xi_n):Q\right] \le \left[Q(\xi_m):Q\right]\left[Q(\xi_n:Q\right]$$

$$= \varphi(m)\varphi(n) = \varphi(mn);$$

Since $\left[Q(\xi_{mn}):Q\right] = \varphi(mn);$ this implies that $Q(\xi_m,\xi_n)=Q(\xi_{nm})$ We know that $Q(\xi_m,\xi_n)$ has degree $\varphi(mn)$ over Q, so we must have

$$\left[Q(\xi_m,\xi_n):Q(\xi_m)\right] = \varphi(n)$$

and

$$\left[Q(\xi_m,\xi_n):Q(\xi_m)\right] = \varphi(m)$$

$$\left[Q(\xi_m):Q(\xi_m)\cap Q(\xi_n)\right] \ge \varphi(m)$$

And thus that $Q(\xi_m)\cap Q(\xi_n) = Q$

PROPOSITION 1.2 For any m and n

$$Q(\xi_m,\xi_n)=Q(\xi_{[m,n]})$$

And

$$Q(\xi_m)\cap Q(\xi_n)=Q(\xi_{(m,n)});$$

here $[m,n]$ and (m,n) denote the least common multiple and the greatest common divisor of m and n, respectively.

PROOF. Write $m = p_1^{e_1}......p_k^{e_k}\ and\ p_1^{f_1}....p_k^{f_k}$ where the p_i are distinct primes. (We allow $e_i\ or\ f_i$ to be zero)

$$Q(\xi_m)=Q(\xi_{p_1^{e_1}})Q(\xi_{p_2^{e_2}})...Q(\xi_{p_k^{e_k}})$$

and

$$Q(\xi_n)=Q(\xi_{p_1^{f_1}})Q(\xi_{p_2^{f_2}})...Q(\xi_{p_k^{f_k}})$$

Thus

$$\begin{aligned} Q(\xi_m,\xi_n) &= Q(\xi_{p_1^{e_1}})........Q(\xi_{p_2^{e_k}})Q(\xi_{p_1^{f_1}})...Q(\xi_{p_k^{f_k}}) \\ &= Q(\xi_{p_1^{e_1}})Q(\xi_{p_1^{f_1}})...Q(\xi_{p_k^{e_k}})Q(\xi_{p_k^{f_k}}) \\ &= Q(\xi_{p_1^{\max(e_1,f_1)}}).......Q(\xi_{p_1^{\max(e_k,f_k)}}) \\ &= Q(\xi_{p_1^{\max(e_1,f_1)}........p_1^{\max(e_k,f_k)}}) \\ &= Q(\xi_{[m,n]}); \end{aligned}$$

An entirely similar computation shows that $Q(\xi_m)\cap Q(\xi_n) = Q(\xi_{(m,n)})$

Mutual information measures the information transferred when x_i is sent and y_i is received, and is defined as

$$I(x_i,y_i)=\log_2 \frac{P(x_i/y_i)}{P(x_i)}\ bits \qquad (1)$$

In a noise-free channel, **each** y_i is uniquely connected to the corresponding x_i , and so they constitute an input –output pair (x_i,y_i) for which

$P(x_i/y_j)=1\ and\ I(x_i,y_j)=\log_2\dfrac{1}{P(x_i)}$ bits; that is, the transferred information is equal to the self-information that corresponds to the input x_i

In a very noisy channel, the output y_i and input x_i would be completely uncorrelated, and so $P(x_i/y_j)=P(x_i)$ and also $I(x_i,y_j)=0;$ that is, there is no transference of information. In general, a given channel will operate between these two extremes. The mutual information is defined between the input and the output of a given channel. An average of the calculation of the mutual information for all input-output pairs of a given channel is the average mutual information:

$$I(X,Y)=\sum_{i.j}P(x_i,y_j)I(x_i,y_j)=\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{P({}^{x_i}\!/_{y_j})}{P(x_i)}\right] \text{ bits}$$

per symbol . This calculation is done over the input and output alphabets. The average mutual information. The following expressions are useful for modifying the mutual information expression:

$$P(x_i,y_j)=P({}^{x_i}\!/_{y_j})P(y_j)=P({}^{y_j}\!/_{x_i})P(x_i)$$

$$P(y_j)=\sum_i P({}^{y_j}\!/_{x_i})P(x_i)$$

$$P(x_i)=\sum_i P({}^{x_i}\!/_{y_j})P(y_j)$$

Then

$$I(X,Y)=\sum_{i.j}P(x_i,y_j)$$

$$=\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{1}{P(x_i)}\right]$$

$$-\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{1}{P({}^{x_i}\!/_{y_j})}\right]$$

$$\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{1}{P(x_i)}\right]$$

$$=\sum_i\left[P({}^{x_i}\!/_{y_j})P(y_j)\right]\log_2\frac{1}{P(x_i)}$$

$$\sum_i P(x_i)\log_2\frac{1}{P(x_i)}=H(X)$$

$$I(X,Y)=H(X)-H({}^{X}\!/_{Y})$$

Where $H({}^{X}\!/_{Y})=\sum_{i,j}P(x_i,y_j)\log_2\dfrac{1}{P({}^{x_i}\!/_{y_j})}$ is usually called the equivocation. In a sense, the equivocation can be seen as the information lost in the noisy channel, and is a function of the backward conditional probability. The observation of an output symbol y_j provides $H(X)-H({}^{X}\!/_{Y})$ bits of information. This difference is the mutual information of the channel. *Mutual Information: Properties* Since

$$P({}^{x_i}\!/_{y_j})P(y_j)=P({}^{y_j}\!/_{x_i})P(x_i)$$

The mutual information fits the condition

$$I(X,Y)=I(Y,X)$$

And by interchanging input and output it is also true that

$$I(X,Y)=H(Y)-H({}^{Y}\!/_{X})$$

Where

$$H(Y)=\sum_j P(y_j)\log_2\frac{1}{P(y_j)}$$

This last entropy is usually called the noise entropy. Thus, the information transferred through the channel is the difference between the output entropy and the noise entropy. Alternatively, it can be said that the channel mutual information is the difference between the number of bits needed for determining a given input symbol before knowing the corresponding output symbol, and the number of bits needed for determining a given input symbol after knowing the corresponding output symbol

$$I(X,Y)=H(X)-H({}^{X}\!/_{Y})$$

As the channel mutual information expression is a difference between two quantities, it seems that this parameter can adopt negative values. However, and is spite of the fact that for some $y_j, H(X/y_j)$ can be larger than $H(X)$, this is not possible for the average value calculated over all the outputs:

$$\sum_{i,j}P(x_i,y_j)\log_2\frac{P({}^{x_i}\!/_{y_j})}{P(x_i)}=\sum_{i,j}P(x_i,y_j)\log_2\frac{P(x_i,y_j)}{P(x_i)P(y_j)}$$

Then

$$-I(X,Y)=\sum_{i,j}P(x_i,y_j)\frac{P(x_i)P(y_j)}{P(x_i,y_j)}\leq 0$$

Because this expression is of the form

$$\sum_{i=1}^{M} P_i \log_2(\frac{Q_i}{P_i}) \leq 0$$

The above expression can be applied due to the factor $P(x_i)P(y_j)$, which is the product of two probabilities, so that it behaves as the quantity Q_i , which in this expression is a dummy variable that fits the condition $\sum_i Q_i \leq 1$. It can be concluded that the average mutual information is a non-negative number. It can also be equal to zero, when the input and the output are independent of each other. A related entropy called the joint entropy is defined as

$$H(X,Y) = \sum_{i,j} P(x_i, y_j) \log_2 \frac{1}{P(x_i, y_j)}$$

$$= \sum_{i,j} P(x_i, y_j) \log_2 \frac{P(x_i)P(y_j)}{P(x_i, y_j)}$$

$$+ \sum_{i,j} P(x_i, y_j) \log_2 \frac{1}{P(x_i)P(y_j)}$$

Theorem 1.5: Entropies of the binary erasure channel (BEC) The BEC is defined with an alphabet of two inputs and three outputs, with symbol probabilities.

$P(x_1) = \alpha \;\; and \;\; P(x_2) = 1 - \alpha,$ and transition probabilities

$$P({}^{y_3}\!/\!_{x_2}) = 1 - p \;\; and \;\; P({}^{y_2}\!/\!_{x_1}) = 0,$$

$$and \;\; P({}^{y_3}\!/\!_{x_1}) = 0$$

$$and \;\; P({}^{y_1}\!/\!_{x_2}) = p$$

$$and \;\; P({}^{y_3}\!/\!_{x_2}) = 1 - p$$

Lemma 1.7. Given an arbitrary restricted time-discrete, amplitude-continuous channel whose restrictions are determined by sets F_n and whose density functions exhibit no dependence on the state s , let n be a fixed positive integer, and $p(x)$ an arbitrary probability density function on Euclidean n-space. $p(y|x)$ for the density $p_n(y_1,...,y_n | x_1,...x_n)$ and $F \;\; for \;\; F_n$. For any real number a, let

$$A = \left\{ (x,y) : \log \frac{p(y|x)}{p(y)} > a \right\} \qquad (1)$$

Then for each positive integer u , there is a code (u, n, λ) such that

$$\lambda \leq ue^{-a} + P\{(X,Y) \notin A\} + P\{X \notin F\} \qquad (2)$$

Where

$$P\{(X,Y) \in A\} = \int_A ... \int p(x,y)dxdy, \qquad p(x,y) = p(x)p(y|x)$$

and

$$P\{X \in F\} = \int_F ... \int p(x)dx$$

Proof: A sequence $x^{(1)} \in F$ *such that*

$$P\left\{Y \in A_{x^1} \,|\, X = x^{(1)}\right\} \geq 1 - \varepsilon$$

$where \;\; A_x = \{y : (x,y) \varepsilon A\};$

Choose the decoding set B_1 to be $A_{x^{(1)}}$. Having chosen $x^{(1)},........,x^{(k-1)}$ and $B_1,...,B_{k-1}$, select $x^k \in F$ such that

$$P\left\{Y \in A_{x^{(k)}} - \bigcup_{i=1}^{k-1} B_i \,|\, X = x^{(k)}\right\} \geq 1 - \varepsilon;$$

Set $B_k = A_{x^{(k)}} - \bigcup_{i=1}^{k-1} B_i$, If the process does not terminate in a finite number of steps, then the sequences $x^{(i)}$ and decoding sets $B_i, \; i = 1, 2, ..., u,$ form the desired code. Thus assume that the process terminates after t steps. (Conceivably $t = 0$). We will show $t \geq u$ by showing that $\varepsilon \leq te^{-a} + P\{(X,Y) \notin A\} + P\{X \notin F\}$. We proceed as follows.

$$B = \bigcup_{j=1}^{t} B_j. \quad (If \;\; t = 0, \;\; take \;\; B = \phi). \;\; Then$$

$$P\{(X,Y) \in A\} = \int_{(x,y) \in A} p(x,y)dx\,dy$$

Let

$$= \int_x p(x) \int_{y \in A_x} p(y|x)dy\,dx$$

$$= \int_x p(x) \int_{y \in B \cap A_x} p(y|x)dy\,dx + \int_x p(x)$$

C. Algorithms

Ideals. Let A be a ring. Recall that an *ideal a* in A is a subset such that a is subgroup of A regarded as a group under addition;

$a \in a, r \in A \Rightarrow ra \in A$

The ideal generated by a subset S of A is the intersection of all ideals A containing a ----- it is easy to verify that this is in fact an ideal, and that it consist of all finite sums of the form $\sum r_i s_i$ with $r_i \in A, s_i \in S$. When $S = \{s_1,.....,s_m\}$, we shall write $(s_1,.....,s_m)$ for the ideal it generates.

Let a and b be ideals in A. The set $\{a+b \mid a \in a, b \in b\}$ is an ideal, denoted by $a+b$. The ideal generated by $\{ab \mid a \in a, b \in b\}$ is denoted by ab. Note that $ab \subset a \cap b$. Clearly ab consists of all finite sums $\sum a_i b_i$ with $a_i \in a$ and $b_i \in b$, and if $a = (a_1,...,a_m)$ and $b = (b_1,...,b_n)$, then $ab = (a_1b_1,...,a_ib_j,...,a_mb_n)$ .Let a be an ideal of A. The set of cosets of a in A forms a ring A/a, and $a \mapsto a+a$ is a homomorphism $\phi : A \mapsto A/a$. The map $b \mapsto \phi^{-1}(b)$ is a one to one correspondence between the ideals of A/a and the ideals of A containing a An ideal p if *prime* if $p \neq A$ and $ab \in p \Rightarrow a \in p$ or $b \in p$. Thus p is prime if and only if A/p is nonzero and has the property that $ab = 0, \quad b \neq 0 \Rightarrow a = 0,$ i.e., A/p is an integral domain. An ideal m is *maximal* if $m \neq| A$ and there does not exist an ideal n contained strictly between m and A. Thus m is maximal if and only if A/m has no proper nonzero ideals, and so is a field. Note that m maximal $\Rightarrow$ m prime. The ideals of $A \times B$ are all of the form $a \times b$, with a and b ideals in A and B. To see this, note that if c is an ideal in $A \times B$ and $(a,b) \in c$, then $(a,0) = (a,b)(1,0) \in c$ and $(0,b) = (a,b)(0,1) \in c$. This shows that $c = a \times b$ with

$a = \{a \mid (a,b) \in c \;\; some \;\; b \in b\}$

and

$b = \{b \mid (a,b) \in c \;\; some \;\; a \in a\}$

Let A be a ring. An A-algebra is a ring B together with a homomorphism $i_B : A \to B$. A *homomorphism of* A-algebra $B \to C$ is a homomorphism of rings $\varphi : B \to C$ such that $\varphi(i_B(a)) = i_C(a)$ for all $a \in A$. An A-algebra B is said to be *finitely generated* (or of *finite-type* over A) if there exist elements $x_1,...,x_n \in B$ such that every element of B can be expressed as a polynomial in the x_i with coefficients in $i(A)$, i.e., such that the homomorphism $A[X_1,...,X_n] \to B$ sending X_i to x_i is surjective. A ring homomorphism $A \to B$ is *finite*, and B is finitely generated as an A-module. Let k be a field, and let A be a k-algebra. If $1 \neq 0$ in A, then the map $k \to A$ is injective, we can identify k with its image, i.e., we can regard k as a subring of A . If 1=0 in a ring R, the R is the zero ring, i.e., $R = \{0\}$.

Polynomial rings. Let k be a field. A *monomial* in $X_1,...,X_n$ is an expression of the form $X_1^{a_1}...X_n^{a_n}$, $\quad a_j \in N$. The *total degree* of the monomial is $\sum a_i$. We sometimes abbreviate it by $X^{\alpha}, \alpha = (a_1,...,a_n) \in \square^n$. The elements of the polynomial ring $k[X_1,...,X_n]$ are finite sums $\sum c_{a_1....a_n} X_1^{a_1}...X_n^{a_n}, \quad c_{a_1....a_n} \in k, \quad a_j \in \square$
With the obvious notions of equality, addition and multiplication. Thus the monomials from basis for $k[X_1,...,X_n]$ as a k-vector space. The ring $k[X_1,...,X_n]$ is an integral domain, and the only units in it are the nonzero constant polynomials. A polynomial $f(X_1,...,X_n)$ is *irreducible* if it is nonconstant and has only the obvious factorizations, i.e., $f = gh \Rightarrow g$ or h is constant. **Division in** $k[X]$. The division

algorithm allows us to divide a nonzero polynomial into another: let f and g be polynomials in $k[X]$ with $g \neq 0$; then there exist unique polynomials $q, r \in k[X]$ such that $f = qg + r$ with either $r = 0$ or $\deg r < \deg g$. Moreover, there is an algorithm for deciding whether $f \in (g)$, namely, find r and check whether it is zero. Moreover, the Euclidean algorithm allows to pass from finite set of generators for an ideal in $k[X]$ to a single generator by successively replacing each pair of generators with their greatest common divisor.

(Pure) ***lexicographic*** *ordering (lex)***.** Here monomials are ordered by lexicographic(dictionary) order. More precisely, let $\alpha = (a_1,...a_n)$ and $\beta = (b_1,...b_n)$ be two elements of □^{n}; then $\alpha > \beta$ *and* $X^\alpha > X^\beta$ (lexicographic ordering) if, in the vector difference $\alpha - \beta \in$ □, the left most nonzero entry is positive. For example, $XY^2 > Y^3Z^4$; $X^3Y^2Z^4 > X^3Y^2Z$. Note that this isn't quite how the dictionary would order them: it would put *XXXYYZZZZ* after *XXXYYZ*. *Graded reverse lexicographic order (grevlex).* Here monomials are ordered by total degree, with ties broken by reverse lexicographic ordering. Thus, $\alpha > \beta$ if $\sum a_i > \sum b_i$, or $\sum a_i = \sum b_i$ and in $\alpha - \beta$ the right most nonzero entry is negative. For example:

$X^4Y^4Z^7 > X^5Y^5Z^4$ *(total degree greater)*

$XY^5Z^2 > X^4YZ^3, \quad X^5YZ > X^4YZ^2.$

Orderings on $k[X_1,...X_n]$. Fix an ordering on the monomials in $k[X_1,...X_n]$. Then we can write an element f of $k[X_1,...X_n]$ in a canonical fashion, by re-ordering its elements in decreasing order. For example, we would write

$f = 4XY^2Z + 4Z^2 - 5X^3 + 7X^2Z^2$

as

$f = -5X^3 + 7X^2Z^2 + 4XY^2Z + 4Z^2 \quad (lex)$

or

$f = 4XY^2Z + 7X^2Z^2 - 5X^3 + 4Z^2 \quad (grevlex)$

Let $\sum a_\alpha X^\alpha \in k[X_1,...,X_n]$, in decreasing order:

$f = a_{\alpha_0} X^{\alpha_0} +_{\alpha_1} X^{\alpha_1} + ..., \qquad \alpha_0 > \alpha_1 > ..., \quad \alpha_0 \neq 0$

Then we define.

- The *multidegree* of f to be multdeg(f)= α_0 ;
- The *leading coefficient of* f to be *LC(* f *)=* a_{α_0} ;
- The *leading monomial of* f to be *LM(* f *)* = X^{α_0} ;
- The *leading term of* f to be *LT(* f *)* = $a_{\alpha_0} X^{\alpha_0}$

For the polynomial $f = 4XY^2Z + ...$, the multidegree is (1,2,1), the leading coefficient is 4, the leading monomial is XY^2Z, and the leading term is $4XY^2Z$. **The division algorithm in $k[X_1,...X_n]$.** Fix a monomial ordering in □^{2}. Suppose given a polynomial f and an ordered set $(g_1,...g_s)$ of polynomials; the division algorithm then constructs polynomials $a_1,...a_s$ and r such that $f = a_1g_1 + ... + a_sg_s + r$ Where either $r = 0$ or no monomial in r is divisible by any of $LT(g_1),...,LT(g_s)$ **Step 1:** If $LT(g_1) | LT(f)$, divide g_1 into f to get

$$f = a_1g_1 + h, \qquad a_1 = \frac{LT(f)}{LT(g_1)} \in k[X_1,...,X_n]$$

If $LT(g_1) | LT(h)$, repeat the process until $f = a_1g_1 + f_1$ (different a_1) with $LT(f_1)$ not divisible by $LT(g_1)$. Now divide g_2 into f_1, and so on, until $f = a_1g_1 + ... + a_sg_s + r_1$ With $LT(r_1)$ not divisible by any $LT(g_1),...LT(g_s)$ **Step 2:** Rewrite $r_1 = LT(r_1) + r_2$, and repeat Step 1 with r_2 for f : $f = a_1g_1 + ... + a_sg_s + LT(r_1) + r_3$ (different a_i's

) **Monomial ideals.** In general, an ideal a will contain a polynomial without containing the individual terms of the polynomial; for example, the ideal $a=(Y^2-X^3)$ contains Y^2-X^3but not Y^2 or X^3.

DEFINITION 1.5. An ideal a is *monomial* if $\sum c_\alpha X^\alpha \in a \Rightarrow X^\alpha \in a$

all α with $c_\alpha \neq 0$.

PROPOSITION 1.3. Let a be a *monomial ideal,* and let $A=\{\alpha \mid X^\alpha \in a\}$. Then A satisfies the condition $\alpha \in A,\ \beta \in \square^n \Rightarrow \alpha+\beta \in$ (*)

And a is the k-subspace of $k[X_1,\dots,X_n]$ generated by the $X^\alpha, \alpha \in A$. Conversely, of A is a subset of $\square^n$ satisfying $(*)$, then the k-subspace a of $k[X_1,\dots,X_n]$ generated by $\{X^\alpha \mid \alpha \in A\}$ is a monomial ideal.

PROOF. It is clear from its definition that a monomial ideal a is the k-subspace of $k[X_1,\dots,X_n]$

generated by the set of monomials it contains. If $X^\alpha \in a$ and $X^\beta \in k[X_1,\dots,X_n]$.

If a permutation is chosen uniformly and at random from the $n!$ possible permutations in S_n, then the counts $C_j^{(n)}$ of cycles of length j are dependent random variables. The joint distribution of $C^{(n)}=(C_1^{(n)},\dots,C_n^{(n)})$ follows from Cauchy's formula, and is given by

$$P[C^{(n)}=c]=\frac{1}{n!}N(n,c)=1\left\{\sum_{j=1}^n jc_j=n\right\}\prod_{j=1}^n\left(\frac{1}{j}\right)^{c_j}\frac{1}{c_j!}, \qquad (1.1)$$

for $c \in \square^n_+$.

Lemma1.7 For nonnegative integers $m_{1,\dots,}m_n$,

$$E\left(\prod_{j=1}^n (C_j^{(n)})^{[m_j]}\right)=\left(\prod_{j=1}^n\left(\frac{1}{j}\right)^{m_j}\right)1\left\{\sum_{j=1}^n jm_j \le n\right\} \qquad (1.4)$$

Proof. This can be established directly by exploiting cancellation of the form $c_j^{[m_j]}/c_j^! = 1/(c_j-m_j)!$ when $c_j \ge m_j$, which occurs between the ingredients in Cauchy's formula and the falling factorials in the moments. Write $m=\sum jm_j$. Then, with the first sum indexed by $c=(c_1,\dots c_n)\in \square^n_+$ and the last sum indexed by $d=(d_1,\dots,d_n)\in \square^n_+$ via the correspondence $d_j=c_j-m_j$, we have

$$E\left(\prod_{j=1}^n (C_j^{(n)})^{[m_j]}\right)=\sum_c P[C^{(n)}=c]\prod_{j=1}^n (c_j)^{[m_j]}$$

$$=\sum_{c:c_j\ge m_j \ \ for\ all\ \ j} 1\left\{\sum_{j=1}^n jc_j=n\right\}\prod_{j=1}^n\frac{(c_j)^{[m_j]}}{j^{c_j}c_j!}$$

$$=\prod_{j=1}^n\frac{1}{j^{m_j}}\sum_d 1\left\{\sum_{j=1}^n jd_j=n-m\right\}\prod_{j=1}^n\frac{1}{j^{d_j}(d_j)!}$$

This last sum simplifies to the indicator $1(m\le n)$, corresponding to the fact that if $n-m\ge 0$, then $d_j=0$ for $j>n-m$, and a random permutation in S_{n-m} must have some cycle structure $(d_1,\dots,d_{n-m})$. The moments of $C_j^{(n)}$ follow immediately as

$$E(C_j^{(n)})^{[r]}=j^{-r}1\{jr\le n\} \qquad (1.2)$$

We note for future reference that (1.4) can also be written in the form

$$E\left(\prod_{j=1}^n (C_j^{(n)})^{[m_j]}\right)=E\left(\prod_{j=1}^n Z_j^{[m_j]}\right)1\left\{\sum_{j=1}^n jm_j\le n\right\}, \qquad (1.3)$$

Where the Z_j are independent Poisson-distribution random variables that satisfy $E(Z_j)=1/j$

The marginal distribution of cycle counts provides a formula for the joint distribution of the cycle counts C_j^n, we find the distribution of

C_j^n using a combinatorial approach combined with the inclusion-exclusion formula.

Lemma 1.8. For $1 \le j \le n$,

$$P[C_j^{(n)} = k] = \frac{j^{-k}}{k!} \sum_{l=0}^{[n/j]-k} (-1)^l \frac{j^{-l}}{l!} \qquad (1.1)$$

Proof. Consider the set I of all possible cycles of length j, formed with elements chosen from $\{1, 2, \ldots n\}$, so that $|I| = n^{[j]/j}$. For each $\alpha \in I$, consider the "property" G_α of having α; that is, G_α is the set of permutations $\pi \in S_n$ such that α is one of the cycles of π. We then have $|G_\alpha| = (n-j)!$, since the elements of $\{1, 2, \ldots, n\}$ not in α must be permuted among themselves. To use the inclusion-exclusion formula we need to calculate the term S_r, which is the sum of the probabilities of the r-fold intersection of properties, summing over all sets of r distinct properties. There are two cases to consider. If the r properties are indexed by r cycles having no elements in common, then the intersection specifies how rj elements are moved by the permutation, and there are $(n-rj)!1(rj \le n)$ permutations in the intersection. There are $n^{[rj]}/(j^r r!)$ such intersections. For the other case, some two distinct properties name some element in common, so no permutation can have both these properties, and the r-fold intersection is empty. Thus

$$S_r = (n-rj)!1(rj \le n)$$

$$\times \frac{n^{[rj]}}{j^r r!} \frac{1}{n!} = 1(rj \le n) \frac{1}{j^r r!}$$

Finally, the inclusion-exclusion series for the number of permutations having exactly k properties is

$$\sum_{l \ge 0} (-1)^l \binom{k+l}{l} S_{k+l},$$

Which simplifies to (1.1) Returning to the original hat-check problem, we substitute j=1 in (1.1) to obtain the distribution of the number of fixed points of a random permutation. For $k = 0, 1, \ldots, n$,

$$P[C_1^{(n)} = k] = \frac{1}{k!} \sum_{l=0}^{n-k} (-1)^l \frac{1}{l!}, \qquad (1.2)$$

and the moments of $C_1^{(n)}$ follow from (1.2) with $j = 1$. In particular, for $n \ge 2$, the mean and variance of $C_1^{(n)}$ are both equal to 1. The joint distribution of $(C_1^{(n)}, \ldots, C_b^{(n)})$ for any $1 \le b \le n$ has an expression similar to (1.7); this too can be derived by inclusion-exclusion. For any $c = (c_1, \ldots, c_b) \in \square_+^b$ with $m = \sum ic_i$,

$$P[(C_1^{(n)}, \ldots, C_b^{(n)}) = c]$$

$$= \left\{ \prod_{i=1}^{b} \left(\frac{1}{i}\right)^{c_i} \frac{1}{c_i!} \right\} \sum_{\substack{l \ge 0 \text{ with} \\ \sum il_i \le n-m}} (-1)^{l_1 + \ldots + l_b} \prod_{i=1}^{b} \left(\frac{1}{i}\right)^{l_i} \frac{1}{l_i!} \qquad (1.3)$$

The joint moments of the first b counts $C_1^{(n)}, \ldots, C_b^{(n)}$ can be obtained directly from (1.2) and (1.3) by setting $m_{b+1} = \ldots = m_n = 0$

The limit distribution of cycle counts

It follows immediately from Lemma 1.2 that for each fixed j, as $n \to \infty$,

$$P[C_j^{(n)} = k] \to \frac{j^{-k}}{k!} e^{-1/j}, \quad k = 0, 1, 2, \ldots,$$

So that $C_j^{(n)}$ converges in distribution to a random variable Z_j having a Poisson distribution with mean $1/j$; we use the notation $C_j^{(n)} \to_d Z_j$ where $Z_j \square P_o(1/j)$ to describe this. Infact, the limit random variables are independent.

Theorem 1.6 The process of cycle counts converges in distribution to a Poisson process of $\square$ with intensity j^{-1}. That is, as $n \to \infty$,

$$(C_1^{(n)}, C_2^{(n)}, \ldots) \to_d (Z_1, Z_2, \ldots) \qquad (1.1)$$

Where the $Z_j, j = 1, 2, ...,$ are independent Poisson-distributed random variables with $E(Z_j) = \frac{1}{j}$

Proof. To establish the converges in distribution one shows that for each fixed $b \geq 1$, as $n \to \infty$,

$$P[(C_1^{(n)}, ..., C_b^{(n)}) = c] \to P[(Z_1, ..., Z_b) = c]$$

Error rates

The proof of Theorem says nothing about the rate of convergence. Elementary analysis can be used to estimate this rate when $b = 1$. Using properties of alternating series with decreasing terms, for $k = 0, 1, ..., n,$

$$\frac{1}{k!}(\frac{1}{(n-k+1)!} - \frac{1}{(n-k+2)!}) \leq \left| P[C_1^{(n)} = k] - P[Z_1 = k] \right|$$
$$\leq \frac{1}{k!(n-k+1)!}$$

It follows that

$$\frac{2^{n+1}}{(n+1)!}\frac{n}{n+2} \leq \sum_{k=0}^{n} \left| P[C_1^{(n)} = k] - P[Z_1 = k] \right| \leq \frac{2^{n+1} - 1}{(n+1)!} \qquad (1.11)$$

Since

$$P[Z_1 > n] = \frac{e^{-1}}{(n+1)!}(1 + \frac{1}{n+2} + \frac{1}{(n+2)(n+3)} + ...) < \frac{1}{(n+1)!},$$

We see from (1.11) that the total variation distance between the distribution $L(C_1^{(n)})$ of $C_1^{(n)}$ and the distribution $L(Z_1)$ of Z_1

Establish the asymptotics of $\mathrm{P}\left[A_n(C^{(n)})\right]$ under conditions (A_0) and (B_{01}), where

$$A_n(C^{(n)}) = \bigcap_{1 \leq i \leq n} \bigcap_{r_i^{'} + 1 \leq j \leq r_i} \left\{ C_{ij}^{(n)} = 0 \right\},$$

and $\zeta_i = (r_i^{'} / r_{id}) - 1 = O(i^{-g^{'}})$ as $i \to \infty$, for some $g^{'} > 0$. We start with the expression

$$P[A_n(C^{(n)})] = \frac{P[T_{0m}(Z^{'}) = n]}{P[T_{0m}(Z) = n]}$$
$$\prod_{\substack{1 \leq i \leq n \\ r_i^{'} + 1 \leq j \leq r_i}} \left\{ 1 - \frac{\theta}{ir_i}(1 + E_{i0}) \right\} \qquad (1.1)$$

$$P[T_{0n}(Z^{'}) = n]$$
$$= \frac{\theta d}{n} \exp\left\{ \sum_{i \geq 1} [\log(1 + i^{-1}\theta d) - i^{-1}\theta d] \right\}$$
$$\left\{ 1 + O(n^{-1}\varphi_{\{1,2,7\}}^{'}(n)) \right\} \qquad (1.2)$$

and

$$P[T_{0n}(Z^{'}) = n]$$
$$= \frac{\theta d}{n} \exp\left\{ \sum_{i \geq 1} [\log(1 + i^{-1}\theta d) - i^{-1}\theta d] \right\}$$
$$\left\{ 1 + O(n^{-1}\varphi_{\{1,2,7\}}(n)) \right\} \qquad (1.3)$$

Where $\varphi_{\{1,2,7\}}^{'}(n)$ refers to the quantity derived from $Z^{'}$. It thus follows that $P[A_n(C^{(n)})] \square Kn^{-\theta(1-d)}$ for a constant K, depending on Z and the $r_i^{'}$ and computable explicitly from (1.1) – (1.3), if Conditions (A_0) and (B_{01}) are satisfied and if $\zeta_i^* = O(i^{-g^{'}})$ from some $g^{'} > 0$, since, under these circumstances, both $n^{-1}\varphi_{\{1,2,7\}}^{'}(n)$ and $n^{-1}\varphi_{\{1,2,7\}}(n)$ tend to zero as $n \to \infty$. In particular, for polynomials and square free polynomials, the relative error in this asymptotic approximation is of order n^{-1} if $g^{'} > 1$.

For $0 \leq b \leq n/8$ and $n \geq n_0$, with n_0

$$d_{TV}(L(C[1,b]), L(Z[1,b]))$$
$$\leq d_{TV}(L(\overset{\square}{C}[1,b]), L(\overset{\square}{Z}[1,b]))$$
$$\leq \varepsilon_{\{7,7\}}(n,b),$$

Where $\varepsilon_{\{7,7\}}(n,b) = O(b/n)$ under Conditions $(A_0), (D_1)$ and (B_{11}) Since, by the Conditioning

Relation,

$L(\overset{\square}{C}[1,b] \mid T_{0b}(C)=l) = L(\overset{\square}{Z}[1,b] \mid T_{0b}(Z)=l),$

It follows by direct calculation that

$$d_{TV}(L(\overset{\square}{C}[1,b]), L(\overset{\square}{Z}[1,b]))$$
$$= d_{TV}(L(T_{0b}(C)), L(T_{0b}(Z)))$$
$$= \max_A \sum_{r\in A} P[T_{0b}(Z)=r]$$
$$\left\{1 - \frac{P[T_{bn}(Z)=n-r]}{P[T_{0n}(Z)=n]}\right\} \qquad (1.4)$$

Suppressing the argument Z from now on, we thus obtain

$$d_{TV}(L(\overset{\square}{C}[1,b]), L(\overset{\square}{Z}[1,b]))$$
$$= \sum_{r\geq 0} P[T_{0b}=r]\left\{1-\frac{P[T_{bn}=n-r]}{P[T_{0n}=n]}\right\}_+$$
$$\leq \sum_{r>n/2} P[T_{0b}=r] + \sum_{r=0}^{[n/2]} \frac{P[T_{0b}=r]}{P[T_{0b}=n]}$$
$$\times \left\{\sum_{s=0}^{n} P[T_{0b}=s](P[T_{bn}=n-s]-P[T_{bn}=n-r]\right\}_+$$
$$\leq \sum_{r>n/2} P[T_{0b}=r] + \sum_{r=0}^{[n/2]} P[T_{0b}=r]$$
$$\times \sum_{s=0}^{[n/2]} P[T_{0b}=s]\frac{\{P[T_{bn}=n-s]-P[T_{bn}=n-r]\}}{P[T_{0n}=n]}$$
$$+\sum_{s=0}^{[n/2]} P[T_{0b}=r] \sum_{s=[n/2]+1}^{n} P[T=s]P[T_{bn}=n-s]/P[T_{0n}=n]$$

The first sum is at most $2n^{-1}ET_{0b}$; the third is bound by

$$(\max_{n/2<s\leq n} P[T_{0b}=s])/P[T_{0n}=n]$$
$$\leq \frac{2\varepsilon_{\{10.5(1)\}}(n/2,b)}{n}\frac{3n}{\theta P_\theta[0,1]},$$
$$\frac{3n}{\theta P_\theta[0,1]} 4n^{-2}\phi^*_{\{10.8\}}(n)\sum_{r=0}^{[n/2]} P[T_{0b}=r]\sum_{s=0}^{[n/2]} P[T_{0b}=s]\frac{1}{2}|r-s|$$
$$\leq \frac{12\phi^*_{\{10.8\}}(n)}{\theta P_\theta[0,1]}\frac{ET_{0b}}{n}$$

Hence we may take

$$\varepsilon_{\{7,7\}}(n,b) = 2n^{-1}ET_{0b}(Z)\left\{1+\frac{6\phi^*_{\{10.8\}}(n)}{\theta P_\theta[0,1]}\right\}P$$
$$+\frac{6}{\theta P_\theta[0,1]}\varepsilon_{\{10.5(1)\}}(n/2,b) \qquad (1.5)$$

Required order under Conditions $(A_0),(D_1)$ and (B_{11}), if $S(\infty)<\infty$. If not, $\phi^*_{\{10.8\}}(n)$ can be replaced by $\phi^*_{\{10.11\}}(n)$ in the above, which has the required order, without the restriction on the r_i implied by $S(\infty)<\infty$. Examining the Conditions $(A_0),(D_1)$ and (B_{11}), it is perhaps surprising to find that (B_{11}) is required instead of just (B_{01}); that is, that we should need $\sum_{l\geq 2} l\varepsilon_{il} = O(i^{-a_1})$ to hold for some $a_1>1$. A first observation is that a similar problem arises with the rate of decay of ε_{i1} as well. For this reason, n_1 is replaced by $\overset{\square}{n}_1$. This makes it possible to replace condition (A_1) by the weaker pair of conditions (A_0) and (D_1) in the eventual assumptions needed for $\varepsilon_{\{7,7\}}(n,b)$ to be of order $O(b/n)$; the decay rate requirement of order $i^{-1-\gamma}$ is shifted from ε_{i1} itself to its first difference. This is needed to obtain the right approximation error for the random mappings example. However, since all the classical applications make far more stringent assumptions about the $\varepsilon_{i1}, l\geq 2$, than are made in (B_{11}). The critical point of the proof is seen where the initial estimate of the difference $P[T_{bn}^{(m)}=s]-P[T_{bn}^{(m)}=s+1]$. The factor $\varepsilon_{\{10.10\}}(n)$, which should be small, contains a far tail element from $\overset{\square}{n}_1$ of the form $\phi_1^\theta(n)+u_1^*(n)$, which is only small if $a_1>1$, being otherwise of order $O(n^{1-a_1+\delta})$ for any $\delta>0$, since $a_2>1$ is in any case assumed. For $s\geq n/2$, this gives rise to a contribution of order $O(n^{-1-a_1+\delta})$ in the estimate of the difference $P[T_{bn}=s]-P[T_{bn}=s+1]$, which, in the remainder of the proof, is translated into a contribution of order $O(tn^{-1-a_1+\delta})$ for differences of the form $P[T_{bn}=s]-P[T_{bn}=s+1]$, finally leading to a contribution of order $bn^{-a_1+\delta}$ for any $\delta>0$ in $\varepsilon_{\{7.7\}}(n,b)$. Some improvement would seem to be possible, defining the function g by $g(w)=1_{\{w=s\}}-1_{\{w=s+t\}}$, differences that are of the form $P[T_{bn}=s]-P[T_{bn}=s+t]$ can be directly estimated, at a cost of only a single contribution of the form $\phi_1^{\theta}(n)+u_1^{*}(n)$. Then, iterating the cycle, in which one estimate of a difference in point probabilities is improved to an estimate of smaller order, a bound of the form $\left|P[T_{bn}=s]-P[T_{bn}=s+t]\right|=O(n^{-2}t+n^{-1-a_1+\delta})$ for any $\delta>0$ could perhaps be attained, leading to a final error estimate in order $O(bn^{-1}+n^{-a_1+\delta})$ for any $\delta>0$, to replace $\varepsilon_{\{7.7\}}(n,b)$. This would be of the ideal order $O(b/n)$ for large enough b, but would still be coarser for small b.

With b and n as in the previous section, we wish to show that

$$\left|d_{TV}(L(C[1,b]),L(Z[1,b]))-\frac{1}{2}(n+1)^{-1}\left|1-\theta\right|E\left|T_{0b}-ET_{0b}\right|\right|$$
$$\le\varepsilon_{\{7,8\}}(n,b),$$

Where $\varepsilon_{\{7.8\}}(n,b)=O(n^{-1}b[n^{-1}b+n^{-\beta_{12}+\delta}])$ for any $\delta>0$ under Conditions $(A_0),(D_1)$ and (B_{12}), with β_{12} . The proof uses sharper estimates. As before, we begin with the formula

$$d_{TV}(L(\overset{\square}{C}[1,b]),L(\overset{\square}{Z}[1,b]))$$
$$=\sum_{r\ge0}P[T_{0b}=r]\left\{1-\frac{P[T_{bn}=n-r]}{P[T_{0n}=n]}\right\}_{+}$$

Now we observe that

$$\left|\sum_{r\ge0}P[T_{0b}=r]\left\{1-\frac{P[T_{bn}=n-r]}{P[T_{0n}=n]}\right\}_{+}-\sum_{r=0}^{[n/2]}\frac{P[T_{0b}=r]}{P[T_{0n}=n]}\right|$$
$$\times\left|\sum_{s=[n/2]+1}^{n}P[T_{0b}=s](P[T_{bn}=n-s]-P[T_{bn}=n-r])\right|$$
$$\le4n^{-2}ET_{0b}^{2}+(\max_{n/2<s\le n}P[T_{0b}=s])/P[T_{0n}=n]$$
$$+P[T_{0b}>n/2]$$
$$\le8n^{-2}ET_{0b}^{2}+\frac{3\varepsilon_{\{10.5(2)\}}(n/2,b)}{\theta P_{\theta}[0,1]},\qquad(1.1)$$

We have

$$\left|\sum_{r=0}^{[n/2]}\frac{P[T_{0b}=r]}{P[T_{0n}=n]}\right.$$
$$\times(\left\{\sum_{s=0}^{[n/2]}P[T_{0b}=s](P[T_{bn}=n-s]-P[T_{bn}=n-r]\right\}_{+}$$
$$\left.-\left\{\sum_{s=0}^{[n/2]}P[T_{0b}=s]\frac{(s-r)(1-\theta)}{n+1}P[T_{0n}=n]\right\}_{+})\ \right|$$

$$\le\frac{1}{n^{2}P[T_{0n}=n]}\sum_{r\ge0}P[T_{0b}=r]\sum_{s\ge0}P[T_{0b}=s]\left|s-r\right|$$
$$\times\left\{\varepsilon_{\{10.14\}}(n,b)+2(r\vee s)\left|1-\theta\right|n^{-1}\left\{K_0\theta+4\phi_{\{10.8\}}^{*}(n)\right\}\right\}$$
$$\le\frac{6}{\theta nP_{\theta}[0,1]}ET_{0b}\varepsilon_{\{10.14\}}(n,b)$$
$$+4\left|1-\theta\right|n^{-2}ET_{0b}^{2}\left\{K_0\theta+4\phi_{\{10.8\}}^{*}(n)\right\}$$
$$(\frac{3}{\theta nP_{\theta}[0,1]})\ \},\qquad(1.2)$$

The approximation in (1.2) is further simplified by noting that

$$\sum_{r=0}^{[n/2]}P[T_{0b}=r]\left|\left\{\sum_{s=0}^{[n/2]}P[T_{0b}=s]\frac{(s-r)(1-\theta)}{n+1}\right\}_{+}\right.$$

$$-\left\{\sum_{s=0} P[T_{0b}=s]\frac{(s-r)(1-\theta)}{n+1}\right\}_+ \quad \Big|$$
$$\le \sum_{r=0}^{[n/2]} P[T_{0b}=r]\sum_{s>[n/2]} P[T_{0b}=s]\frac{(s-r)|1-\theta|}{n+1}$$
$$\le |1-\theta| n^{-1}E(T_{0b}1\{T_{0b}>n/2\}) \le 2|1-\theta| n^{-2}ET_{0b}^2, \qquad (1.3)$$

and then by observing that

$$\sum_{r>[n/2]} P[T_{0b}=r]\left\{\sum_{s\ge 0} P[T_{0b}=s]\frac{(s-r)(1-\theta)}{n+1}\right\}$$
$$\le n^{-1}|1-\theta|(ET_{0b}P[T_{0b}>n/2]+E(T_{0b}1\{T_{0b}>n/2\}))$$
$$\le 4|1-\theta| n^{-2}ET_{0b}^2 \qquad (1.4)$$

Combining the contributions of (1.2) –(1.3), we thus find tha

$$\Big|\ d_{TV}(L(\overset{\square}{C}[1,b]), L(\overset{\square}{Z}[1,b]))$$
$$-(n+1)^{-1}\sum_{r\ge 0} P[T_{0b}=r]\left\{\sum_{s\ge 0} P[T_{0b}=s](s-r)(1-\theta)\right\}_+ \quad \Big|$$
$$\le \varepsilon_{\{7.8\}}(n,b)$$
$$=\frac{3}{\theta P_\theta[0,1]}\left\{\varepsilon_{\{10.5(2)\}}(n/2,b)+2n^{-1}ET_{0b}\varepsilon_{\{10.14\}}(n,b)\right\}$$
$$+2n^{-2}ET_{0b}^2\left\{4+3|1-\theta|+\frac{24|1-\theta|\phi^*_{\{10.8\}}(n)}{\theta P_\theta[0,1]}\right\} \qquad (1.5)$$

The quantity $\varepsilon_{\{7.8\}}(n,b)$ is seen to be of the order claimed under Conditions $(A_0),(D_1)$ and (B_{12}), provided that $S(\infty)<\infty$; this supplementary condition can be removed if $\phi^*_{\{10.8\}}(n)$ is replaced by $\phi^*_{\{10.11\}}(n)$ in the definition of $\varepsilon_{\{7.8\}}(n,b)$, has the required order without the restriction on the r_i implied by assuming that $S(\infty)<\infty$. Finally, a direct calculation now shows that

$$\sum_{r\ge 0} P[T_{0b}=r]\left\{\sum_{s\ge 0} P[T_{0b}=s](s-r)(1-\theta)\right\}_+$$
$$=\frac{1}{2}|1-\theta| E|T_{0b}-ET_{0b}|$$

Example 1.0. Consider the point $O=(0,...,0)\in \square^n$. For an arbitrary vector r, the coordinates of the point $x=O+r$ are equal to the respective coordinates of the vector $r: x=(x^1,...x^n)$ and $r=(x^1,...,x^n)$. The vector r such as in the example is called the position vector or the radius vector of the point x. (Or, in greater detail: r is the radius-vector of x w.r.t an origin O). Points are frequently specified by their radius-vectors. This presupposes the choice of O as the “standard origin”. Let us summarize. We have considered $\square^n$ and interpreted its elements in two ways: as points and as vectors. Hence we may say that we leading with the two copies of $\square^n$: $\square^n$= {points}, $\square^n$= {vectors}
Operations with vectors: multiplication by a number, addition. Operations with points and vectors: adding a vector to a point (giving a point), subtracting two points (giving a vector). $\square^n$ treated in this way is called an *n-dimensional affine space*. (An “abstract” affine space is a pair of sets , the set of points and the set of vectors so that the operations as above are defined axiomatically). Notice that vectors in an affine space are also known as “free vectors”. Intuitively, they are not fixed at points and “float freely” in space. From $\square^n$ considered as an affine space we can precede in two opposite directions: $\square^n$ as an Euclidean space $\Leftarrow$ $\square^n$ as an affine space $\Rightarrow$ $\square^n$ as a manifold.Going to the left means introducing some extra structure which will make the geometry richer. Going to the right means forgetting about part of the affine structure; going further in this direction will lead us to the so-called “smooth (or differentiable) manifolds”. The theory of differential forms does not require any extra geometry. So our natural direction is to the right. The Euclidean structure, however, is useful for examples and applications. So let us say a few words about it:

Remark 1.0. *Euclidean geometry.* In $\square^n$ considered as an affine space we can already do a good deal of geometry. For example, we can consider lines and planes, and quadric surfaces like an ellipsoid. However, we cannot discuss such things as “lengths”, “angles” or “areas” and “volumes”. To be able to do so, we have to introduce some more definitions, making $\square^n$ a Euclidean space. Namely, we define the length of a vector $a=(a^1,...,a^n)$ to be

$$|a|:=\sqrt{(a^1)^2+...+(a^n)^2} \qquad (1)$$

After that we can also define distances between points as follows:

$$d(A,B):=\left|\overline{AB}\right| \qquad (2)$$

One can check that the distance so defined possesses natural properties that we expect: is it always non-negative and equals zero only for coinciding points; the distance from A to B is the same as that from B to A (symmetry); also, for three points, A, B and C, we have $d(A,B)\le d(A,C)+d(C,B)$ (the “triangle inequality”). To define angles, we first introduce the scalar product of two vectors

$$(a,b):=a^1b^1+...+a^nb^n \qquad (3)$$

Thus $|a|=\sqrt{(a,a)}$. The scalar product is also denote by dot: $a.b=(a,b)$, and hence is often referred to as the “dot product” . Now, for nonzero vectors, we define the angle between them by the equality

$$\cos\alpha:=\frac{(a,b)}{|a||b|} \qquad (4)$$

The angle itself is defined up to an integral multiple of 2π . For this definition to be consistent we have to ensure that the r.h.s. of (4) does not exceed 1 by the absolute value. This follows from the inequality

$$(a,b)^2\le|a|^2|b|^2 \qquad (5)$$

known as the Cauchy–Bunyakovsky–Schwarz inequality (various combinations of these three names are applied in different books). One of the ways of proving (5) is to consider the scalar square of the linear combination $a+tb$, where $t\in R$. As $(a+tb,a+tb)\ge 0$ is a quadratic polynomial in t which is never negative, its discriminant must be less or equal zero. Writing this explicitly yields (5). The triangle inequality for distances also follows from the inequality (5).

Example 1.1. Consider the function $f(x)=x^i$ (the i-th coordinate). The linear function dx^i (the differential of x^i) applied to an arbitrary vector h is simply h^i .From these examples follows that we can rewrite df as

$$df=\frac{\partial f}{\partial x^1}dx^1+...+\frac{\partial f}{\partial x^n}dx^n, \qquad (1)$$

which is the standard form. Once again: the partial derivatives in (1) are just the coefficients (depending on x); $dx^1,dx^2,...$ are linear functions giving on an arbitrary vector h its coordinates $h^1,h^2,...,$ respectively. Hence

$$df(x)(h)=\partial_{hf(x)}=\frac{\partial f}{\partial x^1}h^1+$$

$$...+\frac{\partial f}{\partial x^n}h^n, \qquad (2)$$

Theorem 1.7. Suppose we have a parametrized curve $t\mapsto x(t)$ passing through $x_0\in\square^n$ at $t=t_0$ and with the velocity vector $x(t_0)=\upsilon$ Then

$$\frac{df(x(t))}{dt}(t_0)=\partial_\upsilon f(x_0)=df(x_0)(\upsilon) \qquad (1)$$

Proof. Indeed, consider a small increment of the parameter $t:t_0\mapsto t_0+\Delta t$, Where $\Delta t\mapsto 0$. On the other hand, we have $f(x_0+h)-f(x_0)=df(x_0)(h)+\beta(h)|h|$ for an arbitrary vector h, where $\beta(h)\to 0$ when $h\to 0$. Combining it together, for the increment of $f(x(t))$ we obtain

$$\begin{aligned}&f(x(t_0+\Delta t)-f(x_0)\\&=df(x_0)(\upsilon.\Delta t+\alpha(\Delta t)\Delta t)\\&+\beta(\upsilon.\Delta t+\alpha(\Delta t)\Delta t).|\upsilon\Delta t+\alpha(\Delta t)\Delta t|\\&=df(x_0)(\upsilon).\Delta t+\gamma(\Delta t)\Delta t\end{aligned}$$

For a certain $\gamma(\Delta t)$ such that $\gamma(\Delta t)\to 0$ when $\Delta t\to 0$ (we used the linearity of $df(x_0)$). By the definition, this means that the derivative of $f(x(t))$ at $t=t_0$ is exactly $df(x_0)(\upsilon)$. The statement of the theorem can be expressed by a simple formula:

$$\frac{df(x(t))}{dt}=\frac{\partial f}{\partial x^1}x^1+\ldots+\frac{\partial f}{\partial x^n}x^n \qquad (2)$$

To calculate the value Of df at a point x_0 on a given vector υ one can take an arbitrary curve passing Through x_0 at t_0 with υ as the velocity vector at t_0 and calculate the usual derivative of $f(x(t))$ at $t=t_0$.

Theorem 1.8. For functions $f,g:U\to\mathbb{R}$, $U\subset\mathbb{R}^n$,

$$d(f+g)=df+dg \qquad (1)$$

$$d(fg)=df.g+f.dg \qquad (2)$$

Proof. Consider an arbitrary point x_0 and an arbitrary vector υ stretching from it. Let a curve $x(t)$ be such that $x(t_0)=x_0$ and $x(t_0)=\upsilon$.

Hence $d(f+g)(x_0)(\upsilon)=\frac{d}{dt}(f(x(t))+g(x(t)))$ at $t=t_0$ and

$$d(fg)(x_0)(\upsilon)=\frac{d}{dt}(f(x(t))g(x(t)))$$

at $t=t_0$ Formulae (1) and (2) then immediately follow from the corresponding formulae for the usual derivative Now, almost without change the theory generalizes to functions taking values in $\mathbb{R}^m$ instead of $\mathbb{R}$. The only difference is that now the differential of a map $F:U\to\mathbb{R}^m$ at a point x will be a linear function taking vectors in $\mathbb{R}^n$ to vectors in $\mathbb{R}^m$ (instead of $\mathbb{R}$). For an arbitrary vector $h\in|\mathbb{R}^n$,

$$\begin{aligned}F(x+h)&=F(x)+dF(x)(h)\\&+\beta(h)|h| \qquad (3)\end{aligned}$$

Where $\beta(h)\to 0$ when $h\to 0$. We have $dF=(dF^1,\ldots,dF^m)$ and

$$dF=\frac{\partial F}{\partial x^1}dx^1+\ldots+\frac{\partial F}{\partial x^n}dx^n$$

$$=\begin{pmatrix}\frac{\partial F^1}{\partial x^1}\cdots\frac{\partial F^1}{\partial x^n}\\ \ldots\quad\ldots\quad\ldots\\ \frac{\partial F^m}{\partial x^1}\cdots\frac{\partial F^m}{\partial x^n}\end{pmatrix}\begin{pmatrix}dx^1\\ \ldots\\ dx^n\end{pmatrix} \qquad (4)$$

In this matrix notation we have to write vectors as vector-columns.

Theorem 1.9. For an arbitrary parametrized curve $x(t)$ in $\mathbb{R}^n$, the differential of a map $F:U\to\mathbb{R}^m$ (where $U\subset\mathbb{R}^n$) maps the velocity vector $x(t)$ to the velocity vector of the curve $F(x(t))$ in $\mathbb{R}^m$:

$$\frac{dF(x(t))}{dt}=dF(x(t))(\dot{x}(t)) \qquad (1)$$

Proof. By the definition of the velocity vector,

$$x(t+\Delta t)=x(t)+\dot{x}(t).\Delta t+\alpha(\Delta t)\Delta t \qquad (2)$$

Where $\alpha(\Delta t)\to 0$ when $\Delta t\to 0$. By the definition of the differential,

$$F(x+h)=F(x)+dF(x)(h)+\beta(h)|h \qquad (3)|$$

Where $\beta(h)\to 0$ when $h\to 0$. we obtain

$$F(x(t+\Delta t)) = F(x+\underbrace{x(t).\Delta t+\dot{\alpha}(\Delta t)\Delta t)}_{h}$$

$$= F(x)+dF(x)(\dot{x}(t)\Delta t+\alpha(\Delta t)\Delta t)+$$

$$\beta(\dot{x}(t)\Delta t+\alpha(\Delta t)\Delta t).\left|x(t)\Delta t+\dot{\alpha}(\Delta t)\Delta t\right|$$

$$= F(x)+dF(x)(\dot{x}(t)\Delta t+\gamma(\Delta t)\Delta t$$

For some $\gamma(\Delta t)\to 0$ when $\Delta t\to 0$. This precisely means that $dF(x)\dot{x}(t)$ is the velocity vector of $F(x)$. As every vector attached to a point can be viewed as the velocity vector of some curve passing through this point, this theorem gives a clear geometric picture of dF as a linear map on vectors.

Theorem 1.10 Suppose we have two maps $F:U\to V$ and $G:V\to W$, where $U\subset\mathbb{R}^n, V\subset\mathbb{R}^m, W\subset\mathbb{R}^p$ (open domains). Let $F:x\mapsto y=F(x)$. Then the differential of the composite map $GoF:U\to W$ is the composition of the differentials of F and G:

$$d(GoF)(x)=dG(y)odF(x) \qquad (4)$$

Proof. We can use the description of the differential .Consider a curve $x(t)$ in $\mathbb{R}^n$ with the velocity vector $\dot{x}$. Basically, we need to know to which vector in $\mathbb{R}^p$ it is taken by $d(GoF)$. the curve $(GoF)(x(t)=G(F(x(t))$. By the same theorem, it equals the image under dG of the Anycast Flow vector to the curve $F(x(t))$ in $\mathbb{R}^m$. Applying the theorem once again, we see that the velocity vector to the curve $F(x(t))$ is the image under dF of the vector $\dot{x}(t)$. Hence $d(GoF)(x)=dG(dF(\dot{x}))$ for an arbitrary vector $\dot{x}$.

Corollary 1.0. If we denote coordinates in $\mathbb{R}^n$ by $(x^1,...,x^n)$ and in $\mathbb{R}^m$ by $(y^1,...,y^m)$, and write

$$dF=\frac{\partial F}{\partial x^1}dx^1+...+\frac{\partial F}{\partial x^n}dx^n \qquad (1)$$

$$dG=\frac{\partial G}{\partial y^1}dy^1+...+\frac{\partial G}{\partial y^n}dy^n, \qquad (2)$$

Then the chain rule can be expressed as follows:

$$d(GoF)=\frac{\partial G}{\partial y^1}dF^1+...+\frac{\partial G}{\partial y^m}dF^m, \qquad (3)$$

Where dF^i are taken from (1). In other words, to get $d(GoF)$ we have to substitute into (2) the expression for $dy^i=dF^i$ from (3). This can also be expressed by the following matrix formula:

$$d(GoF)=\begin{pmatrix}\frac{\partial G^1}{\partial y^1}\cdots\frac{\partial G^1}{\partial y^m}\\ \cdots\ \cdots\ \cdots\\ \frac{\partial G^p}{\partial y^1}\cdots\frac{\partial G^p}{\partial y^m}\end{pmatrix}\begin{pmatrix}\frac{\partial F^1}{\partial x^1}\cdots\frac{\partial F^1}{\partial x^n}\\ \cdots\ \cdots\ \cdots\\ \frac{\partial F^m}{\partial x^1}\cdots\frac{\partial F^m}{\partial x^n}\end{pmatrix}\begin{pmatrix}dx^1\\ \cdots\\ dx^n\end{pmatrix} \qquad (4)$$

i.e., if dG and dF are expressed by matrices of partial derivatives, then $d(GoF)$ is expressed by the product of these matrices. This is often written as

$$\begin{pmatrix}\frac{\partial z^1}{\partial x^1}\cdots\frac{\partial z^1}{\partial x^n}\\ \cdots\ \cdots\ \cdots\\ \frac{\partial z^p}{\partial x^1}\cdots\frac{\partial z^p}{\partial x^n}\end{pmatrix}=\begin{pmatrix}\frac{\partial z^1}{\partial y^1}\cdots\frac{\partial z^1}{\partial y^m}\\ \cdots\ \cdots\ \cdots\\ \frac{\partial z^p}{\partial y^1}\cdots\frac{\partial z^p}{\partial y^m}\end{pmatrix}$$

$$\begin{pmatrix}\frac{\partial y^1}{\partial x^1}\cdots\frac{\partial y^1}{\partial x^n}\\ \cdots\ \cdots\ \cdots\\ \frac{\partial y^m}{\partial x^1}\cdots\frac{\partial y^m}{\partial x^n}\end{pmatrix}, \qquad (5)$$

Or

$$\frac{\partial z^\mu}{\partial x^a}=\sum_{i=1}^{m}\frac{\partial z^\mu}{\partial y^i}\frac{\partial y^i}{\partial x^a}, \qquad (6)$$

Where it is assumed that the dependence of $y \in \mathbb{R}^m$ on $x \in \mathbb{R}^n$ is given by the map F, the dependence of $z \in \mathbb{R}^p$ on $y \in \mathbb{R}^m$ is given by the map G, and the dependence of $z \in \mathbb{R}^p$ on $x \in \mathbb{R}^n$ is given by the composition GoF.

Definition 1.6. Consider an open domain $U \subset \mathbb{R}^n$. Consider also another copy of $\mathbb{R}^n$, denoted for distinction $\mathbb{R}^n_y$, with the standard coordinates $(y^1 ... y^n)$. A system of coordinates in the open domain U is given by a map $F : V \to U$, where $V \subset \mathbb{R}^n_y$ is an open domain of $\mathbb{R}^n_y$, such that the following three conditions are satisfied :

(1) F is smooth;
(2) F is invertible;
(3) $F^{-1} : U \to V$ is also smooth

The coordinates of a point $x \in U$ in this system are the standard coordinates of $F^{-1}(x) \in \mathbb{R}^n_y$

In other words,

$$F : (y^1 ..., y^n) \mapsto x = x(y^1 ..., y^n) \qquad (1)$$

Here the variables $(y^1 ..., y^n)$ are the "new" coordinates of the point x

Example 1.2. Consider a curve in $\mathbb{R}^2$ specified in polar coordinates as

$$x(t) : r = r(t), \varphi = \varphi(t) \qquad (1)$$

We can simply use the chain rule. The map $t \mapsto x(t)$ can be considered as the composition of the maps $t \mapsto (r(t), \varphi(t)), (r, \varphi) \mapsto x(r, \varphi)$. Then, by the chain rule, we have

$$\dot{x} = \frac{dx}{dt} = \frac{\partial x}{\partial r}\frac{dr}{dt} + \frac{\partial x}{\partial \varphi}\frac{d\varphi}{dt} = \frac{\partial x}{\partial r}\dot{r} + \frac{\partial x}{\partial \varphi}\dot{\varphi} \qquad (2)$$

Here $\dot{r}$ and $\dot{\varphi}$ are scalar coefficients depending on t, whence the partial derivatives $\partial x/\partial r, \partial x/\partial \varphi$ are vectors depending on point in $\mathbb{R}^2$. We can compare this with the formula in the "standard" coordinates: $\dot{x} = e_1 \dot{x} + e_2 \dot{y}$. Consider the vectors $\partial x/\partial r, \partial x/\partial \varphi$. Explicitly we have

$$\frac{\partial x}{\partial r} = (\cos\varphi, \sin\varphi) \qquad (3)$$

$$\frac{\partial x}{\partial \varphi} = (-r\sin\varphi, r\cos\varphi) \qquad (4)$$

From where it follows that these vectors make a basis at all points except for the origin (where $r = 0$). It is instructive to sketch a picture, drawing vectors corresponding to a point as starting from that point. Notice that $\partial x/\partial r, \partial x/\partial \varphi$ are, respectively, the velocity vectors for the curves $r \mapsto x(r, \varphi)$ $(\varphi = \varphi_0 \; fixed)$ and $\varphi \mapsto x(r, \varphi)$ $(r = r_0 \; fixed)$. We can conclude that for an arbitrary curve given in polar coordinates the velocity vector will have components $(\dot{r}, \dot{\varphi})$ if as a basis we take $e_r := \partial x/\partial r, e_\varphi := \partial x/\partial \varphi$:

$$\dot{x} = e_r \dot{r} + e_\varphi \dot{\varphi} \qquad (5)$$

A characteristic feature of the basis e_r, e_φ is that it is not "constant" but depends on point. Vectors "stuck to points" when we consider curvilinear coordinates.

Proposition 1.3. The velocity vector has the same appearance in all coordinate systems.

Proof. Follows directly from the chain rule and the transformation law for the basis e_i.In particular, the elements of the basis $e_i = \partial x/\partial x^i$ (originally, a formal notation) can be understood directly as the velocity vectors of the coordinate lines $x^i \mapsto x(x^1, ..., x^n)$ (all coordinates but x^i are fixed). Since we now know how to handle velocities in arbitrary coordinates, the best way to treat the differential of a map $F : \mathbb{R}^n \to \mathbb{R}^m$ is by its action on the velocity vectors. By definition, we set

$$dF(x_0):\frac{dx(t)}{dt}(t_0)\mapsto\frac{dF(x(t))}{dt}(t_0) \qquad (1)$$

Now $dF(x_0)$ is a linear map that takes vectors attached to a point $x_0\in\square^{\,n}$ to vectors attached to the point $F(x)\in\square^{\,m}$

$$dF=\frac{\partial F}{\partial x^1}dx^1+...+\frac{\partial F}{\partial x^n}dx^n$$

$$(e_1,...,e_m)\begin{pmatrix}\frac{\partial F^1}{\partial x^1}\cdots\frac{\partial F^1}{\partial x^n}\\ ...\;\;...\;\;...\\ \frac{\partial F^m}{\partial x^1}\cdots\frac{\partial F^m}{\partial x^n}\end{pmatrix}\begin{pmatrix}dx^1\\ ...\\ dx^n\end{pmatrix}, \qquad (2)$$

In particular, for the differential of a function we always have

$$df=\frac{\partial f}{\partial x^1}dx^1+...+\frac{\partial f}{\partial x^n}dx^n, \qquad (3)$$

Where x^i are arbitrary coordinates. The form of the differential does not change when we perform a change of coordinates.

Example 1.3 Consider a 1-form in $\square^{\,2}$ given in the standard coordinates:

$A=-ydx+xdy$ In the polar coordinates we will have $x=r\cos\varphi, y=r\sin\varphi$, hence

$$dx=\cos\varphi dr-r\sin\varphi d\varphi$$

$$dy=\sin\varphi dr+r\cos\varphi d\varphi$$

Substituting into A, we get

$$A=-r\sin\varphi(\cos\varphi dr-r\sin\varphi d\varphi)$$

$$+r\cos\varphi(\sin\varphi dr+r\cos\varphi d\varphi)$$

$$=r^2(\sin^2\varphi+\cos^2\varphi)d\varphi=r^2d\varphi$$

Hence $A=r^2d\varphi$ is the formula for A in the polar coordinates. In particular, we see that this is again a 1-form, a linear combination of the differentials of coordinates with functions as coefficients. Secondly, in a more conceptual way, we can define a 1-form in a domain U as a linear function on vectors at every point of U:

$$\omega(\upsilon)=\omega_1\upsilon^1+...+\omega_n\upsilon^n, \qquad (1)$$

If $\upsilon=\sum e_i\upsilon^i$, where $e_i={\partial x}/{\partial x^i}$. Recall that the differentials of functions were defined as linear functions on vectors (at every point), and

$$dx^i(e_j)=dx^i\left(\frac{\partial x}{\partial x^j}\right)=\delta^i_j \qquad (2)$$ at every point x.

Theorem 1.9. For arbitrary 1-form ω and path γ, the integral $\int_\gamma\omega$ does not change if we change parametrization of γ provide the orientation remains the same.

Proof: Consider $\left\langle\omega(x(t)),\frac{dx}{dt^{'}}\right\rangle$ and $\left\langle\omega(x(t(t^{'}))),\frac{dx}{dt^{'}}\right\rangle$ As

$$\left\langle\omega(x(t(t^{'}))),\frac{dx}{dt^{'}}\right\rangle=\left|\left\langle\omega(x(t(t^{'}))),\frac{dx}{dt^{'}}\right\rangle.\frac{dt}{dt^{'}},\right.$$

Let p be a rational prime and let $K=\square(\zeta_p)$. We write ζ for ζ_p or this section. Recall that K has degree $\varphi(p)=p-1$ over $\square$. We wish to show that $O_K=\square[\zeta]$. Note that ζ is a root of x^p-1, and thus is an algebraic integer; since O_K is a ring we have that $\square[\zeta]\subseteq O_K$. We give a proof without assuming unique factorization of ideals. We begin with some norm and trace computations. Let j be an integer. If j is not divisible by p, then ζ^j is a primitive p^{th} root of unity, and thus its conjugates are $\zeta,\zeta^2,...,\zeta^{p-1}$. Therefore

$$Tr_{K/\square}(\zeta^j)=\zeta+\zeta^2+...+\zeta^{p-1}=\Phi_p(\zeta)-1=-1$$

If p does divide j, then $\zeta^j = 1$, so it has only the one conjugate 1, and $Tr_{K/\mathbb{Q}}(\zeta^j) = p-1$ By linearity of the trace, we find that

$$Tr_{K/\mathbb{Q}}(1-\zeta) = Tr_{K/\mathbb{Q}}(1-\zeta^2) = \ldots$$
$$= Tr_{K/\mathbb{Q}}(1-\zeta^{p-1}) = p$$

We also need to compute the norm of $1-\zeta$. For this, we use the factorization

$$x^{p-1} + x^{p-2} + \ldots + 1 = \Phi_p(x)$$
$$= (x-\zeta)(x-\zeta^2)\ldots(x-\zeta^{p-1});$$

Plugging in $x=1$ shows that

$$p = (1-\zeta)(1-\zeta^2)\ldots(1-\zeta^{p-1})$$

Since the $(1-\zeta^j)$ are the conjugates of $(1-\zeta)$, this shows that $N_{K/\mathbb{Q}}(1-\zeta) = p$ The key result for determining the ring of integers O_K is the following.

LEMMA 1.9

$$(1-\zeta)O_K \cap \mathbb{Z} = p\mathbb{Z}$$

Proof. We saw above that p is a multiple of $(1-\zeta)$ in O_K, so the inclusion $(1-\zeta)O_K \cap \mathbb{Z} \supseteq p\mathbb{Z}$ is immediate. Suppose now that the inclusion is strict. Since $(1-\zeta)O_K \cap \mathbb{Z}$ is an ideal of $\mathbb{Z}$ containing $p\mathbb{Z}$ and $p\mathbb{Z}$ is a maximal ideal of $\mathbb{Z}$, we must have $(1-\zeta)O_K \cap \mathbb{Z} = \mathbb{Z}$ Thus we can write

$$1 = \alpha(1-\zeta)$$

For some $\alpha \in O_K$. That is, $1-\zeta$ is a unit in O_K.

COROLLARY 1.1 For any $\alpha \in O_K$, $Tr_{K/\mathbb{Q}}((1-\zeta)\alpha) \in p.\mathbb{Z}$

PROOF. We have

$$Tr_{K/\mathbb{Q}}((1-\zeta)\alpha) = \sigma_1((1-\zeta)\alpha) + \ldots + \sigma_{p-1}((1-\zeta)\alpha)$$
$$= \sigma_1(1-\zeta)\sigma_1(\alpha) + \ldots + \sigma_{p-1}(1-\zeta)\sigma_{p-1}(\alpha)$$
$$= (1-\zeta)\sigma_1(\alpha) + \ldots + (1-\zeta^{p-1})\sigma_{p-1}(\alpha)$$

Where the σ_i are the complex embeddings of K (which we are really viewing as automorphisms of K) with the usual ordering. Furthermore, $1-\zeta^j$ is a multiple of $1-\zeta$ in O_K for every $j \neq 0$. Thus $Tr_{K/\mathbb{Q}}(\alpha(1-\zeta)) \in (1-\zeta)O_K$ Since the trace is also a rational integer.

PROPOSITION 1.4 Let p be a prime number and let $K = \mathbb{Q}(\zeta_p)$ be the p^{th} cyclotomic field. Then

$O_K = \mathbb{Z}[\zeta_p] \cong \mathbb{Z}[x]/(\Phi_p(x))$; Thus $1, \zeta_p, \ldots, \zeta_p^{p-2}$ is an integral basis for O_K.

PROOF. Let $\alpha \in O_K$ and write $\alpha = a_0 + a_1\zeta + \ldots + a_{p-2}\zeta^{p-2}$ With $a_i \in \mathbb{Q}$. Then

$$\alpha(1-\zeta) = a_0(1-\zeta) + a_1(\zeta - \zeta^2) + \ldots$$
$$+ a_{p-2}(\zeta^{p-2} - \zeta^{p-1})$$

By the linearity of the trace and our above calculations we find that $Tr_{K/\mathbb{Q}}(\alpha(1-\zeta)) = pa_0$ We also have $Tr_{K/\mathbb{Q}}(\alpha(1-\zeta)) \in p\mathbb{Z}$, so $a_0 \in \mathbb{Z}$ Next consider the algebraic integer $(\alpha - a_0)\zeta^{-1} = a_1 + a_2\zeta + \ldots + a_{p-2}\zeta^{p-3}$; This is an algebraic integer since $\zeta^{-1} = \zeta^{p-1}$ is. The same argument as above shows that $a_1 \in \mathbb{Z}$, and continuing in this way we find that all of the a_i are in $\mathbb{Z}$. This completes the proof.

Example 1.4 Let $K = \mathbb{Q}$, then the local ring $\mathbb{Z}_{(p)}$ is simply the subring of $\mathbb{Q}$ of rational numbers with denominator relatively prime to p. Note that this ring $\mathbb{Z}_{(p)}$ is not the ring $\mathbb{Z}_p$ of p-adic integers; to get $\mathbb{Z}_p$ one must complete $\mathbb{Z}_{(p)}$. The usefulness of $O_{K,p}$ comes from the fact that it has a particularly simple ideal structure. Let a be any proper ideal of $O_{K,p}$ and consider the ideal $a \cap O_K$ of O_K. We claim that

$a = (a \cap O_K)O_{K,p}$; That is, that a is generated by the elements of a in $a \cap O_K$. It is clear from the definition of an ideal that $a \supseteq (a \cap O_K)O_{K,p}$. To prove the other inclusion, let α be any element of a. Then we can write $\alpha = \beta / \gamma$ where $\beta \in O_K$ and $\gamma \notin p$. In particular, $\beta \in a$ (since $\beta / \gamma \in a$ and a is an ideal), so $\beta \in O_K$ and $\gamma \notin p$. so $\beta \in a \cap O_K$. Since $1/\gamma \in O_{K,p}$, this implies that $\alpha = \beta / \gamma \in (a \cap O_K)O_{K,p}$, as claimed.We can use this fact to determine all of the ideals of $O_{K,p}$. Let a be any ideal of $O_{K,p}$ and consider the ideal factorization of $a \cap O_K$ in O_K. write it as $a \cap O_K = p^n b$ For some n and some ideal b, relatively prime to p. we claim first that $bO_{K,p} = O_{K,p}$. We now find that

$a = (a \cap O_K)O_{K,p} = p^n bO_{K,p} = p^n O_{K,p}$ Since $bO_{K,p}$. Thus every ideal of $O_{K,p}$ has the form $p^n O_{K,p}$ for some n; it follows immediately that $O_{K,p}$ is noetherian. It is also now clear that $p^n O_{K,p}$ is the unique non-zero prime ideal in $O_{K,p}$. Furthermore, the inclusion $O_K \mapsto O_{K,p} / pO_{K,p}$ Since $pO_{K,p} \cap O_K = p$, this map is also surjection, since the residue class of $\alpha / \beta \in O_{K,p}$ (with $\alpha \in O_K$ and $\beta \notin p$) is the image of $\alpha\beta^{-1}$ in $O_{K/p}$, which makes sense since β is invertible in $O_{K/p}$. Thus the map is an isomorphism. In particular, it is now abundantly clear that every non-zero prime ideal of $O_{K,p}$ is maximal. To show that $O_{K,p}$ is a Dedekind domain, it remains to show that it is integrally closed in K. So let $\gamma \in K$ be a root of a polynomial with coefficients in $O_{K,p}$; write this polynomial as $x^m + \frac{\alpha_{m-1}}{\beta_{m-1}} x^{m-1} + \ldots + \frac{\alpha_0}{\beta_0}$ With $\alpha_i \in O_K$ and $\beta_i \in O_{K-p}$. Set $\beta = \beta_0 \beta_1 \ldots \beta_{m-1}$. Multiplying by β^m we find that $\beta\gamma$ is the root of a monic polynomial with coefficients in O_K. Thus $\beta\gamma \in O_K$; since $\beta \notin p$, we have $\beta\gamma / \beta = \gamma \in O_{K,p}$. Thus $O_{K,p}$ is integrally close in K.

COROLLARY 1.2. Let K be a number field of degree n and let α be in O_K then $N^{'}_{K/\square}(\alpha O_K) = \left| N_{K/\square}(\alpha) \right|$

PROOF. We assume a bit more Galois theory than usual for this proof. Assume first that $K/\square$ is Galois. Let σ be an element of $Gal(K/\square)$. It is clear that $\sigma(O_K)/\sigma(\alpha) \cong O_{K/\alpha}$; since $\sigma(O_K) = O_K$, this shows that $N^{'}_{K/\square}(\sigma(\alpha)O_K) = N^{'}_{K/\square}(\alpha O_K)$. Taking the product over all $\sigma \in Gal(K/\square)$, we have $N^{'}_{K/\square}(N_{K/\square}(\alpha)O_K) = N^{'}_{K/\square}(\alpha O_K)^n$ Since $N_{K/\square}(\alpha)$ is a rational integer and O_K is a free□ -module of rank n,

$O_K / N_{K/\square}(\alpha)O_K$ Will have order $N_{K/\square}(\alpha)^n$; therefore

$$N^{'}_{K/\square}(N_{K/\square}(\alpha)O_K) = N_{K/\square}(\alpha O_K)^n$$

This completes the proof. In the general case, let L be the Galois closure of K and set $[L:K] = m$.

3. Adaptive Clustering Self Organized Map

The data segmentation can be cast as a constraint satisfaction problem by interpreting the process as one of the assigning labels to data elements based on some feature similarities and subject to certain spatial constraints. This paper presents a Self-organized feature-map network (SOFM) for label assignment based only on feature similarities. The principal goal of the SOFM network developed by Kohonen [4] I to transform an incoming signal of arbitrary dimension into a one or two dimensional discrete map, and to perform this transformation adaptively in a topological order fashion. Many activation patterns are presented to the network, one at a time. Each input causes a corresponding localized group of neuron in the output of the network to be active.

To ensure region connectivity, the clustering process was followed by a 3D connected component labeling algorithm to generate the final regions. This paper presents a novel adaptive clustering self-organized feature map that combines clustering and connected component labeling in one network. Spatial constraints are imposed on the clustering algorithm so that only data elements that are connected to each other are grouped together in a certain class.

The network consists of K X 1 neurons, each representing one feature cluster, therefore the number of neurons is independent of the Data Size . Each neuron $r_i, 1 \leq i \leq K$ is connected to input data elements by a set of synaptic weights $\mathbf{W}_i^{(x,y,z)}$. Each neuron is associated with a collection of sets $\{\omega_{il}\}$ which hold the coordinates of the contiguous data elements that caused the neuron to be activated . A data element $\mathbf{v}^{(x,y,z)}$ is added to a particular set if the data element is spatially connected to the set.

Network Topology

The network consist of $K \times 1$ neurons, each representing one feature cluster, therefore the number of neurons is independent of the entire volume of data . Each neuron $r_i, 1 \leq i \leq K$ is connected to input data elements by a set of synaptic weights $\mathbf{W}_i^{(x,y,z)}$

$$\mathbf{W}_i^{(x,y,z)} = \begin{pmatrix} w_{i1}^{(x,y,z)} & w_{i2}^{(x,y,z)} & \cdot & \cdot & w_{im}^{(x,y,z)} \end{pmatrix}^T$$

$$i = 1, 2, \ldots K.$$

where m represents the number of reduced feature as discussed in the previous section. Every neuron is also associated with a collection of sets ω_{il} which hold the coordinates of contiguous data elements that caused the neuron to be activated. Suppose that at iteration k, data element $\mathbf{v}^{(xyz)}$ represented by feature pattern $\mathbf{Y}^{(x,y,z)}$ is presented to the network.

The input to the ith neuron is calculated as

$$Net_i^k = \|\mathbf{Y}^{(x,y,z)} - \mathbf{W}_i^{(x,y,z)}(k)\|^2$$
$$= \sum_{j=1}^{d} \left(y_j^{(x,y,z)} - w_{ij}^{(x,y,z)}(k)\right)^2$$

The competitive learning rule Winner-take –all (WTA) is used for updating the weights among the neuron. Only the neuron that receives the minimum input would be considered a the winner neuron, r^*, as well as all the weights that lie in a neighborhood $\Omega_{r^*}^{(x,y,z)}(k)$ are pulled into the direction of the input pattern. This gives the network learning rule

$$\mathrm{W}_{r^*}^{(x,y,z)}(k+1) = \mathrm{W}_{r^*}^{(x,y,z)}(k) + \eta^k U_{r^*}^{(x,y,z)}(\mathrm{Y}^{(x,y,z)} - \mathrm{W}_{r^*}^{(x,y,z)}(k))$$

and

$$\mathrm{W}_{r^*}^{(x',y',z')}(k+1) = \mathrm{W}_{r^*}^{(x',y',z')}(k) + (\eta^k)^2(\mathrm{Y}^{(x',y',z')} - \mathrm{W}_{r^*}^{(x',y',z')}(k))$$

$$\forall \mathrm{W}_{r^*}^{(x',y',z')} \in \Omega_{r^*}^{(x,y,z)}(k)$$

where η^k (a number between 0 and 1) is the learning rate at iteration k.

As shown in fig., the topological neighborhood $\Omega_{r^*}^{(x,y,z)}(k)$ is defined as

$$\Omega_{r^*}^{(x,y,z)}(k) = \{\mathbf{W}_{r^*}^{(x',y',z')}(k) : \mathbf{v}^{(x',y',z')} \in \mathcal{S}\}$$

Where $\mathcal{S}$ is a spherical volume of radius R_k centered at $\mathbf{v}^{(x,y,z)}$. The radius R_k is selected fairly wide in the beginning and then permitted to shrink with iteration k .

Topological neighborhood $\Omega_{r^*}^{(x,y,z)}(k)$: . The neighborhood is a spherical volume of radius R_k centered at $\mathbf{v}^{(x,y,z)}$ .As mentioned earlier, each neuron is associated with a collection of sets $\{\omega_{il}\}$ which hold the coordinate of contiguous data elements that caused the neuron to be activated. A data element $\mathbf{v}^{(x,y,z)}$ is added to a particular set if that data element is spatially connected to the set. Each neuron $r_i, 1 \leq i \leq K$, is initially associated with an empty set ω_{i1}. Suppose that data element $\mathbf{v}^{(x,y,z)}$ activated neuron r_i which has $\{\omega_{il} : 1 \leq l \leq i_{max}\}$ sets. We define neighborhood constraint set of $\mathbf{v}^{(x,y,z)}$ denoted by $\mathcal{N}^{(x,y,z)}$

$$\mathcal{N}^{(x,y,z)} = \bigcup_{i=-1}^{i=1} \bigcup_{j=-1}^{j=1} \bigcup_{k=-1}^{k=1} \{\mathbf{v}^{(x-i,y-j,z-k)}\}$$

A Data element $\mathbf{v}^{(x,y,z)}$ is assigned to a region ω_{il} if the following constraint is satisfied

$$\mathcal{N}^{(x,y,z)} \bigcap \omega_{il} \neq \phi$$

If the previous condition fails for every set $\omega_{il}, 1 \leq l \leq i_{max},$ a new set, $\omega_{i(i_{max}+1)}$ is created to hold $\mathbf{v}^{(x,y,z)}$. Therefore, each set contain only spatially connected data elements. After each iteration, if two sets belonging to the same neuron contain neighbor data elements, these sets are merged together.

Network Convergence

The energy function of the proposed Network is always convergent during the network evolution. For the traditional unsupervised competitive learning algorithm.

The network minimizes an energy function $F(\mathbf{W})$ given by

$$F(\mathrm{W}) = \sum_{x=1}^{N}\sum_{y=1}^{N}\sum_{z=1}^{N}\sum_{k=1}^{K}\sum_{j=1}^{k_{max}} U_k^{(x,y,z)} V_j^k \left\|\mathrm{Y}^{(x,y,z)} - \mathrm{W}_k^{(x,y,z)}\right\|^2$$

where k_{max} is the number of sets associated with neuron k and V_j^k is the binary state of the set ω_{kj}

$$V_j^k = \begin{cases} 1 & if\ \mathcal{N}^{(x,y,z)} \bigcap \omega_{kj} \neq \phi. \\ 0 & otherwise \end{cases}$$

Taking the derivative with respect to $\mathbf{W}$ we have

$$\nabla F(\mathbf{W}) = -U_k^{(x,y,z)} V_j^k [\mathbf{Y}^{(x,y,z)} - \mathbf{W}_k^{(x,y,z)}]$$

We can see that the weight updates are in the direction of negative descent of F

$$\mathrm{W}_{r^*}^{(x,y,z)}(k+1) = \mathrm{W}_{r^*}^{(x,y,z)}(k) - \eta^k \nabla F(\mathrm{W})$$

$$= \mathrm{W}_{r^*}^{(x,y,z)}(k) + \eta^k U_{r^*}^{(x,y,z)}(\mathrm{Y}^{(x,y,z)} - \mathrm{W}_{r^*}^{(x,y,z)}(k))$$

Therefore, the energy function is always non-increasing and the network I convergent.

Selecting the number of the neurons

In this section, we present a quantitative method for selecting the optimal number of neurons (Clusters) of the network. As shown in the previous section, the network partitions the volume $\mathcal{I}$ yielding L non-overlapping, connected regions

$$L = \sum_{k=1}^{K} k_{max}$$

The segmentation results depend heavily on the selection of the number of clusters (neurons), K. A criterion for the selection process can be stated as follow:

Select the number of clusters, K, such that the segmentation produces a partition that maximizes the homogeneity within segmented regions and the heterogeneity among different regions.

Scatter matrices [5] is used in discriminant and cluster analysis. Assume that the d-dimensional vectors $\mathbf{Y}^{(x,y,z)}, 1 \leq x, y, z \leq N,$ have been separated into L regions. The patterns in the $k\text{th}$ group, n_k in number, are denoted by the vectors

$\left(\mathbf{Y}_1^{(k)} \quad \mathbf{Y}_2^{(k)} \quad . \quad . \quad \mathbf{Y}_{n_k}^{(k)} \right)^T$. The scatter matrix for the $k\text{th}$ group is given by

$$\mathcal{S}^{(k)} = \sum_{j=1}^{n_k} (\mathbf{Y}_j^{(k)} - \mathbf{m}^{(k)})(\mathbf{Y}_j^{(k)} - \mathbf{m}^{(k)})^T.$$

where $\mathbf{m}^{(k)}$ is the vector of features means for the $k\text{th}$ group

$$\mathbf{m}^k = \left(m_1^{(k)} \quad m_2^{(k)} \quad . \quad . \quad m_d^{(k)} \right)^T$$

and $m_i^{(k)}$ is the mean for the $i\text{th}$ feature for the $k\text{th}$ group

$$m_i^{(k)} = \frac{1}{n_k} \sum_{j=1}^{n_k} y_{ji}^{(k)}.$$

The scatter matrix, S, for the pooled sample is defined as

$$\mathcal{S} = \sum_{k=1}^{L} \sum_{j=1}^{n_k} (\mathbf{Y}_j^{(k)} - \mathbf{m})(\mathbf{Y}_j^{(k)} - \mathbf{m})^T$$

where the pooled mean, m, I given by

$$\mathbf{m} = \frac{1}{N} \sum_{k=1}^{L} n_k m^{(k)}.$$

The within-group scatter matrix , $\mathcal{S}_W$, is defined as the sum of the group scatter matrix

$$\mathcal{S}_W = \sum_{k=1}^{L} \mathcal{S}^{(k)}$$

Finally, the between-group scatter matrix, $\mathcal{S}_B$;, is defined as the scatter matrix for the group means

$$\mathcal{S}_B = \sum_{k=1}^{L} \sum_{j=1}^{n_k} (\mathbf{m}^{(k)} - \mathbf{m})(\mathbf{m}^{(k)} - \mathbf{m})^T$$

Thus, a clustering quality measure, CQ, that maximizes the between – class scatter with respect to the total scatter can be formulate as

$$CQ_K = \frac{tr(\mathcal{S}_B)}{tr(\mathcal{S})}.$$

Where $tr(\cdot)$ is the trace of the matrix. Large values CQ_K suggest compact, well isolated clusters. The algorithm is performed for increasing values of K, starting with a small value (for example K =2), and each time CQ_K is calculated . We select K that produces the maximum CQ_K .

Adaptive Network Algorithm

The essence of the Network algorithm is that it substitutes a simple geometric computation for more detailed properties of the Hebb – like rule and lateral interactions. There are five basic steps involved in the application of the algorithm after initialization, namely, sampling, similarity matching, weight updating, set assignment, and merging. These five steps are repeated until, for a selected number of neurons K , the map formation is complete. These steps are then repeated for different values of K until a maximum value for the clustering quality measure, CQ, is reached.

Computational Complexity

The complexity of the adaptive network is mainly determined by the size m of the reduced feature vector, the number of clusters K, and the volume ire. For an N X N X N volume, each iteration ha an order of complexity $O(mKN^3)$. Thus in order to reduce the complexity we can

1. Reduce the size of the input volume by sampling in each direction to obtain an acceptable representation that preserves the volume properties at a higher scale.
2. Reduce the number of features by the features selection procedure.

Adaptive Clustering Self Organized map Algorithm

To find the single best state sequence, q=($q_1q_2\ldots q_t$), for the given observation sequence O=($o_1o_2\ldots o_T$), we need to define the quantity $\delta_t(i)$ in below eq.

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P[q_1 q_2 \ldots q_{t-1}, q_t = i, o_1 o_1 \ldots o_t \mid \lambda]$$

$\delta_t(i)$ is the highest score along a single path, at time t, which accounts for the first t observations and ends in state i. By induction we have

$$\delta_{t+1}(j) = \max_i [\delta_t(i) a_{ij}] b_j(o_{t+1})$$

$b_j(o_{t+1})$ means. To retrieve the state sequence, we need to keep track of the argument that maximizes $\delta_{t+1}(j)$ in above eq. for each t and j. To store the argument, an array, $\psi_t(j)$ is needed in the algorithm. The Viterbi search is as follows.

1. Initialization

$$\delta_1(i) = \pi_i b_i(o_1) \qquad 1 \le i \le N \text{ (states)}$$

$$\psi_1(i) = 0$$

2. Recursion

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(o_t) \quad 2 \le t \le T, 1 \le j \le N$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] \quad 2 \le t \le T, 1 \le j \le N$$

3. Termination

$$P^* = \max_{1 \le i \le N} [\delta_T(i)] \qquad q_T^* = \arg\max_{1 \le i \le N} [\delta_T(i)]$$

4. Path (state sequence) backtracking

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \qquad t = T-1, T-2, \ldots, 1$$

Step 1 : Select an initial value for the number of neurons K.

Step 2 : Associate each neuron with an empty set $\omega_{i1}, 1 \le i \le K$

Step 3 : Initialize the synaptic weights of the network $\mathbf{W}_i^{(x,y,z)}(0), 1 \le i \le K,$, $1 \le x, y, z \le N,$ to small, different, random numbers at iteration k = 0

Step 4 : Draw a sample $\mathbf{Y}^{(x,y,z)}$ from the input set.

Step 5 : Find the best – matching (winning) neuron r^* at iteration k using the minimum distance Euclidean criterion

$$Net_{r^*}^k = min\{ \|\mathbf{Y}^{(x,y,z)} - \mathbf{W}_i^{(x,y,z)}(k)\|^2 : 1 \le i \le K \}$$

Step 6 : Update the synaptic weight vectors using the update formula

$$\mathbf{W}_{r^*}^{(x,y,z)}(k+1) = \mathbf{W}_{r^*}^{(x,y,z)}(k) + \eta^k U_{r^*}^{(x,y,z)}(\mathbf{Y}^{(x,y,z)} - \mathbf{W}_{r^*}^{(x,y,z)}(k))$$

and

$$\mathbf{W}_{r^*}^{(x' y' z')}(k+1) = \mathbf{W}_{r^*}^{(x',y',z')}(k) + (\eta^k)^2 (\mathbf{Y}^{(x' y' z')} - \mathbf{W}_{r^*}^{(x',y',z')}(k))$$

Step 7 : Assign the input $\mathbf{v}^{(x,y,z)}$ to a neurodal set : if $r*$ has only one empty set than $\mathbf{v}^{(x,y,z)} \in \omega_{r^*1}$ otherwise $\mathbf{v}^{(x,y,z)}$ is assigned to a region ω_{r^*l} if the following constraint is satisfied

$$\mathcal{N}^{(x,y,z)} \bigcap \omega_{r^*l} \neq \phi.$$

If the previous condition fails for every set $\omega_{r^*l}, 1 \le l \le r^*_{max},$ a new set, $\omega_{r^*(r^*_{max}+1)}$ is created to hold $\mathbf{v}^{(x,y,z)}$

Step 8 : Merge Set $\omega_{r^*l}, 1 \le l \le r^*_{max}$ if they are spatially connected

Step 9 : Increment k by 1, goto step 4 , and continue until the synaptic weights $\mathbf{W}_i^{(x,y,z)}$ reach their steady – state values.

Step 10 : Calculate the clustering quality CQ_K . Increment K by 1 , goto step 1 if $K < K_{max}$

Step 11 : Select K that gives max CQ_K

4. Author's affiliation

Akash Kumar Singh is working with IBM Global Services India, Bangalore as a Technical Manager and PhD researcher, Computational Intelligence with Leeds Metropolitan University, UK.

5. References

[1] A. Hyvarinen, "A fast fixed-point algorithm for independent component analysis", Neural computation, 9(7), (1997)

[2] A. Papoulis, "*Probability random variables and stochastic processes*", McGraw-Hill, New York, (1984)

[3] M. Rosenblatt, "*Stationary sequences and random fields*", Birkhauser, Boston, 1985

[4] T. Kohonen, “Self Organization and associative Memory,” Springer-verlag, 1984

[5] K Fukunga “Introduction to statistical pattern recognition,” Academic press, 1990

[6] CHICOCKI, A., and AMARI, S.-I.: ‘Adaptive blind signal and image processing’ (Wiley & Sons, 2002)

[7] 2 HYVA¨ RINEN, A., KARHUNEN, J., and OJA, E.: ‘Independent component analysis’ (John Wiley & Sons, 2001)

[8] FIORI, S.: ‘A theory for learning by weight flow on Stiefel-Grassman manifold’, Neural Comput., 2001, 13, (7), pp. 1625–1647

[9] FIORI, S.: ‘A theory for learning based on rigid bodies dynamics’, IEEE Trans. Neural Network., 2002, 13, (3), pp. 521–531